



Simulation and modelisation of fluid-structure interaction in two-phase flows

Thibaud Beltzung

► To cite this version:

Thibaud Beltzung. Simulation and modelisation of fluid-structure interaction in two-phase flows. Mathematical Physics [math-ph]. Université Paris Saclay (COMUE), 2019. English. NNT : 2019SACLV052 . tel-02500768

HAL Id: tel-02500768

<https://theses.hal.science/tel-02500768>

Submitted on 6 Mar 2020

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

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

THÈSE DE DOCTORAT

de

L'UNIVERSITÉ PARIS-SACLAY

École doctorale de mathématiques Hadamard (EDMH, ED 574)

Établissement d'inscription : Université de Versailles Saint-Quentin-en-Yvelines

Établissement d'accueil : Commissariat à l'énergie atomique et aux énergies alternatives

Laboratoires d'accueil : Laboratoire de mathématiques de Versailles, UMR 8100 CNRS
Laboratoire de modélisation à l'échelle composant, CEA

Spécialité de doctorat : Mathématiques appliquées

Thibaud BELTZUNG

Simulation et modélisation des interactions fluide-structure en
écoulements diphasiques

Date de soutenance : 25 Octobre 2019, à Gif-sur-Yvette

Après avis des rapporteurs : CLÉMENT CANCÈS (INRIA)
JEAN-MARC HÉRARD (EDF)

<i>Jury de soutenance</i> :	CLÉMENT CANCÈS	(INRIA)	Rapporteur
	CHRISTOPHE CHALONS	(UVSQ)	Directeur de thèse
	LAURENT DUMAS	(UVSQ)	Président
	JEAN-MARC HÉRARD	(EDF)	Rapporteur
	MARIA-GIOVANNA RODIO	(CEA)	Examineur
	OLIVIER THOMINE	(CEA)	Codirecteur de thèse

Résumé

Les générateurs de vapeur (GV) sont l'un des composants majeurs des réacteurs nucléaires, et une connaissance approfondie de leur comportement constitue un enjeu industriel important aussi bien pour le concepteur que pour l'exploitant EDF. Une des problématiques rencontrées pour le dimensionnement des GV concerne la vibration des tubes induite par l'écoulement, ce qui nécessite une évaluation raisonnable de la réponse des tubes à l'excitation provoquée par le fluide environnant. La zone identifiée comme la plus critique est la région en U (partie haute du GV) où l'écoulement est diphasique avec un fort taux de vide et interagit plutôt transversalement avec les tubes. Afin d'évaluer les excitations générées sur les tubes par les fluctuations inhérentes à l'écoulement, les paramètres physiques pertinents doivent être identifiés. Pour les écoulements monophasiques, il semble possible de relier les efforts exercés sur les structures tubulaires au niveau de turbulence de l'écoulement ; à la fois en utilisant des méthodes de réduction des données expérimentales mais également en utilisant des méthodes de simulations numériques. Pour les écoulements diphasiques, les forces induites sur les tubes par l'écoulement ont a priori une autre origine et seraient plutôt liées aux contributions dynamiques de chaque phase ainsi qu'aux transferts interfaciaux (fluctuations de pression liées au passage des discontinuités). Néanmoins, les paramètres physiques pertinents qui permettent de prévoir l'amplitude de ces forces restent largement débattus (taux de vide, régime d'écoulement, etc.) et les mécanismes physiques mal compris. Pour étudier ces instabilités vibratoires lorsque l'écoulement est diphasique, un certain nombre d'expériences analytiques ont été et continuent d'être menées au CEA. Ces expériences analytiques portent sur un tube isolé ou en faisceau, rigide ou flexible, et sur une large gamme de régimes d'écoulement (maquettes AMOVI et DIVA du CEA). Leur objectif est de caractériser ces instabilités vibratoires (mesure des forces exercées sur l'obstacle) en fonction de paramètres globaux de l'écoulement (débit gaz, débit liquide, taux de vide "moyen", etc.) mais aussi de certains paramètres locaux (taux de vide local, taille des bulles, vitesse gaz, etc.). Ces paramètres mesurés ou estimés localement sont ceux qui permettent d'obtenir les adimensionnements les plus pertinents à la fois sur les forces d'excitations aléatoires (spectres d'excitation en diphasique sur tube rigide) et sur les forces de couplage fluide-élastiques (tube flexible seul puis en faisceau). Il reste néanmoins une bande de dispersion sur les résultats obtenus, les mécanismes physiques sont mal compris et ces adimensionnements restent tributaires du choix de la localisation des mesures. L'objectif de la thèse est donc de mettre en oeuvre des simulations numériques avec suivi d'interface dans des configurations proches de celles des expériences analytiques menées au CEA afin d'approfondir l'analyse des phénomènes conduisant aux vibrations des tubes de GV.

Mots clés : interaction fluide-structure ; écoulements diphasiques ; méthodes de frontières immergées ; méthodes anti-diffusives ; simulation numérique

Abstract

Steam generators are a key component of nuclear power reactors, and an in-depth knowledge of their mechanisms is a major industrial challenge for the designer and the operator EDF. Vibration of tube bundles induced by cross-flow is one of the problems encountered by the designer, thus needing to assess the vibration response to the excitation generated by the flow. The critical region is the U shape of the bundle (upper part of the steam generator), where two-phase cross-flow occurs with an important void fraction. In order to measure excitation induced by flow fluctuations on the tube bundle, some physical parameters have to be identified. For single-phase flows, it seems possible to link load on tubular structure to turbulence intensity of the flow, thanks to experimental data reduction methods together with numerical simulation methods. For two-phase flows, it is believed that forces induced on the tubes by the flow have other origins, and might be connected to dynamic contribution of each phase together with energy interfacial transfers (pressure fluctuations induced by density discontinuities). Nevertheless, relevant physical parameters which could predict the amplitude of the forces remain a subject of debate (void fraction, flow regime, etc.) and physical processes not yet fully understood. In order to study mechanical instabilities in two-phase flows, some analytic experiments have been and continue to be conducted at CEA. These analytic experiments focus on isolated tube or tube bundles (rigid or flexible), and on a large regime flow range (AMOV and DIVA mockups at CEA). They aim to describe these mechanical instabilities (forces measurement on the obstacle) based upon average parameters of the flow (gas and liquid flow rates, “mean” void fraction, etc.), but also local parameters (local void fraction, bubble size, gas velocity, etc.). These measured or locally estimated parameters are used to conduct relevant nondimensionalization, both on the random excitation forces (two-phase excitation spectrum on a rigid tube) and the fluid-elastic coupling forces (single flexible tube or flexible bundle). Nonetheless, some dispersion remains on the results, physical mechanisms are not well understood, and the nondimensionalization process remains dependent on metrology. The aim of this PhD thesis is to conduct numerical simulations with interface capture methods in configurations close to the experiments conducted at CEA in order to expand the knowledge on phenomena leading to vibration of tube bundle in steam generators.

Keywords: fluid-structure interaction; two-phase flows; immersed boundary methods; anti-diffusive methods; numerical simulation

Contents

1	Introduction	1
1.1	Computing scale and numerical simulation	2
1.1.1	Scope	2
1.1.2	Upscaling process in CFD codes	3
1.1.3	Interest of anti-diffusive methods	4
1.2	Nuclear applications	6
1.2.1	Nuclear power plant	6
1.2.2	Steam generators	7
1.3	Report outline	9
2	Bibliographic study	11
2.1	Multiphase flows	12
2.1.1	Flow regimes	12
2.1.2	Numerical interface	15
2.1.2.1	Level-set	15
2.1.2.2	Volume of Fluid	16
2.1.2.3	Front-Tracking	16
2.1.2.4	Smoothed Particle Hydrodynamics	17
2.1.2.5	Phase-field models	17
2.1.2.6	Anti-diffusive approach	17
2.2	Fluid-Structure interaction	19
2.2.1	Flow around a cylinder	19
2.2.2	Bundle vibration	21
2.2.3	Bundle stability	22
2.2.4	Fluid-Elastic instability	25
2.3	Conclusion	26
3	Modelling and numerical methods	27
3.1	Models	28
3.1.1	Conservation laws	28
3.1.1.1	Conservation of mass	28
3.1.1.2	Conservation of momentum	28

3.1.1.3	Conservation of energy	30
3.1.1.4	Differential form of the governing equations	30
3.1.1.5	Summary of the governing equations	33
3.1.2	Navier-Stokes equations with source terms	34
3.1.3	Interfacial source terms	35
3.1.3.1	Surface tension model	35
3.1.3.2	Derivation of the capillary pressure	36
3.1.3.3	Regularization process	38
3.1.3.4	Multi-scale computing	40
3.1.4	Fluid properties evolution	50
3.1.5	Incompressible, non-dilatable formulation	51
3.1.6	Pressure jump conditions	55
3.1.7	Bundle model and coupling with numerical fluid dynamics	59
3.1.7.1	Introduction to vibration analysis	59
3.1.7.2	Mathematical description	60
3.1.7.3	Multiple degrees of freedom	63
3.2	Discretization	65
3.2.1	Finite Volume Method	65
3.2.2	Numerical boundary conditions	68
3.2.3	Topological source terms	70
3.2.4	Numerical stability conditions for the Navier-Stokes equations	72
3.3	Numerical methods	76
3.3.1	Anti-diffusive approach for conservative scalar transport	76
3.3.1.1	Flux limiter formalism	76
3.3.1.2	Schemes comparisons	80
3.3.2	Higher order convection operator (ENO)	89
3.3.3	Semi-implication of the viscous operator	90
3.3.4	Solving the variable coefficient Poisson equation	92
3.3.4.1	Krylov subspace methods	92
3.3.4.2	Multi-grid methods	97
3.3.4.3	Spectral methods	103
3.3.5	Multi-dimensional splitting	104
3.3.5.1	General splitting methods	104
3.3.5.2	Convective operator	105
3.3.5.3	Diffusion operator	106
3.3.5.4	Bi-dimensional splitting on high order schemes	108
3.3.5.5	Three-dimensional splitting on high order schemes	111
3.3.6	Fluid - solid coupling strategy	114
3.3.6.1	Coupling schemes	114
3.3.6.2	Implicit coupling	114
3.4	Domain decomposition and implementation	118
3.4.1	Conventions and sequencing	120
3.4.2	Performance	122

3.5	Validation	124
3.5.1	Mono-phase flows	124
3.5.2	Laplace pressure	125
3.5.3	Pressure oscillation regularization	126
3.5.4	Mono-phase cross-flow around a cylinder	132
4	Application examples	137
4.1	Two-phase cases	138
4.1.1	Droplet oscillation	138
4.1.2	Parasitic currents	146
4.1.3	Rayleigh-Taylor instability	151
4.1.4	Dam break of incompressible fluids	156
4.1.5	Kelvin-Helmholtz instability	160
4.1.6	Bubble rise in buoyant flow	165
4.1.7	Conclusion	169
4.2	Fluid-Solid cases	170
4.2.1	Vibration of a cylinder in quiescent fluid	170
4.2.2	Multiphase flow around an obstacle	171
4.2.3	Multiple inclusions through a tube bundle	173
4.2.4	Conclusion	178
5	General conclusion and perspectives	179
A	CEMRACS 2016	181
B	Discrete operators	209
	Bibliography	215

List of Figures

1.1	Moore's Law [1].	3
1.2	Spectrum of a homogeneous isotropic turbulence - Energy as a function of TKE	4
1.3	Up-scaling process in CFD applications [8–11].	4
1.4	Pressurized Water Reactor [15].	6
1.5	Flow paths in a steam generator and close-up on the U-bend (region of interest for Fluid-Structure Interaction study) [16].	7
2.1	Multi-hase flows regimes (adapted from [17]).	13
2.1	Multiphase flows regimes (adapted from [17]).	14
2.2	Horizontal flow regime map ($U_{L,G}$: liquid/gas superficial velocities) [19].	15
2.3	Discrete interface localization.	16
2.4	Proposed classification of mono-phase flow regimes around a cylinder [52].	20
2.5	Vibration level of the flexible central tube in a rigid bundle, subject to a cross-flow [58].	23
2.6	Stability map, abscissa: Scruton number Sc , ordinate: reduced velocity Ur [68].	25
3.1	Stresses exerted on a fluid element.	29
3.2	Interfacial source term accounting for capillary effects.	36
3.3	Interfacial force balance (mechanical equilibrium).	37
3.4	Normal section curve C on a surface S	37
3.5	Normals misalignment induced by scheme stiffness and discretization. . .	38
3.6	Typical flow pattern observed in a tube bundle [80].	40
3.7	Loss of information induced by excessive filtering on a self-similar pattern (fractal).	42
3.8	Filter boundary regularization effect.	45
3.9	Density transition and interfacial source term distribution across the secondary phase.	48
3.10	Initial un-regularized field (top-left), post-triangulation (top-right), pass-band smoothing (bottom-left), recursive filter (bottom-right)	49
3.11	Convergence rates for the Taylor problem, using [110].	53
3.12	Convergence rates for the Taylor problem, using [112].	54

3.13	Convergence rates for the Taylor problem, using [111].	54
3.14	Laplace equation (u, f and $r = f - A \cdot u$) $\beta = 1, [u] = 0, [\beta u_n] = 0, u(0) = 0, u(1) = 1$	55
3.15	Laplace equation (u, f and r) $\beta = 1, [u] = 1, [\beta u_n] = 0, u(0) = 0, u(1) = 1$	56
3.16	Laplace equation (u, f and r) $\beta = 1, [u] = 0, [\beta u_n] = 1, u(0) = 0, u(1) = 1$	56
3.17	Poisson equation (u, f, r, β) with two discontinuities.	57
3.18	Bi-dimensional extension of fig. 3.17 (u, f, r, β).	58
3.19	Coupling by stiffness.	63
3.20	Coupling by inertia.	63
3.21	Bi-dimensional grid quantities, and boundary shadow variables.	66
3.22	One-dimensional domain unknowns and shadow variables.	68
3.23	Smoothness regions of the flux limiter function, for an arbitrary local CFL ζ	79
3.24	Common flux limiter profiles.	81
3.25	General methodology from $PDE(s)$ to numerical solution.	82
3.26	1D plot of a bi-dimensional split periodic transport of a sharp discontinuity.	84
3.27	1D plot of a bi-dimensional split periodic transport of a smooth initial condition (gaussian).	85
3.28	1D plot of a bi-dimensional split periodic transport of a piecewise linear profile.	85
3.29	Convergence rates for the upwind, NB , SB , UB , and DL schemes.	87
3.30	Flux balance for the 'left' velocity, reconstruction of slopes.	89
3.31	Stencil for the CN scheme.	90
3.32	Steady state horizontal (blue) and vertical (orange) velocity profiles.	91
3.33	$BiCGSTAB$ residuals for the two-phase problem.	95
3.34	Post $BiCGSTAB$ velocity divergence pattern.	96
3.35	Multi-grid strategy for linear systems.	97
3.36	Post $MG(2)$ velocity divergence pattern.	99
3.37	Preconditioner using multi-grid acceleration.	102
3.38	$t = T / 2$, instant of maximal deformation.	109
3.39	$t = T$, do we recover the initial conditions ?	110
3.40	Slices along the x and y directions.	110
3.41	Streamlines for the deformative test case [181].	111
3.42	Back and forth movement of a droplet in a deformative velocity field.	113
3.43	Fluid - solid codes coupling [185].	115
3.44	Velocities control volumes, with 'left' convention.	120
3.45	Strong-scaling for the bubble rise problem ($MG(2)$ pressure solver).	123
3.46	Vertical and transverse velocity profiles in a rectangular channel.	124
3.47	Pressure surge on a static bubble, induced by capillary forces at the interface.	125

3.48	Stiff interfacial capillary source term.	126
3.49	Direct forcing ($kt := \Delta t, kx := \Delta x$).	128
3.50	Spread forcing.	129
3.51	Smooth regularization.	130
3.52	Piecewise linear regularization.	131
3.53	Vorticity of cross-flow around a cylinder in a channel with inlet, wall and outflow boundaries $t \in [0; 8]$	134
3.54	Pressure drag and lift coefficients (volumetric and surface integration).	135
3.55	Viscous contribution of the drag and lift coefficients.	135
3.56	Drag and lift coefficients of the fluid forces exerted on the fixed cylinder.	136
3.57	Pressure difference between the stagnation point and the cylinder tail (recirculation zone).	136
4.1	Bi-dimensional droplet oscillation (total kinetic energy).	139
4.2	Grid convergence: relative error of the oscillation frequency.	140
4.3	Total mass evolution.	140
4.4	Convected density field (2D gravity free oscillation case) - Contour before and after regularization.	141
4.5	Density (x and z directions).	141
4.6	Regularized density (colour function) for normals computation, using a standard deviation $\sigma = 3$. (x and z directions).	142
4.7	Curvature (x and z directions).	142
4.8	Surface tension (x and z directions).	142
4.9	Pressure (x and z directions).	143
4.10	Pressure gradient (x and z directions).	143
4.11	Fluid divergence pre incompressible solver (x and z directions).	143
4.12	Fluid divergence post incompressible solver (x and z directions).	144
4.13	Viscous decay fit of the 3D droplet oscillation amplitude.	145
4.14	3D oscillation case, convolution based regularization.	147
4.15	Capillary source term shift.	148
4.16	Capillary numbers $Ca = \mathbf{u}_{max}\mu / \sigma$ and total mass.	149
4.17	Air - Water: (1 / 1000) density ratio simulation with capillary terms.	152
4.18	Three dimensional <i>RTI</i> - First computation steps.	153
4.19	Last steps of the <i>RTI</i> simulation.	154
4.20	Dam break of incompressible fluids.	158
4.21	Cat eye patterns observed numerically [232] and experimentally [233].	160
4.22	Interface perturbation.	161
4.23	Development of the Kelvin-Helmholtz instability (first steps).	162
4.23	Development of the Kelvin-Helmholtz instability (last steps).	163
4.24	Density contour, at time $t = 3.3$ [s], for the first test case 160x320.	166
4.25	Rise and terminal velocity of the inclusion (80x80).	167
4.26	Density contour, at time $t = 3$ [s], for the second test case 80x160.	168

4.27 Impact of an inclusion on a static obstacle in channel with densities $\rho \in \{5, 10\}$, and without surface tension.	172
4.28 Multiphase flow through a bundle (centre tube flexible).	174
4.28 Coalescence of multiple inclusions.	175
4.28 Breakup of inclusions on cylinder impact.	176

List of Tables

3.1	Least-squares fit of the spatial discrete filter.	43
3.2	Filter coefficients.	44
3.3	Complexity of linear algebra methods.	92
3.4	Pressure residual norms for a chosen criterion $1 \cdot 10^{-5}$ in the ℓ^2 norm. . .	95
3.5	Properties of the Newmark algorithm as a function of the parameters α and β	117
3.6	Fine grid parameters for the unsteady 2D case.	132
4.1	Numerical parameters used for the droplet oscillation test case.	138
4.2	Numerical parameters used for the parasitic currents test case.	150
4.3	Numerical parameters used for the <i>RTI</i> case.	152
4.4	Numerical parameters used for the dam break test case.	156
4.5	Numerical parameters used for the Kelvin-Helmholtz instability test case.	160
4.6	Numerical parameters of the bubble rise in a channel (first test case). . .	165
4.7	Numerical parameters of the the bubble rise in a channel (second test case).	167
4.8	Fluid, simulation and solid parameters for the bundle case.	173
4.9	Decomposition of the time spent during a single timestep.	173

Symbols - Nomenclature

$U_P = U_\infty \frac{P}{P-D}$	Monophase inter tubes pitch velocity
Δt	Timestep
Δx	Grid spacing, spatial resolution
Γ	Stiff or smeared interface
Ω	Computational domain
α	Void fraction
$\delta = 2\pi\zeta$	Logarithmic decrement
ds	Infinitesimal surface element
dv	Infinitesimal volume element
η	Kolmogorov length scale
$\frac{P}{D}$	Reduced bundle gap, pitch
κ	Interface mean curvature
λ	Thermal conductivity
\mathcal{M}	Mach number
\mathfrak{F}	Fourier transform
\mathfrak{Im}	Imaginary part
$\mathfrak{L}(r)$	Non linear \mathfrak{L} imited flux selector
\mathfrak{Re}	Real part
μ	Dynamic viscosity
ν	Kinematic viscosity
$\partial\Omega$	Computational domain boundary
ρ	Density
σ	Surface tension coefficient
θ	Implicit fraction of a semi-implicit scheme
φ	Numerical flux

ς	CFL number
\mathbf{A}	Advection operator
\mathbf{C}	Capillary operator
\mathbf{D}	Diffusion operator
\mathbf{M}	Matrix, tensor
\mathbf{P}	Inter-level Prolongation (interpolation) operator
\mathbf{R}	Inter-level restriction operator
\mathbf{T}	Stress tensor
\mathbf{g}	Gravitational force
\mathbf{u}	Velocity vector $\begin{pmatrix} u \\ v \\ w \end{pmatrix}$
ζ_s	Reference damping
c	Sound celerity
c_p	Specific heat
c_s	Solid damping
k_s	Solid stiffness
m_s	Solid mass
p	Pressure
t	Time

Adimensional numbers

$Ca = \frac{U\mu}{\sigma}$	Capillary number: measure of viscous forces versus surface tension
$Fr = \frac{c_s}{c_g} = \frac{c_s}{\sqrt{gL}} = \frac{Fr}{Ur}$	Dynamic Froude number, ratio of the solid propagation waves in the solid versus surface waves
$Fr = \frac{U_\infty}{\sqrt{gL}}$	Froude: measure of inertial versus gravitational forces
$Kc = \frac{U_\infty T}{L}$	Keulegan-Carpenter (called “period number”, with T the oscillations period): ratio of drag forces (turbulence) versus the inertial forces
$Kn = \frac{\lambda}{L}$	Knudsen: ratio of the free mean path between particle and the physical length scale
$La = \frac{Re^2}{We} = Oh^{-2} = \frac{\rho L \sigma}{\mu^2}$	Laplace: dynamic of free surface, ratio of the superficial tension and the moment transfer inside the fluid
$Nu = \frac{hL}{\lambda}$	Nusselt: convective over conductive thermal transfer over an interface
$Oh = \frac{\sqrt{We}}{Re} = \frac{\mu}{\sqrt{\rho_f L \sigma}}$	Ohnesorge: characterizes droplet deformation, with the ratio of viscous forces against inertial and capillary forces
$Pr = \frac{\nu}{\lambda/(\rho c_p)}$	Prandtl: momentum versus thermal diffusivity
$Re = \frac{U_\infty L}{\nu}$	Reynolds: ratio of the inertial over viscous forces
$Sc = \frac{m\delta}{\rho_f L^2} = 2\pi m^* \zeta$	Scruton: reduced mass-damping parameter
$Sh = \frac{f_s L}{U_\infty}$	Strouhal: convection characteristic time L / U_∞ over flow fluctuations (detachment)
$St = \frac{f_s L^2}{\nu}$	Stokes: ratio of the solid inertial forces versus the fluid viscous forces
$Ur = \frac{U_\infty}{c_s} = \frac{U_\infty}{f_s L} = \frac{Re}{St}$	reduced velocity: ration of the fluid velocity and the elastic waves in the solid
$We = \frac{\rho_f U_\infty^2 L}{\sigma}$	Weber: inertial versus capillary forces over multiphase interfaces

Mathematical operators - Indices

- Exponents

$[\diamond]$	Quantity jump (discontinuity) across an interface
$\mathcal{O}(n)$	Bachmann-Landau (asymptotic) notation
$\tilde{\diamond}$	Predicted variable - split before a correction step
\diamond^*	Complex conjugate
\diamond^c	Coarse grid quantity
\diamond^f	Fine grid quantity
\diamond_1	Primary phase
\diamond_2	Secondary phase
\diamond_∞	Free stream variable
\diamond_f	Fluid quantity
\diamond_s	Solid quantity
ℓ^p	Discrete set of \mathcal{L}^p
$\frac{D\diamond}{D\diamond}$	Convective derivative of \diamond with respect to \diamond ($= \frac{\partial \diamond}{\partial \diamond} + \mathbf{u} \cdot \diamond$)
$\frac{\partial \diamond}{\partial \diamond}$	Partial derivative of \diamond with respect to \diamond
\mathcal{L}^p	p-summable functional Lebesgue space
$\nabla \cdot \diamond$	Divergence of field \diamond
$\nabla \diamond$	Gradient of field \diamond
$\nabla \times \diamond$	Rotational of field \diamond
$\diamond \cdot \diamond$	Inner (dot) product of two vectors $= \diamond^t \diamond$
$\diamond \otimes \diamond$	Outer product of two vectors $= \diamond \diamond^t$
\diamond	Vector quantity $\diamond = (x_1 \ \cdots \ x_n)^t$
\diamond^t	Transpose of a vector or a tensor
$\bar{\diamond}$	Regularized variable

Abbreviations - Acronyms

<i>ACM</i>	Artificial Compression Method
<i>ALE</i>	Arbitrary Lagrangian-Eulerian
<i>AMR</i>	Adaptive Mesh Refinement
<i>AVB</i>	Anti-Vibrations Bars
<i>BC(s)</i>	Boundary Condition(s)
<i>BM</i>	Beam-Warming
<i>BRGM</i>	Bureau de Recherches Géologiques et Minières French geological survey
<i>BWR</i>	Boiling Water Reactor
<i>BiCGSTAB</i>	BiConjugate Gradient STABilized
<i>CEA</i>	Commissariat à l'énergie atomique et aux énergies alternatives French Alternative Energies and Atomic Energy Commission
<i>CEMRACS</i>	Centre d'Eté Mathématique de Recherche Avancée en Calcul Scientifique
<i>CFL</i>	Courant-Friedrichs-Lewy
<i>CGAL</i>	Computational Geometry Algorithms Library
<i>CG</i>	Conjugate Gradient
<i>CN</i>	Crank-Nicolson
<i>COFFEE</i>	COmplex Flows For Energy and Environment
<i>CSC</i>	Compressed Sparse Column
<i>CSF</i>	Continuum Surface Force
<i>CSR</i>	Compressed Sparse Row
<i>CSS</i>	Continuous Surface Stress
<i>DC</i>	Dead Cell
<i>DFT</i>	Discrete Fourier Transform
<i>DF</i>	Direct Forcing

<i>DL</i>	Després-Lagoutière
<i>DNS</i>	Direct Numerical Simulation
<i>ENO</i>	Essentially Non-Oscillatory
<i>EOS</i>	Equation(s) Of State
<i>FCT</i>	Flux Corrected Transport
<i>FC</i>	Fresh Cell
<i>FD(s)</i>	Finite Difference(s)
<i>FFT</i>	Fast Fourier Transform
<i>FIV</i>	Flow Induced Vibrations
<i>FLOPS</i>	Floating Point Operations per Second
<i>FPE(s)</i>	Floating Point Exception(s)
<i>FSI</i>	Fluid-Structure Interaction
<i>FT</i>	Front-Tracking
<i>FVM</i>	Finite Volume Method
<i>Fortran</i>	FORmula TRANslator
<i>GFM</i>	Ghost Fluid Method
<i>GMRES</i>	GEneralized Minimal RESidual
<i>GS</i>	Gauss-Seidel
<i>HHH</i>	Helmutz-Hodge Decomposition
<i>HIT</i>	Homogeneous Isotropic Turbulence
<i>HOT</i>	Higher Order Terms
<i>HPC</i>	High Performance Computing
<i>IBM</i>	Immersed Boundaries Methods
<i>ICE(s)</i>	Internal Compiler Error(s)
<i>IDR(s)</i>	Induced Dimension Reduction
<i>IF</i>	Inflow
<i>IIR</i>	Infinite Impulse Response
<i>INRIA</i>	Institut National de Recherche en Informatique et en Automatique French Institute for Research in Computer Science and Automation
<i>KHI</i>	Kelvin-Helmholtz Instability
<i>LBM</i>	Lattice Boltzmann methods
<i>LES</i>	Large Eddy Simulation
<i>LF</i>	Lax-Friedrichs
<i>LHS</i>	Left-Hand Side
<i>LJAD</i>	Laboratoire J.A. Dieudonné (Université de Nice)

<i>LSM</i>	Level-Set Method(s)
<i>LTI</i>	Linear Time-Invariant
<i>LW</i>	Lax-Wendroff
<i>MAC</i>	Marker and Cell
<i>MG</i>	Multi-grid
<i>MIV</i>	Movement Induced Vibrations
<i>MMS</i>	Method of Manufactured Solutions
<i>MPI</i>	Message Passing Interface
<i>NB</i>	N-BEE
<i>NSE(s)</i>	Navier-Stokes Equation(s)
<i>NUMA</i>	Non Uniform Memory Access
<i>OF</i>	Outflow
<i>PDE(s)</i>	Partial Differential Equation(s)
<i>PETSc</i>	Portable Extensible Toolkit for Scientific Computation
<i>PIV</i>	Particle Image Velocimetry
<i>PLIC</i>	Piecewise Linear Interface Calculation
<i>PWR</i>	Pressurized Water Reactor
<i>P</i>	Periodic
<i>RANS</i>	Reynolds-Averaged Navier-Stokes
<i>RBGS</i>	Red-Black Gauss-Seidel
<i>RG</i>	ReGularized DF
<i>RHS</i>	Right-Hand Side
<i>RMA</i>	Remote Memory Access
<i>ROM(s)</i>	Reduced Order Model(s)
<i>RTI</i>	Rayleigh-Taylor Instability
<i>SB</i>	Super-BEE
<i>SFO</i>	Spurious Force Oscillations
<i>SOR</i>	Successive Over-Relaxation
<i>SPH</i>	Smoothed particle hydrodynamics
<i>SWS</i>	Symmetrically Weighted Sequential
<i>S</i>	Slip (no adherence)
<i>TGM</i>	Two Grid Method
<i>TIV</i>	Turbulence Induced Vibrations
<i>TKE</i>	Turbulent Kinetic Energy
<i>TRUST</i>	TrioU Software for Thermohydraulics

<i>TVD</i>	Total Variation Diminishing
<i>UB</i>	Ultra-BEE
<i>VIV</i>	Vortex Induced Vibrations
<i>VOF</i>	Volume Of Fluid
<i>WENO</i>	Weighted Essentially Non-Oscillatory
<i>W</i>	Wall (adherence, no-slip)

Chapter 1

Introduction

Cette thèse se décompose en deux parties : premièrement le développement de méthodes numériques pour les écoulements diphasiques, appliqué à la problématique des corps immergés ; puis la seconde partie qui traite de l'application de ces méthodes à des problèmes physiques à visée industrielle, à savoir les générateurs de vapeur de centrales nucléaires. L'attrait principal de cette méthode consiste en un traitement eulérien des inclusions : le fluide diphasique est traité comme un seul champ de densité transporté de manière conservative à l'aide d'un schéma anti-diffusif. Une méthode de prise en compte de la tension de surface entre les deux fluides à toutes les échelles résolues est également proposée. Pour conclure, un exemple de calcul couplé avec fluide et obstacles immergés dans l'écoulement sera exposé.

1.1 Computing scale and numerical simulation

1.1.1 Scope

Modelling a nuclear power plant, from the movement of combustible bars, up to the confinement building at the finest scale is impossible. Even if it was possible, the cost of computing every heat exchange flux, or mechanical displacement would be quite useless, considering neighbouring particles to the surrounding continuum. The sufficient amount of complexity required for modelling is key to engineers. In the end, it is exactly what is required for one to understand physics in a macroscopic way. Hence the question of scale, which is essential for modelling. At the coarsest level is the system scale, where one is able to describe plant level operations. This kind of modelling requires strong assumptions on sub-scales properties. It also assumes inter physics coupling, and makes use of correlations to match experimental results to numerical behaviour laws. The component level follows, when a whole component of a nuclear power plant is modelled, for example heat transfer in a gas heat-exchanger. This kind of computation focuses generally on a few physical variables of interest, using governing laws. At this level, one can use *DNS*ⁱ, for which the conservation laws are discretized as fine as the smallest physical scales. Hybrid approaches on the other hand, use sub-grid modelling in order to enrich the behaviour of numerical codes, while the coarsest scales are resolved. Under the continuum scale lies the particle scale, which is of interest in some partial regions of the nuclear power plant such as neutronics where elementary atomic particle and collisions are computed. This scale is also of interest in certain fluid numerical methods, for example Lattice Boltzmann methods or when considering multiphase flows.

The scope of this research work is to model and compute fluid/structure interactions, at component level (steam generator), targeting experimental regime. The computational cost is a major challenge when using this approach. Since the number of interfaces and structure components is usually of high order — several thousand interfaces, e.g. bubbly flows, evolving around several thousand of tubes —, the cost of computation is currently prohibitive, and is it unlikely, considering the evolution of Moore’s Law fig. 1.1, that a full *DNS* computation of a steam generator will be done in the near future. The scaling of the numerical computations is necessary for physical interpretation, although being time consuming and having the drawback of significantly increasing complexity of the implementation [2]. We thus have to consider the minimal set of experiments in order to be able to construe and analyse physical behaviour leading up to potential vibration instabilities in steam generators. The flow regime based on the continuum velocities, lies around $Re = \mathcal{O}(10^5)$ and is thus expensive to reach. The need for numerical methods which scale with the number of interfaces in multiphase flows and the number of computing units is key to the resolution of these issues. Several approaches have proven interesting results but are inherently limited to small number of inclusions. We seek a method whose implementation is cheap, whilst at the same time being able to capture most of the physical behaviour. Furthermore, the flow regime in steam generators is

i. Direct Numerical Simulation

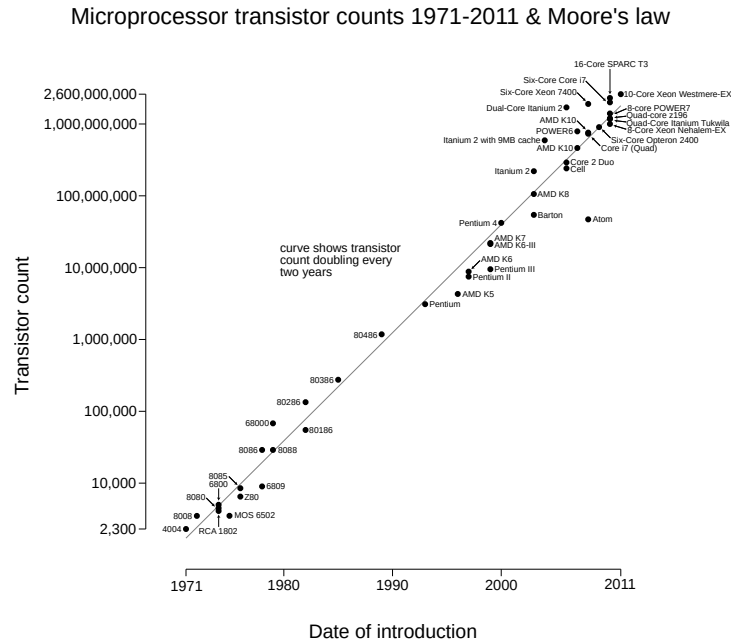


Figure 1.1 – Moore’s Law [1].

still an active subject of study, especially for mixtures involving steam and liquid water instead of water and air.

1.1.2 Upscaling process in CFD codes

Several approaches are available for the resolution of the $NSE(s)$ ⁱⁱ. The Kolmogorov theory [3] states that the large structures are dependent of the configuration of the flow while small structures have a universal nature. The Kolmogorov scales are the spatial, temporal and velocity scales at which the viscous dissipation dominates, as sketched in fig. 1.2. Energy is transferred from the mean flow, giving rise to larger structures, which will dissipate after successive divisions in a cascade manner [4].

Thus, one needs up-scaling process to enrich model-based numerical computations. The up-scaling process on fig. 1.3 depicts information transfer from channel *DNS* computations to bundle flow problem, up to the whole steam generator at the component level, and ending with the power plant where only averaged models can be used. Physical factors are approached empirically, but suffer from non-linearities in the scaling process, statistical uncertainties or measurement errors. As a result, one has to build a confidence index for the results obtained from numerical codes and especially those dependent on empirical conclusions. These factors greatly affect the quality of the quantitative/quali-

ii. Navier-Stokes Equation(s)

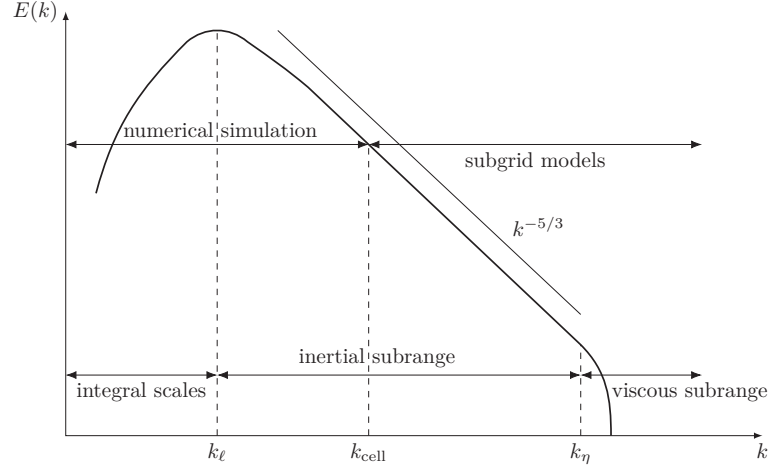


Figure 1.2 – Spectrum of a homogeneous isotropic turbulence - Energy as a function of TKE^{iii} .

tative results. Extra caution is thus required when altering the methodology used or the numerical/experimental chain. An alternative to up-scaling is Reduced Order Model(s), where mathematical complexity is reduced for example by considering only principal directions in a basis. The typical length scales encountered in a nuclear power plant vary by a factor of several orders of magnitude. For example, *DNS* results are usually of the order of 1 [mm] while system results [5] are expressed on a 1 [m] resolution.

At last, results are integrated into a system code. These codes require closure laws [6, 7] since mathematical and/or physical modelling is either not well understood or technical difficulties prevent simulation (balance of the computing costs, time to obtain numerical results, numerical methods drawbacks, ...)

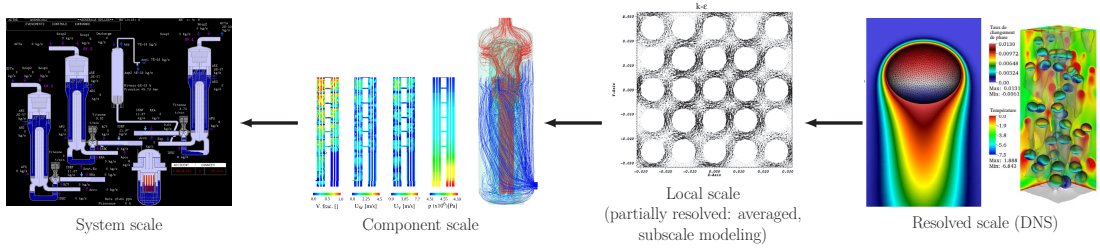


Figure 1.3 – Up-scaling process in CFD applications [8–11].

1.1.3 Interest of anti-diffusive methods

Numerical diffusion of schemes induces a mixing zone at equilibrium, defining a intermediate state between phases. From the discrete point of view, this zone is unavoidable (no numerical scheme is able to rigorously preserve the sharp discontinuity), and im-

plies complications in the modelling scope since arbitrary choices at first glance must be sustained in order to obtain an interpretable result. The interest of such anti-diffusive methods for numerical simulation is above all to formulate the problem with the help of a single eulerian field. Thereby, the scalability of the numerical method is substantially improved, and consequently the performance of the parallel computing code. Indeed, this approach does not require to treat dozens if not hundreds of inclusions in *DNS* in a separate manner (disperse phase [12]). Massive parallel computation becoming more and more accessible on low cost parallel units, development of adapted numerical methods to modern architectures is essential. With the eulerian formulation, computational load is balanced, because the secondary phase is treated on the whole domain. Therefore, all processes in a pool work a fixed amount of time without waiting another. Anti-diffusive methods allow to maintain a fairly reasonable sharp discontinuity over a long period of simulation time. Numerical diffusion inherent to numerical simulation is problematic for interface capture, since the length of the spread zone widens over time, so that it is impossible to distinguish the structures in the surrounding continuous flow in long time. Anti-diffusive schemes allow to conserve the interface discontinuity whilst surface tension allows to conserve coherent structures in the flow. The interface discontinuity has historically been treated in a Lagrangian way, for example in the context of poly-disperse spray flows for the application of combustion [13, 14]. In order to capture correct physics in multiphase flows, it is essential to capture in the finest manner the interface but also the global inclusions structures. In the development of this work, it appeared that the stiffness of the numerical schemes with the diffusion zone was affecting the stability of the method under capillary effect (surface tension). We thus propose a new fast and scalable regularization method for the correct inclusion of source terms, taking into account the inclusion topology at all resolved scales, and allowing to overcome these particular issues.

1.2 Nuclear applications

1.2.1 Nuclear power plant

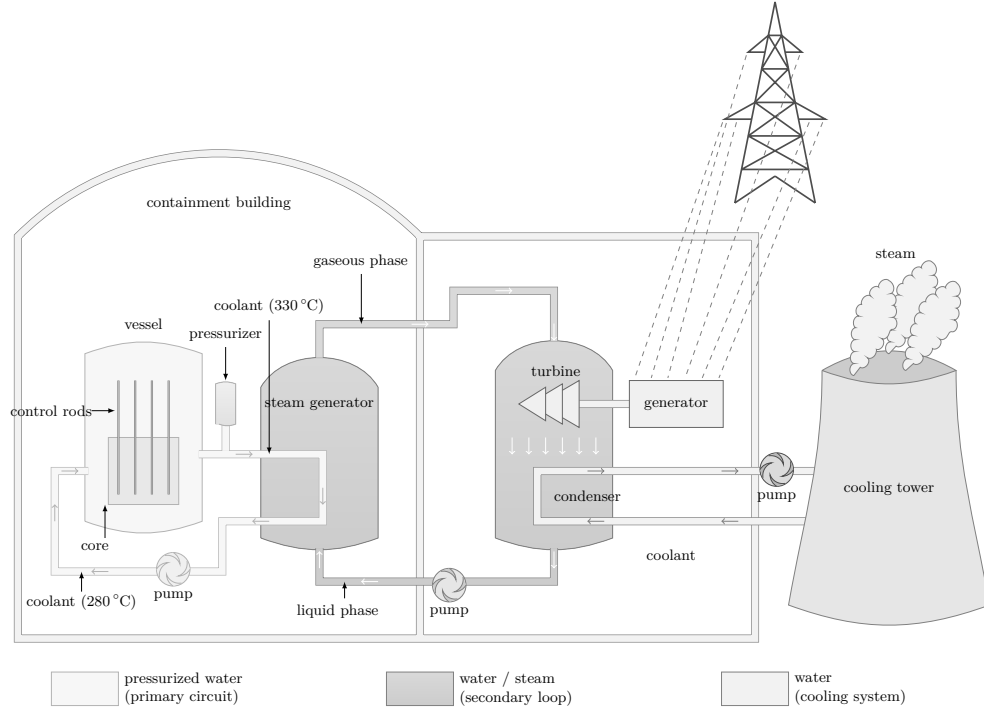


Figure 1.4 – Pressurized Water Reactor [15].

In the context of production of de-carbonated electricity, nuclear power has been ahead of every other technology in the late fifties. A nuclear reaction uses neutrons in order to provoke nuclear fission of an atomic nucleus into smaller and lighter nuclei, whilst liberating a large amount of energy, exploited in terms of heat. A heat-transfer fluid (water), acting as a medium for energy transfer through its change of phase, brings power to a generator connected to the electrical grid.

The fig. 1.4 pictures the layout of a nuclear power plant at the system level. Nuclear fuel in the form of pellets is stacked into fuel assemblies where a bombardment-driven process mutates the pellets' chemical components whilst releasing a large amount of thermal energy. Water is driven around the combustible bundle, allowing heat transfer to occur. Water remains liquid in the primary circuit thanks to the high pressure imposed by the pressurizer. Heat-exchange strictly speaking is conducted by the steam generator where the coolant undergoes phase change (water boiling). Steam output drives a turbo-machine called a turbine, which extracts fluid energy and converts it into mechanical work. The rotating shaft drives in turn an electrical generator outputting a potential of around 20 [kV]. Vapour is re-injected in the circuit after condensation in a cooling tower creating a closed fluid loop.

In the nuclear industry, there is a strong need to predict physical phenomenons, in order to take margins for technical design safety. Time and length scale of these plants give rise to the need of numerical simulation.

1.2.2 Steam generators

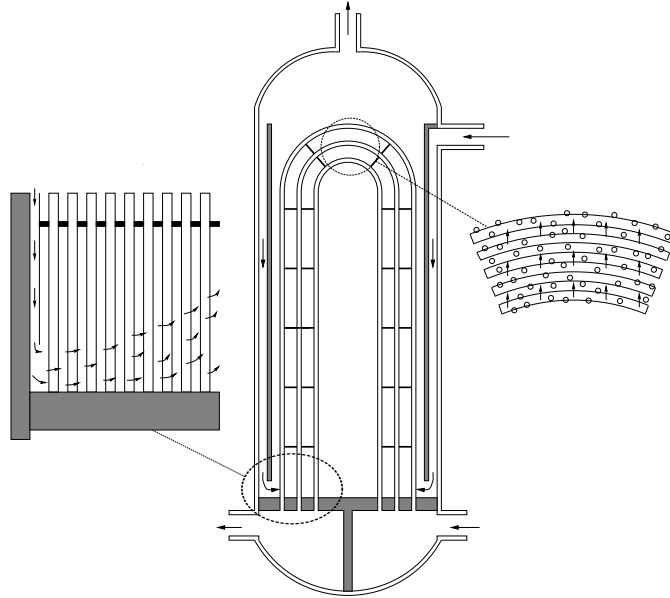


Figure 1.5 – Flow paths in a steam generator and close-up on the U-bend (region of interest for Fluid-Structure Interaction study) [16].

Steam generators are key components of a nuclear power plant, allowing the thermal transfer of the primary fluid to the secondary circuit, along with acting as a safety barrier. A physical barrier is required since the coolant from the primary circuit is slightly radioactive being subject to the neutron fluxes or holding chemical reaction residues from metal corrosion (colloids), as sketched on fig. 1.4. The maximized interfacial exchange surface of this component is made possible through a bundle of three to six thousand two centimetres diameter tubes. This bundle is fitted into a vessel up to a dozen meters high, and around 800 [tons], which can be extracted from a power plant during scheduled costly maintenance phases, that can last for weeks. The primary fluid enters the steam generator at a pressure of 155 [bar] and 325 [°C] in the lower part of the bundle and exits after circulation along the tubes. As opposed to a Boiling Water Reactor, the fluid is kept in a liquid phase in a Pressurized Water Reactor. The fluid from the secondary circuit at an initial mono-phase state — under saturated fluid, where traces of vapour remains — in the lower part of the component begins to flow around the tubes. At this state, the fluid starts to evaporate, due to conductive dominated thermal transfer ongoing at the tube walls. The water-steam mixture thermal properties are now close to 60 [bar] and 275 [°C]. The flow then enters the U-bend section, in which the void

fraction reaches up to 80 [%]. This section is of interest for engineering and design. In the late 1970's, tube denting and corrosion was a major concern since chemistry at the tube walls where thinning the material, leading up to reduced lifetime of the component or degraded efficiency. However, since the improvement of manufacturing processes, thermal treatment, and usage of corrosion resistant alloys, other types of damages are studied in order to improve design margins and maintenance schedules. Mechanical wear, such as tube fretting caused by Flow Induced Vibrations, or fatigue induced vibrations is the context of this work. On current generations, the bundle is fixed in the lower part, while spacer plate between tubes prevent movements in certain directions. Anti-Vibrations Bars also support the bundle in the upper part, in order to attenuate the vibration effect on the tubes themselves, enhancing rigidity of the structure and avoiding micro vibration wear.

Some experimental campaign contribute to the understanding of the coupled physics, but mock-ups are limited in the fluid regime range explored, the nature of fluids used, or the scale observed, even if similarity is widely used in CFD applications. Numerical simulation is thus a valuable asset, allowing to understand the fine scale nature of the aforementioned phenomena observed.

1.3 Report outline

La structure du rapport sera la suivante: le premier chapitre est dédié à l'étude bibliographique des méthodes et approches existantes. La seconde partie regroupe les modèles utilisés (aspect formel du problème, lois de conservation, ...), ainsi que les hypothèses de travail choisies. Des exemples simples unidimensionnels permettent d'étudier chaque phénomène de manière découplée du problème couplé fluide-structure initial (convergence du solveur en pression, termes source topologiques, transport conservatif d'ordre élevé, couplage numérique fluide-structure, régularisation des sauts de pressions induits par les corps immergés). Il s'en suit la discrétisation du problème pour plusieurs phases et de manière dégénérée pour un fluide monophasique. Nous aborderons également dans cette partie l'implémentation effectuée sur un code parallèle créé à cette fin et développé pendant cette thèse. Quelques exemples proposés à titre de validation permettent de s'assurer de la pertinence des méthodes choisies, mais aussi de montrer les limites de la méthodologie proposée.

Le second chapitre sera consacré à l'étude de cas physiques, avec/sans tension de surface, ou bien avec/sans corps immergés. Ce chapitre ouvre des voies à l'exploitation industrielle de la méthode, par exemple reproduire de manière numérique les expériences menées sur des maquettes du *CEA*ⁱ, notamment pour l'étude des interactions fluide-structure dans les générateurs de vapeur.

i. Commissariat à l'énergie atomique et aux énergies alternatives
French Alternative Energies and Atomic Energy Commission

Chapter 2

Bibliographic study

In this chapter, we shall expose the typical flow patterns encountered in hydrodynamics. We will review the numerical methods available as a state of the art for multiphase flows. Then, the immersed boundaries approach will be exposed, together with the problematic of mechanical instabilities.

2.1 Multiphase flows

The ramification of flow regimes encountered in nature, from free surface in a glass of water to large scale hurricanes has been a vast subject of research. Often, fluids are considered as single-phased when densities are subject to a large ratio for example with air and water. In this case the interaction of the gaseous phase is neglected on the liquid state fluid. However, when the densities are similar, we need to consider a two-phase or more approach. How does it differ from single-phase flow, and is it possible to characterize and predict the different flow patterns? In the area of numerical simulation, there is also a strong need for validation. Hence, experimental techniques are important to develop correlations to assess numerical validity range. Often, the measurements are not eased by the multiphase nature of the flow. As with the sodium boiling case (liquid metal), observation or recording with the help of high speed cameras is not feasible. Alternative techniques other than *PIV*ⁱ or hot film probes are needed, for example neutron imaging (tomography).

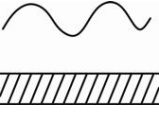
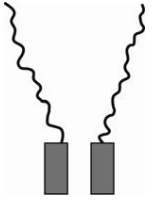
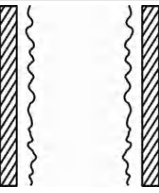
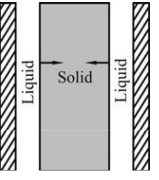
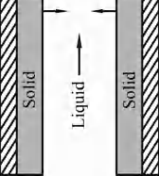
2.1.1 Flow regimes

Multiple flow regimes increases the modelling complexity. On one hand, as seen on fig. 2.1, a slug flow is best represented using *FT*ⁱⁱ or *ALE*ⁱⁱⁱ methods, since a single inclusion travels through narrow channel (such as a pipe). On the other hand, dispersed phase flows with different scales, numerous number of inclusions with small scale such as spray from an atomizer or injector [18] in combustion engines is usually best represented using *SPH*^{iv} methods, where particles with similar dynamical properties are aggregated into larger computing blobs. Melting applications or icing formation requires precise interface positioning in order to compute correct phase changes, and wall films. In the steam generators application, the fluid transitions from bubbly flow at tube boundaries (steam bubbles in a film of water) to droplet flow in the inter-tubes gap (water droplets in a surrounding steam environment). The high density of inclusions together with the obstruction ratio of the tubes make it particularly difficult to achieve numerical simulations in a portion of steam generators. In order to simplify the problem, we will not interest ourselves to phase change (evaporation, solidification, melting described on fig. 2.1) or heat transfers.

A variety of flow regime map exist, based on experimental data. The fig. 2.2 shows the approximative regions for an horizontal multiphase flow, based on the liquid and gas superficial velocities. The annular flow (Ad) is close to vertical flows, with non-uniform film thickness in the vertical direction and sometimes liquid entrainment in the gas core. The dispersed bubbly (DB) occurs when the bubbles separate from the continuum at high velocity rates. With slug flows (I) the bubbles have coalesced to make larger

i. Particle Image Velocimetry
 ii. Front-Tracking
 iii. Arbitrary Lagrangian-Eulerian
 iv. Smoothed particle hydrodynamics

Separated Phases

Regime	Sketch	Example
Phase change on plane surface		Film condensation Film boiling Solidification Melting
Liquid-gas jet flow		Atomization Jet condenser
Liquid-vapor annular flow		Film boiling Film condensation or evaporation
Melting at a single melting point		Melting of ice in a duct
Solidification at a single melting point		Freezing water in a duct

Dispersed Phases

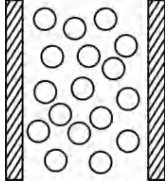
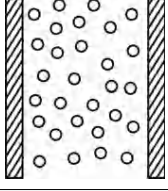
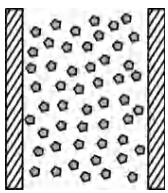
Regime	Sketch	Example
Liquid-vapor bubbly flow (gas)		Chemical reactors Absorbers Evaporators Separating devices
Liquid-vapor droplet flow (gas)		Spray cooling Atomizers Combustors
Particulate flow		Melting, solidification of phase change materials suspension in liquid Combustion of solid fuels Fluidized bed reactors

Figure 2.1 – Multi-phase flows regimes (adapted from [17]).

Mixed Phases

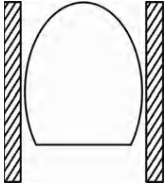
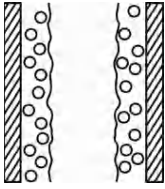
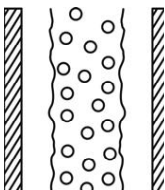
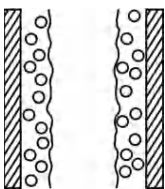
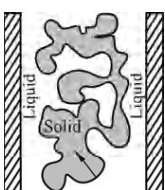
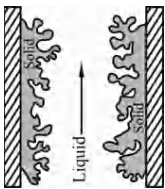
Regime	Sketch	Example
Slug or plug flow		Pulsating heat pipes
Bubbly annular flow		Film evaporation with wall nucleation
Droplet annular flow		Steam generator in boiler
Bubbly droplet annular flow		Boiling nuclear reactor channel
Melting over a temperature range		Melting of binary solid
Solidification over a temperature range		Solidification of binary solution

Figure 2.1 – Multiphase flows regimes (adapted from [17]).

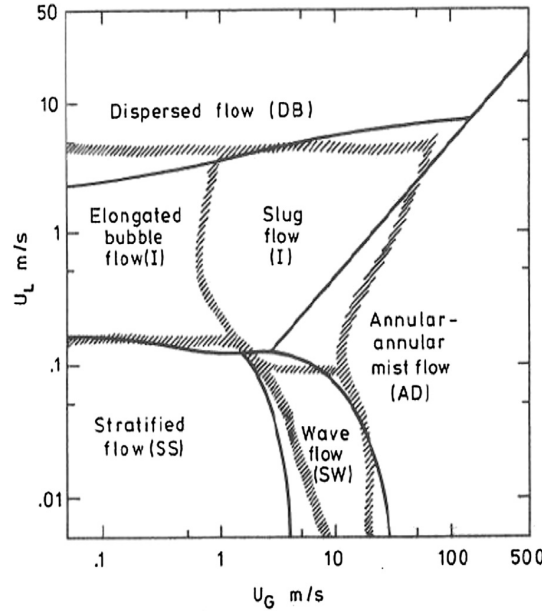


Figure 2.2 – Horizontal flow regime map ($U_{L,G}$: liquid/gas superficial velocities) [19].

bubbles which are close to the diameter of the tube. For elongated bubble flow, large bubbles flow near the top of the tube, affected by buoyancy. At low velocity rates, the gravitational force splits the two-phase into two distinct continuous phase (phase at the bottom and gas at the top of the pipe), in the stratified flow (SS) and (SW) stratified wavy flow. The boundaries between two-phase regimes remains approximate since the exploration of the whole chart is limited (hence the hashed and plain boundaries on the picture). Also, these maps are valid only for specific configurations (horizontal flow in a pipe, vertical flow in a large channel, etc), and are thus not generic.

2.1.2 Numerical interface

The methods presented below try to bring an answer to the non-trivial problem of discrete interface localization as depicted in fig. 2.3, where the true interface $H(x)$ can be mathematically approached by the zero level of the function $F(x)$, or the average value of the smoothed field $I(x)$. We now expose the different approaches for the numerical treatment of the interface in two-phase or multiphase flows.

2.1.2.1 Level-set

The idea behind the level-set method [20–22] is to characterize the interface location implicitly, by the numerical zero of a regular scalar function ϕ , convected on top of a eulerian grid, which compared to other dual mesh methods is computationally cheap.

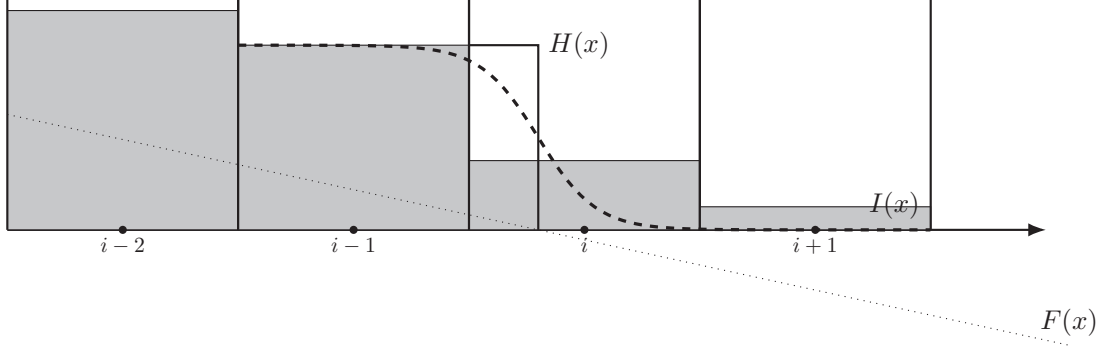


Figure 2.3 – Discrete interface localization.

The interface location can thus be written as $I(t) = \{\mathbf{x} \mid \phi(\mathbf{x}, t) = 0\}$. The Level-Set Method(s) uses a distance function to locate the position of the transition between different fluids: as such, it must be reinitialized frequently in order to keep the distance measure accurate ($\|\nabla\phi(x)\| \simeq 1$) [23]. This redistancing step is costly, reducing the appeal for the method. Furthermore, the method is known to suffer from mass imbalances or particles escape in under-resolved regions, thus requiring consequent computing resources in order to avoid mass losses [24, 25].

2.1.2.2 Volume of Fluid

VOF^v methods make use of a colour function [26] in order to compute the phase volume and subsequently apply conservation laws. A cell containing a mixture is assumed to be cut by a plane representing the local free surface. Hence, in order to find the interface location and the cut plane, one must solve the inverse problem to determine surface out of volumetric information, for example with the $PLIC^{vi}$ method [27]. Normals are estimated with the discrete gradient of the colour function, on which a method for interface reconstruction [28] is applied. VOF methods are conservative, but consequent numerical work must be applied in order to recover the correct interface quantities.

2.1.2.3 Front-Tracking

The most intuitive way of following the phase interface in time and space is to discretize the inclusion using coordinates, and follow their evolution and deformation over time [29, 30]. The FT method depicts an inclusion as a set of points forming a closed surface, which can also be referenced as surface markers convected in a lagrangian fashion [31]. The difficulty resides in the re-meshing process, and the cost of the simulation with multiple inclusions and their interactions (coalescence or break-up). However, these methods benefit from stiff interfacial source terms, thus being less prone to excessive numerical smoothing of the phase discontinuity [32], which can be observed in other

v. Volume Of Fluid

vi. Piecewise Linear Interface Calculation

two-phase numerical methods. Although the method is quite simple to implement on a bi-dimensional space curve, the extension to three dimension is not straightforward. We also mention the *ALE* method [33] which, instead of using markers, deforms an alternate mesh in order to match the inclusion deformation. We also mention that the front tracking results will be used as a reference. With extra difficulty to distribute front properties across multiple processes, the method was implemented only in a serial version and used to roughly validate the cartesian method. Especially, the convergence of the pressure solver on the variable density equation over the thickened interface was of interest, since difficult numerical convergence with preliminary tests was observed.

2.1.2.4 Smoothed Particle Hydrodynamics

The *SPH* method [34, 35] represents fluid medium by an aggregate of particles. The fluid is thus treated in a lagrangian way, usually resulting in a higher resolution for conservation laws, than the methods based on continuum equations, with a trade-off for the computational costs. For example, when solving shocks which are typically of a few mean free paths thick [36], *SPH* method can give accurate results. This mesh-free method simplifies problems on complex domains, even though is usually combined with a classic eulerian method for the another phase, but one must pay attention to the boundary conditions used [37, 38]. Also, even if the aggregate length for particle blobs can be adjusted in terms of physical behaviour and locally, the methods remains costly, and is subject to unbalance of charge between the computing units.

2.1.2.5 Phase-field models

The interface in this method is widen and kept of finite thickness, while being transported thanks to an advection-diffusion equation. The Cahn-Hilliard equation [39, 40] models the diffusion — migration — of species from a chemical point of view. Some authors also interest themselves into conservative versus non-conservative equation of free energy (the chemical potential being the derivative of the free energy) [41–43]. Due to the timescale of the diffusive part, the method is of interest for viscous flows, such as Stokes flows at low $Re \ll 1$.

2.1.2.6 Anti-diffusive approach

The objective of anti-diffusive schemes [44–46] is to maintain sharp discontinuities over a long period of simulation time. These contact discontinuities are usually encountered in compressible fluid dynamics with shock formation where the Euler equations are augmented with jump conditions across states, but also solely travelling waves. For incompressible flows, this approach is of interest for transport phenomenons when numerical smearing has to be avoided to distinguish the different phases over time (transport phenomena). We aim to avoid the artificially compressing approaches for example when using negative diffusion in order to steepen the front, since these methods have a local compressive effect leading to stair-cased solutions [47].

We might also consider alternative methods such as the Artificial Compression Method [48, 49]. In this case, the numerical velocities close to a discontinuous state are modified, resulting in limiting the flux imbalance causing numerical smear. The method however is subject to user tuning: immediate improvement of the accuracy is observed, but the long term solution suffers from error accumulation, and thus excessive diffusion.

In general, high order scheme such as the *WENO*^{vii} scheme [50] are used in order to compute accurate solution with limited truncature errors. However, these schemes do not artificially compress solutions and a regular solution remains smooth. Also, these schemes are more strict for numerical stability conditions. The anti-diffusive method allows backward diffusion, in a sense, to sharpen smooth profiles.

vii. Weighted Essentially Non-Oscillatory

2.2 Fluid-Structure interaction

One of the first examples of numerical simulation of fluid-structure interaction can be found in [51], where the author achieves to reproduce the movement of a heart valve (incompressible blood flow), thanks to a lagrangian description of the immersed boundary. The *FSI*ⁱ phenomenons are recurrent in the building (bridge deck excited by winds), transport (aero-elastic fluttering of aircraft wings), offshore (under-water cables, oil rigs) and nuclear (steam generator) industries. Weak amplitude vibrations can cause fatigue, which propagates through cracks in the material resulting in weakening the whole structure. However, some shorter time-scale phenomenons, more violent, can provoke serious damages. We focus here on transverse flows, which are likely to induce out of control response of the structure.

2.2.1 Flow around a cylinder

Flow around a cylinder is a canonical hydrodynamics example: it has been studied widely and is closely related to experiments from which accurate comparative data can be extracted. With the cylinder movement and its deformation along the transversal axis, the problem also falls within the scope of fluid-structure interaction. As an example, turning the tap on a watering hose carrying an incompressible fluid induces movement of the hose. The instability observed results in large amplitude fluttering of the flexible pipe, originating from complex fluid-structure energy exchanges.

The fig. 2.4 shows the different flow patterns observed experimentally. We now describe the flow behaviour based on the value of the Reynolds number.

- $Re \lesssim 5$: Viscous effect dominates inertial terms. The flow is laminar and follows the cylinder, with a horizontal axis of symmetry: the flow is said to be creeping. The Stokes equations are obtained by linearisation of the $NSE(s)$, and driving Re towards 0.
- $5 \lesssim Re \lesssim 49$: The flow is laminar, and the detachment point evolves upstream of the cylinder together with the velocity of the free stream. Instabilities are damped after the recirculation zone, while two contra-rotating vortices are stable in time. The downstream zone expands as the Reynolds number Re gradually increases.
- $49 \lesssim Re \lesssim 300$: A periodic pattern named Von Karman vortex street appears. Alternate distribution of the pressure of the fluid around the cylinder sheds swirling vortices which, if the detachment frequency matches natural frequency of the flexible tube, makes the system enter into resonance.
- $300 \lesssim Re \lesssim Re_c$: Transition to turbulence phase, under-critical regime. The critical Reynolds number is function of the rugosity of the cylinder and is estimated somewhere between $Re_c \simeq 1 \cdot 10^5$ and $3 \cdot 10^5$.

i. Fluid-Structure Iteration





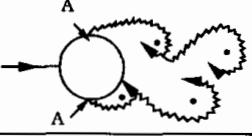
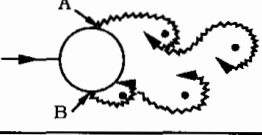
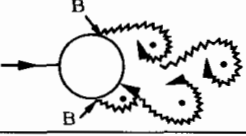
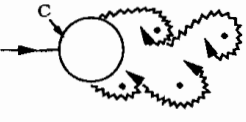

	No separation. Creeping flow	$Re < 5$
	A fixed pair of symmetric vortices	$5 < Re < 40$
	Laminar vortex street	$40 < Re < 200$
	Transition to turbulence in the wake	$200 < Re < 300$
	Wake completely turbulent. A: Laminar boundary layer separation	$300 < Re < 3 \times 10^5$ Subcritical
	A: Laminar boundary layer separation B: Turbulent boundary layer separation; but boundary layer laminar	$3 \times 10^5 < Re < 3.5 \times 10^5$ Critical (Lower transition)
	B: Turbulent boundary layer separation; the boundary layer partly laminar partly turbulent	$3.5 \times 10^5 < Re < 1.5 \times 10^6$ Supercritical
	C: Boundary layer com- pletely turbulent at one side	$1.5 \times 10^6 < Re < 4 \times 10^6$ Upper transition
	C: Boundary layer comple- tely turbulent at two sides	$4 \times 10^6 < Re$ Transcritical

Figure 2.4 – Proposed classification of mono-phase flow regimes around a cylinder [52].

$Re_c \lesssim Re \lesssim 3.5 \cdot 10^6$: The regime loses periodicity because of three-dimensional effects. The shedding spectrum is widened and the wake loses its cohesion. Transition to turbulence happens without separation of the boundary layer in the upper region of the cylinder.

$3.5 \cdot 10^6 \lesssim Re$: The regime reaches super-critical state. The wake recovers some kind of periodic structure, and a steep increase of the fluctuating lift is observed.

2.2.2 Bundle vibration

In the upper part of the steam generator, called U-bend, the tubes are subject to a transverse steam-water flow, with a high void fraction number. Regardless of the flow regime, these tubes are subject to permanent random excitation forces, induced by turbulence and its chaotic nature or by intermittence of interfaces impacting tube boundaries. The induced vibrations level varies with the flow regime. For some set of velocities, fluid-elastic instability is observed experimentally, meaning that a steep transition is observed in the vibration amplitude of the tubes.

Fluid-structure coupling is induced by interaction between the surrounding fluid and the structure in movement. For tube bundle subject to transverse flow, we distinguish three kind of coupling:

- a coupling with an excitation force uncoupled from the proper motion of the obstacles, under the assumption that the unsteadiness of the flow is not modified by the vibrations themselves. We refer to this force as $\mathbf{f}_{rnd}(t)$. For a mono-phase flow, this force is solely driven by turbulence, whereas for multiphase flows random forces stem from turbulence but also intermittence of the steam-water phases;
- a fluid-elastic force, which characterizes the dynamic adaptation of the fluid to the tube movements. Linear analysis of the fluid-elastic coupling allows to write the fluid-elastic force contribution in terms of added mass, damping and stiffness coefficients;
- for mono-phase flows, a third type of coupling can occur, the fluid-structure coupling with a fluid at rest. For this kind of coupling, the fluid initially at rest is put into motion by the vibrations of the immersed solid. Coupling forces are thus function of the movement. This type of coupling is of help in order to validate numerical, since we usually easily impose or measure an analytical movement (e.g. sinusoidal vertical movement), and we are thus able to determine parameters and dependence of the system to input stimuli. It is of help on simple canonical cases, before using fluid-elastic coupling, where the fluid has its own non-linear movement, resulting in a complex behaviour.

Generally, regardless of the flow regime — laminar or turbulent — and the nature of the fluid — mono-phase or multiphase — the effect of the fluid on the solid is modelled as

a single force [53]: the normal component of the fluid strain tensor integrated over the fluid-solid boundary. In order to ease the analysis, most of the authors try to split the fluid to solid force into two components [54]:

- random excitation forces, uncoupled from the tube movements, and whose response is in general of weak amplitude;
- fluid-elastic forces, whose fluctuations depend on the bundle dynamics: in this case, the steady state flow and the structure form a coupled system, and an irreversible energy transfer occurs leading up to potential dynamic instabilities.

The classic approach proposes to describe fluid-elastic forces in the framework of linear analysis of the tube response in terms of mass, stiffness and added damping [16]. The underlying hypothesis is the linearity of the response to the fluid (a work hypothesis since the $NSE(s)$ are highly non-linear). This hypothesis is acceptable as long as the tube vibrations are of small amplitude. Several authors [55], relying on experimental work, show that the effects of the vibration levels on the coefficients of the linearised fluid-elastic forces are negligible for an amplitude of vibration of 70 % of the inter-tube spacing.

Thanks to this modelling, it is theoretically possible to obtain the tube response if the mass, damping and stiffness coefficients, under the assumption that the random excitation is somehow quantified. However, in practice, the main difficulty of this approach is to characterize the dependency of these coefficients and the random fluctuation to the flow parameters. In order to construct a coherent modelling, it is thus necessary to understand the physical mechanisms involved in the *FSI* phenomena in order to isolate the coupling parameters as a first step, and to propose a law for these phenomena as a second step.

Fluid-elastic and random excitation forces vary according to [56, 57]:

- the nature of the fluid (two-phase, mono-phase);
- the type of fluid used (water, steam, ...), or in general the thermo-chemical properties (viscosity, surface tension, volumetric mass ratios, ...);
- the flow regime (laminar or turbulent);
- the flow structure under two-phase flow (bubbly, churn, annular, slug, ...);
- the geometric characteristics of the bundle and the tubes.

2.2.3 Bundle stability

From fig. 2.5 in mono-phase flow, multiple distinct zones appear:

- a random excitation zone with Turbulence Induced Vibrations. The amplitude is limited, and the levels grow with the free stream velocity;

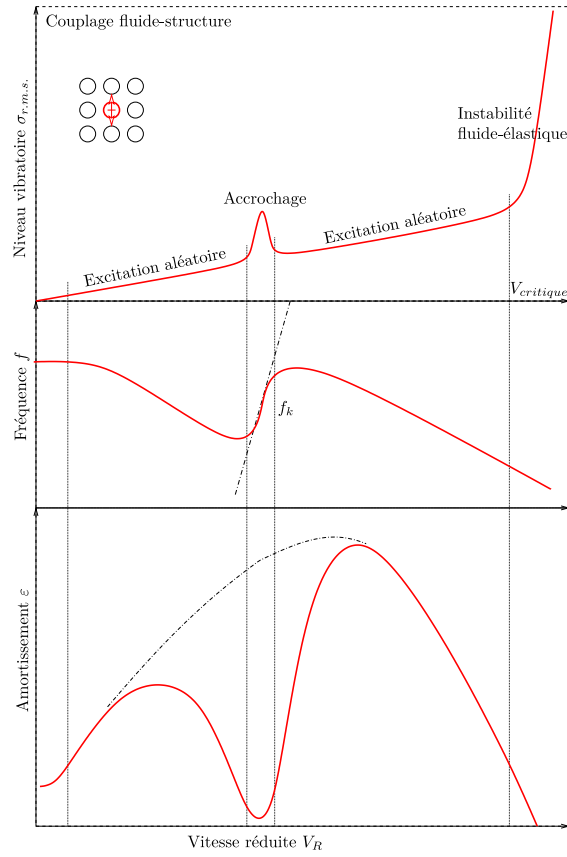


Figure 2.5 – Vibration level of the flexible central tube in a rigid bundle, subject to a cross-flow [58].

- a frequency lock zone with Vortex Induced Vibrations occurs, characterized by a Strouhal number around $Sh = .2$. The phenomenon is similar to resonance, where one of the normal modes of the structure is excited by the unsteady wake. Amplitude levels do rise, but the system remains stable;
- Beyond a certain critical reduced velocity, an aero-elastic coupling called Movement Induced Vibrations is observed. The systems can become unstable, and amplitude of the vibrations diverge until rupture of the structure;
- For small Scruton numbers (reduced mass damping parameter) $Sc \ll 1$, negative damping is at the origin of this instability (damping dominated instability);
- For high Scruton numbers $Sc \gg 1$, displacements provoke the instability by a stiffness coupling between two modes (thus requiring two or more tubes).

In the study of vibration levels of a tube bundle under cross-flow, modal analysis is of great help, allowing to measure physical characteristics non accessible through experiments. The movement of the tubes is projected onto the bending natural modes of the tubes, and the coupled characteristics are described in the modal basis. Some authors have developed semi-analytic models which are able to predict vibration behaviour of the bundle (stability/instability boundary). The most notorious models are:

- Connors-Blevins [59–62];
- Price-Paidoussis [63, 64];
- Lever-Weaver [65, 66].

Let us write the Connors criterion, for a mono-phase flow:

$$Ur = \alpha Sc,$$

where α is a constant dependent on the problem (especially the geometry). This parameter allows to predict empirically the velocity at which the tube is departing into fluid-elastic instability. The parameters allowing nondimensionalization of the vibration spectra are under intense discussion [67], but a consensus seems to emerge on the Scruton number $Sc = m^*\zeta$ as a reduction parameter. The fig. 2.6 shows the Connors map, with a clear boundary between the stable and unstable zone.

The major issue remains to develop a stability criterion for multiphase flows, since the reference damping in the Sc number is by definition the damping of the tube coupled with a fluid at rest, and this definition makes no sense for multiphase flows.

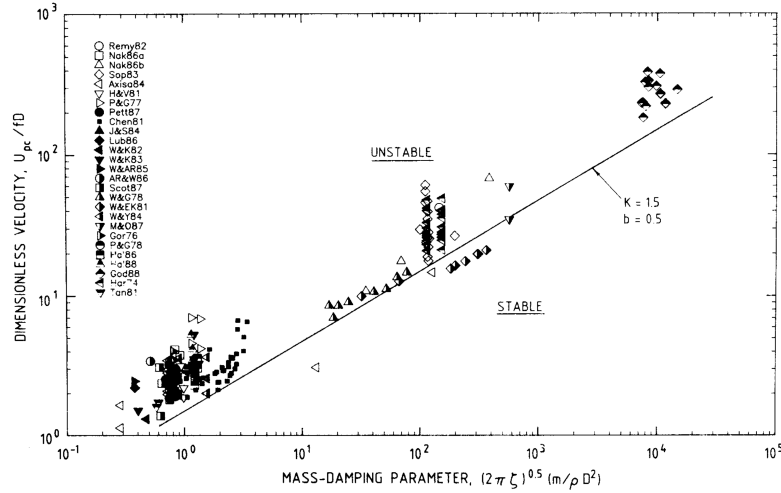


Figure 2.6 – Stability map, abscissa: Scruton number Sc , ordinate: reduced velocity Ur [68].

2.2.4 Fluid-Elastic instability

One of the hypothesis to explain fluid elastic instability is the predominance of similar interfaces hitting a tube bundle.

The choice of the model is based on several assumptions on fluid/solid physical behaviour in steam generators, including:

- multiple inclusions, droplets into liquid or gas pockets into a liquid phase;
- topological changes induced by impact on obstacles, or thermodynamic considerations (phase change), etc.

Fluid-elastic result from an interaction between a moving structure and the mean surrounding flow, in the sense of either spatial, temporal or ensemble averaging. Contrasting with random forces, fluid-elastic forces are coupled with the flow and result in a dynamic adaptation of the fluid to the obstacle movements. Unstable modes can appear if energy is transferred from the flow to the structure. Contrariwise, if the solid loses energy to the flow, the system is damped in a stable manner.

We know some of the mechanisms leading to instability, for instance loss of damping, loss of stiffness, antisymmetric modes; but the main issue is to quantify the dependence of the system mechanical characteristics to the flow or structure parameters.

2.3 Conclusion

Dans ce chapitre, nous avons exposé l'état de l'art des méthodes de simulation numérique pour les écoulements diphasiques, ainsi qu'une courte introduction concernant les phénomènes d'interaction fluide-structure. Les deux problèmes seront traités de manière découplés avant d'envisager de traiter un problème couplé, qui pose des problématiques d'ordre numérique, mais également d'ordre physique pour les discontinuités au voisinage des parois. Le champ d'exploration de la discipline nous oriente vers des exemples simples de simulation, afin d'atteindre ou de tendre vers les conditions d'exploitation des générateurs de vapeurs, à savoir des inclusions multiples dans un environnement confiné à des vitesses fluide élevées. Nous rappelons que l'objectif du code de calcul est de développer une méthode numérique permettant de mesurer les efforts fluide sur un solide à Reynolds élevé et pour un écoulement transverse diphasique. Nous nous intéressons donc dans le chapitre suivant aux hypothèses constitutives des modèles utilisés, ainsi qu'aux méthodes numériques utilisées et/ou développées.

Chapter 3

Modelling and numerical methods

This chapter is dedicated to the development of numerical methods for multiphase flows followed by models and methods for the simulation of interaction between fluid and un-deformable solids.

3.1 Models

In this section, we will derive the conservation laws for an incompressible fluid, with potential extension to heat transfer problem (beyond the scope of this work). These continuous equations will serve as a basis for the discretization and implementation strategy. We consider viscous, incompressible and isothermal flows, since our objective is the observation of mechanical quantities such as load spectrum distribution around a cylinder or oscillation frequencies. Longer scale fluctuations such as thermal exchange by conduction or phase change are not considered throughout this work. Nevertheless, a future addition could include thermal transfers without phase change. We distinguish the multi-fluid formulation [69] versus a multiphase formulation, where each control volume sees several interfaces. In the multi-fluid formulation, the phase in a control volume is either quasi-continuous, quasi-discontinuous or is filled by a mixture around the interface, where resolution is too poor to compensate for numerical diffusion.

3.1.1 Conservation laws

We interest ourselves to the conservation laws of the system. Mass is conserved throughout the system if no internal source or sinks exists. Also, the body forces exerted on a fluid element equal the time change of momentum. Finally, summing up the heat addition and the work done on a fluid volume balances the rate of change of energy. In our application, the fluid is viewed as a continuous medium with macroscopic properties and not as a molecular aggregate.

3.1.1.1 Conservation of mass

We write the mass conservation law for a single control volume. The rate of increase of mass is the net influx of mass in the control volume V .

In its integral form, the equation for the conservation of mass reads:

$$\underbrace{\frac{\partial}{\partial t} \int_V \rho \, dv}_{\text{rate of change of mass}} = - \underbrace{\oint_S \rho \mathbf{u} \cdot \mathbf{n} \, ds}_{\text{net inflow of mass}},$$

where ds is an infinitesimal surface element of the control surface S and dv is an infinitesimal volume element of the control volume V .

3.1.1.2 Conservation of momentum

The rate of increase of momentum stems from the net influx of momentum, the body forces and surface forces:

- momentum per unit volume: $\rho \mathbf{u}$;
- momentum per control volume: $\int_V \rho \mathbf{u} \, dv$.

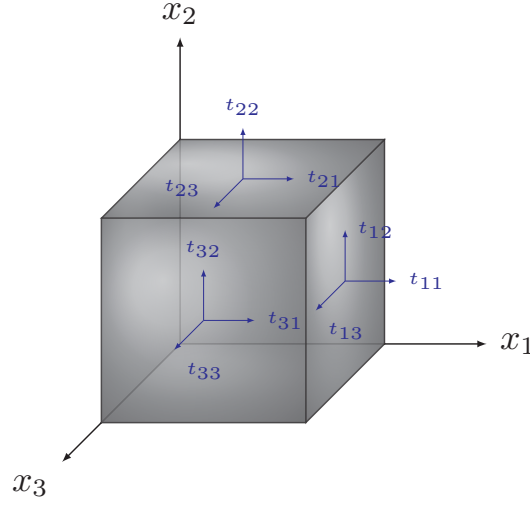


Figure 3.1 – Stresses exerted on a fluid element.

The Cauchy stress tensor over a fluid element is defined as the stress exerted over a fluid element in a deformed state, as seen on fig. 3.1. Stresses over a Newtonian fluid are induced by p^i (volumetric stress $-p\mathbf{I}$) and by viscous stresses (deviatoric stress $\boldsymbol{\tau}$).

The stress tensor \mathbf{T} is written:

$$\mathbf{T} = \left(-p + \lambda \nabla \cdot \mathbf{u} \right) \mathbf{I} + 2\mu \mathbf{D},$$

where \mathbf{I} is the identity tensor, and the rate-of-strain tensor \mathbf{D} is defined as:

$$\mathbf{D} = \frac{1}{2} (\nabla \mathbf{u} + \nabla \mathbf{u}^t),$$

for an isotropic newtonian incompressible fluid.

Gathering the terms results in:

$$\frac{\partial}{\partial t} \int_V \rho \mathbf{u} dv = - \oint_S \rho \mathbf{u} \mathbf{u} ds - \oint_S \mathbf{T} \mathbf{n} ds.$$

If we substitute the stress tensor $\mathbf{T} = -p\mathbf{I} + 2\mu\mathbf{D}$, in the previous equation, we obtain:

$$\underbrace{\frac{\partial}{\partial t} \int_V \rho \mathbf{u} dv}_{\text{rate of change of momentum}} = - \underbrace{\oint_S \rho \mathbf{u} \mathbf{u} \cdot \mathbf{n} ds}_{\text{net inflow of momentum}} - \underbrace{\oint_S p \mathbf{n} ds}_{\text{total pressure}} + \underbrace{\oint_S 2\mu \mathbf{D} \cdot \mathbf{n} ds}_{\text{total viscous force}} + \underbrace{\int_V \rho \mathbf{f} dv}_{\text{total body force}}.$$

i. Pressure

3.1.1.3 Conservation of energy

Conservation of energy is not used in the current fluid model, we expose it however as a basis and perspective for a model evolution. At low Mach numbers, the fluid can be considered as incompressible, and the energy equation can be decoupled from the mass and momentum equations. Consequently, if the problem involves heat transfer, the energy equation can be solved separately. We write the conservation of energy on fluid control volumes:

$$\underbrace{\frac{\partial}{\partial t} \int_V \rho \left(e + \frac{1}{2} \mathbf{u}^2 \right) dv}_{\text{rate of change of kinetic + internal energy}} = - \underbrace{\oint_S \rho \left(e + \frac{1}{2} \mathbf{u}^2 \right) \mathbf{u} \cdot \mathbf{n} ds}_{\text{net inflow of kinetic + internal energy}} + \underbrace{\int_V \mathbf{u} \cdot \mathbf{f} dv}_{\text{work done by body forces}} + \underbrace{\oint_S \mathbf{n} \cdot (\mathbf{u} \mathbf{T}) ds}_{\text{net work done by the stress tensor}} - \underbrace{\oint_S \mathbf{n} \cdot \mathbf{q} ds}_{\text{net heat flux}},$$

where e is the internal energy and \mathbf{q} is the heat flux.

For a conduction dominant heat transfer, the heat flux is expressed as a function of the gradient of temperature:

$$\mathbf{q} = -\lambda \nabla T,$$

where T is the temperature field and λ the thermal conductivity.

3.1.1.4 Differential form of the governing equations

The Divergence/Gauss Theorem is used to convert surface integrals to volume integrals:

$$\int_V \nabla \cdot \mathbf{a} dv = \oint_S \mathbf{a} \cdot \mathbf{n} ds.$$

Mass conservation

We begin by writing the integral form of the mass conservation equation:

$$\frac{\partial}{\partial t} \int_V \rho dv = - \oint_S \rho \mathbf{u} \cdot \mathbf{n} ds. \quad (3.1)$$

Using the Gauss theorem:

$$\oint_S \rho \mathbf{u} \cdot \mathbf{n} ds = \int_V \nabla \cdot (\rho \mathbf{u}) dv.$$

The mass conservation eq. (3.1) becomes:

$$\frac{\partial}{\partial t} \int_V \rho dv = - \int_V \nabla \cdot (\rho \mathbf{u}) dv.$$

Since the control volume is fixed, the derivative can be positioned under the integral sign:

$$\int_V \left(\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) \right) dv = 0.$$

The equation must hold for any control volume, no matter what shape and size. Therefore, the integrand must be equal to zero:

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

Expanding the divergence term reads:

$$\nabla \cdot (\rho \mathbf{u}) = \mathbf{u} \cdot \nabla \rho + \rho \nabla \cdot \mathbf{u}.$$

The mass conservation equation finally becomes:

$$\frac{\partial \rho}{\partial t} + \mathbf{u} \cdot \nabla \rho + \rho \nabla \cdot \mathbf{u} = \frac{D\rho}{Dt} + \rho \nabla \cdot \mathbf{u}.$$

Momentum conservation

The continuity equation for a variable ϕ over a control volume reads:

$$\frac{d}{dt} \int_V \phi dv = - \oint_S \phi \mathbf{u} \cdot \mathbf{n} ds - \int_V \mathbf{s} dv,$$

where \mathbf{s} defines generic sources and sinks (either physical for example with gravitational terms or numerical in order for example to ease convergence).

The Reynolds transport allows to write:

$$\int_V \frac{\partial \phi}{\partial t} dv = - \int_V \nabla \cdot (\phi \mathbf{u}) dv - \int_V \mathbf{s} dv.$$

We now regroup the integrals:

$$\int_V \left(\frac{\partial \phi}{\partial t} + \nabla \cdot (\phi \mathbf{u}) + \mathbf{s} \right) dv = 0,$$

which must hold for any volume.

Using momentum $\rho \mathbf{u}$ as the balanced quantity:

$$\int_V \left(\frac{\partial \rho \mathbf{u}}{\partial t} + \nabla \cdot (\rho \mathbf{u} \otimes \mathbf{u}) + \mathbf{s} \right) dv = 0, \quad (3.2)$$

where \otimes denotes the outer product of two vectors, giving a second order tensor.

The identity expressing the divergence of a dyad product:

$$\nabla \cdot (\mathbf{a} \otimes \mathbf{b}) = (\nabla \cdot \mathbf{a})\mathbf{b} + \mathbf{a} \cdot \nabla \mathbf{b} ,$$

is used to expand the equation eq. (3.2).

After expansion and regrouping the terms:

$$\mathbf{u} \left(\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) \right) + \rho \left(\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} \right) + \mathbf{s} = 0 .$$

The first part is the transport equation for the scalar ρ ⁱⁱ. Under the hypothesis of conservation of mass, the continuity equation of momentum is written:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = 0 . \quad (3.3)$$

The eq. (3.3) will be used in section 3.3.1.2 for the transport of a passive scalar. In our final application, the density will be transported using this method.

The second equation is the particular derivative of the velocity field (derivative of a quantity along the trajectory of a particle with velocity \mathbf{u}):

$$\frac{D\mathbf{u}}{Dt} = \frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} . \quad (3.4)$$

The equation eq. (3.4) links the lagrangian derivative $\frac{D\mathbf{u}}{Dt}$ to the eulerian derivative $\frac{\partial \mathbf{u}}{\partial t}$. The physical meaning of this inertial term is the transport of momentum by the velocity \mathbf{u} .

As a result, the material derivative of \mathbf{u} can be written as:

$$\frac{\partial \mathbf{u}}{\partial t} = -\frac{\mathbf{s}}{\rho} - \frac{\mathbf{u}}{\rho} \left(\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) \right) - \mathbf{u} \cdot \nabla \mathbf{u} .$$

The differential form of the momentum equation is derived in the same way, as follows.

We begin with:

$$\frac{\partial}{\partial t} \int_V \rho \mathbf{u} dv = \int_V \rho \mathbf{f} dv + \oint_S (\mathbf{n} \mathbf{T} - \rho \mathbf{u} (\mathbf{u} \cdot \mathbf{n})) ds .$$

With simplifications, we obtain:

$$\frac{\partial \rho \mathbf{u}}{\partial t} = \rho \mathbf{f} + \nabla \cdot (\mathbf{T} - \rho \mathbf{u} \cdot \mathbf{u}) .$$

The previous expression can be rewritten as

$$\rho \frac{D\mathbf{u}}{Dt} = \rho \mathbf{f} + \nabla \cdot \mathbf{T} .$$

ii. Density

Energy conservation

Although written here, the energy conservation laws will not be used in the models nor in the code, and is left for future evolution of the method.

The mechanical energy equation is obtained by taking the dot product of the momentum equation and the velocity:

$$\mathbf{u} \cdot \left(\rho \frac{\partial \mathbf{u}}{\partial t} + \rho \mathbf{u} \cdot \nabla \mathbf{u} = \rho \mathbf{f} + \nabla \cdot \mathbf{T} \right) .$$

Expanding the previous terms lead to the definition of the mechanical energy:

$$\rho \frac{\partial}{\partial t} \left(\frac{\mathbf{u}^2}{2} \right) = -\rho \mathbf{u} \cdot \nabla \left(\frac{\mathbf{u}^2}{2} \right) + \rho \mathbf{u} \cdot \mathbf{f} + \mathbf{u} \cdot (\nabla \cdot \mathbf{T}) .$$

With the specific energy defined as the sum of internal energy e and kinetic energy $\frac{\mathbf{u}^2}{2}$, the final energy equation in differential form is:

$$\rho \frac{De}{Dt} = \mathbf{T} \cdot \nabla \mathbf{u} + \nabla \cdot \mathbf{q} .$$

Expanding the constitutive relation for the stress tensor and using Fourier's law for the heat flux:

$$\begin{aligned} \mathbf{T} &= (-p + k \nabla \cdot \mathbf{u}) \mathbf{I} + 2\mu \mathbf{D} \cdot \mathbf{D} , \\ \mathbf{q} &= -\lambda \nabla T . \end{aligned}$$

The final result reads:

$$\rho \frac{De}{Dt} + p \nabla \cdot \mathbf{u} = \Phi + \nabla \cdot T , \tag{3.5}$$

where

$$\Phi = \lambda (\nabla \cdot \mathbf{u})^2 + 2\mu \mathbf{D} \cdot \mathbf{D}$$

is the dissipation function and is the rate at which work is converted into heat.

We also need an Equation(s) Of State in the form $p = p(e, \rho)$, $T = T(e, \rho)$, or if we use alternate state variable (T, ρ) : functions for $p = p(T, \rho)$, $e = e(T, \rho)$. Closure laws for μ, λ, k are also mandatory. The model should also account for thermodynamic consistency for the energy conservation law [70].

3.1.1.5 Summary of the governing equations

We finally list the summary of the different fluid governing equations.

Integral form:

$$\begin{aligned}\frac{\partial}{\partial t} \int_V \rho \, dv &= - \oint_S \mathbf{u} \cdot \mathbf{n} \, ds, \\ \frac{\partial}{\partial t} \int_V \rho \mathbf{u} \, dv &= \int_V \rho \mathbf{f} \, dv + \oint_S (\mathbf{n} \mathbf{T} - \rho \mathbf{u} (\mathbf{u} \cdot \mathbf{n})) \, ds, \\ \frac{\partial}{\partial t} \int_V \rho (e + \frac{1}{2} \mathbf{u} \cdot \mathbf{u}) \, dv &= \int_V \mathbf{u} \cdot \rho \mathbf{f} \, dv + \oint_S \mathbf{n} \cdot (\mathbf{u} \mathbf{T} - \rho (e + \frac{1}{2} \mathbf{u}^2) - \mathbf{q}) \, ds.\end{aligned}$$

Conservative form:

$$\begin{aligned}\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) &= 0, \\ \frac{\partial \rho \mathbf{u}}{\partial t} &= \rho \mathbf{f} + \nabla \cdot (\mathbf{T} - \rho \mathbf{u} \otimes \mathbf{u}), \\ \frac{\partial \rho (e + \frac{1}{2} \mathbf{u} \cdot \mathbf{u})}{\partial t} &= \nabla \cdot \left(\rho (e + \frac{1}{2} \mathbf{u}^2) \mathbf{u} - \mathbf{u} \mathbf{T} + \mathbf{q} \right).\end{aligned}$$

Convective form:

$$\begin{aligned}\frac{D\rho}{Dt} + \rho \nabla \cdot \mathbf{u} &= 0, \\ \rho \frac{D\mathbf{u}}{Dt} &= \rho \mathbf{f} + \nabla \cdot \mathbf{T}, \\ \rho \frac{De}{Dt} &= \mathbf{T} \nabla \cdot \mathbf{u} - \nabla \cdot \mathbf{q}.\end{aligned}$$

After establishing the mathematical description for the fluid, we will now restrict ourselves to the modelling of isothermal fluid-structure problems. We will not make use of the equation described in section 3.1.1.4, although these equation might be needed to consider more complex physical cases such as the pulsating bubble with the Rayleigh-Plesset hypothesis [71, 72].

3.1.2 Navier-Stokes equations with source terms

As a first approach, we shall consider isothermal flows. Thus, in our formulation, we do not consider the temperature field in the list of physical quantities, and we do not make use of the corresponding conservation laws. The justification, is that modelling interfacial energy exchanges in terms of mass and heat transfers has a different time scale in this particular industrial application. Furthermore, the phase change aspect requires non trivial modelling, such as evaporation and/or condensation. Finally, we are primarily interested in the frequency analysis and mechanical response from the tube bundle under multiphase flows, and this response can be decoupled from thermal problems.

The Navier-Stokes Equation(s) are derived into the following mono-equation, bi-fluid formulation [73]:

$$\underbrace{\underbrace{\frac{\partial \mathbf{u}}{\partial t}}_{\text{variation}} + \underbrace{\nabla \cdot (\mathbf{u} \otimes \mathbf{u})}_{\text{convection}}}_{\text{inertia}} + \underbrace{\frac{1}{\rho} \nabla p - \frac{1}{\rho} \nabla \cdot \mu (\nabla \mathbf{u} + \nabla^t \mathbf{u})}_{\text{volumetric stress divergence}} = \underbrace{\frac{\mathbf{F}}{\rho}}_{\text{external forces}}, \quad (3.6)$$

where \mathbf{F} encompasses a forcing term for a solid interface, an interfacial source term for a gas/liquid interface (effect of surface tension), or volumetric forces such as gravity. The formulation is mono-field for the two-phase problem, with the phase transition characterized by the density transition in ρ .

3.1.3 Interfacial source terms

We interest ourselves to interfacial source terms, such as an additional diffusive term source located at the interface (presence of species, Stefan flow), or a source term modelling the evaporation or condensation processes (change of phase). In our simplified approach of modelling, we only discuss some momentum source terms which illustrative the regularization effect of surface tension forces exerted at the interface. We note that alternative source terms may apply on momentum, for example diffusion of momentum through the interface, induced by viscosity.

3.1.3.1 Surface tension model

Surface tension is acting on the interface between two fluids. The effect of surface tension is inherent to the inclusion: a surface tension force exists for a droplet of liquid surrounded by vacuum. In our modelling, a kinematic constraint is added to the predicted velocity before solving for the incompressibility of the fluid, either as an explicit acceleration term [74], or by deriving a stress tensor formulation [75], expressed from a continuum point of view. The acceleration term by surface tension is seen on fig. 3.2, where σ^{iii} is the physical surface tension coefficient, κ^{iv} the interface mean curvature, δ the source term spreading and \mathbf{n} the unit normal to the interface. In our approach, since we implicitly describe the interface, the surface tension contribution is spread over a few cells (δ) when penalizing the $NSE(s)$. This approach is transposing mesoscopic interactions (unbalance between attraction forces on neighbouring molecules) onto a smooth macroscopic effect [76]. However, due to the discretization, the interface loses the smooth properties of a n-dimensional manifold. We aim at recovering this smoothness property so that the correct amount of surface tension force is exerted onto the interface, together with the correct normal direction \mathbf{n} .

iii. Surface tension coefficient

iv. Interface mean curvature

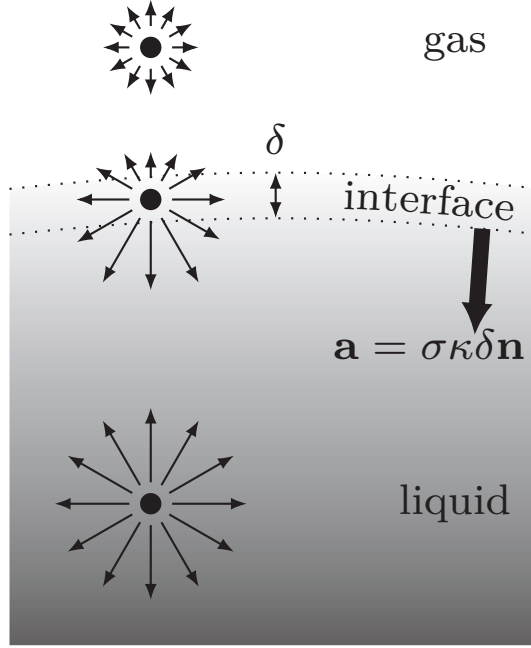


Figure 3.2 – Interfacial source term accounting for capillary effects.

3.1.3.2 Derivation of the capillary pressure

The capillary pressure is the pressure difference across an interface between two fluids. It is function of a thermo-chemical parameter, the surface tension coefficient σ , and the principal curvatures k_{min}, k_{max} [77, 78]:

The Young-Laplace equation is written:

$$\Delta p = \sigma(k_{min} + k_{max}) = \sigma 2\kappa, \quad (3.7)$$

where the choice for κ is discussed later in section 3.2.3.

We consider the non spherical cap sketched in fig. 3.3, where α is the distance between the center of the cap and the node m_1 or m_2 (cap radius). The infinitesimal capillary force applied on the segment dr around the node m_1 projected onto \mathbf{n} :

$$\sigma dr \theta \underset{\lim \theta \rightarrow 0}{=} \sigma dr \sin(\theta) = \sigma dr \frac{\alpha}{r_1}.$$

Summing up the infinitesimal contributions between the four nodes $m_{1..4}$ results in:

$$d\mathbf{f} = 2\alpha\sigma dr \left(\frac{1}{r_1} + \frac{1}{r_2} \right). \quad (3.8)$$

The normal section is the intersection of the plane determined by \mathbf{n}, \mathbf{t} and the surface S , which determines a curve C on the surface S , as shown on fig. 3.4.

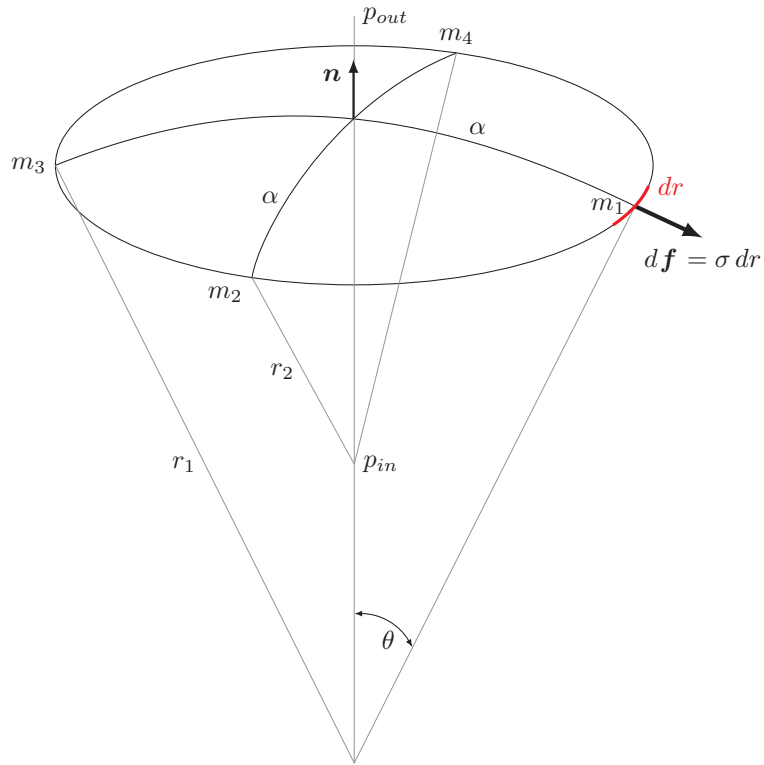
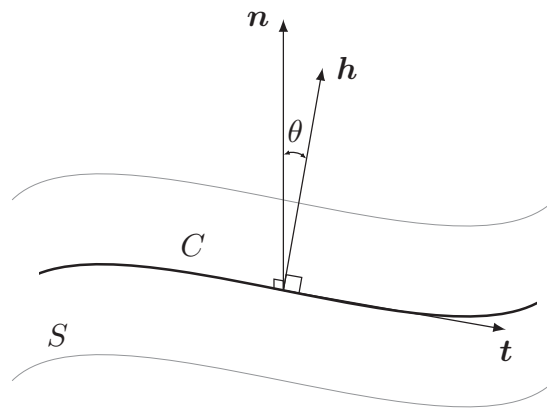


Figure 3.3 – Interfacial force balance (mechanical equilibrium).

Figure 3.4 – Normal section curve C on a surface S .

Since the curvatures of two perpendicular normal sections sums up to $k_{min} + k_{max}$, [79], the integration of dr over the circumference is independent of the positioning of the nodes m_1 and m_2 .

By integration of eq. (3.8):

$$\mathbf{F} = \int_{m_1}^{m_2} d\mathbf{f} \, dr = 2\alpha\sigma \left(\frac{1}{r_1} + \frac{1}{r_2} \right) \int_{m_1}^{m_2} dr = 2\alpha^2\sigma \left(\frac{1}{r_1} + \frac{1}{r_2} \right) \frac{\pi}{2}. \quad (3.9)$$

The force resulting in the pressure difference across the surface is thus:

$$\mathbf{F} = (p_{in} - p_{out})\pi\alpha^2.$$

Combining eq. (3.9) and the previous equation yields the Young-Laplace equation eq. (3.7). The Young-Laplace equation can also be found when simplifying the Rayleigh-Plesset equation in the static case (without radius perturbation).

In this formulation, since the stresses are continuous across the interface, we have:

$$[-p + \tau]\mathbf{n} = -\sigma\kappa\mathbf{n}.$$

3.1.3.3 Regularization process

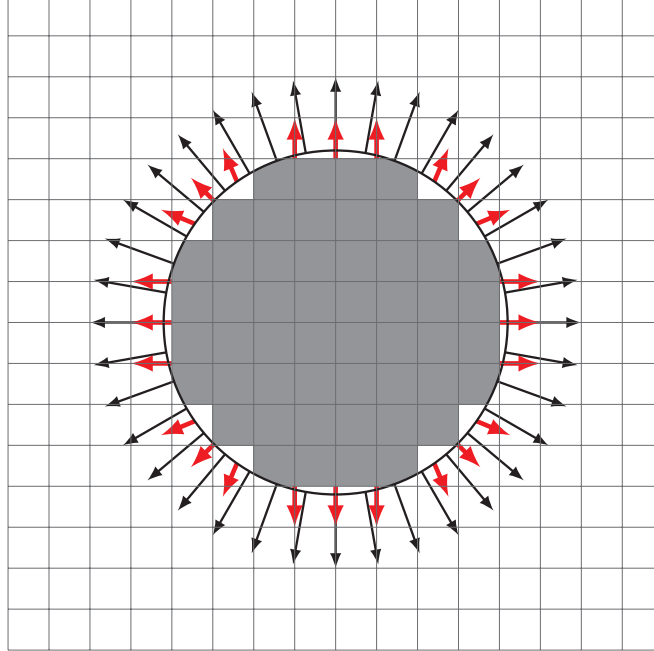


Figure 3.5 – Normals misalignment induced by scheme stiffness and discretization.

In order to correctly estimate the normals field needed to apply the surface tension term, we have to regularize the implicit surface embedded in the convected density field. Designing an efficient filter is key to the elaboration of a fast eulerian solver. The ideal filter for this problem would conserve the anisotropy of the manifold by applying an isotropic regularization. The continuous to discrete mapping of the manifold results in loss of the surface regularity as seen on fig. 3.5 where the cartesian coarse meshing of the interface induces large errors when computing normals. Since too much regularization leads to losses of information at the fine-scale, the idea is to smooth as little as possible (concentration of the smoothing effect), while taking into account the whole domain. Failing to do so leads to unbalance of the penalization term at locations where the discrete density contour has a local discontinuity (between cells). This imbalance results in spurious currents which lead to destabilization of the numerical interface, and thus inevitable break-up of the inclusion.

The best candidate for smoothing would be the gaussian distribution since it is the only filter with a separable kernel. The first approach was to use a separable filter in the three euclidean directions. We started by considering the convolution of the density field with a variable size filter, even though this approach will be abandoned afterwards.

We write the definition of the convolution between two functions f and k :

$$(f * k)(\mathbf{x}) = \int_{-\infty}^{+\infty} f(\mathbf{x} - \tau)k(\tau)d\tau .$$

By choosing an appropriate kernel, we can truncate the convolution operator to a finite sum, which yields the following discrete sequence:

$$(f * k)[n] = \sum_{m=-M}^M f[n-m]k[m] .$$

Nominal 3D convolution is $\mathcal{O}(k_x \cdot k_y \cdot k_z \cdot f_x \cdot f_y \cdot f_z)$ with x, y, z standing for the dimensions of the input f . Instead of this costly naive implementation, nesting convolution (separating the filters, because the operator $*$ is associative) has computational cost of order:

$$\mathcal{O}(k_x \cdot f_x \cdot f_y \cdot f_z) + \mathcal{O}(k_y \cdot f_x \cdot f_y \cdot f_z) + \mathcal{O}(k_z \cdot f_x \cdot f_y \cdot f_z) ,$$

which reduces the cost from $\mathcal{O}(6)$ down to $\mathcal{O}(4)$.

The theoretical speed-up is thus $\frac{k_x \cdot k_y \cdot k_z}{k_x + k_y + k_z} = \frac{\alpha^{n-1}}{n}$, where n is the dimensionality of the problem and α a unit size for the convolution kernel.

This implementation of convolution based filtering has substantial drawbacks. The filter size needs to be at least of the order of the largest inclusion, which is to be determined insitu and not a priori. Since inclusions evolve with time, a conservative choice must be made before the computation can be run, and assumptions on the inclusion size must

be made. We implemented this filter for comparison purposes, but it cannot practically be used in the context of a fast eulerian fluid solver. Approximation of a gaussian blur using recursive box filtering (central limit theorem) seems also unusable, especially in the case of non-cubic cells, where the spread over the interface must be controlled.

3.1.3.4 Multi-scale computing

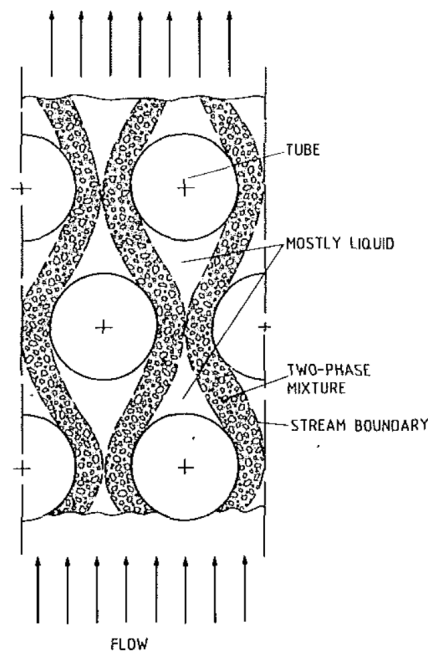


Figure 3.6 – Typical flow pattern observed in a tube bundle [80].

Different two-phase patterns can be observed in a steam generator bundle as seen on fig. 3.6. Small inclusions, due to shear-off on tube impact or droplet breakup, as well as higher scale structures upstream of the bundle inlet, or in the recirculation/stagnation zones make this problem a multi-scale one. In the context of *DNS*, or coarse *DNS* where the flow is fully resolved up to the boundary layers, correct resolution of the small scale structures is necessary. Therefore, a regularization method must be found in order to preserve multi-scale properties of the flow up to a chosen criterion. *DNS* is opposed to *LES*^v method where the Navier-Stokes Equation(s) are spatially filtered and the effect of smaller scale eddies are modelled, and *RANS*^{vi} methods where the whole turbulence spectrum is modelled, and the equations are time-averaged.

Some experiments were first implemented using convolution or boxed filter. We also try to formulate non linear regularization by backward diffusion [81–83]. Where forward diffusion has a smoothing effect, backward diffusion has a sharpening effect. We can

v. Large Eddy Simulation

vi. Reynolds-Averaged Navier-Stokes

also mention the curvature driven regularization methods [84, 85] where regions of high curvature (pinch) are regularized by using a diffusion equation: we could for example use the curvature to regularize a colour function c based on the density field $\frac{\partial c}{\partial t} = \nabla \cdot (d \nabla c)$, with d an artificial diffusion coefficient. Furthermore, curvature regularization is linked to the concept of minimal surface [86–88], which in practice happens when the capillary forces balance around a spherical bubble. Regularization by mean curvature also reduces the time-stepping constraint imposed by capillary effects [89]. An other alternative would consist of the implication of the contribution of capillary terms as exposed in [90, 91].

Ideally, the regularization process should preserve causality; hence, no new topological feature should be introduced by the method. Also, the blurring effect should preserve anisotropy. This criterion is important in the context of fluid mechanics since most of the time, algorithms are directionally split and an arbitrary choice is made over the order of application, thus requiring the algorithm to be space invariant [92].

The problem can also be exposed in the scale-space theory, where we view the surface as the “signal” with increasing levels of details [93, 94]. In the fig. 3.7, we figure out a surface as a fractal, successively smoothed increasing standard deviation filters. On the right, the second derivative of the bi-dimensional curve is represented. We observe that excessive filtering leads to loss of details from a topological point of view, hence the regular circle shape limit on the left.

From this analysis, we require that the theoretical filter should ideally combine two contradictory features:

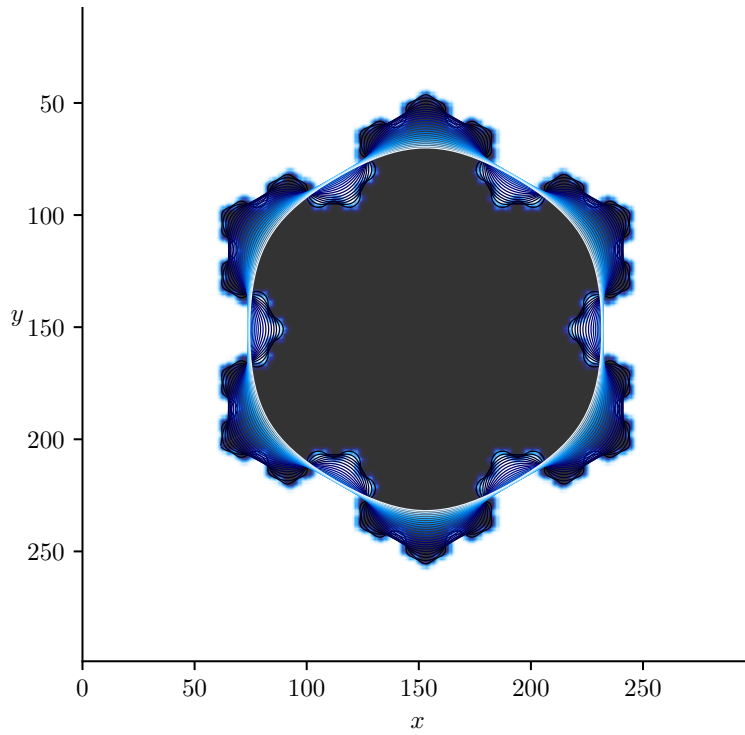
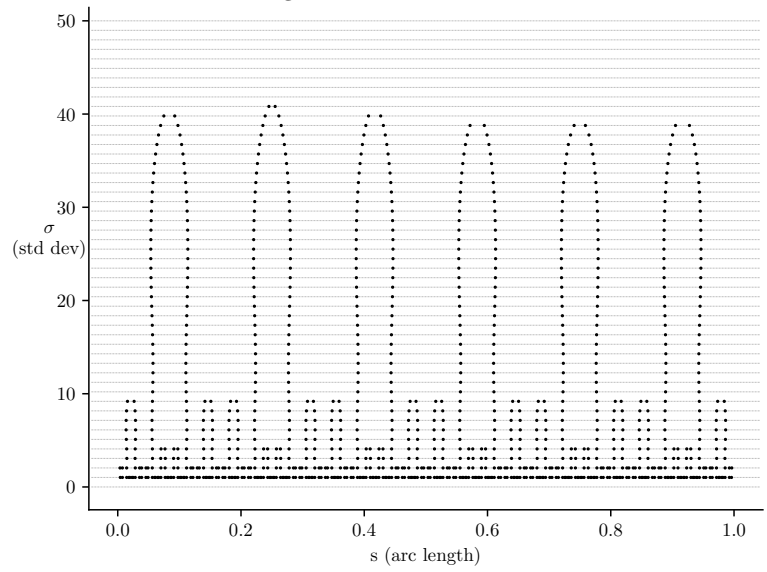
- taking into account the whole inclusion topology, hence maximize the numerical influence of the neighboring stencil;
- minimize the diffusion and the loss of fine scale details as demonstrated on fig. 3.7.

However, a fast approximation of gaussian filters can be made, and we shall use the filter described in [95, 96].

We exploit a property of causality described in the theory of signal processing. The input to the filter is akin to volumetric images, with the analogy of the density field the image to filter. The causality in time is analogous to a dependency in space in this case, making the current discrete point being regularized dependent on the previous point in space. What about the dependency to the following pixels/data ? The filter can be separated by cascading two stable filters whose discrete transfer functions can be written as:

$$\begin{aligned} H_+(z) &= \frac{1}{b_0 + b_1 z^{-1} + b_2 z^{-2} + b_3 z^{-3}}, \\ H_-(z) &= \frac{B}{b_0 + b_1 z^1 + b_2 z^2 + b_3 z^3}, \end{aligned} \quad (3.10)$$

where $b_{0..3}$ are the chosen filter coefficients. The causal filter H_+ solely depends on the

(a) Shape evolution with increasing standard deviation σ .

(b) Scale-space representation.

Figure 3.7 – Loss of information induced by excessive filtering on a self-similar pattern (fractal).

previous discrete point, while the anti-causal part H_- depends only on the next data (pixel).

We detail how the filter coefficients are determined using IIR^{vii} analysis for a LTI^{viii} system.

We first consider the spatial gaussian:

$$g(\mathbf{x}) = \frac{\exp(-\mathbf{x}^2 / 2)}{\sqrt{2\pi}}. \quad (3.11)$$

The eq. (3.11) is approximated by an algebraic fraction (rational function) with truncated HOT^{ix} :

$$\frac{1}{a_0 + a_2 \mathbf{x}^2 + a_4 \mathbf{x}^4 + a_6 \mathbf{x}^6} + HOT. \quad (3.12)$$

a_0	a_2	a_4	a_6
2.49231	1.44842	$-1.46374e - 11$	0.17167

Table 3.1 – Least-squares fit of the spatial discrete filter.

For example, with $x \in [-5; 5]$, the function in eq. (3.12) is fitted using a robust least-squares curve fitting algorithm [97] to give the set of coefficients presented in table 3.1:

The eq. (3.11) is then parametrized by a standard deviation σ :

$$g(\mathbf{x}|\sigma) = \frac{\exp(-\mathbf{x}^2 / (2\sigma^2))}{\sqrt{2\pi}\sigma}. \quad (3.13)$$

The eq. (3.13) is expanded into its Fourier transform:

$$\mathfrak{F}(eq. (3.13)) := G(\omega) = \exp(-\sigma^2 \omega^2 / 2), \quad (3.14)$$

for a pulsation ω .

Using the Laplace transform $F(s) = \int_0^\infty f(x) \exp(-sx) dx$ with $s = i\omega$, we write the approximation of eq. (3.14) as:

$$G_q(s) = a_0 - (a_2 q)s + (a_4 q^4)s^4 - (a_6 q^6)s^6, \quad (3.15)$$

where q is a mapping of σ as explained in [95] and the relation between q and σ is explicited in algorithm 1.

vii. Infinite Impulse Response

viii. Linear Time-Invariant

ix. Higher Order Terms

Factoring the eq. (3.15) into transfer functions with complex negative poles ($G_l(s)$, left hand side of the s plane) and positive poles $G_r(s)$:

$$G_q(s) = G_l(s) \cdot G_r(s), \quad (3.16)$$

with:

$$G_l(s) = \frac{A_0}{f_0(f_1^2 + f_2^2) - iq^3s^3 - q^2s^2(f_0 + 2f_1) + iqs(2f_0f_1 + f_1^2 + f_2^2)},$$

and

$$G_r(s) = \frac{1}{-f_0(f_1^2 + f_2^2) - iq^3s^3 + q^2s^2(f_0 + 2f_1) + iqs(2f_0f_1 + f_1^2 + f_2^2)}.$$

With identification of eq. (3.16) with eq. (3.15), the system to solve is thus:

$$\begin{aligned} f_0^2(f_1^2 + f_2^2)^2 - \frac{a_0}{a_6} &= 1, \\ 2f_0^2(f_1^2 - f_2^2) + f_1^2(f_1^2 + 2f_2^2) + f_2^4 - \frac{a_2}{a_6} &= 1, \\ -f_0^2 - 2f_1^2 + 2f_2^2 + \frac{a_4}{a_6} &= 1, \end{aligned}$$

which yields the set of coefficients referenced in table 3.2.

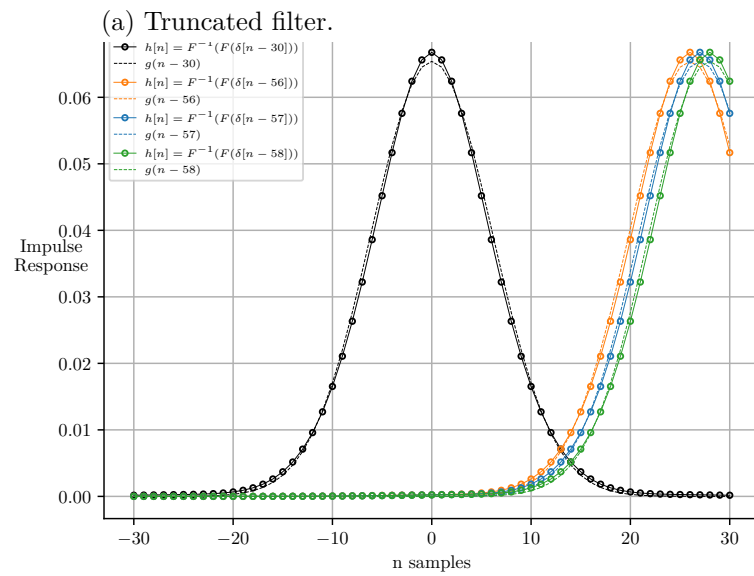
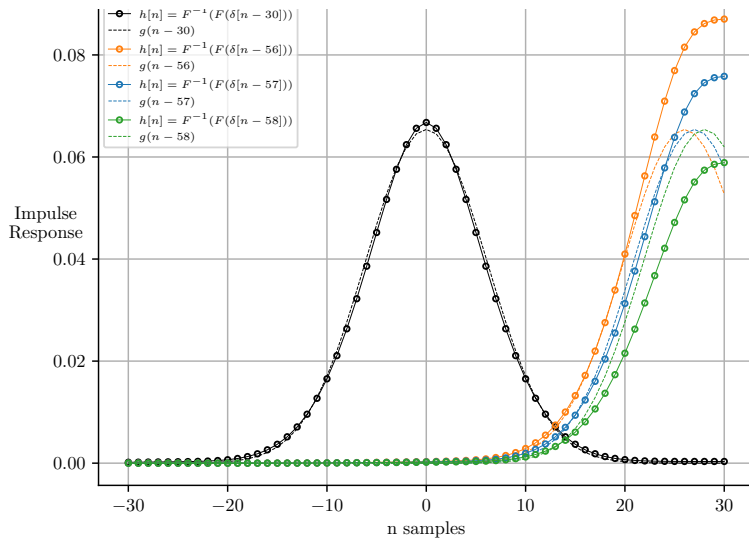
f_0	f_1	f_2
1.18209	1.12353	1.40036

Table 3.2 – Filter coefficients.

The effect of boundary conditions is key to any kind of regularization method. With this approach, truncation of the input signal can result in the loss of symmetry of the filter at the end of the domain as described in [98]. We reproduced the issue with an impulsion stimulus applied to the filter, in order to analyse its impulse response. On fig. 3.8a, amplitude and phase distortion is observed when using heuristic boundary data. On the opposite, when extending the filter to fictitious extents, the formulation at boundaries yields the correct impulse response as seen on fig. 3.8b.

The regularization effect over the volumetric image can be controlled by the standard deviation of the approximated gaussian filter. The authors in [95] recommend to pick a standard deviation above a value of .5 in order to correctly approximate the gaussian filter. The accuracy of the filter is enhanced with increasing standard deviation, but in our application, the concentration of the source term is required in order to asymptotically tend to a Dirac distribution across the thin interface width.

The algorithm 2 is used in order to distribute the regularization phase on multiple processes. The filter is by construction sequential in the considered direction (1D): the



(b) Corrected filter [98].

Figure 3.8 – Filter boundary regularization effect.

Algorithm 1: Python pseudo-code for the sequential directional recursive filter under a standard deviation σ .

```

1 function filter_inplace_1d( $\sigma$ , line)
    #  $s \rightarrow q$  transformation [95]
2     if  $\sigma < 3.556$  then
3         |  $q := -.2568 + .5784\sigma + .0561\sigma^2$ 
4     else
5         |  $q := .9804(\sigma - 3.556) + 2.5091$ 
6     end
7      $\iota := (f_0 + q) \cdot (f_1^2 + f_2^2 + 2f_1 \cdot q + q^2)$ 
8      $b_0 := f_0 \cdot (f_1^2 + f_2^2) / \iota$ 
9      $b_1 := q \cdot (2f_0 \cdot f_1 + f_1^2 + f_2^2 + (2f_0 + 4f_1) \cdot q + 3q^2) / \iota$ 
10     $b_2 := -q^2 \cdot (f_0 + 2f_1 + 3q) / \iota$ 
11     $b_3 := q^3 / \iota$ 
12     $\beta := 1 / (1 - (b_1 + b_2 + b_3))$ 
    #filter initialization
13     $\alpha := \text{line}[n - 1]$                                 #saved value before first pass
14     $\text{buffer}[1 : 4] := \beta \cdot \text{line}[0]$                     #boundary data
    #causal pass
15    foreach pixel in 1D line, forward sweep  $\forall i \in [0; n - 1]$  do
16        |  $\text{buffer}[0] = \text{line}[i] := \text{line}[i] + (\text{buffer}[1] \cdot b_1 + \text{buffer}[2] \cdot b_2 + \text{buffer}[3] \cdot b_3)$ 
17        |  $\text{buffer}[[3, 2, 1]] := \text{buffer}[[2, 1, 0]]$         #buffer shift
18    end
    #boundary regularization [98]
19     $m_0 := -b_3 \cdot b_1 + 1 - b_3^2 - b_2$ 
20     $m_1 := (b_3 + b_1) \cdot (b_2 + b_3 b_1)$ 
21     $m_2 := b_3(b_1 + b_3 \cdot b_2)$ 
22     $m_3 := b_1 + b_3 \cdot b_2$ 
23     $m_4 := -(b_2 - 1) \cdot (b_2 + b_3 \cdot b_1)$ 
24     $m_5 := -b_3(b_3 \cdot b_1 + b_3^2 + b_2 - 1)$ 
25     $m_6 := b_3 \cdot b_1 + b_2 + b_1^2 - b_2^2$ 
26     $m_7 := b_1 \cdot b_2 + b_3 \cdot b_2^2 - b_1 \cdot b_3^2 - b_3^3 - b_3 \cdot b_2 + b_3$ 
27     $m_8 := b_3(b_1 + b_3 \cdot b_2)$ 
28     $c_1 := \text{buffer}[1] - \alpha \cdot \beta$ 
29     $c_2 := \text{buffer}[2] - \alpha \cdot \beta$ 
30     $c_3 := \text{buffer}[3] - \alpha \cdot \beta$ 
31     $\text{buffer}[0] = \text{line}[0] := b_0^2(m_0 \cdot c_1 + m_1 \cdot c_2 + m_2 \cdot c_3 + \alpha \cdot \beta^2)$ 
32     $\text{buffer}[1] := b_0^2(m_3 \cdot c_1 + m_4 \cdot c_2 + m_5 \cdot c_3 + \alpha \cdot \beta^2)$ 
33     $\text{buffer}[2] := b_0^2(m_6 \cdot c_1 + m_7 \cdot c_2 + m_8 \cdot c_3 + \alpha \cdot \beta^2)$ 
    #anti-causal pass
34     $\text{buffer}[[3, 2, 1]] := \text{buffer}[[2, 1, 0]]$ 
35    foreach pixel in 1D line, backward sweep  $\forall i \in [n - 1; 0]$  do
36        |  $\text{buffer}[0] = \text{line}[i] := \text{line}[i] + (\text{buffer}[1] \cdot b_1 + \text{buffer}[2] \cdot b_2 + \text{buffer}[3] \cdot b_3)$ 
37        |  $\text{buffer}[[3, 2, 1]] := \text{buffer}[[2, 1, 0]]$ 
38    end
39 end

```

Algorithm 2: Domain decomposition parallel recursive blur (Fortran Coarray implementation).

```

1 sync images(*)                ! Synchronization point for all images
2 if  $\exists$  left image then
3   | sync images(left)         ! Barrier, synchronization point with previous
   |   image
4   | sync_halo(left)           ! Fetch data from remote window on a remote
   |   process, and fill the current processors halos
5 end
6 forward_filter()              ! Apply a local recursive filter in the forward
   direction using a limited stencil using algorithm 1
7 if image == rightmost image then
8   | regularize for truncated data at boundaries [98]
9 end
10 sync images(right)            ! Ready to sync data with a remote process
11 sync images(*)                ! Synchronization point for all images (barrier)
12 if  $\exists$  right image then
13   | sync images(right)        ! Synchronize with the following image
14   | sync_halo(right)
15 end
16 backward_filter()             ! Apply the local filter in the backward direction,
   on each process
17 sync images(left)             ! Signal the remote process for halo synchronization
18 sync images(*)                ! Synchronization point for all images

```

right process must wait boundary data from the left process on 3 layers (since we chose a 3 coefficients filter in table 3.2) and reverse this dependency for the anti-causal pass; this results overall in negligible slow-down since the numerical implementation is three-dimensional. The splitting of the filter thus occurs in a cascading manner on the x , y and z directions.

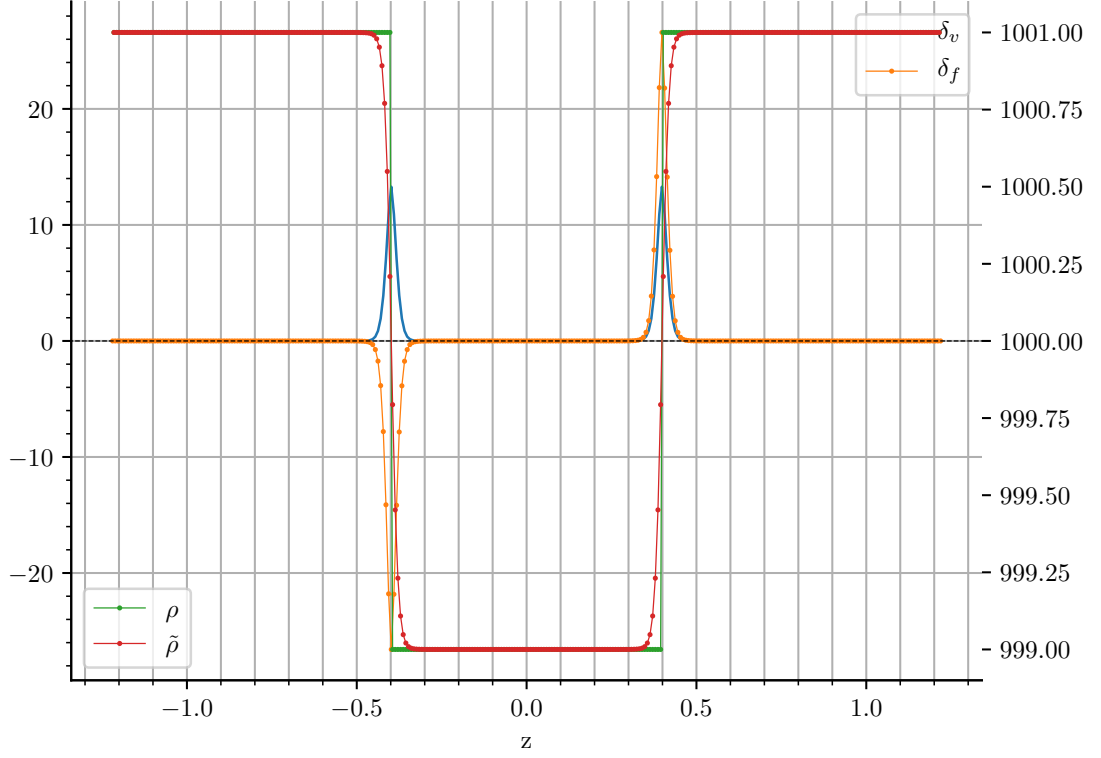
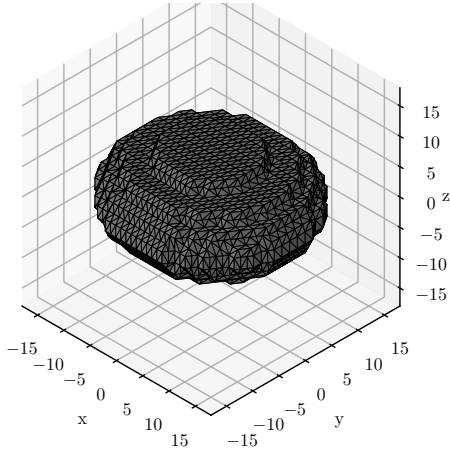


Figure 3.9 – Density transition and interfacial source term distribution across the secondary phase.

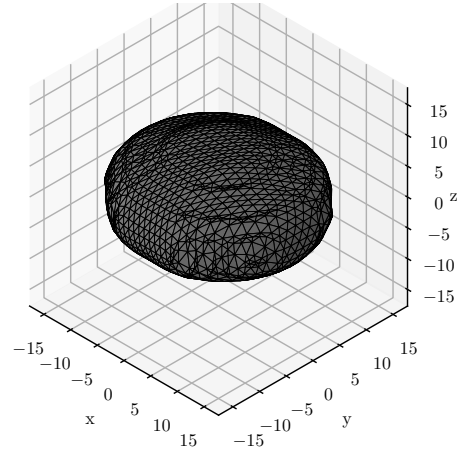
On the fig. 3.9, the source term is distributed over a thin transition region obtained by regularization of the convected density. The integration of the distribution over the interface thickness sums up to unity. The width of the pulse determines the recirculation in the transition region, which is due to gradients in the spatial distribution of κ , when considering interfaces evolving out of mechanical equilibrium.

We tried to alleviate the problem of multi-scale feature by combining multiple filters.

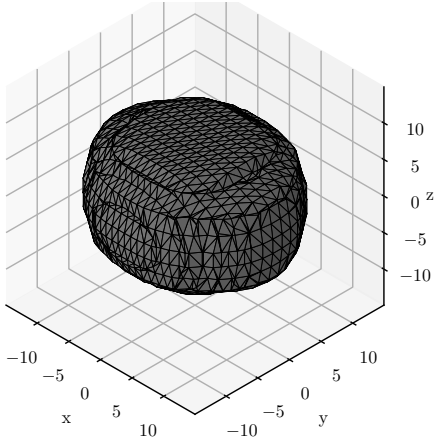
The fig. 3.10a shows a Delaunay triangulation (using the implementation from matplotlib) drawn top of a cartesian density volumetric field, with the iso-value set to 0. The discontinuity imposed by the non-conforming mesh yields large errors in numerical curvature, thus erroneous interfacial forces. The fig. 3.10c shows a regularization us-



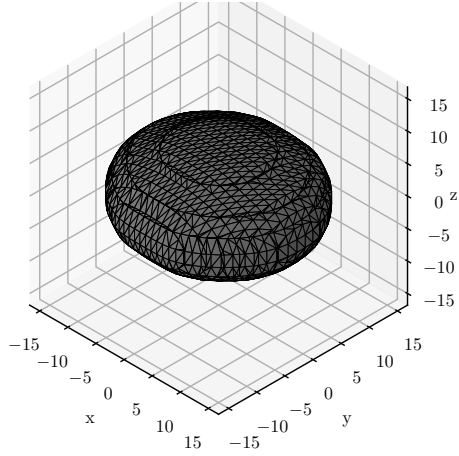
(a) Convected density after triangulation and without smoothing.



(b) Conforming mesh smoothing (post processing)[99].



(c) Pass band filter output.



(d) gaussian approximation smoothing [95, 96].

Figure 3.10 – Initial un-regularized field (top-left), post-triangulation (top-right), pass-band smoothing (bottom-left), recursive filter (bottom-right)

Algorithm 3: Pass band filter applied to the input manifold f .

```

1 choose  $\lambda > 0$  and  $\mu < 0$ , with  $|\mu| > |\lambda|$ 
2 while  $\text{norm}(\text{increment}) < \text{threshold}$  do
3    $f \leftarrow f + \lambda \Delta f$ 
4    $f \leftarrow f + \mu \Delta f$ 
5 end
```

ing a combination of an expanding filter based on the Laplace operator applied to the input field (minimizing the curvature variations), and a consecutive contracting filter algorithm 3, inspired by the work of [100].

The fig. 3.10d exposes the result after one forward and one backward pass of the gaussian approximation recursive filter. This is the closest result to the post-processed fair surface fig. 3.10b. The fair surface was obtained using triangulation of the surface as an input to various regularization methods. This is un-applicable insitu, since our objective is to use only cartesian description of inclusions and not conforming methods such as *ALE* or *FT*. However, since these methods are accurate, we are inclined to compare our method with these post-processed results. Hence, fig. 3.10b shows the smoothed surface, with minimal curvature variations $\nabla(\Delta f) \rightarrow 0$.

A tentative to formulate the problem of curvature approximation on an highly discontinuous manifold, as an elliptic equation was experimented. However, for the time being, this approach did not provide satisfactory results although it remained worth exploring.

3.1.4 Fluid properties evolution

Algorithm 4: Conservative transport of the thermodynamic properties

```

1 for each spatial direction do
2   compute the flux balance (in/out) for each cell using a chosen limiter on
   fig. 3.23
3   update the scalar density field using the three fluxes computed in the
   previous step
4   extrapolate viscosity as a spatial function of density
5 end
```

The density is transported in a conservative manner using algorithm 4. The dynamic viscosity is extrapolated from the density in order to keep a constant kinematic viscosity for each phase, and the transition in the mixture layer smooth:

$$\mu(t, \mathbf{x}) = \min(\mu_1, \mu_2) + \rho(t, \mathbf{x}) \cdot \frac{\max(\mu_1, \mu_2) - \min(\mu_1, \mu_2)}{\max(\rho_1, \rho_2) - \min(\rho_1, \rho_2)}. \quad (3.17)$$

We also mention the fact that viscosity and density are not necessarily bound between the primary and the secondary phase values. Clamping — truncating — the fields to respect the bounds results in the loss of total mass and is not necessary for the code to avoid *FPE(s)*^x. Thus, it is better to keep values slightly outside the allowed intervals, and optionally reset for mass conservation from time to time.

The density is averaged at facet locations where needed, for example when updating velocities in the *MAC*^{xi} convention for finite volume calculations. As for the update

x. Floating Point Exception(s)

xi. Marker and Cell

of viscosities, a choice has to be made on the averaging method. Especially when the density ratio is important e.g. air-water flow with $\rho_1 = 1000$ and $\rho_2 = 1$, the intermediate facet values are key for the dynamic evolution for the transfer of momentum. The choice of choosing the averaging method is left to the user: harmonic, geometric or arithmetic mean. The harmonic mean, however is accurate when the function or field is discontinuous [101–103]. A numerical study of the influence of the averaging method should be interesting, especially the influence on the convergence of the two-phase incompressible pressure solver developed in section 3.3.4, which uses interpolated inverse of facet densities.

3.1.5 Incompressible, non-dilatable formulation

A vector field can be uniquely decomposed into a sum of a solenoidal part and a irrotational part [104–106]. The HHD ^{xii} of this vector field is defined as:

$$\mathbf{v} = \mathbf{v}_d + \mathbf{v}_r + \mathbf{v}_h, \quad (3.18)$$

where \mathbf{v}_d is irrotational and thus derives from of a scalar potential $\mathbf{v}_d = \nabla D$, $\mathbf{v}_r = \nabla \times R$ is a solenoidal part and \mathbf{v}_h is an harmonic part. The three components are mutually orthogonal with respect to the inner product in \mathcal{L}^2 .

The analogy of eq. (3.18) with fluid motion is that a fluid element can move in \mathcal{R}^3 by:

- expansion or contraction (irrotational movement: \mathbf{v}_d);
- rotation around an axis (incompressible displacement: \mathbf{v}_r);
- translation \mathbf{v}_h .

The translation term vanishes in infinite space, or in finite domain with null boundary flux. In the following, part, we will assume $\mathbf{v}_h = 0$.

Both solenoidal and irrotational subspaces of $\mathcal{L}^2(\Omega)$ span the vector space of \mathbf{v} :

$$\mathcal{L}^2(\Omega) = \{\mathbf{v} \in \mathcal{L}^2(\Omega) | \nabla \cdot \mathbf{v} \in \mathcal{L}^2(\Omega), \mathbf{v} \cdot \mathbf{n}_{|\partial\Omega} = 0\} \oplus \nabla \mathcal{H}^1(\Omega).$$

A correction term, homogeneous to a pressure gradient is computed to correct the predicted velocity field. The literature refers this method as the classic prediction-correction method described in [107]. This implicit phase is often costly, requiring inversion of a matrix using a direct inversion method such as the LU factorization/Cholesky decomposition, or usage of an iterative methods based on updating the solution based on a minimization criterion — for instance the orthogonality of the residual vectors of the Krylov subspace for the CG ^{xiii} method —, or minimizing the norm of the residual in the span of the set $\{b, Ab, A^2b, \dots, A^{n-1}b\}$ for the $GMRES$ ^{xiv} method.

xii. Helmotz-Hodge Decomposition

xiii. Conjugate Gradient

xiv. GEneralized Minimal RESidual

We note that the pressure computed in this step, although homogeneous to a physical pressure is no longer a thermodynamic variable, but a Lagrange multiplier to capture the divergence-free condition.

We enforce a divergence free velocity with the correction step:

$$\mathbf{u}^{n+1} = \tilde{\mathbf{u}}^n - \frac{\Delta t}{\rho} \nabla p^{n+1}. \quad (3.19)$$

Taking the divergence of the previous equation results in:

$$\cancel{\nabla \cdot (\mathbf{u}^{n+1})} \stackrel{0}{=} \nabla \cdot (\tilde{\mathbf{u}}^n) - \Delta t \nabla \cdot \left(\frac{\nabla p^{n+1}}{\rho} \right).$$

We obtain a variable coefficient Poisson equation on the domain Ω , and open of \mathcal{R}^3 , with homogeneous boundary $\partial\Omega$:

$$\begin{cases} \nabla \cdot \left(\frac{\nabla p^{n+1}}{\rho} \right) = \overbrace{\frac{1}{\Delta t} \nabla \cdot (\tilde{\mathbf{u}}^n)}^b \text{ on } \Omega, \\ \frac{\partial p}{\partial n} = \underbrace{0}_g \text{ on } \partial\Omega. \end{cases} \quad (3.20)$$

A compatibility condition of the system eq. (3.20) requires that following equality be satisfied:

$$\int_V b \, dv = \int_S g \, ds,$$

where b is the RHS^{xv} of the linear system and g the boundary condition applied, which is zero valued in our case (no pressure gradient across the walls).

Also, when the linear operator has a constant null-space (all Neumann $BC(s)^{xvi}$), we enforce the multiplier $\int_\Omega p \, dv = 0$ at each iteration of the incremental Krylov method, so that the solution is defined up to an zero additive constant. The meaning of the additive constant and its physical meaning is to be discussed: it is believed that this pressure gauge tends towards the real pressure in a thermodynamic sense (dynamic pressure). We could also modify the linear system in order to fix an arbitrary pressure point in the domain (solution valid up to a constant).

Despite chosen boundary conditions can be discussed [108], the usual approach is to use zero flux pressure boundary conditions at walls. Since the problem with Neumann boundary conditions is defined up to a constant, we subtract the average of the RHS of eq. (3.20).

xv. Right-Hand Side

xvi. Boundary Condition(s)

The problem was first solved in two dimensions, for the Navier-Stokes Equation(s) with constant density ρ . We use simple harmonic functions for the velocity and the pressure [109]:

$$\begin{cases} u = -\cos(x) \sin(y), \\ v = \sin(x) \cos(y), \\ p = \frac{-Re}{4} (\cos(2x) + \cos(2y)). \end{cases} \quad (3.21)$$

We compare the methods described in [110–112], and compute the error between the computed field against the analytical solution.

In [110], the authors reuse the pressure gradient at the previous iteration in the prediction step, before the projection step. The diffusive step is computed using an average between an explicit step and an implicit step (CN^{xvii} method). For the velocities, we find a second order convergence rate in time as shown in fig. 3.11.

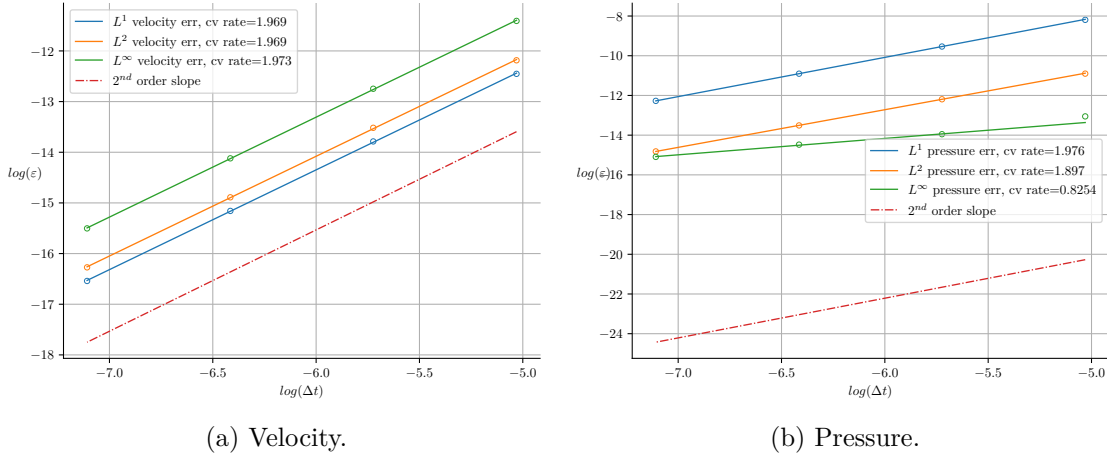


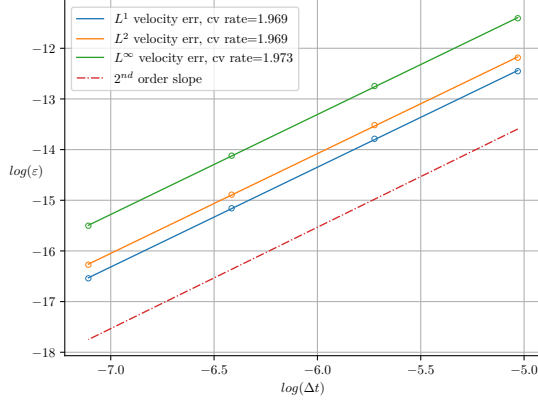
Figure 3.11 – Convergence rates for the Taylor problem, using [110].

The second method of [112] enhances the ℓ^∞ convergence rate on the pressure by subtracting a correction term to the pressure used in the prediction step, while keeping the velocity update un-modified. As shown on fig. 3.12, the method is second order accurate for the velocity fields, second order accurate in the ℓ^1 and ℓ^2 norms on the pressure but only slightly more than first order in the ℓ^∞ norm.

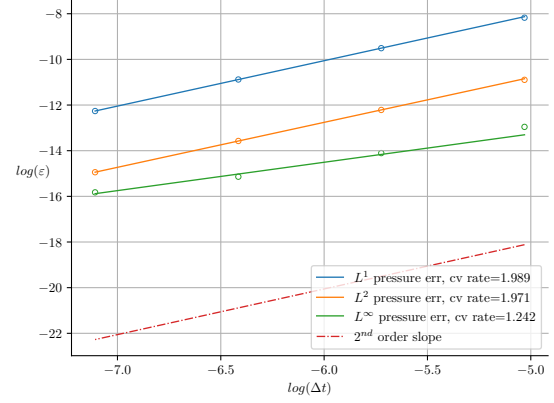
The third method, proposed in [111], differs with the two others, since no pressure gradient is applied to the predicted velocity field. Hence, we solve for a term homogeneous to a pressure. A stabilization term is also added to make the pressure second order accurate: $p^{n+1} = \varpi^{n+1} - \frac{\Delta t}{2Re} \Delta \varpi^{n+1}$.

We also explored the promising method described in [113] without success: the incompressibility of the fluid at the interface was never enforced.

xvii. Crank-Nicolson

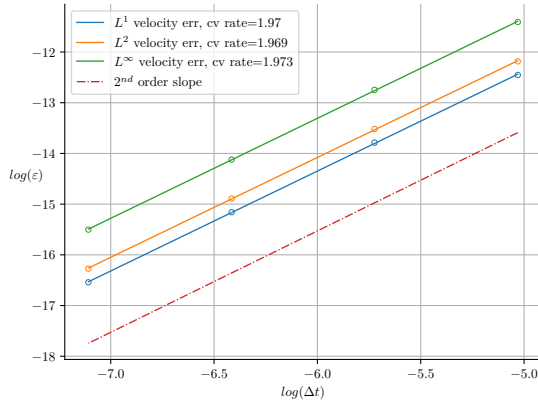


(a) Velocity.

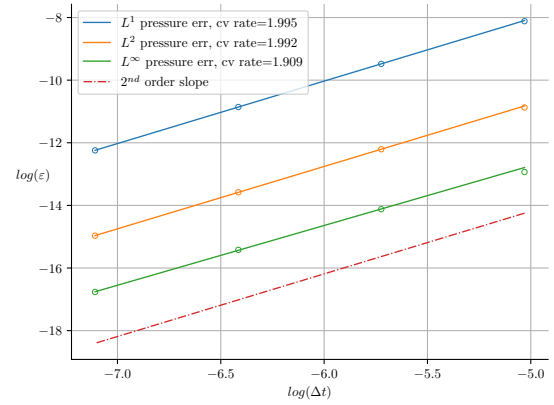


(b) Pressure.

Figure 3.12 – Convergence rates for the Taylor problem, using [112].



(a) Velocity.



(b) Pressure.

Figure 3.13 – Convergence rates for the Taylor problem, using [111].

3.1.6 Pressure jump conditions

The convergence of the pressure solver was difficult to achieve: since the interface was spread over a few cells, the solution presents strong discontinuities, and derivatives are difficult to compute accurately. We thus seek an alternative numerical method, ideally a discontinuity preserving one, where the interfacial gap must be recovered on the pressure. In this part, we follow the reasoning of [114–116]. From eq. (3.20), we consider the family of spatially variable coefficient Poisson equation, with Dirichlet or Neumann $BC(s)$:

$$\begin{cases} \nabla \cdot (\beta(\mathbf{x}) \nabla u(\mathbf{x})) = f(\mathbf{x}), \mathbf{x} \in \Omega, \\ u(\mathbf{x}) = d(\mathbf{x}) \text{ or } u_n(\mathbf{x}) = g(\mathbf{x}), \mathbf{x} \in \partial\Omega, \end{cases} \quad (3.22)$$

where $d(\mathbf{x})$ is the Dirichlet boundary condition, $g(\mathbf{x})$ is the imposed derivative on the boundary, \mathbf{x} is the vector of spatial coordinates in \mathcal{R}^3 , and $\beta(\mathbf{x})$ is the spatially variable coefficient in the linear operator (divergence term). The previous equation behaves numerically well when $\beta(\mathbf{x})$ varies smoothly with \mathbf{x} . However, in the context of incompressible flows, the density in the domain composed of non-miscible fluids is discontinuous at/around the interface. For this issue, we consider a partitioning of the domain Ω into multiple subsets Ω_i , where subsets are bound by interfaces Γ_i and/or computation boundaries. The unit normal determines the direction of the jump from Ω^- into Ω^+ .

Then, jump conditions (bracketed notation) can be written as:

$$\begin{cases} [u]_\Gamma = u^+(\mathbf{x}) - u^-(\mathbf{x}), \\ [\beta u_n]_\Gamma = \beta^+(\mathbf{x}) u_n^+(\mathbf{x}) - \beta^-(\mathbf{x}) u_n^-(\mathbf{x}), \end{cases} \quad (3.23)$$

where $u_n = \nabla u \cdot \mathbf{n}$ is the normal derivative.

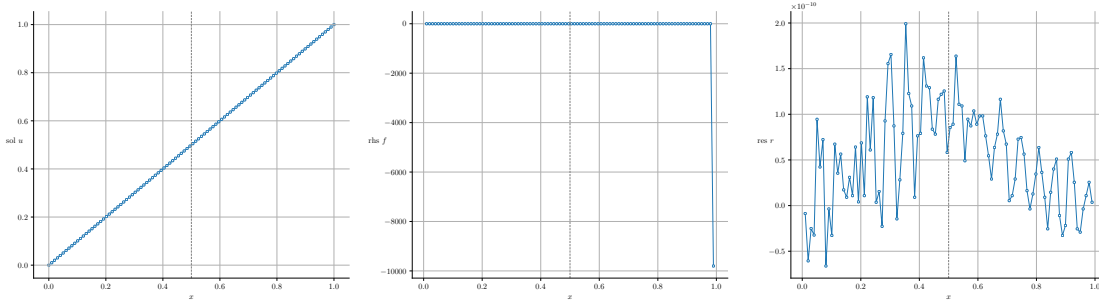


Figure 3.14 – Laplace equation (u , f and $r = f - A \cdot u$)
 $\beta = 1, [u] = 0, [\beta u_n] = 0, u(0) = 0, u(1) = 1.$

The one-dimensional results were obtained by considering the Laplace equation $f(x) = 0$ and $\beta = 1$ with imposed Dirichlet $BC(s)$ $u(0) = 0$ at the left of the domain and $u(1) = 1$ on the right, and interface located at $x = .5$.

The various interfacial source terms are shown on figs. 3.14 to 3.16. We observe the solution u , obtained when solving the Laplace equation with the jump conditions inserted

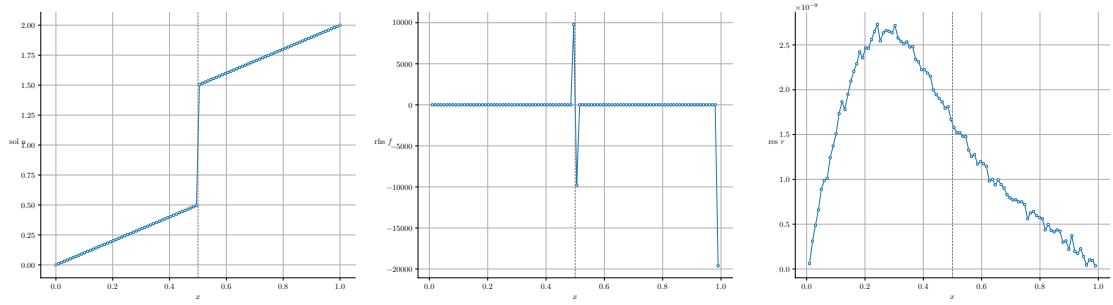


Figure 3.15 – Laplace equation (u , f and r)
 $\beta = 1$, $[u] = 1$, $[\beta u_n] = 0$, $u(0) = 0$, $u(1) = 1$.

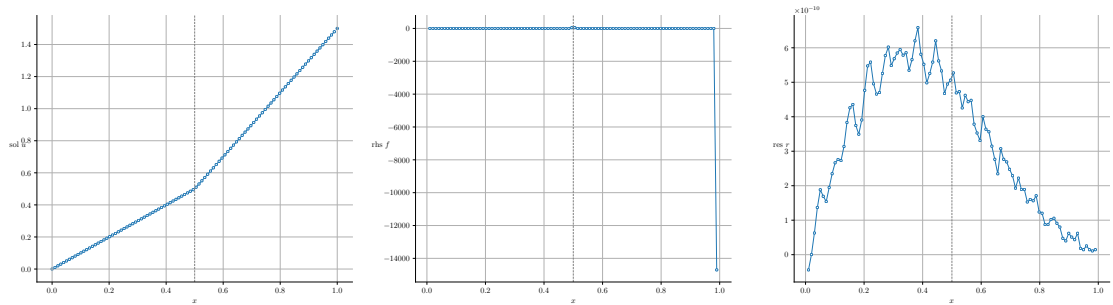


Figure 3.16 – Laplace equation (u , f and r)
 $\beta = 1$, $[u] = 0$, $[\beta u_n] = 1$, $u(0) = 0$, $u(1) = 1$.

and the *RHS* f , together with the residual r . We note that the behaviour of the residual obtained using the [117] algorithm is not dependent of the position of the discontinuity, which confirms that solving eq. (3.22) with eq. (3.23) is numerically well behaved and is suitable for the multiphase problem with disjoint densities.

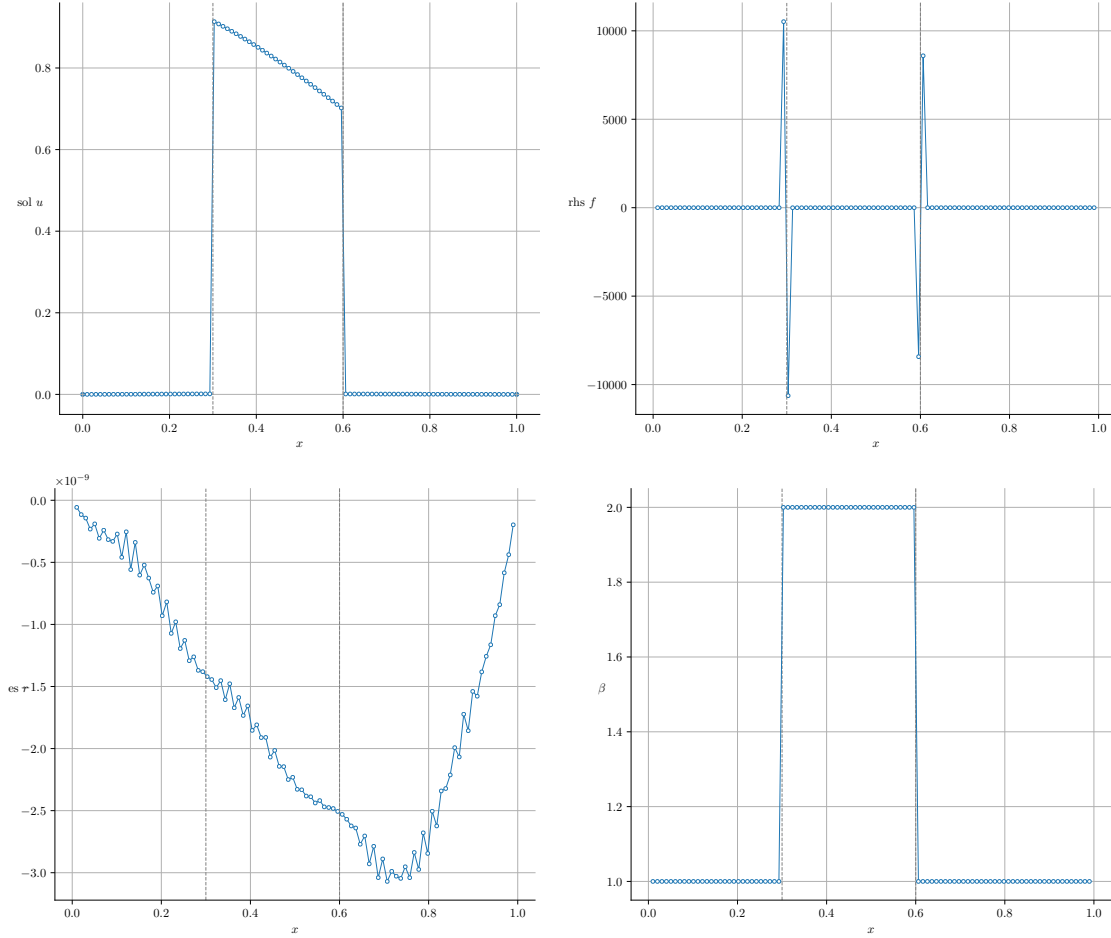


Figure 3.17 – Poisson equation (u, f, r, β) with two discontinuities.

The parameters for fig. 3.17 are:

$$\beta = \begin{cases} 1 & \text{if } x \in [0; .3] \cap [.6; 1], \\ 2 & \text{otherwise;} \end{cases}$$

$$f(x) = \begin{cases} (8x^2 - 4)e^{-x^2} & \text{if } x \in [.3; 6], \\ 0 & \text{otherwise.} \end{cases}$$

$$\text{At } x = .3 : \begin{cases} [u] = e^{-.09}, \\ [\beta u_n] = -1.2e^{-.09}. \end{cases}$$

$$\text{At } x = .6 : \begin{cases} [u] = -e^{-.36}, \\ [\beta u_n] = -2.4e^{-.36}. \end{cases}$$

Finally, no source term originates from boundaries: we impose $u(0) = 0$ and $u(1) = 0$.

Remarkably, the solution u in fig. 3.17 produces no intermediate or smeared values around jumps, although β is discontinuous. Stiff source terms can also be inserted at the interface, in place of smeared terms which lead to diffusion of the interface.

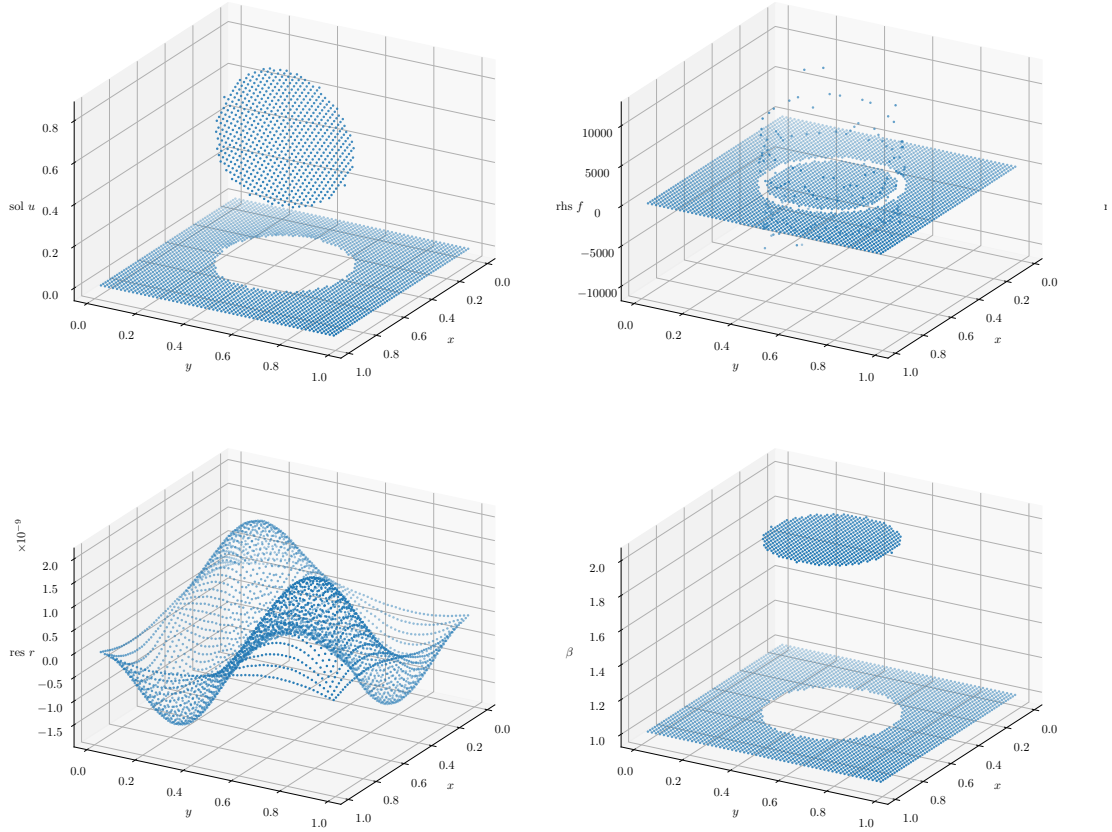


Figure 3.18 – Bi-dimensional extension of fig. 3.17 (u, f, r, β).

The fig. 3.18 extends the one-dimensional problem exposed in fig. 3.17 with the following set of parameters:

$$\beta = \begin{cases} 1 & \text{if } (x - .5)^2 + (y - .5)^2 \leq .25^2, \\ 2 & \text{otherwise;} \end{cases}$$

$$f(x) = \begin{cases} 8(x^2 + y^2 - 1)e^{-x^2 - y^2} & \text{if } x \in \Omega^-, \\ 0 & \text{otherwise.} \end{cases}$$

The jumps conditions are:

$$\begin{cases} [u] = e^{-x^2-y^2}, \\ [\beta u_n] = 8(2x^2 + 2y^2 - x - y)e^{-x^2-y^2}, \end{cases}$$

with null Dirichlet $BC(s)$: $u(0) = 0$ and $u(1) = 0$.

In our opinion, this is the correct way to treat the spatially variable coefficient Poisson equation. However, for this application, constrained numerical dissipation using the anti-diffusive formulation (section 3.3.1.2) for the conservative transport of density yields stiff coefficients for the eq. (3.20). This is thus a perspective in order to enhance the slow convergence observed in fig. 3.33 with classic iterative methods. Some alternative references on the *GFM*^{xviii} and other approaches can be found in [118–120].

3.1.7 Bundle model and coupling with numerical fluid dynamics

In order to study different physics, one has to couple different numerical methods together. Nowadays, expertise in specific domains is quite robust, but the coupling is a brand new challenging area for numerical engineers and research scientists. The goal of this part is to expose the mathematical description of the immersed portion of a steam generator bundle, and how we might approach the coupling between tubes and the influence of a single tube together with the surrounding fluid.

3.1.7.1 Introduction to vibration analysis

In our application, the target is to place obstacles in an evolving flow, while taking into account the retro-action of the solid movement onto the fluid. For simplicity and computational costs, we have chosen to solve the *IBM*^{xix} equation in the fluid framework, which implies to use the same discretization for the solid as the one used for the fluid. Alternative choices could have been made – for instance a boundary fitting modelling (*ALE* method) – but this poses other issues such as information transfer, interpolation and non-coincident discretization. Hence, the solid interface is described implicitly, but the effect is included onto the fluid solver as a penalization term on the $NSE(s)$. The canonical and academic set up to be studied is a non-deformable cylinder evolving in the vertical and horizontal direction, with free transverse boundaries. An enriched model should be able to reproduce the deformation of tubes with clamped/clamped $BC(s)$.

The goal is to reproduce the flow configuration on a portion of the tube bundle under the following hypotheses:

- nominal pressure and temperature conditions in a bundle section;
- air/water mixture.

xviii. Ghost Fluid Method

xix. Immersed Boundaries Methods

It is generally assumed that the bundle is subject to random fluid-elastic forces [121] induced by turbulence and phase intermittence of the secondary circuit (density transitions between a lighter — usually steam — and a heavier — usually water — fluid). The wide excitation spectrum contributes to different modes along the whole structure. Vibrations of the tubes induces a fluid motion, which results in a fluid-elastic force exerted on the tube. The physical characters of the structure is thus dynamically modified and unstable response can be observed, leading to potential damage of the structure. When subject to cross-flow, two kinds of fluid-structure coupling can occur:

- coupling with a fluid elastic force, decoupled from the tube movement: it is assumed that the flow unsteadiness is not impacted by the tube vibrations. For mono-phase flows, it is the sole consequence of the turbulence of the stream, whereas for a two-phase flow, the local fluid density has a key role [122];
- a fluid-elastic force which characterizes the adaptation of the fluid to the tube movements, modelled as added mass, stiffness and damping coefficients [80, 123].

We also mention the coupling with a fluid at rest in case of a mono-phase flow, where analytic solutions can be used in order to determine the solid parameters through simple relaxation of initial tube movement in a viscous fluid.

Numerical simulation is key, since it is able to approach regime avoided during exploitation. Experimentally, the fluid-elastic instability is obtained for specific flow regimes using costly and complex mock-ups. However, those are far from nominal flows observed in power plant exploitation. Numerically, more parameters can be explored by fine tuning configurations, avoiding measure bias and/or reproducibility issues.

The underlying goal of numerical work is to provide nondimensionalization of the fluid-elastic and random excitation forces in order to provide predictive knowledge of the vibration spectrum.

3.1.7.2 Mathematical description

The first assumption is that the fluid response to the solid movement is linear. This is a strong assumption since the $NSE(s)$ are highly non-linear.

Fluid elastic instability occurs when the dynamic damping $c_s + c_a \rightarrow 0$ [124]. The coupling between the fluid and the solid solver is a weak one, since a Taylor expansion of the fluid forces is used to compute the tube response. In the context of *IBM* [125], we use a regularization process in order to smooth pressure jumps on the cartesian mesh.

The effect of the immersed boundary is enforced on the closest fluid nodes, even if the solid boundary does not coincide with the eulerian mesh. The computational expense is thus deterministic, since only the displacement of the centre of gravity is tracked.

The first approach chosen is the displacement of the centre of gravity of a single tube. However, in a steam generator, the bundle can vibrate in the U bend plane or orthogo-

nally relative to this plane. It might then be necessary to include a bending model with moments and study the bundle in the transverse direction.

The fluid-solid force result in the fluid pressure and viscous stresses exerted on the cylinder surface.

The static pressure part is retrieved from the diagonal part of the stress tensor:

$$\begin{aligned} F_p &= \iiint_V -\nabla p \, dv \\ &= \oint_S -p \cdot \mathbf{n} \, ds, \end{aligned} \quad (3.24)$$

where S is the cylinder surface.

The viscous contribution comes from the deviatoric part of stress tensor:

$$\begin{aligned} F_v &= \iiint_V \nabla \cdot \mathbf{T} \, dv \\ &= \iiint_V \nabla \cdot \mu(\nabla \mathbf{u} + \nabla^t \mathbf{u}) \, dv \\ &= \oint_S \mu(\nabla \mathbf{u} + \nabla^t \mathbf{u}) \cdot \mathbf{n} \, ds. \end{aligned} \quad (3.25)$$

The nondimensionalization of the forces gives the pressure and viscous coefficients, which are projected onto the movement axes to form the drag and lift coefficients:

$$C_p = \frac{\oint_S -p \cdot \mathbf{n} \, ds}{.5A\rho U^2}, \quad (3.26)$$

$$C_\nu = \frac{\oint_S \mathbf{T} \cdot \mathbf{n} \, ds}{.5A\rho U^2}, \quad (3.27)$$

$$C_T = C_p + C_\nu. \quad (3.28)$$

The next hypothesis is to use a discrete solid description instead of a continuous one. Thus, displacements are characterized by the degrees of freedom. This description allows to use modal theory in order to describe the bundle dynamics. For example, the discrete description of a cantilever split into n segments depicted by springs converges to the continuous description of the system when $n \rightarrow \infty$.

The degrees of freedom of the system span a vector space. The dot product in the Hilbert space represent an energy in mechanics.

If we take a displacement field $\mathbf{x}(\mathbf{r}, t)$ and a strain field $\mathbf{f}(\mathbf{r}, t)$, then the dot product $\langle \cdot, \cdot \rangle$ on a control volume represent the work of the applied forces:

$$\langle \mathbf{x}(\mathbf{r}, t), \mathbf{f}(\mathbf{r}, t) \rangle = \iiint_V \mathbf{x}(\mathbf{r}, t) \mathbf{f}(\mathbf{r}, t) \, dv \underset{\text{discrete system}}{=} \sum_i x_i(t) f_i(t). \quad (3.29)$$

The dot product used is defined on the complex space:

$$\langle X, F \rangle = \iiint_V X^*(\mathbf{r}) F(\mathbf{r}) dv = \sum_i X_i^* F_i. \quad (3.30)$$

If we take \mathbf{x}_s to be the displacement of the tube, we write the matrix equation:

$$M_s \ddot{\mathbf{x}}_s + C_s \dot{\mathbf{x}}_s + K_s \mathbf{x}_s = \iint_S \boldsymbol{\sigma}(\mathbf{x}_s, t) \cdot \mathbf{n}(\mathbf{x}_s, t) ds = f_{f \rightarrow s}(\mathbf{x}_s, t), \quad (3.31)$$

where S is the fluid-solid interface, M_s, C_s, K_s are the mass, damping and stiffness matrices, and \mathbf{x}_s is the vector of displacement of the tube along its degrees of freedom.

A simplification hypothesis is to consider the solid as non-deformable, which reduces the n degrees of freedom to six (three in rotation and three in translation). The clamped-clamped boundary condition, used in order to study the behaviour on a single tube segments, further allows the reduction from six to two degrees, namely drag and lift translations. This simplest description might however be too restrictive to capture the complex dynamics of the tubes, and might evolve if the initial numerical results are failing to provide conclusive answers.

The fluid-to-solid force appearing on the *RHS* of eq. (3.31) must be characterized. The common approach is to split the force into a random excitation force \mathbf{f}_{rnd} and a fluid elastic contribution \mathbf{f}_{fe} [54] in one-phase or two-phase cross-flow [126]. The random excitation force is seen as white noise with wide band excitation spectrum.

The eq. (3.31) is projected into the modal base of the tube:

$$M_s \ddot{\mathbf{q}} + C_s \dot{\mathbf{q}} + K_s \mathbf{q} = \mathbf{f}_{rnd}(t) + \mathbf{f}_{fe}(\mathbf{q}, t), \quad (3.32)$$

where $\mathbf{q}(t)$ is the generalized displacement of the tube (linear combination of displacement degrees of freedom).

The fluid elastic force is linearised into:

$$\mathbf{f}_{fe} = -m_f \ddot{\mathbf{q}} - c_f \dot{\mathbf{q}} - k_f \mathbf{q}. \quad (3.33)$$

Combining eq. (3.32) and eq. (3.33), we write

$$(m_s + m_f) \ddot{\mathbf{q}}(t) + (c_s + c_f) \dot{\mathbf{q}}(t) + (k_s + k_f) \mathbf{q}(t) = \mathbf{f}_{rnd}(t). \quad (3.34)$$

The equation eq. (3.34) $m\ddot{\mathbf{x}} + c\dot{\mathbf{x}} + k\mathbf{x} = \mathbf{f}$ is identified as a second order damped system to a generic nondimensionalized oscillator equation:

$$\frac{d^2 q}{dt^2} + 2\zeta\omega \frac{dq}{dt} + \omega^2 q = \mathbf{0}, \quad (3.35)$$

where $\omega^2 = \frac{k}{m}$ is the undamped frequency and $2\omega\zeta = \frac{c}{m}$ is the damping ratio.

By identification of eq. (3.34) with eq. (3.35):

$$\omega = \sqrt{\frac{k_s + k_f}{m_s + m_f}}, \quad (3.36)$$

$$\zeta = \frac{c_s + c_f}{2\omega(m_s + m_f)}. \quad (3.37)$$

3.1.7.3 Multiple degrees of freedom

When a system has multiple degrees of freedom, coupling occurs between the different directions.

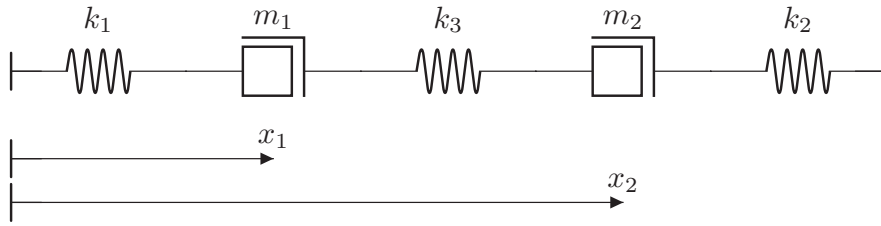


Figure 3.19 – Coupling by stiffness.

$$\begin{cases} m_1 \ddot{x}_1 + (k_1 + k_3)x_1 - k_3 x_2 = f_1, \\ m_2 \ddot{x}_2 + (k_2 + k_3)x_2 - k_3 x_1 = f_2. \end{cases} \quad (3.38)$$

The system eq. (3.38) depicted in fig. 3.19 shows a symmetric stiffness cross-coupling term $k_3 x_{1,2}$:

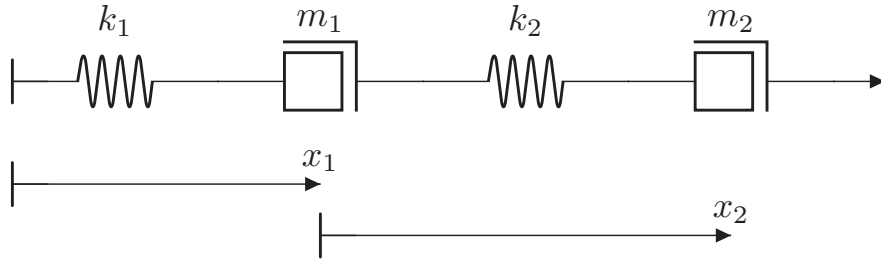


Figure 3.20 – Coupling by inertia.

$$\begin{cases} (m_1 + m_2) \ddot{x}_1 + k_1 x_1 - m_2 \ddot{x}_2 = f_1 + f_2, \\ m_2 \ddot{x}_2 + k_2 x_2 + m_2 \ddot{x}_1 = f_2. \end{cases} \quad (3.39)$$

The system eq. (3.39) depicted in fig. 3.20 shows a symmetric inertial cross-coupling term $m_2 \ddot{x}_{1,2}$.

Uncoupled system by operating a change of variables $a_1 = (x_2 + x_1) / 2$; $a_2 = (x_2 - x_1) / 2$; $m = m_1 = m_2$; $k = k_1 = k_2$ in the system eq. (3.38):

$$\begin{cases} m\ddot{a}_1 + ka_1 = (f_1 + f_2) / 2, \\ m\ddot{a}_2 + (k + 2k_3)a_2 = (f_2 - f_1) / 2. \end{cases} \quad (3.40)$$

The equivalent system eq. (3.40) shows no more cross-coupling term. This simple example demonstrates that the real coupling term lies in the natural frequencies ($f_1 = \frac{1}{2\pi} \sqrt{\frac{k}{m}}$ and $f_2 = \frac{1}{2\pi} \sqrt{\frac{k+2k_3}{m}}$) and the normal modes, more than the mathematical modelling of the system.

3.2 Discretization

The discrete approximation of $PDE(s)$ ⁱ is a pre-requisite step required before evaluation of a numerical solution. The computer architectures are sequential and can be viewed as a finite-state machine. We thus need to define discrete variables and more generally a discrete framework with the idea in mind that discretization always induces some error in the solution.

3.2.1 Finite Volume Method

In order to obtain a numerical solution, one must transpose the continuous equations into their discrete counterparts, in time (Euler stepping for example) or in space (unstructured grid with arbitrary volumes, structured grid for the volume of fluid method, ...). Tessellating the euclidean space has consequences on the numerical solution. On one hand, poorly resolved areas with strong derivatives lead to large errors. On the other hand, excessive meshing in smooth regions is costly and does not improve global accuracy of the solution. We use the *FVM*ⁱⁱ on a Cartesian mesh, with *MAC* convention [127], where fluxes are located on cube facets and intensive variables are located at cell centres. The *MAC* method uses Finite Difference(s) to approximate the Navier-Stokes Equation(s). A staggered mesh is used for the velocities, and the scalar pressure p or the density ρ are evaluated at cell centres as seen on fig. 3.21. This method enhances velocity-pressure decoupling and is known to avoid checker-board patterns induced by odd-even decoupling on dependent variables.

The convention taken for the conservation laws is a positive outflow $\mathbf{u} \cdot \mathbf{n} > 0$, with the corresponding outward pointing normal.

The net inflow through the boundary of the control volume is thus $-\oint_S \rho \mathbf{u} \cdot \mathbf{n} ds$. For example, the average pressure gradient on a single cell is:

$$\nabla_h p = \frac{1}{V} \iiint_V \nabla p dv = \frac{1}{V} \oint_S p \mathbf{n} ds.$$

The momentum equation with the full stress tensor reads:

$$\frac{\partial \rho \mathbf{u}}{\partial t} + \nabla \cdot (\rho \mathbf{u} \otimes \mathbf{u}) = -\nabla p + \nabla \cdot \mu (\nabla \mathbf{u} + \nabla^t \mathbf{u}) + \rho \mathbf{f}.$$

We now define control volume averages for the diffusive \mathbf{D} and the convective operators \mathbf{A} :

$$\begin{aligned} \mathbf{D} &= \frac{1}{V} \iiint_V \nabla \cdot \mu (\nabla \mathbf{u} + \nabla^t \mathbf{u}) dv = \frac{1}{V} \oint_S \mu (\nabla \mathbf{u} + \nabla^t \mathbf{u}) \cdot \mathbf{n} ds, \\ \mathbf{A} &= \frac{1}{V} \iiint_V \nabla \cdot (\rho \mathbf{u} \otimes \mathbf{u}) dv = \frac{1}{V} \oint_S (\rho \mathbf{u} \otimes \mathbf{u}) \cdot \mathbf{n} ds. \end{aligned}$$

i. Partial Differential Equation(s)

ii. Finite Volume Method

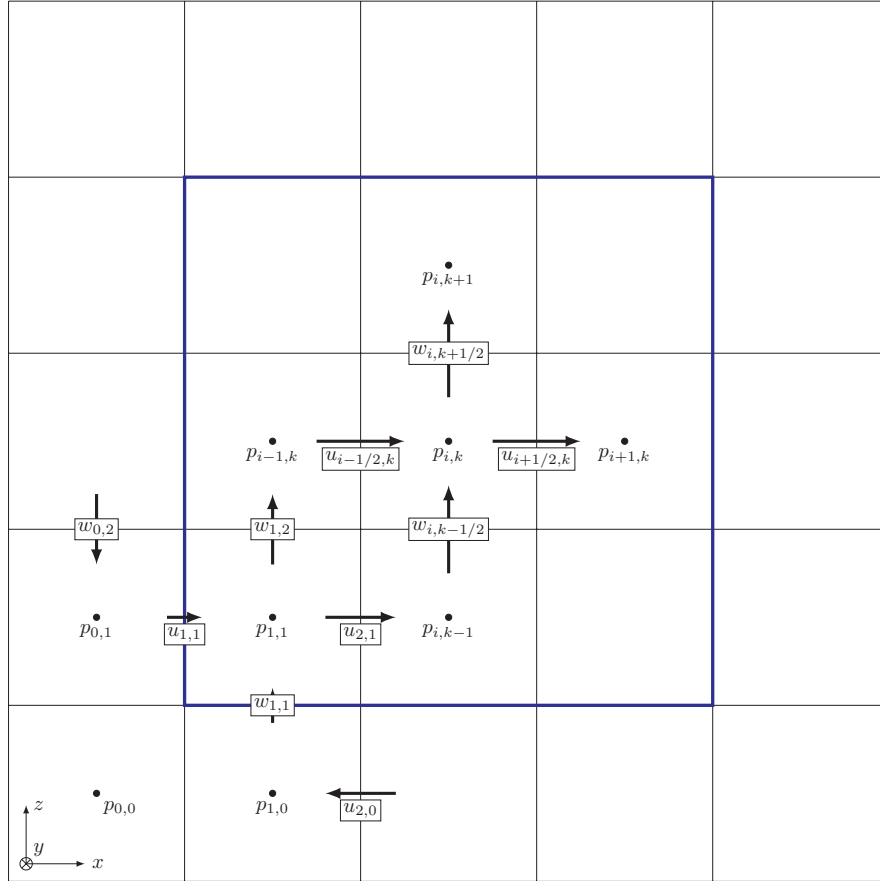


Figure 3.21 – Bi-dimensional grid quantities, and boundary shadow variables.

The discretization of these operators is discussed later in sections 3.3.5.2 and 3.3.5.3.

3.2.2 Numerical boundary conditions

Any physical boundary condition (Periodic, Inflow, Outflow, Slip (no adherence), Wall (adherence, no-slip)) is mapped to a numerical boundary condition (copy, reflection, mirroring, periodic, ...). For example for outflow on the x axis, we impose $\frac{\partial u}{\partial x} = 0$ by copying the last velocity in the x direction to the wall velocity, so that the first order gradient vanishes. For the vertical velocity w in the x direction, we wish to impose $\frac{\partial w}{\partial x} = 0$: we thus copy the last velocity in the domain onto the ghost layer vertical velocity.

Since the flow is considered viscous, velocities across the interface are continuous, hence:

$$[u] = [v] = [w] = 0. \quad (3.41)$$

No specific treatment is applied to velocities near the interface. In *DNS*, if the resolution is sufficient for velocity boundary layers, then the velocity transition across the interface is considered continuous enough.

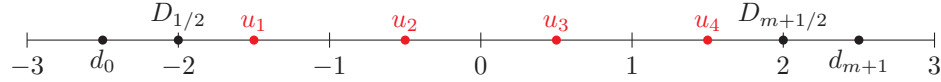


Figure 3.22 – One-dimensional domain unknowns and shadow variables.

We will now expose the numerical boundary conditions needed for the linear operators in the implicitation phase. For simplicity, we will take a one dimensional discretization with $m = 4$ unknowns: $u_{i=1..4}$, as sketched in fig. 3.22.

We consider the discretization of the Laplace operator:

$$\Delta \mathbf{u} = \mathbf{f}, \quad (3.42)$$

and the associated Poisson equation with a *RHS* \mathbf{f} .

The equation for u_1 reads:

$$\frac{u_2 - 2u_1 + u_0}{\Delta x^2} = f_1 = (u_{xx})_i. \quad (3.43)$$

Neumann

The flux at point $i = 1 / 2$ reads:

$$(u_x)_{1/2} = \frac{u_1 - u_0}{\Delta x} = g(u_{1/2}). \quad (3.44)$$

Then, rewriting eq. (3.43) and using eq. (3.44) yields:

$$\begin{aligned} f_1 &= \frac{u_2 - u_1 - (u_1 - u_0)}{\Delta x^2} = \frac{u_2 - u_1 - g(u_{1/2})\Delta x}{\Delta x^2}, \\ \implies u_2 - u_1 &= \Delta x^2 f_1 + g(u_{1/2})\Delta x. \end{aligned}$$

Similarly, the flux located at point $i = m + 1 / 2$ is:

$$(u_x)_{m+1/2} = \frac{u_{m+1} - u_{m-1}}{\Delta x} = g(u_{m+1/2}).$$

So that:

$$\begin{aligned} f_m &= \frac{(u_{m+1} - u_m) - u_M + u_{m-1}}{\Delta x^2} = \frac{g(u_{m+1/2})\Delta x - u_m + u_{m-1}}{\Delta x^2} \\ \Rightarrow u_{m-1} - u_m &= \Delta x^2 f_m - g(u_{m+1/2})\Delta x. \end{aligned}$$

In matrix formulation, the eq. (3.42) with Neumann $BC(s)$ results in:

$$\begin{pmatrix} -1 & 1 & & & \\ 1 & -2 & 1 & & \\ & & \ddots & & \\ & & & 1 & -2 & 1 \\ & & & & \ddots & \\ & & & & & 1 & -2 & 1 \\ & & & & & & 1 & -1 \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \\ \vdots \\ u_i \\ \vdots \\ u_{m-1} \\ u_m \end{pmatrix} = \begin{pmatrix} \Delta x^2 f_1 + \Delta x g(u_{1/2}) \\ \Delta x^2 f_2 \\ \vdots \\ \Delta x^2 f_i \\ \vdots \\ \Delta x^2 f_{m-1} \\ \Delta x^2 f_m - \Delta x g(u_{m+1/2}) \end{pmatrix}.$$

Dirichlet

For Dirichlet $BC(s)$, u_0 is imposed so eq. (3.43) becomes:

$$\frac{u_2 - 2u_1}{\Delta x^2} = f_1 - \frac{u_0}{\Delta x^2}.$$

In matrix formulation, we thus have:

$$\begin{pmatrix} -2 & 1 & & & \\ 1 & -2 & 1 & & \\ & & \ddots & & \\ & & & 1 & -2 & 1 \\ & & & & \ddots & \\ & & & & & 1 & -2 & 1 \\ & & & & & & 1 & -2 \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \\ \vdots \\ u_i \\ \vdots \\ u_{m-1} \\ u_m \end{pmatrix} = \begin{pmatrix} \Delta x^2 f_1 - d(u_0) \\ \Delta x^2 f_2 \\ \vdots \\ \Delta x^2 f_i \\ \vdots \\ \Delta x^2 f_{m-1} \\ \Delta x^2 f_m - d(u_{m+1}) \end{pmatrix},$$

Dirichlet midpoint

$u_{1/2}$ is the prescribed fictitious unknown: $u_{1/2} = \frac{u_0 + u_1}{2}$, meaning that:

$$u_0 = 2u_{1/2} - u_1. \quad (3.45)$$

Using eq. (3.43) and eq. (3.45) leads to:

$$\frac{u_2 - 2u_1 + (2u_{1/2} - u_1)}{\Delta x^2} = f_1,$$

and thus $u_2 - 3u_1 = \Delta x^2 f_1 - 2u_{1/2}$.

As a result:

$$\begin{pmatrix} -3 & 1 & & & & \\ 1 & -2 & 1 & & & \\ & & \ddots & & & \\ & & & 1 & -2 & 1 \\ & & & & \ddots & \\ & & & & 1 & -2 & 1 \\ & & & & & 1 & -3 \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \\ \vdots \\ u_i \\ \vdots \\ u_{m-1} \\ u_m \end{pmatrix} = \begin{pmatrix} \Delta x^2 f_1 - 2D(u_{1/2}) \\ \Delta x^2 f_2 \\ \vdots \\ \Delta x^2 f_i \\ \vdots \\ \Delta x_{m-1}^2 \\ \Delta x^2 f_m - 2D(u_{m+1/2}) \end{pmatrix}.$$

Periodic cell centred

With periodic $BC(s)$, the linear system becomes:

$$\begin{pmatrix} -2 & 1 & & & & 1 \\ 1 & -2 & 1 & & & \\ & & \ddots & & & \\ & & & 1 & -2 & 1 \\ & & & & \ddots & \\ & & & & 1 & -2 & 1 \\ 1 & & & & & 1 & -2 \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \\ \vdots \\ u_i \\ \vdots \\ u_{m-1} \\ u_m \end{pmatrix} = \begin{pmatrix} \Delta x^2 f_1 \\ \Delta x^2 f_2 \\ \vdots \\ \Delta x^2 f_i \\ \vdots \\ \Delta x^2 f_{m-1} \\ \Delta x^2 f_m \end{pmatrix}.$$

3.2.3 Topological source terms

Curvatures

Several curvatures can be defined on a surface of \mathcal{R}^3 :

- the normal curvature k ;
- the principal curvatures k_{min} and k_{max} ;
- the mean curvature κ ;
- the gaussian curvature G .

We first define the shape operator C as the directional derivative of the normal vector in the direction of a tangent vector [128]:

$$C(\mathbf{t}(p)) = D_t(\mathbf{n}(p)). \quad (3.46)$$

It is a measure of how the surface bends in \mathcal{R}^3 .

The normal curvature is dependent on the direction of the tangent plane. The normal curvature is the dot product between the derivative of the normal vector to the surface (unique) and a chosen tangent vector, at a point $p \in \mathcal{S}$.

We can now define the principal curvatures as the maximum (respectively minimum) of the normal curvature along all the planes tangent to the surface $\mathcal{S} \subset \mathcal{R}^3$.

The gaussian curvature G is the determinant of the shape operator eq. (3.46): $G(p) = \det(C(p))$, or the product of the two principal curvatures $G = k_{min} \cdot k_{max}$.

The mean curvature κ is a function of the trace of the shape operator $\kappa(p) = \frac{1}{2}\text{trace}(C(p))$, or the arithmetic average of the principal curvatures: $\kappa = \frac{1}{2}(k_{min} + k_{max})$.

As opposed to the gaussian curvature, the sign of the mean curvature depends on the convention chosen for the normal to the surface (pointing inwards or outwards).

The parametric representation of \mathcal{S} is defined by the vector mapping $\mathbf{f} : \mathcal{R}^2 \rightarrow \mathcal{R}^3$:

$$\mathbf{f} : (u, v) \mapsto (x(u, v), y(u, v), z(u, v)). \quad (3.47)$$

For surface tension applications, we use the arithmetic mean curvature. On a cylinder surface, the gaussian curvature is null, since one of the principal curvatures is null (the curvature is either zero-valued or positive on a cylinder). The mean curvature, on the other hand is non-zero on the cylinder surface, allowing a non-zero capillary force. The curvature must also be invariant by rotation. Thus, the correct choice for the curvature in capillary applications is the mean curvature between the principal curvatures.

Numerical implementation

The surface force acting on the interface at a microscopic level must be extrapolated to a macroscopic term at the simulation scale, hence on the control volume.

The CSF ⁱⁱⁱ model [74] expresses the interfacial surface force term as $\mathbf{f}_s = \sigma \kappa \mathbf{n}$, where σ is the surface tension coefficient, κ the interfacial mean curvature and \mathbf{n} the unit normal to the interface surface. The singular surface force is spread onto a volumetric force by the mean of a delta function (Dirac mass) centred on the interface.

Since the interface is implicitly tracked, the interfacial term must be spread out around the interface locus. The interfacial term is thus spread out onto a volumetric term:

$$\mathbf{F}_v = \sigma \kappa \delta \mathbf{n}, \quad (3.48)$$

where δ should represent a smooth pulsing term around the interface with unit $[L^{-1}]$.

The delta function can be expressed as

$$\delta = \left| \frac{\nabla \bar{\rho}}{[\rho]} \right|,$$

where $\bar{\rho}$ is the smoothed density used as a colour function, and $[\rho]$ represents the global density jump between the two phases [129, 130]. Quasi-discontinuous density can also be used: $\delta = \left| \frac{\nabla \rho}{[\rho]} \right|$.

iii. Continuum Surface Force

On the other hand, if σ is spatially variable, as it can be the case in surfactant solutions or when considering the Marangoni effect [131], a more elaborate model is to be used, such as the CSS^{iv} method [28, 75].

The normal of a surface is the unit vector to a surface in \mathcal{R}^3 . For closed surfaces, we shall use the outward pointing convention. The unit normal to a surface S parametrized by its spatial coordinates x, y, z is the gradient of the surface, normalized by its length. The metrics used in the \mathcal{L}^2 space is the euclidean distance, provided by the ℓ^2 norm of the vector.

$$\mathbf{n} = \begin{pmatrix} \nabla_x S(x, y, z) \\ \nabla_y S(x, y, z) \\ \nabla_z S(x, y, z) \end{pmatrix} / \|\nabla S\|_2,$$

where $\nabla(\cdot)$ is the gradient along a specific direction of the euclidean space.

The mean curvature κ is defined as the divergence of the unit normal:

$$\kappa = \nabla \cdot \mathbf{n}. \quad (3.49)$$

Hence, to compute the normal of the implicit surface describing the two phases, we must capture the interface position. Since the high order scheme has a local compressive effect, the density is unreliable, and must be regularized, as discussed in section 3.1.3.3.

Practically, one must pay attention to the consistency between gradient when evaluating curvatures, since the pressure gradient shall balance formally the capillary forces. Any inconsistency between discretization will lead to parasitic numerical flow, as resumed in [132]. The following computational sequence is found to be consistent and stable:

- compute vertices \mathbf{n}_v and cells \mathbf{n}_c normals using regularized density $\bar{\rho}$;
- compute facet curvatures κ_f from \mathbf{n}_v ;
- compute the interfacial facet pulse $\delta_f := \left| \frac{\nabla \rho}{|\rho|} \right| \text{interp}(\mathbf{n}_c)$, using un-regularized density ρ (integral distribution of the source term), and orient the vector using interpolation of \mathbf{n}_c ;
- compute the facet capillary source term $\mathbf{C}_f := \sigma \kappa_f \delta_f$.

3.2.4 Numerical stability conditions for the Navier-Stokes equations

Numerical approximation of eq. (3.6) requires stability conditions for timestepping, imposed by multiple contributions of the different parts of the equation. The stability of the full set of the Navier-Stokes Equation(s) is usually too impractical to be performed. When the flow is dominated by convective terms, the $NSE(s)$ degenerate into hyperbolic equations. On the opposite, the parabolic part remains when the flow is dominated by diffusion (the tendency of a system to become uniform and homogeneous). The stability

iv. Continuous Surface Stress

of the $NSE(s)$ is assessed using the Von Neumann stability analysis [133], which consists of disrupting the set of equations with a small parameter ϵ , and study the decay or growth of the error. Stability requires decay of the error, which implies that the amplification factor must remain bounded by unity.

Convective and viscous constraints

We consider the one-dimensional advection-diffusion equation:

$$\frac{\partial f}{\partial t} + u \frac{\partial f}{\partial x} = d \frac{\partial^2 f}{\partial x^2}. \quad (3.50)$$

We advect and diffuse a small perturbation in the form $\varepsilon_j^n := f_j^n$.

Injecting the previous equation in the discrete counterpart of eq. (3.50) yields:

$$\frac{\varepsilon_j^{n+1} - \varepsilon_j^n}{\Delta t} + u \frac{\varepsilon_{j+1}^n - \varepsilon_{j-1}^n}{2\Delta x} = d \frac{\varepsilon_{j+1}^n - 2\varepsilon_j^n + \varepsilon_{j-1}^n}{(\Delta x)^2}.$$

We can expand the error as a Fourier series: $\varepsilon_j^n = \varepsilon^n(x_j) = \sum_{k=-\infty}^{+\infty} \varepsilon_k^n e^{i.kx_j}$, giving:

$$\frac{\varepsilon^{n+1}}{\varepsilon^n} = 1 - 4 \frac{d\Delta t}{(\Delta x)^2} \sin^2\left(k \frac{\Delta x}{2}\right) - i \frac{u\Delta t}{\Delta x} \sin(k\Delta x).$$

The stability condition requires that the error decreases i.e. $\left| \frac{\varepsilon^{n+1}}{\varepsilon^n} \right| \leq 1$.

Consequently, for a one-dimensional problem, the stability of the numerical scheme requires that:

$$4 \frac{d\Delta t}{\Delta x^2} \leq 2 \quad \quad \quad \frac{u\Delta t}{d} \leq 2.$$

For a bi-dimensional problem, we assume that the error can be expressed in the form $\varepsilon_{i,k}^n = \varepsilon^n e^{i(\lambda_x x_i + \lambda_z z_k)}$:

$$4d\Delta t \frac{\Delta x^2 + \Delta z^2}{\Delta x^2 \Delta z^2} \leq 2 \left(\begin{array}{c} \implies \\ \Delta x = \Delta z \\ = \Delta h \end{array} \frac{d\Delta t}{\Delta h^2} \leq \frac{1}{4} \right) \quad \frac{(|u| + |w|)^2 \Delta t}{d} \leq 4.$$

For a three-dimensional problem, the analysis is similar:

$$4d\Delta t \frac{\Delta y^2 \Delta z^2 + \Delta x^2 \Delta z^2 + \Delta x^2 \Delta y^2}{\Delta x^2 \Delta y^2 \Delta z^2} \leq 2 \left(\begin{array}{c} \implies \\ \Delta x = \Delta y \\ = \Delta z = \Delta h \end{array} \frac{d\Delta t}{\Delta h^2} \leq \frac{1}{6} \right) \quad \frac{(|u| + |v| + |w|)^2 \Delta t}{d} \leq 8.$$

The convective restriction can be written as follows:

$$\Delta t \left(\frac{|u|_{max}}{\Delta x} + \frac{|v|_{max}}{\Delta y} + \frac{|w|_{max}}{\Delta z} \right) < \alpha, \quad (3.51)$$

where $\alpha \in]0; 1[$ is a chosen criterion.

Source term constraint

If any source term is included (topological, gravity), then we can rewrite eq. (3.51) as follows:

$$\Delta t \left(\frac{|u|_{max} + \Delta t |C_x|}{\Delta x} + \frac{|v|_{max} + \Delta t |C_y|}{\Delta y} + \frac{|w|_{max} + \Delta t |C_z| + g \Delta t}{\Delta z} \right) < \alpha,$$

where gravity is oriented in the z direction and C_x, C_y, C_z represent capillary forces.

This leads to a quadratic equation on Δt giving the following time step stability bound [134]:

$$\begin{aligned} \Delta t & \left(\frac{|u|_{max}}{\Delta x} + \sqrt{\left(\frac{|u|_{max}}{\Delta x} \right)^2 + \frac{4(|g_x| + |C_x|)}{\Delta x}} + \frac{|v|_{max}}{\Delta y} \right. \\ & \left. + \sqrt{\left(\frac{|v|_{max}}{\Delta y} \right)^2 + \frac{4(|g_y| + |C_y|)}{\Delta y}} + \frac{|w|_{max}}{\Delta z} + \sqrt{\left(\frac{|w|_{max}}{\Delta z} \right)^2 + \frac{4(|g_z| + |C_z|)}{\Delta z}} \right) < 2\alpha. \end{aligned}$$

Capillary constraint

The expression of the capillary force is of order $\sigma \kappa \delta \rho^{-1}$, using the formulation in section 3.1.3.1.

A minimal computational bound for the curvature would be an inclusion with a mesh size in all directions. In this case, the curvature is $\kappa_{max} = .5 \left(\frac{1}{\Delta x} + \frac{1}{\Delta z} \right) < \min \left(\frac{1}{\Delta x}, \frac{1}{\Delta z} \right)$.

The numerical spread of the source term is of order of the mesh size, hence $\delta \simeq \min(\Delta x, \Delta z)$.

Consequently, the capillary criterion in two dimensions can be approached by:

$$\text{crit}_{\text{cap}}^2 = \frac{\sigma \kappa_{max}}{\min(\rho_1, \rho_2) \min(\Delta x, \Delta z)^2}.$$

As exposed in [89], the explicit treatment of surface usually requires a restrictive time-stepping. Generally, an explicit numerical computation is stable when the ratio of the physical velocity to the numerical velocity is smaller than unity. Hence, the numerical algorithm must resolve capillary wave which is of order:

$$\sqrt{\frac{\sigma k}{\rho_1 + \rho_2}},$$

with $k = \frac{2\pi}{\lambda}$ the wave number.

We can then write:

$$\Delta t \text{ crit}_{\text{cap}} < 1 \implies \text{crit}_{\text{cap}} = \sqrt{\frac{(2\pi)^3 \sigma}{(\rho_1 + \rho_2) \min(\Delta)^3}}. \quad (3.52)$$

A second stability criterion can be retrieved from [134]. From our experiments, it seems that the eq. (3.52) can be alleviated by the regularization of the curvatures.

Combined constraint

The convective and viscous stability constraints can be merged into a single inequality:

$$\Delta t \left((\text{crit}_{\text{conv}} + \text{crit}_{\text{visc}}) + \sqrt{(\text{crit}_{\text{conv}} + \text{crit}_{\text{visc}})^2 + 4\text{crit}_{\text{src}}^2 + 4\text{crit}_{\text{cap}}^2} \right) < 2\alpha, \quad (3.53)$$

where $\text{crit}_{\text{src}} = \sqrt{\sum_{i=1}^3 \frac{|g_i|}{\Delta_i}}$ is the upper bound criterion of entrainment terms induced by gravity and crit_{cap} represents the upper limit for capillary terms. A safety factor $\alpha \in]0; 1[$ is left to the choice of the user.

3.3 Numerical methods

After having defined the discretization strategy and context, we develop the numerical methods for fluid and structure problems. In this part, we aim to develop a numerical model for the transport of two-phase flows, using a continuous description for the two phases. Hence, there is no disperse treatment of the secondary phase (Lagrangian treatment of the disperse phase [135]), which should be resolved up to the relevant scale.

3.3.1 Anti-diffusive approach for conservative scalar transport

Keeping track of the exact position of a multi-component fluid interface through time is easy when the numerical description of the interface is explicit. However, following the interface moving through time with an explicit representation is quite costly for example with *FT*, when the change of topology requires to balance the number of points. Furthermore, this method does not scale well since the number of points required to describe the interface follows the density of inclusions. In the context of superimposed meshes such as *ALE*, a re-meshing process induces mesh quality issues as well as increased memory and computing costs induced by the re-meshing process. The issue of coalescence and breakup which are areas requiring physical interpretation and modelling up to finer scales comes also in mind. Another family of methods is the interface capturing approach where the interface is described either by an additional field (in which the interface is implicit), or by an additional equation solving for the localization as with the *LSM*ⁱ. The *VOF* method reconstructs the interface within each cell, by targeting a conservation of volume by compensation of the diffusive term, and thus induces additional computing costs for accurate volume conservation. The goal sought by using this numerical scheme is to bound over time the thickening mode of the interface imposed by the numerical diffusion of the interface. Since no chemical reaction is solved, the effect of the width of the transition layer is thought to be negligible for boundary layers.

3.3.1.1 Flux limiter formalism

Flux limited schemes were widely studied between the sixties and the eighties and are derived from *FCT*ⁱⁱ schemes, where a flux correction stage follows the low order advection phase [136].

We will first recall some classic schemes encountered for numerical approximation of *PDE(s)*. This will help in the understanding of numerical errors between the continuous solution and the discrete approximation, when constructing a flux limiting scheme. We consider the following one-dimensional first order hyperbolic equation (linear advection) [137] in order to construct the scalar transport:

$$\frac{\partial f}{\partial t} + u \frac{\partial f}{\partial x} = 0, \quad (3.54)$$

i. Level-Set Method(s)

ii. Flux Corrected Transport

where u is a constant velocity. For this equation, the solution $f(x, t)$ is constant along the characteristics lines in the range of influence cone.

We note that the conservative non-linear form of eq. (3.54) reads:

$$\frac{\partial f}{\partial t} + \frac{\partial \varphi(f)}{\partial x} = 0, \quad (3.55)$$

where $\varphi(f)$ represents the flux across the control volume facets.

We expand $\frac{\partial f}{\partial t}$ up to the second order in time using Taylor series:

$$f_i^{n+1} = f_i^n + \Delta t (f_t)_i^n + \frac{\Delta t^2}{2} (f_{tt})_i^n + \mathcal{O}(\Delta t^3),$$

and replace time derivatives with spatial derivatives since eq. (3.54) implies $f_t = -u f_x$:

$$f_i^{n+1} = f_i^n - u \Delta t (f_x)_i^n + \frac{u^2 \Delta t^2}{2} (f_{xx})_i^n + \mathcal{O}(\Delta t^3).$$

We define $\varsigma = u \frac{\Delta t}{\Delta x}$ to be the Courant number associated to the CFL ⁱⁱⁱ condition [138].

We also introduce the concept of monotonicity-preserving schemes, which satisfy the following property:

$$f_{i+1}^n \geq f_i^n \implies f_i^{n+1} \geq f_i^{n+1} \forall i.$$

Moreover, a scheme is said to be linear if it can be written as a linear combination of its arguments, hence:

$$f_i^{n+1} = \sum_k u_k \cdot f_{i+k}^n.$$

When $u_k \geq 0 \forall k$, the scheme is said to be monotone. As an example, the LF ^{iv} scheme [139]:

$$f_i^{n+1} = \frac{1}{2} (f_{i-1}^n + f_{i+1}^n) - \frac{\varsigma}{2} (f_{i+1}^n - f_{i-1}^n),$$

which is forward in time and centred in space with first order accuracy, is monotone under the CFL condition.

Theorem 1 (Godunov [140])

A linear, monotonicity-preserving scheme is at most first order accurate.

This theorem exposes the issue usually encountered, since higher order scheme tends to develop oscillations in non-smooth loci (and thus loose the TVD ^v property), whereas first order scheme will tend to show excessive dissipative behaviour. Thus, in order to

iii. Courant-Friedrichs-Lewy

iv. Lax-Friedrichs

v. Total Variation Diminishing

conjugate higher order accuracy and monotonicity, non-linear scheme must be developed, in the sense that the algorithm toggles (non-linear operation) between different behaviours, based on a chosen criterion.

We now introduce the following schemes, which will provide the base state for the high order method.

The LW^{vi} scheme [141] is obtained using second order central $FD(s)^{\text{vii}}$ for the spatial derivatives,

$$(f_x)_i = \frac{f_{i+1} - f_{i-1}}{2\Delta x}, (f_{xx})_i = \frac{f_{i+1} - 2f_i + f_{i-1}}{\Delta x^2},$$

resulting in:

$$f_i^{n+1} = f_i^n - \frac{\varsigma}{2}(f_{i+1}^n - f_{i-1}^n) + \frac{\varsigma^2}{2}(f_{i+1}^n - 2f_i^n + f_{i-1}^n). \quad (3.56)$$

The BM^{viii} scheme [142] is obtained using upwind differences for the spatial derivatives (second, and first order of accuracy):

$$(f_x)_i = \frac{3f_i - 4f_{i-1} - f_{i-2}}{2\Delta x}, (f_{xx})_i = \frac{f_{i-2} - 2f_{i-1} + f_i}{\Delta x^2}.$$

$$f_i^{n+1} = f_i^n - \frac{\varsigma}{2}(3f_i^n - 4f_{i-1}^n - f_{i-2}^n) + \frac{\varsigma^2}{2}(f_i^n - 2f_{i-1}^n + f_{i-2}^n). \quad (3.57)$$

We now rewrite the LW scheme eq. (3.56) forcing the appearance of the low order upwind term $f_i^n - f_{i-1}^n$:

$$f_i^{n+1} = f_i^n - \underbrace{\varsigma(f_i^n - f_{i-1}^n)}_{\text{low order upwind term}} + \overbrace{\left[-\frac{\varsigma(1-\varsigma)}{2}f_{i+1}^n + \varsigma(1-\varsigma)f_i^n - \frac{\varsigma(1-\varsigma)}{2}f_{i-1}^n \right]}^{\text{correction}}. \quad (3.58)$$

The equation eq. (3.58) is now rewritten in the conservative form:

$$f_i^{n+1} = f_i^n - \frac{\Delta t}{\Delta x} (\varphi(f^n, i + 1/2) - \varphi(f^n, i - 1/2)),$$

where $\varphi(f^n, i \pm 1/2)$ is the numerical flux on facets.

This yields the expression of the numerical flux:

$$\varphi(f^n, i + 1/2) = \underbrace{u \cdot f_i^n}_{\text{low order upwind flux}} + \overbrace{\frac{1}{2}u(1-\varsigma)(f_{i+1}^n - f_i^n)}^{\text{high order facet flux}}. \quad (3.59)$$

vi. Lax-Wendroff

vii. Finite Difference(s)

viii. Beam-Warming

A weight is then applied to the correction term in order to change the scheme behaviour based on a measure of smoothness. A common measure is the ratio of consecutive gradients:

$$r_i = \frac{f_i^n - f_{i-1}^n}{f_{i+1}^n - f_i^n},$$

so that the non-linearity induced can be written as $\mathfrak{L}(r)$. When the ratio r approaches unity, then f is sufficiently smooth. On the contrary, a zero valued ratio tends to indicate some discontinuity in f .

With the additional term $\mathfrak{L}(r)$, the equation eq. (3.59) becomes:

$$\varphi(f^n, i + 1 / 2) = u \cdot f_i + \frac{1}{2}u(1 - \varsigma)\mathfrak{L}(r)(f_{i+1}^n - f_i^n). \quad (3.60)$$

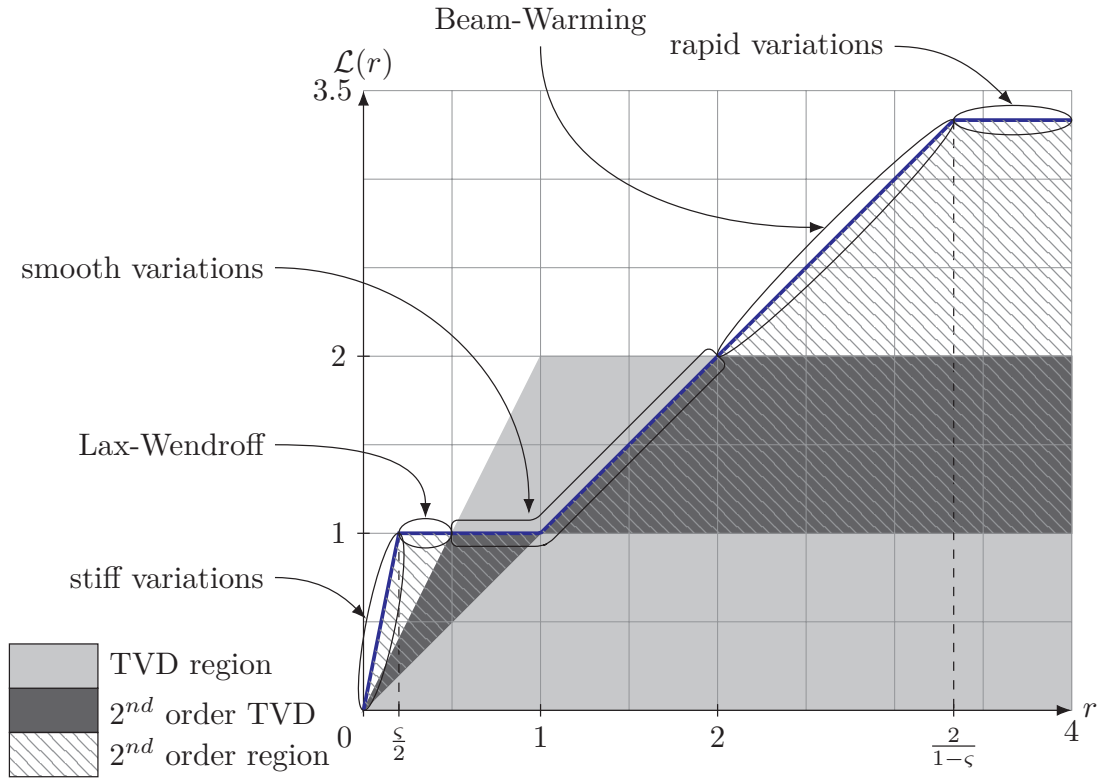


Figure 3.23 – Smoothness regions of the flux limiter function, for an arbitrary local CFL ς .

Local constraints over the non-linear mapping are derived in [143].

Some desirable properties of a numerical scheme are high resolution to counteract excessive numerical diffusion, and oscillation free schemes, meaning that the numerical solution obtained does not take non-physical values for example when the solution is

close to zero and negative values are neither allowed by the model nor by the numerical method. Although preserving the discontinuity of solutions is key, numerical methods fail to do so, due to numerical diffusion. A significant number of work has been conducted in order to find exact numerical solutions. For example, in shock computations (the state evolves towards discontinuous solution) [144] or in the generic case of phase transition for a general $PDE(s)$, such as the toggling between congested and free flow in urban traffic modelling [145]. The method of [146] is appealing because it is toggling between full jump or no increment at the discontinuity [147]. The solution is discontinuous (piecewise linear), and despite the nature of the solution, the integral properties are preserved [148].

3.3.1.2 Schemes comparisons

The test case presented in [149] exposes a test case with representative numerical issues (usually encountered in concrete situations):

- a smooth varying shape, with continuous derivatives (e.g. gaussian profile);
- a constant derivative shape (cone), with a two valued derivative at the top point;
- a slotted disk, with a clean discontinuity and thin transition space.

The first item is interesting in order to study the compressive behaviour of these schemes. For the second one, we focus on the conservation of local extrema, since the singularity could represent the tip of a distorted bubble in our case. It is key here, to be able to conserve physical properties such as density of the two phases: primary ρ_1 and secondary ρ_2 . The third item is representative of a stiff interface between liquid and gaseous phase. The theory predicts a numerical diffusion of one or two points [44, 46]. Four schemes are compared: a diffuse stable upwind scheme, and three flux limiter based schemes as seen on fig. 3.24.

What we can infer from fig. 3.24 is that we allow a wider region for the UB^{ix} and NB^x schemes. The resulting scheme is outside the $TV D$ region ($0 \leq \mathfrak{L}(r) \leq \min(2r, r)$), which results in reducing the smoothness of the solution for initially smooth solutions. However, for initial discontinuous profiles (shocks, phase transition), this allows the conservation of the interface locus over numerous timesteps with limited amount of numerical diffusion.

We consider the following one-dimensional equation, with appropriate initial conditions:

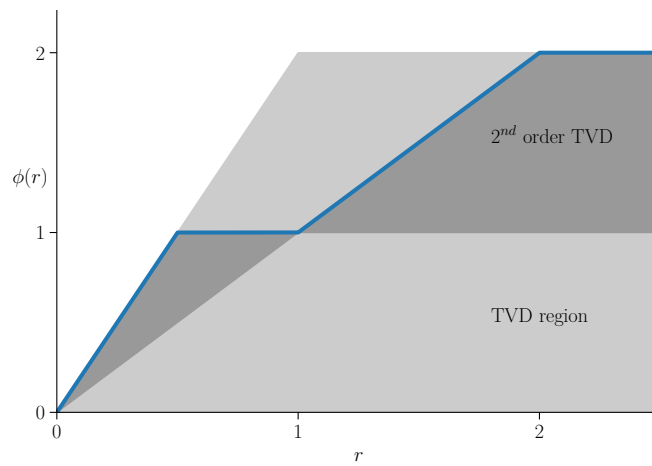
$$\begin{cases} f_t + u(x)f_x = 0, \\ f(t = 0, x) = f_0(x). \end{cases} \quad (3.61)$$

The space and time discrete sequences are written as follows: $x_i = i\Delta x$ and $t_n = n\Delta t$. The discrete approximation of eq. (3.61) reads:

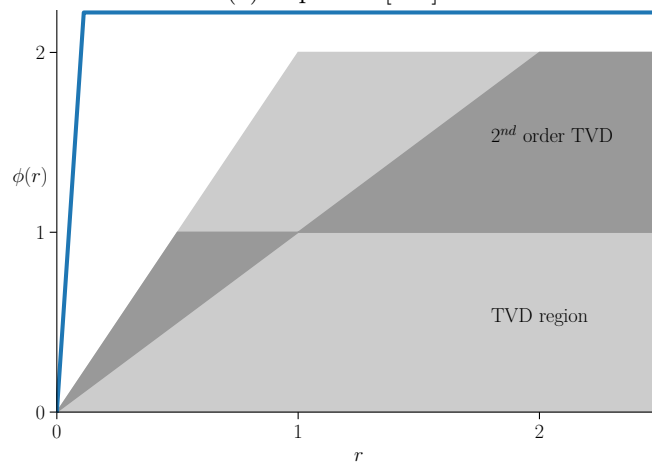
$$\frac{f_i^{n+1} - f_i^n}{\Delta t} + u(x_i) \frac{F_{i+1/2}^{n,L} - F_{i-1/2}^{n,R}}{\Delta x} = 0.$$

ix. Ultra-BEE

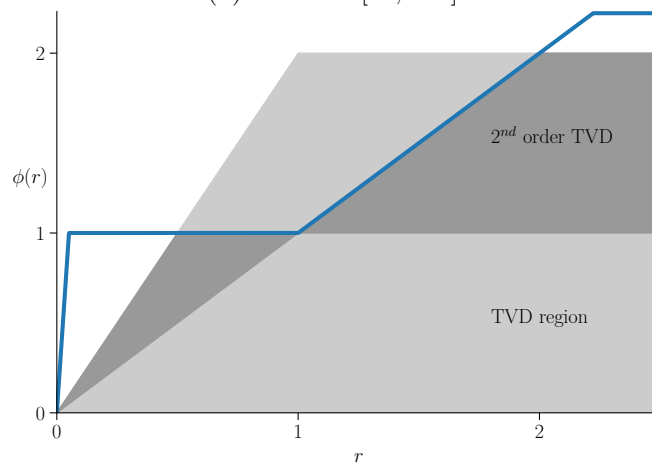
x. N-BEE



(a) Superbee [150].

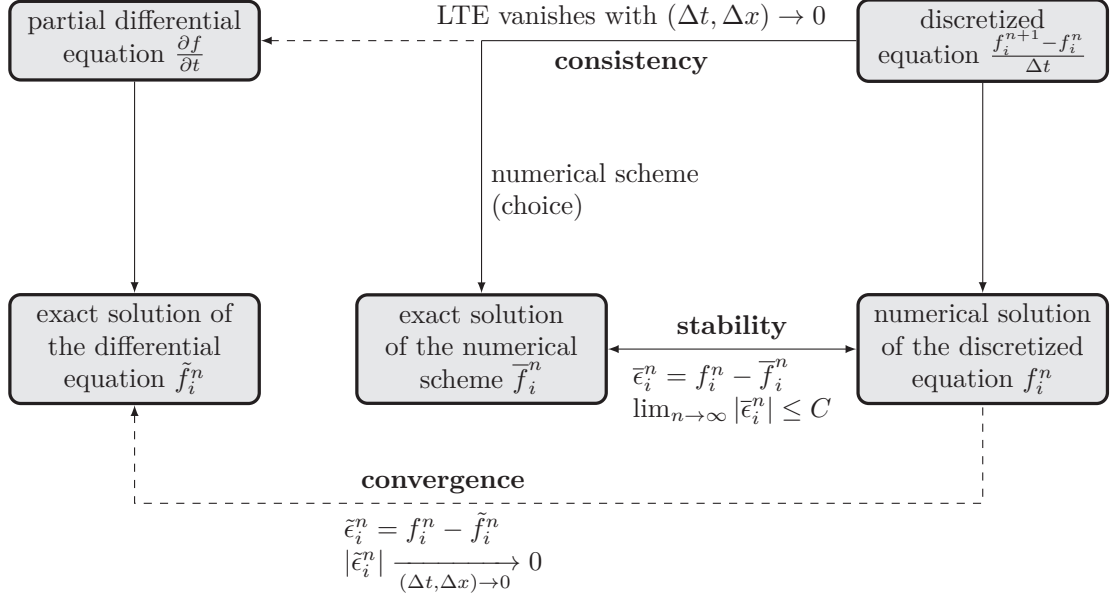


(b) Ultrabee [46, 151].



(c) Nbee [152].

Figure 3.24 – Common flux limiter profiles.

Figure 3.25 – General methodology from $PDE(s)$ to numerical solution.

The conservative form is:

$$f_i^{n+1} = f_i^n - \varsigma_i \left(F_{i+1/2}^{n,L} - F_{i-1/2}^{n,R} \right),$$

with $\varsigma_i = \frac{\Delta t}{\Delta x} u(x_i)$ the ratio of the physical velocity to the numerical velocity, and F the non-linear quantity.

Furthermore, the consecutive ratio in the direction of the flow is written $r_i^+ = \frac{f_i - f_{i-1}}{f_{i+1} - f_i}$. When the flow is reversed (velocity sign change), the ratio is written $r_i^- = \frac{f_i - f_{i+1}}{f_{i-1} - f_i} = \frac{1}{r_i^+}$.

The stability of the numerical scheme is based on the concept of *TVD* methods, introduced by [143]. The Total Variation is defined, in the continuous form as $TV = \int \left| \frac{\partial f}{\partial x} \right| dx$, and in the discrete form as $TV = \sum_i |f_{i+1} - f_i|$. When the property $TV(f^{n+1}) \leq TV(f^n)$ is verified, the scheme is said to be *TVD*. The global methodology used for the convergence of numerical schemes is exposed in fig. 3.25.

Based on the local cell velocity indicator ς_i , we resume the facet fluxes selection in algorithm 5 (\square^R denotes the immediate right side of the considered facet, and \square^L its left side):

When using the scheme [46], then the flux selection resumes to algorithm 6. Consistency bounds are dynamically computed as a function of the local cell velocity in order to allow extension of the stability margin.

The schemes of interest are:

Algorithm 5: Facet fluxes selection for cell i , in the flux limiter framework (NB, UB) .

```

1 pure function  $\mathfrak{L}(r, \varsigma)$ 
2   switch chosen limiter do
3     case None do return  $r$ 
4     case UB do
5       | return  $\max(0, \min(2r / |\varsigma|, 2 / (1 - |\varsigma|)))$ 
6     case NB do
7       | return  $\max(0, \min(2r / |\varsigma|, 1), \min(r, 2 / (1 - |\varsigma|)))$ 
8     end
9 end
10 if  $\varsigma_{i-1} > 0$  then
11   |  $F_{i-1/2}^{n,R} := f_{i-1} + \frac{1-\varsigma_{i-1}}{2} \mathfrak{L}(r_{i-1}^+, \varsigma_{i-1})(f_i - f_{i-1})$ 
12 else
13   |  $F_{i-1/2}^{n,R} := f_i + \frac{1+\varsigma_i}{2} \mathfrak{L}(r_i^-, -\varsigma_i)(f_{i-1} - f_i)$ 
14 end
15 if  $\varsigma_{i+1} < 0$  then
16   |  $F_{i+1/2}^{n,L} := f_{i+1} + \frac{1+\varsigma_{i+1}}{2} \mathfrak{L}(r_{i+1}^-, -\varsigma_{i+1})(f_i - f_{i+1})$ 
17 else
18   |  $F_{i+1/2}^{n,L} := f_i + \frac{1-\varsigma_i}{2} \mathfrak{L}(r_i^+, \varsigma_i)(f_{i+1} - f_i)$ 
19 end

```

Algorithm 6: Facet fluxes selection for cell i , for the DL^{xi} scheme [46].

```

1 pure function  $\mathfrak{L}(\varsigma, f_{\text{upwind}}, f_{\text{centre}}, f_{\text{downwind}})$ 
2   |  $f_{\min} := \min(f_{\text{centre}}, f_{\text{upwind}})$ 
3   |  $f_{\max} := \max(f_{\text{centre}}, f_{\text{upwind}})$ 
4   | lower :=  $\max(\min(f_{\text{downwind}}, f_{\text{centre}}), \frac{f_{\text{centre}} - f_{\max}}{\varsigma} + f_{\max})$ 
5   | upper :=  $\min(\max(f_{\text{downwind}}, f_{\text{centre}}), \frac{f_{\text{centre}} - f_{\min}}{\varsigma} + f_{\min})$ 
6   | return  $\max(\text{lower}, \min(f_{\text{downwind}}, \text{upper}))$ 
7 end
8 if  $\varsigma_i > 0$  then
9   |  $F_{i-1/2}^{n,R} := \mathfrak{L}(\varsigma_i, f_{i-2}, f_{i-1}, f_i)$ 
10  |  $F_{i+1/2}^{n,L} := \mathfrak{L}(\varsigma_i, f_{i-1}, f_i, f_{i+1})$ 
11 else
12  |  $F_{i-1/2}^{n,R} := \mathfrak{L}(-\varsigma_i, f_{i+1}, f_i, f_{i-1})$ 
13  |  $F_{i+1/2}^{n,L} := \mathfrak{L}(-\varsigma_i, f_{i-2}, f_{i+1}, f_i)$ 
14 end

```

- The SB^{xii} scheme as formulated in [150];
- The UB scheme of [151];
- The DL scheme developed in the formalism of [46];
- The NB scheme proposed by [152].

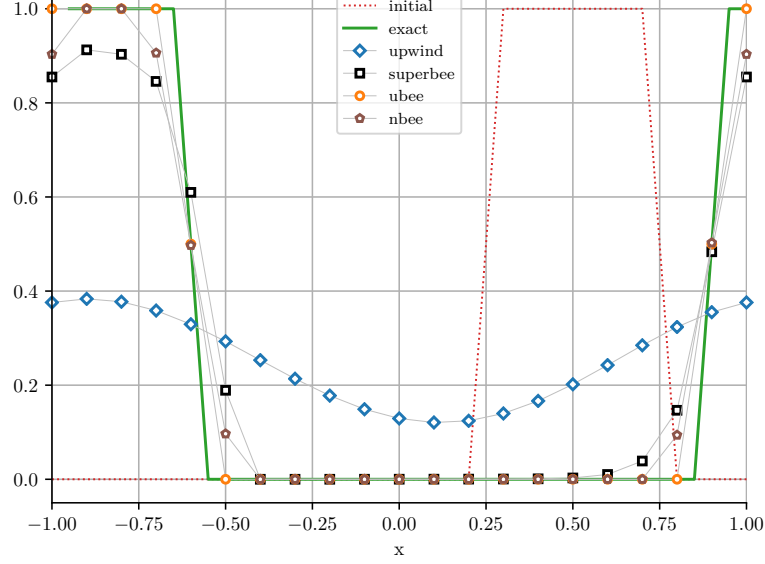


Figure 3.26 – 1D plot of a bi-dimensional split periodic transport of a sharp discontinuity.

The only scheme able to maintain the stiffness of the step function over discrete time is the UB scheme on fig. 3.26. However, since the exact location of the interface position is lost in the averaged unit volume using the FVM , artificial compression of the interface is not a wished-for property.

For smooth functions, the NB scheme seems to be the most appropriate since the global error is minimal as seen on fig. 3.27. Also, this scheme is able to maintain global maxima over time.

The slice in fig. 3.28 displays the transport of a bi-dimensional piecewise linear profile. We interest ourselves to the behaviour of the singular (tip of the triangle) over time. This example can be thought of being representative of a discrete tip of a droplet, when the mesh resolution is of order of a single cell in coarse DNS computations. We target mass conservation: the numerical scheme must thus be able to conserve global properties such as extrema. Over multiple time steps, the UB scheme artificially stiffens the interface, hence the local compression observed. The NB scheme performs best: the maximum is slightly reduced, whilst the error in the piecewise region remains small. The SB

xii. Super-BEE

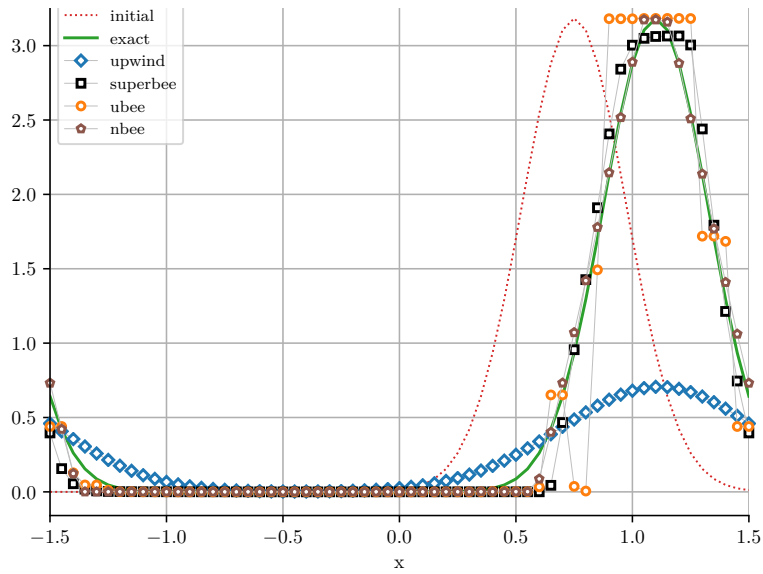


Figure 3.27 – 1D plot of a bi-dimensional split periodic transport of a smooth initial condition (gaussian).

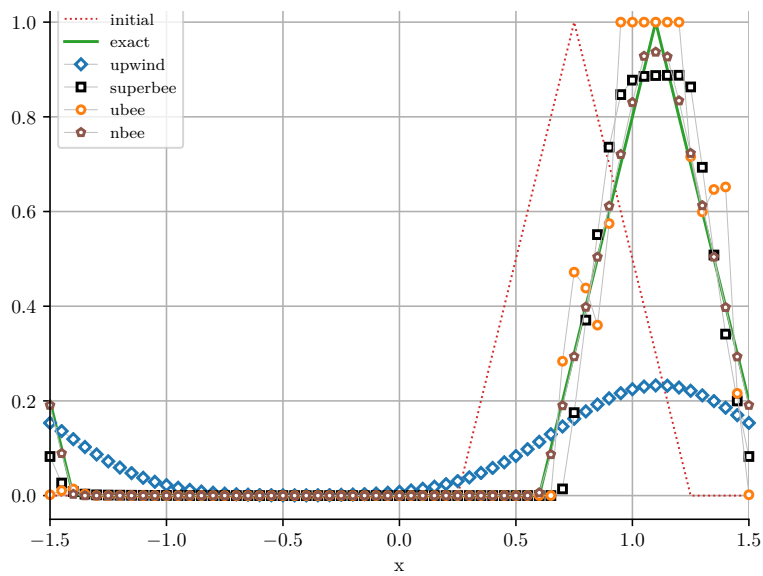


Figure 3.28 – 1D plot of a bi-dimensional split periodic transport of a piecewise linear profile.

scheme has a global compressive behaviour, and consequent loss of information through numerical dissipation at the tip of the data is observed.

Anti-diffusive schemes are high-order scheme which have interesting properties for scalar transport. One looks for second order scheme in general but also *TVD* schemes. Scalar transport in the context of multiphase flows is required as explained before to conserve maxima in order to respect momentum balance, and compute the correct volumetric forces exerted on the interface. Oscillatory schemes must thus be avoided.

For scalar transport (the density in our context), where φ defines the facet flux, we can write:

$$\begin{aligned}
& \frac{\partial}{\partial t} \int_V \rho(\mathbf{x}, t) dv + \oint_S \mathbf{flux} \cdot \mathbf{n} ds = 0 \\
& \xRightarrow{\text{Gauss}} \int_V \left[\frac{\partial \rho}{\partial t} + \nabla \cdot \mathbf{flux} \right] dv = 0 \\
& \xRightarrow{\text{arbitrary volume}} \frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = 0.
\end{aligned} \tag{3.62}$$

The eq. (3.62) is further developed into:

$$\begin{aligned}
& \frac{\rho^{n+1} - \rho^n}{\Delta t} = \nabla \cdot (\rho \mathbf{u}) \\
& \implies \rho^{n+1} = \rho^n + \Delta t \cdot \nabla \cdot (\rho \mathbf{u}) \\
& \implies \rho^{n+1} = \rho^n + \frac{\Delta t}{\Delta x} (\varphi_{i+1/2}^L - \varphi_{i-1/2}^R) \\
& \quad + \frac{\Delta t}{\Delta y} (\varphi_{j+1/2}^L - \varphi_{j-1/2}^R) + \frac{\Delta t}{\Delta z} (\varphi_{k+1/2}^L - \varphi_{k-1/2}^R).
\end{aligned} \tag{3.63}$$

The equation eq. (3.63) will be used in the flux limiter formalism (for the fluxes computations) for high order convection of the density in multiphase flows.

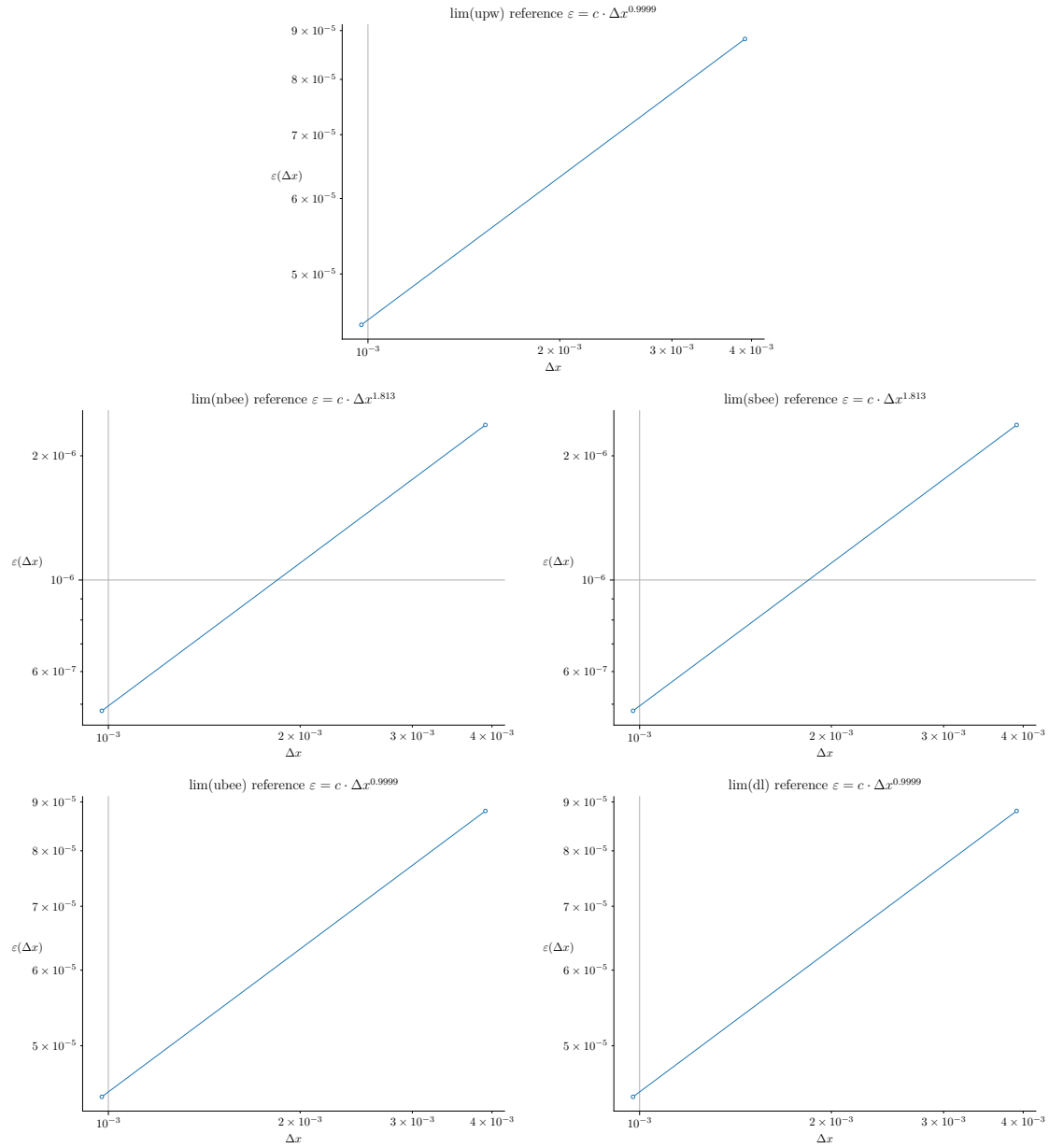
We assume that the error between the discrete solution \tilde{u}_h and the continuous solution u depends smoothly on a small parameter h , which can be the unit mesh size Δx for spatial convergence, or the timestep Δt for temporal convergence study.

We can thus write:

$$|\varepsilon| = |\tilde{u}_h - u| = kh^p + \mathcal{O}(h^{p+1}), \tag{3.64}$$

where k is a constant to be determined and p is the convergence rate of the scheme used to compute \tilde{u}_h .

The extrapolation using series in eq. (3.64) is only valid if the first order term is dominant [153].

Figure 3.29 – Convergence rates for the upwind, NB , SB , UB , and DL schemes.

If a reference solution u is known, then $\log|\varepsilon| = \log(k) + p \log(h) + \mathcal{O}(h)$, provides the slope p on a logarithmic scale plot of ε versus h .

On the contrary, if the solution is unknown, consecutive difference between approached solution is used in order to obtain the value of p .

For a fixed multiplier of the mesh size α , we define consecutive difference between solutions as: $f_h - f_{\alpha h}$, allowing to write:

$$\frac{f_h - f_{\alpha h}}{f_{\alpha h} - f_{\alpha^2 h}} = \frac{kh^p - k(\alpha h)^p + \mathcal{O}(h^{p+1})}{k(\alpha h)^p - k(\alpha^2 h)^p + \mathcal{O}(h^{p+1})} = \alpha^{-p} + \mathcal{O}(h). \quad (3.65)$$

Consequently, the ratio eq. (3.65) is used to estimate the convergence rate p .

The fig. 3.29 shows convergence rates for multiple schemes, for the advection problem of a smooth initial condition (taken as a combination of sigmoids using hyperbolic tangents).

Error rate was computed against a reference solution. The timestep used is taken as $\Delta t = \Delta x^3$ so that one may avoid influence of the timestepping truncature error in the solution. We expect a unit convergence order for the upwind scheme, and we obtain $p = .999$. For the high order scheme, the only scheme able to capture the smoothness of the solution is the *NB* scheme, and the order of convergence $p = 1.813$ tends to indicate a second order behaviour.

In the computations, we will use either the *NB* scheme or the stiffer scheme of [154], which we have found to perform best compared to other limiters. The choice of the limiter is then left to the user of the code with the implementation presented in algorithms 5 and 6. For longer simulations or the simulations running under the *IBM*, we prefer to use the stiffest [154] or *UB* limiters. If numerical convergence is difficult due to strong velocity gradients impacting the diffusive term, we use the smoother *NB* scheme.

3.3.2 Higher order convection operator (ENO)

The finite difference scheme applied to the advection operator in section 3.3.5.2 is unstable when considering low viscous flows, or in the absence of viscosity (for instance with the inviscid Euler equations). In order to increase the stability margin of the computation, and try to alleviate the viscous time-stepping constraints, we might consider using a higher order method, as the methods discussed later in section 3.3.1. As an example, we consider the Essentially Non-Oscillatory method [155]. The goal of these methods is to avoid first order spurious oscillations in stiff regions where the solution is discontinuous. As in signal theory, with the Gibbs phenomenon, these oscillations violate the principle of maximum conservation of the solution and are often a source of poorly resolved equations. This will further be developed in section 3.3.1.1 when introducing *TVD* methods.

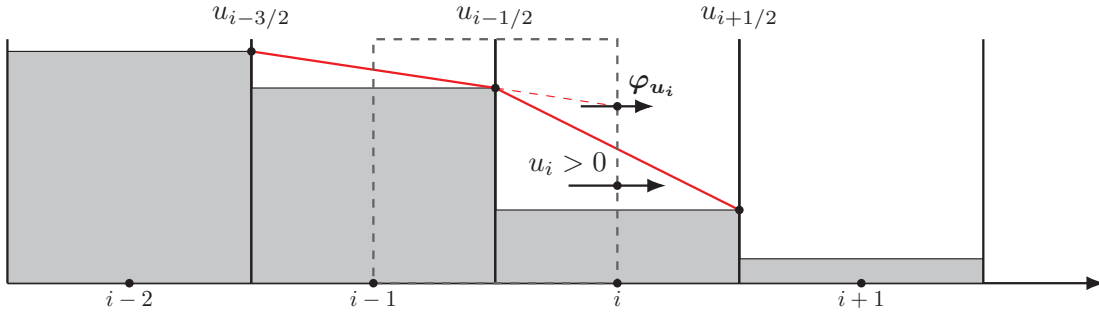


Figure 3.30 – Flux balance for the 'left' velocity, reconstruction of slopes.

The spirit of the method is to select the downwind flux by using the minimal rate of change of the variable over a control volume. On the fig. 3.30, with a positive downwind cell velocity u_i , the two candidates for the downwind flux φ_{u_i} at the control volume facet are:

$$\varphi_{u_i} = \begin{cases} u_{i-1/2} + \text{amin}(\Delta^+ u_{i-1/2}, \Delta^- u_{i-1/2}) / 2 & \text{if } u_i > 0, \\ 0 & \text{if } u_i = 0, \\ u_{i+1/2} - \text{amin}(\Delta^+ u_{i+1/2}, \Delta^- u_{i+1/2}) / 2 & \text{if } u_i < 0, \end{cases}$$

where amin represents the magnitude based switch:

$$\text{amin}(a, b) = \begin{cases} a & \text{if } |a| < |b|, \\ b & \text{otherwise,} \end{cases}$$

and Δ the difference operator:

$$\begin{cases} \Delta^+ u_{i-1/2} = u_{i+1/2} - u_{i-1/2}, \\ \Delta^- u_{i-1/2} = u_{i-1/2} - u_{i-3/2}. \end{cases}$$

3.3.3 Semi-implication of the viscous operator

After having obtained the expected analytical result, the timestep restriction induced by the explicit step for the diffusion operator leads us to consider implicitation of the diffusive term using a *CN* scheme [156], which is formally second-order accurate and unconditionally stable:

$$u^{n+3/8} = u^{n+1/4} + \frac{\Delta t_1}{\rho} \nabla \cdot \mu (\nabla u^{n+1/4} + \nabla^t u^{n+1/4}),$$

$$u^{n+1/2} - \frac{\Delta t_2}{\rho} \nabla \cdot \mu (\nabla u^{n+1/2} + \nabla^t u^{n+1/2}) = u^{n+3/8}.$$

The first equation is a forward explicit step using Δt_1 , while the second equation represents the implicit resolution for the next half substep Δt_2 , with the timestep split as $\Delta t = \Delta t_1 + \Delta t_2$.

Algorithm 7: Implicitation of the viscous operator.

- 1 compute the transport term $(\tilde{\mathbf{u}} \cdot \nabla) \tilde{\mathbf{u}}|^{n+1/4}$ and the capillary terms, using an explicit formulation
 - 2 solve the linear system $\mathbf{A} \tilde{\mathbf{u}}^{n+1/2} = \mathbf{B} \tilde{\mathbf{u}}^n + f(\tilde{\mathbf{u}}^n, \tilde{\mathbf{u}}^{n+3/8}, \dots)$, where \mathbf{A} and \mathbf{B} account for the $FD(s)$ spatial part of the *CN* scheme, applied to the $\nabla \cdot \mu (\nabla u + \nabla^t u)$ term
-

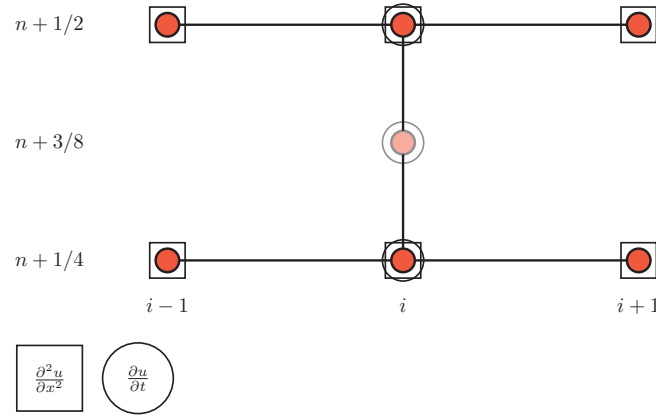


Figure 3.31 – Stencil for the *CN* scheme.

The implicitation pseudo-algorithm is found in algorithm 7, using the *CN* method whose spatial and temporal stencil is depicted in fig. 3.31. The difficulty of the implementation resides in additional unknowns (three velocities: u, v, w), and the imposition/consistency of boundary conditions appearing in the linear system. Also, one must use spatially variable edge centred viscosities since the viscosity is variable close to the interface in the mixing transition layer for a multiphase system.

For the validation of the *CN* scheme, we consider the lid-driven cavity problem in two dimensions as described in [157]. The fluid is put in to motion by the imposed top lid velocity, and by viscous diffusion. One carefully imposes the velocity on outer tangential facets using eq. (3.45), so that the average velocity on the boundary results in the imposed lid velocity.

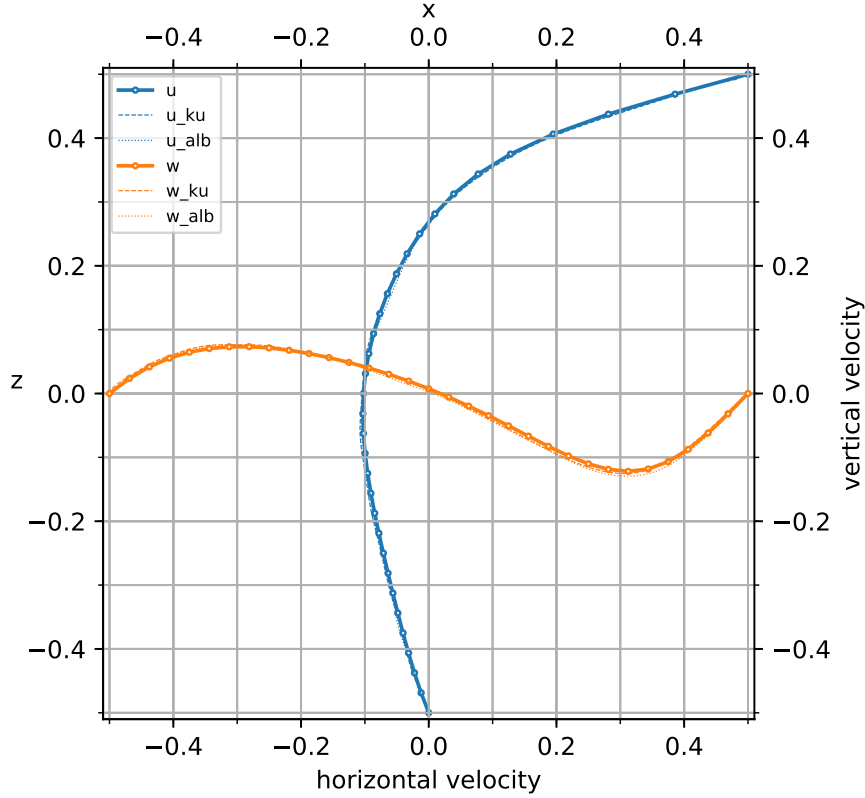


Figure 3.32 – Steady state horizontal (blue) and vertical (orange) velocity profiles.

The fig. 3.32 shows good agreement with the reference values from [158, 159]. This steady state solution validates the behaviour of the multiphase pressure solver, when the equation eq. (3.20) degenerates to a Poisson equation when the densities of the two phases are set equal: $\rho_1 = \rho_2$.

We also tested the convergence of the pressure solver on the lid driven cavity test case, enforcing specific corner boundary velocities in order to avoid the known edge pressure singularities, where the system is ill-conditioned.

3.3.4 Solving the variable coefficient Poisson equation

Multiple techniques can be used to solve the elliptic equation eq. (3.20), among which the direct approach [160], the spectral methods [159] and the multi-level approach [161, 162].

method	computational complexity
LU, Cholesky	$\mathcal{O}(n^3)$ (<i>factor</i>) + $\mathcal{O}(n^2)$ (<i>solve</i>)
Jacobi, GS^{xiii}	$\mathcal{O}(n^3)$
SOR^{xiv} with optimal ω	$\mathcal{O}(n^2)$
Spectral	$\mathcal{O}(n \cdot \log(n))$
Multi-grid	$\mathcal{O}(n)$

Table 3.3 – Complexity of linear algebra methods.

We distinguish direct methods from iterative methods. Whilst direct methods are suitable for small problems, computational and storage costs associated for matrices inversion are usually prohibitive for real sized problems. Solving a system of linear equations with a direct method is of complexity between $\mathcal{O}(n^2)$ and $\mathcal{O}(n^3)$ as exposed in table 3.3.

The problem studied results in a sparse linear operator, with at most $5n$ entries for a five points stencil. Since the inverse of a sparse matrix is generally dense, and since the operator is variable at each time-step, a preconditioning approach such as ILU(0) might not be interesting (solving a linear system depends on the condition number of the associated matrix, but also of the eigenvalues distribution) in terms of storage requirements but also in terms of computing complexity for inverting the linear operator at each time-step. An alternative is to use multi-grid or diagonal preconditioner, which are relatively simple to implement in a matrix free implementation, as explained in section 3.3.4.1.

The initial problem to solve is:

$$\mathbf{A} \mathbf{x} = \mathbf{b} . \quad (3.66)$$

We suppose that we can approximate in some way the inverse of \mathbf{A} by a matrix \mathbf{M}^{-1} , which shall be a good sparse approximation of \mathbf{A}^{-1} , and is relatively easy to compute. Then, the alternate problem preconditioned by \mathbf{M} :

$$\mathbf{M}^{-1} \mathbf{A} = \mathbf{M}^{-1} \mathbf{b} , \quad (3.67)$$

is better conditioned since $\mathbf{M}^{-1} \mathbf{A} \approx \mathbf{I}$, and the solution of eq. (3.67) is also a solution of eq. (3.66).

3.3.4.1 Krylov subspace methods

Multiple parallel Krylov methods were implemented for the incompressible problem eq. (3.20):

- *CG* [163];
- *GMRES* [164];
- *BiCGSTAB*^{xv} [117];
- *IDR*(*s*)^{xvi} [165].

Algorithm 8: Preconditioned parallel Bi-CGSTAB [117, 166].

```

1 sync_halo( $\mathbf{x}_0$ )                                ! MPI - planes
2  $\mathbf{x} := \mathbf{x}_0$                                 ! initial guess
3  $\mathbf{r}_0 := \mathbf{b} - \mathbf{A} \mathbf{x}$                         ! initial residual
4  $\mathbf{p} := \mathbf{r} := \mathbf{r}_0$ 
5 while  $\text{norm}^2(\mathbf{r}) > \text{tolerance}$  do
6    $\rho := \mathbf{r} \cdot \mathbf{r}_0$                                 ! co_sum()
7    $\dot{\mathbf{p}} := (\text{prec})? \mathbf{M}^{-1} \mathbf{p} : \mathbf{p}$           ! optional preconditioner step
8   sync_halo( $\dot{\mathbf{p}}$ )                                ! MPI - planes
9    $\mathbf{q} := \mathbf{A} \dot{\mathbf{p}}$ 
10   $\alpha := \rho / (\mathbf{q} \cdot \mathbf{r}_0)$                                 ! co_sum()
11   $\mathbf{s} := \mathbf{r} - \alpha \mathbf{q}$                                 ! stabilizer
12   $\dot{\mathbf{s}} := (\text{prec})? \mathbf{M}^{-1} \mathbf{s} : \mathbf{s}$           ! optional preconditioner step
13  sync_halo( $\dot{\mathbf{s}}$ )                                ! MPI - planes
14   $\mathbf{t} := \mathbf{A} \dot{\mathbf{s}}$ 
15   $\omega := (\mathbf{t} \cdot \mathbf{s}) / (\mathbf{t} \cdot \mathbf{t})$                                 ! co_sum()
16   $\mathbf{x} += \alpha \dot{\mathbf{p}} + \omega \dot{\mathbf{s}}$ 
17   $\mathbf{r} := \mathbf{s} - \omega \mathbf{t}$ 
18   $\beta := ((\mathbf{r} \cdot \mathbf{r}_0) / \rho) * (\alpha / \omega)$                                 ! co_sum()
19   $\mathbf{p} := \mathbf{r} + \beta(\mathbf{p} - \omega \mathbf{q})$                     ! search direction vector
20 end

```

We describe the parallel *BiCGSTAB* pseudo-code in algorithm 8, which has been found to outperform all other methods for this particular problem. The lines with `sync_halo` indicate inter process communications which usually are a bottleneck for strong scaling of the problem (prefixed with **co** to denote **collective** operations). \mathbf{p} is the search direction, \mathbf{r} is the residual, and \mathbf{s} is the residual after application of a Bi-CG step. ω is chosen to minimize $\mathbf{r} = (\mathbf{I} - \omega \mathbf{A}) \mathbf{s}$. Consecutive residuals are orthogonal, and consecutive search directions are conjugate. \mathbf{q} and \mathbf{t} are temporary storage for matrix vector products. Application of a preconditioner is denoted by \square .

We note that the matrix-free implementation strategy only allows to use preconditioners which do not explicitly require to compute the inverse of a linear operator (\mathbf{M}^{-1}). In practice, when solving for $\dot{\mathbf{p}}$ in algorithm 8, we iterate the linear system $\mathbf{A} \dot{\mathbf{p}} = \mathbf{p}$. Since

xv. BiConjugate Gradient STABilized

xvi. Induced Dimension Reduction

M^{-1} approaches in some sense A^{-1} , we have the relationship:

$$A \dot{p} := \overbrace{A M^{-1}}^{\simeq I} p \simeq p .$$

For the algorithm 8, we need to define a proper matrix vector product. Dense matrix storage is prohibited for computational costs and storage issues. Sparse matrix storage (CSR^{xvii} , CSC^{xviii} , ...) is memory efficient, but algorithms have to be rewritten to account for data distribution and distributed operations which require more complexity. Alternatively, for a compact stencil, we can store the linear operator coefficient in the $n + 1^{th}$ dimension of a regular n dimensional array, as shown in listing 1, to define a matrix-free algorithm. The macros `IJK` represents the multi-dimensional bounds, `SIJK(...)` allows to shift the input vector block along a specific direction, and `map(...)` is a mapping of the displacement to an array index in order to fetch the coefficients for the five points stencil. With this approach, some extra computations are done with zero valued coefficients, but those extra operations are overall a minor extra cost over clarity, simplicity of the implementation and communications.

```

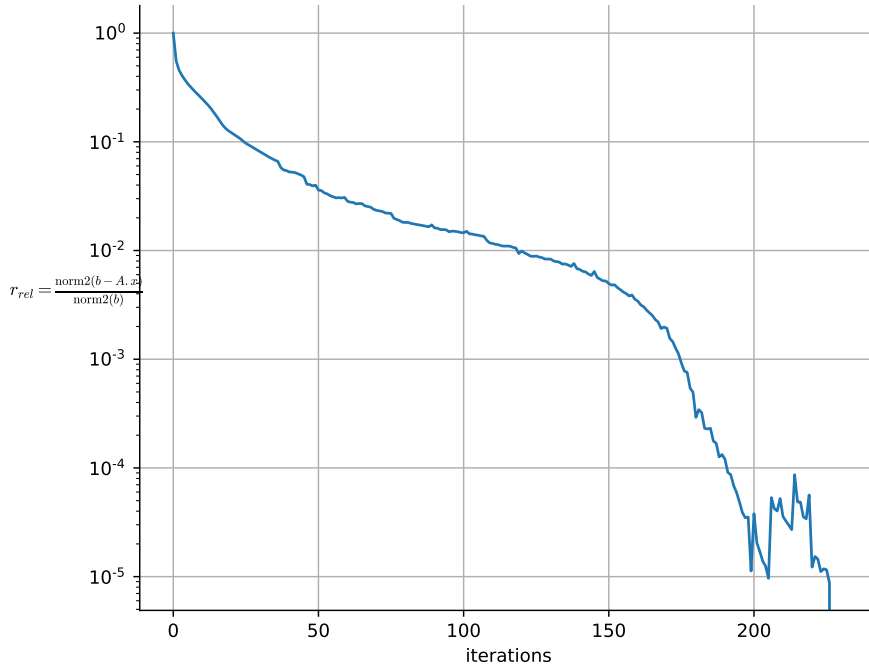
! pre-condition: synchronization of local the input vector b
! IJK := (lo(1):hi(1),lo(2):hi(2),lo(3):hi(3))
! > block selection in the x, y and z directions
! SIJK(-1,0,0) := (lo(1)-1:hi(1)-1,lo(2):hi(2),lo(3):hi(3))
! > shift by -1 unit in the x direction
! map(0, 0, 0)
! > diagonal coefficient of the linear operator
c(IJK) = ( &
  A(IJK, map(0, 0, 0)) * b(IJK) + &
  A(IJK, map(-1, 0, 0)) * b(SIJK(-1, 0, 0)) + &
  A(IJK, map(+1, 0, 0)) * b(SIJK(+1, 0, 0)) + &
  A(IJK, map(0, -1, 0)) * b(SIJK(0, -1, 0)) + &
  A(IJK, map(0, +1, 0)) * b(SIJK(0, +1, 0)) + &
  A(IJK, map(0, 0, -1)) * b(SIJK(0, 0, -1)) + &
  A(IJK, map(0, 0, +1)) * b(SIJK(0, 0, +1)) &
)
```

Listing 1 – Compact matrix-free vector product for the five-points stencil, in Fortran.

The fig. 3.33 shows the residual reduction in the ℓ^2 norm. We observe three phases: first a super-linear rate of convergence below the first ten iterations, then a sub-linear phase for the next 150 iterations, followed by a second super-linear convergence rate for the next 50 iterations. The convergence criterion was set to $1 \cdot 10^{-5}$ on the norm of the

xvii. Compressed Sparse Row

xviii. Compressed Sparse Column

Figure 3.33 – *BiCGSTAB* residuals for the two-phase problem.

relative residual:

$$\text{res}_{\text{rel}} = \frac{\text{norm}^2(b - \mathbf{A}x)}{\text{norm}^2(b)}.$$

The slow convergence of the iterative method, and the amount of required communication increasing with the domain decomposition indicate that a better pressure solving method is needed in the multiphase formulation. This is why we interest ourselves in accelerators such as the multi-grid method in section 3.3.4.2.

norm	ℓ^∞	ℓ^1	ℓ^2
pre-solve	273.1	$.9045 \cdot 10^5$	3538.0
post-solve	$.5221 \cdot 10^{-7}$	$.2180 \cdot 10^{-2}$	$.9392 \cdot 10^{-5}$
reduction factor	$5231 \cdot 10^{10}$	$.4148 \cdot 10^8$	$.3768 \cdot 10^9$

Table 3.4 – Pressure residual norms for a chosen criterion $1 \cdot 10^{-5}$ in the ℓ^2 norm.

The iterative method allows to strictly reduce the divergence of the fluid in any chosen norm table 3.4. We usually choose the ℓ^2 norm, but the ℓ^∞ can also be chosen as a

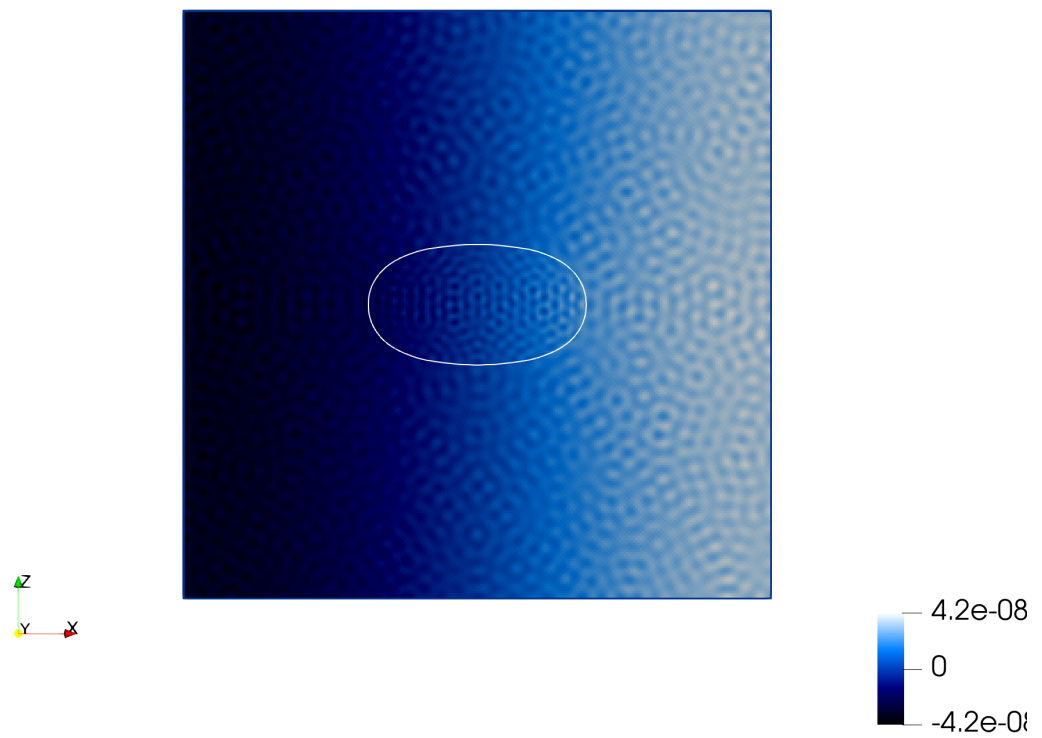


Figure 3.34 – Post *BiCGSTAB* velocity divergence pattern.

relaxed criterion since the local expansion/compression of the fluid is of interest and not necessarily the whole domain reduction. Hence, as soon as the maximum value of the residual is small enough, the incompressibility of the fluid is locally enforced up to the given precision. In our opinion, the ℓ^∞ is thus the best compromise between accuracy and costs in terms of $FLOPS^{\text{xix}}$.

3.3.4.2 Multi-grid methods

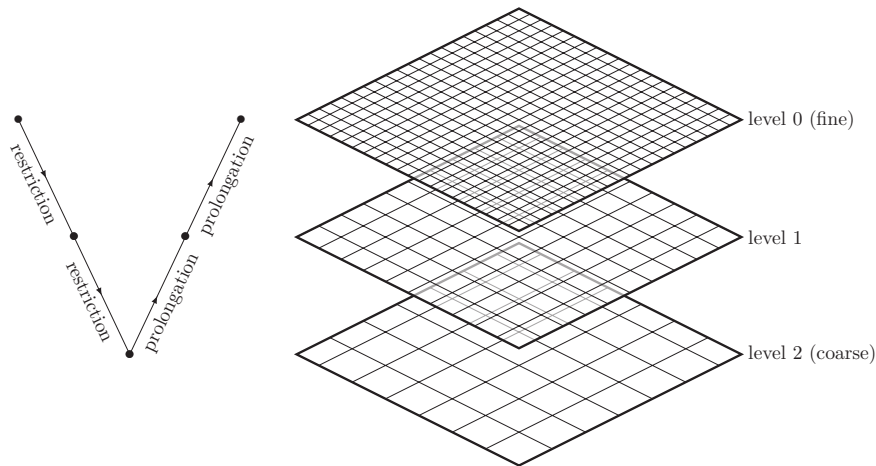


Figure 3.35 – Multi-grid strategy for linear systems.

Algorithm 9: Two Grid Method.

1	$x := S_1^{\nu_{\text{pre}}}(x, b)$!pre-smoothing with ν_{pre} iterations
2	$d := R_1(b - A_1 x)$!defect computation and restriction to a coarser grid
3	$e := A_2^{-1} d$!solve the coarse problem (error as a function of the defect)
4	$x += P_2 e$!prolongation and correction step
5	$x := S_1^{\nu_{\text{post}}}(x, b)$!post-smoothing with ν_{post} iterations

Multiple multi-level strategies are available, usually allowing to spend more time on the coarsest levels, where computations are the cheapest.

For the inter-level restriction, we use simple weighted cell averaging, and straight injection on boundary nodes when needed. Following the conclusions of [167], we use a higher order interpolation operator (surface interpolation in two-dimensions and cubic interpolation in three-dimensions).

The restriction of the linear operator to the coarser level is prone to discussion. The Galerkin method restricts the linear operator onto the subspace of the coarser grid

xix. Floating Point Operations per Second

$\mathbf{A}^c := \mathbf{R} \mathbf{A}^f \mathbf{P}$. The method is quite costly since it requires time consuming parallel matrix-matrix products, and additional storage since each interpolation and restriction operator spreads out the finer stencil used [161, 168]. An alternative approach is the assembly of the linear operator on each sub-grid. This approach is problem dependent, since it now depends on how the $PDE(s)$ on the fine grid is solved on the coarse grid. For cell centred multi-grid, we use the same boundary conditions, and the same averaging process for the densities onto the facets. The density field is projected onto the coarser grid using the same weighted operator presented before.

Relaxation on the grid also needs to be discussed. For the multiphase problem, we found that using either Jacobi or parallel $RBGS^{xx}$ did not improve the reduction of the residual. Even worse, the residual followed an oscillating pattern (periodic increase and reduction), often leading to non-convergence of the pseudo-time pressure loop. We first considered potential issues with the imposed $BC(s)$, and afterwards with the domain decomposition and parallelism, but the implementation proved to be correct since the two-phase solver with density ρ set equal for the two-phases was converging to the required tolerance. The only convergent method was to reuse the Krylov methods (direct coarsest solver) as a pre/post smoother. Some authors [169, 170] seemed to hit similar issues.

The MG method using non-Galerkin approach shows on fig. 3.36 the concentration of the residual velocity divergence along the interface, as opposed to the fig. 3.34, where interface position cannot be inferred from the residual pattern.

The listing 2 shows the convergence results for a one-dimensional multi-grid on 6 levels with fine-grid resolution of 1024 cells and 1025 nodes. The error is computed with respect to an analytical solution for the chosen RHS . The Poisson problem is solved for a sinusoidal RHS on an appropriate domain with 0 valued Dirichlet boundary condition. The convergence factor around $1/3$ is reasonable and increasing with the number of V-cycles as expected. Grid unknowns are kept odd for nodal multi-grid and even for the cell centred version across all levels, for consistency with the spatial spacing along boundary conditions. The non-square interpolation and restriction operators (not necessary symmetric: $\mathbf{P} \neq \mathbf{R}^t$ for a cell-centred multigrid) are key to the correct convergence of the method, especially the consistency of $BC(s)$ injection/interpolation on fine and coarser levels. The linear operator is restricted to the coarser grid (Galerkin method) resulting in denser matrix as the depth increases, and is not practically usable as is for three-dimensional real case problems, or as in our case where the operator coefficients are stored in a fixed size array as exposed in listing 1. We use $\nu_{pre} = 5$ and $\nu_{post} = 5$ pre/post-smoothing GS iterations on each grid level, and the recursive version of algorithm 9. This version will be kept for reference mostly for inter-level transfer operators (injection, half or full-weighting) since we use a cell-centred approach, and non-Galerkin linear operator, and will serve as a basis for the development of a variable coefficient Poisson multi-grid accelerator.

xx. Red-Black Gauss-Seidel

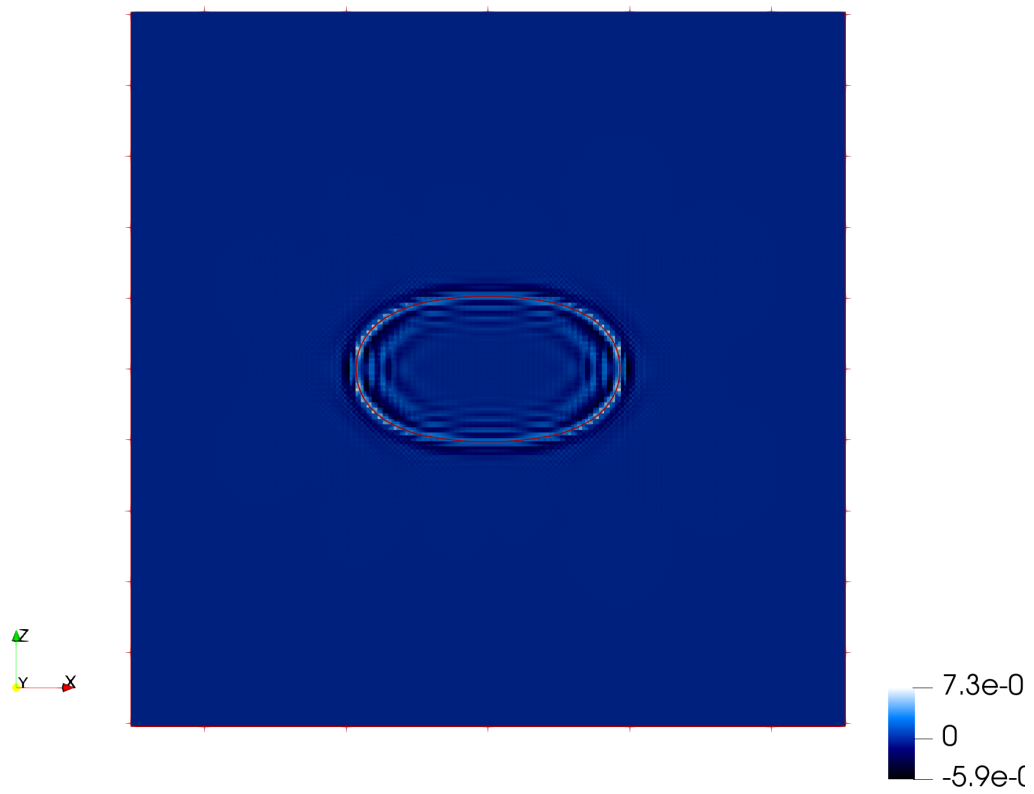
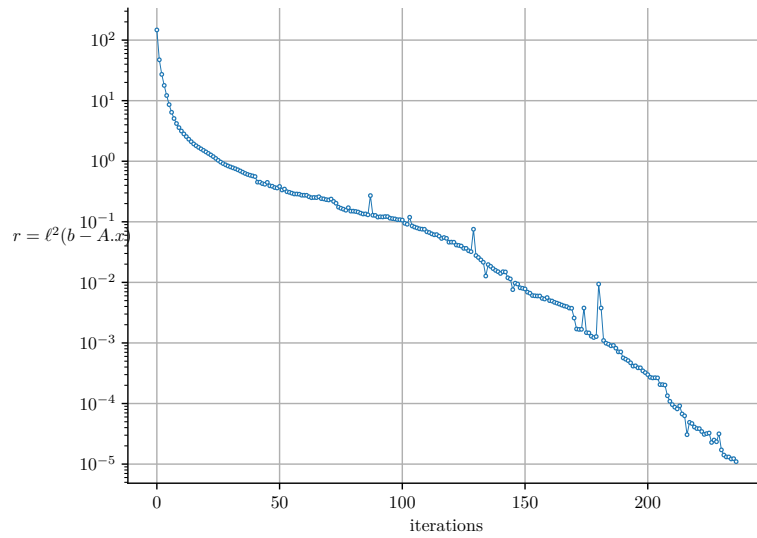


Figure 3.36 – Post $MG^{xxi}(2)$ velocity divergence pattern.

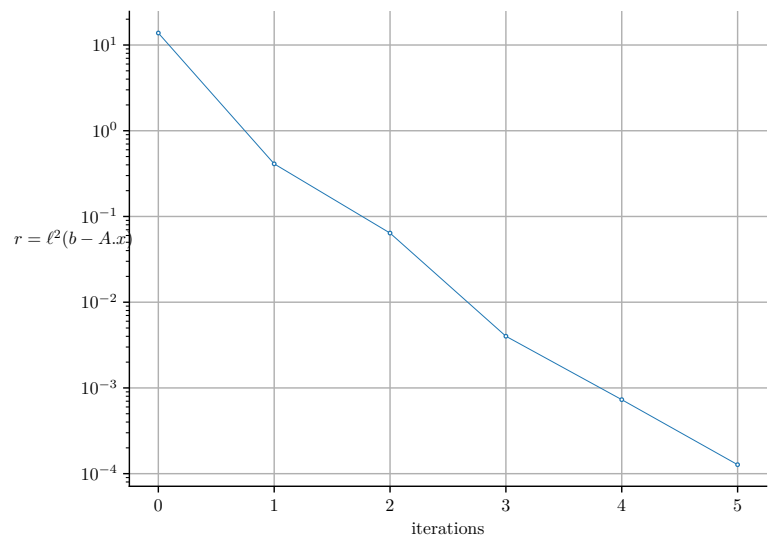
step	rel_err	rel_res	cv	step	rel_err	rel_res	cv
0	6.19e+01	3.90e+04	1.00	0	4.85e+00	3.90e+04	1.00
1	5.35e+00	2.85e+03	0.07	1	1.61e+00	3.06e+03	0.08
2	1.74e+00	5.75e+02	0.20	2	5.64e-01	6.99e+02	0.23
3	3.30e-01	1.33e+02	0.23	3	2.01e-01	1.83e+02	0.26
4	9.69e-02	3.30e+01	0.25	4	7.23e-02	5.14e+01	0.28
5	3.18e-02	8.55e+00	0.26	5	2.61e-02	1.50e+01	0.29
6	1.03e-02	2.31e+00	0.27	6	9.45e-03	4.59e+00	0.30
7	3.47e-03	6.51e-01	0.28	7	3.42e-03	1.43e+00	0.31
8	1.15e-03	1.90e-01	0.29	8	1.24e-03	4.58e-01	0.32
9	3.84e-04	5.73e-02	0.30	9	4.52e-04	1.48e-01	0.32
10	1.28e-04	1.76e-02	0.31	10	1.64e-04	4.86e-02	0.33
11	4.30e-05	5.52e-03	0.31	11	5.98e-05	1.60e-02	0.33
12	1.45e-05	1.75e-03	0.32	12	2.17e-05	5.32e-03	0.33
13	4.92e-06	5.60e-04	0.32	13	7.92e-06	1.77e-03	0.33
14	1.67e-06	1.80e-04	0.32	14	2.88e-06	5.92e-04	0.33
15	5.72e-07	5.83e-05	0.32	15	1.04e-06	1.98e-04	0.34
16	1.96e-07	1.89e-05	0.33	16	3.80e-07	6.69e-05	0.34
17	6.76e-08	6.20e-06	0.33	17	1.38e-07	2.25e-05	0.34
18	2.34e-08	2.03e-06	0.33	18	5.02e-08	7.64e-06	0.34
19	8.12e-09	6.72e-07	0.33	19	1.82e-08	2.59e-06	0.34
20	2.82e-09	2.22e-07	0.33	20	6.62e-09	8.84e-07	0.34
21	9.87e-10	7.42e-08	0.33	21	2.40e-09	3.02e-07	0.34
22	3.45e-10	2.48e-08	0.33	22	8.71e-10	1.03e-07	0.34
23	1.21e-10	8.34e-09	0.34	23	3.16e-10	3.55e-08	0.34
24	4.25e-11	2.81e-09	0.34	24	1.14e-10	1.22e-08	0.34
25	1.50e-11	9.54e-10	0.34	25	4.15e-11	4.22e-09	0.35
26	5.20e-12	3.24e-10	0.34	26	1.50e-11	1.46e-09	0.35
27	1.92e-12	1.10e-10	0.34	27	5.49e-12	5.06e-10	0.35
				28	2.04e-12	1.75e-10	0.35

Listing 2 – Error, residuals and convergence rate for cell-centred (left) and nodal (right) one-dimensional V-cycle multi-grid.

In our late experiments, we have found that using the algorithm 8 with a multi-grid preconditioner is more efficient than solving the problem using a multi-grid such as the algorithm 9 with Krylov methods or *RBGS* as smoother and bottom solver. On fig. 3.37 we solve a three-dimensional problem with 32^3 unknowns, with and without preconditioning the algorithm 8. In fig. 3.37a, the solver reaches the requested decrease ratio of the residual of $1 \cdot 10^{-5}$ in 237 iterations. However, when using 4 grid levels, the iterations for each level towards the coarsest are $\{71, 65, 65, 65\}$, for a total of 266 iterations. The solver is optimal in a sense since one iteration of the preconditioned problem reaches a decrease of one decade on the residual on fig. 3.37b. The total number of iterations is similar (minus the extra cost for the inter-grid interpolation (prolongation) and restriction), but the resolution is cheaper since the number of unknowns on each grid decrease with a factor 2^n , where n is the dimensionality of the code.



(a) Sub-optimal bi-cgstab (without preconditioner).



(b) Optimal bi-cgstab (with multi-grid preconditioner).

Figure 3.37 – Preconditioner using multi-grid acceleration.

3.3.4.3 Spectral methods

The method exploits the fact that the basis functions of the Fourier transform are eigenfunctions of the discrete Laplace operator. Hence, the algorithm performs a forward Fourier transform on the *RHS* of eq. (3.20), a scaling is performed on the Fourier expansion, and a backward Fourier expansion is applied to yield the pressure. However, the fast spectral solvers using the Fourier transform expansion require constant spatial coefficients [171].

As an alternative to this problem, a strategy is proposed to split the problem into a standard Poisson equation. The *LHS*^{xxii} can be split into an explicit and an implicit parts:

$$\frac{1}{\rho^{n+1}} \nabla p^{n+1} \rightarrow \frac{1}{\rho_{min}} \nabla p^{n+1} + \left(\frac{1}{\rho^{n+1}} - \frac{1}{\rho_{min}} \right) \nabla \hat{p},$$

where \hat{p} is an explicit pressure extrapolation of p^n .

This yields a Poisson type equation with constant coefficients (thus avoiding the need to assemble a linear system per timestep), which is to be solved by a fast spectral solver:

$$\nabla^2 p^{n+1} = \nabla \cdot \left[\left(1 - \frac{\rho_{min}}{\rho^{n+1}} \right) \nabla \hat{p} \right] + \frac{\rho_{min}}{\Delta t} \nabla \cdot \tilde{\mathbf{u}}.$$

We implemented this approach using the open-source library *PoisFFT*^{xxiii} [172], which uses a parallel implementation of the *FFT* [173, 174]. However, results were inconsistent and the method has to be reworked, starting with a simple one-dimensional *FFT* code from scratch that we would extend to n-dimensions. Moreover, the incompressibility of the fluid at the interface was not enforced, contrarily to the approaches exposed in section 3.3.4.1.

xxii. Left-Hand Side

xxiii. Fast Fourier Transform

3.3.5 Multi-dimensional splitting

The numerical methods developed all along the section 3.3 will be implemented on one-dimensional cases before being extended to multiple dimensions. We therefore expose the concept of splitting methods, and how the choice of a particular method over another can affect the numerical results. We will expose the splitting of $NSE(s)$ operators, and expose some validation test cases.

3.3.5.1 General splitting methods

We now reconsider eq. (3.54) in multiple dimensions:

$$\frac{\partial f}{\partial t} + \nabla \cdot \varphi(f) = 0, \quad (3.68)$$

where $\varphi(f)$ corresponds to the numerical flux of f across cell boundaries for the *FVM*.

Multiple strategies exist for updating the variable f . One can first compute all fluxes from the variable available at time n , and once the fluxes are computed, update the variable to time $n + 1$: this is the unsplit method.

Alternatively, composing the directional update is called a Lie splitting [175, 176]. As an example, we consider the following initial value problem with ordinary linear differential operators $A = a \frac{\partial}{\partial x}$ and $B = b \frac{\partial}{\partial y}$:

$$\begin{cases} f_t = Af + Bf, \\ f(0) = f_0. \end{cases} \quad (3.69)$$

The exact solution of the system eq. (3.69) is $f(t) = f_0 e^{(A+B)t}$.

Using a Lie splitting, we write:

$$f_L(\Delta t) = \begin{cases} e^{A\Delta t} e^{B\Delta t} f_0, & \text{or} \\ e^{B\Delta t} e^{A\Delta t} f_0. \end{cases} \quad (3.70)$$

The splitting error, after Taylor expansion and appropriate truncature is thus:

$$\begin{aligned} f_L(\Delta t) - f(\Delta t) &= \left\{ \left(1 + A\Delta t + A^2 \frac{\Delta t^2}{2} + \mathcal{O}(\Delta t^3) \right) \cdot \left(1 + B\Delta t + B^2 \frac{\Delta t^2}{2} + \mathcal{O}(\Delta t^3) \right) \right. \\ &\quad \left. - \left(1 + (A+B)\Delta t + (A+B)^2 \frac{\Delta t^2}{2} + \mathcal{O}(\Delta t^3) \right) \right\} f_0 \\ &= [A, B] \frac{\Delta t^2}{2} f_0 + \mathcal{O}(\Delta t^3), \end{aligned}$$

where $[A, B]$ is the commutator $AB - BA$.

Since the truncature error is of order $\mathcal{O}(\Delta t^2)$, the Lie splitting is thus, in general, only first order accurate.

An alternative is the Strang splitting [177]:

$$f_L(\Delta t) = \begin{cases} e^{A\Delta t/2} e^{B\Delta t} e^{A\Delta t/2} f_0, & \text{or} \\ e^{B\Delta t/2} e^{A\Delta t} e^{B\Delta t/2} f_0. \end{cases} \quad (3.71)$$

The truncature error when using this splitting reads:

$$\begin{aligned} f_S(\Delta t) - f(\Delta t) &= \left\{ \left(1 + A\frac{\Delta t}{2} + A^2\frac{\Delta t^2}{8} + A^3\frac{\Delta t^3}{48} + \mathcal{O}(\Delta t^4) \right) \right. \\ &\quad \cdot \left(1 + B\Delta t + B^2\frac{\Delta t^2}{2} + B^3\frac{\Delta t^3}{6} + \mathcal{O}(\Delta t^4) \right) \\ &\quad \cdot \left(1 + A\frac{\Delta t}{2} + A^2\frac{\Delta t^2}{8} + A^3\frac{\Delta t^3}{48} + \mathcal{O}(\Delta t^4) \right) \\ &\quad \left. - \left(1 + (A+B)\Delta t + (A+B)^2\frac{\Delta t^2}{2} + (A+B)^3\frac{\Delta t^3}{6} + \mathcal{O}(\Delta t^4) \right) \right\} f_0 \\ &= \left(\frac{B^2A}{12} - \frac{BAB}{6} + \frac{AB^2}{12} - \frac{A^2B}{24} + \frac{ABA}{12} - \frac{BA^2}{24} \right) f_0 \Delta t^3 + \mathcal{O}(\Delta t^4) \\ &= \left(\frac{1}{12}[B, [B, A]] - \frac{1}{24}[A, [A, B]] \right) f_0 \Delta t^3 + \mathcal{O}(\Delta t^4). \end{aligned}$$

Using the Strang splitting leads to a second order accurate scheme.

From the splitting methods eq. (3.71) and eq. (3.70), one can construct a family of weighted splitting for example the $SW S^1$ splitting [178, 179]:

$$f_{SW S}(\Delta t) = \left(e^{A\Delta t} e^{B\Delta t} + e^{B\Delta t} e^{A\Delta t} \right) f_0 / 2.$$

This method uses algebraic cancellation of the commutator term in the Lie splitting, and allows a global $\mathcal{O}(\Delta t^2)$ time accuracy.

According to [140], bi-dimensional split schemes are at most first order accurate in time.

3.3.5.2 Convective operator

The convective operator of the Navier-Stokes Equation(s) is written:

$$\mathbf{A} = \frac{1}{V} \oint_S \rho \begin{pmatrix} u^2 & uv & uw \\ vu & v^2 & vw \\ wu & vw & w^2 \end{pmatrix} \cdot \mathbf{n} \, ds.$$

i. Symmetrically Weighted Sequential

Applying second order central $FD(s)$ on the spatial x, y, z components of the convective operator \mathbf{A} :

$$\begin{aligned}
 (Ax)_{i-1/2,j,k}^n &= \frac{1}{V} \oint_S (\rho \mathbf{u} \otimes \mathbf{u}) \cdot \mathbf{n} \, ds \Big|_x \\
 &= \frac{1}{\Delta x \Delta y \Delta z} \left[((\rho u u)_{i,j,k} - (\rho u u)_{i-1,j,k}) \Delta y \Delta z \right. \\
 &\quad + ((\rho u v)_{i-1/2,j+1/2,k} - (\rho u v)_{i-1/2,j-1/2,k}) \Delta x \Delta z \\
 &\quad \left. + ((\rho u w)_{i-1/2,j,k+1/2} - (\rho u w)_{i-1/2,j,k-1/2}) \Delta x \Delta y \right]
 \end{aligned} \tag{3.72}$$

$$\begin{aligned}
 (Ay)_{i,j-1/2,k}^n &= \frac{1}{V} \oint_S (\rho \mathbf{u} \otimes \mathbf{u}) \cdot \mathbf{n} \, ds \Big|_y \\
 &= \frac{1}{\Delta x \Delta y \Delta z} \left[((\rho v u)_{i+1/2,j-1/2,k} - (\rho v u)_{i-1/2,j-1/2,k}) \Delta y \Delta z \right. \\
 &\quad + ((\rho v v)_{i,j,k} - (\rho v v)_{i,j-1,k}) \Delta x \Delta z \\
 &\quad \left. + ((\rho v w)_{i,j-1/2,k+1/2} - (\rho v w)_{i,j-1/2,k-1/2}) \Delta x \Delta y \right]
 \end{aligned}$$

$$\begin{aligned}
 (Az)_{i,j,k-1/2}^n &= \frac{1}{V} \oint_S (\rho \mathbf{u} \otimes \mathbf{u}) \cdot \mathbf{n} \, ds \Big|_z \\
 &= \frac{1}{\Delta x \Delta y \Delta z} \left[((\rho w u)_{i+1/2,j,k-1/2} - (\rho w u)_{i-1/2,j,k-1/2}) \Delta y \Delta z \right. \\
 &\quad + ((\rho w v)_{i,j+1/2,k-1/2} - (\rho w v)_{i,j-1/2,k-1/2}) \Delta x \Delta z \\
 &\quad \left. + ((\rho w w)_{i,j,k} - (\rho w w)_{i,j,k-1}) \Delta x \Delta y \right] .
 \end{aligned}$$

3.3.5.3 Diffusion operator

We first consider the bi-dimensional case:

$$\mathbf{D} = \frac{1}{V} \oint_S \mu \begin{pmatrix} 2u_x & v_x + u_y \\ u_y + v_x & 2v_y \end{pmatrix} \cdot \mathbf{n} \, ds . \tag{3.73}$$

The divergence-free flow condition:

$$u_x + v_y = 0 \tag{3.74}$$

is then used to simply eq. (3.73).

Taking the x or y derivative of eq. (3.74) yields:

$$\begin{aligned} u_{xx} + v_{yx} &= 0, \\ u_{xy} + v_{yy} &= 0. \end{aligned} \quad (3.75)$$

If we use spatial constant viscosity, as it can be the case with sharp interfaces [134], considering each fluid solely, we can write:

$$\nabla \cdot \mu(\nabla \mathbf{u} + \nabla \mathbf{u}^t) = \mu \begin{pmatrix} 2u_{xx} + v_{xy} + u_{yy} \\ u_{yx} + v_{xx} + 2v_{yy} \end{pmatrix}.$$

Then, using eq. (3.75):

$$\nabla \cdot \mu(\nabla \mathbf{u} + \nabla \mathbf{u}^t) = \mu \begin{pmatrix} u_{xx} + u_{yy} \\ v_{xx} + v_{yy} \end{pmatrix}. \quad (3.76)$$

The equation eq. (3.76) is appealing, because the diffusive term does not depend on coupled terms. However, this is not the case when the equation considered are three-dimensional.

We now consider the three-dimensional case:

$$\mathbf{D} = \frac{1}{V} \oint_S \mu \begin{pmatrix} 2u_x & v_x + u_y & w_x + u_z \\ u_y + v_x & 2v_y & w_y + v_z \\ u_z + w_x & v_z + w_y & 2w_z \end{pmatrix} \cdot \mathbf{n} \, ds.$$

The divergence-free flow condition:

$$u_x + v_y + w_z = 0$$

implies:

$$\begin{aligned} u_{xx} + v_{yx} + w_{zx} &= 0, \\ u_{xy} + v_{yy} + w_{zy} &= 0, \\ u_{xz} + v_{yz} + w_{zz} &= 0. \end{aligned} \quad (3.77)$$

As previously, assume a constant viscosity for each phase, on both sides of the interface:

$$\nabla \cdot \mu(\nabla \mathbf{u} + \nabla \mathbf{u}^t) = \mu \begin{pmatrix} 2u_{xx} + v_{xy} + u_{yy} + w_{xz} + u_{zz} \\ u_{yx} + v_{xx} + 2v_{yy} + w_{yz} + v_{zz} \\ u_{zx} + w_{xx} + v_{zy} + w_{yy} + 2w_{zz} \end{pmatrix}.$$

Then, using eq. (3.77):

$$\nabla \cdot \mu(\nabla \mathbf{u} + \nabla \mathbf{u}^t) = \mu \begin{pmatrix} u_{xx} + u_{yy} + u_{zz} \\ v_{xx} + v_{yy} + v_{zz} \\ w_{xx} + w_{yy} + w_{zz} \end{pmatrix}.$$

For the discretization of the diffusive operator \mathbf{D} , we use first order $FD(s)$:

$$\begin{aligned}
 (Dx)_{i-1/2,j,k}^n &= \frac{1}{V} \oint_S \mu(\nabla \mathbf{u} + \nabla \mathbf{u}^t) \cdot \mathbf{n} \, ds \Big|_x \\
 &= \frac{1}{\Delta x \Delta y \Delta z} \left\{ [\mu(2u_x)_{i,j,k} - \mu(2u_x)_{i-1,j,k}] \Delta y \Delta z \right. \\
 &\quad + [\mu(u_y + v_x)_{i-1/2,j+1/2,k} - \mu(u_y + v_x)_{i-1/2,j-1/2,k}] \Delta x \Delta z \\
 &\quad \left. + [\mu(u_z + w_x)_{i-1/2,j,k+1/2} - \mu(u_z + w_x)_{i-1/2,j,k-1/2}] \Delta x \Delta y \right\}
 \end{aligned} \tag{3.78}$$

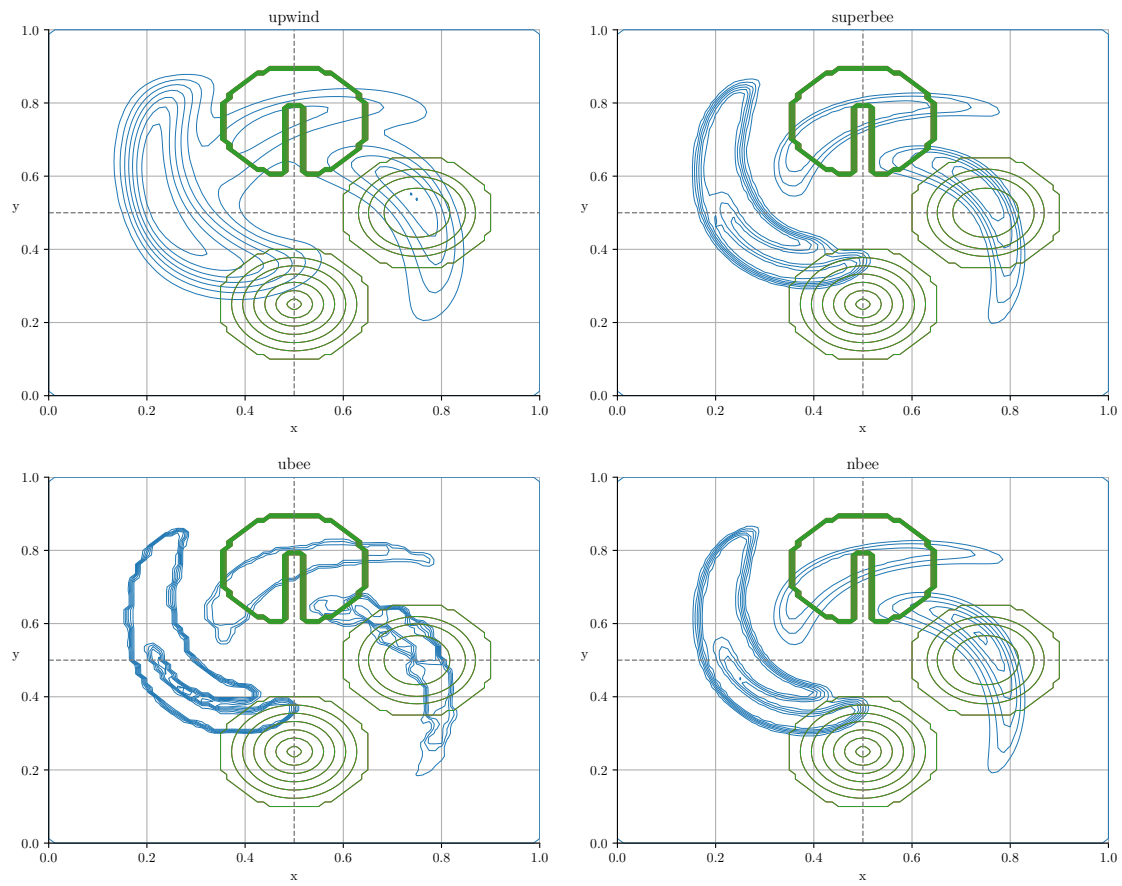
$$\begin{aligned}
 (Dy)_{i,j-1/2,k}^n &= \frac{1}{V} \oint_S \mu(\nabla \mathbf{u} + \nabla \mathbf{u}^t) \cdot \mathbf{n} \, ds \Big|_y \\
 &= \frac{1}{\Delta x \Delta y \Delta z} \left\{ [\mu(v_x + u_y)_{i+1/2,j-1/2,k} - \mu(v_x + u_y)_{i-1/2,j-1/2,k}] \Delta y \Delta z \right. \\
 &\quad + [\mu(2v_y)_{i,j,k} - \mu(2v_y)_{i,j-1,k}] \Delta x \Delta z \\
 &\quad \left. + [\mu(v_z + w_y)_{i,j-1/2,k+1/2} - \mu(v_z + w_y)_{i,j-1/2,k-1/2}] \Delta x \Delta y \right\}
 \end{aligned}$$

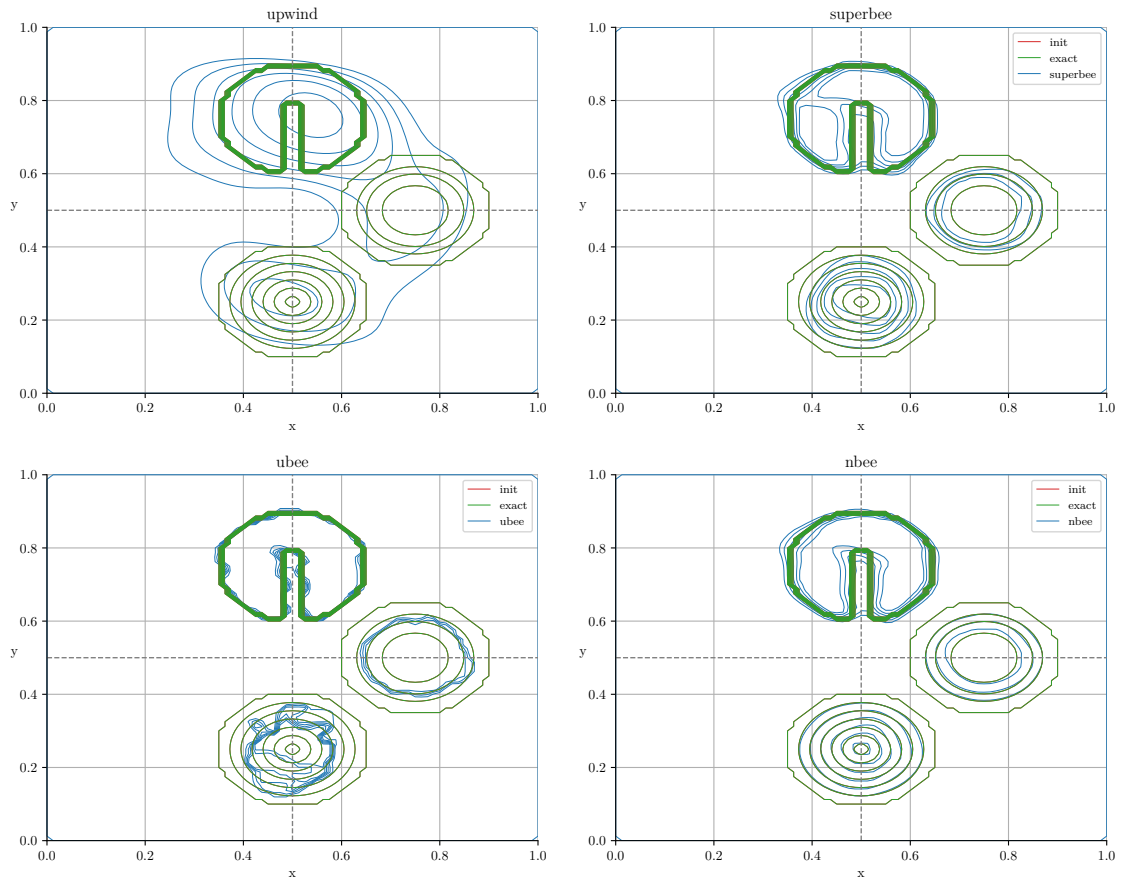
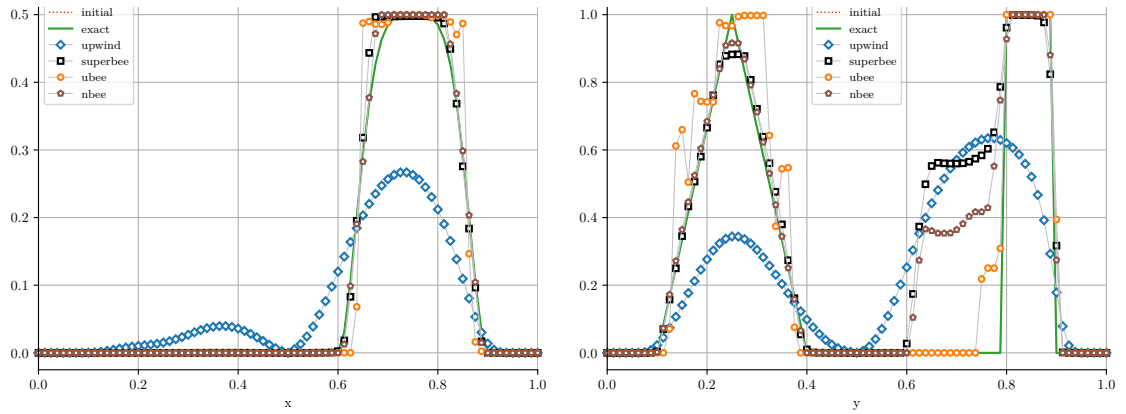
$$\begin{aligned}
 (Dz)_{i,j,k-1/2}^n &= \frac{1}{V} \oint_S \mu(\nabla \mathbf{u} + \nabla \mathbf{u}^t) \cdot \mathbf{n} \, ds \Big|_z \\
 &= \frac{1}{\Delta x \Delta y \Delta z} \left\{ [\mu(w_x + u_z)_{i+1/2,j,k-1/2} - \mu(w_x + u_z)_{i-1/2,j,k-1/2}] \Delta y \Delta z \right. \\
 &\quad + [\mu(w_y + v_z)_{i,j+1/2,k-1/2} - \mu(w_y + v_z)_{i,j-1/2,k-1/2}] \Delta x \Delta z \\
 &\quad \left. + [\mu(2w_z)_{i,j,k} - \mu(2w_z)_{i,j,k-1}] \Delta x \Delta y \right\}.
 \end{aligned}$$

3.3.5.4 Bi-dimensional splitting on high order schemes

In this test case [149, 180], a deformative velocity field is applied to a bi-dimensional field. The flow sign is reversed at $t = T/2$, so that the time $t = T$ should in theory yield the initial condition at $t = 0$.

This test case takes an initial bi-dimensional scalar field in a unit length box $x, y \in [0; 1]$, with an initial condition set to represent height levels. We thus plot in 3D a cone, with a singular maximum, a smoothly varying shape - bump, and a slotted disk (U-shape). Each shape has interesting properties from a numerical point of view, as mentioned in section 3.3.1.2. The radial velocity field (counter-clockwise rotation) is applied with maximum located at $R = .25$, centred at $(x, y) = (.5, .5)$, and a minimum at $R = 0$ and 1 . Consequently, the maximal deformation is observed in the annular portion around $R = .25$ for $\theta \in [0; 2\pi]$. The fig. 3.38 exposes the field at instant of maximal deformation $t = T/2$ in blue with the initial condition in green. The flow sign is now reversed so that the velocity field acts positively in the clockwise direction. At

Figure 3.38 – $t = T/2$, instant of maximal deformation.

Figure 3.39 – $t = T$, do we recover the initial conditions ?Figure 3.40 – Slices along the x and y directions.

$t = T$, the fig. 3.39 shows the numerical computed field in blue superimposed with the initial condition, the goal being to recover the initial field with the least spatial error in the chosen norm. The diffusive upwind scheme is also plotted in order to assess the improvements brought by the flux limiting approaches. Even with fine grid resolution, the upwind scheme eventually smears the interface in transient or steady state. Slices at $x = .5$ and $y = .5$ are extracted from fig. 3.39 (top view) and displayed in fig. 3.40 (side view). For the pyramid on the right figure, the UB seems to be unstable, with a local compressive effect. However, the maxima is conserved accurately (maximum principle). On the other hand, the SB scheme has a global compressive behaviour as seen on the left figure, with the bump gradually becoming a gate function. Also, we observe consequent loss of the maximum value on the pyramid singular point. Finally, the NB seems to be a good compromise between accuracy of the solution (conservation of the integral properties, hence mass) and spatial diffusion. This behaviour is also consistent on the right figure, with the fusion of the points of the U shape in the slotted disk. As the theory predict, a stiff interface should diffuse on at most one cell (1D) and 2 or 3 cells (2D).

3.3.5.5 Three-dimensional splitting on high order schemes

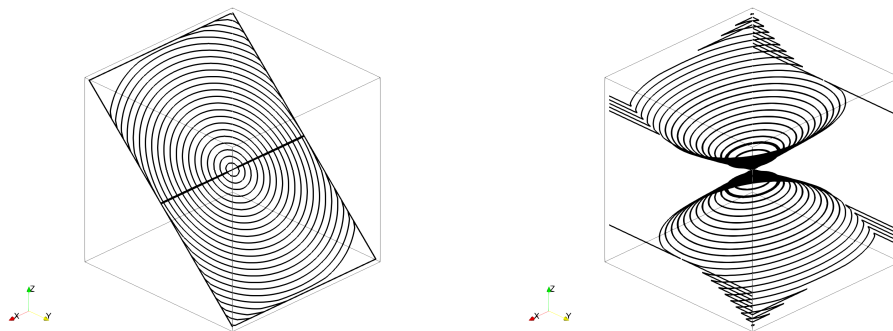
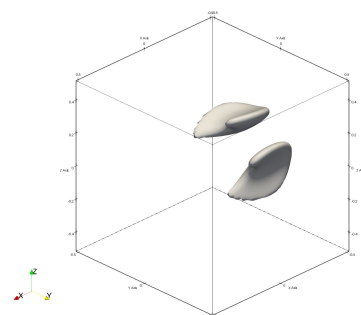
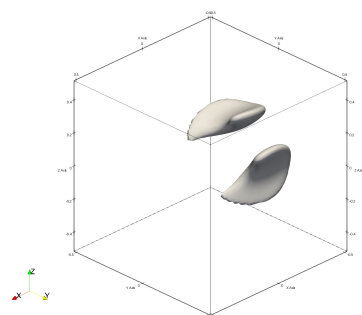
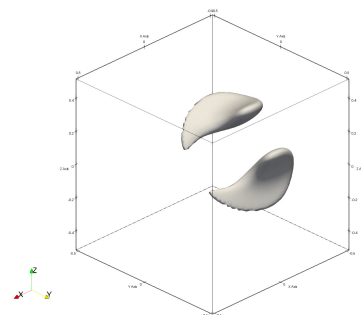
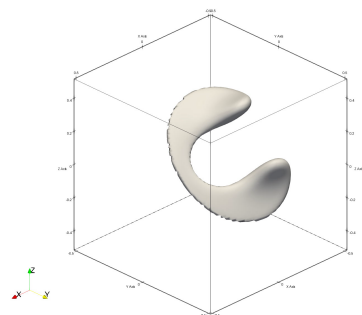
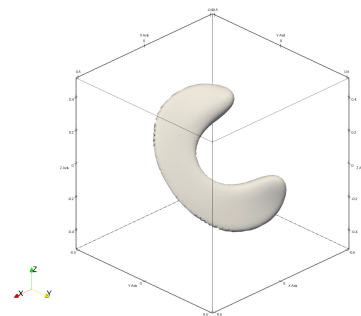
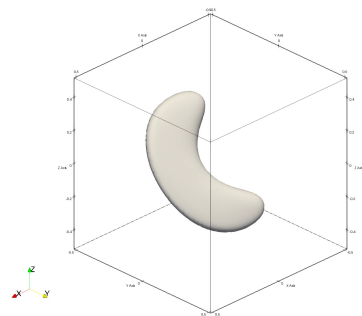
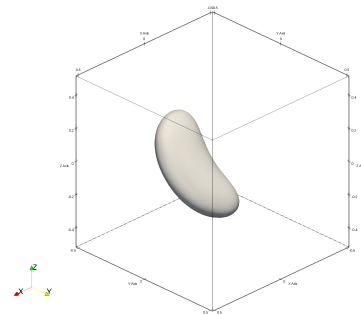
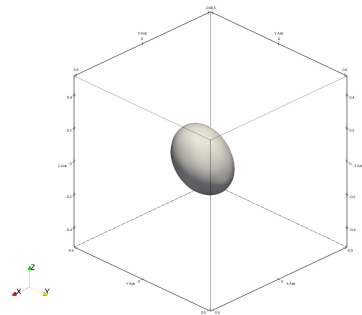


Figure 3.41 – Streamlines for the deformative test case [181].

The fig. 3.41 exposes the streamlines corresponding to a periodic velocity field modulated in time. At time $T/2$, the sign of the velocity is reversed, and we should in theory be able to recover the initial condition with some numerical errors. The sequence in fig. 3.42 shows the droplet deformation within the range $t \in [0; T]$. This was used for example in [24] with the *LSM*. We use here the scheme of [46], with a density ratio of 1000, and an isocontour of 500. The sphere is deformed by the counter rotating vortices, and some limitation of the numerical method used is to be seen when a trailing film appears because of the squeezing effect. The challenge is to recover the total mass when the film is of the order of a unit or a few cells. The thin interface is challenging for the algorithm, since it performs less well with velocities tangent to the interface. In order to increase the accuracy of the method with flux limited schemes a candidate approach would be to use adaptive refinement where the thinning of the interface is excessive [182].



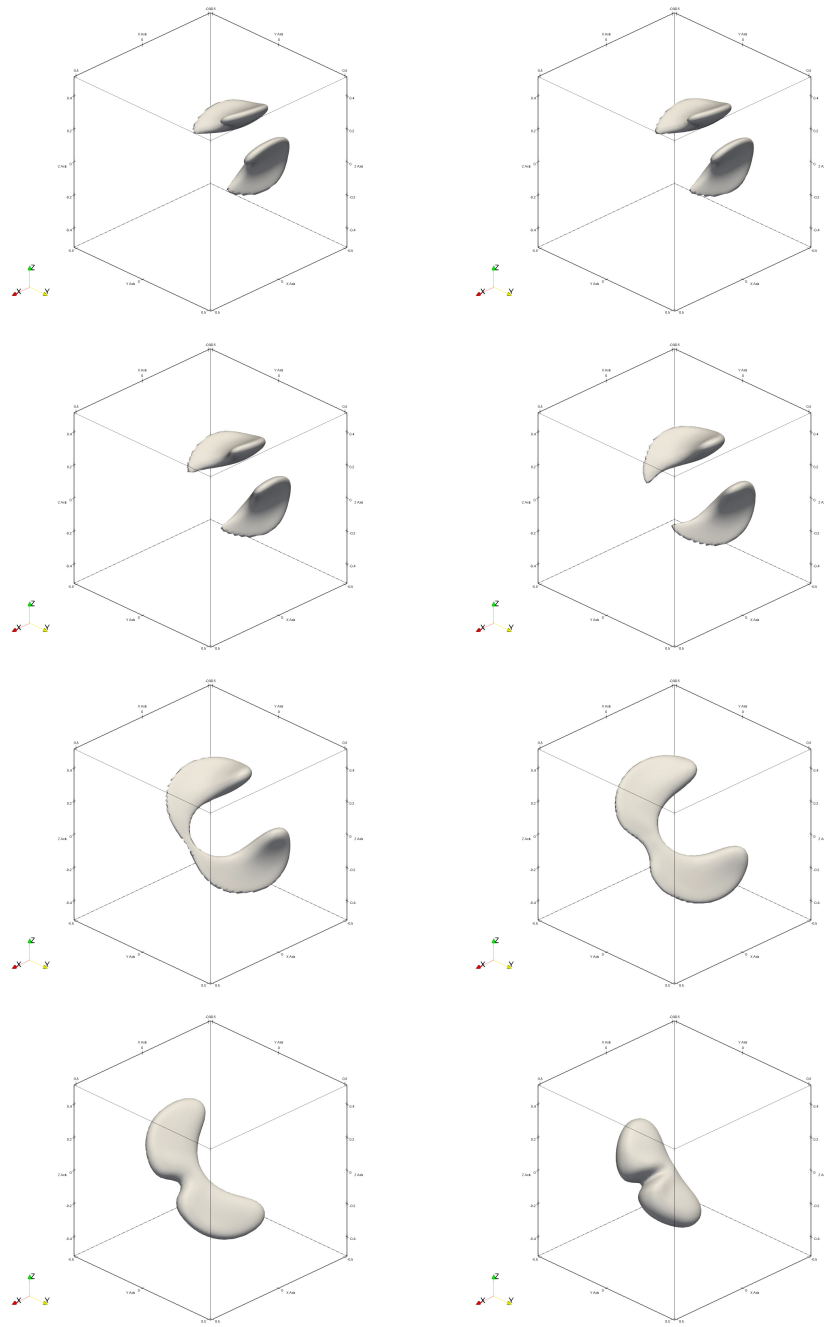


Figure 3.42 – Back and forth movement of a droplet in a deformative velocity field.

3.3.6 Fluid - solid coupling strategy

The previous parts was dedicated to the numerical methods for the fluid part. We will now justify the choices made for the numerical simulation of immersed boundaries, and how the information from one solver is transferred to another.

3.3.6.1 Coupling schemes

In order to transfer information from one solver to another, one has to choose the order of integration. Monolithic algorithms solve the equations simultaneously (strong coupling [183]), while loosely coupled algorithms induces time lag between solvers, and thus errors [184]. The possible choices are:

- explicit;
- semi-explicit;
- implicit.

A full implicit fluid solver might be interesting when the computational overhead induced by assembling and solving a linear system for multiple unknowns (velocities, pressure, ...) is balanced by allowing larger time-steps. Also, an implicit scheme induces a residual criterion, which is prone to tuning. In some of our test cases, topological source term (surface tension) dominates the time-step restriction, and we chose to take the explicit approach for the fluid solver.

The different forcing methods [125] are:

- continuous forcing, where a term is added to the continuous $NSE(s)$, before discretization;
- discrete forcing, for which a term is inserted after discretization: this offers interesting properties in term of numerical spreading, numerical stability, and discrete laws of conservation.

3.3.6.2 Implicit coupling

Numerical boundary conditions can be imposed in a direct or an indirect manner. Indirect forcing was introduced by [186, 187]: the imposed velocity field originates from the prediction step in the method of resolution of the $NSE(s)$. The boundary is thus necessarily spread over one or more conservation volumes, which locally modifies the physics close to the tube walls. On the contrary, direct forcing imposes numerical behaviour on the boundary, for instance by modifying the local stencil. However, this method is likely to cause numerical issues, when derivatives are non-continuous. For instance in [188], the authors propose to linearly interpolate the velocity in the cylinder tangential direction (i.e. when velocity gradients are sufficiently smooth), while proposing a higher order (quadratic interpolation) in the normal direction.

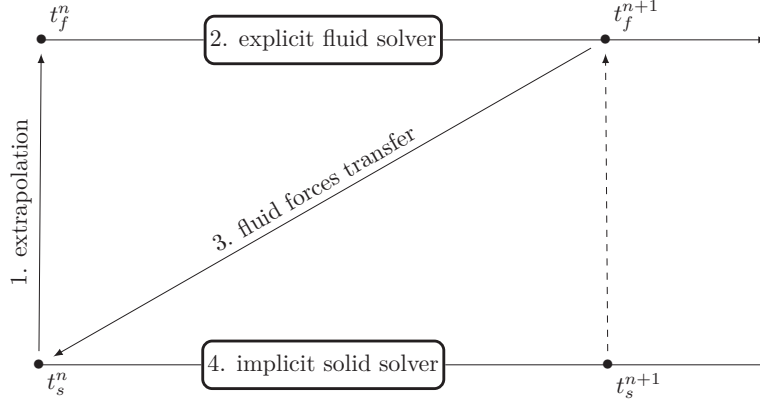


Figure 3.43 – Fluid - solid codes coupling [185].

As a work hypothesis, we choose to model the tube behaviour with a damped oscillator equation [189]. Then, we suppose that the fluid force exerted on the solid can be decomposed into a random excitation force and a fluid-elastic force [54]. This decomposition is only valid under the hypothesis of small displacements of the tube against its geometry, hence in this case the diameter of the tube. Finally, we suppose that a linearisation of the mass, damping and stiffness terms is possible under the previous hypothesis [55].

We begin with the equation of a damped oscillator:

$$(m_s + m_a)\ddot{\mathbf{q}} + (c_s + c_a)\dot{\mathbf{q}} + (k_s + k_a)\mathbf{q} = \mathbf{f}_{fe}(t) + \mathbf{f}_{al}(t). \quad (3.79)$$

The numerical integration of eq. (3.79) leaves the choice of the method of integration. The Newmark method is a numerical method, whose purpose is to solve non-linear second order differential equations. This method is used in the coupling of the solver as depicted in fig. 3.43.

The displacement and the velocity can be decomposed into a Taylor expansion:

$$\sum_{k=0}^{\infty} \frac{f^{(k)}(a)(x-a)^k}{k!},$$

under the following formulation:

$$f(t_n+h) = f(t_n) + h \cdot f'(t_n) + \frac{h^2}{2} f''(t_n) + \dots + \frac{h^k}{k!} f^{(k)}(t_n) + \frac{1}{k!} \int_{t_n}^{t_{n+1}} f^{(k+1)}(\tau) (t_n+h-\tau)^k d\tau.$$

We can thus write:

$$\dot{\mathbf{q}}_{n+1} = \dot{\mathbf{q}}_n + \int_{t_n}^{t_{n+1}} \ddot{\mathbf{q}}(\tau) d\tau, \quad (3.80a)$$

$$\mathbf{q}_{n+1} = \mathbf{q}_n + h\dot{\mathbf{q}}_n + \int_{t_n}^{t_{n+1}} \ddot{\mathbf{q}}(\tau)(t_{n+1} - \tau) d\tau. \quad (3.80b)$$

On the other hand, for $\tau \in [t_n; t_{n+1}]$:

$$\ddot{\mathbf{q}}_n = \ddot{\mathbf{q}}(\tau) + \mathbf{q}^{(3)}(\tau)(t_n - \tau) + \mathbf{q}^{(4)}(\tau)\frac{(t_n - \tau)^2}{2} + \dots, \quad (3.81a)$$

$$\ddot{\mathbf{q}}_{n+1} = \ddot{\mathbf{q}}(\tau) + \mathbf{q}^{(3)}(\tau)(t_{n+1} - \tau) + \mathbf{q}^{(4)}(\tau)\frac{(t_{n+1} - \tau)^2}{2} + \dots. \quad (3.81b)$$

By multiplying eq. (3.81a) by $1 - \gamma$ and eq. (3.81b) by γ , we obtain the following equality:

$$\ddot{\mathbf{q}}(\tau) = (1 - \gamma)\ddot{\mathbf{q}}_n + \gamma\ddot{\mathbf{q}}_{n+1} + \mathbf{q}^{(3)}(\tau)(\tau - h\gamma - t_n) + \mathcal{O}(h^2\mathbf{q}^{(4)}).$$

We now write:

$$\begin{aligned} \int_{t_n}^{t_{n+1}} \ddot{\mathbf{q}}(\tau) d\tau &= \int_{t_n}^{t_{n+1}} \left[(1 - \gamma)\ddot{\mathbf{q}}_n + \gamma\ddot{\mathbf{q}}_{n+1} + \mathbf{q}^{(3)}(\tau)(\tau - h\gamma - t_n) + \mathcal{O}(h^2\mathbf{q}^{(4)}) \right] d\tau, \\ &= (1 - \gamma)h\ddot{\mathbf{q}}_n + \gamma h\ddot{\mathbf{q}}_{n+1} + \int_{t_n}^{t_{n+1}} \mathbf{q}^{(3)}(\tau)(\tau - h\gamma - t_n) d\tau + \mathcal{O}(h^3\mathbf{q}^{(4)}), \\ &\stackrel{t_n < \tilde{\tau} < t_{n+1}}{=} (1 - \gamma)h\ddot{\mathbf{q}}_n + \gamma h\ddot{\mathbf{q}}_{n+1} + \mathbf{q}^{(3)}(\tilde{\tau}) \left[\frac{(\tau - h\gamma - t_n)^2}{2} \right]_{t_n}^{t_{n+1}} + \mathcal{O}(h^3\mathbf{q}^{(4)}), \\ &= (1 - \gamma)h\ddot{\mathbf{q}}_n + \gamma h\ddot{\mathbf{q}}_{n+1} + \underbrace{\left(\frac{1}{2} - \gamma \right) h^2 \mathbf{q}^{(3)}(\tilde{\tau})}_{r_n} + \mathcal{O}(h^3\mathbf{q}^{(4)}). \end{aligned} \quad (3.82)$$

In a similar fashion,

$$\begin{aligned} &(1 - 2\beta)eq. (3.81a) + (2\beta)eq. (3.81b) \\ &\implies \ddot{\mathbf{q}}(\tau) \\ &= (1 - 2\beta)\ddot{\mathbf{q}}_n + 2\beta\ddot{\mathbf{q}}_{n+1} + \mathbf{q}^{(3)}(\tau)(\tau - 2h\beta - t_n) + \mathcal{O}(h^2\mathbf{q}^{(4)}), \end{aligned}$$

leads to:

$$\int_{t_n}^{t_{n+1}} \ddot{\mathbf{q}}(\tau)(t_{n+1} - \tau) d\tau = \left(\frac{1}{2} - \beta\right) h^2 \ddot{\mathbf{q}}_n + \beta h^2 \ddot{\mathbf{q}}_{n+1} + \underbrace{\left(\frac{1}{6} - \beta\right) h^3 \mathbf{q}^{(3)}(\tilde{\tau})}_{\mathbf{r}'_n} + \mathcal{O}(h^4 \mathbf{q}^{(4)}) . \quad (3.83)$$

Finally, by reporting eq. (3.81b) in eq. (3.80a) and eq. (3.83) in eq. (3.80b) we obtain:

$$\begin{aligned} \dot{\mathbf{q}}_{n+1} &= \dot{\mathbf{q}}_n + (1 - \gamma)h\ddot{\mathbf{q}}_n + \gamma h\ddot{\mathbf{q}}_{n+1} , \\ \mathbf{q}_{n+1} &= \mathbf{q}_n + h\dot{\mathbf{q}}_n + h^2 \left(\frac{1}{2} - \beta\right) \ddot{\mathbf{q}}_n + h^2 \beta \ddot{\mathbf{q}}_{n+1} . \end{aligned} \quad (3.84)$$

A necessary and sufficient condition for the convergence of a numerical scheme is the consistency and the stability of the scheme. Consistency is achieved if the discretized numerical tends to the exact solution when the spatial and temporal spacing are refined, which can be written formally as: $\lim_{h \rightarrow 0} \frac{\mathbf{u}_{n+1} - \mathbf{u}_n}{h} = \dot{\mathbf{u}}(t_n)$.

Taking the former and applying it to the chosen numerical method:

$$\lim_{h \rightarrow 0} \frac{\mathbf{u}_{n+1} - \mathbf{u}_n}{h} = \lim_{h \rightarrow 0} \begin{bmatrix} (1 - \gamma)\ddot{\mathbf{q}}_n + \gamma\ddot{\mathbf{q}}_{n+1} \\ \dot{\mathbf{q}}_n + h(\frac{1}{2} - \beta)\ddot{\mathbf{q}}_n + \beta h\ddot{\mathbf{q}}_{n+1} \end{bmatrix} = \begin{bmatrix} \ddot{\mathbf{q}}_n \\ \dot{\mathbf{q}}_n \end{bmatrix} .$$

algorithm	γ	β	properties
explicit	0	0	explicit - unstable
finite differences	1/2	0	explicit - conditionally stable
Fox & Goodwin	1/2	1/12	implicit - conditionally stable
linear acceleration	1/2	1/6	—"
average acceleration	1/2	1/4	implicit - unconditionally stable

Table 3.5 – Properties of the Newmark algorithm as a function of the parameters α and β .

On the other hand, the stability of the method is subject to the parameters γ and β , as exposed in table 3.5.

3.4 Domain decomposition and implementation

After having defined a discretization and suitable numerical methods, we expose the domain decomposition approach for parallel simulation. Therefore, we will have to choose a programming language based on several requirements. For structured grids, modern *Fortran*ⁱ is an adapted language with its native multi-dimensional array support. Domain decomposition has been implemented with the help of both Coarrays [190] (*Fortran* 2008 standard) and regular arrays with the help of a *MPI*ⁱⁱ library. The dual communication-layer strategy serves as a validation of the inter-process (or inter-nodal) data exchange since the Coarrays library mimics shared memory access on a remote process and *MPI* provides the performance of blocked communications. However, we found that Coarray Fortran, as implemented in open-source compilers lacks performance and robustness, and should only be used for prototyping codes. Especially, the compiler delegates communication to an external library, and this change of behaviour with the `-fcoarray=lib` compiler flag can — and does in real practice — trigger specific *ICE(s)*ⁱⁱⁱ which are hard to get rid of (however, recent compiler aware OpenCoarrays version > 2.2.0 should be relatively bug free). For communications other than halo synchronizations, one-sided communications using *RMA*^{iv} is used as described in algorithm 10. The whole code was prototyped in Python language with *C* language bindings for critical parts in a serial version. The code was then ported to *Fortran* for distributed computation among nodes with possible heterogeneous processing units.

i. FORMula TRANslator
 ii. Message Passing Interface
 iii. Internal Compiler Error(s)
 iv. Remote Memory Access

Algorithm 10: One sided communication.

```

1 acquire a global window lock
  ! phase 1
2 for each process in a communicator do
3   | insert the slab information in an integer shared window on the target
   | process
4 end
5 release the global window lock
  ! phase 2
6 for each process in a communicator do
7   | if request present in local window then
8   |   | perform the non-blocking send of the required array section using the
   |   | local window
9   | end
10 end
    ! phase 3
11 for each process in a communicator do
12   | if data requested in phase 1 then
13   |   | wait until reception and move the data buffer into the final array section
14   | end
15 end

```

3.4.1 Conventions and sequencing

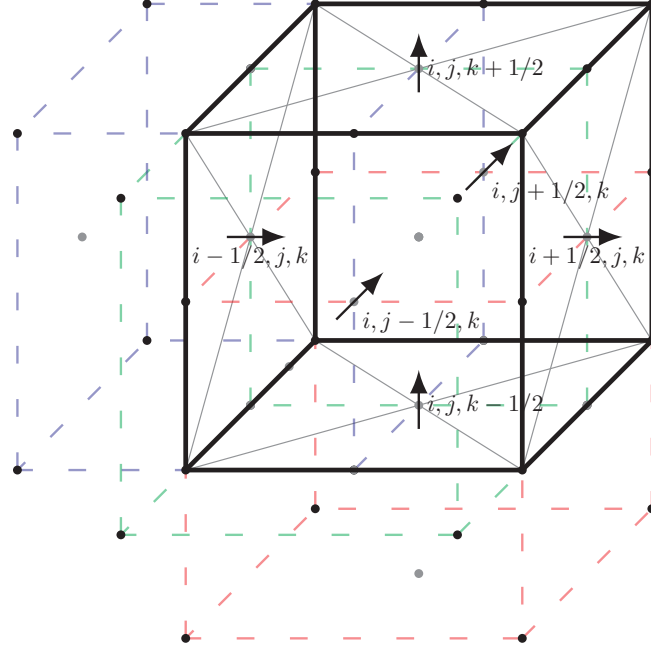


Figure 3.44 – Velocities control volumes, with 'left' convention.

Regular array decomposition induces overlap of unknowns. The convention used is the ownership of 'left' facet data by the current process. Hence, the right facet on the fig. 3.44 must locally be overwritten by data fetched — and thus computed or owned — on the remote process. The same reasoning applies for nodal and edge unknowns, since only cell-centred data does not overlap. The decomposition of sub-blocks is as follows: each process owns a three dimensional number of $i \cdot j \cdot k$ cells in the euclidean space. The process locally sees $(i + 1) \cdot (j + 1) \cdot (k + 1)$ nodes, together with:

- $(i + 1) \cdot j \cdot k$ facet velocities (u), along the x direction;
- $i \cdot (j + 1) \cdot k$ facet velocities (v), along the y direction;
- $i \cdot j \cdot (k + 1)$ facet velocities (w), along the z direction.

Accordingly, n layers of phantom (ghost) cells are added along each direction for halo synchronization. This decomposition, closest to the discretization taken, allows vectorized computation, strided access and easy pointer reshapes.

The top-level solver loop is described in algorithm 11, and reflects a high-level implementation structure, and we will now describe each computational step. We first compute the current timestep for the range $[t^n; t^{n+1}]$ based on the current velocities using the eq. (3.53). The immersed boundaries are updated using an extrapolation of $F^n \simeq F^{n-1}$ from the previous timestep, under the assumption of small displacements (using an explicit formulation for the fluid solver enforces this restriction). The density field is

regularized using algorithm 1:

$$\bar{\rho} := \text{function}(\rho).$$

Vertices normals are computed using at least second order finite differences for accuracy using $\bar{\rho}$. Facet curvatures can now be extracted from vertex normals using eq. (3.49). Control volume facet surface tension pulse is computed using density gradients magnitude and is oriented along the aforementioned normals. The predicted velocity $\tilde{\mathbf{u}}^{n+1/2}$ is assembled using:

- convective operator of section 3.3.5.2 and eq. (3.72), or a higher order scheme described in section 3.3.2;
- diffusive operator of section 3.3.5.3 and eq. (3.78);
- buoyancy terms eq. (4.9);
- capillary terms eq. (3.48).

An optional implicit step is performed for the viscous operator applying algorithm 7 to compute $\tilde{\mathbf{u}}^{n+1}$. Velocity is imposed from the solid part to the fluid fields, with the pressure spikes regularization method described in section 3.5.3. This step occurs before the pressure splitting step, because the velocity transition from the solid phase to the fluid phase is not divergence free, resulting in potential expansion of the fluid and gain/loss of mass. An implicitation step is now applied for the two-phase Poisson equation as exposed in section 3.3.4:

$$\nabla \cdot \left(\frac{1}{\rho} \nabla p^{n+1} \right) = \frac{1}{\Delta t} \nabla \cdot \tilde{\mathbf{u}}^{n+1},$$

using iterative methods and optional multi-grid acceleration of section 3.3.4.2. Hereby follows the correction step using the new pressure multiplier eq. (3.19). The forces around the immersed obstacle \mathbf{F}_s^{n+1} are computed using the divergence free velocity \mathbf{u}^{n+1} and pressure p^{n+1} :

$$\mathbf{F}_s^{n+1} = \text{function}(u^{n+1}, p^{n+1}),$$

and will be used in the next timestep using extrapolation. The density is convected in a conservative way as a passive scalar using eq. (3.63), where each flux is computed using either algorithm 5 or algorithm 6, depending on the chosen formalism for flux limited schemes as seen in section 3.3.1.2. From the new computed density ρ^{n+1} , we extrapolate the updated viscosity μ^{n+1} using eq. (3.17).

Algorithm 11: Temporal discretization.

```

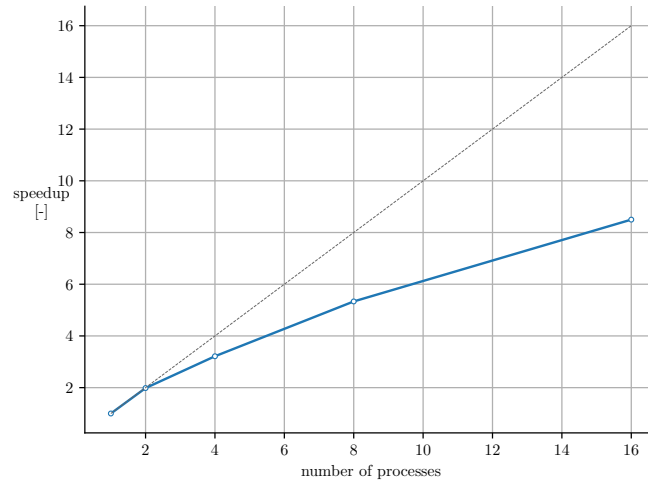
1 fields initialization or computation restart from a checkpoint (density, velocities,
  pressure)
2 while end of computation not reached do
3   timestep restriction update eq. (3.53)
4   IBM displacement of the obstacles, compute new positions
5   regularization of the density field for capillary term using algorithm 2
6   advection operator contribution from the previous velocity field  $\mathbf{u}^n$ 
7   diffusion operator contribution from the previous velocity field  $\mathbf{u}^n$ 
8   capillary operator contribution (curvatures and capillary force computation)
9   velocity prediction step (using explicit or semi-implicit step)
10  IBM contribution (force solid velocities inside the fluid solver)
11  implicit pressure step for the spatially variable pressure part of the  $NSE(s)$ 
12  velocity correction step for divergence free flows
13  computation of the fluid forces on the obstacles for the IBM
14  conservative phase transport eq. (3.63)
15  viscosities extrapolation eq. (3.17)
16  periodic insitu post treatment and data input/output
17 end
18 memory release and exit

```

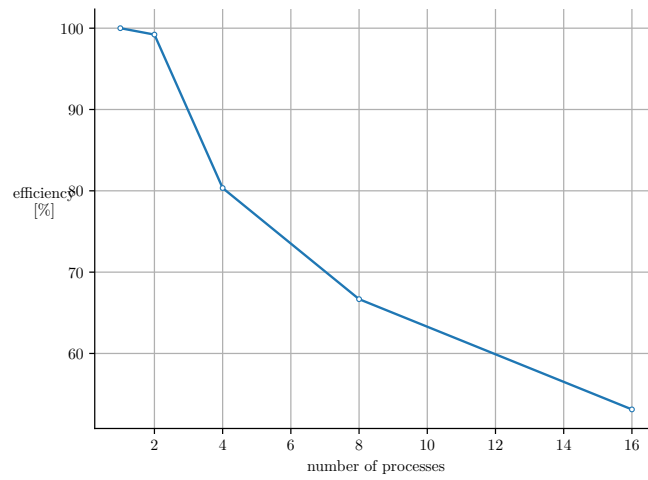
3.4.2 Performance

The fig. 3.45 shows the strong scaling on the first steps on the bubble rise test case. For a fixed problem size, the program is run under increasing number of processes, with extra care for performance flags (compilation, linking to external libraries), CPU - memory affinities, input/output minimization. The speed-up and efficiency observed on fig. 3.45 are reasonable for such application, but the code has to be run over multiple nodes in a cluster *NUMA*^v environment. Indeed, the test is biased since data movement does not transit over the network. Using hyper-threading (logical cores exceeding the numbers of physical cores) deteriorates speed-up: we only show speed-up for $p \leq 16$ processes. Although it is interesting to develop a highly optimized code with excellent scaling capabilities, it is beyond the development of a “proof of concept” code.

v. Non Uniform Memory Access



(a) Speed-up with theoretical bound.



(b) Efficiency of the code.

Figure 3.45 – Strong-scaling for the bubble rise problem ($MG(2)$ pressure solver).

3.5 Validation

Before analysing and interpreting results, there is a need to quantify and assess the limitations of the numerical methods or models. It thus necessary to establish a validation criterion that allows an objective quantification of the difference between the results and what is observed in reality. This validation process is usually done by testing each model or scheme within an isolated framework, where analytic solution or references are available.

3.5.1 Mono-phase flows

In order to validate the implementation of eq. (3.6), we compare code results to analytical solutions, or make use of the *MMS*ⁱ [191]. The method uses analytic solutions as input fields, and the output is derived by hand for the multiple operators. Then, the numerical solution is compared to the expected analytical solution, and an error measure can be derived from these results.

As an example, in order to validate the diffusion operator, we impose a pressure gradient on a three-dimensional rectangular channel and observe the Poiseuille profile. The flow parameters are $Re = 100$ and $\mu = .01$.

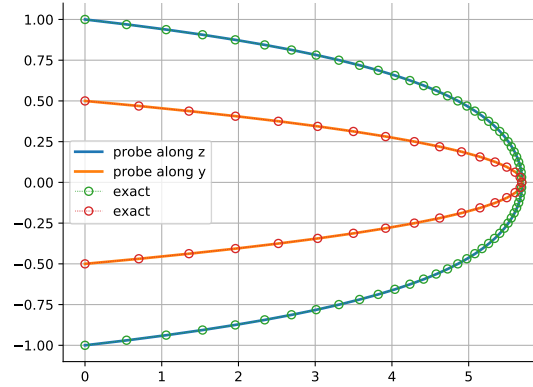


Figure 3.46 – Vertical and transverse velocity profiles in a rectangular channel.

The horizontal velocity profile for an arbitrary rectangular channel of width w and height h in the $y \in [-w/2; w/2]$, $z \in [0; h]$ directions has the analytical solution:

$$u(y, z) = \frac{4h^2 \Delta p}{\pi^3 \mu L_x} \sum_{n, \text{odd}} \frac{1}{n^3} \left(1 - \frac{\cosh(n\pi y / h)}{\cosh(n\pi w / (2h))} \right) \sin \left(n\pi \frac{z}{h} \right), \quad (3.85)$$

where Δp is the imposed pressure gradient in the x direction of the length L_x .

The fig. 3.46 shows satisfying superposition (agreement) of the steady state computed profiles and the analytical profiles eq. (3.85).

i. Method of Manufactured Solutions

3.5.2 Laplace pressure

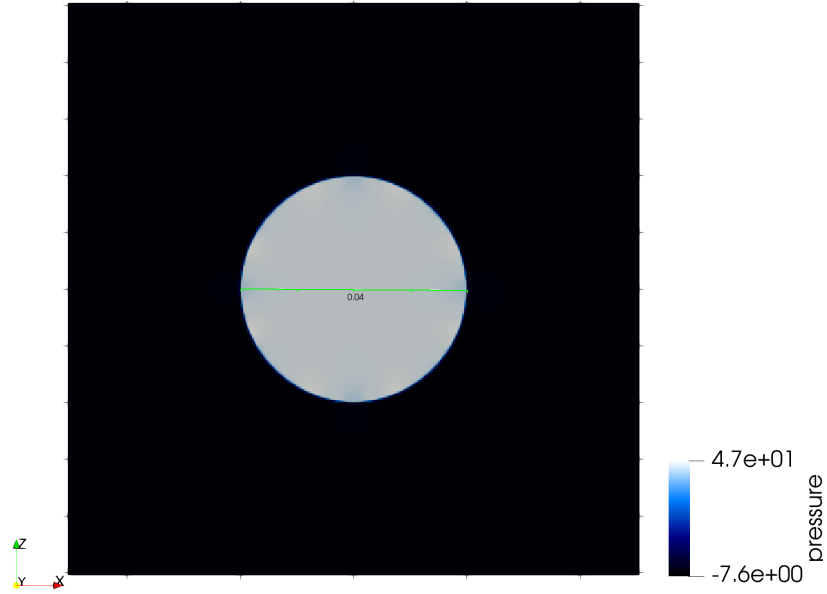


Figure 3.47 – Pressure surge on a static bubble, induced by capillary forces at the interface.

On a static droplet, with stiff interfacial terms as shown on fig. 3.48, we recover a pressure jump of 54.6, and with a radius of .04 [m] on fig. 3.47, the mean curvature being $\kappa = \frac{1}{.04} = 25$, and the error of eq. (3.7) being equal to 9 [%]. This error might be induced by curvature gradients, and errors in normals estimations. The density is regularized using the techniques described in section 3.1.3.3. In the static droplet case, a dynamic equilibrium of the phases must be reached in order to yield the exact pressure, and parasitic currents as evoked in section 4.1.2 prevent this exact steady state.

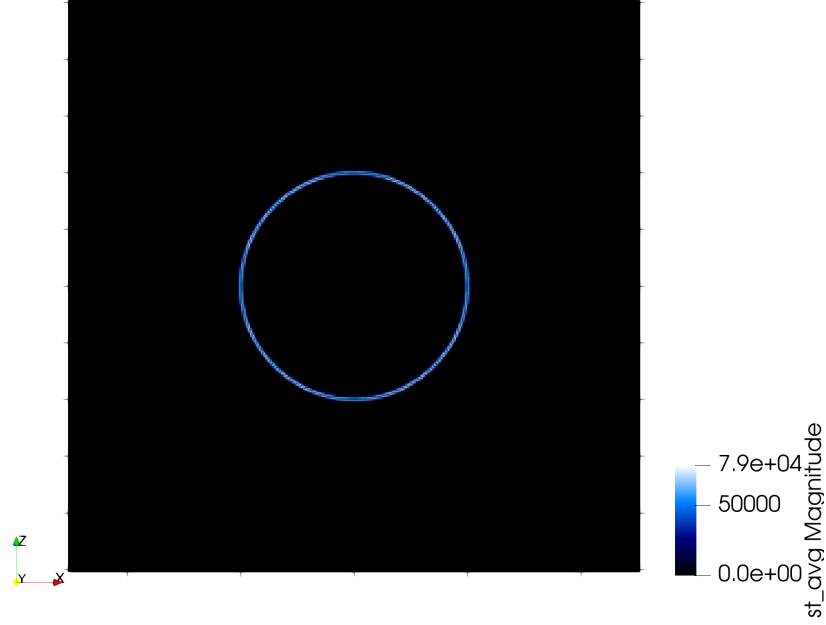


Figure 3.48 – Stiff interfacial capillary source term.

3.5.3 Pressure oscillation regularization

Displacement of the centre of gravity toggles a state transition from a fluid state onto a solid state and vice-versa. The effect of this transition is reflected on the pressure field because of the instantaneous relaxation of incompressible flows. A method is therefore sought in order to reduce the amplitude of these pressure spikes. We use the regularization method proposed by [192].

The direct forcing consists of imposing the solid velocity on all surround fluid cells. Consequently, the pressure solver will adapt the pressure to satisfy the numerical incompressibility constraint.

When using a direct forcing, the DC^{ii} contribution to the pressure oscillation evolves as $\mathcal{O}\left(\frac{\Delta x^2}{\Delta t}\right)$, whereas the FC^{iii} contribution follows a $\mathcal{O}(\Delta x)$ behaviour. The main contributing mechanism is thus the transition from a fluid cell to a solid cell DC .

We use a pressure oscillation indicator: $C_p^{2\delta} = C_p^{n+1} - 2C_p^n + C_p^{n-1}$, as an indicator of pressure jump and discontinuities [125].

$C_{P,MAX}^{2\delta}$ quantifies the pressure oscillations, which originates from the transition of control volume from fluid to solid state DC , whereas $C_{P,\sigma}^{2\delta}$ quantifies the solid to fluid

ii. Dead Cell

iii. Fresh Cell

FC change of state but also the contribution of the DC cells. In [193] it is also shown that the pressure oscillations for the DF^{iv} method evolves as $\mathcal{O}(1) + \mathcal{O}(\Delta x) + \mathcal{O}\left(\frac{\Delta x^2}{\Delta t}\right)$, and $\mathcal{O}(1) + \mathcal{O}(\Delta x) + \mathcal{O}(\Delta t)$ for the RG^{v} method.

We use the directional derivative $D_{\mathbf{u}} f = \nabla(f) \cdot \mathbf{u} \cos(\theta)$ in order to compute the surface cylinder quantities.

In the bi-dimensional (x, z) plane we have:

$$\mathbf{u} = \begin{pmatrix} u_1 \\ u_3 \end{pmatrix}, \mathbf{n} = \begin{pmatrix} n_1 \\ n_3 \end{pmatrix} \text{ and } \mathbf{t} = \begin{pmatrix} n_3 \\ -n_1 \end{pmatrix}.$$

Using the previously defined vectors, we write the normal derivative of the tangential velocity:

$$\frac{\partial \mathbf{u}_t}{\partial \mathbf{n}} = (u_{1_x} n_3 + u_{3_x} n_1) n_1 \mathbf{e}_1 + (u_{1_z} n_3 + u_{3_z} n_1) n_3 \mathbf{e}_3. \quad (3.86)$$

Since the cylinder velocity is imposed, the tangential velocity is constant along the tangential direction:

$$\begin{cases} \frac{\partial u_1}{\partial \mathbf{t}} = u_{1_x} n_3 - u_{1_z} n_1 = 0, \\ \frac{\partial u_3}{\partial \mathbf{t}} = u_{3_x} n_3 - u_{3_z} n_1 = 0. \end{cases} \quad (3.87)$$

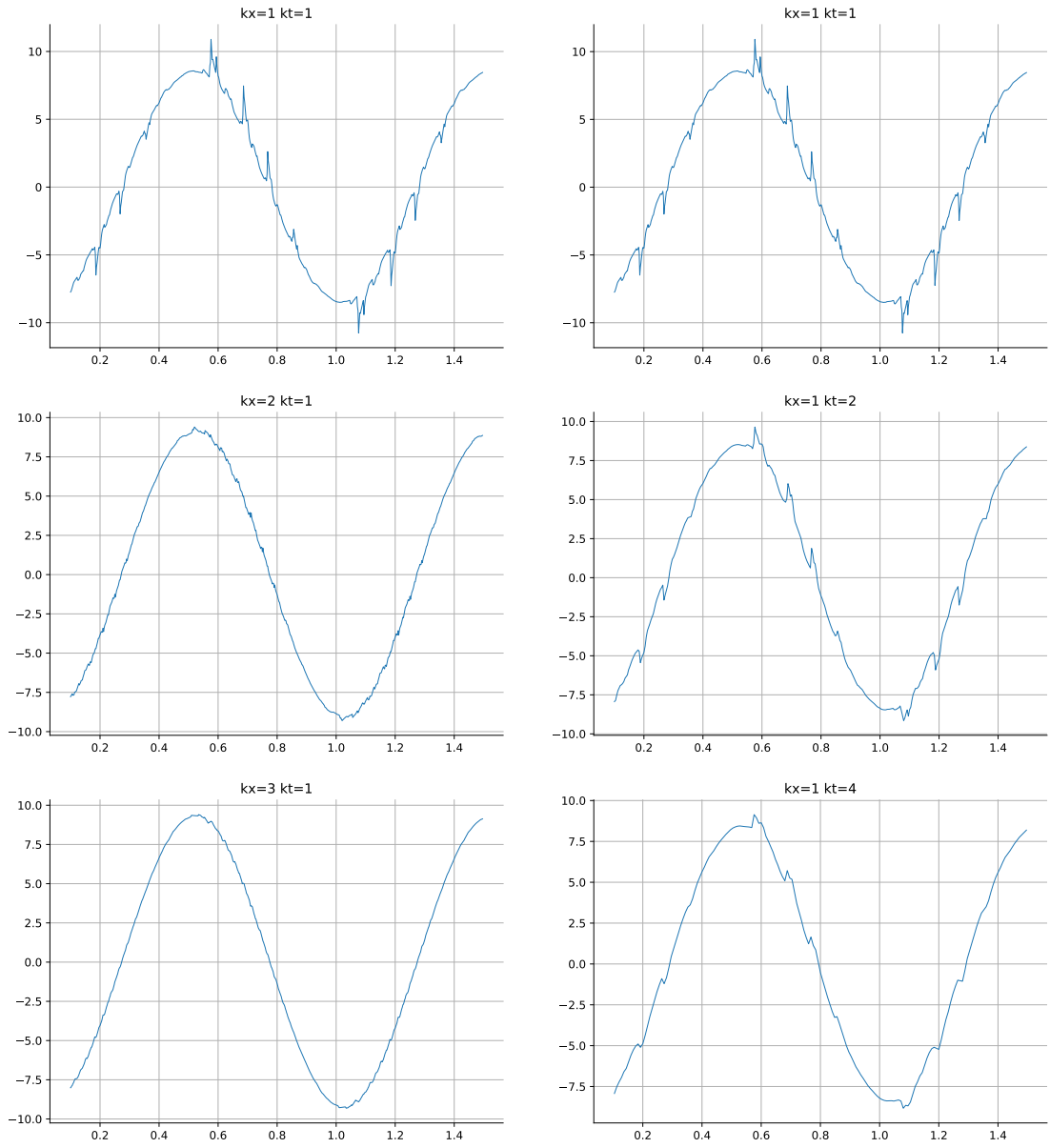
Consequently, using eq. (3.87) in eq. (3.86), we can write:

$$\begin{aligned} \frac{\partial \mathbf{u}_t}{\partial \mathbf{n}} &= (u_{1_z} n_1 - u_{3_x} n_1) n_1 \mathbf{e}_1 + (u_{1_z} n_3 - u_{3_x} n_3) n_1 \mathbf{e}_3 \\ &= u_{1_z} \underbrace{(n_1^2 + n_3^2)}_{=1} \mathbf{e}_1 - u_{3_x} \underbrace{(n_1^2 + n_3^2)}_{=1} \mathbf{e}_3 \\ &= u_{1_z} \mathbf{e}_1 - u_{3_x} \mathbf{e}_3. \end{aligned} \quad (3.88)$$

The alternative methods shown on figs. 3.50 to 3.52 expose the clear contribution to the reduction of the numerical pressure surges observed on fig. 3.49 (coarser space step with a factor $k \in \{1, 2, 3\}$ on the left, and reduced timestep with a factor $k \in \{1, 2, 4\}$).

The DF method shows spurious pressure surges on the lift coefficient. The spread forcing method fig. 3.50 allows reduction of the pressure spikes. A finer mesh contributes to the reduction of the effect observed, whilst relaxing the timestep impacts the reduction of the surges. The smooth regularization on fig. 3.51 and the piecewise linear regularization of fig. 3.52 performs best for the objective.

iv. Direct Forcing
v. ReGularized DF

Figure 3.49 – Direct forcing ($kt := \Delta t, kx := \Delta x$).

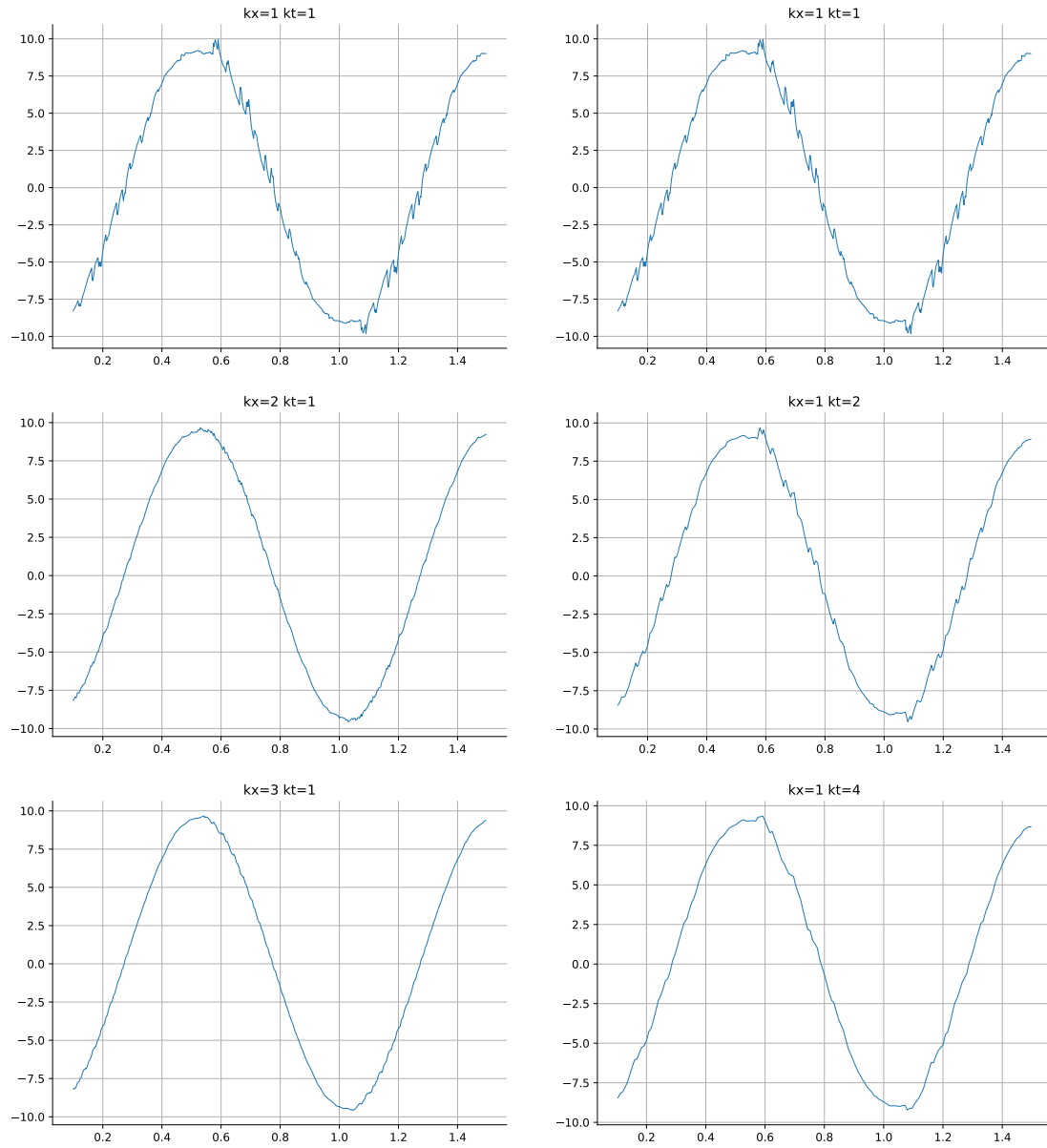


Figure 3.50 – Spread forcing.

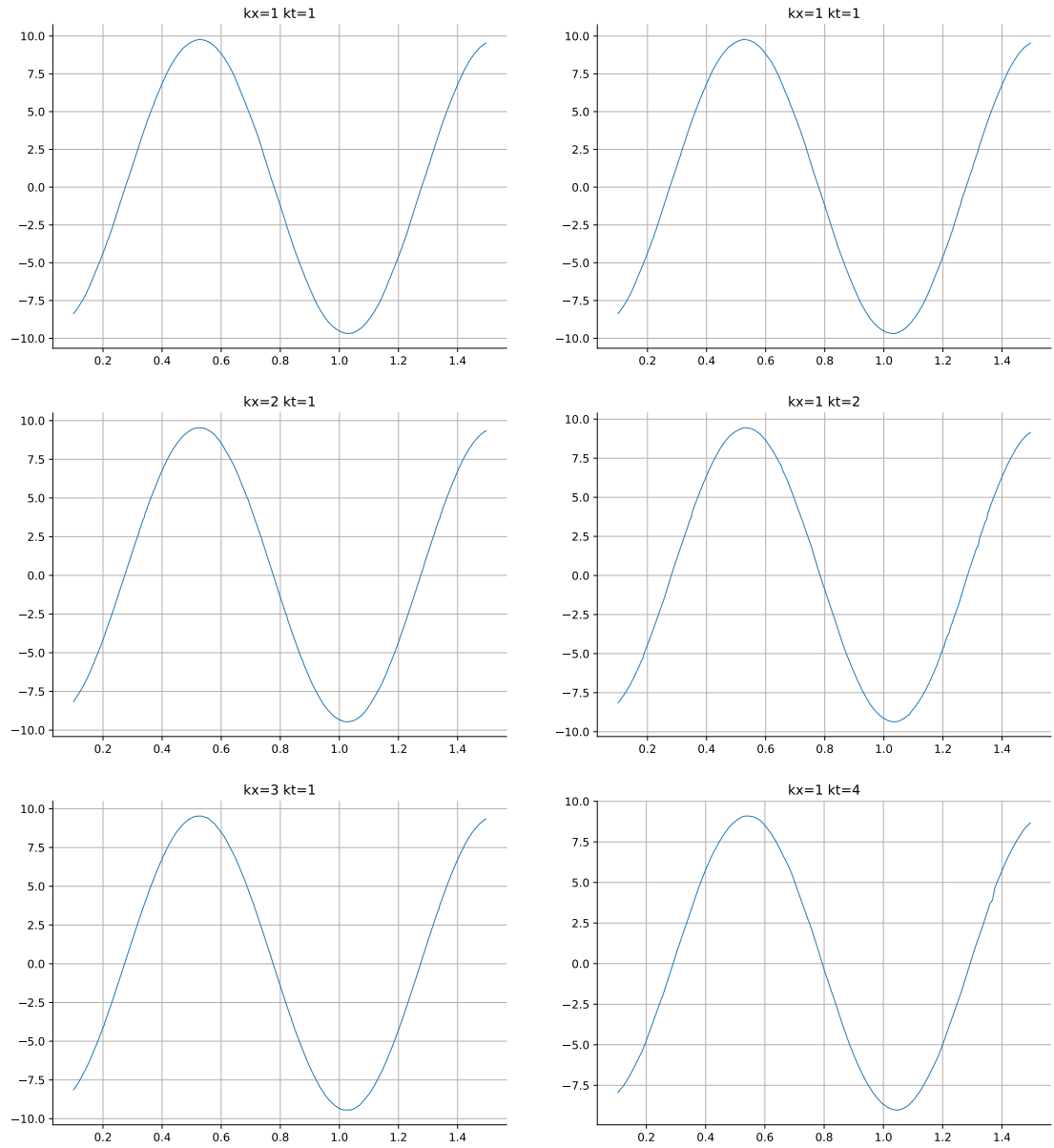


Figure 3.51 – Smooth regularization.

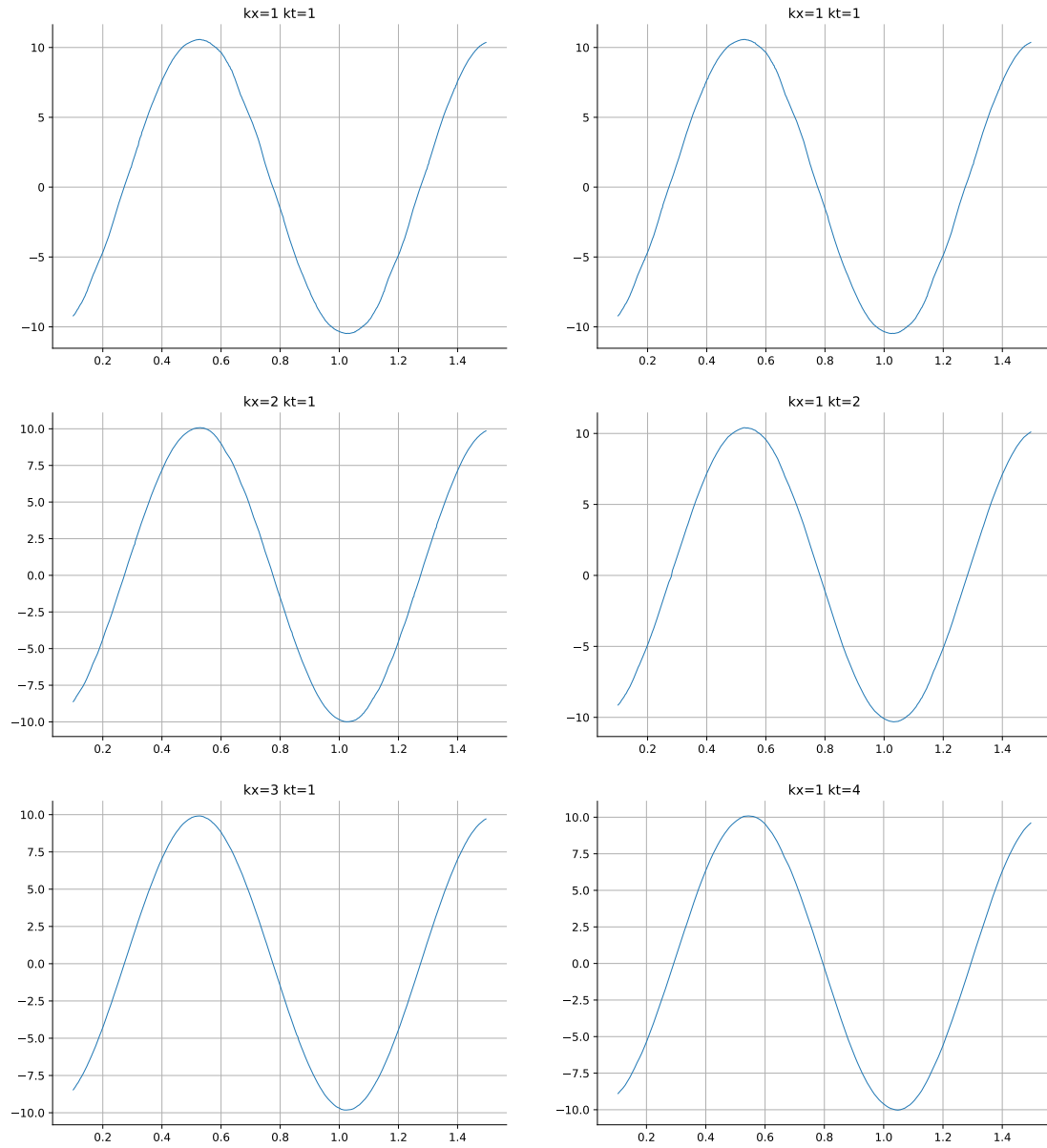


Figure 3.52 – Piecewise linear regularization.

3.5.4 Mono-phase cross-flow around a cylinder

The cross-flow around a cylinder test case is used in order to validate the numerical behaviour of immersed boundaries in mono-phase flow. A stagnation zone appears to form upstream of the cylinder. The flow departs on the upper or the lower section of the cylinder, with possible detachment depending on the laminar/turbulent regime. The downstream region exhibits well know Von Karman street of vortices [194] as observed on fig. 3.53. With this kind of flow, the correct resolution of the fluid boundary layer on the cylinder is key to capture the correct detachment frequencies in the wake of the cylinder.

ρ [kg·m ⁻³]	μ [kg·m ⁻¹ ·s ⁻¹]	Re [-]	U_{inlet} [m·s ⁻¹]			
1.	$1 \cdot 10^{-3}$	100	1.5			
				U [m·s ⁻¹]	D [m]	domain [m]
				1.	.1	2.2x.41
						no. nodes
						1025x193

Table 3.6 – Fine grid parameters for the unsteady 2D case.

We use the unsteady test case presented in [195], with focus on the 2D results of [196]. The configuration parameters are found in table 3.6, where D is the cylinder diameter, and $U = 2U_{\text{inlet}} / 3$ is the nondimensionalization velocity.

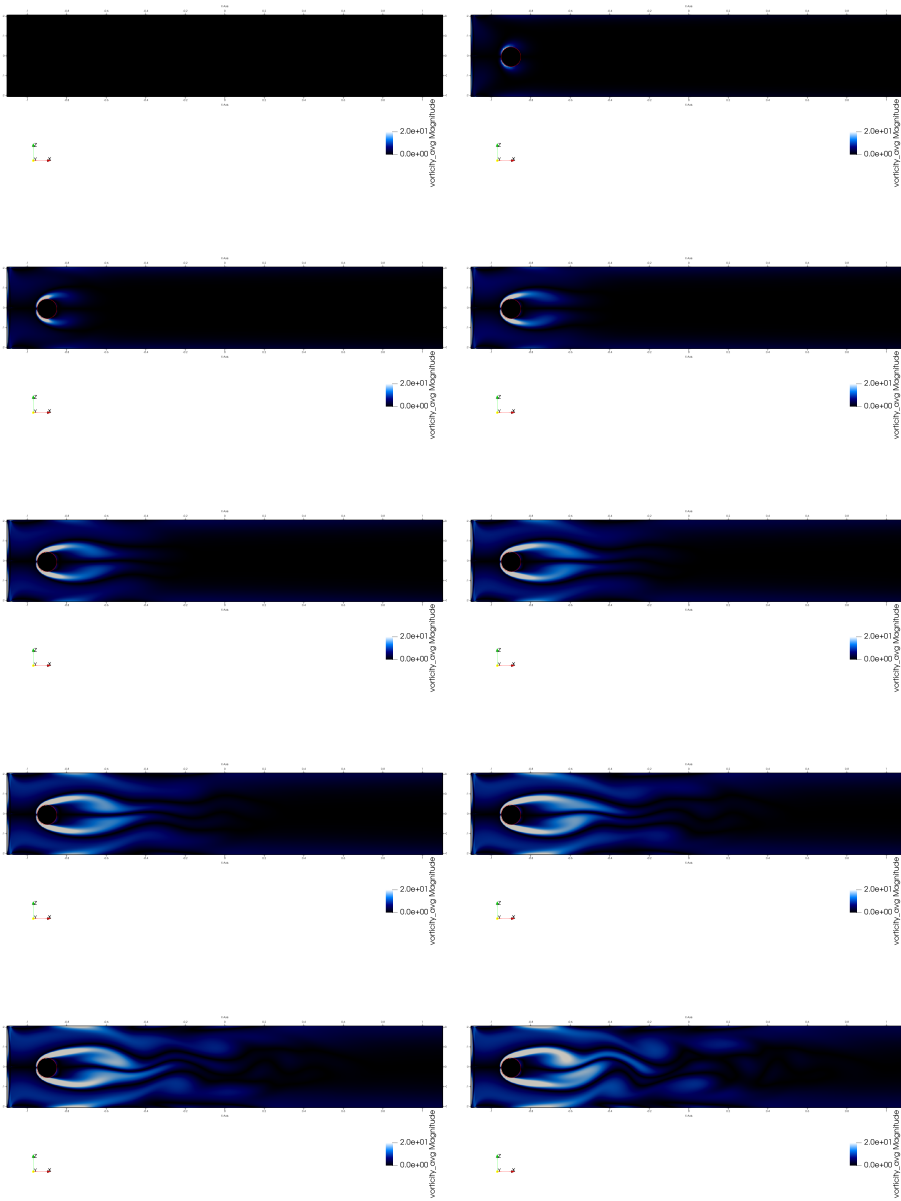
The pressure and viscous contributions to the total forces are computed using volumetric integrals \iiint or surface integrals \oint on the fixed cylinder. The inlet velocity profile is set to $u(0, z, t) = 4Uz(L_z - z)\sin(\pi t / 8) / L_z^2$ [m·s⁻¹], and is time dependent, so that the maximal inlet velocity is reached at $t = 4$ [s].

The number of control volume used for volumetric integration is $n^{\iiint} = 1700$, whereas the number of surface segments is $n^{\oint} = 1440$.

On fig. 3.56, the lift coefficient matches the reference $c_{l,\text{max}} = .5$ at $t = 4.9$ [s]. However, the drag coefficient seems to have been underestimated ($c_{d,\text{max}} = 2.$) at $t = 4.5$ [s].

By examination of the pressure and viscous contributions on figs. 3.54 and 3.55, this difference might be explained by the nondimensionalized coefficient based on either U or U_{inlet} .

We obtain $\Delta p(8s) = -.067$ [kg·m⁻¹·s⁻²] versus the reference [196] of $\Delta p(t = 8s) = -.111$ [kg·m⁻¹·s⁻²]. The maximum value of the pressure difference on fig. 3.57 seems to match the reference (maximum around $\Delta p = 2.35$ [kg·m⁻¹·s⁻²] at $t = 4.5$ [s]).



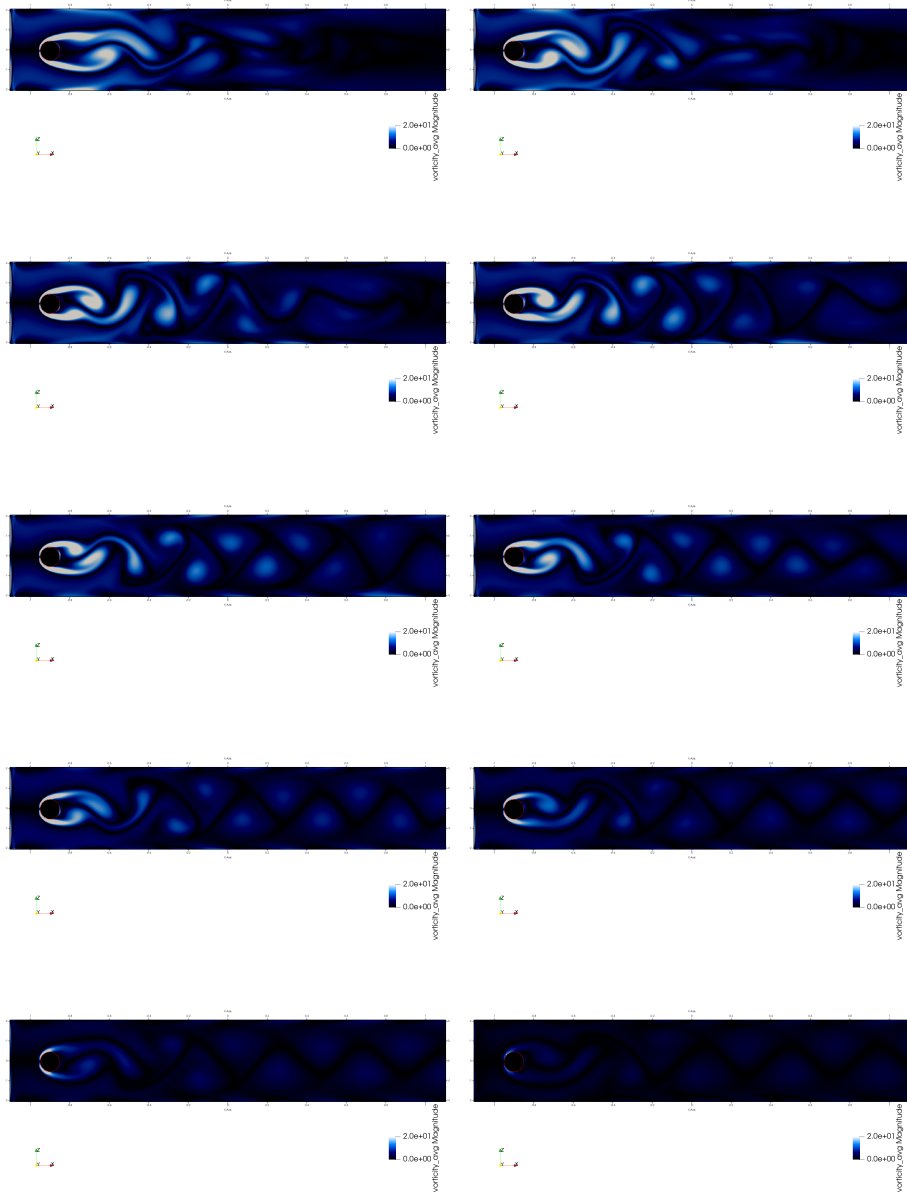


Figure 3.53 – Vorticity of cross-flow around a cylinder in a channel with inlet, wall and outflow boundaries $t \in [0; 8]$.

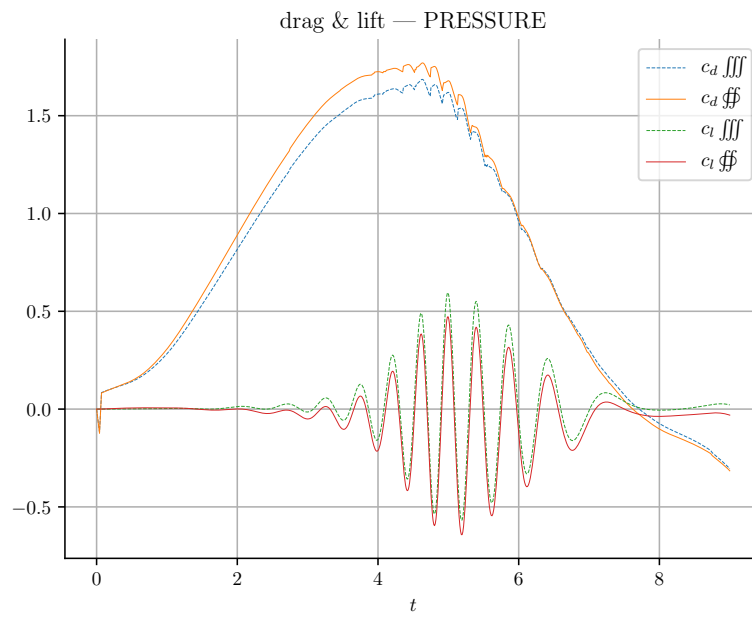


Figure 3.54 – Pressure drag and lift coefficients (volumetric and surface integration).

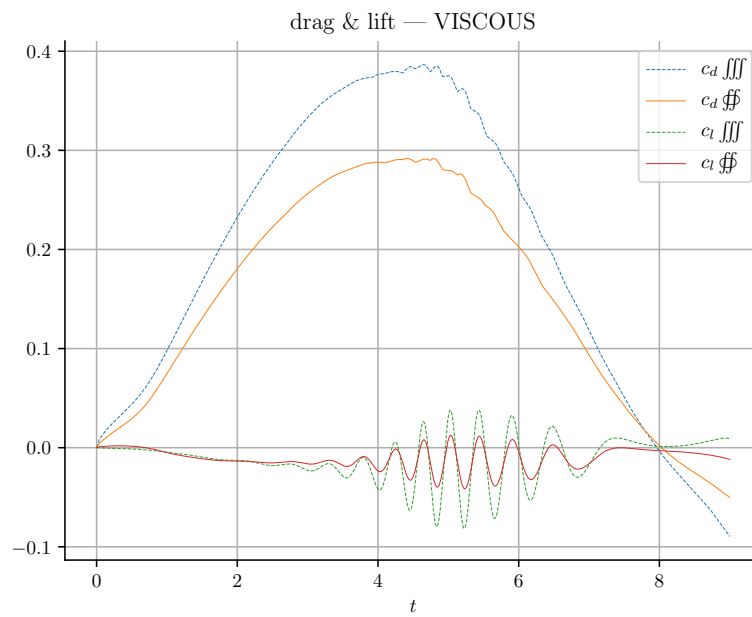


Figure 3.55 – Viscous contribution of the drag and lift coefficients.

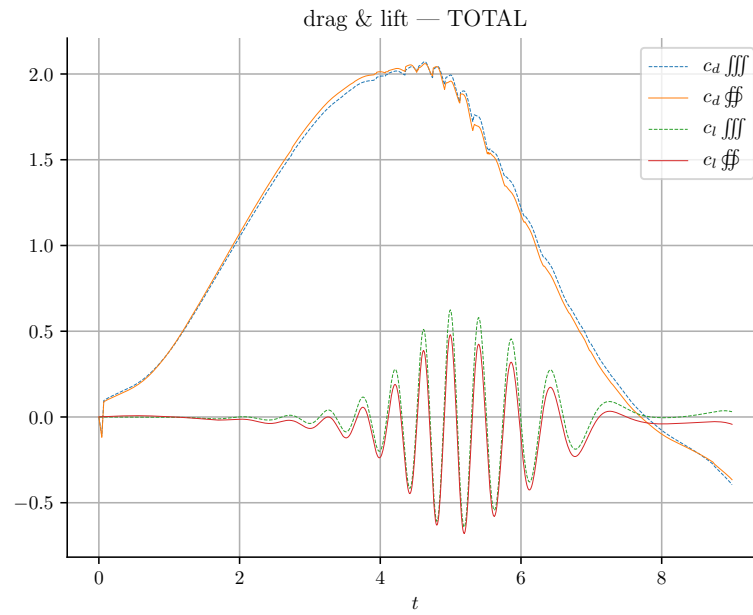


Figure 3.56 – Drag and lift coefficients of the fluid forces exerted on the fixed cylinder.

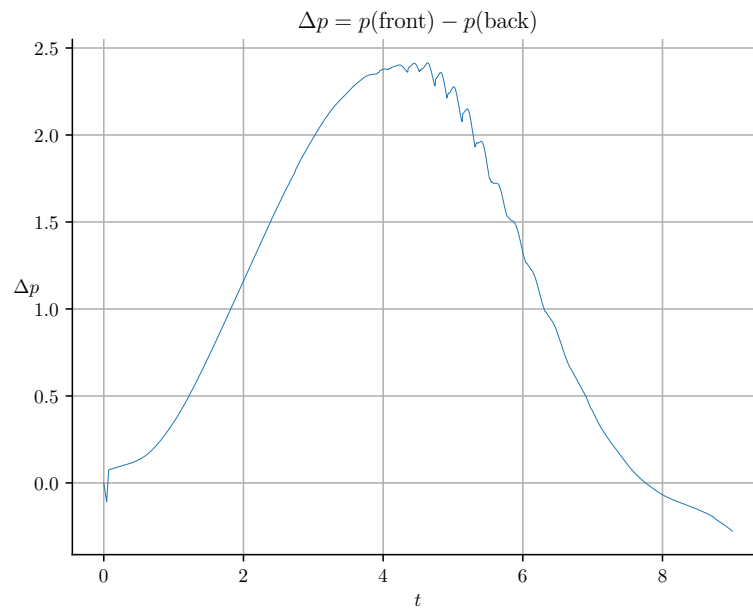


Figure 3.57 – Pressure difference between the stagnation point and the cylinder tail (recirculation zone).

Chapter 4

Application examples

Ce chapitre liste les exemples applicatifs obtenus à partir de la maquette finale à la suite des validations 0D, 1D et 2D obtenues dans la partie précédente. Nous exposons des résultats diphasiques et couplés fluide-structure. Ces exemples sont pour la plupart observables dans la nature. Nous commençons la sous-section diphasique par le cas de la goutte ou bulle oscillante. Cet exemple ne peut être que reproduit de manière numérique ou bien en conditions d'absence de gravité. Cet exemple expose également la problématique des courants parasites résiduels observables avec ce type de méthode. L'instabilité de Rayleigh-Taylor, consécutive à la superposition d'un fluide lourd au dessus d'un fluide plus léger permet de se rendre compte des effets d'échelle de simulation. La rupture de barrage, cas concret dans lequel l'écoulement rentre en régime torrentiel avec impact brutal sur un plan vertical, expose les limites de la méthode utilisée: manque de l'effet de compressibilité pour un gaz mais également limite d'exploitation pour des résultats sous-résolus. L'instabilité de Kelvin-Helmholtz induite par le cisaillement de l'interface par des vitesses opposées rend également compte des effets d'échelle pour les schémas anti-diffusifs. Le cas test de la bulle nous permet de comparer les résultats numériques à un benchmark de plusieurs codes numériques exposé dans la littérature. Dans ce cas, des résultats non plus uniquement qualifiables mais également quantifiables sont exposés. Dans la sous-section fluide-structure, la vibration d'un cylindre sous hypothèse de petits mouvements en eau stagnante permettra d'obtenir les coefficients de masse et d'amortissement ajoutés en monophasique, qui sont à la base d'un travail sur le couplage fluide diphasique-structure. Nous nous intéressons ensuite à la séparation d'une goutte sur un barreau, afin d'observer le comportement numérique du code en couplage, notamment à cause de la présence de points triples (gaz, liquide et solide) qui posent des questions d'ordre physique et numérique. Enfin, nous terminons par un exemple simple d'écoulement diphasique en faisceau de tubes, avec un tube central flexible et plusieurs inclusions de taille différente qui pourront coalescer ou bien se séparer (phénomène de rupture, ou break-up). Cette sous-section a pour objectif d'évaluer de manière qualitative les méthodes: une validation fine à proprement parler représente donc des perspectives d'évolution pour des travaux futurs.

4.1 Two-phase cases

In this section, we expose physical test cases using the multiphase method developed in the section 3.3: we will use the regularization method developed in section 3.1.3.3, the pressure solver tested in section 3.3.4 and the anti-diffusive method of section 3.3.1.2 for conservative density transport.

4.1.1 Droplet oscillation

We now consider a test case for the multiphase problem with capillary interfacial terms. This problem should validate the implementation of the method described in section 3.1.3.1 and section 3.2.3. According to [197–199], we can link the period of oscillation of a deformed bubble with an ellipsoidal initial shape to the surface tension coefficient $T = 2\pi\sqrt{\frac{r_{eq}^3\rho}{6\sigma}}$, where r_{eq} is the equilibrium radius of the drop and ρ is the density of the fluid in the droplet.

The n^{th} mode of oscillation of a inviscid droplet/bubble in three-dimensions can be written as [200–202]:

$$\omega^n := 2\pi f^n = \sqrt{\frac{n(n+1)(n-1)(n+2)\sigma}{((n+1)\rho_2 + n\rho_1)r_{eq}^3}}, \quad (4.1)$$

where the $n = 0$ corresponds to a volume pulsation, $n = 1$ to a translational oscillation, and $n = 2$ to a fixed volume periodic deformation.

Bi-dimensional

In 2D, the work of [203] extends the linear theory of oscillations of cylindrical jets in the transverse plane [199] to the contribution of an external fluid surrounding the inclusion. When the interface undergoes a perturbation in the form of $r(\mathbf{x}) = r_{eq} + \epsilon \cos(n\theta)$, the bi-dimensional period of oscillation is written:

$$\omega^n = \sqrt{\frac{(n^3 - n)\sigma}{(\rho_1 + \rho_2)r_{eq}^3}}$$

ρ_1 [kg·m ⁻³]	ρ_2 [kg·m ⁻³]	μ_1 [kg·m ⁻¹ ·s ⁻¹]	μ_2 [kg·m ⁻¹ ·s ⁻¹]	\mathbf{g} [m·s ⁻²]			
1.	20.	$5 \cdot 10^{-4}$	$1 \cdot 10^{-2}$	0.			
		r_{eq} [m]	σ [kg·s ⁻²]	no. cells	domain [m]	$BC(s)$	lim
		.2	$5e - 3$	128x128	1.x1.	$P \times P$	NB

Table 4.1 – Numerical parameters used for the droplet oscillation test case.

We use the case described in [204] over the one of [205] which is computationally expensive, even in 2D. For the set of parameters in table 4.1, we should recover a theoretical droplet oscillation period of $T^{n=2} = 14.87$ [s], or equivalently $f^{n=2} = .067$ [Hz]. The

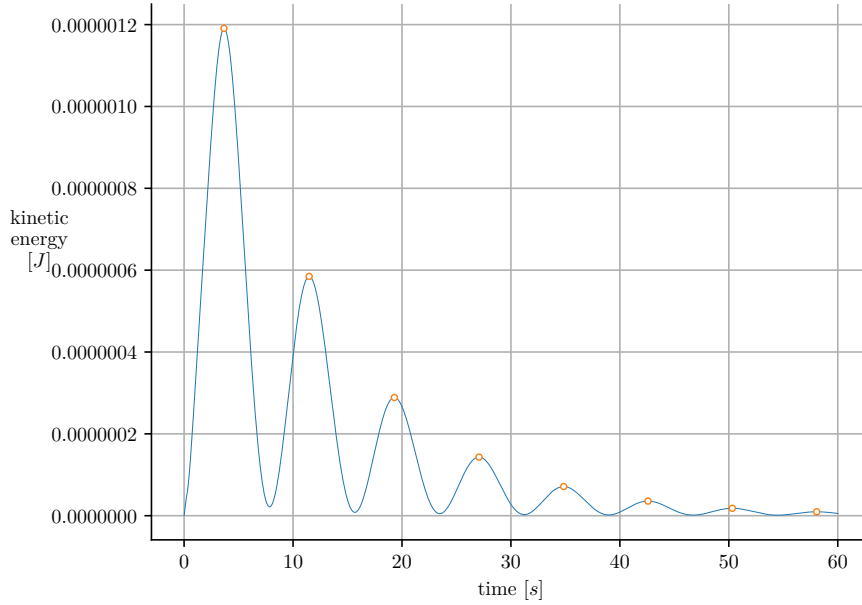


Figure 4.1 – Bi-dimensional droplet oscillation (total kinetic energy).

oscillation frequency was post-treated using the kinetic energy of fig. 4.1, and is half of the kinetic energy frequency. Alternate indicators are used in conjunction of the kinetic energy: either the quadratic moment $\langle x^2 \rangle = \int x^2 dx dz$, the accumulation of density ρ along a central probe or the amplitude of the tip of the droplet to measure the ellipse major/minor axes alternation. Depending on the stiffness of the case (viscosity ratio, limiter used, ...) some numerical noise is sometimes observed for an indicator.

The grid convergence study of fig. 4.2 shows the error relative to the theoretical period of 14.87 [s]. The frequency is .35 [%] longer on a 32^2 grid, .28 [%] with 64^2 cells and .25 [%] for the finest 128^2 grid. For the same case, in [204], the errors are respectively 9.3 [%], 3.4 [%] and 1.9 [%].

Multiple factor can influence the oscillation results such as:

- capillary method used to model surface tension effect (discussed in section 4.1.2);
- truncature error of the different operators (order of the discretization method);
- boundary conditions retro-action (recirculation);
- post-treatment method.

The fig. 4.3 shows the total mass evolution on the domain. The maximal mass fluctuation is overall about $3 \cdot 10^{-3}$ [%], using the most accurate limiter NB .

We show a bi-dimensional slice of the density of the ellipse after 100 steps of the simulation on fig. 4.4, before extracting the two corresponding one-dimensional curves in fig. 4.5.

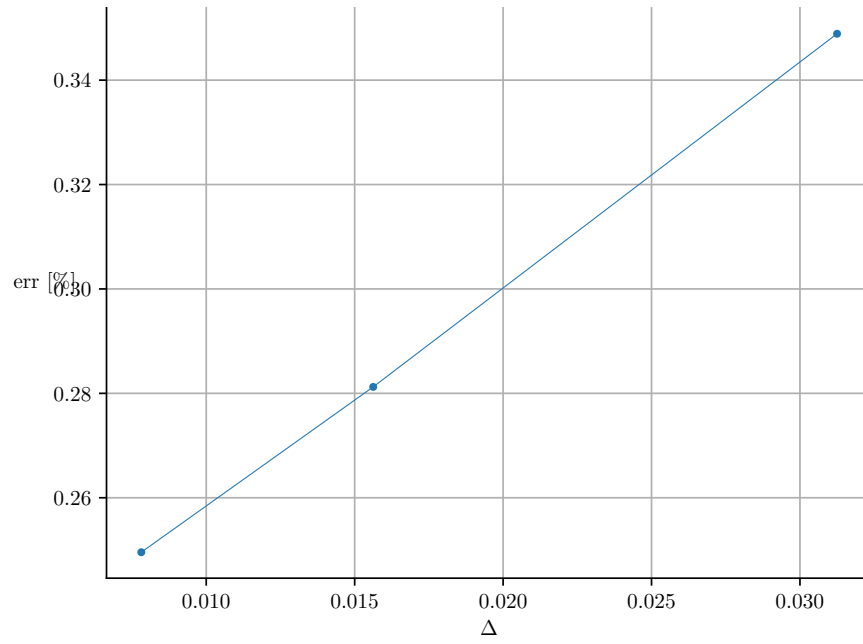


Figure 4.2 – Grid convergence: relative error of the oscillation frequency.

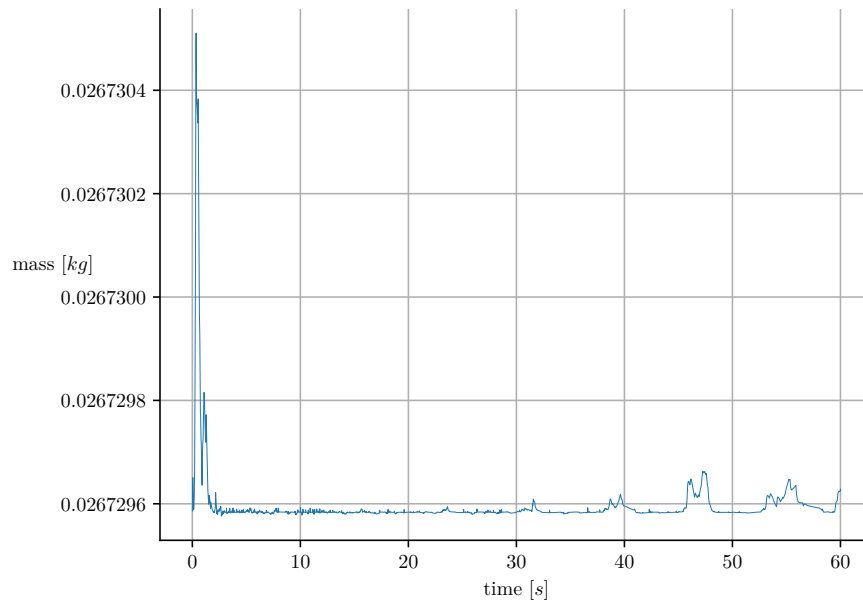


Figure 4.3 – Total mass evolution.

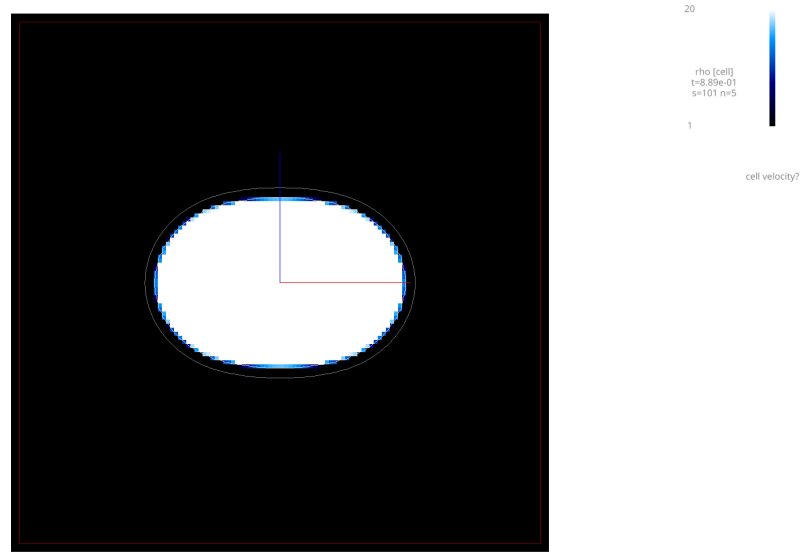


Figure 4.4 – Convected density field (2D gravity free oscillation case) - Contour before and after regularization.

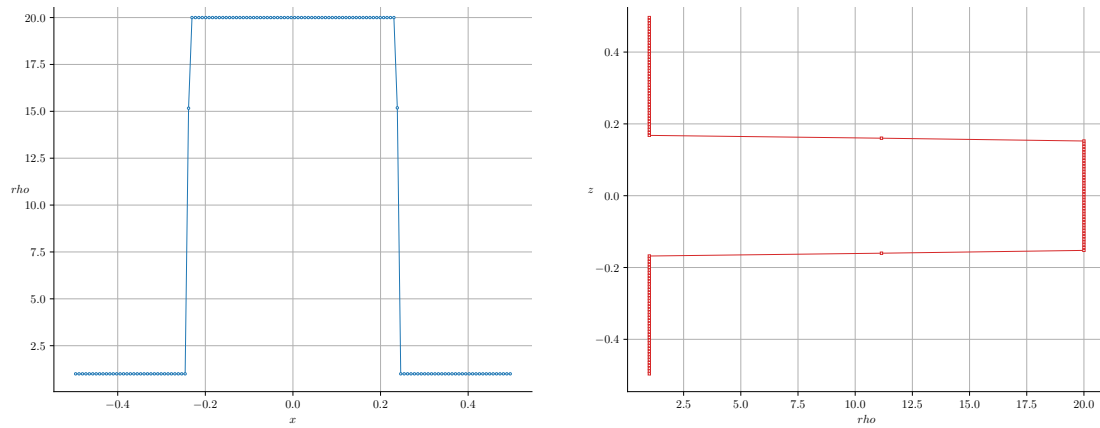


Figure 4.5 – Density (x and z directions).

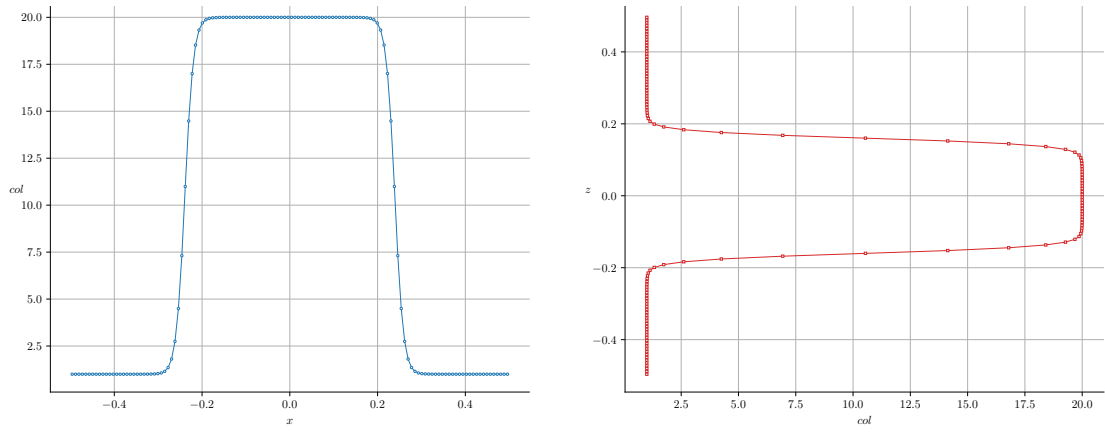


Figure 4.6 – Regularized density (colour function) for normals computation, using a standard deviation $\sigma = 3$. (x and z directions).

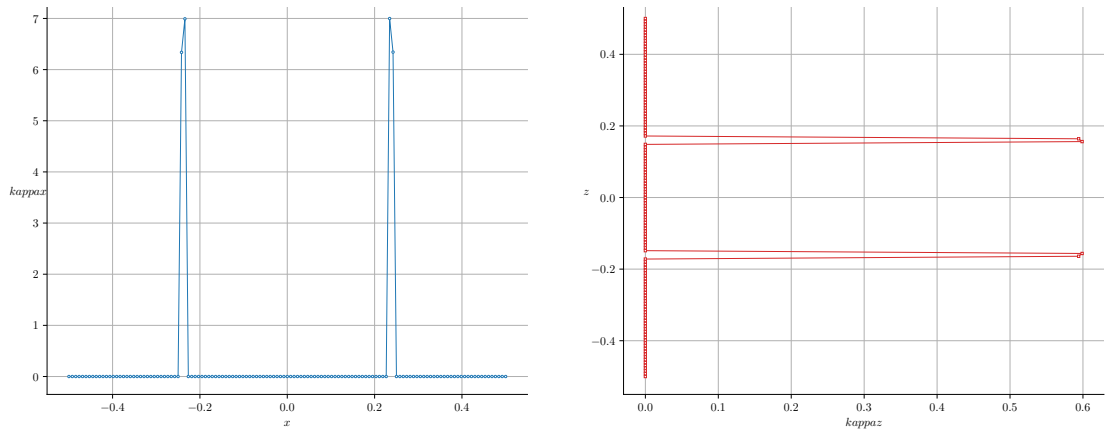


Figure 4.7 – Curvature (x and z directions).

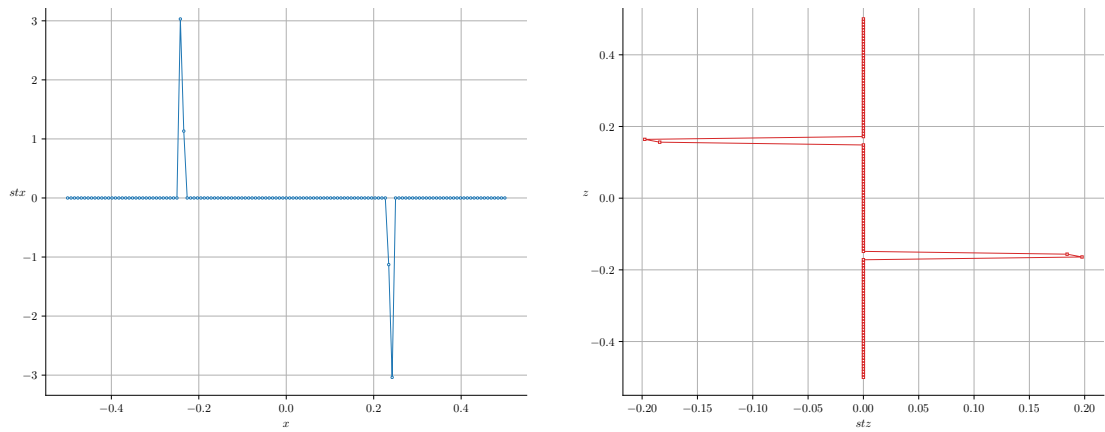


Figure 4.8 – Surface tension (x and z directions).

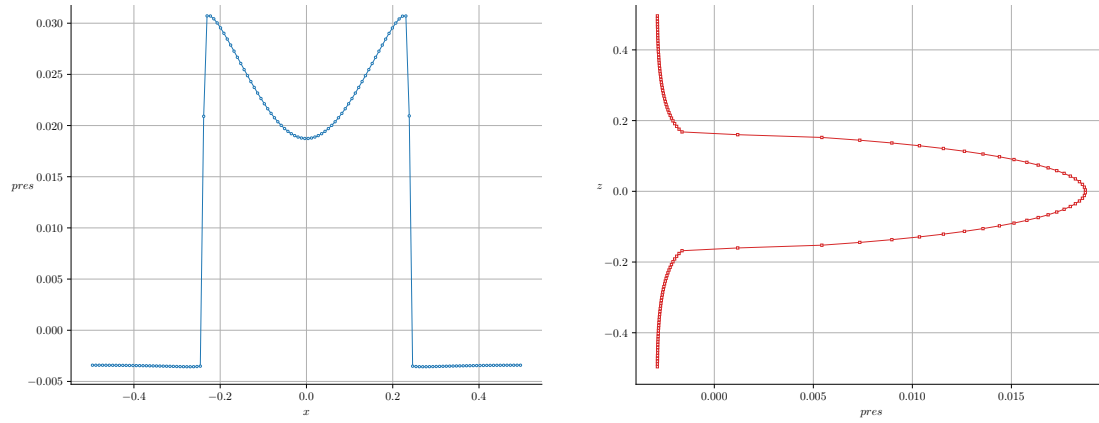


Figure 4.9 – Pressure (x and z directions).

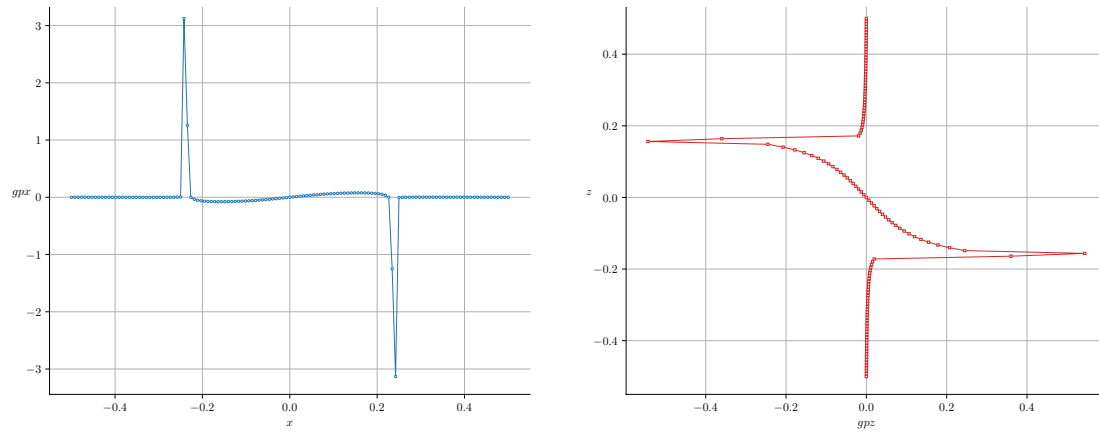


Figure 4.10 – Pressure gradient (x and z directions).

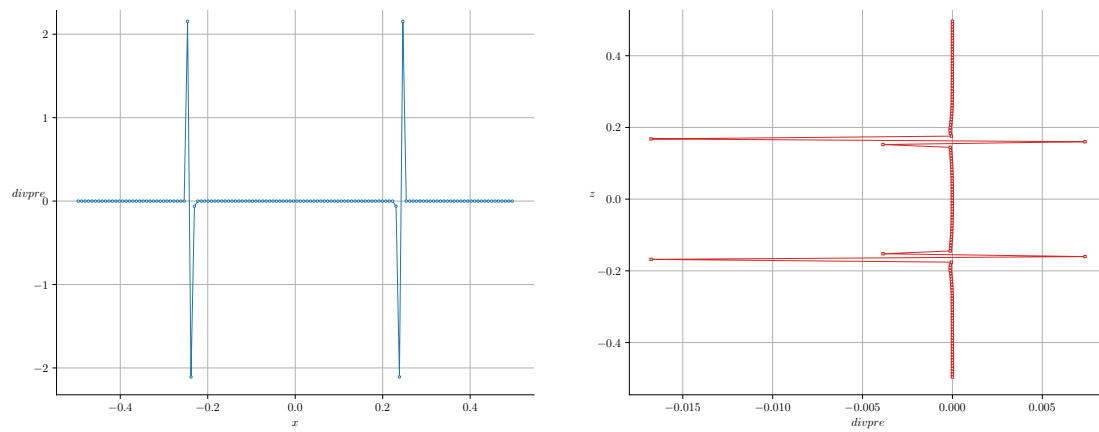


Figure 4.11 – Fluid divergence pre incompressible solver (x and z directions).

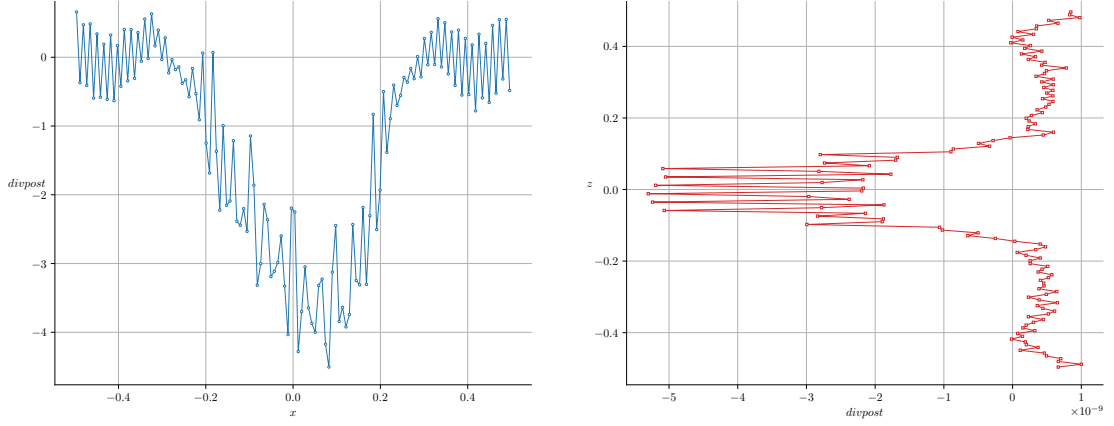


Figure 4.12 – Fluid divergence post incompressible solver (x and z directions).

On fig. 4.5, the density has been convected for numerous timesteps: the numerical diffusion is limited to 3 stencil points. This was expected and observed in section 3.3.5.4 with the use of non-linear high-order schemes for the conservative transport of density.

The fig. 4.7 shows the curvature computed with the regularized density $\bar{\rho}$ used as a colour function. The spread around the interface is controlled by the standard deviation of the filter described in section 3.1.3.4 and a threshold is applied around the interface to avoid spurious terms contribution in the bulk of the domain in the primary phase or inside the bubble/droplet in the secondary phase. The curvature is increasing towards the inclusion centre as expected.

The fig. 4.8 shows the interfacial source term modelling capillary effect. The effect on the fluid is a velocity contribution consecutive to momentum increment at the interface (density transition). The source term is localized as an interface pulse with no apparent interface spread thanks to the anti-diffusive scheme. Later, in the simulation, the spread zone might widen to two or three cells. The interfacial source term has been computed using the regularized density field plotted on fig. 4.6 and the $\mathcal{O}(n)$ filter described in section 3.1.3.3. We note the regularity of fig. 4.6 versus the stiff density seen on fig. 4.5. The regularized solution allows correct computation of unit normals \mathbf{n} to the surface:

$$\mathbf{n} = \frac{\nabla \bar{\rho}}{\|\nabla \bar{\rho}\|}.$$

A pulse (Dirac mass) is applied after curvature computations in order to spread the source term around the interface. Since we wish to insert stiff source terms at the interface to limit the numerical diffusive layer, we have found that using a smooth pulse could affect the cohesion of the inclusion over time in a negative way, by magnifying the negative effect of parasitic currents.

The fig. 4.10 shows the term homogeneous to a pressure gradient used to correct the predicted velocity field. The ∇p term is non zero in the diffuse interface. The pressure

value in fig. 4.9 is implicit and non-physical, even if homogeneous to a pressure unit: the numerical value taken is conditioned by the incompressibility constraint of the fluid. The divergence of the fluid in the domain after velocity prediction and insertion of source terms is seen on fig. 4.11. The divergence is driven towards a chosen criterion (usually greater than the machine epsilon: $\epsilon \simeq 1 \cdot 10^{-16}$ for double precision floating point values) by the implicit pressure step. With this non multi-grid solver (*BiCGSTAB*), the exact interface position cannot be inferred from the residual pattern as seen on fig. 4.12.

Three-dimensional

A 3D oscillation computation was performed on a 32^3 grid. We use the case described in [206], with a unit radius droplet immersed in a 4.0 [m] wide cubic box. The parameters are stiff for the method using implicit interface, with a density ratio of 100, and a viscosity ratio of 10000. Nonetheless, we recover a period of $T = 34.06$ [s] which is 8.9 [%] longer than the analytical period using eq. (4.1).

The theoretical viscous decay time [200, 207, 208] is:

$$\tau = \frac{r_{eq}^2}{\nu_2(n-1)(2n+1)}.$$

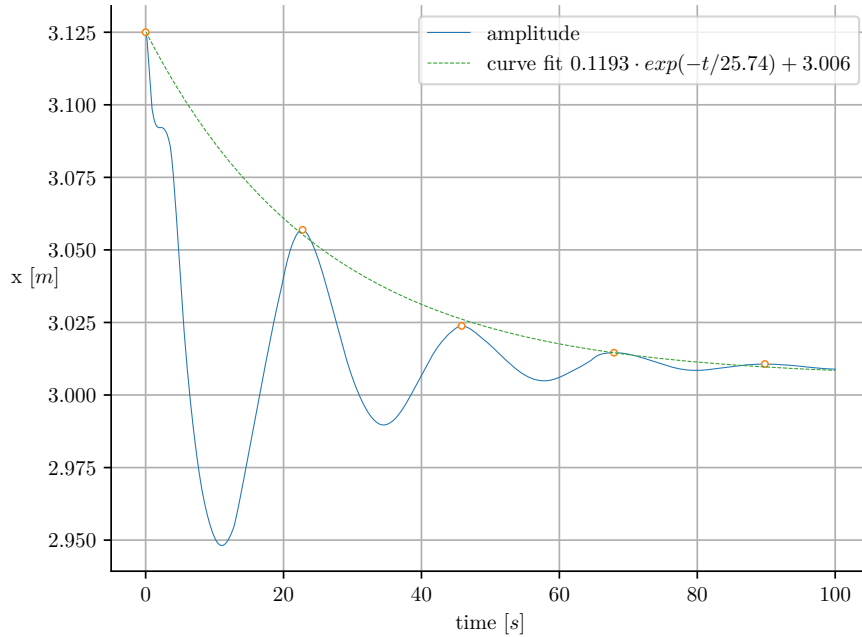


Figure 4.13 – Viscous decay fit of the 3D droplet oscillation amplitude.

On fig. 4.13, an exponential law is fitted onto the droplet amplitude. The amplitude is slightly overestimated with 3.006 instead of 3. for the static amplitude. For the theoretical value of $\tau = 20$, we obtain a fitted value of $\tau = 25.74$.

4.1.2 Parasitic currents

A drawback of the numerical method used where the interfacial terms are spread onto a thin transition layer is the production of parasitic currents. These spurious velocities should vanish under equilibrium: the pressure gradient should balance capillary forces exactly. However, it is mathematically challenging to compensate an explicit term with the result of an iterative implicit process up to numerical precision or even less. The origin of these spurious currents is believed to be numerical curvature gradients, which are the source of local production of vorticity [209].

At static equilibrium, the momentum equation eq. (3.6) reduces to:

$$\nabla p = \sigma \kappa \delta \mathbf{n} . \quad (4.2)$$

Applying the curl $\nabla \times (\cdot)$ operator to the equation eq. (4.2) leads to:

$$0 = \sigma \kappa \delta \nabla \times \mathbf{n} + \nabla(\sigma \kappa \delta) \times \mathbf{n} = \nabla(\kappa \delta) \times \mathbf{n} , \quad (4.3)$$

assuming a constant in space surface tension coefficient (which has been the assumption throughout this work), and the validity of the gradient of the Dirac functional under a set of test functions (smooth functions with compact support).

If the eq. (4.3) is not satisfied, then the static equilibrium condition is violated, hence the production of eddy currents - sources of vorticity. We conclude that the numerical challenge of parasitic currents free computations lies in computing a gradient free curvature. With slight parasitic gradients, the interface becomes unstable and in our application leads to excessive numerical diffusion induced by strong local velocity gradients and later loss of the interface with exponential growth of a wavy pattern along the surface.

In our early computations using a convolution kernel as exposed in section 3.1.3.3, we observed strong parasitic currents patterns, leading eventually to rupture and desegregation of the interface. The kinetic energy on fig. 4.14 shows that a non-zero steady average value of the kinetic energy during oscillation tends to indicate a spurious residual mean value at steady state. In three dimensions, parasitic currents do appear along the domain diagonals with a periodic pattern: this tends to indicate a splitting effect (which remains to be studied). We mention that the results of fig. 4.14 were obtained using a total different set of parameters than the test case resulting in fig. 4.13, and cannot be compared as is.

Experiments with the case described in section 4.1.1, or with a static droplet clearly show a non-zero residual velocity. Other alternatives or parasitic current reduction attempts worth mentioning can be found in [41, 170, 210–213].

An approach for the reduction of parasitic currents consists of shifting the interfacial source term towards region of high density. This allows reduction of the effect of fluid acceleration induced by capillary forces by applying the acceleration term preferably to

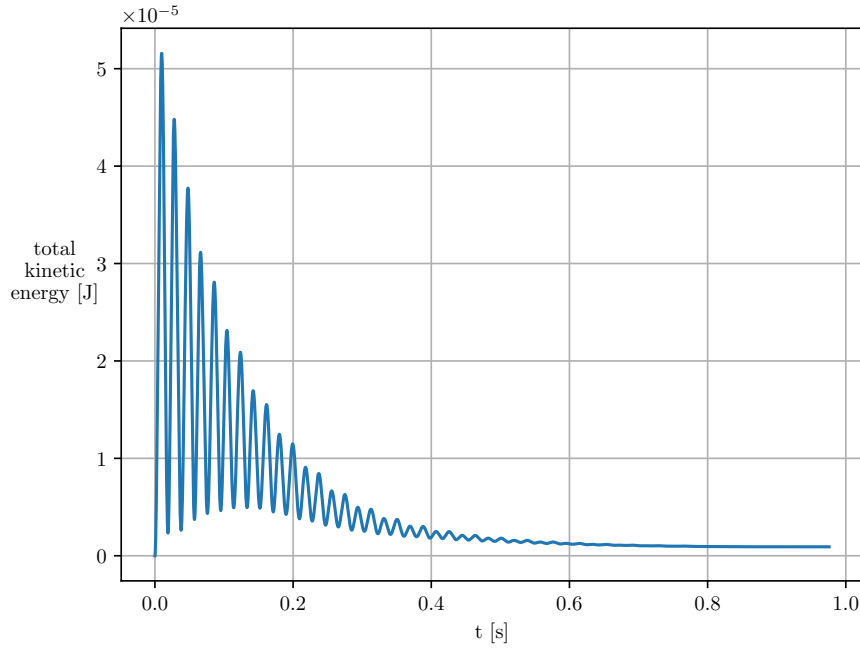


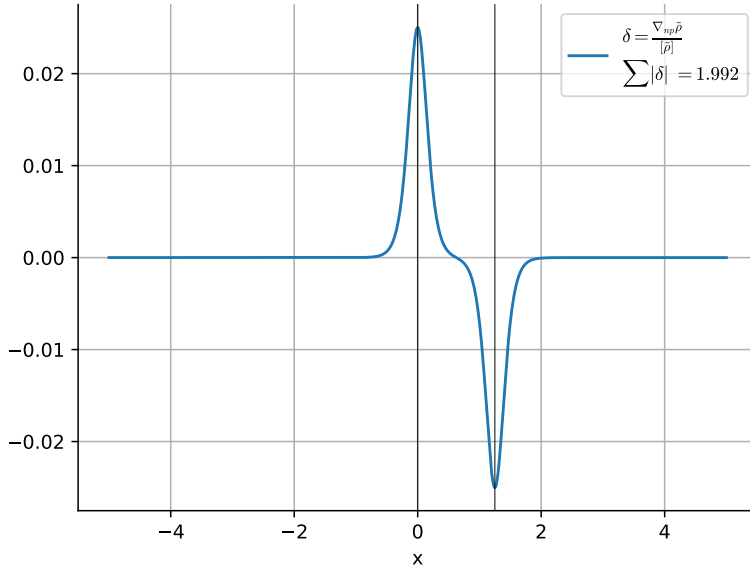
Figure 4.14 – 3D oscillation case, convolution based regularization.

denser phase, while keeping the integral value un-modified across the interface. The δ_f facet impulse is modified according to:

$$\delta_f \ast = 2. \left(\frac{\rho_f - \rho_{min}}{\rho_{max} - \rho_{min}} \right)^\lambda, \quad (4.4)$$

where $\lambda > 0$ is an arbitrary factor. The factor 2. in eq. (4.4) accounts for the truncation of the external part in the region of weak density, so that the integral value of δ_f over the interface remains approximatively equal to unity as seen on fig. 4.15. For strong density ratio, failing to shift the source term results in un-physical accumulation or depletion of mass across the domain because of the incorrect accuracy of the continuity equation with erroneous fluxes. However, this approach leads to wrong total dynamics and was thus abandoned. Another approach for the total conservation of mass because of small variations is to shift the average point of mass across the whole domain periodically. This is the approach retained to avoid non-negligible variation of total mass.

In order to assess the effect of parasitic currents, we compare our method to front tracking computations found in [206, 214]. The simulation are exposed in table 4.2, with strict convergence tolerance of the pressure solver set to 10^{-15} in the ℓ^∞ norm. The density is initialized on a regular cartesian grid. A transient regime thus appears for a limited amount of time in order for the phases to equilibrate. We use the least compressive, yet the most accurate (second order) limiter of [152]. After a short transient, the pressure forces balances with surface tension and the total kinetic energy decreases in the do-



(a) Source term impulse, centred on the interface.

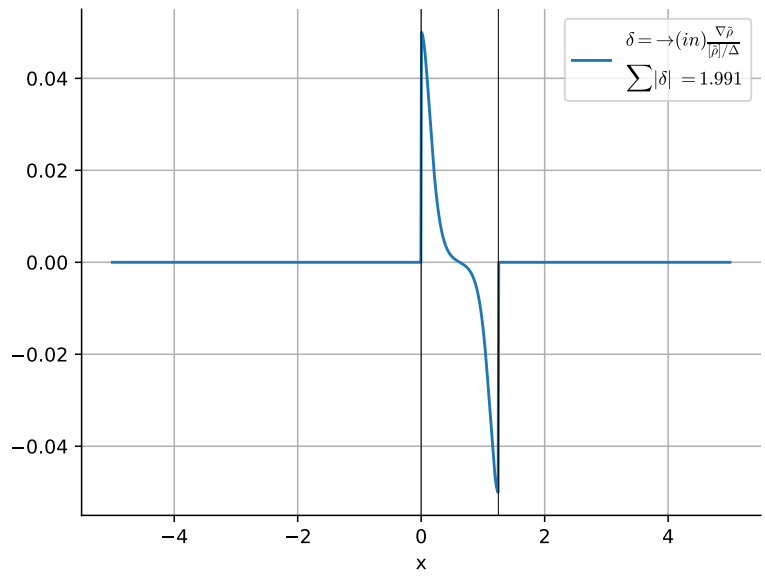
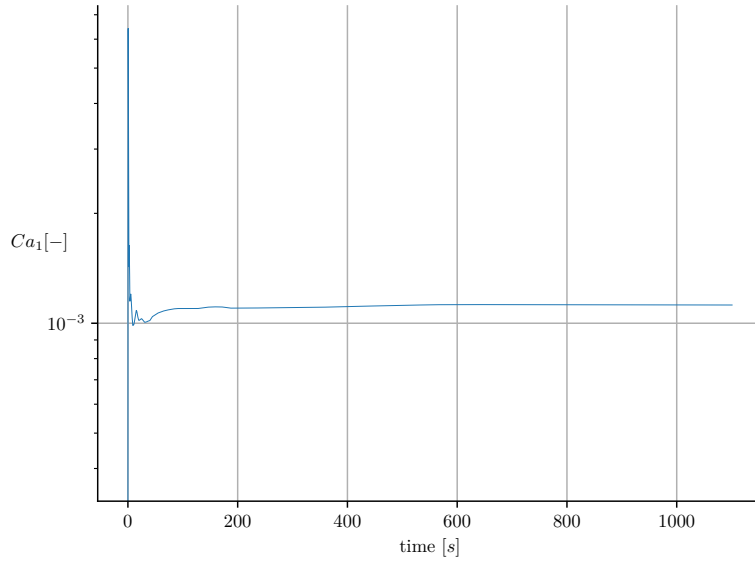
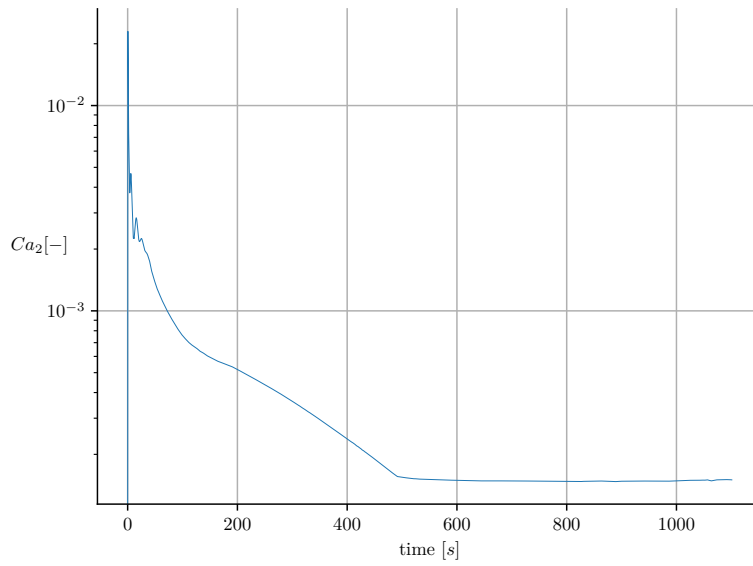
(b) Interface impulse, shifted towards the inclusion with $\rho_2 > \rho_1$.

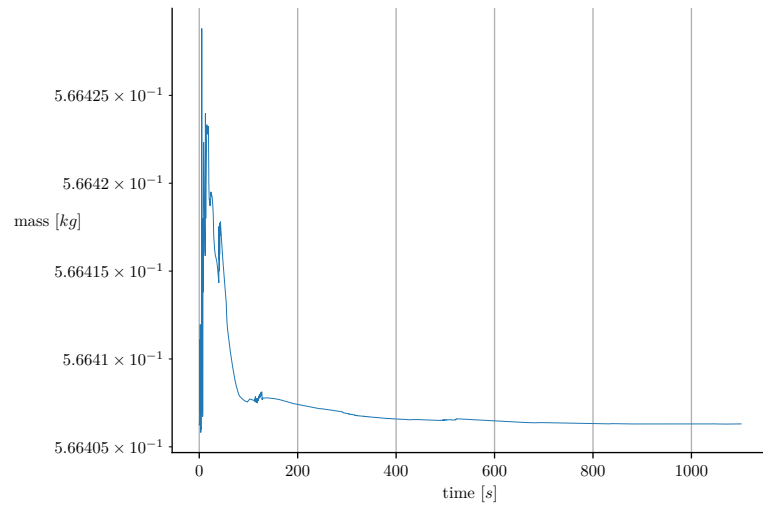
Figure 4.15 – Capillary source term shift.



(a) Capillary number outside the droplet.



(b) Capillary number inside the droplet.



(c) Total domain mass evolution.

Figure 4.16 – Capillary numbers $Ca = \mathbf{u}_{max}\mu / \sigma$ and total mass.

main. We compute the capillary number $Ca = \frac{U\mu}{\sigma}$ ⁱ by measuring the maximal velocity magnitude for each phase. The method shows capillary numbers of $Ca_1 = \mathcal{O}(10^{-3})$ outside the inclusion on fig. 4.16a and $Ca_2 = \mathcal{O}(10^{-4})$ inside the inclusion on fig. 4.16b. Total mass conservation is ensured within $5 \cdot 10^{-3} [\%]$, as seen on fig. 4.16c. Our method thus behaves better than methods such as [75] with $Ca_2 = \mathcal{O}(10^{-2})$ and is equivalent to *FT* methods which expose $Ca_2 = \mathcal{O}(10^{-4})$ parasite currents behaviour. However, other specialized methods can get as far as $\mathcal{O}(10^{-20})$ [210, 215] (using an energy conservative method instead of a momentum conservation one); or $\mathcal{O}(10^{-14})$ under *LSM* approaches [209]. We compare the capillary number for the inclusion: the capillary-viscous length R_ν is computed using the droplet parameters [214].

ρ_1 [kg·m ⁻³]	ρ_2 [kg·m ⁻³]	μ_1 [kg·m ⁻¹ ·s ⁻¹]	μ_2 [kg·m ⁻¹ ·s ⁻¹]	crit(ℓ^∞)	σ [kg·s ⁻²]		
10.	50.	$2 \cdot 10^{-2}$	$1 \cdot 10^{-1}$	$1 \cdot 10^{-15}$.1		
		La [—]	r_{eq} [m]	no. cells	domain [m]	$BC(s)$	lim
		250	.25	33x33	1.x1.	P_xP_xP	NB

Table 4.2 – Numerical parameters used for the parasitic currents test case.

i. Capillary number: measure of viscous forces versus surface tension

4.1.3 Rayleigh-Taylor instability

We consider a physical hydrodynamic test case called the *RTI*ⁱⁱ. This phenomenon occurs when a dense fluid is placed over a less-dense fluid in a gravitational field. An initial perturbation of the interface (required to numerically trigger the instability) results in the growth of bubbles and dense, falling tear-drops of fluid [216–218]. This example is non-trivial to reproduce through experiments since the initial condition are unstable, and influence of the start-up mechanism over the interface is not negligible. Numerous analysis is available in the following references, for the set-up of the complex test case [219–222].

According to [131, 223], the exponential growth rate of instability $\exp(nt)$ is:

$$n^2 = k|\mathbf{g}| \left(A - \frac{k^2 \sigma}{|\mathbf{g}|(\rho_1 + \rho_2)} \right), \quad (4.5)$$

where $A = \frac{\rho_2 - \rho_1}{\rho_2 + \rho_1}$ is the Atwood number and ρ_2 is assumed to be the density of the heavier fluid.

In order to validate the test case, a theoretical stability parameter is derived from a critical surface tension value σ_0 , for which $n_0 = 0$.

The critical surface tension is expressed using eq. (4.5):

$$\sigma_0 = \left(A - \frac{n_0^2}{k \cdot |\mathbf{g}|} \right) \frac{|\mathbf{g}|(\rho_2 + \rho_1)}{k^2}.$$

Departure from this equilibrium induces local shears, and movement of one fluid into another, in a filamentary manner. The shape of the dense fluid evolves in a well known mushroom shape pattern in the surrounding lighter fluid medium as it can be observed on fig. 4.17. We also observe capture of a secondary sub-scale instability at the beginning of the simulation, which are eventually damped over time. The primary instability shows the typical shape of the fluid at the end of the simulation. Numerical convergence with air water mixture and capillary terms is excessively slow using explicit discretization of the diffusion operator at the sub-scale level.

The fine-scale artefacts observed at the beginning of the simulation on fig. 4.17 are likely to originate from the stair-cased interface (regular cartesian mesh). This Moiré like pattern is dependent on initial conditions for density. We also tested a slight perturbation of the gravitational field in the centre of the interface at the begin of the simulation before establishing a uniform field for the rest of the simulation. We first used a smooth varying function of the horizontal coordinate x , with one spatial period in the horizontal direction to trigger the instability. The perturbation initially triggers the instability and allows local small-scale movements of the lighter phase into the denser fluid.

ii. Rayleigh-Taylor Instability

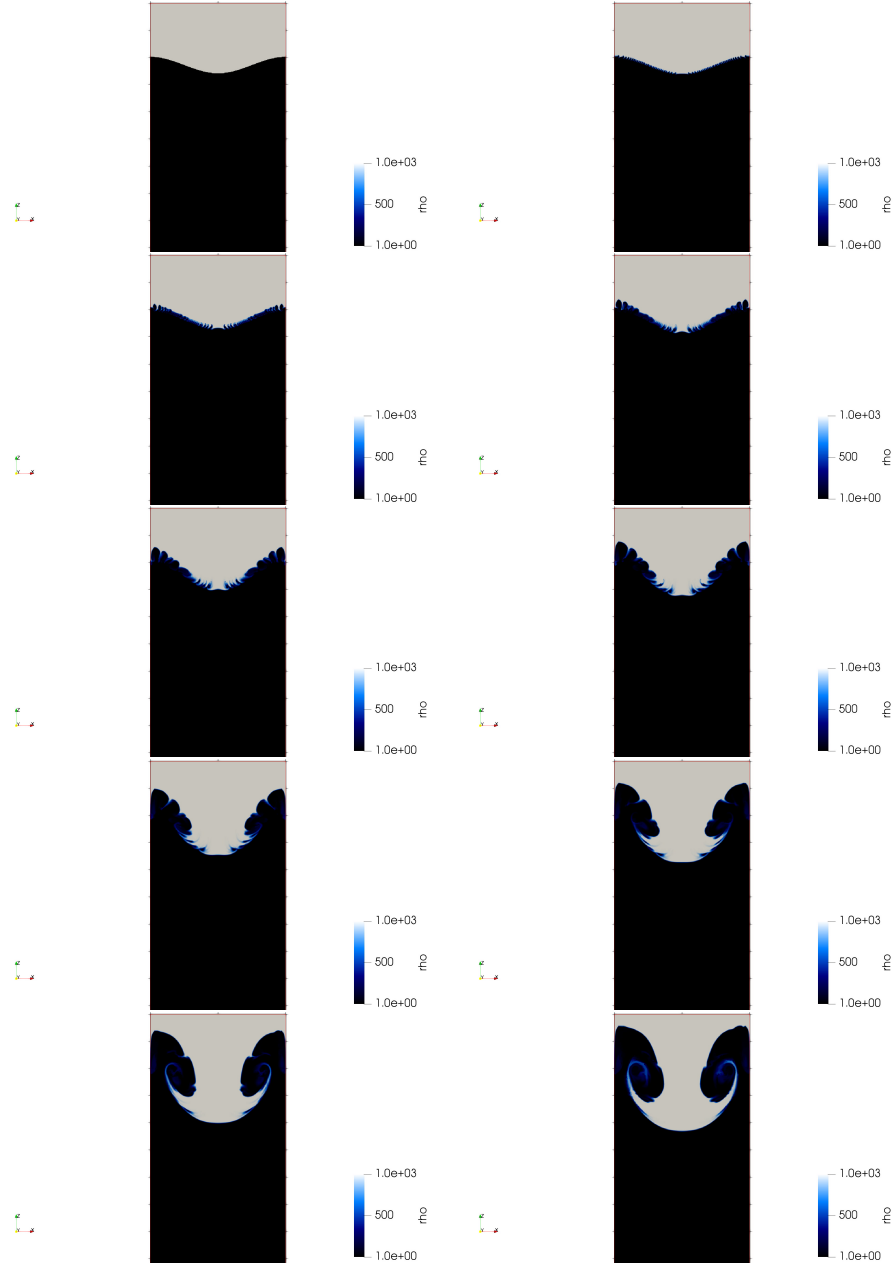


Figure 4.17 – Air - Water: (1 / 1000) density ratio simulation with capillary terms.

ρ_1 [kg·m ⁻³]	ρ_2 [kg·m ⁻³]	μ_1 [kg·m ⁻¹ ·s ⁻¹]	μ_2 [kg·m ⁻¹ ·s ⁻¹]	\boldsymbol{g} [m·s ⁻²]		
10.	100.	.1	$1 \cdot 10^{-2}$	9.8		
σ [kg·s ⁻²]	no. cells	domain [m]	$BC(s)$	lim	no. levels	smoother
$1 \cdot 10^{-3}$	64x64x256	.1x.1x.4	$W^{\text{ix}}xW_xW$	NB	4	$RBGS$

Table 4.3 – Numerical parameters used for the *RTI* case.

We choose a surface tension coefficient, for which the interface growth is stable: $\sigma = k\sigma_0$, with $k < 1$. The simulation parameters are found in parameters in table 4.3, with the conditions close to those of a steam generator with a 1/10 density ratio. The computation was run on a single computing unit under 16 threads (data decomposition) and 14h wall time.

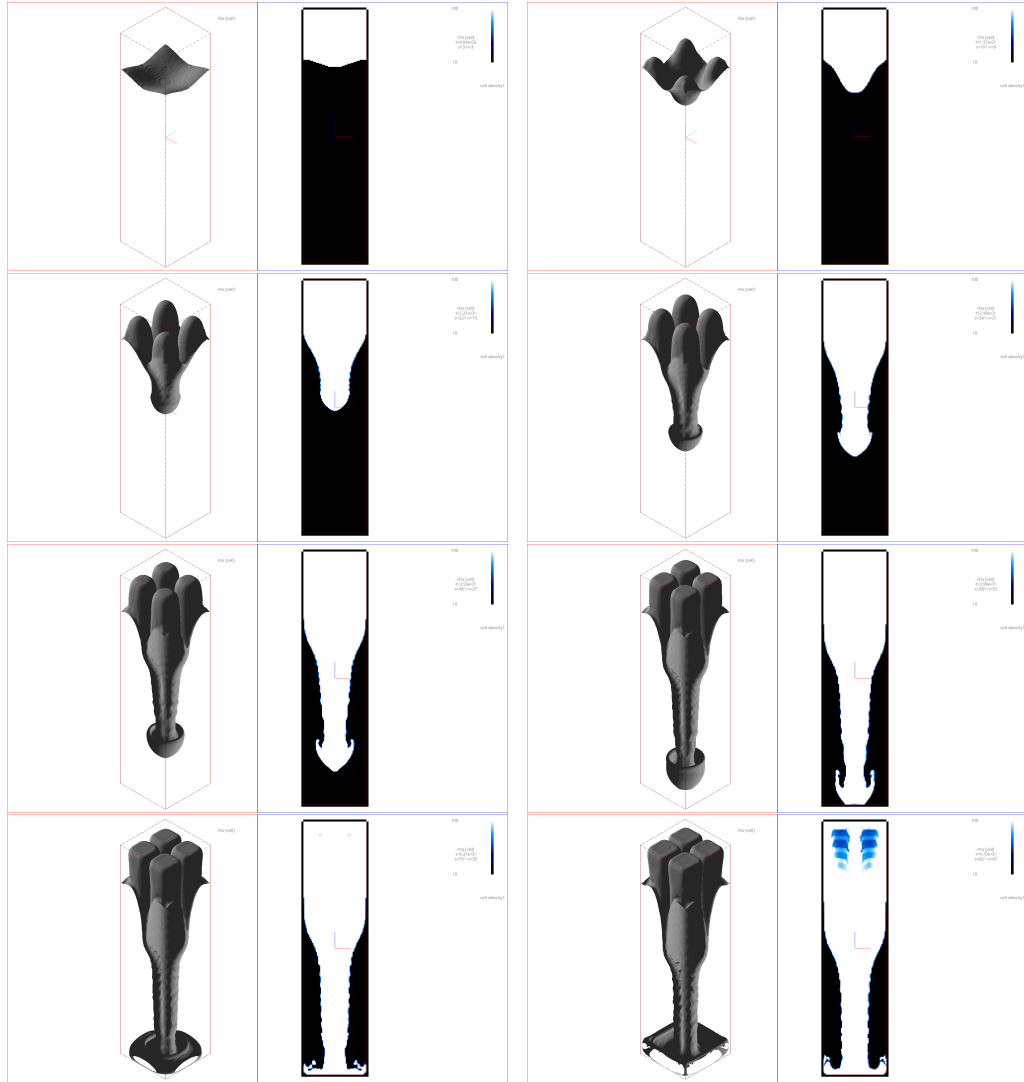
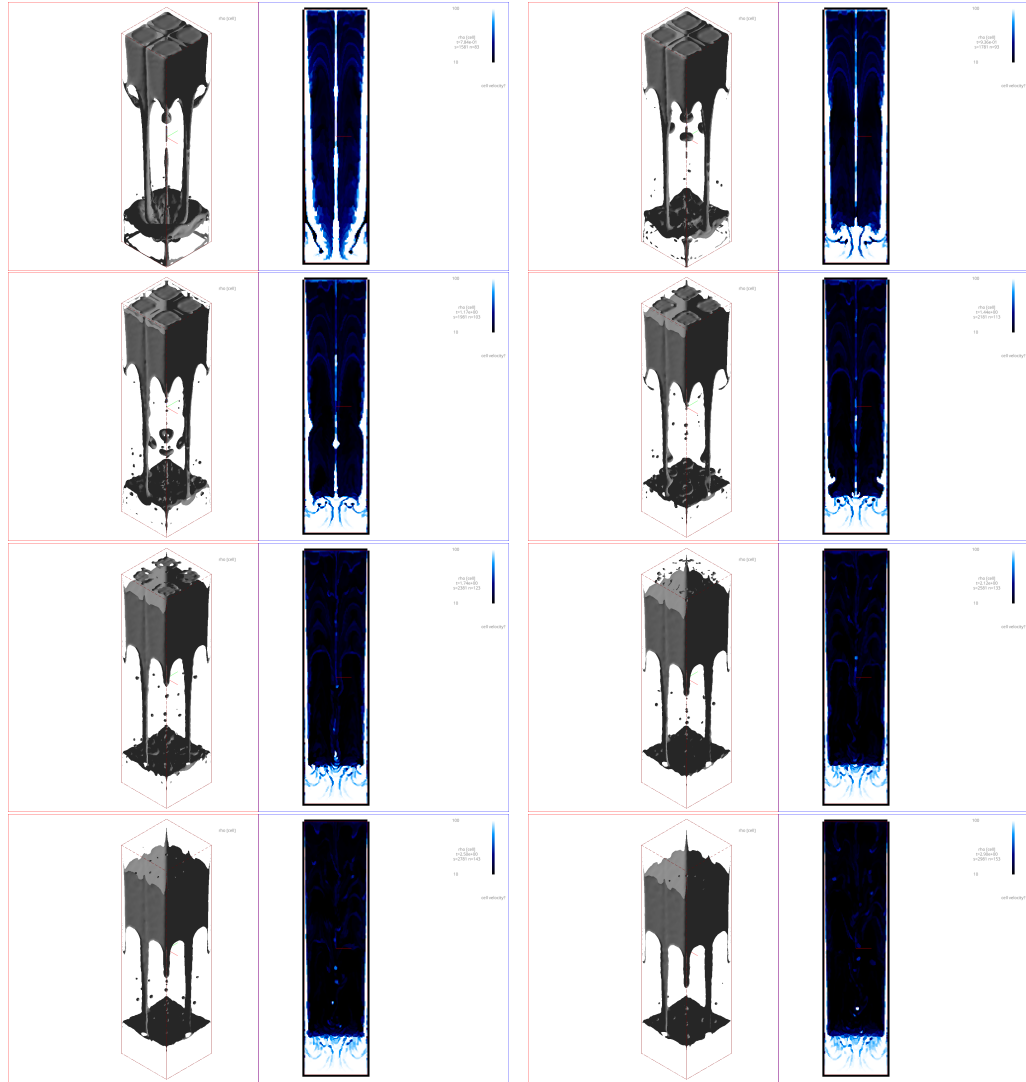


Figure 4.18 – Three dimensional *RTI* - First computation steps.

Figure 4.19 – Last steps of the *RTI* simulation.

The instability is triggered by a cosine spatial perturbation. The three-dimensional computation exposes the transverse behaviour, especially the entrainment of the fluid behind the main column. Local perturbations in the form of waves forming a KHI ⁱⁱⁱ develops on the central fluid column. Wall impact is observed at the bottom of the rectangular computational domain, as seen in the first steps on fig. 4.18, as well as splashing effect on the surrounding walls. Run-off is observed on the walls, where no-slip boundary conditions are applied, together with ceiling detachment and fall of the remaining viscous fluid droplets at the end of the simulation as observed on fig. 4.19. The wall impact is crucial for the preparation of immersed boundary cases where the interfaces will hit tubes in a bundle.

iii. Kelvin-Helmholtz Instability

4.1.4 Dam break of incompressible fluids

Sudden release and impact of a consequent quantity of fluid is of interest for example in designing protection walls against cataclysmic events. The dam break application allows comparison with experimental data for example the height of the wave which impacts the boundaries after release. A block of fluid is released at time $t = 0$ and then flows on top of a dry bed. This method allows comparison with the shallow water equations results (integration of the $NSE(s)$ along the depth) [224]. Our objective is to compare data obtained in [225] with numerical results and numerical experiment similar to the references [127, 226].

ρ_1 [kg·m ⁻³]	ρ_2 [kg·m ⁻³]	μ_1 [kg·m ⁻¹ ·s ⁻¹]	μ_2 [kg·m ⁻¹ ·s ⁻¹]	g [m·s ⁻²]
5.	10.	$1 \cdot 10^{-3}$	$1 \cdot 10^{-2}$	10.

σ [kg·s ⁻²]	no. cells	domain [m]	$BC(s)$	lim
.5	512x256	2.x1.	$S^{\text{iii}}_{\text{xW}}$	NB

Table 4.4 – Numerical parameters used for the dam break test case.

The parameters of the bi-dimensional numerical simulation are listed in table 4.4, with a constrained density ratio. The limit of this method is the compressibility of the continuous phase. The reversed flow induced by fluid movement consecutive to the wall impact, and observed on fig. 4.20, is noticeably missing compressible behaviour, at the end of the simulation. The gaseous phase is convecting a non-physical amount of inertial energy - which shall instead be consumed by a compression of the fluid. Consequently, the gaseous vortex splits the liquid phase in two distinct portions after wall impact.

We also observe excessive numerical diffusion, which questions the choice of the mesh resolution which is likely to be un-adapted for DNS computations. Two-phases structures are under the cell size, hence they cannot be captured and excessive mixture prevents distinction of the interface. The surface tension takes erroneous values (direction and magnitude) on coarse control volumes. The limit of the capillary method used lies in the way coarse DNS is unable to capture the fine-scale behaviour of surface tension.

Alternatively, one can consider an evolution law for the pressure in addition to the evolution law for the velocity:

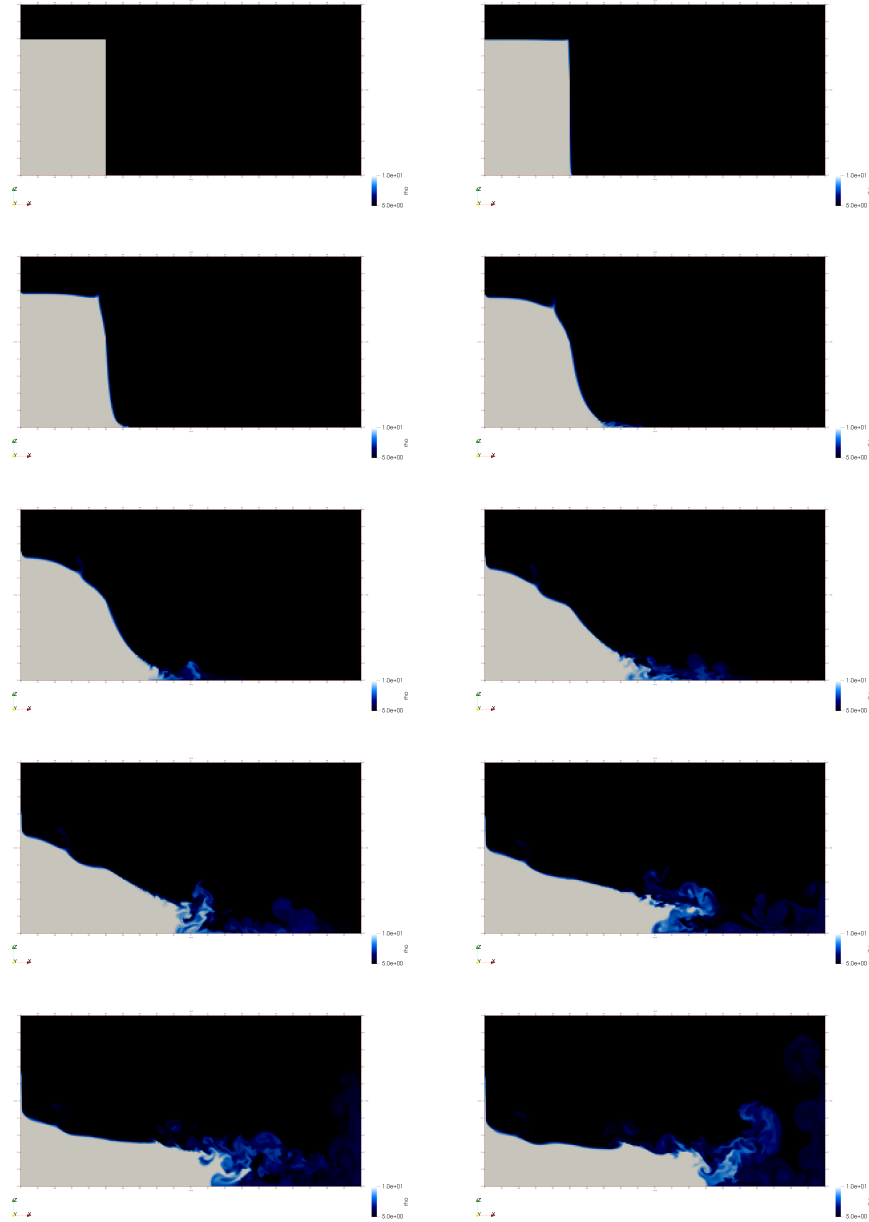
$$p_t + c^2 \nabla \cdot \mathbf{u} = 0.$$

The system is hyperbolic, and c^{iv} brings an additional computational constraint on the timestep.

More advanced methods such as the low Mach number approach ($\mathcal{M} \ll 1$) might perform better since they include the contraction/expansion of the fluid without involving phase change [227–229].

The dilatable approach adds a non-zero contribution to the RHS of eq. (3.20), requiring closure equations for the expression of the source term. On the opposite, taking a

iv. Sound celerity



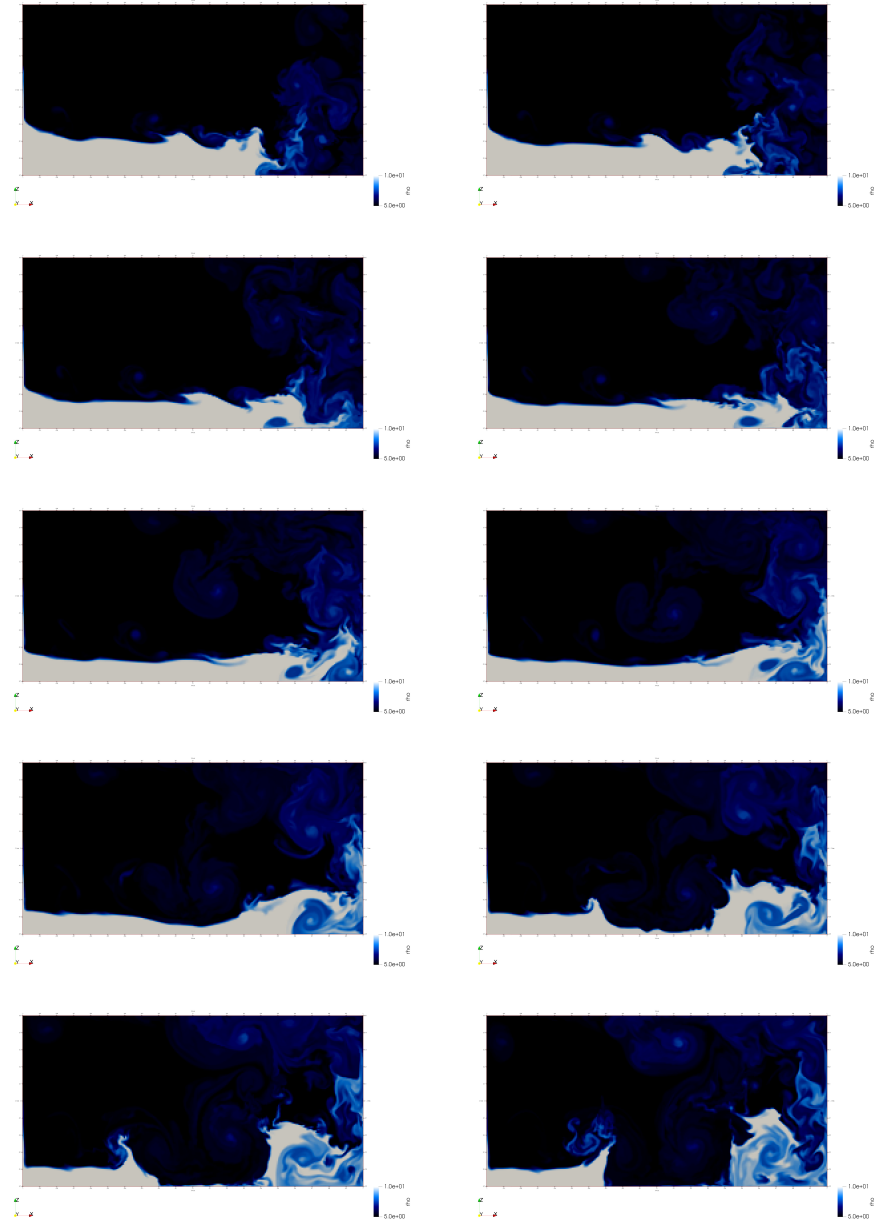


Figure 4.20 – Dam break of incompressible fluids.

compressible approach would allow to compute the shock waves induced by the water hammer [230, 231] effect when the fluid is suddenly forced to stop or changes direction in an abrupt manner. We might also consider the case where the domain is not a closed rectangle with walls, but an open box with free surface boundary conditions at the top. However, this approach posed numerical issues for the imposition of boundary condition for the pressure solver and the velocities.

4.1.5 Kelvin-Helmholtz instability

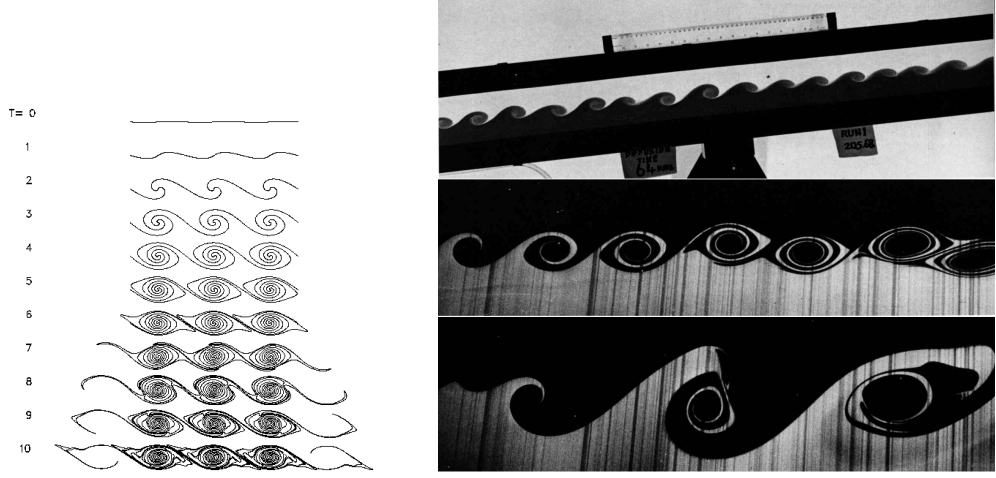


Figure 4.21 – Cat eye patterns observed numerically [232] and experimentally [233].

This test case should reproduce static equilibrium of a droplet or fluids under a shear flow. The flow which will undergo a perturbation of finite wave length will stabilize under surface tension and the presence of different fluids [234]. Although un-necessary for the instability to develop from a theoretical point of view, viscosity and capillary terms were used in the discrete set of equations.

ρ_1 [kg·m ⁻³]	ρ_3 [kg·m ⁻³]	μ_1 [kg·m ⁻¹ ·s ⁻¹]	μ_2 [kg·m ⁻¹ ·s ⁻¹]	σ [kg·s ⁻²]		
998.2	960.	$1.003 \cdot 10^{-3}$	$4.8 \cdot 10^{-2}$	$7.37 \cdot 10^{-2}$		
g [m·s ⁻²]	u_1 [m·s ⁻¹]	u_2 [m·s ⁻¹]	no. nodes	domain [m]	$BC(s)$	lim
0.	-1	3.	513x257	1.x.5	$P_{\text{x}}W$	DL

Table 4.5 – Numerical parameters used for the Kelvin-Helmholtz instability test case.

In order to ease numerical computation, a high viscosity fluid (oil) is chosen, with a density close to the aqueous phase in table 4.5. The slow settling of the lowest boundary layer requires the use of the implicitation step, or the initialization of the velocity field to wall velocities. Interface movement is induced by momentum diffusion through the boundary layer originating from wall boundaries. We imposed opposite wall velocities with different magnitudes, resulting in overall movement of the fluid towards the eastern part of the domain (periodic in the x direction). The capillary forces were inserted at the interface transition, as described in section 3.1.3.3. The density flux evaluated during conservative transported is computed using the most compressive scheme DL of [46].

A theoretical stability criterion is developed, as for the section 4.1.3. If we write $\eta(x, t)$ as seen on fig. 4.22, with the interface deviating from the initial flat interface, we seek solutions in the form $\eta(x, t) = \Re(A_0 e^{i(kx - \omega t)})$, where A_0 is the interface initial disturbance amplitude. It can be shown that the growth rate stability condition of the

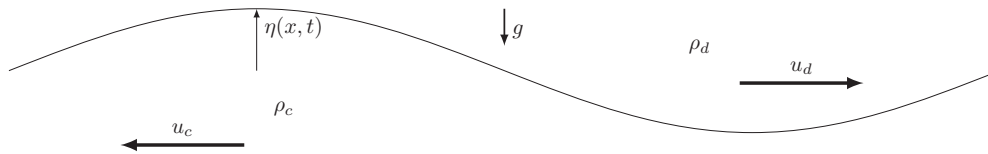


Figure 4.22 – Interface perturbation.

interface a function of k :

$$F(k) = \frac{\rho_1 \rho_2 (U_1 - U_2)^2}{(\rho_1 + \rho_2)^2} + s(k),$$

where $s(k) = \frac{\rho_1 - \rho_2}{\rho_1 + \rho_2} \frac{s}{k} + \frac{\sigma}{\rho_1 + \rho_2} k$ is a function of the wave-number k .

The details are omitted, but if $\rho_1 < \rho_2$, and the initial shear velocities are null, the growth is unstable if $0 < k < \sqrt{(\rho_2 - \rho_1)g / \sigma}$: this is the case exposed in section 4.1.3 [235].

On the contrary, if $\rho_1 > \rho_2$ (lighter fluid on top of the heaviest), and if the phase velocities are different, the system is unstable [131, 236] under the condition:

$$|U_1 - U_2| > (\rho_1 > \rho_2) \sqrt{g / (\rho_1 \rho_2)}.$$

A vortex sheet should develop, induced by the vorticity source at the interface. Vorticity originates from strong interfacial tangential discontinuity due to imposed velocity boundary effects migration from the horizontal walls towards the interface. Under shear dominated flow, these vortices should develop and roll up in a spiral manner once the growth rate has settled. The fig. 4.21 shows such development examples.

On fig. 4.23, when the vortex sheet starts to roll up, strong shear provokes detachment of droplets in a chaotic manner. These droplets or inclusions live in the opposite phase until latter coalescence. The method is in general able to capture the development of vortices, and the growth of the vortex layer thickness [237]. However, numerical dissipation is important and interfacial terms are likely to be under resolved. Due to limited computation resources available, we were not able to investigate the problem in three dimensions, neither with increased resolution nor with different anti-diffusive/capillary schemes. We believe that the interfacial term on the tip of the expected spiral, being difficult to capture since the interface is pinched with high curvatures values, is most likely erroneous as seen on fig. 4.23. Comparison with experimental data could thus show very different results. We also mention the fact that accumulated numerical diffusion of the various schemes used has the tendency to stabilize the flow.

This test case also shows the limit of the method applied on regular meshes requiring a substantial amount of computing resources in order to capture areas where dissipation of the numerical scheme dominates (inner spiral in the cat eye patterns). However,

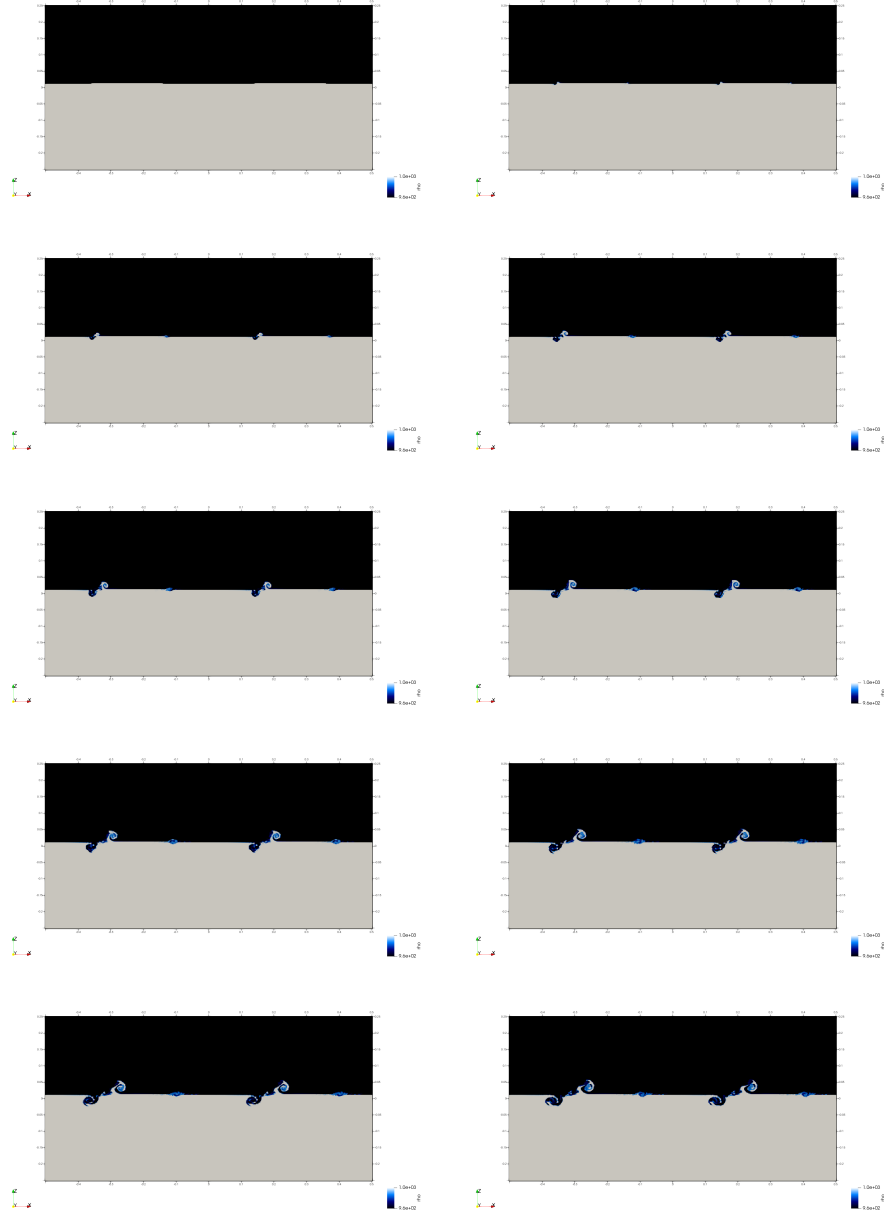


Figure 4.23 – Development of the Kelvin-Helmholtz instability (first steps).

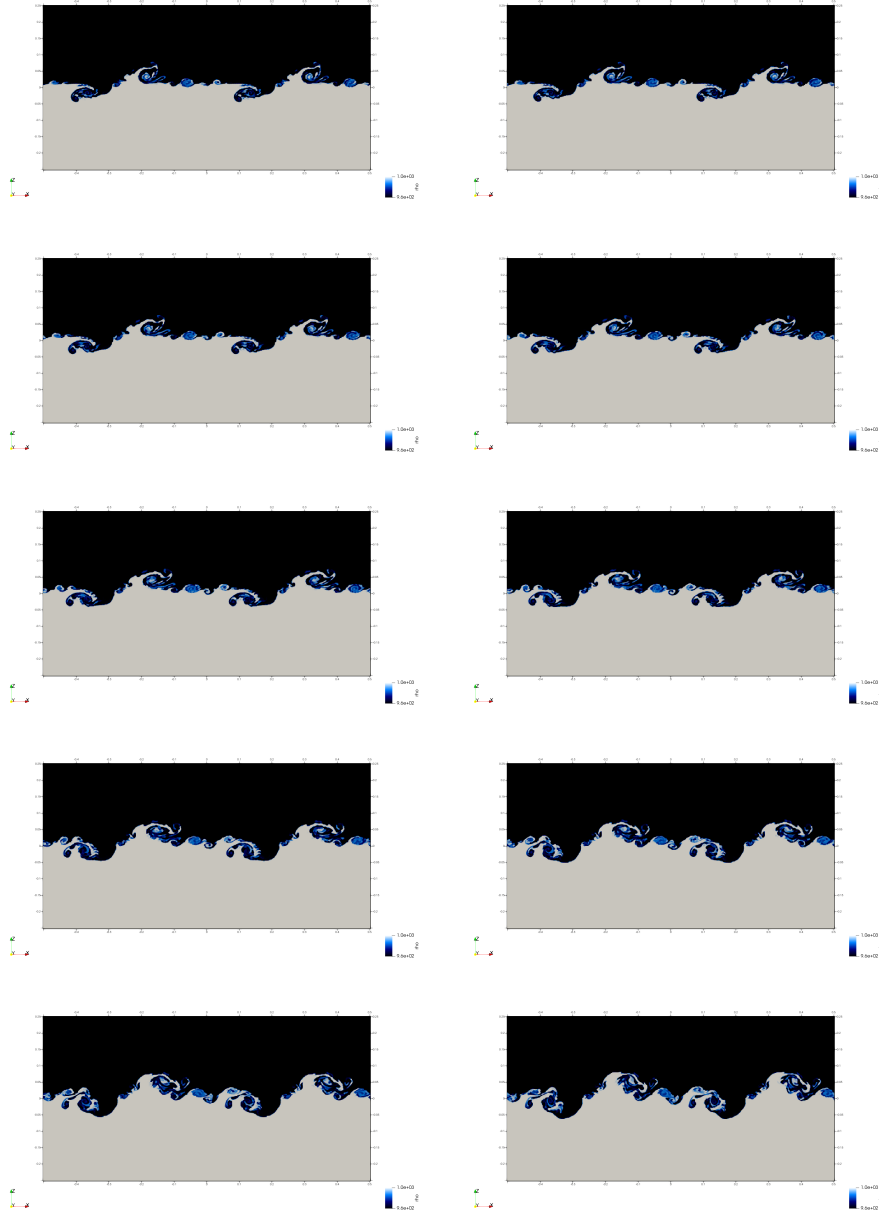


Figure 4.23 – Development of the Kelvin-Helmholtz instability (last steps).

the AMR^v method is able to reduce the computing costs whilst increasing resolution in regions of strong density gradients [238]. With this multi-level approach, we might be able to capture the development of the spiralling pattern with sufficient accuracy. Another way to stabilize the development of the structure is to tune the viscosity up. With a higher viscous fluid, the structures are more coherent. With a too weak viscosity, the structures rapidly degenerate and finer scale inclusions detach from the fluid bulk.

4.1.6 Bubble rise in buoyant flow

ρ_1 [kg·m ⁻³]	ρ_2 [kg·m ⁻³]	μ_1 [kg·m ⁻¹ ·s ⁻¹]	μ_2 [kg·m ⁻¹ ·s ⁻¹]	σ [kg·s ⁻²]		
1000.	100.	10.	1.	24.5		
		g [m·s ⁻²]	no. cells	domain [m]	$BC(s)$	lim
		.98	160x320	1.x2.	$W \times W$	NB

Table 4.6 – Numerical parameters of the bubble rise in a channel (first test case).

First test case: terminal velocity

In this test, buoyancy induces a bubble rise in a fluid initially at rest in a channel [239]. This test case is interesting for the wall recirculation, the droplet deformation induced with the boundaries proximity, and the competition between capillary forces, pressure and buoyancy. We use the parameters of [240], with a mesh step of $\Delta = \frac{1}{160}$, and an adaptive time-stepping, as exposed in table 4.6. Capillary terms are applied on the stiff interface, with no artificial thickening (singular pulse). We use the least compressive limiter from [152], and the *CSF* method for the capillary terms.

In fig. 4.24, we recover the results of [240], with a delay of .3 [s], whose origin must be investigated. We also note that the shape of the bubble lacks some deformation towards the ellipsoidal shape.

The rise velocity of the droplet, as defined in [240], is the average bubble velocity in the direction of gravity. An overestimation of the velocities versus the reference article can be seen on fig. 4.25. The difference might lie in the way we recover the bubble velocity, since we use an implicit defined interface, and some mixture cells (continuous and dense phase) contribute to the reduction of the terminal velocity.

The bubble reaches a terminal velocity, and deformation occurs until stabilization at the steady state, when buoyancy is balanced by viscous dissipation. A drag coefficient quantifying the resistance of the droplet to buoyancy induced movement and the congestion of the channel by the secondary phase could be extracted from fig. 4.25. The larger blocking ratio due to deformation induces a slight reduction of the terminal velocity relative to the peak velocity observed. The bubble remains symmetric as expected, which suggests that the code is bug free (synchronization points, halo exchanges) under domain decomposition.

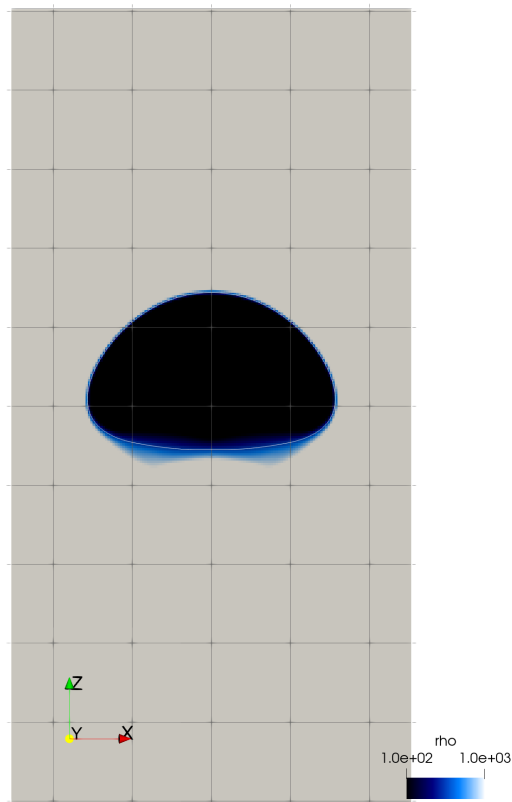


Figure 4.24 – Density contour, at time $t = 3.3$ [s], for the first test case 160x320.

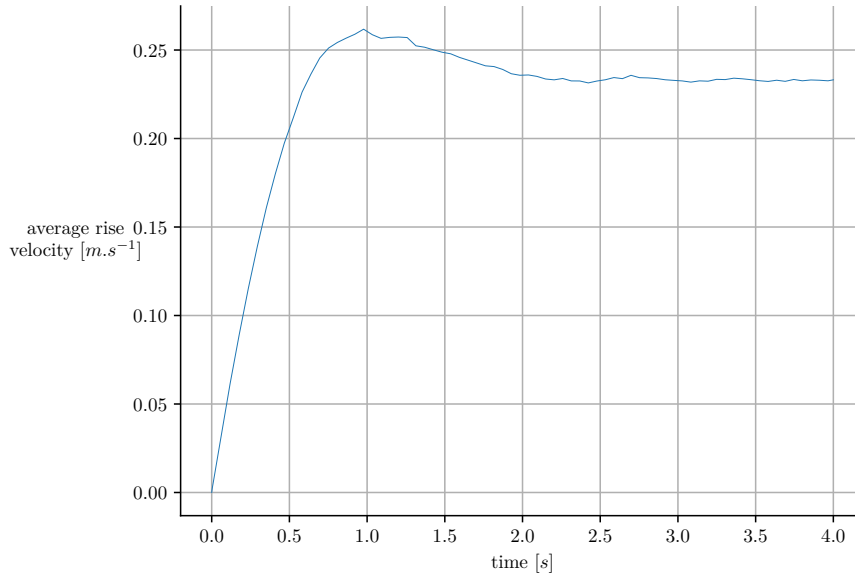


Figure 4.25 – Rise and terminal velocity of the inclusion (80x80).

ρ_1 [kg·m ⁻³]	ρ_2 [kg·m ⁻³]	μ_1 [kg·m ⁻¹ ·s ⁻¹]	μ_2 [kg·m ⁻¹ ·s ⁻¹]	σ [kg·s ⁻²]
1000.	1.	10.	.1	1.96

g [m·s ⁻²]	no. cells	domain [m]	$BC(s)$	lim
.98	80x160	1.x2.	$W \times W$	DL

Table 4.7 – Numerical parameters of the the bubble rise in a channel (second test case).

Second test case: high density ratio

In fig. 4.26, we use a high density ratio (1 : 1000) as seen in table 4.7, and the anti-diffusive formulation of [46], together with a conservative formulation. This scheme is known to compress profiles, even for initially smooth solutions. A multi-grid solver with two levels described in algorithm 9, with pressure splitting of [241] and constant coefficients Poisson equation is used. The problem with $\mathcal{O}(15000)$ unknowns is solved on 4 processes, and results are obtained within 13 minutes for 46.000 steps.

We can observe filaments detaching from the secondary phase. The side filaments are induced by fluid circulation between the walls and the droplets: they also appear in the benchmark of [240]. However, the origin of the small fragments in the wake of the droplet in the recirculation zone is more difficult to explain. Those fragments do not appear in the results of [240]. It is likely that the Weber number is now high enough, so that the droplets starts to disaggregate under increasing inertial forces and weaker effect of surface tension.

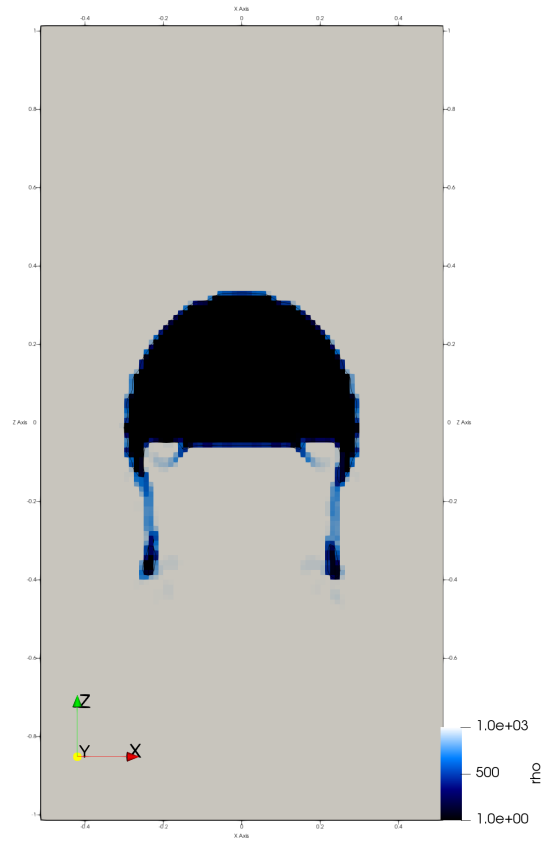


Figure 4.26 – Density contour, at time $t = 3$ [s], for the second test case 80x160.

4.1.7 Conclusion

Nous avons présenté une méthode de résolution des écoulements diphasiques, à savoir le transport anti-diffusif des phases sur maillages cartésien régulier, qui semble adaptée à la modélisation des phénomènes d'interaction fluide-structure sur des maquettes représentant une portion de générateur de vapeur. Nous allons par la suite coupler cette méthode multi-fluides à une méthode d'inclusion des corps immergés dans un écoulement fluide, l'objectif final étant de pouvoir simuler le cas canonique de la circulation d'inclusions au sein d'un faisceau de tubes dont le tube central est flexible et les tubes l'entourant rigides.

4.2 Fluid-Solid cases

The classic examples developed in this section stress out the numerical response of the immersed object to the fluid forces, and the retro-action on the bodies onto the fluid equations. The multiphase tube bundle case is close to the experiments conducted at *CEA*: the future prospects using this method should thus be directed towards an experimental and physical validation.

4.2.1 Vibration of a cylinder in quiescent fluid

In order to determine the added mass and damping coefficients we impose a sinusoidal movement on the tube:

$$q_x(t) = A_0 \sin(\omega t). \quad (4.6)$$

The added mass and damping coefficients can be determined using the phase method:

$$M_a = + \frac{F_{max} \cos(\phi)}{\omega^2 A_0}, \quad (4.7)$$

$$C_a = - \frac{F_{max} \sin(\phi)}{\omega A_0}, \quad (4.8)$$

where ϕ is the phase shift between the imposed displacement $q_x(t)$ and the total fluid force $F(t)$ acting on the cylinder (pressure and viscous contributions).

The phase shift can be determined using different methods such as curve fitting, zero crossing detection, or using the correlation between two (not necessarily harmonic) signals. Spatial convergence analysis has to be made in order to ensure that the integration method is sufficiently resolved for the computation of pressure and viscous forces around the cylinder.

The drag and lift forces are estimated using discrete integration of eqs. (3.26) and (3.27). The phase shift is then estimated using the imposed displacement eq. (4.6) and the total force around the cylinder eq. (3.28). In order to separate the physics, it seems necessary to separate the contribution of the pressure forces and the viscous forces exerted on the cylinder. This will be of use for the stability analysis and the interpretation of the fluid-structure numerical coupling.

4.2.2 Multiphase flow around an obstacle

This simple test with a droplet impacting a static obstacle is solely set up to demonstrate convergence of the numerical behaviour, and a correct behaviour of the phase separation close to the solid boundaries. The inclusion is separated not on the obstacle itself, as seen on fig. 4.27, but around the velocity transition layer induced by the regularization method (section 3.5.3) over a few cells, and because of the velocity boundary layer on the nose of the obstacle. Surface tension is disabled for this test, resulting in longer separation time, and trailing density on the obstacle.

Numerically, the volumetric source term is expressed as an acceleration term function of the deviation from the continuous phase:

$$\mathbf{S}(\mathbf{x}, t) = \mathbf{g} (\rho(\mathbf{x}, t) - \rho_1) , \quad (4.9)$$

such that the velocity increment induced by buoyancy results in:

$$\mathbf{u}_{\text{buoyancy}} = \Delta t \frac{\mathbf{S}}{\rho} .$$

It would be interesting to profile the immersed object in order to provoke separation of the inclusion, or at the opposite use a larger obstacle provoking adhesion of the inclusion to the obstacle under buoyant forces. Theoretically, the inclusion would remain attached and symmetry is to be observed. We observe this symmetry for at least one passage across periodic boundaries.



Figure 4.27 – Impact of an inclusion on a static obstacle in channel with densities $\rho \in \{5, 10\}$, and without surface tension.

4.2.3 Multiple inclusions through a tube bundle

ρ_1 [kg·m ⁻³]	ρ_2 [kg·m ⁻³]	μ_1 [kg·m ⁻¹ ·s ⁻¹]	μ_2 [kg·m ⁻¹ ·s ⁻¹]	σ [kg·s ⁻²]		
100.	10.	$1 \cdot 10^{-2}$	$1 \cdot 10^{-3}$.1		
no. cells		domain [m]	$BC(s)$	lim	time [s]	wall time[h]
512x256		.2x.1	W_{xP}	DL	.6	1
n° procs			ρ_s [kg·m ⁻³]	ζ_s [−]	k_s [kg·s ⁻²]	IBM
16			500.	5 [%]	$1 \cdot 10^{-2}$	RG

Table 4.8 – Fluid, simulation and solid parameters for the bundle case.

We use a semi-implicit $CN \theta^i$ scheme in order to avoid stringent restriction due to the viscous part of the $NSE(s)$. The forcing method is applied on the centre tube with two degrees of freedom in the x and z directions. The forcing is regularized with the method exposed in [192] in order to avoid numerical pressure surges. The solid numerical parameters were arbitrarily chosen as exposed in table 4.8, and more tuning needs to be performed in order to reach water-air flow with industrial characteristics of the bundle. Numerical study of the influence of the parameters remains to be performed.

recursive regularization	surface tension	semi-implicit prediction phase	immersed boundaries	incompressible projection	conservative transport
3 [%]	7 [%]	33 [%]	3 [%]	51 [%]	3 [%]

Table 4.9 – Decomposition of the time spent during a single timestep.

The table 4.9 shows that most of the time is spend during the projection method exposed in section 3.3.4. With two unknowns in 2D (u, w velocities), the semi-implicit θ scheme converges faster in only a few iterations and a third of the CPU time is spent in the prediction phase. We expose some hints in order to enhance the convergence of the implicitation steps, but this goes beyond the scope of this thesis and requires rework of the code (maybe with parallel sparse matrix structures), with only hypothetical enhancement on computation performance and scalability. Another approach would be to consider specific methods such as treating the interface pressure jump as described in section 3.1.6.

i. Implicit fraction of a semi-implicit scheme

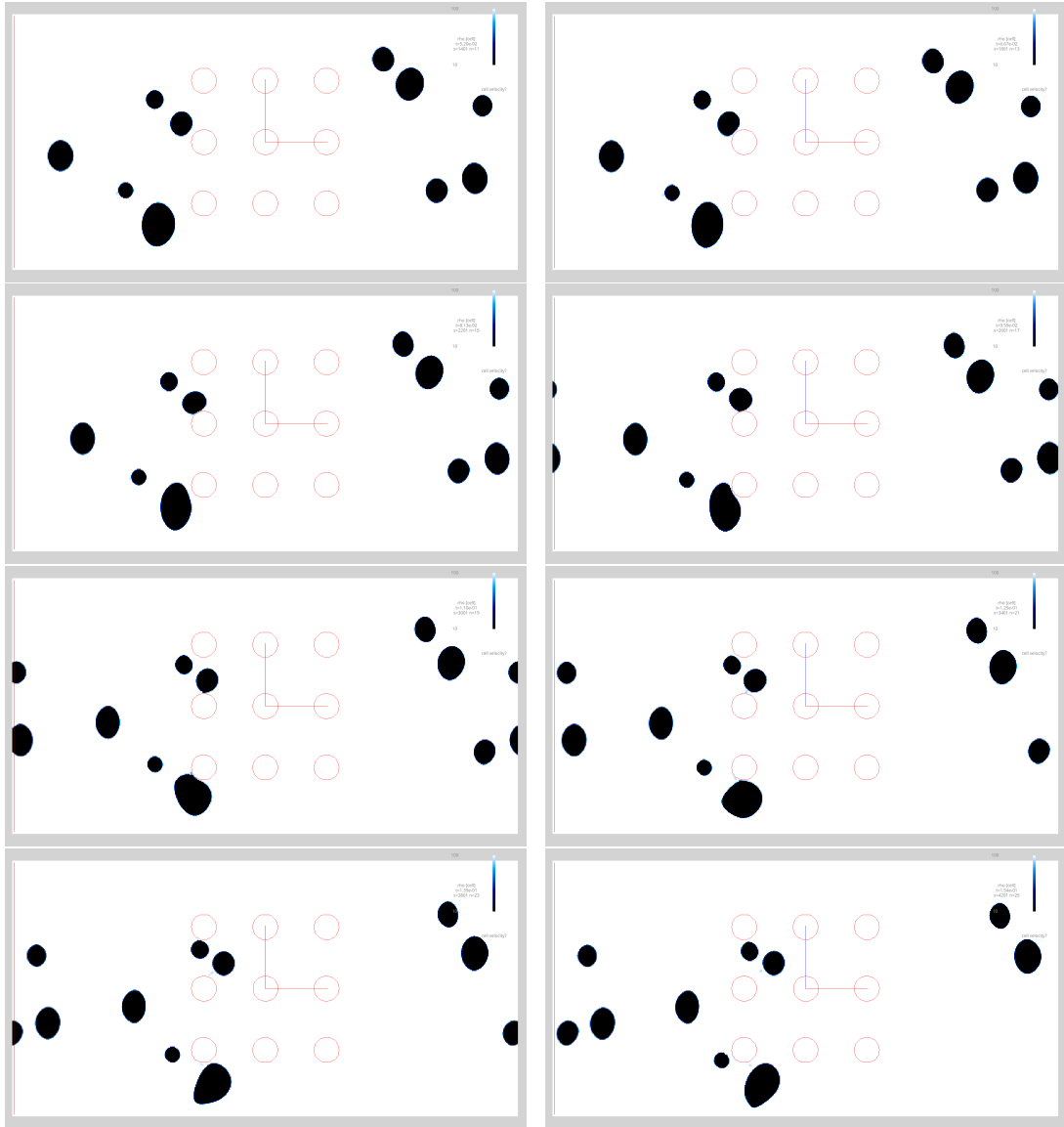


Figure 4.28 – Multiphase flow through a bundle (centre tube flexible).



Figure 4.28 – Coalescence of multiple inclusions.

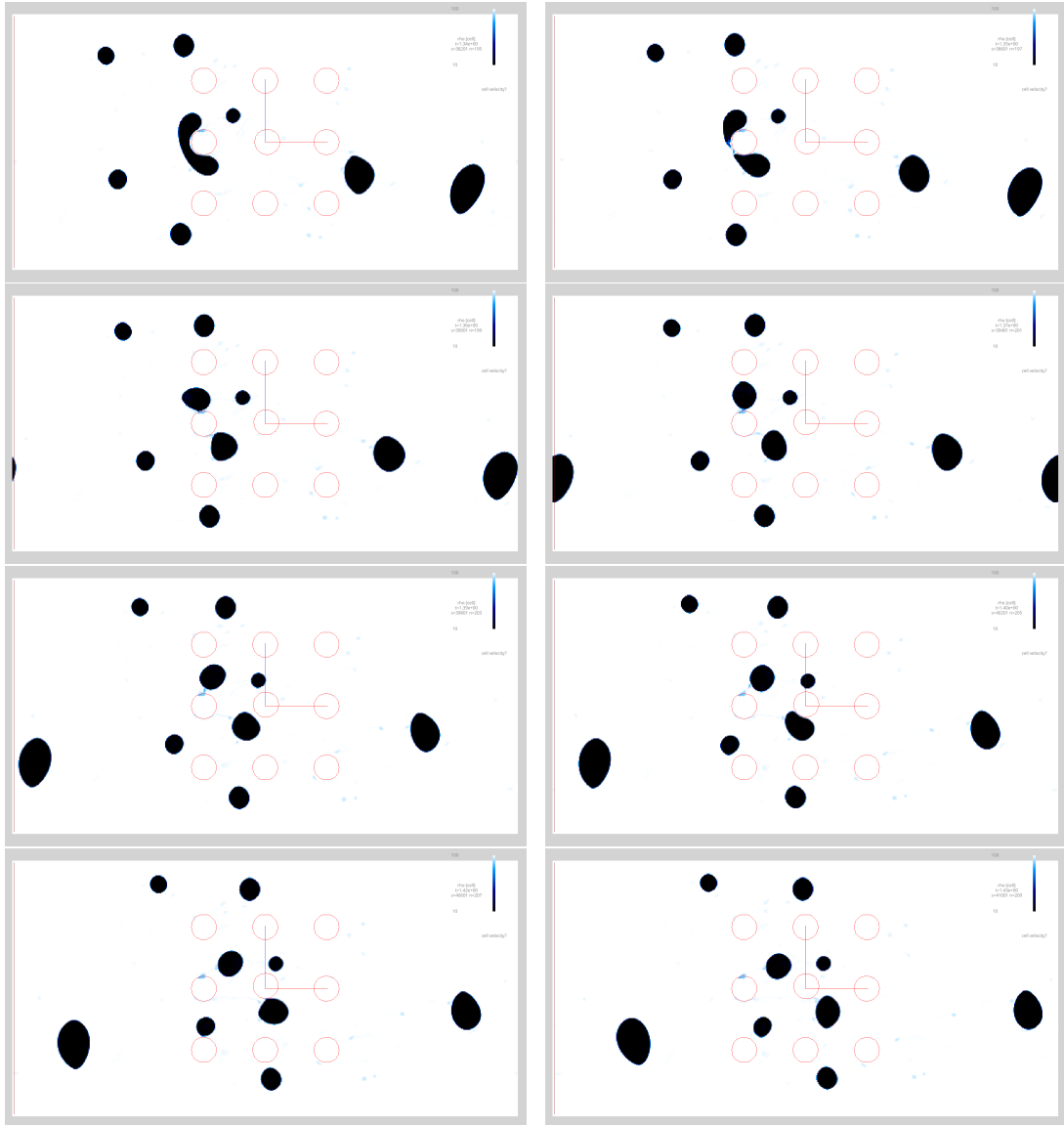


Figure 4.28 – Breakup of inclusions on cylinder impact.

Discrete time sequences extracted from the simulation are exposed in fig. 4.28. At the beginning of the simulation on fig. 4.28, the inclusions start to move under buoyancy (acceleration term in the x direction). The acceleration term is applied to the secondary fluid only as a drifting term, with the standard gravity magnitude of $9.8 \text{ [m}\cdot\text{s}^{-2}]$. The smallest inclusion is immediately carried into the wake of the lowest inclusion in the bottom left hand corner. The interfaces follow the obstacle contours, with consequent deformation, and they recover a circular shape under the effect of surface tension with the algorithm described in section 3.1.3.3. The surface tension coefficient between the two fluids is kept important enough in order to maintain droplets cohesion (interface regularity) and avoid breakup. Leakage of one fluid into another is limited with the use of the most compressive anti-diffusive scheme [46], in a conservative flux formulation. Periodic boundary conditions are applied on the x direction in order to assess the eulerian methods under a high number of timesteps.

After a few seconds of simulation, the two bubbles near the center tube coalesce to form a larger bubble as seen on fig. 4.28. The front bubble is slowed down by the confinement of the tubes, allowing the two inclusions to merge. There is systematic coalescence, and the method does not allow two droplets to bounce on each other. The limit for droplets coalescence is the continuous phase film resolution when the two bubbles approach one another. Especially, how this fluid is forced away determines the physics of the merge. In our application, the boundary layer is under resolved and can only be captured with specific high resolution methods (even with a conforming method such as *FT*, it is not guaranteed to capture the correct behaviour). Another coalescence is observed, where the front inclusion is slowed down and flattened by the repulsion of the fluid at the leading edge. This allows the smaller inclusion (resolution of about 20 cells in diameter) to catch up in the wake of the front bubble and coalesce.

Breakup is observed at the end of the simulation subset on fig. 4.28. As the bubble hits the center left obstacle, the secondary phase fluid is stretched until separation. We observe small leakage of density near the obstacle boundaries which is likely to come from shear effect on velocities. The restoring force effect induced by surface tension is especially visible after separation. The breakup phenomenon would be interesting to investigate for example against alternate methods such as the *LSM* [242].

4.2.4 Conclusion

Dans ce chapitre, nous avons exposé le couplage de la partie corps immergés aux méthodes diphasiques monochamp proposées dans le chapitre précédent. L'inclusion d'obstacles mobiles dans le domaine a nécessité une méthode de régularisation permettant d'atténuer les oscillations de pression lors de la transition d'état des mailles allouées au solide. Nous n'avons cependant pas réussi à exploiter cette méthode de manière intensive afin d'analyser les spectres d'efforts sur les tubes, à des fins d'adimensionnement des forces de couplage fluide-structure. Cette étape de mesure des spectres nécessitera une instrumentation fine du code, sur un cas test reproduisant toutes les conditions d'un essai expérimental en boucle air-eau (vitesses, taux de vide, générateur de phases, structure de la turbulence d'entrée) pour réaliser des études paramétriques. Il semble également nécessaire de revoir la performance du code en terme de scalabilité, dominée par le solveur en pression (bien que les derniers essais sur le préconditionnement de l'algorithme bi-cgstab par une méthode multi-niveaux soient prometteurs), avant de pouvoir atteindre les régimes en condition générateur de vapeur. Cependant, le démonstrateur permet d'exposer la faisabilité de la simulation numérique ainsi que la pertinence ou encore les limites des méthodes proposées.

Chapter 5

General conclusion and perspectives

This thesis falls within the category of *DNS* of multiphase flows coupled with *FSI*, for the safety studies of nuclear power plants.

In the second chapter, we recalled the scope of the thesis and the existing methods in a bibliographic study. Choices have been made with the scope of future industrial applications, according to the existing knowledge. We retained the Front Tracking method for comparison, super-imposed on top of a cartesian mesh for numerical juxtaposition with the pure eulerian method used. This method is suitable for cases where the *DNS* computations are cheap (microfluidics, ...).

The third chapter exposed the development of numerical methods for the multiphase liquid phase and the fluid-structure coupling. The first contribution is the use of an anti-diffusive method for density transport. After this, a convergent pressure solver was developed from scratch for incompressible multiphase flows, since numerous methods for single phase flows were found to be unable to scale well or sometimes converge for multiphase flows. Then, a regularization method suited for a fast eulerian solver has been developed to account for relevant topological terms at the interface singularity (surface tension). This algorithm is key since it is able to maintain the droplet cohesion and correct interfacial balance at all resolved scales, under the constraint of usage of an anti-diffusive scheme for the phase discontinuity transport, without the use of recalibration methods of mesh conforming nodes. At last, we interested ourselves to methods embedding moving solid in fluids, especially the bi-directional coupling between the fluid and solid physics. A specific method was used in order to reduce the spurious pressure oscillations in the context of eulerian methods.

The fourth chapter was dedicated to the exploitation of relevant test cases. First, the two-phase tests including buoyancy driven flows and zero-gravity test case. In a second phase, the fluid-structure examples with immersed boundaries in mono-phase or two-

phase flows. We were only able to run the fluid-structure simulations at the end of the thesis, thus the lack of quantitative results on flow regime, parametric studies of the influence of the bundle parameters, etc. We believe that full physical cases are reachable with our method, provided time to run the simulations, analyse and interpret the results is available. A wide variety of studies is to be investigated as perspective: the numerical impact of spurious currents, the triplet points gas - liquid - solid [243], the breakup and coalescence behaviours, contraction/expansion of a weakly compressible phase, statistical analysis [244] of multiple inclusions for upscaling in CFD codes.

Appendix A

CEMRACS 2016: Numerical challenges in parallel scientific computing

A detached work was conducted during the 2016 *CEMRACS*ⁱ summer school (numerical challenges in parallel scientific computing) on the topic "High Performance simulation of geothermal systems: ComPASS-Geothermal". This work was published in June 2018 under the title "Parallel Geothermal Numerical Model With Fractures And Multi-branch Wells". The 6 weeks long project, whilst dealing with different physics, was an applied introduction to team work on research projects. I also benefited from enhancing my skills on parallel coding and modern *Fortran* language. The project was a joint work between *BRGM*ⁱⁱ, *INRIA*ⁱⁱⁱ: team *COFFEE*^{iv}, and *LJAD*^v. The goal was to represent and implement the numerical model of a well into the Compass code, by the mean of a connected graph through an unstructured discretized medium. The first task was to represent well-reservoir interaction by the mean of a Darcy flux. After a thermodynamic balance, computed at each timestep from a equilibrium flash of a multicomponent liquid. Then, the well mode was toggled between imposed pressure or imposed flux by a non-linear Newton update. The assembly of the Jacobian matrix was complex since it required multicomponent equations, an elliptic part for the pressure, and an hyperbolic part for the saturations/compositions of the multi-species flow. The time-stepping of geological phenomena required an all implicit formulation, whilst being constrained by a strong heterogeneity and anisotropy induced by the reservoir interaction with a net-

i. Centre d'Eté Mathématique de Recherche Avancée en Calcul Scientifique

ii. Bureau de Recherches Géologiques et Minières

French geological survey

iii. Institut National de Recherche en Informatique et en Automatique

French Institute for Research in Computer Science and Automation

iv. COmplex Flows For Energy and Environment

v. Laboratoire J.A. Dieudonné (Université de Nice)

work of planar fractures and a mostly orthogonal graph of well nodes. The meshing and partitioning uses the *CGAL*^{vi}/METIS libraries while the linear solver was implemented using the *PETSc*^{vii} library in *Fortran*, with Schur decomposition beforehand. For performance issues, an algebraic multi-grid preconditioner was used for the elliptic part and an incomplete LU factorization for the global matrix. I also had the opportunity to interact with another team about code coupling issues on a thermo-hydraulic code *TRUST*^{viii} [245, 246] developed at *CEA*.

vi. Computational Geometry Algorithms Library

vii. Portable Extensible Toolkit for Scientific Computation

viii. TrioU Software for Thermohydraulics

PARALLEL GEOTHERMAL NUMERICAL MODEL WITH FRACTURES AND MULTI-BRANCH WELLS

LAURENCE BEAUDE¹, THIBAUD BELTZUNG², KONSTANTIN BRENNER³, SIMON LOPEZ⁴,
ROLAND MASSON⁵, FARID SMAI⁶, JEAN-FRÉDÉRIC THEBAULT⁷ AND FENG XING⁸

Abstract. To answer the need for an efficient and robust geothermal simulation tool going beyond existing code capabilities in terms of geological and physical complexity, we have started to develop a parallel geothermal simulator based on unstructured meshes. The model takes into account complex geology including fault and fracture networks acting as major heat and mass transfer corridors and complex physics coupling the mass and energy conservations to the thermodynamic equilibrium between the gas and liquid phases. The objective of this Cemracs project was to focus on well modeling which is a key missing ingredient in our current simulator in order to perform realistic geothermal studies both in terms of monitoring and in terms of history matching. The well is discretized by a set of edges of the mesh in order to represent efficiently slanted or multi-branch wells on unstructured meshes. The connection with the 3D matrix and the 2D fracture network at each node of the well is accounted for using Peaceman's approach. The non-isothermal flow model inside the well is based on the usual single unknown approach assuming the hydrostatic and thermodynamical equilibrium inside the well. The parallelization of the well model is implemented in such a way that the assembly of the Jacobian at each Newton step and the computation of the pressure drops inside the well can be done locally on each process without MPI communications.

¹ Université Côte d'Azur, Inria, CNRS, LJAD, UMR 7351 CNRS, team Coffee, Parc Valrose 06108 Nice Cedex 02, France, laurence.beaude@unice.fr

² CEA Saclay, DEN/DANS/DM2S/STMF/LMEC thibaud.beltzung@cea.fr

³ Université Côte d'Azur, Inria, CNRS, LJAD, UMR 7351 CNRS, team Coffee, Parc Valrose 06108 Nice Cedex 02, France, konstantin.brenner@unice.fr

⁴ BRGM, 3 avenue Claude-Guillemin, BP 36009, 45060 Orléans Cedex 2, France, s.lopez@brgm.fr

⁵ Université Côte d'Azur, Inria, CNRS, LJAD, UMR 7351 CNRS, team Coffee, Parc Valrose 06108 Nice Cedex 02, France, roland.masson@unice.fr

⁶ BRGM, 3 avenue Claude-Guillemin, BP 36009, 45060 Orléans Cedex 2, France, f.smai@brgm.fr

⁷ Storengy, jean-frederic.thebault@storengy.com

⁸ Université Côte d'Azur, Inria, CNRS, LJAD, UMR 7351 CNRS, team Coffee, Parc Valrose 06108 Nice Cedex 02, and BRGM, feng.xing@outlook.com

Résumé. Afin de dépasser les limites des codes actuels de simulation des systèmes géothermiques en matière de complexité géologique et physique, nous avons initié le développement d'un nouveau simulateur d'écoulements géothermiques parallèle à base de maillages non structurés. Le modèle prend en compte une géologie complexe incorporant notamment les réseaux de failles qui jouent un rôle majeur dans le transport de masse et d'énergie, ainsi qu'une physique complexe couplant les conservations de la masse et de l'énergie à l'équilibre thermodynamique entre les phases liquide et gazeuse. L'objectif de ce projet Cemracs était d'y incorporer un modèle de puits qui constitue un ingrédient essentiel pour réaliser des études géothermiques réalistes à la fois pour la surveillance du réservoir et la reproduction des historiques de production. Le puits est discrétisé par un sous ensemble d'arêtes du maillage de façon à pouvoir représenter efficacement des puits déviés ou multi-branches. La connection avec la matrice 3D et le réseau de failles 2D repose sur des indices de productivité de type Peaceman. Le transport de masse et d'énergie dans le puits se base sur un modèle classique en simulation de réservoir à une inconnue par puits qui suppose l'équilibre hydrostatique et thermodynamique dans le puits. La parallélisation du modèle de puits est réalisée de façon à pouvoir assembler la Jacobienne et à calculer les pertes de charge dans le puits localement sur chaque processus sans nécessiter de communications MPI.

INTRODUCTION

Geothermal energy is a carbon-free steady energy source with low environmental impact. In countries with a favorable geological context, high temperature geothermal energy can make a significant contribution to power production. On the French territory, it is already an attractive option in volcanic islands context compared to importing fossil fuel. Today, about 5 percent of yearly electricity consumption of Guadeloupe already comes from geothermal energy and it is essential for achieving energetic and environmental targets, according to which the overseas territories should produce 50 percent of their electricity consumption from renewable resources by 2020 and achieve their self sufficiency in 2030. As for other parts of the world, the geothermal development potential of the Caribbean islands is high and several industrial projects are under development or already underway, in French overseas territories (Guadeloupe, Martinique) as well as in nearby islands (Dominica, Montserrat, St. Kitts & Nevis, St Lucia...) that currently depend mainly on diesel for power generation.

Numerical modeling has become essential in all phases of geothermal operations. It is used in the exploration phases to assess the geothermal potential, validate conceptual hypothesis and help well siting. Field development and resource management need quantitative estimation to prevent resource exhaustion and achieve its sustainable exploitation (production/injection scenarios). Finally, numerical modeling is also helpful in studying exploitation related industrial risks such as the interaction with shallow water levels (drinking water resources, hydrothermal vents or eruption).

There is a need to develop new efficient and robust simulation tools to go beyond existing code capabilities in terms of geological and physical complexity [25,32]. In particular such code should be able to deal with fault and fracture networks acting as major heat and mass transfer corridors in high energy geothermal reservoirs and also to simulate both under critical and super critical thermodynamical domains. Existing tools such as Tough2 [31], used for more than 25 years in geothermy, are limited to structured meshes and are not able to integrate conductive fractures. Moreover, their parallel efficiency is very limited.

This has motivated the development of a new geothermal simulator based on unstructured meshes and adapted to parallel distributed architectures with the ability to represent fractures as co-dimension 1 surfaces connected to the surrounding matrix domain. The current version of this simulator is described in [42]. The objective of this Cemracs project is to bring the development of this simulator to a level where operational use is possible and real geothermal test cases can be considered. In this regard, wells are central features of geothermal exploitation and are the main focus of this work.

The use of lower dimensional rather than equi-dimensional entities to represent fracture or fault networks has been introduced in [3, 9, 19, 23, 28] to facilitate the grid generation and to reduce the number of degrees of freedom of the discretized model. The reduction of dimension in the fracture network is obtained from the equi-dimensional model by integration and averaging along the width of each fracture. The resulting so called hybrid-dimensional model couple the 3D model in the matrix with a 2D model in the fracture network taking into account the jump of the normal fluxes as well as additional transmission conditions at the matrix-fracture interfaces. These transmission conditions depend on the mathematical nature of the equi-dimensional model and on additional physical assumptions. They are typically derived for a single phase Darcy flow for which they specify either the continuity of the pressure in the case of fractures acting as drains [3, 10] or Robin type conditions in order to take into account the discontinuity of the pressure for fractures acting either as drains or barriers [4, 13, 19, 28]. In our case, the fractures will be assumed to act as drains both for the Darcy flow and for the thermal conductivity leading us to set the pressure and temperature continuity as transmission conditions at the matrix fracture interfaces.

The discretization of hybrid-dimensional Darcy flow models has been the object of many works using cell-centered Finite Volume schemes with either Two Point or Multi Point Flux Approximations (TPFA and MPFA) [1, 2, 4, 20, 23, 36, 40], Mixed or Mixed Hybrid Finite Element methods (MFE and MHFE) [3, 22, 28], Hybrid Mimetic Mixed Methods (HMM, which contains mixed-hybrid finite volume and mimetic finite difference schemes [15]) [5, 10, 12, 18], Control Volume Finite Element Methods (CVFE) [9, 20, 29, 30, 35].

This article focus on the Vertex Approximate Gradient (VAG) scheme which has been introduced for the discretization of multiphase Darcy flows in [17] and extended to hybrid-dimensional models in [10–13, 41, 42]. The VAG scheme uses nodal and fracture face unknowns in addition to the cell unknowns which can be eliminated without any fill-in. Thanks to its essentially nodal feature, it leads to a sparse discretization on tetrahedral or mainly tetrahedral meshes. It has the benefit, compared with the CVFE methods of [9, 29, 30, 35], to avoid the mixing of the control volumes at the matrix fracture interfaces, which is a key feature for its coupling with a transport model. As shown in [11] for two phase flow problems, this allows for a coarser mesh size at the matrix fracture interface for a given accuracy.

At the reservoir scale of a few kilometers, the mesh cannot resolve the well boundary with a radius of say 10 cm and the well is modeled as a Dirac source term along the well trajectory. Most well models in reservoir simulations are defined by a set of connected perforations, each perforation belonging to a cell of the mesh [33, 34]. This type of approach is adapted to cell-centered finite volume discretization. In order to take advantage of unstructured meshes and of the nodal feature of the VAG scheme, it is more convenient in our case to discretize each well as a subset of edges of the mesh. This alternative approach provides an efficient way to represent slanted and multi-branch wells. The fluxes connecting the well with the 3D matrix and the 2D fracture network at each node of the well will be computed using Peaceman's approach [14, 33, 34]. It is based on a Two Point Flux Approximation with a transmissibility taking into account the unresolved singularity of the pressure (or temperature) solution in the neighborhood of the well. The non-isothermal flow model inside the well is defined in the spirit of what is conventionally done in oil reservoir simulators [7] using a single implicit unknown for each well corresponding to a reference pressure often called the *bottom hole pressure*. The pressures along the well will be deduced from the bottom hole pressure assuming that the pressure is hydrostatic inside the well. The temperatures along the well will be computed assuming thermal equilibrium and a stationary flow inside the well. Then, the well equation is obtained by the complementary conditions between a specified well mass flow rate and a specified limit bottom hole pressure. By connecting all the nodes along the well trajectory to the well reference pressure unknown, the well equation introduces an additional connectivity. This difficulty will be accounted for by the definition of ghost and own wells for each process and by extension of the ghost nodes of each process in order to take into account the additional connections induced by the own and ghost wells. This allows to assemble the Jacobian and to compute the well pressure drops locally on each process without the need of MPI communications.

The outline of the remaining of the paper is as follows. In section 1, the hybrid-dimensional model presented in [42] is recalled. Although the implementation has been done for the multi-phase compositional model defined in [42], we focus here on the particular case of a non-isothermal single-component single-phase Darcy flow model in order to simplify the presentation. Section 2 introduces the space and time discretization of the model. The definitions of the multi-branch well data structure and of the well model are detailed in subsection 2.3. Section 3 presents the parallel implementation of the model including the partitioning of the mesh and wells, as well as the parallel assembly of the nonlinear and linear systems to be solved at each time step of the simulation. The solution of the linear systems uses the parallel linear solver library PETSc [8] and is based on the GMRES iterative solver preconditioned by a CPR-AMG preconditioner [27,37]. The implementation of the CPR-AMG preconditioner takes into account the well equations in the definition of the pressure block. Two numerical tests are presented in section 4. The first test case is used to validate our model. It considers an isothermal single-phase stationary Darcy flow on a simple geometry with one horizontal fracture and one vertical well for which an analytical pressure solution can be obtained. The second test case considers a single-phase non-isothermal transient flow on a complex geometry including three intersecting fractures, one slanted injection well and one multi-branch production well.

1. HYBRID-DIMENSIONAL NON-ISOTHERMAL SINGLE-PHASE DISCRETE FRACTURE MODEL

This section recalls, in the particular case of a non-isothermal single-component single-phase Darcy flow model, the hybrid-dimensional model introduced in [42].

1.1. Discrete Fracture Network

Let Ω denote a bounded domain of \mathbb{R}^3 assumed to be polyhedral. Following [3,10,12,19,28] the fractures are represented as interfaces of codimension 1. Let J be a finite set and let $\bar{\Gamma} = \bigcup_{j \in J} \bar{\Gamma}_j$ and its interior $\Gamma = \bar{\Gamma} \setminus \partial\bar{\Gamma}$ denote the network of fractures $\Gamma_j \subset \Omega$, $j \in J$, such that each Γ_j is a planar polygonal simply connected open domain included in a plane of \mathbb{R}^3 . The fracture width is denoted by d_f and is such that $0 < \underline{d}_f \leq d_f(\mathbf{x}) \leq \bar{d}_f$

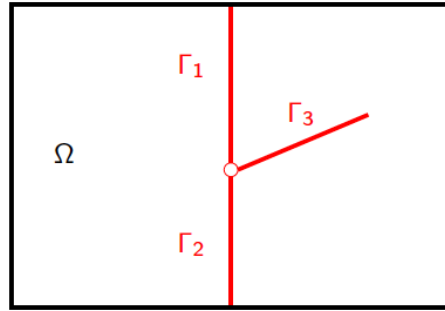


FIGURE 1. Example of a 2D domain with 3 intersecting fractures $\Gamma_1, \Gamma_2, \Gamma_3$.

for all $\mathbf{x} \in \Gamma$. We can define, for each fracture $j \in J$, its two sides $+$ and $-$. For scalar functions on Ω , possibly discontinuous at the interface Γ (typically in $H^1(\Omega \setminus \bar{\Gamma})$), we denote by γ^\pm the trace operators on the side \pm of Γ . Continuous scalar functions u at the interface Γ (typically in $H^1(\Omega)$) are such that $\gamma^+ u = \gamma^- u$ and we denote by γ the trace operator on Γ for such functions. At almost every point of the fracture network, we denote by \mathbf{n}^\pm the unit normal vector oriented outward to the side \pm of Γ such that $\mathbf{n}^+ + \mathbf{n}^- = 0$. For vector fields on Ω , possibly discontinuous at the interface Γ (typically in $H_{\text{div}}(\Omega \setminus \bar{\Gamma})$), we denote by γ_n^\pm the normal trace operator on the side \pm of Γ oriented w.r.t. \mathbf{n}^\pm .

The gradient operator in the matrix domain $\Omega \setminus \bar{\Gamma}$ is denoted by ∇ and the tangential gradient operator on the fracture network is denoted by ∇_τ such that

$$\nabla_\tau u = \nabla u - (\nabla u \cdot \mathbf{n}^+) \mathbf{n}^+.$$

We also denote by div_τ the tangential divergence operator on the fracture network, and by $d\tau(\mathbf{x})$ the Lebesgue measure on Γ .

We denote by Σ the dimension 1 open set defined by the intersection of the fractures excluding the boundary of the domain Ω , i.e. the interior of $\bigcup_{\{(j,j') \in J \times J \mid j \neq j'\}} \partial\Gamma_j \cap \partial\Gamma_{j'} \setminus \partial\Omega$.

For the matrix domain, Dirichlet (subscript D) and Neumann (subscript N) boundary conditions are imposed on the two dimensional open sets $\partial\Omega_D$ and $\partial\Omega_N$ respectively where $\partial\Omega_D \cap \partial\Omega_N = \emptyset$, $\partial\Omega = \overline{\partial\Omega_D} \cup \overline{\partial\Omega_N}$. Similarly for the fracture network, the Dirichlet and Neumann boundary conditions are imposed on the one dimensional open sets $\partial\Gamma_D$ and $\partial\Gamma_N$ respectively where $\partial\Gamma_D \cap \partial\Gamma_N = \emptyset$, $\partial\Gamma \cap \partial\Omega = \overline{\partial\Gamma_D} \cup \overline{\partial\Gamma_N}$.

Let $\gamma_{n_{\partial\Gamma_j}}, j \in J$ denote the normal trace operator at the fracture Γ_j boundary oriented outward to Γ_j .

1.2. Non-isothermal single-phase flow model

To focus on the implementation aspects related to well modeling, the physics of the fluid is kept relatively simple and we refer to [42] for a compositional multiphase non-isothermal modeling of reservoir flow. The fluid is monophasic and is described by its thermodynamical variables $X = (P, T)$ where P is the pressure and T the temperature. We denote by $\rho(X)$ its mass density, by $\mu(X)$ its dynamic viscosity, by $e(X)$ its specific internal energy, and by $h(X)$ its specific enthalpy. The rock energy density is denoted by $E_r(X)$.

The reduction of dimension in the fractures leading to the hybrid-dimensional model is obtained by integration of the conservation equations along the width of the fractures complemented by transmission conditions at both sides of the matrix fracture interfaces (see [42]). In the following, $X_m = (P_m, T_m)$ denote the pressure and temperature in the matrix domain $\Omega \setminus \bar{\Gamma}$, and $X_f = (P_f, T_f)$ are the pressure and temperature in the fractures averaged along the width of the fractures. The permeability tensor is denoted by \mathbf{K}_m in the matrix domain and is assumed to be constant in the width of the fractures and to have the normal vector \mathbf{n}^+ as principal direction. We denote by \mathbf{K}_f the tangential permeability tensor in the fractures. The porosity (resp. thermal conductivity of the rock and fluid mixture) is denoted by ϕ_m (resp. λ_m) in the matrix domain. It is assumed to be constant in the width of the fractures and denoted by ϕ_f (resp. λ_f). The gravity acceleration vector is denoted by \mathbf{g} .

The set of unknowns of the hybrid-dimensional model is defined by X_m in the matrix domain $\Omega \setminus \bar{\Gamma}$, by X_f in the fracture network Γ , and by $X_\Sigma = (P_\Sigma, T_\Sigma)$ at the fracture intersection Σ . The set of equations couples the mass and energy conservation equations in the matrix

$$\begin{aligned} \phi_m \partial_t \rho(X_m) + \text{div}(\mathbf{q}_m) &= 0, \\ \phi_m \partial_t (\rho(X_m) e(X_m)) + (1 - \phi_m) \partial_t E_r(X_m) + \text{div}(\mathbf{q}_{e,m}) &= 0, \end{aligned} \quad (1)$$

in the fracture network

$$\begin{aligned} d_f \phi_f \partial_t \rho(X_f) + \text{div}_\tau(\mathbf{q}_f) - \gamma_n^+ \mathbf{q}_m - \gamma_n^- \mathbf{q}_m &= 0, \\ d_f \phi_f \partial_t (\rho(X_f) e(X_f)) + d_f (1 - \phi_f) \partial_t E_r(X_f) + \text{div}_\tau(\mathbf{q}_{e,f}) - \gamma_n^+ \mathbf{q}_{e,m} - \gamma_n^- \mathbf{q}_{e,m} &= 0, \end{aligned} \quad (2)$$

and at the fracture intersection

$$\sum_{j \in J} (\gamma_{n_{\partial\Gamma_j}} \mathbf{q}_f)|_\Sigma = 0, \quad \sum_{j \in J} (\gamma_{n_{\partial\Gamma_j}} \mathbf{q}_{e,f})|_\Sigma = 0, \quad (3)$$

as well as the Darcy and Fourier laws providing the mass and energy fluxes in the matrix

$$\mathbf{q}_m = \frac{\rho(X_m)}{\mu(X_m)} \mathbf{V}_m, \quad \mathbf{q}_{e,m} = h(X_m) \mathbf{q}_m - \lambda_m \nabla T_m, \quad (4)$$

and in the fracture network

$$\mathbf{q}_f = \frac{\rho(X_f)}{\mu(X_f)} \mathbf{V}_f, \quad \mathbf{q}_{e,f} = h(X_f) \mathbf{q}_f - d_f \lambda_f \nabla_\tau T_f, \quad (5)$$

where

$$\mathbf{V}_m = -\mathbf{K}_m (\nabla P_m - \rho(X_m) \mathbf{g}), \quad \mathbf{V}_f = -d_f \mathbf{K}_f (\nabla_\tau P_f - \rho(X_f) \mathbf{g}_\tau), \quad \mathbf{g}_\tau = \mathbf{g} - (\mathbf{g} \cdot \mathbf{n}^+) \mathbf{n}^+.$$

The system (1)-(2)-(3)-(4)-(5) is closed with the transmission conditions at the matrix fracture interface Γ . These conditions state the continuity of the pressure and temperature at the matrix fracture interfaces assuming that the fractures do not act as barrier neither for the Darcy flow nor for the thermal conductivity (see [3, 19, 28, 42]).

$$\begin{aligned} \gamma^+ P_m &= \gamma^- P_m = \gamma P_m = P_f, \\ \gamma^+ T_m &= \gamma^- T_m = \gamma T_m = T_f. \end{aligned} \quad (6)$$

Note also that the pressure P_f (resp. the temperature T_f) is assumed continuous and equal to P_Σ (resp. T_Σ) at the fracture intersection Σ , and that homogeneous Neumann boundary conditions are applied for the mass \mathbf{q}_f and energy $\mathbf{q}_{e,f}$ fluxes at the fracture tips $\partial\Gamma \setminus \partial\Omega$.

2. VAG FINITE VOLUME DISCRETIZATION

2.1. Space and time discretizations

The VAG discretization of hybrid-dimensional two-phase Darcy flows introduced in [11] considers generalized polyhedral meshes of Ω in the spirit of [16]. Let \mathcal{M} be the set of cells that are disjoint open polyhedral subsets of Ω such that $\bigcup_{K \in \mathcal{M}} \bar{K} = \bar{\Omega}$, for all $K \in \mathcal{M}$, \mathbf{x}_K denotes the so-called “center” of the cell K under the assumption that K is star-shaped with respect to \mathbf{x}_K . The set of faces of the mesh is denoted by \mathcal{F} and \mathcal{F}_K is the set of faces of the cell $K \in \mathcal{M}$. The set of edges of the mesh is denoted by \mathcal{E} and \mathcal{E}_σ is the set of edges of the face $\sigma \in \mathcal{F}$. The set of vertices of the mesh is denoted by \mathcal{V} and \mathcal{V}_σ is the set of vertices of the face σ . For each $K \in \mathcal{M}$ we define $\mathcal{V}_K = \bigcup_{\sigma \in \mathcal{F}_K} \mathcal{V}_\sigma$.

The faces are not necessarily planar. It is just assumed that for each face $\sigma \in \mathcal{F}$, there exists a so-called “center” of the face $\mathbf{x}_\sigma \in \sigma \setminus \bigcup_{e \in \mathcal{E}_\sigma} e$ such that $\mathbf{x}_\sigma = \sum_{\mathbf{s} \in \mathcal{V}_\sigma} \beta_{\sigma,\mathbf{s}} \mathbf{x}_\mathbf{s}$, with $\sum_{\mathbf{s} \in \mathcal{V}_\sigma} \beta_{\sigma,\mathbf{s}} = 1$, and $\beta_{\sigma,\mathbf{s}} \geq 0$ for all $\mathbf{s} \in \mathcal{V}_\sigma$; moreover the face σ is assumed to be defined by the union of the triangles $T_{\sigma,e}$ defined by the face center \mathbf{x}_σ and each edge $e \in \mathcal{E}_\sigma$. The mesh is also supposed to be conforming w.r.t. the fracture network Γ in the sense that for all $j \in J$ there exist the subsets \mathcal{F}_{Γ_j} of \mathcal{F} such that

$$\bar{\Gamma}_j = \bigcup_{\sigma \in \mathcal{F}_{\Gamma_j}} \bar{\sigma}.$$

We will denote by \mathcal{F}_Γ the set of fracture faces

$$\mathcal{F}_\Gamma = \bigcup_{j \in J} \mathcal{F}_{\Gamma_j},$$

and by

$$\mathcal{V}_\Gamma = \bigcup_{\sigma \in \mathcal{F}_\Gamma} \mathcal{V}_\sigma,$$

the set of fracture nodes. This geometrical discretization of Ω and Γ is denoted in the following by \mathcal{D} .

In addition, the following notations will be used

$$\mathcal{M}_\mathbf{s} = \{K \in \mathcal{M} \mid \mathbf{s} \in \mathcal{V}_K\}, \quad \mathcal{M}_\sigma = \{K \in \mathcal{M} \mid \sigma \in \mathcal{F}_K\},$$

and

$$\mathcal{F}_{\Gamma,\mathbf{s}} = \{\sigma \in \mathcal{F}_\Gamma \mid \mathbf{s} \in \mathcal{V}_\sigma\}.$$

For $N_{t_f} \in \mathbb{N}^*$, let us consider the time discretization $t^0 = 0 < t^1 < \dots < t^{N_{t_f}-1} < t^{N_{t_f}} = t_f$ of the time interval $[0, t_f]$. We denote the time steps by $\Delta t^n = t^n - t^{n-1}$ for all $n = 1, \dots, N_{t_f}$.

2.2. VAG fluxes and control volumes

The VAG discretization is introduced in [16] for diffusive problems on heterogeneous anisotropic media. Its extension to the hybrid-dimensional Darcy flow model is proposed in [11] based upon the following vector space of degrees of freedom:

$$V_{\mathcal{D}} = \{v_K \in \mathbb{R}, v_\mathbf{s} \in \mathbb{R}, v_\sigma \in \mathbb{R}, K \in \mathcal{M}, \mathbf{s} \in \mathcal{V}, \sigma \in \mathcal{F}_\Gamma\}.$$

The degrees of freedom are exhibited in Figure 2 for a given cell K with one fracture face σ in bold.

The matrix degrees of freedom are defined by the set of cells \mathcal{M} and by the set of nodes $\mathcal{V} \setminus \mathcal{V}_\Gamma$ excluding the nodes at the matrix fracture interface Γ . The fracture faces \mathcal{F}_Γ and the fracture nodes \mathcal{V}_Γ are shared between the matrix and the fractures but the control volumes associated with these degrees of freedom will belong to the fracture network (see Figure 3). The degrees of freedom at the fracture intersection Σ are defined by the set of nodes $\mathcal{V}_\Sigma \subset \mathcal{V}_\Gamma$ located on $\overline{\Sigma}$. The set of nodes at the Dirichlet boundaries $\overline{\partial\Omega_D}$ and $\overline{\partial\Gamma_D}$ is denoted by \mathcal{V}_D .

The VAG scheme is a control volume scheme in the sense that it results, for each non Dirichlet degree of freedom in a mass or energy balance equation. The matrix diffusion tensor is assumed to be cellwise constant and the tangential diffusion tensor in the fracture network is assumed to be facewise constant. The two main ingredients are therefore the conservative fluxes and the control volumes. The VAG matrix and fracture fluxes are exhibited in Figure 2. For $u_{\mathcal{D}} \in V_{\mathcal{D}}$, the matrix fluxes $F_{K,\nu}(u_{\mathcal{D}})$ connect the cell $K \in \mathcal{M}$ to the degrees of freedom located at the boundary of K , namely $\nu \in \Xi_K = \mathcal{V}_K \cup (\mathcal{F}_K \cap \mathcal{F}_\Gamma)$. The fracture fluxes $F_{\sigma,\mathbf{s}}(u_{\mathcal{D}})$ connect each fracture face $\sigma \in \mathcal{F}_\Gamma$ to its nodes $\mathbf{s} \in \mathcal{V}_\sigma$. The expression of the matrix (resp. the fracture) fluxes is linear and local to the cell (resp. fracture face). More precisely, the matrix fluxes are given by

$$F_{K,\nu}(u_{\mathcal{D}}) = \sum_{\nu' \in \Xi_K} T_K^{\nu,\nu'}(u_K - u_{\nu'}),$$

with a symmetric positive definite transmissibility matrix $T_K = (T_K^{\nu,\nu'})_{(\nu,\nu') \in \Xi_K \times \Xi_K}$ depending only on the cell K geometry (including the choices of \mathbf{x}_K and of $\mathbf{x}_\sigma, \sigma \in \mathcal{F}_K$) and on the cell matrix diffusion tensor. The fracture fluxes are given by

$$F_{\sigma,\mathbf{s}}(u_{\mathcal{D}}) = \sum_{s \in \mathcal{V}_\sigma} T_\sigma^{\mathbf{s},s'}(u_\sigma - u_{s'}),$$

with a symmetric positive definite transmissibility matrix $T_\sigma = (T_\sigma^{\mathbf{s},s'})_{(\mathbf{s},s') \in \mathcal{V}_\sigma \times \mathcal{V}_\sigma}$ depending only on the fracture face σ geometry (including the choice of \mathbf{x}_σ) and on the fracture face width and tangential diffusion tensor. Let us refer to [11] for a more detailed presentation and for the definition of T_K and T_σ .

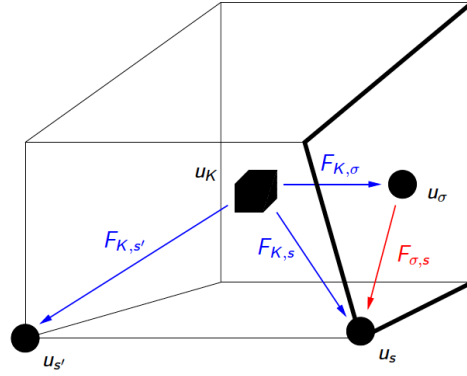


FIGURE 2. For a cell K and a fracture face σ (in bold), examples of VAG degrees of freedom u_K , u_s , u_σ , $u_{s'}$ and VAG fluxes $F_{K,\sigma}$, $F_{K,s}$, $F_{K,s'}$, $F_{\sigma,s}$.

The construction of the control volumes at each degree of freedom is based on partitions of the cells and of the fracture faces. These partitions are respectively denoted, for all $K \in \mathcal{M}$, by

$$\bar{K} = \bar{\omega}_K \cup \left(\bigcup_{s \in \mathcal{V}_K \setminus \mathcal{V}_D} \bar{\omega}_{K,s} \right),$$

and, for all $\sigma \in \mathcal{F}_\Gamma$, by

$$\bar{\sigma} = \bar{\Sigma}_\sigma \cup \left(\bigcup_{s \in \mathcal{V}_\sigma \setminus \mathcal{V}_D} \bar{\Sigma}_{\sigma,s} \right).$$

It is important to notice that in the usual case of cellwise constant rocktypes in the matrix and facewise constant rocktypes in the fracture network, the implementation of the scheme does not require to build explicitly the geometry of these partitions. In that case, it is sufficient to define the matrix volume fractions

$$\alpha_{K,s} = \frac{\int_{\omega_{K,s}} d\mathbf{x}}{\int_K d\mathbf{x}}, s \in \mathcal{V}_K \setminus (\mathcal{V}_D \cup \mathcal{V}_\Gamma), K \in \mathcal{M},$$

constrained to satisfy $\alpha_{K,\nu} \geq 0$, and $\sum_{s \in \mathcal{V}_K \setminus (\mathcal{V}_D \cup \mathcal{V}_\Gamma)} \alpha_{K,s} \leq 1$, as well as the fracture volume fractions

$$\alpha_{\sigma,s} = \frac{\int_{\Sigma_{\sigma,s}} d_f(\mathbf{x}) d\tau(\mathbf{x})}{\int_\sigma d_f(\mathbf{x}) d\tau(\mathbf{x})}, s \in \mathcal{V}_\sigma \setminus \mathcal{V}_D, \sigma \in \mathcal{F}_\Gamma,$$

constrained to satisfy $\alpha_{\sigma,s} \geq 0$, and $\sum_{s \in \mathcal{V}_\sigma \setminus \mathcal{V}_D} \alpha_{\sigma,s} \leq 1$, where we denote by $d\tau(\mathbf{x})$ the 2 dimensional Lebesgue measure on Γ . Let us also set

$$\phi_K = (1 - \sum_{s \in \mathcal{V}_K \setminus (\mathcal{V}_D \cup \mathcal{V}_\Gamma)} \alpha_{K,s}) \int_K \phi_m(\mathbf{x}) d\mathbf{x} \quad \text{for } K \in \mathcal{M},$$

and

$$\phi_\sigma = (1 - \sum_{s \in \mathcal{V}_\sigma \setminus \mathcal{V}_D} \alpha_{\sigma,s}) \int_\sigma \phi_f(\mathbf{x}) d_f(\mathbf{x}) d\tau(\mathbf{x}) \quad \text{for } \sigma \in \mathcal{F}_\Gamma,$$

as well as

$$\phi_{\mathbf{s}} = \sum_{K \in \mathcal{M}_{\mathbf{s}}} \alpha_{K,\mathbf{s}} \int_K \phi_m(\mathbf{x}) d\mathbf{x} \quad \text{for } \mathbf{s} \in \mathcal{V} \setminus (\mathcal{V}_D \cup \mathcal{V}_\Gamma),$$

and

$$\phi_{\mathbf{s}} = \sum_{\sigma \in \mathcal{F}_{\Gamma,\mathbf{s}}} \alpha_{\sigma,\mathbf{s}} \int_{\sigma} \phi_f(\mathbf{x}) d_f(\mathbf{x}) d\tau(\mathbf{x}) \quad \text{for } \mathbf{s} \in \mathcal{V}_\Gamma \setminus \mathcal{V}_D,$$

which correspond to the porous volumes distributed to the degrees of freedom excluding the Dirichlet nodes. The rock complementary volume in each control volume $\nu \in \mathcal{M} \cup \mathcal{F}_\Gamma \cup (\mathcal{V} \setminus \mathcal{V}_D)$ is denoted by $\bar{\phi}_\nu$.

As shown in [11], the flexibility in the choice of the control volumes is a crucial asset, compared with usual CVFE approaches and allows to significantly improve the accuracy of the scheme when the permeability field is highly heterogeneous. As exhibited in Figure 3, as opposed to usual CVFE approaches, this flexibility allows to define the control volumes in the fractures with no contribution from the matrix in order to avoid to artificially enlarge the flow path in the fractures.

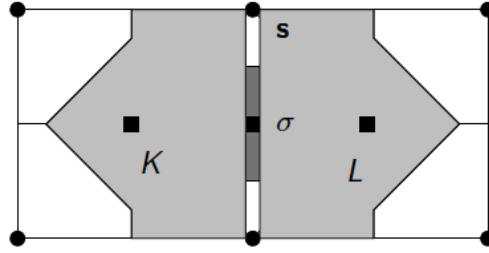


FIGURE 3. Example of control volumes at cells, fracture face, and nodes, in the case of two cells K and L splitted by one fracture face σ (the width of the fracture is enlarged in this figure). The control volumes are chosen to avoid mixing fracture and matrix rocktypes.

In the following, we will keep the notation $F_{K,s}$, $F_{K,\sigma}$, $F_{\sigma,s}$ for the VAG Darcy fluxes defined with the cellwise constant matrix permeability \mathbf{K}_m and the facewise constant fracture width d_f and tangential permeability \mathbf{K}_f . Since the rock properties are fixed, the VAG Darcy fluxes transmissibility matrices T_K and T_σ are computed only once.

The VAG Fourier fluxes are denoted in the following by $G_{K,s}$, $G_{K,\sigma}$, $G_{\sigma,s}$. They are obtained with the isotropic matrix and fracture thermal conductivities averaged in each cell and in each fracture face using the previous time step fluid properties. Hence VAG Fourier fluxes transmissibility matrices need to be recomputed at each time step.

2.3. Multi-branch non-isothermal well model

Let \mathcal{W} denote the set of wells. Each multi-branch well $\omega \in \mathcal{W}$ is defined by a set of oriented edges of the mesh assumed to define a rooted tree oriented away from the root. This orientation corresponds to the drilling direction of the well. The set of nodes of a well $\omega \in \mathcal{W}$ is denoted by $\mathcal{V}_\omega \subset \mathcal{V}$ and its root node is denoted by \mathbf{s}_ω^{root} . A partial ordering is defined on the set of vertices \mathcal{V}_ω with $\mathbf{s} <_\omega \mathbf{s}'$ if and only if the unique path from the root \mathbf{s}_ω^{root} to \mathbf{s}' passes through \mathbf{s} . The set of edges of the well ω is denoted by \mathcal{E}_ω and for each edge $\epsilon \in \mathcal{E}_\omega$ we set $\epsilon = \mathbf{s}_1 \mathbf{s}_2$ with $\mathbf{s}_1 <_\omega \mathbf{s}_2$. It is assumed that $\mathcal{E}_{\omega_1} \cap \mathcal{E}_{\omega_2} = \emptyset$ for any $\omega_1, \omega_2 \in \mathcal{W}$ such that $\omega_1 \neq \omega_2$.

We focus on the part of the well that is connected to the reservoir through open hole, production liners or perforations. In this section, exchanges with the reservoir are dominated by convection and we decided to neglect heat losses as a first step. The latest shall be taken into account when modeling the wellbore flow up to the surface. It is assumed that the radius r_ω of each well $\omega \in \mathcal{W}$ is small compared to the cell sizes in the

neighborhood of the well. It results that the Darcy flux between the reservoir and the well at a given well node $\mathbf{s} \in \mathcal{V}_\omega$ is obtained using the Two Point Flux Approximation

$$V_{\mathbf{s},\omega} = WI_{\mathbf{s},\omega}(P_{\mathbf{s}} - P_{\mathbf{s},\omega}),$$

where $P_{\mathbf{s}}$ is the reservoir pressure at node \mathbf{s} and $P_{\mathbf{s},\omega}$ is the well pressure at node \mathbf{s} . The Well Index $WI_{\mathbf{s},\omega}$ is typically computed using Peaceman's approach (see [14,33,34]) and takes into account the unresolved singularity of the pressure solution in the neighborhood of the well. Fourier fluxes between the reservoir and the well could also be discretized using such Two Point Flux Approximation but they are assumed to be small compared with thermal convective fluxes and will be neglected in the following well model. The temperature inside the well is denoted by $T_{\mathbf{s},\omega}$ at each well node $\mathbf{s} \in \mathcal{V}_\omega$.

For any $a \in \mathbb{R}$, let us define $a^+ = \max(a, 0)$ and $a^- = \min(a, 0)$. The mass flow rate between the reservoir and the well ω at a given node $\mathbf{s} \in \mathcal{V}_\omega$ is defined by the upwind formula:

$$q_{m,\mathbf{s},\omega} = \beta_\omega^{inj} \frac{\rho(X_{\mathbf{s},\omega})}{\mu(X_{\mathbf{s},\omega})} WI_{\mathbf{s},\omega}(P_{\mathbf{s}} - P_{\mathbf{s},\omega})^- + \beta_\omega^{prod} \frac{\rho(X_{\mathbf{s}})}{\mu(X_{\mathbf{s}})} WI_{\mathbf{s},\omega}(P_{\mathbf{s}} - P_{\mathbf{s},\omega})^+, \quad (7)$$

and the energy flow rate by

$$q_{e,\mathbf{s},\omega} = h(X_{\mathbf{s},\omega})q_{m,\mathbf{s},\omega}^- + h(X_{\mathbf{s}})q_{m,\mathbf{s},\omega}^+. \quad (8)$$

The well coefficients β_ω^{inj} and β_ω^{prod} are used to impose specific well behavior. The general case corresponds to $\beta_\omega^{inj} = \beta_\omega^{prod} = 1$. Yet, for an injection well, it will be convenient as explained in subsection 2.3.2, to impose that the mass flow rates $q_{m,\mathbf{s},\omega}$ are non positive for all nodes $\mathbf{s} \in \mathcal{V}_\omega$ corresponding to set $\beta_\omega^{inj} = 1$ and $\beta_\omega^{prod} = 0$. Likewise, for a production well, it will be convenient as explained in subsection 2.3.3, to set $\beta_\omega^{inj} = 0$ and $\beta_\omega^{prod} = 1$ which corresponds to assume that the mass flow rates $q_{m,\mathbf{s},\omega}$ are non negative for all nodes $\mathbf{s} \in \mathcal{V}_\omega$. These simplifying options currently prevent the modeling of cross flows where injection and production occur in different places of the same well, as it sometimes happen in geothermal wells, typically in closed wells.

2.3.1. Well model

Our conceptual model inside the well assumes that the flow is stationary at the reservoir time scale along with perfect mixing and thermal equilibrium. The pressure distribution along the well is also assumed hydrostatic.

For the sake of simplicity, the flow rate between the reservoir and the well is considered concentrated at each node \mathbf{s} of the well. Hence the mass flow rate along each edge $\epsilon \in \mathcal{E}_\omega$ depends only on time. It is denoted by q_ϵ and is oriented positively from \mathbf{s}_1 to \mathbf{s}_2 with $\epsilon = \mathbf{s}_1\mathbf{s}_2$. Since Fourier fluxes are neglected, the specific enthalpy depends as well only on time along the edge ϵ and is denoted by h_ϵ .

The set of well unknowns is defined by the well pressure $P_{\mathbf{s},\omega}$ and the well temperature $T_{\mathbf{s},\omega}$ at each node $\mathbf{s} \in \mathcal{V}_\omega$, the mass flow rate q_ϵ and specific enthalpy h_ϵ at each edge $\epsilon \in \mathcal{E}_\omega$, the well total mass flow rate q_ω (non negative for production wells and non positive for injection wells) as well as the well specific enthalpy h_ω for injection wells.

For each edge $\epsilon = \mathbf{s}_1\mathbf{s}_2 \in \mathcal{E}_\omega$, the specific enthalpy h_ϵ satisfies the equation

$$h_\epsilon = \begin{cases} h(X_{\mathbf{s}_1,\omega}) & \text{if } q_\epsilon \geq 0, \\ h(X_{\mathbf{s}_2,\omega}) & \text{if } q_\epsilon < 0. \end{cases} \quad (9)$$

For all $\mathbf{s}_1\mathbf{s}_2 = \epsilon \in \mathcal{E}_\omega$, let us set $\kappa_{\epsilon,\mathbf{s}} = 1$ if $\mathbf{s} = \mathbf{s}_2$ and $\kappa_{\epsilon,\mathbf{s}} = -1$ if $\mathbf{s} = \mathbf{s}_1$. The well equations account for the mass and energy conservations at each node of the well. Let $\mathcal{E}_\mathbf{s}$ denote the set of edges sharing the node \mathbf{s} , then

for all $\mathbf{s} \in \mathcal{V}_\omega$ we obtain the equations

$$\left\{ \begin{array}{l} \sum_{\epsilon \in \mathcal{E}_\mathbf{s} \cap \mathcal{E}_\omega} \kappa_{\epsilon, \mathbf{s}} q_\epsilon + q_{m, \mathbf{s}, \omega} = \delta_{\mathbf{s}}^{\mathbf{s}^{root}} q_\omega, \\ \sum_{\epsilon \in \mathcal{E}_\mathbf{s} \cap \mathcal{E}_\omega} \kappa_{\epsilon, \mathbf{s}} h_\epsilon q_\epsilon + q_{e, \mathbf{s}, \omega} = \delta_{\mathbf{s}}^{\mathbf{s}^{root}} (h_\omega q_\omega^- + h(X_{\mathbf{s}, \omega}) q_\omega^+), \end{array} \right. \quad (10)$$

where δ stands for the Kronecker symbol. Inside the well, the hypothesis of hydrostatic pressure distribution implies that

$$\left\{ \begin{array}{l} \frac{\partial P}{\partial z} = -g\rho(X) \text{ on } \epsilon = \mathbf{s}_1 \mathbf{s}_2, \\ h(X) = h_\epsilon, \\ P(z_{\mathbf{s}_1}) = P_{\mathbf{s}_1, \omega}, \\ P(z_{\mathbf{s}_2}) = P_{\mathbf{s}_2, \omega}, \end{array} \right. \quad (11)$$

for each edge $\epsilon \in \mathcal{E}_\omega$. To close the system, the well boundary conditions prescribe a limit total mass flow rate \bar{q}_ω and a limit bottom hole pressure \bar{P}_ω . Then, complementary constraints accounting for usual well monitoring conditions, are imposed between $q_\omega - \bar{q}_\omega$ and $P_\omega - \bar{P}_\omega$ with $P_\omega = P_{\mathbf{s}^{root}, \omega}$. In addition, the well specific enthalpy h_ω is also prescribed for an injection well.

In the following subsections, we consider the particular cases of an injection well and a production well. The flow rates are enforced to be non positive (resp. non negative) at all well nodes for injection wells (resp. production wells). It corresponds to set $\beta_\omega^{inj} = 1$, $\beta_\omega^{prod} = 0$ for an injection well and $\beta_\omega^{inj} = 0$, $\beta_\omega^{prod} = 1$ for a production well. The limit bottom hole pressure \bar{P}_ω is a maximum (resp. minimum) pressure and the limit total mass flow rate \bar{q}_ω is a minimum non positive (resp. maximum non negative) flow rate for injection (resp. production) wells. In both cases, using an explicit computation of the hydrostatic pressure drop, the well model can be reduced to a single equation and a single implicit unknown corresponding to the well reference pressure P_ω (see *e.g.* [6]).

2.3.2. Injection wells

The injection well model sets $\beta_\omega^{inj} = 1$, $\beta_\omega^{prod} = 0$ and prescribes the minimum well total mass flow rate $\bar{q}_\omega \leq 0$, the well maximum bottom hole pressure \bar{P}_ω and the well specific enthalpy h_ω .

Since $\beta_\omega^{inj} = 1$ and $\beta_\omega^{prod} = 0$, the mass flow rates q_ϵ are enforced to be non negative and it results from (10) that $h_\epsilon = h_\omega$ for all $\epsilon \in \mathcal{E}_\omega$.

To compute the pressures along the well, we first solve numerically the equations (11) using the well reference pressure $P_\omega^{n-1} = P_{\mathbf{s}^{root}, \omega}^{n-1}$ obtained at the previous time step $n-1$ and $h_\epsilon = h_\omega$. It provides the pressure drop $\Delta P_{\mathbf{s}, \omega}^{n-1} = P(z_{\mathbf{s}}) - P_\omega^{n-1}$ at each node $\mathbf{s} \in \mathcal{V}_\omega$, from which we deduce the well pressures using the bottom well pressure at the current time step n

$$P_{\mathbf{s}, \omega}^n = P_\omega^n + \Delta P_{\mathbf{s}, \omega}^{n-1}.$$

From the equation $h(X_{\mathbf{s}, \omega}^n) = h(P_{\mathbf{s}, \omega}^n, T_{\mathbf{s}, \omega}^n) = h_\omega$, the well temperature $T_{\mathbf{s}, \omega}^n$ at each node $\mathbf{s} \in \mathcal{V}_\omega$ depends only on the implicit unknown P_ω^n . The mass and energy flow rates at each node $\mathbf{s} \in \mathcal{V}_\omega$ between the reservoir and the well are defined by (7)-(8) with $\beta_\omega^{inj} = 1$ and $\beta_\omega^{prod} = 0$ and depend only on the implicit unknowns $X_\mathbf{s}^n$ and P_ω^n :

$$q_{m, \mathbf{s}, \omega}(X_\mathbf{s}^n, P_\omega^n) = \frac{\rho(X_{\mathbf{s}, \omega}^n)}{\mu(X_{\mathbf{s}, \omega}^n)} W_{\mathbf{s}, \omega}(P_\mathbf{s}^n - P_{\mathbf{s}, \omega}^n)^-, \quad q_{e, \mathbf{s}, \omega}(X_\mathbf{s}^n, P_\omega^n) = h_\omega q_{m, \mathbf{s}, \omega}(X_\mathbf{s}^n, P_\omega^n).$$

The well equation at the current time step is defined by the following complementary constraints between the prescribed minimum well total mass flow rate and the prescribed maximum bottom hole pressure

$$\left(\sum_{\mathbf{s} \in \mathcal{V}_\omega} q_{m, \mathbf{s}, \omega}(X_\mathbf{s}^n, P_\omega^n) - \bar{q}_\omega \right) (\bar{P}_\omega - P_\omega^n) = 0, \quad \sum_{\mathbf{s} \in \mathcal{V}_\omega} q_{m, \mathbf{s}, \omega}(X_\mathbf{s}^n, P_\omega^n) - \bar{q}_\omega \geq 0, \quad \bar{P}_\omega - P_\omega^n \geq 0. \quad (12)$$

2.3.3. Production wells

The production well model sets $\beta_\omega^{inj} = 0$, $\beta_\omega^{prod} = 1$ and prescribes the maximum well total mass flow rate $\bar{q}_\omega \geq 0$ and the well minimum bottom hole pressure \bar{P}_ω .

The solution at the previous time step $n - 1$ provides the pressure drop $\Delta P_{s,\omega}^{n-1}$ at each node $\mathbf{s} \in \mathcal{V}_\omega$. This computation is detailed below. As for the injection well, we deduce the well pressures using the bottom well pressure at the current time step n

$$P_{s,\omega}^n = P_\omega^n + \Delta P_{s,\omega}^{n-1}.$$

The mass and energy flow rates at each node $\mathbf{s} \in \mathcal{V}_\omega$ between the reservoir and the well are defined by (7)-(8) with $\beta_\omega^{inj} = 0$ and $\beta_\omega^{prod} = 1$ and depend only on the implicit unknowns $X_\mathbf{s}^n$ and P_ω^n :

$$q_{m,s,\omega}(X_\mathbf{s}^n, P_\omega^n) = \frac{\rho(X_\mathbf{s}^n)}{\mu(X_\mathbf{s}^n)} W_{s,\omega}(P_\mathbf{s}^n - P_{s,\omega}^n)^+, \quad q_{e,s,\omega}(X_\mathbf{s}^n, P_\omega^n) = h(X_\mathbf{s}^n) q_{m,s,\omega}(X_\mathbf{s}^n, P_\omega^n).$$

The well equation at the current time step is defined by the following complementary constraints between the prescribed maximum well total mass flow rate and the prescribed minimum bottom hole pressure

$$\left(\bar{q}_\omega - \sum_{\mathbf{s} \in \mathcal{V}_\omega} q_{m,s,\omega}(X_\mathbf{s}^n, P_\omega^n) \right) (P_\omega^n - \bar{P}_\omega) = 0, \quad \bar{q}_\omega - \sum_{\mathbf{s} \in \mathcal{V}_\omega} q_{m,s,\omega}(X_\mathbf{s}^n, P_\omega^n) \geq 0, \quad P_\omega^n - \bar{P}_\omega \geq 0. \quad (13)$$

Let us now detail the computation of the pressure drop at each node $\mathbf{s} \in \mathcal{V}_\omega$ using the previous time step solution $n - 1$. We first compute the well temperature $T_{s,\omega}^{n-1}$ at each node \mathbf{s} using equations (10). The mass and energy flow rates from the upstream nodes of the node \mathbf{s} are given by

$$Q_{m,s,\omega} = \sum_{\mathbf{s}' \in \mathcal{V}_\omega | \mathbf{s}' \geq \mathbf{s}} q_{m,s',\omega}(X_{\mathbf{s}'}^{n-1}, P_\omega^{n-1}), \quad Q_{e,s,\omega} = \sum_{\mathbf{s}' \in \mathcal{V}_\omega | \mathbf{s}' \geq \mathbf{s}} q_{e,s',\omega}(X_{\mathbf{s}'}^{n-1}, P_\omega^{n-1}).$$

The temperature $T_{s,\omega}^{n-1}$ inside the well at node \mathbf{s} is the solution of the nonlinear system

$$h(X_{s,\omega}^{n-1}) = \frac{Q_{e,s,\omega}}{Q_{m,s,\omega}}$$

from which we deduce the mass density $\rho_{s,\omega}^{n-1} = \rho(X_{s,\omega}^{n-1})$ inside the well at node \mathbf{s} . These mass densities and the reference pressure P_ω^{n-1} are then used to compute the hydrostatic pressure drop $\Delta P_{s,\omega}^{n-1}$ for each node $\mathbf{s} \in \mathcal{V}_\omega$ using equations (11).

2.4. Discretization of the hybrid-dimensional non-isothermal single-phase flow model

The time integration is based on a fully implicit Euler scheme to avoid severe restrictions on the time steps due to the small volumes and high velocities in the fractures. An upwind scheme is used for the approximation of the mobilities in the mass and energy fluxes that is to say the same scheme that is already used in the computation of the well mass and energy fluxes (see e.g. [7]). At the matrix fracture interfaces, we avoid mixing matrix and fracture rocktypes by choosing appropriate control volumes for $\sigma \in \mathcal{F}_\Gamma$ and $\mathbf{s} \in \mathcal{V}_\Gamma$ (see Figure 3). In order to avoid tiny control volumes at the nodes $\mathbf{s} \in \mathcal{V}_\Sigma$ located at the fracture intersection, the volume is distributed to such a node \mathbf{s} from all the fracture faces containing the node \mathbf{s} . It results that the volumes of the control volumes $\mathbf{s} \in \mathcal{V}_\Sigma$ at the fracture intersection are not smaller than at any other matrix fracture degrees of freedom. This solves the problems reported in [23] and [36] related to the small volumes at the fracture intersections and avoids the Star-Delta transformation used in [23] which is not valid when coupled with a transport model.

For each $\nu \in \mathcal{M} \cup \mathcal{F}_\Gamma \cup \mathcal{V}$ the couple of reservoir pressure and temperature is denoted by $X_\nu = (P_\nu, T_\nu)$. We denote by $X_\mathcal{D}$, the set of reservoir unknowns

$$X_\mathcal{D} = \{X_\nu, \nu \in \mathcal{M} \cup \mathcal{F}_\Gamma \cup \mathcal{V}\},$$

and similarly by $P_\mathcal{D}$ and $T_\mathcal{D}$ the sets of reservoir pressures and temperatures. The set of bottom hole pressures is denoted by $P_\mathcal{W} = \{P_\omega, \omega \in \mathcal{W}\}$.

The Darcy fluxes taking into account the gravity term are defined by

$$\begin{cases} V_{K,\nu}(X_\mathcal{D}) = F_{K,\nu}(P_\mathcal{D}) + \frac{\rho(X_K) + \rho(X_\nu)}{2} F_{K,\nu}(\mathcal{G}_\mathcal{D}), & \nu \in \Xi_K, K \in \mathcal{M}, \\ V_{\sigma,s}(X_\mathcal{D}) = F_{\sigma,s}(P_\mathcal{D}) + \frac{\rho(X_\sigma) + \rho(X_s)}{2} F_{\sigma,s}(\mathcal{G}_\mathcal{D}), & s \in \mathcal{V}_\sigma, \sigma \in \mathcal{F}_\Gamma, \end{cases} \quad (14)$$

where $\mathcal{G}_\mathcal{D}$ denotes the vector $(\mathbf{g} \cdot \mathbf{x}_\nu)_{\nu \in \mathcal{M} \cup \mathcal{F}_\Gamma \cup \mathcal{V}}$.

For each Darcy flux, let us define the upwind control volume $cv_{\mu,\nu}$ such that

$$cv_{K,\nu} = \begin{cases} K & \text{if } V_{K,\nu}(X_\mathcal{D}) \geq 0 \\ \nu & \text{if } V_{K,\nu}(X_\mathcal{D}) < 0 \end{cases} \quad \text{for } K \in \mathcal{M}, \nu \in \Xi_K,$$

for the matrix fluxes, and such that

$$cv_{\sigma,s} = \begin{cases} \sigma & \text{if } V_{\sigma,s}(X_\mathcal{D}) \geq 0 \\ s & \text{if } V_{\sigma,s}(X_\mathcal{D}) < 0 \end{cases} \quad \text{for } \sigma \in \mathcal{F}_\Gamma, s \in \mathcal{V}_\sigma,$$

for fracture fluxes. Using this upwinding, the mass and energy fluxes are given by

$$q_{m,\nu,\nu'}(X_\mathcal{D}) = \frac{\rho(X_{cv_{\nu,\nu'}})}{\mu(X_{cv_{\nu,\nu'}})} V_{\nu,\nu'}(X_\mathcal{D}), \quad q_{e,\nu,\nu'}(X_\mathcal{D}) = h(X_{cv_{\nu,\nu'}}) q_{m,\nu,\nu'}(X_\mathcal{D}) + G_{\nu,\nu'}(T_\mathcal{D}).$$

In each control volume ν , the mass and energy accumulations are denoted by

$$\mathcal{A}_{m,\nu}(X_\nu) = \phi_\nu \rho(X_\nu), \quad \mathcal{A}_{e,\nu}(X_\nu) = e(X_\nu) \mathcal{A}_{m,\nu}(X_\nu) + \bar{\phi}_\nu E_r(X_\nu).$$

We can now state the system of discrete equations at each time step $n = 1, \dots, N_{t_f}$ which accounts for the mass ($i = m$) and energy ($i = e$) conservation equations in each cell $K \in \mathcal{M}$

$$R_{K,i}(X_\mathcal{D}^n) := \frac{\mathcal{A}_{i,K}(X_K^n) - \mathcal{A}_{i,K}(X_K^{n-1})}{\Delta t^n} + \sum_{s \in \mathcal{V}_K} q_{i,K,s}(X_\mathcal{D}^n) + \sum_{\sigma \in \mathcal{F}_\Gamma \cap \mathcal{F}_K} q_{i,K,\sigma}(X_\mathcal{D}^n) = 0, \quad (15)$$

in each fracture face $\sigma \in \mathcal{F}_\Gamma$

$$R_{\sigma,i}(X_\mathcal{D}^n) := \frac{\mathcal{A}_{i,\sigma}(X_\sigma^n) - \mathcal{A}_{i,\sigma}(X_\sigma^{n-1})}{\Delta t^n} + \sum_{s \in \mathcal{V}_\sigma} q_{i,\sigma,s}(X_\mathcal{D}^n) + \sum_{K \in \mathcal{M}_\sigma} -q_{i,K,\sigma}(X_\mathcal{D}^n) = 0, \quad (16)$$

and at each node $\mathbf{s} \in \mathcal{V} \setminus \mathcal{V}_\mathcal{D}$

$$R_{\mathbf{s},i}(X_\mathcal{D}^n, P_\mathcal{W}^n) := \frac{\mathcal{A}_{i,\mathbf{s}}(X_\mathbf{s}^n) - \mathcal{A}_{i,\mathbf{s}}(X_\mathbf{s}^{n-1})}{\Delta t^n} + \sum_{\sigma \in \mathcal{F}_\Gamma, \mathbf{s}} -q_{i,\sigma,\mathbf{s}}(X_\mathcal{D}^n) + \sum_{K \in \mathcal{M}_\mathbf{s}} -q_{i,K,\mathbf{s}}(X_\mathcal{D}^n) + \sum_{\omega \in \mathcal{W} | \mathbf{s} \in \mathcal{V}_\omega} q_{i,\mathbf{s},\omega}(X_\mathbf{s}^n, P_\omega^n) = 0. \quad (17)$$

It is coupled with the well equations for the injection wells $\omega \in \mathcal{W}_{inj}$

$$R_\omega(X_{\mathcal{D}}^n, P_{\mathcal{W}}^n) := -\min\left(\sum_{\mathbf{s} \in \mathcal{V}_\omega} q_{m,\mathbf{s},\omega}(X_{\mathbf{s}}^n, P_\omega^n) - \bar{q}_\omega, \bar{P}_\omega - P_\omega^n\right) = 0, \quad (18)$$

and for the production wells $\omega \in \mathcal{W}_{prod}$

$$R_\omega(X_{\mathcal{D}}^n, P_{\mathcal{W}}^n) := \min(\bar{q}_\omega - \sum_{\mathbf{s} \in \mathcal{V}_\omega} q_{m,\mathbf{s},\omega}(X_{\mathbf{s}}^n, P_\omega^n), P_\omega^n - \bar{P}_\omega) = 0, \quad (19)$$

reformulating respectively (12) and (13) using the min function.

The system is closed with the Dirichlet boundary conditions

$$X_{\mathbf{s}}^n = X_{\mathbf{s},D},$$

for all $\mathbf{s} \in \mathcal{V}_D$, where $X_{\mathbf{s},D} = (P_{\mathbf{s},D}, T_{\mathbf{s},D})$ are the imposed pressure and temperature at node \mathbf{s} .

Let us denote by R_ν the vector $\left(R_{\nu,i}, i \in \{m, e\}\right)$, and let us rewrite the conservation equations (15), (16), (17), (18), (19) as well as the Dirichlet boundary conditions in vector form defining the following nonlinear system at each time step $n = 1, 2, \dots, N_{t_f}$

$$\mathbf{0} = \mathcal{R}(X_{\mathcal{D}}, P_{\mathcal{W}}) := \begin{cases} R_{\mathbf{s}}(X_{\mathcal{D}}, P_{\mathcal{W}}), & \mathbf{s} \in \mathcal{V}, \\ R_\sigma(X_{\mathcal{D}}), & \sigma \in \mathcal{F}_T, \\ R_K(X_{\mathcal{D}}), & K \in \mathcal{M}, \\ R_\omega(X_{\mathcal{D}}, P_{\mathcal{W}}), & \omega \in \mathcal{W}, \end{cases} \quad (20)$$

where the superscript n is dropped to simplify the notations and where the Dirichlet boundary conditions have been included at each Dirichlet node $\mathbf{s} \in \mathcal{V}_D$ in order to obtain a system size independent of the boundary conditions.

The nonlinear system $\mathcal{R}(X_{\mathcal{D}}, P_{\mathcal{W}}) = 0$ is solved by a Newton-min algorithm [26]. Our implementation is based on an active set method which enforces either the total mass flow rate or the bottom hole pressure at each Newton iterate and use the remaining inequality constraint to switch from prescribed total mass flow rate to prescribed bottom hole pressure and vice versa.

3. PARALLEL IMPLEMENTATION

In this section, the extension to our well model of the original parallel implementation described in [42] is detailed. The distribution of wells to each MPI process p is such that any well with a node belonging to the set of own nodes of p belongs to the set of own and ghost wells of p (see subsection 3.1). Then, the set of own and ghost nodes of p is extended to include all the nodes belonging to the own and ghost wells of p . These definitions ensure that (i) the local linearized systems can be assembled locally on each process without communication as in [42], and (ii) the pressure drops of the wells can be computed locally on each process without communication. This last property is convenient since the pressure drop is a sequential computation along the well rooted tree. This parallelization strategy of the well model is based on the assumption that the number of additional ghost nodes resulting from the connectivity of the wells remains very small compared with the number of own nodes of the process.

In subsection 3.2, the new structure of the local linearized systems is described as well as the local elimination of the cell unknowns. Then, the modification of the pressure block in the CPR-AMG preconditioner and the extension of the synchronization to the ghost well unknowns are addressed.

3.1. Mesh decomposition

Let us denote by N_p the number of MPI processes. The set of cells \mathcal{M} is partitioned into N_p subsets $\mathcal{M}^p, p = 1, \dots, N_p$ using the library METIS [24]. The non overlapping partitioning of the set of nodes \mathcal{V} , of the set of fracture faces \mathcal{F}_Γ , and of the set of wells \mathcal{W} is defined as follows: assuming we have defined a global index of the cells $K \in \mathcal{M}$ let us denote by $K(\mathbf{s}), \mathbf{s} \in \mathcal{V}$ (resp. $K(\sigma), \sigma \in \mathcal{F}_\Gamma$) the cell with the smallest global index among those of $\mathcal{M}_\mathbf{s}$ (resp. \mathcal{M}_σ). Then we set

$$\mathcal{V}^p = \{\mathbf{s} \in \mathcal{V} \mid K(\mathbf{s}) \in \mathcal{M}^p\}, \quad \mathcal{F}_\Gamma^p = \{\sigma \in \mathcal{F}_\Gamma \mid K(\sigma) \in \mathcal{M}^p\},$$

and

$$\mathcal{W}^p = \{\omega \in \mathcal{W} \mid \mathbf{s}_\omega^{root} \in \mathcal{V}^p\}.$$

These sets of own elements are then extended with ghost elements as follows. The overlapping decomposition of \mathcal{M} into the sets

$$\overline{\mathcal{M}}^p, \quad p = 1, \dots, N_p,$$

is chosen in such a way that any compact finite volume scheme such as the VAG scheme can be assembled locally on each process. Hence, as exhibited in Figure 4, $\overline{\mathcal{M}}^p$ is defined as the set of cells sharing a node with a cell of \mathcal{M}^p . The overlapping decompositions of the set of wells, of the set of nodes and of the set of fracture faces for $p = 1, \dots, N_p$ are performed in such a way that the linearized systems can be assembled locally on each process and that the pressure drops of the own and ghost wells can be computed locally on each process p . It results that any well with a node belonging to \mathcal{V}^p is included in the set $\overline{\mathcal{W}}^p$ of own and ghost wells of the process p . Then, the set of own and ghost nodes $\overline{\mathcal{V}}^p$ is extended compared to the definition of [42] in such a way that any node of a well in $\overline{\mathcal{W}}^p$ belongs to $\overline{\mathcal{V}}^p$. The definition of the set of own and ghost fracture faces is unchanged compared with its original definition in [42]. This leads to the following definitions:

$$\overline{\mathcal{W}}^p = \{\omega \in \mathcal{W} \mid \mathcal{V}_\omega \cap \mathcal{V}^p \neq \emptyset\}, \quad \overline{\mathcal{V}}^p = \mathcal{V}_{\overline{\mathcal{M}}^p} \cup \mathcal{V}_{\overline{\mathcal{W}}^p}, \quad \overline{\mathcal{F}}_\Gamma^p = \bigcup_{K \in \overline{\mathcal{M}}^p} \mathcal{F}_K \cap \mathcal{F}_\Gamma,$$

where

$$\mathcal{V}_{\overline{\mathcal{M}}^p} := \bigcup_{K \in \overline{\mathcal{M}}^p} \mathcal{V}_K, \quad \mathcal{V}_{\overline{\mathcal{W}}^p} := \bigcup_{\omega \in \overline{\mathcal{W}}^p} \mathcal{V}_\omega.$$

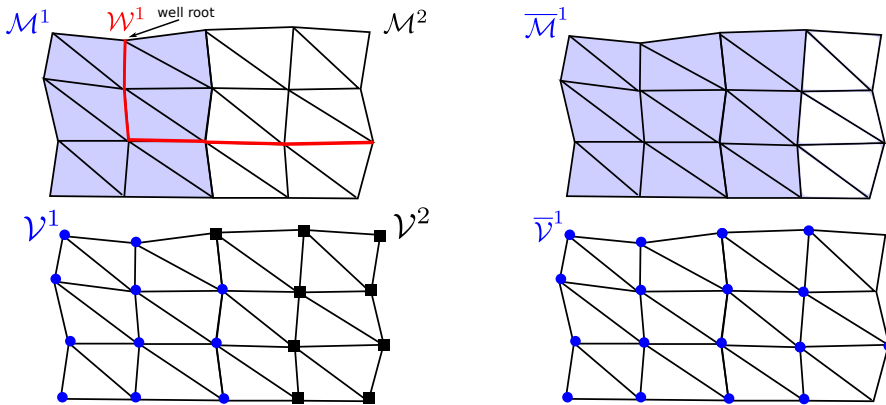


FIGURE 4. Example of mesh decomposition.

The partitioning of the mesh is performed by the master process (process 1), and then, each local mesh is distributed to its process. Therefore, each MPI process contains the local mesh $(\overline{\mathcal{M}}^p, \overline{\mathcal{V}}^p, \overline{\mathcal{F}}_\Gamma^p, \overline{\mathcal{W}}^p)$, $p =$

1, 2, ..., N_p which is splitted into two parts:

$$\begin{aligned} \text{own elements: } & (\mathcal{M}^p, \mathcal{V}^p, \mathcal{F}_\Gamma^p, \mathcal{W}^p), \\ \text{ghost elements: } & (\overline{\mathcal{M}}^p \setminus \mathcal{M}^p, \overline{\mathcal{V}}^p \setminus \mathcal{V}^p, \overline{\mathcal{F}}_\Gamma^p \setminus \mathcal{F}_\Gamma^p, \overline{\mathcal{W}}^p \setminus \mathcal{W}^p). \end{aligned}$$

We now turn to the parallel implementation of the Jacobian system to be solved at each Newton iteration of each time step.

3.2. Parallelization of the Jacobian system

The Jacobian of the nonlinear system (20) is assembled locally on each process $p = 1, \dots, N_p$ resulting in the following rectangular linear system

$$\begin{pmatrix} J_{ss}^p & J_{sf}^p & J_{sc}^p & J_{sw}^p \\ J_{fs}^p & J_{ff}^p & J_{fc}^p & 0 \\ J_{cs}^p & J_{cf}^p & J_{cc}^p & 0 \\ J_{ws}^p & 0 & 0 & J_{ww}^p \end{pmatrix} \begin{pmatrix} \overline{U}_s^p \\ \overline{U}_f^p \\ \overline{U}_c^p \\ \overline{U}_w^p \end{pmatrix} = \begin{pmatrix} b_s^p \\ b_f^p \\ b_c^p \\ b_w^p \end{pmatrix}. \quad (21)$$

In (21), $\overline{U}_s^p \in \mathbb{R}^{(2\#\overline{\mathcal{V}}^p)}$, $\overline{U}_f^p \in \mathbb{R}^{(2\#\overline{\mathcal{F}}_\Gamma^p)}$, $\overline{U}_c^p \in \mathbb{R}^{(2\#\overline{\mathcal{M}}^p)}$ denote the vectors of pressure and temperature unknowns at nodes $\mathbf{s} \in \overline{\mathcal{V}}^p$, fracture faces $\sigma \in \overline{\mathcal{F}}_\Gamma^p$, and cells $K \in \overline{\mathcal{M}}^p$. The vector $\overline{U}_w^p \in \mathbb{R}^{(\#\overline{\mathcal{W}}^p)}$ is the vector of well reference pressures. Likewise, $b_s^p \in \mathbb{R}^{(2\#\mathcal{V}^p)}$ and $b_f^p \in \mathbb{R}^{(2\#\mathcal{F}_\Gamma^p)}$ are the right hand side vectors of own node and fracture faces equations, $b_c^p \in \mathbb{R}^{(2\#\mathcal{M}^p)}$ is the right hand side vector of own and ghost cell equations, and $b_w^p \in \mathbb{R}^{(\#\mathcal{W}^p)}$ is the right hand side vector of own well equations.

The matrix J_{cc}^p is a non singular block diagonal matrix with 2×2 blocks, and the cell unknowns can be easily eliminated without fill-in leading to the following Schur complement system

$$J^p \overline{U}^p := \left(\begin{array}{c|c} J_{\text{schur}}^p & J_{sw}^p \\ \hline J_{ws}^p & 0 \end{array} \right) \begin{pmatrix} \overline{U}_s^p \\ \overline{U}_f^p \\ \overline{U}_w^p \end{pmatrix} = \begin{pmatrix} b_{\text{schur}}^p \\ b_w^p \end{pmatrix} \quad (22)$$

with

$$J_{\text{schur}}^p := \begin{pmatrix} J_{ss}^p & J_{sf}^p \\ J_{fs}^p & J_{ff}^p \end{pmatrix} - \begin{pmatrix} J_{sc}^p \\ J_{fc}^p \end{pmatrix} (J_{cc}^p)^{-1} \begin{pmatrix} J_{cs}^p & J_{cf}^p \end{pmatrix}, \quad b_{\text{schur}}^p := \begin{pmatrix} b_s^p \\ b_f^p \end{pmatrix} - \begin{pmatrix} J_{sc}^p \\ J_{fc}^p \end{pmatrix} (J_{cc}^p)^{-1} b_c^p,$$

and

$$\overline{U}_c^p = (J_{cc}^p)^{-1} (b_c^p - J_{cs}^p \overline{U}_s^p - J_{cf}^p \overline{U}_f^p). \quad (23)$$

The linear system (22) is built locally on each process p and transferred to the parallel linear solver library PETSc [8]. The parallel matrix and the parallel vector in PETSc are stored in a distributed manner, i.e. each process stores its own rows. We construct the following parallel linear system

$$JU = b, \quad (24)$$

with

$$J := \begin{pmatrix} J^1 R^1 \\ J^2 R^2 \\ \vdots \\ J^{N_p} R^{N_p} \end{pmatrix} \begin{array}{l} \} \text{process 1} \\ \} \text{process 2} \\ \vdots \\ \} \text{process } N_p \end{array},$$

and

$$U := \begin{pmatrix} U^1 \\ U^2 \\ \vdots \end{pmatrix} \left. \begin{array}{l} \} \text{process 1} \\ \} \text{process 2} \\ \vdots \end{array} \right\}, \quad U^p := \begin{pmatrix} U_s^p \\ U_f^p \\ U_w^p \end{pmatrix}, \quad b := \begin{pmatrix} b_{schur}^1 \\ b_w^1 \\ b_{schur}^2 \\ b_w^2 \\ \vdots \end{pmatrix} \left. \begin{array}{l} \} \text{process 1} \\ \} \text{process 2} \\ \vdots \end{array} \right\}$$

where $R^p, p = 1, 2, \dots, N_p$ is a restriction matrix satisfying

$$R^p U = \bar{U}^p.$$

The matrix $J^p R^p$, the vector \bar{U}^p and the vector $\begin{pmatrix} b_{schur}^p \\ b_w^p \end{pmatrix}$ are stored in process p .

The linear system (24) is solved using the GMRES iterative solver preconditioned by a CPR-AMG preconditioner introduced in [27, 37]. This preconditioner combines multiplicatively a parallel algebraic multigrid preconditioner (AMG) [21] for a pressure block of the linear system with a block Jacobi ILU0 preconditioner for the full system. In our case, the columns of the pressure block are defined by the node, the fracture face and the well pressure unknowns, and its lines by the node and the fracture face mass conservation equations as well as the well equations.

The solution of the linear system provides on each process p the solution vector (U_s^p, U_f^p, U_w^p) of own node, fracture-face and well unknowns. Then, the ghost node unknowns $U_\nu^p, \nu \in (\bar{\mathcal{V}}^p \setminus \mathcal{V}^p)$, the ghost fracture face unknowns $U_\nu^p, \nu \in (\bar{\mathcal{F}}_\Gamma^p \setminus \mathcal{F}_\Gamma^p)$ and the ghost well unknowns $U_\nu^p, \nu \in (\bar{\mathcal{W}}^p \setminus \mathcal{W}^p)$ are recovered by a synchronization step with MPI communications. This synchronization is efficiently implemented using a PETSc matrix vector product

$$\bar{U} = S U \tag{25}$$

where

$$\bar{U} := \begin{pmatrix} \bar{U}^1 \\ \bar{U}^2 \\ \vdots \end{pmatrix}$$

is the vector of own and ghost node, fracture-face and well unknowns on all processes. The matrix S , containing only 0 and 1 entries, is assembled once and for all at the beginning of the simulation.

Finally, thanks to (23), the vector of own and ghost cell unknowns \bar{U}_c^p is computed locally on each process p .

4. NUMERICAL RESULTS

All the numerical tests have been implemented on the Cicada cluster of the University Nice Sophia Antipolis composed of 72 nodes (16 cores/node, Intel Sandy Bridge E5-2670, 64GB/node). We always fix 1 core per process and 16 processes per node. The communications are handled by OpenMPI 1.8.2 (GCC 4.9) and PETSc 3.5.3. Gravity acceleration is set to 0 in all of the test cases.

4.1. Numerical convergence for an analytical solution with one horizontal fracture and a vertical well

We consider the domain $\Omega = (-H, H)^3, H = 1000$ m, with one horizontal fracture $\Gamma = \{(x, y, z) \in \Omega \mid z = 0\}$ of width $d_f = 0.5$ m and one vertical well of radius $r_w = 0.1$ m and defined by the line $\{(x, y, z) \in \Omega \mid x = y = 0\}$. Both the matrix and the fracture are isotropic and homogeneous with permeability $\mathbf{K}_m = k_m I, k_m = 10^{-14} \text{ m}^2$ in the matrix and with tangential permeability $\mathbf{K}_f = k_f I, k_f = 10^{-11} \text{ m}^2$ in the fracture. For such a simple

geometry, an analytical solution of the isothermal stationary linear Darcy equation is defined by the radial pressure

$$P(r) = P_\omega + \frac{\bar{q}_\omega \mu}{2\pi k_m \rho} \ln\left(\frac{r}{r_\omega}\right), \quad r = \sqrt{x^2 + y^2} > 0, \quad (x, y, z) \in \Omega, \quad (26)$$

where \bar{q}_ω is the mass flow rate per unit well length. The total mass flow rate is

$$q_\omega = (2H + \frac{k_f}{k_m} d_f) \bar{q}_\omega.$$

This solution will be used to test the convergence of our discretization for both an injection and a production well with fixed temperature. For both test cases, the fluid properties are set to $\mu = 10^{-3}$ Pa.s for the viscosity and to $\rho = 10^3$ kg.m⁻³ for the mass density.

We consider a uniform Cartesian mesh of size $n_x \times n_x \times n_x$ of the domain Ω conforming to the fracture and to the well. The well indices $W_{\mathbf{s},\omega}$ for $\mathbf{s} \in \mathcal{V}_\omega$ are computed following Peaceman's methodology [14, 33, 34] using the analytical solution (26). Since the mesh is uniform it suffices to solve numerically a local 2D problem with four (or more) horizontal faces around a given well node \mathbf{s} . The pressure equation is solved using the VAG scheme in the 2D domain composed of the four (or more) faces with the flow rate \bar{q}_ω imposed at the well node \mathbf{s} and the radial pressure analytical solution (26) imposed at the boundary nodes. From the pressure numerical solution at the well node $P_{\mathbf{s}}$ we deduce the Peaceman radius r_0 defined by

$$(P_{\mathbf{s}} - P_\omega) = \frac{\bar{q}_\omega \mu}{2\pi k_m \rho} \ln\left(\frac{r_0}{r_\omega}\right).$$

This computation leads to the following solution for the numerical Peaceman indices at the limit of a large number of faces around the well node. Let us set

$$dx = \frac{H}{n_x}, \quad r_0 = 0.14036 dx, \quad W_0 = \frac{2\pi}{\ln\left(\frac{r_0}{r_\omega}\right)}.$$

Then, denoting by \mathcal{E}_ω the set of edges of the well, the well index of a given node $\mathbf{s} \in \mathcal{V}_\omega$ is given by

$$W_{\mathbf{s},\omega} = \left(\sum_{\epsilon \in \mathcal{E}_\omega | \mathbf{s} \in \epsilon} \frac{dx}{2} k_m \right) W_0,$$

for a matrix node $\mathbf{s} \in \mathcal{V}_\omega \setminus \mathcal{V}_\Gamma$, and by

$$W_{\mathbf{s},\omega} = \left(d_f k_f + \sum_{\epsilon \in \mathcal{E}_\omega | \mathbf{s} \in \epsilon} \frac{dx}{2} k_m \right) W_0$$

for the fracture node $\mathbf{s} \in \mathcal{V}_\omega \cap \mathcal{V}_\Gamma$. Since there is no coupling between the fracture and the matrix for this radial pressure solution, note that for the fracture node, the Peaceman index is just obtained by summing up the contributions from the fracture and from the matrix.

For both test cases, Dirichlet boundary conditions given by the analytical solution are imposed at the lateral boundaries of Ω and at the boundary of Γ . Homogeneous Neumann boundary conditions are imposed at the top and bottom boundaries of Ω . The well boundary condition is set to either a specified bottom hole pressure P_ω or a specified total flow rate q_ω .

Let us first consider the case of an injection well with the well pressure $P_\omega = 2 \times 10^7$ Pa and the flow rate per unit well length set to $\bar{q}_\omega = 0.1$ kg.s⁻¹.m⁻¹. The corresponding analytical solution defined by (26) with these parameters is shown in Figure 5. Figure 6 shows, for both a specified pressure or flow rate, the convergence

of the relative L^2 errors between the analytical solution and the numerical solution both in the matrix domain and in the fracture as a function of the mesh size $n_x = 10, 20, 40, 80$. We obtain a convergence of order 1 in all cases.

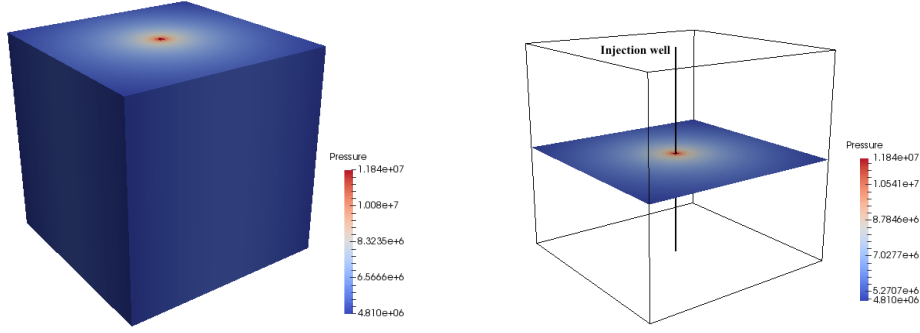


FIGURE 5. Analytical solution with the injection well in the matrix domain (left) and in the fracture (right).

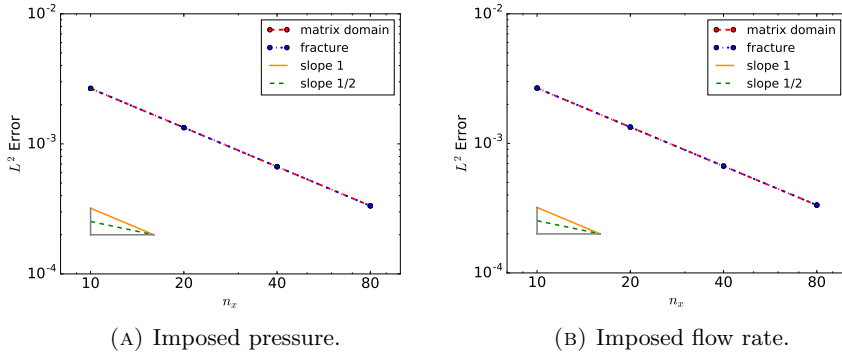


FIGURE 6. Relative L^2 errors between the analytical solution and the numerical solution with one injection well in the matrix domain and in the fracture, where the pressure is imposed (A) or the flow rate is imposed (B).

Next, we consider the case of a production well with the well pressure $P_w = 5 \times 10^6$ Pa and the well flow rate per unit well length set to $\bar{q}_w = 0.1 \text{ kg.s}^{-1}.\text{m}^{-1}$. Figure 7 shows the analytical solution defined by (26) with these parameters.

We present in Figure 8 the convergence of the relative L^2 errors between the analytical solution and the numerical solution as a function of the mesh size $n_x = 10, 20, 40, 80$, both in the matrix domain and in the fracture and for both for a specified pressure or a specified flow rate. We obtain as previously a convergence of order 1 in all cases.

4.2. Non-isothermal single-phase flow

This test case considers a non-isothermal liquid flow with mass density, viscosity, specific internal energy and enthalpy obtained from [38]. The thermal conductivity is fixed to $\lambda = 2 \text{ W.m}^{-1}.\text{K}^{-1}$ and the rock internal energy density is defined by $E_r(T) = c_p^r T$ with $c_p^r = 16.10^5 \text{ J.m}^{-3}.\text{K}^{-1}$.

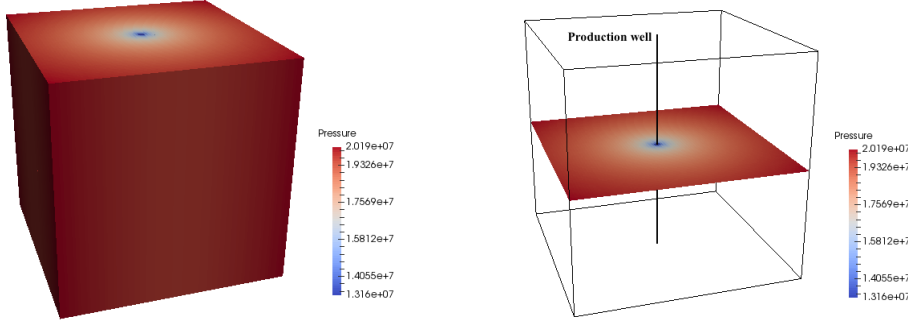


FIGURE 7. Analytical solution with the production well in the matrix domain (left) and in the fracture (right).

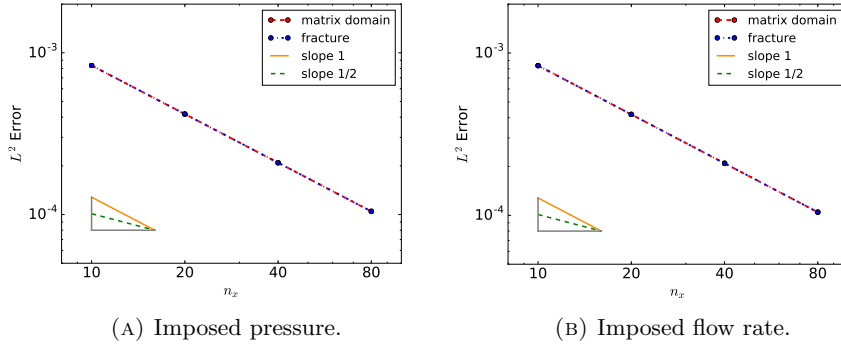


FIGURE 8. Relative L^2 errors between the analytical solution and the numerical solution obtained with the production well in the matrix domain and in the fracture, where the pressure is imposed (A) or the flow rate is imposed (B).

The simulation domain is defined by $\Omega = (0, 2000)^3$ in meters. The mesh is a 3D tetrahedral mesh conforming to the fracture network and to the wells. It was generated using the implicit framework from the 3D mesh generation package from the Computational Geometry Algorithms Library (CGAL [39]). As shown in Figure 9, there is one injection well (red line) and one multi-branch production well (green line). This mesh contains about 4.9×10^6 cells, 2.8×10^4 fracture faces and 8.0×10^5 nodes. The radius of both wells is set to 0.1 meter and the fracture width is fixed to $d_f = 1$ meter. The permeabilities are isotropic and set to $\mathbf{K}_m = 10^{-14} I \text{ m}^2$ in the matrix domain and to $\mathbf{K}_f = 10^{-11} I \text{ m}^2$ in the fracture network. The porosities in the matrix domain and in the fractures are $\phi_m = 0.1$ and $\phi_f = 0.4$ respectively. The computation of numerical Peaceman indices would require an analytical solution for the linear diffusion equation, which is not known for such a complex geometry involving fractures and multi-branch wells. This solution could also be obtained numerically using a mesh at the scale of the wells but its generation is out of the scope of this test case. Alternatively, we will use for this test case approximate analytical Peaceman type formulas providing a good order of magnitude for the Peaceman indices.

The domain is initially at the constant temperature 413 K and the constant pressure 2.0×10^7 Pa. The temperature is fixed to 413 K and the pressure is fixed to 2.0×10^7 Pa at the lateral boundaries of the domain. Zero fluxes for both mass and energy are imposed at the top and bottom boundaries of the domain. At the injection well, a cold water at temperature $T_{inj} = 333$ K is injected with the maximum bottom hole pressure $\bar{P}_{inj}^{max} = 3.0 \times 10^7$ Pa and the total mass flow rate $\bar{q}_{inj} = -27.78$ kg/s (i.e. -100 t/h). At the production well,

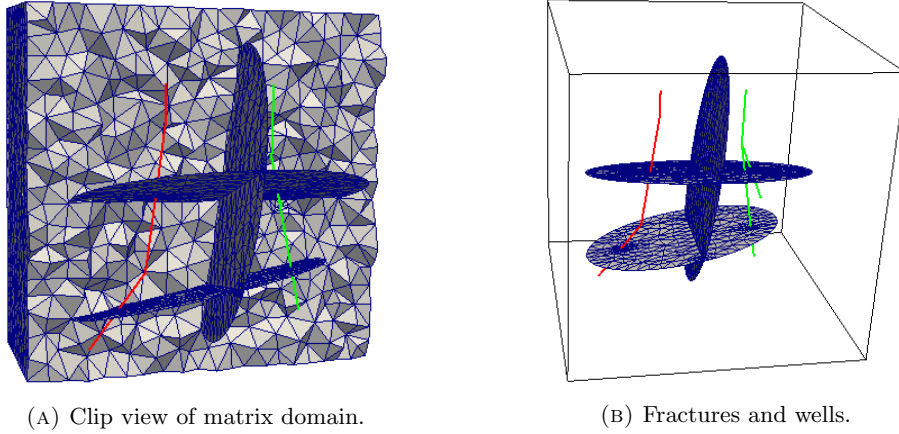


FIGURE 9. Coarse 3D tetrahedral mesh conforming to the fracture network and to the wells. There are one injection well (red line) and one production well (green line).

t_f	$5 \cdot 10^6$ days	final simulation time
Δt	$4 \cdot 10^4$ days	time step
N_{newton}^{max}	40	maximum number of nonlinear iterations
N_{gmres}^{max}	150	maximum number of linear iterations
ϵ_{newton}	10^{-5}	nonlinear relative residual stopping criteria
ϵ_{gmres}	10^{-6}	linear relative residual tolerance

TABLE 1. Simulation parameters

hot water is produced with the minimum bottom hole pressure $\bar{P}_{prod}^{min} = 1.0 \times 10^7$ Pa and the total mass flow rate $\bar{q}_{prod} = -\bar{q}_{inj} = +27.78$ kg/s.

Figures 10 and 11 exhibit the temperature in the matrix domain and in the fractures at times $t = 4 \times 10^4$ days and $t = t_f$. The temperature at the root node of the production well as a function of time is shown in Figure 12.

Then, we show in Figures 13 and 14 the pressure in the matrix domain and in the fractures at times $t = 4 \times 10^4$ days and $t = t_f$. In addition, the pressures at the root nodes of both wells as a function of time are shown in Figure 15.

Finally, we present in Figure 16 the total computational time in hours for different numbers of MPI processes $N_p = 4, 8, 16, 32, 64, 128$. The scalability behaves as expected for fully implicit time integration and AMG type preconditioners. It is well known that the AMG preconditioner requires a sufficient number of unknowns per MPI process, say 10^5 as typical order of magnitude, to achieve a linear strong scaling. For this mesh size, leading to roughly 8.2×10^5 unknowns for the pressure block, the scalability is still not far from linear on up to 64 processes and then degrades more rapidly for $N_p = 128$.

5. CONCLUSION

In this paper, the non-isothermal hybrid-dimensional Darcy flow model presented in [42] has been extended to incorporate thermal well models coupled with both the matrix domain and the fracture network. The well data structure is based on a rooted tree defined by a set of edges of the mesh. This allows to represent efficiently both slanted and multi-branch wells taking advantage of the unstructured mesh and of the nodal feature of the VAG discretization. The fluxes connecting the well with the 3D matrix and the 2D fracture network at each

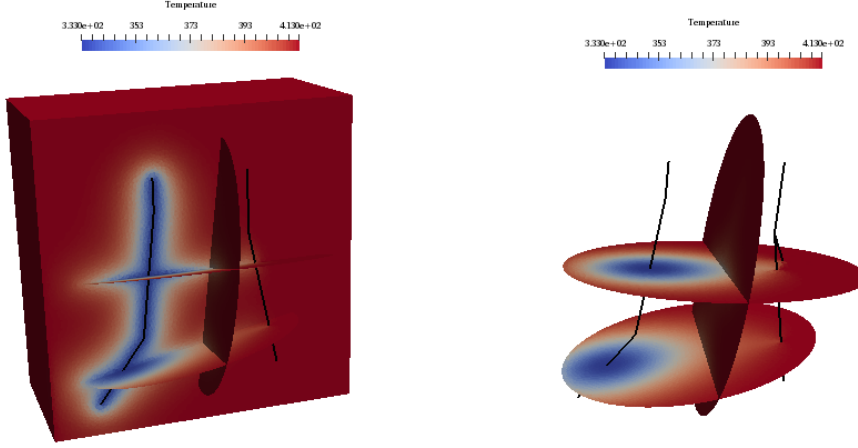


FIGURE 10. Left: temperature in the matrix domain and in the fractures at $t = 4 \times 10^4$ days where a clip view on plane $\{y = 1000\}$ is used in the matrix domain. Right: temperature in the fractures at $t = 4 \times 10^4$ days. The wells are drawn as black lines in both figures.

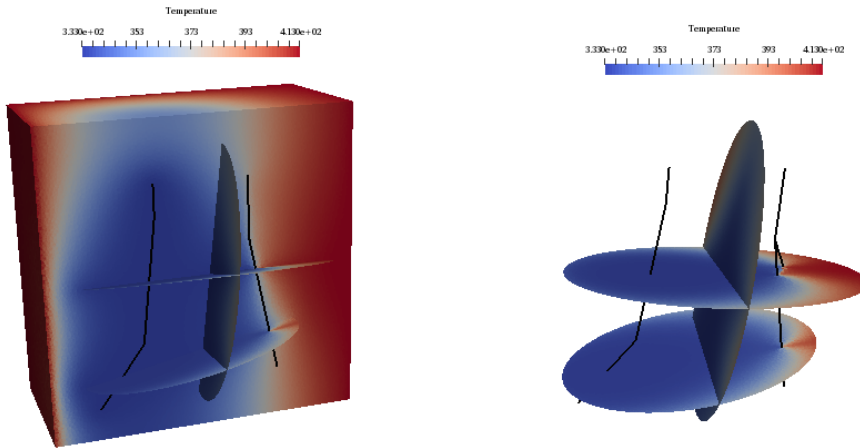


FIGURE 11. Left: temperature in the matrix domain and in the fractures at $t = 5 \times 10^6$ days where a clip view on plane $\{y = 1000\}$ is used in the matrix domain. Right: temperature in the fractures at $t = 5 \times 10^6$ days. The wells are drawn as black lines in both figures.

node of the well are computed using Peaceman's approach, and the well non-isothermal flow model is based on the usual single unknown approach assuming the hydrostatic and thermodynamical equilibrium inside the well. The parallelization of the well model is performed by definition of own and ghost wells for each process and by extension of the ghost nodes in order to account for the additional connectivity induced by the own and ghost well equations. This allows to assemble the Jacobian and to compute the well pressure drops locally on each process without MPI communications.

The model has been validated using a pressure analytical solution on a simple geometry with one horizontal fracture and one vertical well. The efficiency of the model, both in terms of ability to account for complex geology and in terms of parallel scalability, is demonstrated on a non-isothermal single-phase flow test case using a tetrahedral mesh with roughly 4.9×10^6 cells and including three intersecting fractures, one slanted

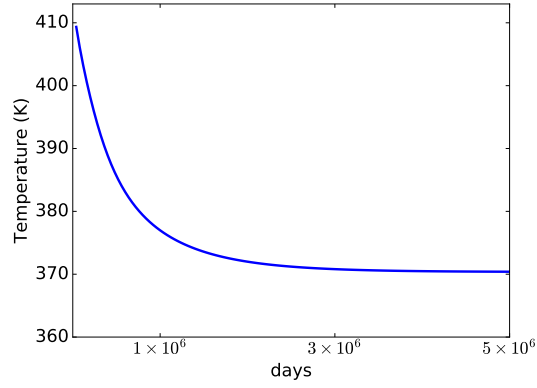


FIGURE 12. Temperature at the root node of the production well as a function of time.

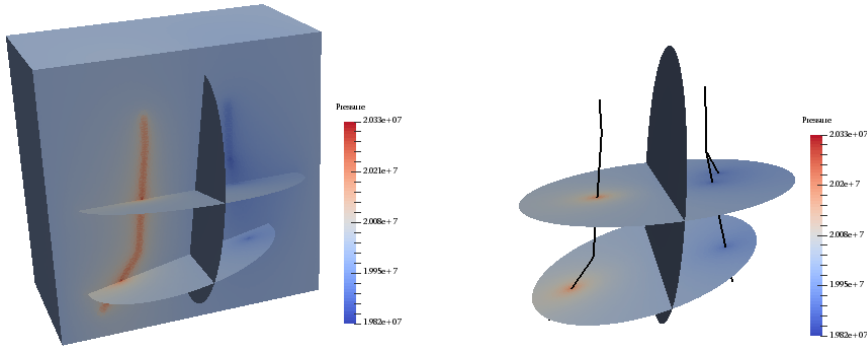


FIGURE 13. Left: pressure in the matrix domain and in the fractures at $t = 4 \times 10^4$ days where a clip view on plane $\{y = 1000\}$ is used in the matrix domain. Right: pressure in the fractures at $t = 4 \times 10^4$ days where the wells are drawn with black lines.

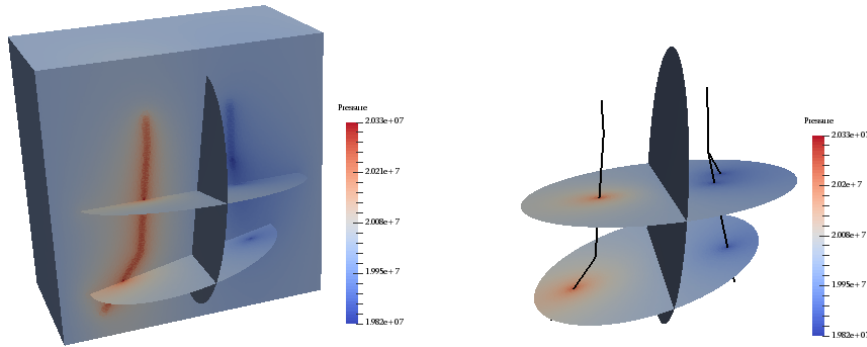


FIGURE 14. Left: pressure in the matrix domain and in the fractures at $t = 5 \times 10^6$ days where a clip view on plane $\{y = 1000\}$ is used in the matrix domain. Right: pressure in the fractures at $t = 5 \times 10^6$ days where the wells are drawn with black lines.

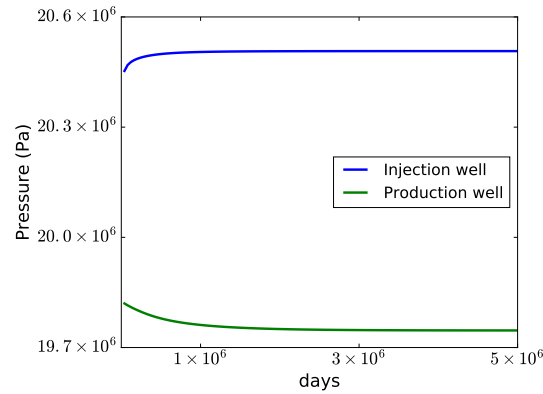


FIGURE 15. Pressures at the root nodes of both wells as a function of time.

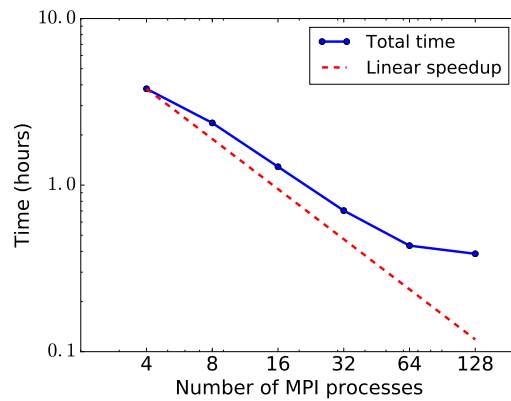


FIGURE 16. Total computational time vs. number of MPI processes.

injection well, and one multi-branch production well. This is an important step toward the application of our simulator to real geothermal test cases in a near future.

ACKNOWLEDGMENTS

This work was supported by a joint project between INRIA and BRGM Carnot institutes (ANR, INRIA, BRGM). It was granted access to the HPC and visualization resources of “Centre de Calcul Interactif” hosted by University Nice Sophia-Antipolis. The authors would also like to thank the région PACA and BRGM for co-funding the PhD of Laurence Beaudé as well as CEA for funding the participation of Thibaud Beltzung to the project. Writing of this article was supported by the CHARMS ANR project (ANR-16-CE06-0009).

REFERENCES

- [1] R. Ahmed, M. Edwards, S. Lamine, B. Huisman, and M. Pal. Control-volume distributed multi-point flux approximation coupled with a lower-dimensional fracture model. *Journal of Computational Physics*, 284:462–489, mar 2015.

- [2] R. Ahmed, M. G. Edwards, S. Lamine, B. A. Huisman, and M. Pal. Three-dimensional control-volume distributed multi-point flux approximation coupled with a lower-dimensional surface fracture model. *Journal of Computational Physics*, 303:470–497, dec 2015.
- [3] C. Alboin, J. Jaffré, J. Roberts, and C. Serres. Modeling fractures as interfaces for flow and transport in porous media. volume 295, pages 13–24, 2002.
- [4] P. Angot, F. Boyer, and F. Hubert. Asymptotic and numerical modelling of flows in fractured porous media. *ESAIM: Mathematical Modelling and Numerical Analysis*, 43(2):239–275, mar 2009.
- [5] P. F. Antonietti, L. Formaggia, A. Scotti, M. Verani, and N. Verzott. Mimetic finite difference approximation of flows in fractured porous media. *ESAIM M2AN*, 50:809–832, 2016.
- [6] Z. P. Aunzo, G. Björnsson, and G. S. Bodvarsson. Wellbore Models GWELL, GWNACL, and HOLA, user’s guide. Technical Report LBL-31428, Earth Sciences Division, Lawrence Berkeley National Laboratory, University of California, 1991.
- [7] K. Aziz and A. Settari. *Petroleum Reservoir Simulation*. Applied Science Publishers, 1979.
- [8] S. Balay, M. Adams, J. Brown, P. Brune, K. Buschelman, V. Eijkhout, and H. Zhang. PETSc Users Manual. Revision 3.5. Technical report, 2015.
- [9] I. I. Bogdanov, V. V. Mourzenko, J.-F. Thovert, and P. M. Adler. Two-phase flow through fractured porous media. *Physical Review E*, 68(2), aug 2003.
- [10] K. Brenner, M. Groza, C. Guichard, G. Lebeau, and R. Masson. Gradient discretization of hybrid-dimensional Darcy flows in fractured porous media. *Numerische Mathematik*, 134(3):569–609, nov 2016.
- [11] K. Brenner, M. Groza, C. Guichard, and R. Masson. Vertex Approximate Gradient scheme for hybrid-dimensional two-phase Darcy flows in fractured porous media. *ESAIM: Mathematical Modelling and Numerical Analysis*, 2(49):303–330, 2015.
- [12] K. Brenner, J. Hennicker, R. Masson, and P. Samier. Gradient discretization of hybrid-dimensional Darcy flow in fractured porous media with discontinuous pressures at matrix-fracture interfaces. *IMA Journal of Numerical Analysis*, sep 2016.
- [13] K. Brenner, J. Hennicker, R. Masson, and P. Samier. Hybrid-dimensional modelling and discretization of two phase darcy flow through DFN in porous media. In *ECMOR XV- 15th European Conference on the Mathematics of Oil Recovery*, 2016.
- [14] Z. Chen and Y. Zhang. Well flow models for various numerical methods. *J. Numer. Anal. Model.*, 6:375–388, 2009.
- [15] J. Droniou, R. Eymard, T. Gallouët, and R. Herbin. A unified approach to mimetic finite difference, hybrid finite volume and mixed finite volume methods. *Math. Models Methods Appl. Sci.*, 20(2):265–295, 2010.
- [16] R. Eymard, C. Guichard, and R. Herbin. Small-stencil 3D schemes for diffusive flows in porous media. *ESAIM: Mathematical Modelling and Numerical Analysis*, 46(2):265–290, 2012.
- [17] R. Eymard, C. Guichard, R. Herbin, and R. Masson. Vertex-centred discretization of multiphase compositional Darcy flows on general meshes. *Computational Geosciences*, 16(4):987–1005, 2012.
- [18] I. Faille, A. Fumagalli, J. Jaffré, and J. E. Roberts. Model reduction and discretization using hybrid finite volumes of flow in porous media containing faults. *Computational Geosciences*, 20:317–339, 2016.
- [19] E. Flauraud, F. Nataf, I. Faille, and R. Masson. Domain decomposition for an asymptotic geological fault modeling. *Comptes Rendus Mécanique*, 331(12):849–855, dec 2003.
- [20] H. Haegland, A. Assteerawatt, H. Dahle, G. Eigestad, and R. Helmig. Comparison of cell- and vertex-centered discretization methods for flow in a two-dimensional discrete-fracture-matrix system. *Advances in Water resources*, 32:1740–1755, 2009.
- [21] V. E. Henson and U. M. Yang. BoomerAMG: A parallel algebraic multigrid solver and preconditioner. *Applied Numerical Mathematics*, 41(1):155–177, 2002.
- [22] H. Hoteit and A. Firoozabadi. An efficient numerical model for incompressible two-phase flow in fractured media. *Advances in Water Resources*, 31(6):891–905, jun 2008.
- [23] M. Karimi-Fard, L. Durlinsky, and K. Aziz. An efficient discrete-fracture model applicable for general-purpose reservoir simulators. *SPE Journal*, 9(02):227–236, jun 2004.
- [24] G. Karypis and V. Kumar. A Fast and high quality multilevel scheme for partitioning irregular graphs. *SIAM Journal on Scientific Computing*, 20(1):359–392, jan 1998.
- [25] J. Ketilsson, G. Gunnarsson, H. Palsson, R. Podgorney, J. Fairley, and S. Ingebritsen. Geothermal Reservoir Modeling Recommendations for Research and Development. Technical report, 2012.
- [26] S. Kräutle. The semi-smooth newton method for multicomponent reactive transport with minerals. *Advances in Water Resources*, 34:137–151, 2011.
- [27] S. Lacroix, Y. V. Vassilevski, and M. F. Wheeler. Decoupling preconditioners in the implicit parallel accurate reservoir simulator (IPARS). *Numerical Linear Algebra with Applications*, 8(8):537–549, dec 2001.
- [28] V. Martin, J. Jaffré, and J. E. Roberts. Modeling fractures and barriers as interfaces for flow in porous media. *SIAM Journal on Scientific Computing*, 26(5):1667–1691, 2005.
- [29] S. K. Matthai, A. A. Mezentsev, and M. Belayneh. Finite element - node-centered finite-volume two-phase-flow experiments with fractured rock represented by unstructured hybrid-element meshes. *SPE Reservoir Evaluation & Engineering*, 10(06):740–756, dec 2007.
- [30] J. E. Monteagudo and A. Firoozabadi. Control-volume model for simulation of water injection in fractured media: incorporating matrix heterogeneity and reservoir wettability effects. *SPE Journal*, 12(03):355–366, sep 2007.

- [31] J. O’Sullivan, A. Croucher, A. Yeh, and M. O’Sullivan. Further improvements in the convergence of Tough2 simulations. In *In 11th World Congress on Computational Mechanics, Barcelona, Spain*, 2014.
- [32] M. O’Sullivan. Future directions in geothermal modelling. In *In 35th New Zealand Geothermal Workshop. Rotorua, New Zealand*, 2013.
- [33] D. Peaceman. Interpretation of Well-Block Pressures in Numerical . *Reservoir Simulation Symposium Journal SEPJ*, pages 183–194, 1978.
- [34] D. Peaceman. Interpretation of Well-Block Pressures in Numerical Reservoir Simulation with Nonsquare Grid Blocks and Anisotropic Permeability. *Reservoir Simulation Symposium Journal SEPJ*, pages 531–543, 1983.
- [35] V. Reichenberger, H. Jakobs, P. Bastian, and R. Helmig. A mixed-dimensional finite volume method for two-phase flow in fractured porous media. *Advances in Water Resources*, 29(7):1020–1036, jul 2006.
- [36] T. Sandve, I. Berre, and J. Nordbotten. An efficient multi-point flux approximation method for Discrete Fracture-Matrix simulations. *Journal of Computational Physics*, 231(9):3784–3800, may 2012.
- [37] R. Scheichl, R. Masson, and J. Wendebourg. Decoupling and block preconditioning for sedimentary basin simulations. *Computational Geosciences*, 7(4):295–318, 2003.
- [38] E. Schmidt. *Properties of water and steam in S.I. units*. Springer-Verlag, 1969.
- [39] The CGAL Project. *CGAL User and Reference Manual*. CGAL Editorial Board, 4.9 edition, 2016.
- [40] X. Tunc, I. Faille, T. Gallouët, M. C. Cacas, and P. Havé. A model for conductive faults with non-matching grids. *Computational Geosciences*, 16(2):277–296, mar 2012.
- [41] F. Xing, R. Masson, and S. Lopez. Parallel Vertex Approximate Gradient discretization of hybrid-dimensional Darcy flow and transport in discrete fracture networks. *Computational Geosciences*, 2016.
- [42] F. Xing, R. Masson, and S. Lopez. Parallel numerical modeling of hybrid-dimensional compositional non-isothermal Darcy flows in fractured porous media. *Journal of Computational Physics*, (345):637–664, 2017.

Appendix B

Discrete operators

The explicit expression of the convective operator in three dimensional space, for the *FVM* reads:

$$\begin{aligned}
(Ax)_{i-1/2,j,k}^n &= \frac{1}{V} \oint_S (\rho \mathbf{u} \otimes \mathbf{u}) \cdot \mathbf{n} \, ds \Big|_x \\
&= \frac{1}{\Delta x \Delta y \Delta z} \left[((\rho u u)_{i,j,k} - (\rho u u)_{i-1,j,k}) \Delta y \Delta z \right. \\
&\quad + \left((\rho u v)_{i-1/2,j+1/2,k} - (\rho u v)_{i-1/2,j-1/2,k} \right) \Delta x \Delta z \\
&\quad \left. + \left((\rho u w)_{i-1/2,j,k+1/2} - (\rho u w)_{i-1/2,j,k-1/2} \right) \Delta x \Delta y \right] \\
&= \frac{1}{\Delta x} \left[\rho_{i,j,k} \left(\frac{u_{i-1/2,j,k} + u_{i+1/2,j,k}}{2} \right)^2 - \rho_{i-1,j,k} \left(\frac{u_{i-3/2,j,k} + u_{i-1/2,j,k}}{2} \right)^2 \right] + \\
&\quad \frac{1}{\Delta y} \left[\left(\frac{\rho_{i,j,k} + \rho_{i-1,j,k} + \rho_{i,j+1,k} + \rho_{i-1,j+1,k}}{4} \right) \right. \\
&\quad \cdot \left(\frac{u_{i-1/2,j,k} + u_{i-1/2,j+1,k}}{2} \right) \left(\frac{v_{i-1,j+1/2,k} + v_{i,j+1/2,k}}{2} \right) \\
&\quad - \left(\frac{\rho_{i,j,k} + \rho_{i-1,j,k} + \rho_{i,j-1,k} + \rho_{i-1,j-1,k}}{4} \right) \\
&\quad \cdot \left(\frac{u_{i-1/2,j-1,k} + u_{i-1/2,j,k}}{2} \right) \left(\frac{v_{i-1,j-1/2,k} + v_{i,j-1/2,k}}{2} \right) \Big] + \\
&\quad \frac{1}{\Delta z} \left[\left(\frac{\rho_{i,j,k} + \rho_{i-1,j,k} + \rho_{i,j,k+1} + \rho_{i-1,j,k+1}}{4} \right) \right. \\
&\quad \cdot \left(\frac{u_{i-1/2,j,k} + u_{i-1/2,j,k+1}}{2} \right) \left(\frac{w_{i-1,j,k+1/2} + w_{i,j,k+1/2}}{2} \right) \\
&\quad - \left(\frac{\rho_{i,j,k} + \rho_{i-1,j,k} + \rho_{i,j,k-1} + \rho_{i-1,j,k-1}}{4} \right) \\
&\quad \cdot \left(\frac{u_{i-1/2,j,k-1} + u_{i-1/2,j,k}}{2} \right) \left(\frac{w_{i-1,j,k-1/2} + w_{i,j,k-1/2}}{2} \right) \Big]
\end{aligned}$$

$$\begin{aligned}
(Ay)_{i,j-1/2,k}^n &= \frac{1}{V} \oint_S (\rho \mathbf{u} \otimes \mathbf{u}) \cdot \mathbf{n} \, ds \Big|_y \\
&= \frac{1}{\Delta x \Delta y \Delta z} \left[\left((\rho v u)_{i+1/2,j-1/2,k} - (\rho v u)_{i-1/2,j-1/2,k} \right) \Delta y \Delta z \right. \\
&\quad \left. + \left((\rho v v)_{i,j,k} - (\rho v v)_{i,j-1,k} \right) \Delta x \Delta z \right. \\
&\quad \left. + \left((\rho v w)_{i,j-1/2,k+1/2} - (\rho v w)_{i,j-1/2,k-1/2} \right) \Delta x \Delta y \right] \\
&= \frac{1}{\Delta x} \left[\left(\frac{\rho_{i,j,k} + \rho_{i+1,j,k} + \rho_{i,j-1,k} + \rho_{i+1,j-1,k}}{4} \right) \right. \\
&\quad \cdot \left(\frac{v_{i,j-1/2,k} + v_{i+1,j-1/2,k}}{2} \right) \left(\frac{u_{i+1/2,j-1,k} + u_{i+1/2,j,k}}{2} \right) \\
&\quad - \left(\frac{\rho_{i,j,k} + \rho_{i-1,j,k} + \rho_{i,j-1,k} + \rho_{i-1,j-1,k}}{4} \right) \\
&\quad \cdot \left(\frac{v_{i-1,j-1/2,k} + v_{i,j-1/2,k}}{2} \right) \left(\frac{u_{i-1/2,j-1,k} + u_{i-1/2,j,k}}{2} \right) \Big] + \\
&\quad \frac{1}{\Delta y} \left[\rho_{i,j,k} \left(\frac{v_{i,j-1/2,k} + v_{i,j+1/2,k}}{2} \right)^2 - \rho_{i,j-1,k} \left(\frac{v_{i,j-3/2,k} + v_{i,j-1/2,k}}{2} \right)^2 \right] + \\
&\quad \frac{1}{\Delta z} \left[\left(\frac{\rho_{i,j,k} + \rho_{i,j-1,k} + \rho_{i,j,k+1} + \rho_{i,j-1,k+1}}{4} \right) \right. \\
&\quad \cdot \left(\frac{v_{i,j-1/2,k} + v_{i,j-1/2,k+1}}{2} \right) \left(\frac{w_{i,j-1,k+1/2} + w_{i,j,k+1/2}}{2} \right) \\
&\quad - \left(\frac{\rho_{i,j,k} + \rho_{i,j-1,k} + \rho_{i,j,k-1} + \rho_{i,j-1,k-1}}{4} \right) \\
&\quad \cdot \left(\frac{v_{i,j-1/2,k-1} + v_{i,j-1/2,k}}{2} \right) \left(\frac{w_{i,j-1,k-1/2} + w_{i,j,k-1/2}}{2} \right) \Big]
\end{aligned}$$

$$\begin{aligned}
(Az)_{i,j,k-1/2}^n &= \frac{1}{V} \oint_S (\rho \mathbf{u} \otimes \mathbf{u}) \cdot \mathbf{n} \, ds \Big|_z \\
&= \frac{1}{\Delta x \Delta y \Delta z} \left[\left((\rho w u)_{i+1/2,j,k-1/2} - (\rho w u)_{i-1/2,j,k-1/2} \right) \Delta y \Delta z \right. \\
&\quad \left. + \left((\rho w v)_{i,j+1/2,k-1/2} - (\rho w v)_{i,j-1/2,k-1/2} \right) \Delta x \Delta z \right. \\
&\quad \left. + \left((\rho w w)_{i,j,k} - (\rho w w)_{i,j,k-1} \right) \Delta x \Delta y \right] \\
&= \frac{1}{\Delta x} \left[\left(\frac{\rho_{i,j,k} + \rho_{i+1,j,k} + \rho_{i,j,k-1} + \rho_{i+1,j,k-1}}{4} \right) \right. \\
&\quad \cdot \left(\frac{w_{i,j,k-1/2} + w_{i+1,j,k-1/2}}{2} \right) \left(\frac{u_{i+1/2,j,k-1} + u_{i+1/2,j,k}}{2} \right) \\
&\quad \left. - \left(\frac{\rho_{i,j,k} + \rho_{i-1,j,k} + \rho_{i,j,k-1} + \rho_{i-1,j,k-1}}{4} \right) \right. \\
&\quad \cdot \left(\frac{w_{i-1,j,k-1/2} + w_{i,j,k-1/2}}{2} \right) \left(\frac{u_{i-1/2,j,k-1} + u_{i-1/2,j,k}}{2} \right) \Big] + \\
&\quad \frac{1}{\Delta y} \left[\left(\frac{\rho_{i,j,k} + \rho_{i,j+1,k} + \rho_{i,j,k-1} + \rho_{i,j+1,k-1}}{4} \right) \right. \\
&\quad \cdot \left(\frac{w_{i,j,k-1/2} + w_{i,j+1,k-1/2}}{2} \right) \left(\frac{v_{i,j+1/2,k-1} + v_{i,j+1/2,k}}{2} \right) \\
&\quad \left. - \left(\frac{\rho_{i,j,k} + \rho_{i,j-1,k} + \rho_{i,j,k-1} + \rho_{i,j-1,k-1}}{4} \right) \right. \\
&\quad \cdot \left(\frac{w_{i,j-1,k-1/2} + w_{i,j,k-1/2}}{2} \right) \left(\frac{v_{i,j-1/2,k-1} + v_{i,j-1/2,k}}{2} \right) \Big] + \\
&\quad \frac{1}{\Delta z} \left[\rho_{i,j,k} \left(\frac{w_{i,j,k-1/2} + w_{i,j,k+1/2}}{2} \right)^2 - \rho_{i,j,k-1} \left(\frac{w_{i,j,k-3/2} + w_{i,j,k-1/2}}{2} \right)^2 \right]
\end{aligned}$$

For the diffusion operator, the expressions are:

$$\begin{aligned}
(Dx)_{i-1/2,j,k}^n &= \frac{1}{V} \oint_S \mu(\nabla \mathbf{u} + \nabla \mathbf{u}^t) \cdot \mathbf{n} \, ds \Big|_x \\
&= \frac{1}{\Delta x \Delta y \Delta z} \left\{ [\mu(2u_x)_{i,j,k} - \mu(2u_x)_{i-1,j,k}] \Delta y \Delta z \right. \\
&\quad + [\mu(u_y + v_x)_{i-1/2,j+1/2,k} - \mu(u_y + v_x)_{i-1/2,j-1/2,k}] \Delta x \Delta z \\
&\quad \left. + [\mu(u_z + w_x)_{i-1/2,j,k+1/2} - \mu(u_z + w_x)_{i-1/2,j,k-1/2}] \Delta x \Delta y \right\} \\
&= \frac{1}{\Delta x} \left\{ 2\mu_{i,j,k} \cdot \left(\frac{u_{i+1/2,j,k} - u_{i-1/2,j,k}}{\Delta x} \right) - 2\mu_{i-1,j,k} \right. \\
&\quad \left. \cdot \left(\frac{u_{i-1/2,j,k} - u_{i-3/2,j,k}}{\Delta x} \right) \right\} + \\
&\quad \frac{1}{\Delta y} \left\{ \frac{\mu_{i-1,j,k} + \mu_{i,j,k} + \mu_{i-1,j+1,k} + \mu_{i,j+1,k}}{4} \right. \\
&\quad \cdot \left(\frac{u_{i-1/2,j+1,k} - u_{i-1/2,j,k}}{\Delta y} + \frac{v_{i,j+1/2,k} - v_{i-1,j+1/2,k}}{\Delta x} \right) \\
&\quad - \frac{\mu_{i-1,j-1,k} + \mu_{i,j-1,k} + \mu_{i-1,j,k} + \mu_{i,j,k}}{4} \\
&\quad \left. \cdot \left(\frac{u_{i-1/2,j,k} - u_{i-1/2,j-1,k}}{\Delta y} + \frac{v_{i,j-1/2,k} - v_{i-1,j-1/2,k}}{\Delta x} \right) \right\} + \\
&\quad \frac{1}{\Delta z} \left\{ \frac{\mu_{i-1,j,k} + \mu_{i,j,k} + \mu_{i-1,j,k+1} + \mu_{i,j,k+1}}{4} \right. \\
&\quad \cdot \left(\frac{u_{i-1/2,j,k+1} - u_{i-1/2,j,k}}{\Delta z} + \frac{w_{i,j,k+1/2} - w_{i-1,j,k+1/2}}{\Delta x} \right) \\
&\quad - \frac{\mu_{i-1,j,k-1} + \mu_{i,j,k-1} + \mu_{i-1,j,k} + \mu_{i,j,k}}{4} \\
&\quad \left. \cdot \left(\frac{u_{i-1/2,j,k} - u_{i-1/2,j,k-1}}{\Delta z} + \frac{w_{i,j,k-1/2} - w_{i-1,j,k-1/2}}{\Delta x} \right) \right\}
\end{aligned}$$

$$\begin{aligned}
(Dy)_{i,j-1/2,k}^n &= \frac{1}{V} \oint_S \mu(\nabla \mathbf{u} + \nabla \mathbf{u}^t) \cdot \mathbf{n} \, ds \Big|_y \\
&= \frac{1}{\Delta x \Delta y \Delta z} \left\{ [\mu(v_x + u_y)_{i+1/2,j-1/2,k} - \mu(v_x + u_y)_{i-1/2,j-1/2,k}] \Delta y \Delta z \right. \\
&\quad + [\mu(2v_y)_{i,j,k} - \mu(2v_y)_{i,j-1,k}] \Delta x \Delta z \\
&\quad \left. + [\mu(v_z + w_y)_{i,j-1/2,k+1/2} - \mu(v_z + w_y)_{i,j-1/2,k-1/2}] \Delta x \Delta y \right\} \\
&= \frac{1}{\Delta x} \left\{ \frac{\mu_{i,j-1,k} + \mu_{i+1,j-1,k} + \mu_{i,j,k} + \mu_{i+1,j,k}}{4} \right. \\
&\quad \cdot \left(\frac{v_{i+1,j-1/2,k} - v_{i,j-1/2,k}}{\Delta x} + \frac{u_{i+1/2,j,k} - u_{i+1/2,j-1,k}}{\Delta y} \right) \\
&\quad - \frac{\mu_{i-1,j-1,k} + \mu_{i,j-1,k} + \mu_{i-1,j,k} + \mu_{i,j,k}}{4} \\
&\quad \cdot \left(\frac{v_{i,j-1/2,k} - v_{i-1,j-1/2,k}}{\Delta x} + \frac{u_{i-1/2,j,k} - u_{i-1/2,j-1,k}}{\Delta y} \right) \Big\} + \\
&\quad \frac{1}{\Delta y} \left\{ 2\mu_{i,j,k} \cdot \left(\frac{v_{i,j+1/2,k} - v_{i,j-1/2,k}}{\Delta y} \right) - 2\mu_{i,j-1,k} \cdot \left(\frac{v_{i,j-1/2,k} - v_{i,j-3/2,k}}{\Delta y} \right) \right\} \\
&\quad + \\
&\quad \frac{1}{\Delta z} \left\{ \frac{\mu_{i,j-1,k} + \mu_{i,j,k} + \mu_{i,j-1,k+1} + \mu_{i,j,k+1}}{4} \right. \\
&\quad \cdot \left(\frac{v_{i,j-1/2,k+1} - v_{i,j-1/2,k}}{\Delta z} + \frac{w_{i,j,k+1/2} - w_{i,j-1,k+1/2}}{\Delta y} \right) \\
&\quad - \frac{\mu_{i,j-1,k-1} + \mu_{i,j,k-1} + \mu_{i,j-1,k} + \mu_{i,j,k}}{4} \\
&\quad \cdot \left(\frac{v_{i,j-1/2,k} - v_{i,j-1/2,k-1}}{\Delta z} + \frac{w_{i,j,k-1/2} - w_{i,j-1,k-1/2}}{\Delta y} \right) \Big\}
\end{aligned}$$

$$\begin{aligned}
(Dz)_{i,j,k-1/2}^n &= \frac{1}{V} \oint_S \mu(\nabla \mathbf{u} + \nabla \mathbf{u}^t) \cdot \mathbf{n} \, ds \Big|_z \\
&= \frac{1}{\Delta x \Delta y \Delta z} \left\{ [\mu(w_x + u_z)_{i+1/2,j,k-1/2} - \mu(w_x + u_z)_{i-1/2,j,k-1/2}] \Delta y \Delta z \right. \\
&\quad + [\mu(w_y + v_z)_{i,j+1/2,k-1/2} - \mu(w_y + v_z)_{i,j-1/2,k-1/2}] \Delta x \Delta z \\
&\quad \left. + [\mu(2w_z)_{i,j,k} - \mu(2w_z)_{i,j,k-1}] \Delta x \Delta y \right\} \\
&= \frac{1}{\Delta x} \left\{ \frac{\mu_{i,j,k-1} + \mu_{i+1,j,k-1} + \mu_{i,j,k} + \mu_{i+1,j,k}}{4} \right. \\
&\quad \cdot \left(\frac{w_{i+1,j,k-1/2} - w_{i,j,k-1/2}}{\Delta x} + \frac{u_{i+1/2,j,k} - u_{i+1/2,j,k-1}}{\Delta z} \right) \\
&\quad - \frac{\mu_{i-1,j,k-1} + \mu_{i,j,k-1} + \mu_{i-1,j,k} + \mu_{i,j,k}}{4} \\
&\quad \cdot \left(\frac{w_{i,j,k-1/2} - w_{i-1,j,k-1/2}}{\Delta x} + \frac{u_{i-1/2,j,k} - u_{i-1/2,j,k-1}}{\Delta z} \right) \Big\} + \\
&\frac{1}{\Delta y} \left\{ \frac{\mu_{i,j,k-1} + \mu_{i,j+1,k-1} + \mu_{i,j,k} + \mu_{i,j+1,k}}{4} \right. \\
&\quad \cdot \left(\frac{w_{i,j+1,k-1/2} - w_{i,j,k-1/2}}{\Delta y} + \frac{v_{i,j+1/2,k} - v_{i,j+1/2,k-1}}{\Delta z} \right) \\
&\quad - \frac{\mu_{i,j-1,k-1} + \mu_{i,j,k-1} + \mu_{i,j-1,k} + \mu_{i,j,k}}{4} \\
&\quad \cdot \left(\frac{w_{i,j,k-1/2} - w_{i,j-1,k-1/2}}{\Delta y} + \frac{v_{i,j-1/2,k} - v_{i,j-1/2,k-1}}{\Delta z} \right) \Big\} + \\
&\frac{1}{\Delta z} \left\{ 2\mu_{i,j,k} \cdot \left(\frac{w_{i,j,k+1/2} - w_{i,j,k-1/2}}{\Delta z} \right) - 2\mu_{i,j,k-1} \cdot \left(\frac{w_{i,j,k-1/2} - w_{i,j,k-3/2}}{\Delta z} \right) \right\}
\end{aligned}$$

Bibliography

- [1] Wgsimon, *Transistor Count and Moore's Law*, 2011 (cit. on p. 3).
- [2] J. Bigot, V. Grandgirard, G. Latu, C. Passeron, F. Rozar, and O. Thomine, « Scaling gysela code beyond 32K-cores on bluegene/Q », in *ESAIM: Proceedings*, EDP Sciences, vol. 43, 2013, pp. 117–135 (cit. on p. 2).
- [3] A Kolmogorov, « The Local Structure of Turbulence in Incompressible Viscous Fluid for Very Large Reynolds' Numbers », vol. 30, pp. 301–305, 1941 (cit. on p. 3).
- [4] G. I. Taylor, « Diffusion by continuous movements », *Proceedings of the london mathematical society*, vol. 2, no. 1, pp. 196–212, 1922 (cit. on p. 3).
- [5] F Barre and M Bernard, « The CATHARE code strategy and assessment », *Nuclear engineering and design*, vol. 124, no. 3, pp. 257–284, 1990 (cit. on p. 4).
- [6] D. Bestion, « From the Direct Numerical Simulation to System Codes-Perspective for the Multi-scale Analysis of LWR Thermalhydraulics », *Nuclear Engineering and Technology*, vol. 42, no. 6, pp. 608–619, 2010 (cit. on p. 4).
- [7] —, « The physical closure laws in the CATHARE code », *Nuclear Engineering and Design*, vol. 124, no. 3, pp. 229–245, 1990 (cit. on p. 4).
- [8] D Tenchine, R Baviere, P Bazin, F Ducros, G Geffraye, D Kadri, F Perdu, D Pialla, B Rameau, and N Tauveron, « Status of CATHARE code for sodium cooled fast reactors », *Nuclear Engineering and Design*, vol. 245, pp. 140–152, 2012 (cit. on p. 4).
- [9] S. F. Corzo, D. M. Godino, N. M. Nigro, and D. E. Ramajo, « Thermal hydraulics simulation of the RD-14M steam generator facility », *Annals of Nuclear Energy*, vol. 105, pp. 282–301, 2017 (cit. on p. 4).
- [10] U. Bieder, F. Falk, and G. Fauchet, « CFD analysis of the flow in the near wake of a generic PWR mixing grid », *Annals of Nuclear Energy*, vol. 82, pp. 169–178, 2015 (cit. on p. 4).
- [11] B. Bois Guillaume, « Heat and mass transfers at liquid/vapor interfaces with phase-change : proposal for a large-scale modeling of interfaces », Theses, Université de Grenoble, Feb. 2011 (cit. on p. 4).

- [12] O. Thomine, « Développement de méthodes multi-échelles pour la simulation numérique des écoulements réactifs diphasiques », PhD thesis, 2011 (cit. on p. 5).
- [13] F. Doisneau, O. Thomine, F. Laurent, A. Vié, J. Dupays, and M. Massot, « Eulerian modeling and simulation of small scale trajectory crossing and coalescence for moderate-Stokes-number spray flows », in *Proceedings of the summer program*, 2012, pp. 365–374 (cit. on p. 5).
- [14] L. Fréret, O. Thomine, F. Laurent, J. Reveillon, and M. Massot, « On the ability of the Eulerian multi-fluid model to predict preferential segregation and flame dynamics in polydisperse evaporating sprays », *Submitted to Combustion and Flame*, p. 72, 2013 (cit. on p. 5).
- [15] G. Faccanoni, « Study of a Fine Model of Liquid-Vapor Phase Change. Contribution to the Boiling Crisis Study. », These en cotutelle entre l'Ecole Polytechnique et l'Universita di Trento (Italie), Theses, Ecole Polytechnique X, Nov. 2008 (cit. on p. 6).
- [16] F. Baj, « Amortissement et instabilités fluide-élastique d'un faisceau de tubes sous écoulement diphasique », PhD thesis, Université de Paris VI Pierre et Marie Curie, 1998 (cit. on pp. 7, 22).
- [17] A. Faghri and Y. Zhang, *Transport phenomena in multiphase systems*. Elsevier, 2006 (cit. on pp. 13, 14).
- [18] T. Ménard, S. Tanguy, and A. Berlemont, « Coupling level set/VOF/ghost fluid methods: Validation and application to 3D simulation of the primary break-up of a liquid jet », *International Journal of Multiphase Flow*, vol. 33, no. 5, pp. 510–524, 2007 (cit. on p. 12).
- [19] G. Hetsroni, « Handbook of multiphase systems », 1982 (cit. on p. 15).
- [20] S. Osher and J. A. Sethian, « Fronts propagating with curvature-dependent speed: algorithms based on Hamilton-Jacobi formulations », *Journal of computational physics*, vol. 79, no. 1, pp. 12–49, 1988 (cit. on p. 15).
- [21] S. Osher and R. P. Fedkiw, « Level set methods: an overview and some recent results », *Journal of Computational physics*, vol. 169, no. 2, pp. 463–502, 2001 (cit. on p. 15).
- [22] S. Osher and R. Fedkiw, *Level set methods and dynamic implicit surfaces*. Springer Science & Business Media, 2006, vol. 153 (cit. on p. 15).
- [23] J. A. Sethian, *Level set methods and fast marching methods: evolving interfaces in computational geometry, fluid mechanics, computer vision, and materials science*. Cambridge university press, 1999, vol. 3 (cit. on p. 16).
- [24] D. Enright, R. Fedkiw, J. Ferziger, and I. Mitchell, « A hybrid particle level set method for improved interface capturing », *Journal of Computational physics*, vol. 183, no. 1, pp. 83–116, 2002 (cit. on pp. 16, 111).

- [25] S. Tanguy, T. Ménard, and A. Berlemont, « A level set method for vaporizing two-phase flows », *Journal of Computational Physics*, vol. 221, no. 2, pp. 837–853, 2007 (cit. on p. 16).
- [26] C. W. Hirt and B. D. Nichols, « Volume of fluid (VOF) method for the dynamics of free boundaries », *Journal of computational physics*, vol. 39, no. 1, pp. 201–225, 1981 (cit. on p. 16).
- [27] J. Li, « Piecewise Linear Interface Calculation [Calcul d’Interface Affine par Morceaux] », *Comptes Rendus - Academie des Sciences*, vol. Serie II: Mecanique, Physique, Chimie, Astronomie, no. 320, pp. 391–396, 1995 (cit. on p. 16).
- [28] D. Gueyffier, J. Li, A. Nadim, R. Scardovelli, and S. Zaleski, « Volume-of-fluid interface tracking with smoothed surface stress methods for three-dimensional flows », *Journal of Computational Physics*, vol. 152, no. 2, pp. 423–456, 1999 (cit. on pp. 16, 72).
- [29] S. O. Unverdi and G. Tryggvason, « A front-tracking method for viscous, incompressible, multi-fluid flows », *Journal of computational physics*, vol. 100, no. 1, pp. 25–37, 1992 (cit. on p. 16).
- [30] G. Tryggvason, R. Scardovelli, and S. Zaleski, *Direct numerical simulations of gas-liquid multiphase flows*. Cambridge University Press, 2011 (cit. on p. 16).
- [31] R. Scardovelli and S. Zaleski, « Direct numerical simulation of free-surface and interfacial flow », *Annual review of fluid mechanics*, vol. 31, no. 1, pp. 567–603, 1999 (cit. on p. 16).
- [32] S. Popinet and S. Zaleski, « A front-tracking algorithm for accurate representation of surface tension », *International Journal for Numerical Methods in Fluids*, vol. 30, no. 6, pp. 775–793, 1999 (cit. on p. 16).
- [33] C. W. Hirt, A. A. Amsden, and J. Cook, « An arbitrary Lagrangian-Eulerian computing method for all flow speeds », *Journal of computational physics*, vol. 14, no. 3, pp. 227–253, 1974 (cit. on p. 17).
- [34] L. B. Lucy, « A numerical approach to the testing of the fission hypothesis », *The astronomical journal*, vol. 82, pp. 1013–1024, 1977 (cit. on p. 17).
- [35] R. A. Gingold and J. J. Monaghan, « Smoothed particle hydrodynamics: theory and application to non-spherical stars », *Monthly notices of the royal astronomical society*, vol. 181, no. 3, pp. 375–389, 1977 (cit. on p. 17).
- [36] L. D. Landau and E. M. Lifshitz, *Fluid Mechanics*, ser. Course of theoretical physics. Pergamon Press, 1987, vol. 6 (cit. on p. 17).
- [37] S. Adami, X. Hu, and N. A. Adams, « A generalized wall boundary condition for smoothed particle hydrodynamics », *Journal of Computational Physics*, vol. 231, no. 21, pp. 7057–7075, 2012 (cit. on p. 17).
- [38] M. Doring, « Développement d’une méthode SPH pour les applications à surface libre en hydrodynamique », PhD thesis, Nantes, 2005 (cit. on p. 17).

- [39] J. W. Cahn and J. E. Hilliard, « Free energy of a nonuniform system. I. Interfacial free energy », *The Journal of chemical physics*, vol. 28, no. 2, pp. 258–267, 1958 (cit. on p. 17).
- [40] J. Kim, « Phase-field models for multi-component fluid flows », *Communications in Computational Physics*, vol. 12, no. 03, pp. 613–661, 2012 (cit. on p. 17).
- [41] D. Jacqmin, « An energy approach to the continuum surface tension method », in *34th Aerospace Sciences Meeting and Exhibit*, 1996, p. 858 (cit. on pp. 17, 146).
- [42] —, « Calculation of two-phase Navier–Stokes flows using phase-field modeling », *Journal of Computational Physics*, vol. 155, no. 1, pp. 96–127, 1999 (cit. on p. 17).
- [43] V. Badalassi, H. Cenicerros, and S. Banerjee, « Computation of multiphase systems with phase field models », *Journal of Computational Physics*, vol. 190, no. 2, pp. 371–397, 2003 (cit. on p. 17).
- [44] S. Dellacherie, « Principe de construction du schema anti-diffusif de Despres-Lagoutière », CEA, Tech. Rep., 2009 (cit. on pp. 17, 80).
- [45] S. Kokh S. and F. Lagoutière, « An Anti-Diffusive Method For Simulating Interface Flows with a Five-Equation Model », *Numerical Methods for Multi-material Fluid Flows*, 2007 (cit. on p. 17).
- [46] B. Despres and F. Lagoutière, « Un schéma non linéaire anti-dissipatif pour l'équation d'advection linéaire », *Comptes Rendus de l'Académie des Sciences-Series I-Mathematics*, vol. 328, no. 10, pp. 939–943, 1999 (cit. on pp. 17, 80–84, 111, 160, 167, 177).
- [47] B. Leonard, « Universal limiter for transient interpolation modeling of the advective transport equations: the ULTIMATE conservative difference scheme », 1988 (cit. on p. 17).
- [48] A. Harten, « The artificial compression method for computation of shocks and contact discontinuities. III. Self-adjusting hybrid schemes », *Mathematics of Computation*, vol. 32, no. 142, pp. 363–389, 1978 (cit. on p. 18).
- [49] —, « The artificial compression method for computation of shocks and contact discontinuities. I. Single conservation laws », *Communications on Pure and Applied Mathematics*, vol. 30, no. 5, pp. 611–638, 1977 (cit. on p. 18).
- [50] X.-D. Liu, S. Osher, and T. Chan, « Weighted essentially non-oscillatory schemes », *Journal of computational physics*, vol. 115, no. 1, pp. 200–212, 1994 (cit. on p. 18).
- [51] C. S. Peskin, « Numerical analysis of blood flow in the heart », *Journal of computational physics*, vol. 25, no. 3, pp. 220–252, 1977 (cit. on p. 19).
- [52] G. Schewe, « On the force fluctuations acting on a circular cylinder in crossflow from subcritical up to transcritical Reynolds numbers », *Journal of fluid mechanics*, vol. 133, pp. 265–285, 1983 (cit. on p. 20).
- [53] E. de Langre, *Fluides et Solides*. Les éditions de l'Ecole Polytechnique, 2002 (cit. on p. 22).

- [54] S. Chen, « A general theory for dynamic instability of tube arrays in crossflow », *Journal of Fluids and Structures*, vol. 1, no. 1, pp. 35–53, 1987 (cit. on pp. 22, 62, 115).
- [55] S. Chen, S. Zhu, and J. Jendrzejczyk, « Fluid damping and fluid stiffness of a tube row in crossflow », *Journal of pressure vessel technology*, vol. 116, no. 4, pp. 370–383, 1994 (cit. on pp. 22, 115).
- [56] S. Pascal-Ribot and Y. Blanchet, « Buffeting lift forces and local air-water flow aspects around a rigid cylinder », *Int. J. Multiphase Flow*, vol. 33, pp. 1237–1254, 2007 (cit. on p. 22).
- [57] —, « An improved scaling model of buffeting lift forces in air-water flows », *Journal of Pressure Vessel Technology*, vol. 133, pp. 1–8, 2011 (cit. on p. 22).
- [58] M. Girao de Moraes, « Qualification numérique des méthodes de modélisation des forces fluide-élastique s'exerçant dans un faisceau de tubes en écoulement transversal », PhD thesis, Université d'Evry du val d'Essonne, FRANCE, 2006 (cit. on p. 23).
- [59] H. J. Connors, « Fluid Elastic Vibration of Tube Arrays Excited by Cross Flow », *ASME*, 1970 (cit. on p. 24).
- [60] H. Connors, « Fluidelastic vibration of heat exchanger tube arrays », *Journal of Mechanical Design*, vol. 100, no. 2, pp. 347–353, 1978 (cit. on p. 24).
- [61] R. Blevins, « Fluid-elastic whirling of a tube row. », *Journal of Pressure Vessels Technology*, 1974 (cit. on p. 24).
- [62] R. Blevins, « Fluid-elastic whirling of a tube rows and tube arrays. », *Journal of Fluid Engineering.*, 1977 (cit. on p. 24).
- [63] M. Paidoussis, M. M.P, and P. S.J, « A potential flow theory for dynamics of cylinder arrays in cross-flow. », *Journal of Fluids Mechanics.*, vol. Vol. 146, p.227–252, 1984 (cit. on p. 24).
- [64] S. Price and M. Paidoussis, « A single-flexible-cylinder analysis for the fluidelastic instability of an array of flexible cylinders in cross-flow », *Journal of Fluids Engineering*, vol. 108, no. 2, pp. 193–199, 1986 (cit. on p. 24).
- [65] J. Lever and D. Weaver, « A theoretical model for fluid-elastic instability in heat exchanger tube bundles. », *Journal of Pressure Vessel Technology*, vol. Vol. 104, p.147–158, 1982 (cit. on p. 24).
- [66] J. Lever and D. Weaver, « On the stability of heat exchanger tube bundles, part i: Modified theoretical model », *Journal of Sound and vibration*, vol. 107, no. 3, pp. 375–392, 1986 (cit. on p. 24).
- [67] A. Khalak and C. Williamson, « Motions, forces and mode transitions in vortex-induced vibrations at low mass-damping », *Journal of fluids and Structures*, vol. 13, no. 7, pp. 813–851, 1999 (cit. on p. 24).

- [68] M. Pettigrew and C. Taylor, « Fluidelastic instability of heat exchanger tube bundles », 1991 (cit. on p. 25).
- [69] R. Saurel and R. Abgrall, « A multiphase Godunov method for compressible multifluid and multiphase flows », *Journal of Computational Physics*, vol. 150, no. 2, pp. 425–467, 1999 (cit. on p. 28).
- [70] R. Klein and O. Pauluis, « Thermodynamic consistency of a pseudoincompressible approximation for general equations of state », *Journal of the Atmospheric Sciences*, vol. 69, no. 3, pp. 961–968, 2012 (cit. on p. 33).
- [71] M. S. Plesset and A. Prosperetti, « Bubble dynamics and cavitation », *Annual review of fluid mechanics*, vol. 9, no. 1, pp. 145–185, 1977 (cit. on p. 34).
- [72] L. Rayleigh, « On the pressure developed in a liquid during the collapse of a spherical cavity », *Philosophical Magazine*, vol. Series 6, no. 34, pp. 94–98, 1917 (cit. on p. 34).
- [73] J. B. Bell and D. L. Marcus, « A second-order projection method for variable-density flows », *Journal of Computational Physics*, vol. 101, no. 2, pp. 334–348, 1992 (cit. on p. 35).
- [74] J. Brackbill, D. B. Kothe, and C. Zemach, « A continuum method for modeling surface tension », *Journal of computational physics*, vol. 100, no. 2, pp. 335–354, 1992 (cit. on pp. 35, 71).
- [75] B. Lafaurie, C. Nardone, R. Scardovelli, S. Zaleski, and G. Zanetti, « Modelling merging and fragmentation in multiphase flows with SURFER », *Journal of Computational Physics*, vol. 113, no. 1, pp. 134–147, 1994 (cit. on pp. 35, 72, 150).
- [76] R. Defay, I. Prigogine, A. Bellemans, and D. Everett, *Surface Tension and Adsorption*. Longmans, Green, 1966 (cit. on p. 35).
- [77] T. Young, « III. An essay on the cohesion of fluids 95 Philosophical Transactions of the Royal Society of London <http://doi.org/10.1098/rstl.1805.0005> », 1805 (cit. on p. 36).
- [78] P. Laplace, « Mécanique céleste, Supplement to the tenth edition », *Courcier, Paris*, vol. 7, 1806 (cit. on p. 36).
- [79] C. E. Weatherburn, *Differential geometry of three dimensions*. Cambridge University Press, 1955, vol. 1 (cit. on p. 38).
- [80] M. Pettigrew, C. Taylor, and B. Kim, « Vibration of tube bundles in two-phase cross-flow: Part 1-Hydrodynamic mass and damping », *Journal of pressure vessel technology*, vol. 111, no. 4, pp. 466–477, 1989 (cit. on pp. 40, 60).
- [81] J. Weickert, « A review of nonlinear diffusion filtering », *Scale-space theory in computer vision*, pp. 1–28, 1997 (cit. on p. 40).
- [82] F. Catte, P.-L. Lions, J.-M. Morel, and T. Coll, « Image selective smoothing and edge detection by nonlinear diffusion », *SIAM Journal on Numerical analysis*, vol. 29, no. 1, pp. 182–193, 1992 (cit. on p. 40).

- [83] L. Alvarez and L. Mazorra, « Signal and image restoration using shock filters and anisotropic diffusion », *SIAM Journal on Numerical Analysis*, vol. 31, no. 2, pp. 590–605, 1994 (cit. on p. 40).
- [84] S.-H. Lee and J. K. Seo, « Noise removal with Gauss curvature-driven diffusion », *IEEE Transactions on Image Processing*, vol. 14, no. 7, pp. 904–909, 2005 (cit. on p. 41).
- [85] C. Olsson and Y. Boykov, « Curvature-based regularization for surface approximation », in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, IEEE, 2012, pp. 1576–1583 (cit. on p. 41).
- [86] Y. Gong, « Bernstein filter: A new solver for mean curvature regularized models », in *Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on*, IEEE, 2016, pp. 1701–1705 (cit. on p. 41).
- [87] J. L. Lagrange, *Essai d’une nouvelle methode pour de’terminer les maxima, et les minima des formules integrales indefinies*. 1761 (cit. on p. 41).
- [88] J. B. Meusnier, « Memoire sur la courbure des surfaces », *Mem des savan etrangers*, vol. 10, no. 1776, pp. 477–510, 1785 (cit. on p. 41).
- [89] M. Sussman and M. Ohta, « A stable and efficient method for treating surface tension in incompressible two-phase flow », *SIAM Journal on Scientific Computing*, vol. 31, no. 4, pp. 2447–2471, 2009 (cit. on pp. 41, 74).
- [90] A. Jarauta, P. Ryzhakov, J. Pons-Prats, and M. Secanell, « An implicit surface tension model for the analysis of droplet dynamics », *Journal of Computational Physics*, vol. 374, pp. 1196–1218, 2018 (cit. on p. 41).
- [91] S. Hysing, « A new implicit surface tension implementation for interfacial flows », *International Journal for Numerical Methods in Fluids*, vol. 51, no. 6, pp. 659–672, 2006 (cit. on p. 41).
- [92] P. Perona and J. Malik, « Scale-space and edge detection using anisotropic diffusion », *IEEE Transactions on pattern analysis and machine intelligence*, vol. 12, no. 7, pp. 629–639, 1990 (cit. on p. 41).
- [93] A. Witkin, « Scale-space filtering: A new approach to multi-scale description », in *Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP’84.*, IEEE, vol. 9, 1984, pp. 150–153 (cit. on p. 41).
- [94] T. Lindeberg, « Scale-space theory: A basic tool for analyzing structures at different scales », *Journal of applied statistics*, vol. 21, no. 1-2, pp. 225–270, 1994 (cit. on p. 41).
- [95] I. T. Young and L. J. Van Vliet, « Recursive implementation of the Gaussian filter », *Signal processing*, vol. 44, no. 2, pp. 139–151, 1995 (cit. on pp. 41, 43, 44, 46, 49).
- [96] I. T. Young, L. J. Van Vliet, and M. Van Ginkel, « Recursive gabor filtering », *IEEE Transactions on Signal Processing*, vol. 50, no. 11, pp. 2798–2805, 2002 (cit. on pp. 41, 49).

- [97] M. A. Branch, T. F. Coleman, and Y. Li, « A subspace, interior, and conjugate gradient method for large-scale bound-constrained minimization problems », *SIAM Journal on Scientific Computing*, vol. 21, no. 1, pp. 1–23, 1999 (cit. on p. 43).
- [98] B. Triggs and M. Sdika, « Boundary conditions for Young-van Vliet recursive filtering », *IEEE Transactions on Signal Processing*, vol. 54, no. 6, pp. 2365–2367, 2006 (cit. on pp. 44–47).
- [99] G. Taubin, « Curve and surface smoothing without shrinkage », in *Computer Vision, 1995. Proceedings., Fifth International Conference on*, IEEE, 1995, pp. 852–857 (cit. on p. 49).
- [100] —, « A signal processing approach to fair surface design », in *Proceedings of the 22nd annual conference on Computer graphics and interactive techniques*, ACM, 1995, pp. 351–358 (cit. on p. 50).
- [101] K. Brenner, C. Cancès, and D. Hilhorst, « Finite volume approximation for an immiscible two-phase flow in porous media with discontinuous capillary pressure », *Computational Geosciences*, vol. 17, no. 3, pp. 573–597, 2013 (cit. on p. 51).
- [102] A. Michel, « A finite volume scheme for two-phase immiscible flow in porous media », *SIAM Journal on Numerical Analysis*, vol. 41, no. 4, pp. 1301–1317, 2003 (cit. on p. 51).
- [103] R. Eymard, T. Gallouet, and R. Herbin, « Finite volume methods », *Handbook of numerical analysis*, vol. 7, pp. 713–1018, 2000 (cit. on p. 51).
- [104] W. V. D. Hodge, *The theory and applications of harmonic integrals*. CUP Archive, 1941 (cit. on p. 51).
- [105] H. Helmholtz, « Über Integrale der hydrodynamischen Gleichungen, welche den Wirbelbewegungen entsprechen. », ger, *Journal für die reine und angewandte Mathematik*, vol. 55, pp. 25–55, 1858 (cit. on p. 51).
- [106] H. Bhatia, G. Norgard, V. Pascucci, and P.-T. Bremer, « The Helmholtz-Hodge Decomposition; A Survey », *Visualization and Computer Graphics, IEEE Transactions on*, vol. 19, no. 8, pp. 1386–1404, 2013 (cit. on p. 51).
- [107] A. J. Chorin, « Numerical solution of the Navier-Stokes equations », *Mathematics of computation*, vol. 22, no. 104, pp. 745–762, 1968 (cit. on p. 51).
- [108] P. M. Gresho and R. L. Sani, « On pressure boundary conditions for the incompressible Navier-Stokes equations », *International Journal for Numerical Methods in Fluids*, vol. 7, no. 10, pp. 1111–1145, 1987 (cit. on p. 52).
- [109] G. Taylor and A. Green, « Mechanism of the production of small eddies from large ones », *Proceedings of the Royal Society of London. Series A, Mathematical and Physical Sciences*, vol. 158, no. 895, pp. 499–521, 1937 (cit. on p. 53).
- [110] J. B. Bell, P. Colella, and H. M. Glaz, « A second-order projection method for the incompressible Navier-Stokes equations », *Journal of Computational Physics*, vol. 85, no. 2, pp. 257–283, 1989 (cit. on p. 53).

- [111] J Kim and P Moin, « Application of a fractional-step method to incompressible Navier-Stokes equations », *Journal of Computational Physics*, vol. 59, no. 2, pp. 308–323, 1985 (cit. on pp. 53, 54).
- [112] D. L. Brown, R. Cortez, and M. L. Minion, « Accurate projection methods for the incompressible Navier–Stokes equations », *Journal of computational physics*, vol. 168, no. 2, pp. 464–499, 2001 (cit. on pp. 53, 54).
- [113] J.-L. Guermond and A. Salgado, « A splitting method for incompressible flows with variable density based on a pressure Poisson equation », *Journal of Computational Physics*, vol. 228, no. 8, pp. 2834–2846, 2009 (cit. on p. 53).
- [114] X.-D. Liu, R. P. Fedkiw, and M. Kang, « A boundary condition capturing method for Poisson’s equation on irregular domains », *Journal of computational Physics*, vol. 160, no. 1, pp. 151–178, 2000 (cit. on p. 55).
- [115] R. P. Fedkiw, T. Aslam, B. Merriman, and S. Osher, « A non-oscillatory Eulerian approach to interfaces in multimaterial flows (the ghost fluid method) », *Journal of computational physics*, vol. 152, no. 2, pp. 457–492, 1999 (cit. on p. 55).
- [116] J.-M. Hong, T. Shinar, M. Kang, and R. Fedkiw, « On boundary condition capturing for multiphase interfaces », *Journal of Scientific Computing*, vol. 31, no. 1-2, pp. 99–125, 2007 (cit. on p. 55).
- [117] H. A. Van der Vorst, « Bi-CGSTAB: A fast and smoothly converging variant of Bi-CG for the solution of nonsymmetric linear systems », *SIAM Journal on scientific and Statistical Computing*, vol. 13, no. 2, pp. 631–644, 1992 (cit. on pp. 57, 93).
- [118] M. M. Francois, S. J. Cummins, E. D. Dendy, D. B. Kothe, J. M. Sicilian, and M. W. Williams, « A balanced-force algorithm for continuous and sharp interfacial surface tension models within a volume tracking framework », *Journal of Computational Physics*, vol. 213, no. 1, pp. 141–173, 2006 (cit. on p. 59).
- [119] R. J. Leveque and Z. Li, « The immersed interface method for elliptic equations with discontinuous coefficients and singular sources », *SIAM Journal on Numerical Analysis*, vol. 31, no. 4, pp. 1019–1044, 1994 (cit. on p. 59).
- [120] J. W. Wan and X.-D. Liu, « A Boundary Condition–Capturing Multigrid Approach to Irregular Boundary Problems », *SIAM Journal on Scientific Computing*, vol. 25, no. 6, pp. 1982–2003, 2004 (cit. on p. 59).
- [121] M. J. Pettigrew and C. E. Taylor, « Two-Phase Flow-Induced Vibration: An Overview », *Journal of Pressure Vessel Technology*, vol. Vol. 116, p.233–253, 1994 (cit. on p. 60).
- [122] C. Taylor, I. Currie, M. Pettigrew, and B. Kim, « Vibration of tube bundles in two-phase cross-flow: Part 3-Turbulence induced excitation », *Journal of pressure vessel technology*, vol. 111, no. 4, pp. 488–500, 1989 (cit. on p. 60).

- [123] M. Pettigrew, J. Tromp, C. Taylor, and B. Kim, « Vibration of tube bundles in two-phase cross-flow: Part 2-Fluid-elastic instability », *Journal of Pressure Vessel Technology*, vol. 111, no. 4, pp. 478–487, 1989 (cit. on p. 60).
- [124] S. Granger, R. Campistron, and J. Lebreton, « Motion-dependent excitation mechanisms in a square in-line tube bundle subject to water cross-flow : an experimental modal analysis. », *Journal of Fluids and Structures*, vol. Vol. 7, p.521–550, 1993 (cit. on p. 60).
- [125] R. Mittal and G. Iaccarino, « Immersed boundary methods », *Annu. Rev. Fluid Mech.*, vol. 37, pp. 239–261, 2005 (cit. on pp. 60, 114, 126).
- [126] C. Taylor, M. Pettigrew, and I. Currie, « Random excitation forces in tube bundles subjected to two-phase cross-flow », *Journal of pressure vessel technology*, vol. 118, no. 3, pp. 265–277, 1996 (cit. on p. 62).
- [127] F. H. Harlow and J. E. Welch, « Numerical calculation of time-dependent viscous incompressible flow of fluid with free surface », *The physics of fluids*, vol. 8, no. 12, pp. 2182–2189, 1965 (cit. on pp. 65, 156).
- [128] W. Blaschke, *Elementare differentialgeometrie*. Springer-Verlag, 1921, vol. 1 (cit. on p. 70).
- [129] J. U. Brackbill and D. B. Kothe, « Dynamical modeling of surface tension », *Los Alamos National Laboratory*, 1996 (cit. on p. 71).
- [130] S. Popinet, « An accurate adaptive solver for surface-tension-driven interfacial flows », *Journal of Computational Physics*, vol. 228, no. 16, pp. 5838–5866, 2009 (cit. on p. 71).
- [131] S Chandrasekhar, *Hydrodynamic and hydromagnetic stability*, D. Publications, Ed. New York, 1981 (cit. on pp. 72, 151, 161).
- [132] S. Popinet, « Numerical models of surface tension », *Annual Review of Fluid Mechanics*, vol. 50, pp. 49–75, 2018 (cit. on p. 72).
- [133] J. G. Charney, R. Fjortoft, and J. v. Neumann, « Numerical integration of the barotropic vorticity equation », *Tellus*, vol. 2, no. 4, pp. 237–254, 1950 (cit. on p. 73).
- [134] M. Kang, R. P. Fedkiw, and X.-D. Liu, « A boundary condition capturing method for multiphase incompressible flow », *Journal of Scientific Computing*, vol. 15, no. 3, pp. 323–360, 2000 (cit. on pp. 74, 75, 107).
- [135] L. Freret, O. Thomine, J. Reveillon, S. De Chaisemartin, F. Laurent, and M. Massot, « On the role of preferential segregation in flame dynamics in polydisperse evaporating sprays », in *Proceedings of the Summer Program*, 2010, pp. 383–392 (cit. on p. 76).
- [136] J. P. Boris and D. L. Book, « Flux-corrected transport. I. SHASTA, a fluid transport algorithm that works », *Journal of computational physics*, vol. 11, no. 1, pp. 38–69, 1973 (cit. on p. 76).

- [137] P. K. Sweby, « High resolution schemes using flux limiters for hyperbolic conservation laws », *SIAM journal on numerical analysis*, vol. 21, no. 5, pp. 995–1011, 1984 (cit. on p. 76).
- [138] R. Courant, K. Friedrichs, and H. Lewy, « On the partial difference equations of mathematical physics », *IBM journal*, vol. 11, no. 2, pp. 215–234, 1967 (cit. on p. 77).
- [139] P. D. Lax, « Weak solutions of nonlinear hyperbolic equations and their numerical computation », *Communications on pure and applied mathematics*, vol. 7, no. 1, pp. 159–193, 1954 (cit. on p. 77).
- [140] S. K. Godunov, « A difference method for numerical calculation of discontinuous solutions of the equations of hydrodynamics », *Matematicheskii Sbornik*, vol. 89, no. 3, pp. 271–306, 1959 (cit. on pp. 77, 105).
- [141] P. Lax and B. Wendroff, « Systems of conservation laws », *Communications on Pure and Applied mathematics*, vol. 13, no. 2, pp. 217–237, 1960 (cit. on p. 78).
- [142] R. Warming and R. M. Beam, « Upwind second-order difference schemes and applications in aerodynamic flows », *AIAA Journal*, vol. 14, no. 9, pp. 1241–1249, 1976 (cit. on p. 78).
- [143] A. Harten, « High resolution schemes for hyperbolic conservation laws », *Journal of computational physics*, vol. 49, no. 3, pp. 357–393, 1983 (cit. on pp. 79, 82).
- [144] C. Chalons and F. Coquel, « Modified Suliciu relaxation system and exact resolution of isolated shock waves », *Mathematical Models and Methods in Applied Sciences*, vol. 24, no. 05, pp. 937–971, 2014 (cit. on p. 80).
- [145] C. Chalons and P. Goatin, « Godunov scheme and sampling technique for computing phase transitions in traffic flow modeling », *Interfaces and Free Boundaries*, vol. 10, no. 2, pp. 197–221, 2008 (cit. on p. 80).
- [146] J. Glimm, « Solutions in the large for nonlinear hyperbolic systems of equations », *Communications on pure and applied mathematics*, vol. 18, no. 4, pp. 697–715, 1965 (cit. on p. 80).
- [147] P. L. Roe, « Approximate Riemann solvers, parameter vectors, and difference schemes », *Journal of computational physics*, vol. 43, no. 2, pp. 357–372, 1981 (cit. on p. 80).
- [148] A. J. Chorin, « Random choice solution of hyperbolic systems », *Journal of Computational Physics*, vol. 22, no. 4, pp. 517–533, 1976 (cit. on p. 80).
- [149] R. J. LeVeque, « High-resolution conservative algorithms for advection in incompressible flow », *SIAM Journal on Numerical Analysis*, vol. 33, no. 2, pp. 627–665, 1996 (cit. on pp. 80, 108).
- [150] P. Roe and M. Baines, « Algorithms for advection and shock problems », in *Numerical Methods in Fluid Mechanics*, 1982, pp. 281–290 (cit. on pp. 81, 84).

- [151] P. L. Roe, « Some contributions to the modelling of discontinuous flows », pp. 163–193, 1985 (cit. on pp. 81, 84).
- [152] O. Bokanowski and H. Zidani, « Anti-dissipative schemes for advection and application to Hamilton–Jacobi–Bellmann equations », *Journal of Scientific Computing*, vol. 30, no. 1, pp. 1–33, 2007 (cit. on pp. 81, 84, 147, 165).
- [153] L. F. Richardson, « IX. The approximate arithmetical solution by finite differences of physical problems involving differential equations, with an application to the stresses in a masonry dam », *Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character*, vol. 210, no. 459–470, pp. 307–357, 1911 (cit. on p. 86).
- [154] B. Despres and F. Lagoutière, « Contact discontinuity capturing schemes for linear advection and compressible gas dynamics », *Journal of Scientific Computing*, vol. 16, no. 4, pp. 479–524, 2001 (cit. on p. 88).
- [155] A. Harten, B. Engquist, S. Osher, and S. R. Chakravarthy, « Uniformly high order accurate essentially non-oscillatory schemes, III », *Journal of Computational Physics*, vol. 131, no. 1, pp. 3–47, 1987 (cit. on p. 89).
- [156] J. Crank and P. Nicolson, « A practical method for numerical evaluation of solutions of partial differential equations of the heat-conduction type », in *Mathematical Proceedings of the Cambridge Philosophical Society*, Cambridge Univ Press, vol. 43, 1947, pp. 50–67 (cit. on p. 90).
- [157] U. Ghia, K. N. Ghia, and C. Shin, « High-Re solutions for incompressible flow using the Navier-Stokes equations and a multigrid method », *Journal of computational physics*, vol. 48, no. 3, pp. 387–411, 1982 (cit. on p. 91).
- [158] S. Albensoeder and H. C. Kuhlmann, « Accurate three-dimensional lid-driven cavity flow », *Journal of Computational Physics*, vol. 206, no. 2, pp. 536–558, 2005 (cit. on p. 91).
- [159] H. C. Ku, R. S. Hirsh, and T. D. Taylor, « A pseudospectral method for solution of the three-dimensional incompressible Navier-Stokes equations », *Journal of Computational Physics*, vol. 70, no. 2, pp. 439–462, 1987 (cit. on pp. 91, 92).
- [160] T. Banachiewicz, « Principes d’une nouvelle technique de la méthode des moindres carrés. Méthode de résolution numérique des équations linéaires, du calcul des déterminants et des inverses, et de réduction des formes quadratiques », *Bulletin international de l’Académie polonaise des sciences et des lettres, Classe des sciences mathématiques, Série A*, pp. 134–135, 393–404, 1938 (cit. on p. 92).
- [161] A. Brandt, « Multi-level adaptive solutions to boundary-value problems », *Mathematics of computation*, vol. 31, no. 138, pp. 333–390, 1977 (cit. on pp. 92, 98).
- [162] P. Wesseling, *Introduction To Multigrid Methods*. John Wiley & Sons, Ltd., 1995 (cit. on p. 92).
- [163] M. R. Hestenes and E. Stiefel, « Methods of conjugate gradients for solving linear systems », vol. 49, no. 1, 1952 (cit. on p. 93).

- [164] Y. Saad and M. H. Schultz, « GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems », *SIAM Journal on scientific and statistical computing*, vol. 7, no. 3, pp. 856–869, 1986 (cit. on p. 93).
- [165] P. Sonneveld and M. B. Van Gijzen, « IDR (s): A family of simple and fast algorithms for solving large nonsymmetric systems of linear equations », *SIAM Journal on Scientific Computing*, vol. 31, no. 2, pp. 1035–1062, 2008 (cit. on p. 93).
- [166] Y. Saad, *Iterative methods for sparse linear systems*. SIAM, 2003 (cit. on p. 93).
- [167] M. Mohr and R. Wienands, « Cell-centred multigrid revisited », *Computing and Visualization in Science*, vol. 7, no. 3-4, pp. 129–140, 2004 (cit. on p. 97).
- [168] A. Brandt and O. E. Livne, *Multigrid techniques: 1984 guide with applications to fluid dynamics*. SIAM, 2011, vol. 67 (cit. on p. 98).
- [169] M. Sussman, A. S. Almgren, J. B. Bell, P. Colella, L. H. Howell, and M. L. Welcome, « An adaptive level set approach for incompressible two-phase flows », *Journal of Computational Physics*, vol. 148, no. 1, pp. 81–124, 1999 (cit. on p. 98).
- [170] W. Rider, D. Kothe, E. Puckett, and I. Aleinov, « Accurate and robust methods for variable density incompressible flows with discontinuities », *Barriers and Challenges in Computational Fluid Dynamics*, (Kluwer Academic Publishers, 1998, Boston, MA, ISBN 0-7923-4855-9) V. Venkatakrishnan, MD Salas, and SR Chakravarthy, Editors, pp. 213–230, 1998 (cit. on pp. 98, 146).
- [171] M. S. Dodd and A. Ferrante, « A fast pressure-correction method for incompressible two-fluid flows », *Journal of Computational Physics*, vol. 273, pp. 416–434, 2014 (cit. on p. 103).
- [172] V. Fuka, « PoisFFT—A free parallel fast Poisson solver », *Applied Mathematics and Computation*, vol. 267, pp. 356–364, 2015 (cit. on p. 103).
- [173] M. Pippig, « PFFT: An extension of FFTW to massively parallel architectures », *SIAM Journal on Scientific Computing*, vol. 35, no. 3, pp. C213–C236, 2013 (cit. on p. 103).
- [174] M. Frigo, « A fast Fourier transform compiler », in *Acm sigplan notices*, ACM, vol. 34, 1999, pp. 169–180 (cit. on p. 103).
- [175] S. Lie, « Theorie der transformationsgruppen I », *Mathematische Annalen*, vol. 16, no. 4, pp. 441–528, 1880 (cit. on p. 104).
- [176] H. F. Trotter, « On the product of semi-groups of operators », *Proceedings of the American Mathematical Society*, vol. 10, no. 4, pp. 545–551, 1959 (cit. on p. 104).
- [177] G. Strang, « On the construction and comparison of difference schemes », *SIAM Journal on Numerical Analysis*, vol. 5, no. 3, pp. 506–517, 1968 (cit. on p. 105).
- [178] —, « Accurate partial difference methods I: Linear Cauchy problems », *Archive for Rational Mechanics and Analysis*, vol. 12, no. 1, pp. 392–402, 1963 (cit. on p. 105).

- [179] P Csomós, I Faragó, and Á Havasi, « Weighted sequential splittings and their analysis », *Computers & Mathematics with Applications*, vol. 50, no. 7, pp. 1017–1031, 2005 (cit. on p. 105).
- [180] S. T. Zalesak, « Fully multidimensional flux-corrected transport algorithms for fluids », *Journal of computational physics*, vol. 31, no. 3, pp. 335–362, 1979 (cit. on p. 108).
- [181] W. J. Rider and D. B. Kothe, « Reconstructing volume tracking », *Journal of computational physics*, vol. 141, no. 2, pp. 112–152, 1998 (cit. on p. 111).
- [182] S. Popinet, « Gerris: a tree-based adaptive solver for the incompressible Euler equations in complex geometries », *Journal of Computational Physics*, vol. 190, no. 2, pp. 572–600, 2003 (cit. on p. 111).
- [183] Q. Zhang and T. Hisada, « Studies of the strong coupling and weak coupling methods in FSI analysis », *International Journal for Numerical Methods in Engineering*, vol. 60, no. 12, pp. 2013–2029, 2004 (cit. on p. 114).
- [184] M. F. Barone and J. L. Payne, « Methods for simulation-based analysis of fluid-structure interaction », *SANDIA Report No. SAND2005-6573*, 2005 (cit. on p. 114).
- [185] N. M. Newmark, « A method of computation for structural dynamics », *Journal of the Engineering Mechanics Division*, vol. 85, no. 3, pp. 67–94, 1959 (cit. on p. 115).
- [186] J. Mohd-Yusofz, « Combined immersed-boundary/B-spline methods for simulations of flow in complex geometries », *Center for Turbulence Research, Annual Research Briefs*, 1997 (cit. on p. 114).
- [187] E. A. Fadlun, R. Verzicco, P. Orlandi, and J. Mohd-Yusof, « Combined immersed-boundary finite-difference methods for three-dimensional complex flow simulations », *Journal of computational physics*, vol. 161, no. 1, pp. 35–60, 2000 (cit. on p. 114).
- [188] S. Majumdar, G. Iaccarino, and P. Durbin, « RANS solvers with adaptive structured boundary non-conforming grids », *Annual Research Briefs, Center for Turbulence Research, Stanford University*, pp. 353–466, 2001 (cit. on p. 114).
- [189] R.-J. Gibert, *Vibrations des structures. Interactions avec les fluides, sources d'excitation aléatoires*. Collection De La Direction Des Études et Recherches D'électricité De France, 1988 (cit. on p. 115).
- [190] R. W. Numrich and J. Reid, « Co-arrays in the next Fortran Standard », in *ACM SIGPLAN Fortran Forum*, ACM, vol. 24, 2005, pp. 4–17 (cit. on p. 118).
- [191] K. Salari and P. Knupp, « Code verification by the method of manufactured solutions », Sandia National Labs., Tech. Rep., 2000 (cit. on p. 124).

- [192] M. Belliard, M. Chandesris, J. Dumas, Y. Gorsse, D. Jamet, C. Josserand, and B. Mathieu, « An analysis and an affordable regularization technique for the spurious force oscillations in the context of direct-forcing immersed boundary methods », *Computers & Mathematics with Applications*, vol. 71, no. 5, pp. 1089–1113, 2016 (cit. on pp. 126, 173).
- [193] M. Belliard, M. Chandesris, J. Dumas, Y. Gorsse, D. Jamet, C. Josserand, and B. Mathieu, « Analysis and remedy for spurious force oscillations in the context of direct forcing methods », 2014 (cit. on p. 127).
- [194] E. Guilmineau and P. Queutey, « A numerical simulation of vortex shedding from an oscillating circular cylinder », *Journal of Fluids and Structures*, vol. 16, no. 6, pp. 773–794, 2002 (cit. on p. 132).
- [195] M. Schafer, S. Turek, F. Durst, E. Krause, and R. Rannacher, « Benchmark computations of laminar flow around a cylinder », in *Flow simulation with high-performance computers II*, Springer, 1996, pp. 547–566 (cit. on p. 132).
- [196] J. Volker, « Reference values for drag and lift of a two-dimensional time-dependent flow around a cylinder », *International Journal for Numerical Methods in Fluids*, vol. 44, no. 7, pp. 777–788, 2004 (cit. on p. 132).
- [197] S. Nugent and H. A. Posch, « Liquid drops and surface tension with smoothed particle applied mechanics », *Phys. Rev. E*, vol. 62, pp. 4968–4975, 4 Oct. 2000 (cit. on p. 138).
- [198] A. Tartakovsky and P. Meakin, « Modeling of surface tension and contact angles with smoothed particle hydrodynamics », *Phys. Rev. E*, vol. 72, p. 026 301, 2 Aug. 2005 (cit. on p. 138).
- [199] L. Rayleigh, « On the capillary phenomena of jets », *Royal Society*, vol. 29, no. 196-199, pp. 71–97, 1879 (cit. on p. 138).
- [200] H. Lamb, *Hydrodynamics*. Cambridge university press, 1932 (cit. on pp. 138, 145).
- [201] S. Park, « The splitting of drops and bubbles by turbulent fluid flow », *Journal of Fluids Engineering MARCH*, p. 53, 1973 (cit. on p. 138).
- [202] M. Sussman, K. M. Smith, M. Y. Hussaini, M. Ohta, and R Zhi-Wei, « A sharp interface method for incompressible two-phase flows », *Journal of computational physics*, vol. 221, no. 2, pp. 469–505, 2007 (cit. on p. 138).
- [203] D. E. Fyfe, E. S. Oran, and M. Fritts, « Surface tension and viscosity with Lagrangian hydrodynamics on a triangular mesh », *Journal of Computational Physics*, vol. 76, no. 2, pp. 349–384, 1988 (cit. on p. 138).
- [204] G Tryggvason, B Bunner, O Ebrat, and W Tauber, « Computations of multiphase flows by a finite difference/front tracking method. I. Multi-fluid flows », *Lecture Series-von Karman Institute For Fluid Dynamics*, pp. 7–7, 1998 (cit. on pp. 138, 139).

- [205] D. Torres and J. Brackbill, « The point-set method: front-tracking without connectivity », *Journal of Computational Physics*, vol. 165, no. 2, pp. 620–644, 2000 (cit. on p. 138).
- [206] S. Shin and D. Juric, « Modeling three-dimensional multiphase flow using a level contour reconstruction method for front tracking without connectivity », *Journal of Computational Physics*, vol. 180, no. 2, pp. 427–470, 2002 (cit. on pp. 145, 147).
- [207] E. Becker, W. Hiller, and T. Kowalewski, « Experimental and theoretical investigation of large-amplitude oscillations of liquid droplets », *Journal of Fluid Mechanics*, vol. 231, pp. 189–210, 1991 (cit. on p. 145).
- [208] A. Prosperetti, « Free oscillations of drops and bubbles: the initial-value problem », *Journal of Fluid Mechanics*, vol. 100, no. 2, pp. 333–347, 1980 (cit. on p. 145).
- [209] T. Abadie, J. Aubin, and D. Legendre, « On the combined effects of surface tension force calculation and interface advection on spurious currents within Volume of Fluid and Level Set frameworks », *Journal of Computational Physics*, vol. 297, pp. 611–636, 2015 (cit. on pp. 146, 150).
- [210] D. Jamet, D. Torres, and J. Brackbill, « On the theory and computation of surface tension: the elimination of parasitic currents through energy conservation in the second-gradient method », *Journal of Computational Physics*, vol. 182, no. 1, pp. 262–276, 2002 (cit. on pp. 146, 150).
- [211] D. Jacqmin, « A variational approach to deriving smeared-interface surface tension models », in *Barriers and Challenges in Computational Fluid Dynamics*, Springer, 1998, pp. 231–240 (cit. on p. 146).
- [212] E. Aulisa, S. Manservigi, and R. Scardovelli, « A novel representation of the surface tension force for two-phase flow with reduced spurious currents », *Computer Methods in Applied Mechanics and Engineering*, vol. 195, no. 44, pp. 6239–6257, 2006 (cit. on p. 146).
- [213] M. Meier, G. Yadigaroglu, and B. L. Smith, « A novel technique for including surface tension in PLIC-VOF methods », *European Journal of Mechanics-B/Fluids*, vol. 21, no. 1, pp. 61–73, 2002 (cit. on p. 146).
- [214] G. Tryggvason, B. Bunner, A. Esmaeeli, D. Juric, N. Al-Rawahi, W. Tauber, J. Han, S. Nas, and Y.-J. Jan, « A front-tracking method for the computations of multiphase flow », *Journal of Computational Physics*, vol. 169, no. 2, pp. 708–759, 2001 (cit. on pp. 147, 150).
- [215] D. Jamet, O. Lebaigue, N. Coutris, and J. Delhay, « The second gradient method for the direct numerical simulation of liquid–vapor flows with phase change », *Journal of Computational Physics*, vol. 169, no. 2, pp. 624–651, 2001 (cit. on p. 150).

- [216] D. H. Sharp, « An overview of Rayleigh-Taylor instability », *Physica D: Nonlinear Phenomena*, vol. 12, no. 1-3, 3IN111–10IN1018, 1984 (cit. on p. 151).
- [217] L. Rayleigh, « Investigation of the Character of the Equilibrium of an Incompressible Heavy Fluid of Variable Density », *Proceedings of the London Mathematical Society*, vol. s1-14, no. 1, pp. 170–177, Nov. 1882 (cit. on p. 151).
- [218] G. I. Taylor, « The instability of liquid surfaces when accelerated in a direction perpendicular to their planes. I », *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences*, vol. 201, no. 1065, pp. 192–196, 1950 (cit. on p. 151).
- [219] K. O. Mikaelian, « Rayleigh-Taylor instability in finite-thickness fluids with viscosity and surface tension », *Physical Review E*, vol. 54, no. 4, p. 3676, 1996 (cit. on p. 151).
- [220] H.-J. Kull, « Theory of the Rayleigh-Taylor instability », *Physics Reports*, vol. 206, no. 5, pp. 197–325, 1991 (cit. on p. 151).
- [221] G. Tryggvason and S. O. Unverdi, « Computations of three-dimensional Rayleigh-Taylor instability », *Physics of Fluids A: Fluid Dynamics*, vol. 2, no. 5, pp. 656–659, 1990 (cit. on p. 151).
- [222] D. Livescu, T. Wei, and M. Petersen, « Direct numerical simulations of Rayleigh-Taylor instability », in *Journal of Physics: Conference Series*, IOP Publishing, vol. 318, 2011, p. 082007 (cit. on p. 151).
- [223] P. G. Drazin and W. H. Reid, *Hydrodynamic stability*. Cambridge university press, 2004 (cit. on p. 151).
- [224] A. d. Saint-Venant, « Theorie du mouvement non permanent des eaux, avec application aux crues des rivières et à l'introduction de marées dans leurs lits », *Comptes rendus des séances de l'Académie des Sciences*, vol. 36, pp. 174–154, 1871 (cit. on p. 156).
- [225] L. Lobovsky, E. Botia-Vera, F. Castellana, J. Mas-Soler, and A. Souto-Iglesias, « Experimental investigation of dynamic pressure loads during dam break », *Journal of Fluids and Structures*, vol. 48, pp. 407–434, 2014 (cit. on p. 156).
- [226] A. Colagrossi and M. Landrini, « Numerical simulation of interfacial flows by smoothed particle hydrodynamics », *Journal of computational physics*, vol. 191, no. 2, pp. 448–475, 2003 (cit. on p. 156).
- [227] B. Muller, « Low-mach-number asymptotics of the Navier-Stokes equations », in *Floating, Flowing, Flying*, Springer, 1998, pp. 97–109 (cit. on p. 156).
- [228] A. Almgren, J. Bell, C. Rendleman, and M. Zingale, « Low Mach number modeling of type Ia supernovae. I. Hydrodynamics », *The Astrophysical Journal*, vol. 637, no. 2, p. 922, 2006 (cit. on p. 156).
- [229] C.-D. Munz, S. Roller, R. Klein, and K. J. Geratz, « The extension of incompressible flow solvers to the weakly compressible regime », *Computers & Fluids*, vol. 32, no. 2, pp. 173–196, 2003 (cit. on p. 156).

- [230] L. F. Menabrea, *Note sur les effets du choc de l'eau dans les conduites*. Mallet-Bachelier, 1858 (cit. on p. 159).
- [231] J. Michaud, « Coups de bélier dans les conduites. Étude des moyens employés pour en atténuer les effets », *Bull. Soc. Vaudoise Ing. Arch*, vol. 4, no. 3, p. 4, 1878 (cit. on p. 159).
- [232] R. Krasny, « Desingularization of periodic vortex sheet roll-up », *Journal of Computational Physics*, vol. 65, no. 2, pp. 292–313, 1986 (cit. on p. 160).
- [233] M. Van Dyke, *An album of fluid motion*. The Parabolic Press, 1982 (cit. on p. 160).
- [234] J. Kim, « A continuous surface tension force formulation for diffuse-interface models », *Journal of Computational Physics*, vol. 204, no. 2, pp. 784–804, 2005 (cit. on p. 160).
- [235] C. K. Batchelor and G. Batchelor, *An introduction to fluid dynamics*. Cambridge university press, 1967 (cit. on p. 161).
- [236] D. W. Moore, « The spontaneous appearance of a singularity in the shape of an evolving vortex sheet », *Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences*, vol. 365, no. 1720, pp. 105–119, 1979 (cit. on p. 161).
- [237] G. R. Baker, D. I. Meiron, and S. A. Orszag, « Generalized vortex methods for free-surface flow problems », *Journal of Fluid Mechanics*, vol. 123, pp. 477–501, 1982 (cit. on p. 161).
- [238] M. J. Berger and P. Colella, « Local adaptive mesh refinement for shock hydrodynamics », *Journal of computational Physics*, vol. 82, no. 1, pp. 64–84, 1989 (cit. on p. 164).
- [239] J. Hua and J. Lou, « Numerical simulation of bubble rising in viscous liquid », *Journal of Computational Physics*, vol. 222, no. 2, pp. 769–795, 2007 (cit. on p. 165).
- [240] S.-R. Hysing, S. Turek, D. Kuzmin, N. Parolini, E. Burman, S. Ganesan, and L. Tobiska, « Quantitative benchmark computations of two-dimensional bubble dynamics », *International Journal for Numerical Methods in Fluids*, vol. 60, no. 11, pp. 1259–1288, 2009 (cit. on pp. 165, 167).
- [241] D. Kim and P. Moin, « Pressure-correction algorithm to solve Poisson systems with constant coefficients for fast two-phase simulations », *Center for Turbulence Research Annual Research Briefs*, 2009 (cit. on p. 167).
- [242] T. Colin, C.-H. Bruneau, and S. Tancogne, « Simulation of the break-up of a diphasic jet in a microchannel », 2008 (cit. on p. 177).
- [243] G. Li, Y. Lian, Y. Guo, M. Jemison, M. Sussman, T. Helms, and M. Arienti, « Incompressible multiphase flow and encapsulation simulations using the moment-of-fluid method », *International Journal for Numerical Methods in Fluids*, vol. 79, no. 9, pp. 456–490, 2015 (cit. on p. 180).

- [244] L. Goudenège, A. Larat, J. Llobell, M. Massot, D. Mercier, O. Thomine, and A. Vié, « Statistical and probabilistic modeling of a cloud of particles coupled with a turbulent fluid », *ESAIM: Proceedings and Surveys*, vol. 65, pp. 401–424, 2019 (cit. on p. 180).
- [245] CEA, *sourceforge.net/projects/trust-platform*, 2015 (cit. on p. 182).
- [246] —, *sourceforge.net/projects/triocfd*, 2015 (cit. on p. 182).

Titre : Simulation et modélisation des interactions fluide-structure en écoulements diphasiques

Mots clés : interaction fluide-structure ; écoulements diphasiques ; méthodes de frontières immergées ; méthodes anti-diffusives ; simulation numérique

Résumé : Les générateurs de vapeur (GV) sont l'un des composants majeurs des réacteurs nucléaires, et une connaissance approfondie de leur comportement constitue un enjeu industriel important aussi bien pour le concepteur que pour l'exploitant EDF. Une des problématiques rencontrées pour le dimensionnement des GV concerne la vibration des tubes induite par l'écoulement, ce qui nécessite une évaluation raisonnable de la réponse des tubes à l'excitation provoquée par le fluide environnant. La zone identifiée comme la plus critique est la région en U (partie haute du GV) où l'écoulement est diphasique avec un fort taux de vide et interagit plutôt transversalement avec les tubes. Afin d'évaluer les excitations générées sur les tubes par les fluctuations inhérentes à l'écoulement, les paramètres physiques pertinents doivent être identifiés. Pour les écoulements monophasiques, il semble possible de relier les efforts exercés sur les structures tubulaires au niveau de turbulence de l'écoulement ; à la fois en utilisant des méthodes de réduction des données expérimentales mais également en utilisant des méthodes de simulations numériques. Pour les écoulements diphasiques, les forces induites sur les tubes par l'écoulement ont a priori une autre origine et seraient plutôt liées aux contributions dynamiques de chaque phase ainsi qu'aux transferts interfaciaux (fluctuations de pression liées au passage des discontinuités). Néanmoins, les paramètres physiques pertinents qui permettent de prévoir l'amplitude de ces forces restent lar-

gement débattus (taux de vide, régime d'écoulement, etc.) et les mécanismes physiques mal compris. Pour étudier ces instabilités vibratoires lorsque l'écoulement est diphasique, un certain nombre d'expériences analytiques ont été et continuent d'être menées au CEA. Ces expériences analytiques portent sur un tube isolé ou en faisceau, rigide ou flexible, et sur une large gamme de régimes d'écoulement (maquettes AMOVI et DIVA du CEA). Leur objectif est de caractériser ces instabilités vibratoires (mesure des forces exercées sur l'obstacle) en fonction de paramètres globaux de l'écoulement (débit gaz, débit liquide, taux de vide "moyen", etc.) mais aussi de certains paramètres locaux (taux de vide local, taille des bulles, vitesse gaz, etc.). Ces paramètres mesurés ou estimés localement sont ceux qui permettent d'obtenir les adimensionnements les plus pertinents à la fois sur les forces d'excitations aléatoires (spectres d'excitation en diphasique sur tube rigide) et sur les forces de couplage fluide-élastiques (tube flexible seul puis en faisceau). Il reste néanmoins une bande de dispersion sur les résultats obtenus, les mécanismes physiques sont mal compris et ces adimensionnements restent tributaires du choix de la localisation des mesures. L'objectif de la thèse est donc de mettre en oeuvre des simulations numériques avec suivi d'interface dans des configurations proches de celles des expériences analytiques menées au CEA afin d'approfondir l'analyse des phénomènes conduisant aux vibrations des tubes de GV.

Title : Simulation and modelling of fluid-structure interactions in two-phase flows

Keywords : fluid-structure interaction ; two-phase flows ; immersed boundary methods ; anti-diffusive methods ; numerical simulation

Abstract : Steam generators are a key component of nuclear power reactors, and an in-depth knowledge of their mechanisms is a major industrial challenge for the designer and the operator EDF. Vibration of tube bundles induced by cross-flow is one of the problems encountered by the designer, thus needing to assess the vibration response to the excitation generated by the flow. The critical region is the U shape of the bundle (upper part of the steam generator), where two-phase cross-flow occurs with an important void fraction. In order to measure excitation induced by flow fluctuations on the tube bundle, some physical parameters have to be identified. For single-phase flows, it seems possible to link load on tubular structure to turbulence intensity of the flow, thanks to experimental data reduction methods together with numerical simulation methods. For two-phase flows, it is believed that forces induced on the tubes by the flow have other origins, and might be connected to dynamic contribution of each phase together with energy interfacial transfers (pressure fluctuations induced by density discontinuities). Nevertheless, relevant physical parameters which could predict the amplitude of the forces remain a subject of debate (void fraction, flow regime, etc.) and physical processes not yet fully understood.

In order to study mechanical instabilities in two-phase flows, some analytic experiments have been and continue to be conducted at CEA. These analytic experiments focus on isolated tube or tube bundles (rigid or flexible), and on a large regime flow range (AMOVI and DIVA mockups at CEA). They aim to describe these mechanical instabilities (forces measurement on the obstacle) based upon average parameters of the flow (gas and liquid flow rates, "mean" void fraction, etc.), but also local parameters (local void fraction, bubble size, gas velocity, etc.). These measured or locally estimated parameters are used to conduct relevant nondimensionalization, both on the random excitation forces (two-phase excitation spectrum on a rigid tube) and the fluid-elastic coupling forces (single flexible tube or flexible bundle). Nonetheless, some dispersion remains on the results, physical mechanisms are not well understood, and the nondimensionalization process remains dependent on metrology. The aim of this PhD thesis is to conduct numerical simulations with interface capture methods in configurations close to the experiments conducted at CEA in order to expand the knowledge on phenomena leading to vibration of tube bundle in steam generators.