A penalization method for the simulation of bubbly flows

Antoine Morente, Jérôme Laviéville, Dominique Legendre

A penalization method for the simulation of bubbly flows

Antoine Morente [a,b], Jérôme Laviéville [a], Dominique Legendre [b,*]

[a] EDF (Électricité de France) R&D, 6 Quai Watier, 78400 Chatou, France
[b] Institut de Mécanique des Fluides de Toulouse, IMFT, Université de Toulouse, CNRS – Toulouse, France

Keywords:
Bubbly flows
Penalization method
Shear free condition
Bubble induced agitation

This work is devoted to the development of a penalization method for the simulation of bubbly flows. Spherical bubbles are considered as moving penalized obstacles interacting with the fluid and a numerical method for ensuring the shear free condition at the liquid–bubble interface is proposed. Three test-cases (curved channel, inclined channel and 3D translating bubble) are used to validate the accuracy of the discretization ensuring the slip condition at the interface. Numerical simulations of a rising bubble in a quiescent liquid are performed for moderate Reynolds numbers. Considering bubble terminal velocities, initial accelerations and wake decay, the effect of the penalization viscosity used to ensure a uniform velocity in the penalized object is discussed. Finally, simulations of bubble swarms have been carried out in a fully periodic box with a large range of void fractions from 1% to 15%. The statistics provided by the simulations characterizing the bubble-induced agitation are found in remarkable agreement with the experiments.

1. Introduction

Building reliable simulation tools able to predict a wide range of multiphase flows regimes is a major challenge in industry. Today, such simulations of boiling flows (dispersed or separate phases) can be carried out using the local 3D CFD code NEPTUNE_CFD [12] developed in the frame of the project NEPTUNE (EDF, CEA, Areva, IRSN). Making accurate simulations of bubbly flows is of importance for providing closures laws on momentum transfer, bubble–liquid interaction and induced turbulence. Bunner and Tryggvason [5,6] and Esmaeeli and Tryggvason [9] carried out simulations with non-deformable and deformable bubbles providing PDFs of the bubble velocity. Roghair et al. [28] developed a new drag correlation for bubbles in bubble swarms at intermediate and high Reynolds numbers through their numerical simulations. Direct numerical simulations (DNS) were performed by Roghair et al. [29] to study the behavior of a swarm of rising air bubbles for the comparison of the liquid energy spectra and bubble velocity probability density functions (PDFs) with experimental data.

To get a local description of the flow, the DNS simulation approaches of liquid–bubble flows are widely used. The notable variation among all the DNS approaches concerns the representation and the numerical treatment of the bubbles interface: the body fitted approach proposed by McLaughlin [19] for a single bubble; the front tracking method by Tryggvason and Unverdi [32,35] requiring markers on the interface. Two major drawbacks arise from this kind of approach: a clear limitation on both the Reynolds number and the number of bubbles. DNS of bubbly flows are often limited to moderate bubble Reynolds numbers ($Re = O(10–100)$) for spherical bubbles [32,41]. Recently, the simulation of 16 rising bubbles at $Re =$

* Corresponding author.
E-mail address: legendre@imft.fr (D. Legendre).

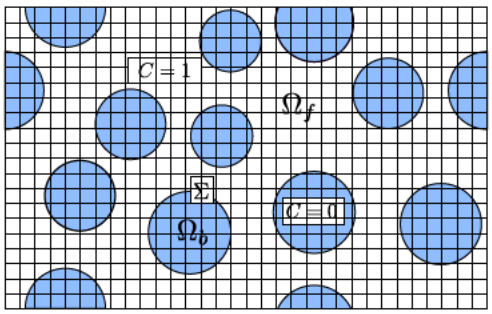


Fig. 1. Sketch of the representation of Lagrangian bubbles distributed in the eulerian grid.

1000 have been performed by Roghair et al. [29], still being quite far from the high void fraction concentrations observed in most of the industrial configurations.

Some alternative methods have been explored to get rid of these limitations. New avenues of investigation concern the representation of the bubbles and their interface in the flow, and the numerical method used to solve the flow field around them, both being related. Immersed boundary methods (IBM) know a growing interest due to their easy implementation (mostly based on cartesian grids) and use. Two approaches can be distinguished: the discrete forcing approach and the continuous forcing approach [21]. In the discrete approach, the interface is spread out on the cells of the fluid grid. Interpolation schemes are used to impose a given boundary condition at the interface. As an example, in the frame of IBM discrete approaches, [16] proposed an interpolation scheme satisfying the no-slip condition on the boundary for the simulation of flows over or inside complex geometries. The second approach is based on the use of Dirac distributions for the treatment of transfer of quantities between the fluid and markers located at the interface [33,14].

IBM methods have been developed for simulations involving bubbles or solid particles. Uhlmann and coworkers performed the simulation of the sedimentation of 1000 rigid particles in a periodic box [33,34] and a direct numerical investigation of dilute turbulent particulate flows in a vertical plane channel by considering thousands of finite-size rigid particles with a forcing approach [23,24]. The application of IBM methods to bubbly flows is more recent: Schwarz et al. [30] performed the simulation of bubbles with varying shape at low Reynolds number. Kempe et al. [15] performed simulations of spherical and ellipsoidal bubbles in purified fluids with slip conditions.

We present in this study a new numerical approach based on the use of an IBM method for the simulation of bubbly flows. Our objective is to propose a numerical method able to reproduce the main characteristics of bubbly flows for moderate to large void fractions without having a detailed description (as DNS can do) of the flow close to the bubble. In comparison with other approaches in the literature, this method does not require to manage markers at the interface. We use a first order volume penalization method [2,37,38]. The bubbles are represented as penalized spherical and non-deformable objects whose movement in the fluid is determined by the hydrodynamics forces acting on them. Concerning the representation of the interface, we use a classical VoF method [4,36] with additional local specific geometrical quantities such as surface fractions, in order to locally characterize the interface in each cell. We propose an adapted discretization of the Navier–Stokes equations based on those quantities, in order to take into account the presence of a shear free condition at the interface. The general methodology of the discretization process is inspired from cut-cell methods [7,40], and in a more general way this discretization process is fully part of the IBM (Immersed Boundary Method) family.

This paper is organized as follows. In section 2, we describe the general numerical method. The local characterization of the interface and its advection are detailed. The governing equations, i.e. the penalized problem is also presented and the adapted discretization (divergence, mass fluxes, diffusive term) ensuring the slip condition is detailed. Then, three test cases used for the validation and the convergence order of the method are reported in section 3. The simulations of a single bubble rising in a quiescent liquid is described is section 4. Finally, bubble swarm simulations are compared with experiments in section 5.

2. Numerical method

The studied domain Ω contains N_b non-deformable spherical bubbles moving in a continuous fluid at velocities $\mathbf{u_{b}}_i$ $(i = 1, .., N_b)$. In the following the subscripts i will be dropped for simplicity. We define Ω_f the domain occupied by the fluid, Ω_b the penalized domain occupied by the bubbles and Σ the liquid bubble interface (Fig. 1).

2.1. Bubble description and VoF function

We denote by r_b the bubbles radius and $\mathbf{x_b} = (x_b, y_b, z_b)$ the bubble center position. The bubble interface is then defined at time $t_n = n\Delta t$ where Δt is the time step by

$$F^n(x, y, z) = (x - x_b^n)^2 + (y - y_b^n)^2 + (z - z_b^n)^2 - {r_b}^2 = 0 \tag{1}$$

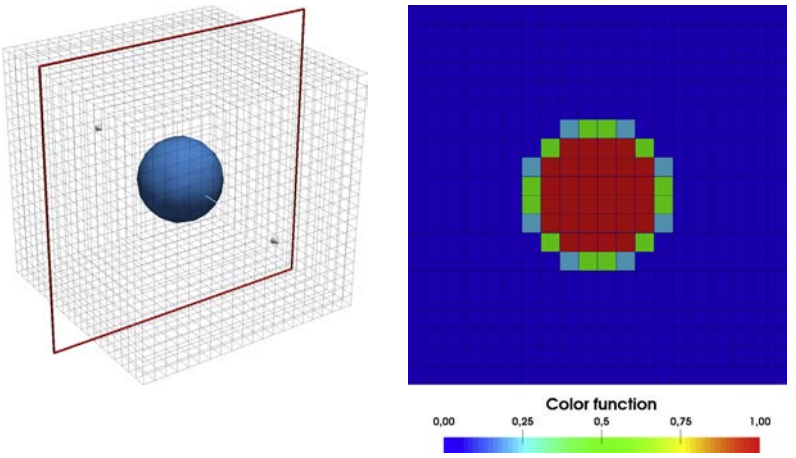


Fig. 2. Use of the Bna et al. [4] method for the interface volume tracking. (a) Contours of the volume fraction for a single bubble; (b) volume fraction in the corresponding cut plane.

Table 1
Single bubble test case for the volume conservation. The relative errors ϵ at the end of the simulation and the CPU time t_s of the simulation are reported for the four grids considered.

	20^3	40^3	80^3	160^3
ϵ	4.5×10^{-16}	2.4×10^{-17}	3.1×10^{-18}	3.9×10^{-19}
t_s (ms)	35	52	84	103

The bubbles are transported in a Lagrangian way. When the velocity $\mathbf{u_b}^{n+1}$ of a bubble at time t_{n+1} is calculated (see section 2.8), the bubble mass center $\mathbf{x_b}^n$ is then determined using the first order Euler explicit scheme:

$$\mathbf{x_b}^{n+1} = \mathbf{x_b}^n + \Delta t\, \mathbf{u_b}^{n+1} \tag{2}$$

Once, the updated position of the bubble is known, the bubble surface F^{n+1} at time $n+1$ is calculated through equation (1). We introduce the classical VoF function C as the volume fraction of the continuous fluid. The VoF function C^{n+1} at time $n+1$ is computed from the interface position F^{n+1} using the VoF function initialization found in Bna et al. [4] (Fig. 2). We stress that the VoF function C is not transported here but determined from the bubble surface equation F. We also define $\alpha_f = C$ and $\alpha_b = 1 - C$ the volume fraction of the liquid and the gas, respectively.

We tested the method on a single bubble test case to evaluate its precision and in particular the mass loss during the simulation. For this test, we consider a spherical bubble with a radius $r_b = 0.2$ m centered in a 1 m $\times$ 1 m $\times$ 1 m cubic computational domain. A uniform grid of N cells is used and the cell size is h_0. The computational domain is periodic. The bubble is initially located at the center of the computational domain. At t_0 in a uniform velocity $\mathbf{u_b} = (0.10, 0.15, 0.20)$ m/s is imposed to the bubble. The error and CPU measurements (ϵ and t_s) are made at t_f, when the bubble has traveled a distance corresponding to 1 m. The velocity of the bubble is chosen to avoid any case of favorable alignment with a raw of cells.

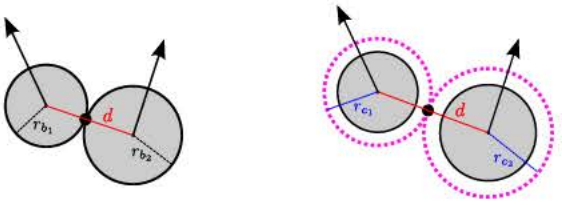
The volume of the bubble is measured as

$$V_C = h_0^3 \sum_{i=1}^{N} C_i \tag{3}$$

and we introduce the relative L_1 error ϵ defined as:

$$\epsilon = \frac{|V_{th} - V_C|}{V_{th}} \tag{4}$$

where V_{th} is the theoretical volume. Four different grids 20^3, 40^3, 80^3 and 160^3 are considered, leading to the ratio of cells per radius 4, 8, 16 and 32, respectively. The relative errors ϵ is almost constant during the simulation and does not depend on the position of the bubble on the grid. The value of ϵ at the end of the simulation and the time t_s of the simulation are reported in Table 1. The results highlight three characteristics of the selected VoF method: the high precision of the color function computation, even for the smallest ratio of cells per radius, the decrease of the error with mesh refinement and the control of the CPU time.



(a) (b)

Fig. 3. Bubble collision treatment. (a) Classical collision with contact: $d = (r_{b_1} + r_{b_2})$; (b) collision with an artificial increase of the radii as used here: $d = (r_{c_1} + r_{c_2}) = k(r_{b_1} + r_{b_2})$.

2.2. Bubble interaction model

We detail in this section how are handled interactions between two bubbles getting close to each other during the simulation. We recall that coalescence effects are not incorporated in the method, and that the bubbles are non-deformable. We model the interaction between bubbles as elastic collisions. Collision are managed using an artificial increase of the radii of the colliding bubbles in order to avoid (i) contact between penalized domains and (ii) the presence of two or more interfaces in the same cell. Let's consider two colliding bubbles, bubble 1 and bubble 2, with radii and velocities noted r_{b_1} and r_{b_2}, $\mathbf{u_{b1}}$ and $\mathbf{u_{b2}}$. The velocities of the bubbles after the collision are noted $\mathbf{u}^*_{\mathbf{b}_1}$ and $\mathbf{u}^*_{\mathbf{b}_2}$. The elastic collision between the two bubbles are described through the conservation of momentum and kinetic energy:

$$\begin{aligned} m_1\mathbf{u_{b1}} + m_2\mathbf{u_{b2}} &= m_1\mathbf{u}^*_{\mathbf{b}_1} + m_2\mathbf{u}^*_{\mathbf{b}_2} \\ \frac{1}{2}m_1\mathbf{u_{b1}}.\mathbf{u_{b1}} + \frac{1}{2}m_2\mathbf{u_{b2}}.\mathbf{u_{b2}} &= \frac{1}{2}m_1\mathbf{u}^*_{\mathbf{b}_1}.\mathbf{u}^*_{\mathbf{b}_1} + \frac{1}{2}m_2\mathbf{u}^*_{\mathbf{b}_2}.\mathbf{u}^*_{\mathbf{b}_2} \end{aligned} \tag{5}$$

where the masses m_1 and m_2 to consider in the collision between bubbles are the added masses. For mono-dispersed bubbles, the mass can be simplified in these relations. We also introduce the parameter controlling the radius increase k so that the new radii of the two bubbles during the collision process writes as $r_{c_1} = k \times r_{b_1}$ and $r_{c_2} = k \times r_{b_2}$ (Fig. 3). During the simulation, a collision is detected when the distance d between two bubbles satisfies $d < (r_{c_1} + r_{c_2})$. To avoid with certainty dealing with contact configurations between the bubbles, we noted that $k = 1.05$ was a suitable value. An Event Driven algorithm has been selected to deal with all the collisions in a given time lapse corresponding to a time step. This approach relies on dealing with all collisional events as a chronological sequence of events. One of the very first description of Event Driven simulations was realized by Alder and Wainwright [1] for the simulations of hard-disk particle systems in the frame of a molecular dynamics method. Sigurgeirsson et al. [31] later brought improvements to the algorithm such as the reduction of computation cost. More details on the Event Driven algorithms and their alternative approaches are described in [39]. We sum up here the main steps of the algorithm:

1. Compute the time t_c corresponding to the next collisional event for all the bubbles.
2. All the bubbles are moved for a time corresponding to $t = t_c$.
3. The collision is treated by the collisional model and new velocities and positions are computed.
4. Update of the reference time $t_r = t_c$. Back to step **1**.

The whole algorithm developed for this work is fully detailed in [22]. One important aspect of the choice of the algorithm concerns its computational cost during the simulation. We noted that the CPU time required by the algorithm was less than 5% of the whole CPU time for the simulations of bubble swarms, which is a relatively low computational cost.

2.3. Governing equations: the penalization formulation

The fluid motion is governed by mass and momentum conservation equations. The fluids properties (viscosity and density) are considered as constant. The mass conservation equation writes:

$$\frac{\partial}{\partial t}(\alpha_f \rho_f) + \nabla.(\alpha_f \rho_f \mathbf{u}) = 0 \tag{6}$$

with α_f the fluid fraction as defined in section 2.1, ρ_f the liquid density and $\mathbf{u} = (u_x, u_y, u_z)$ its velocity. The presented volume penalization method [2] leads to the addition of a penalization term in the momentum equation for the fluid velocity. The objective is to set (by forcing) a specific behavior inside the bubble. Introducing the Lagrangian velocity of the bubble $\mathbf{u_b}$, the penalized momentum equation is then

$$\frac{\partial}{\partial t}(\alpha_f \rho_f \mathbf{u}) + \nabla.(\alpha_f \rho_f \mathbf{u}\mathbf{u}) = \nabla.(\alpha_f \tau_f) - \alpha_f \nabla P + \alpha_b K(\mathbf{u_b} - \mathbf{u}) \tag{7}$$

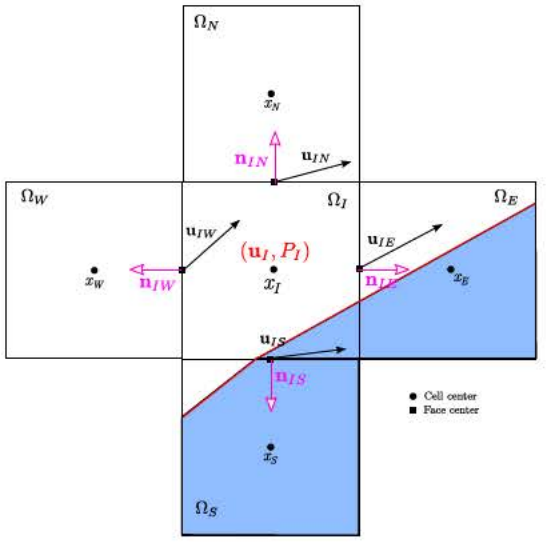


Fig. 4. Variables definition on a given set of volumes distributed around Ω_I. The penalized domain is shown in blue. (For interpretation of the colors in the figure(s), the reader is referred to the web version of this article.)

with P the pressure field, $\tau = \mu_f(\nabla \mathbf{u} + \nabla^T \mathbf{u})$ is the viscous stress tensor, μ_f the fluid viscosity and $K = \rho_b/\tau$ the penalization parameter defined using the density of the bubble ρ_b and a characteristic time τ. Its weighting by the bubble fraction α_b leads to:

$$\begin{aligned} &\alpha_b K = 0 \text{ in } \Omega_f \\ &\alpha_b K = \frac{\alpha_b \rho_b}{\tau} \text{ in } \Omega_b \end{aligned} \tag{8}$$

We tested the influence of the penalization parameter with the simulation of the monodisperse bubble swarm (see section 5) for 3 different values, $\tau = 10^{-6}$ s, 10^{-9} s and 10^{-14} s. The PDFs of both the fluid and the bubble velocity showed no influence of τ on the results. We also noted that the stability of the simulations were not impacted by the values of τ we tested. In this study, the parameter is set to $\tau = 10^{-14}$ s for all the simulations presented in the following sections.

In order to take into account the presence of moving bubbles in the flow, the differential operators of the Navier–Stokes must be adapted to the problem. Their original discretization inherited from NEPTUNE_CFD for solid object did not ensure the shear free condition at the liquid–bubble interface. The boundary condition at the bubble surface is described with a condition of impermeability and a zero tangential shear stress:

$$\mathbf{u}.\mathbf{n} = \mathbf{u_b}.\mathbf{n}, \quad (\tau_f)_T = 0 \tag{9}$$

where $\mathbf{n}$ is the normal at the penalized bubble surface and $(\tau_f)_T$ is defined as the tangential component of the viscous stress.

2.4. Discretization and notations

The general algorithm implemented in NEPTUNE_CFD [12] is based on the use of colocalized finite volume schemes. The domain Ω is described using N^3 uniform cells Ω_I. The velocity and the pressure are localized in the center of the cells. We detail in Fig. 4 the position of each variable of interest and the main notations used in this paper. For the sake of simplicity, the figure is presented in the plane (x, y). The penalized domain is shown in blue. Considering a cell Ω_I, we note x_I the center of the cell, $\mathbf{u}_I = (u_{Ix}, u_{Iy}, u_{Iz})$ the corresponding velocity that can be the fluid velocity or in the penalized velocity, and P_I the pressure. We note x_{IJ} the center of the face F_{IJ} separating the cells Ω_I and Ω_J. We introduce the set $V(I)$ containing the indices of the neighboring cells of Ω_I such as $V(I) = \{W, N, E, S\}$ (West, North, East and South cells, respectively Ω_W, Ω_N, Ω_E and Ω_S). Their corresponding cell centers are $\{x_W, x_N, x_E, x_S\}$. The velocities calculated on the corresponding faces F_{IJ} of center x_{IJ} and normal $\mathbf{n}_{IJ}$ are noted $\mathbf{u}_{IJ}$ with $J \in V(I)$.

We also introduce the area s_{IJ} intercepted by the penalized domain on the face F_{IJ} between the cells Ω_I and Ω_J (Fig. 5). The fluid surface fraction on F_{IJ} is then:

$$\alpha_{f,IJ} = 1 - \frac{s_{IJ}}{F_{IJ}} \tag{10}$$

The fluid surface fractions will be used further in this paper for the computation of the normals in each cell (see Equation (12)) and for improving the discretization of the fluxes between cells (see equations (13) and (20)). We present

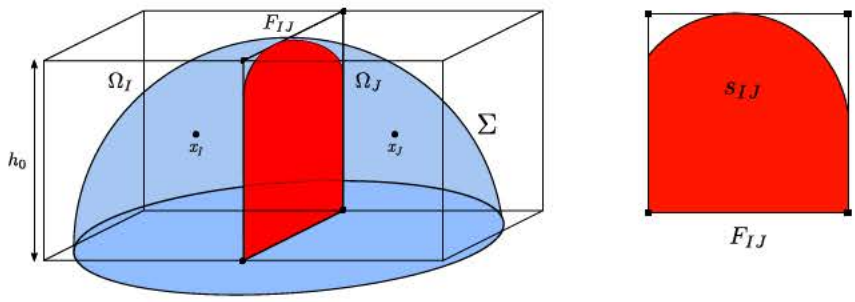


Fig. 5. The penalized surface of area S_{IJ} corresponding to the intersection between the face F_{IJ} (separating the cells Ω_I and Ω_J) and a bubble (blue) is shown in red.

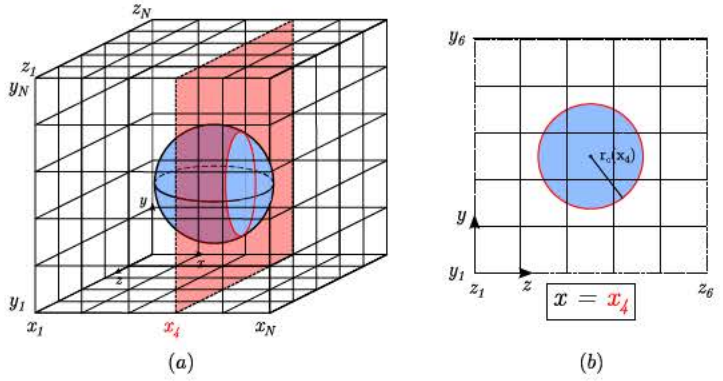


Fig. 6. (a) The plane in dash contains all vertices verifying $x = x_4$. (b) Pseudo-grid composed by all the faces in the plane $x = x_4$.

the method (considering spheres) developed to compute surface fractions in a 3D framework. Let's consider a Cartesian grid with N^3 cells and a spherical bubble with surface equation given by (1) (Fig. 6(a)). Given each vertex of the face, we can easily determine the intersection between the plane containing the face and the sphere. In case of intersection, this intersection is a circle of radius r_c. Joining each face on a given plane leads to the construction of a Cartesian pseudo-grid (Fig. 6(b)). The computation of s_{IJ} leads to a two-dimensional computation of a surface fraction for a given circle. These surface fractions are computed following the VoF function initialization from [4].

2.5. Mass fluxes and continuity equation

In a given volume crossed by the interface, the divergence free condition must be verified for the continuous liquid. The original discretization of the continuity equation (6) on a given cell Ω_I is set in NEPTUNE_CFD as:

$$\rho_f \Omega_I \frac{\alpha_{f,I}^{n+1} - \alpha_{f,I}^{n}}{\Delta t} + \rho_f \sum_{J \in V(I)} \alpha_{f,IJ}^{n} \mathbf{u}_{IJ}^{n}.\mathbf{n}_{IJ} F_{IJ} = 0 \tag{11}$$

The instationary term $\rho_f \frac{\alpha_{f,I}^{n+1} - \alpha_{f,I}^{n}}{\Delta t}$ is inherited from the multiphase formalism of NEPTUNE_CFD. The modification proposed here concerns the surface term $\rho_f \sum_{J \in V(I)} \alpha_{f,IJ}^{n} \mathbf{u}_{IJ}^{n}.\mathbf{n}_{IJ} F_{IJ}$. The contribution of the velocity of the penalized object $\mathbf{u_b}$ is now considered in the flux balance. The interface is described by a plane surface (Fig. 7). Based on the relation between the normals of each face involving the surface fractions $\alpha_{f,IJ}$ with each surrounding cell J, we compute the normal at the interface as:

$$\mathbf{n}_p^n = - \sum_{J \in V(I)} \alpha_{f,IJ}^{n} \mathbf{n}_{IJ} \tag{12}$$

The area of the interface in Ω_I is noted F_Σ. Adding the mass flux of the new face $\rho_f \mathbf{u}_\mathbf{b}^n.\mathbf{n}_p^n F_\Sigma$ to the total flux balance (11) leads to the following formulation:

$$\rho_f \Omega_I \frac{\alpha_{f,I}^{n+1} - \alpha_{f,I}^{n}}{\Delta t} + \rho_f \sum_{J \in V(I)} \alpha_{f,IJ}^{n} \mathbf{u}_{IJ}^{n}.\mathbf{n}_{IJ} F_{IJ} + \rho_f \mathbf{u}_\mathbf{b}^n.\mathbf{n}_p^n F_\Sigma^n = 0 \tag{13}$$

where $\mathbf{n}_p^n$ is determined using relation (12).

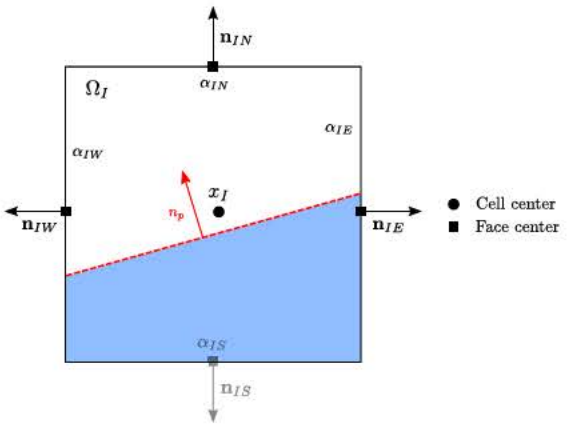


Fig. 7. Volume Ω_I containing both fluid and the penalized domain (in blue). The fluid surface fractions $\alpha_{f,IW}$, $\alpha_{f,IE}$ and $\alpha_{f,IN}$ reported in the figure are computed following relation (10) on each face in order to determine the normal $\mathbf{n}_p$ at the interface.

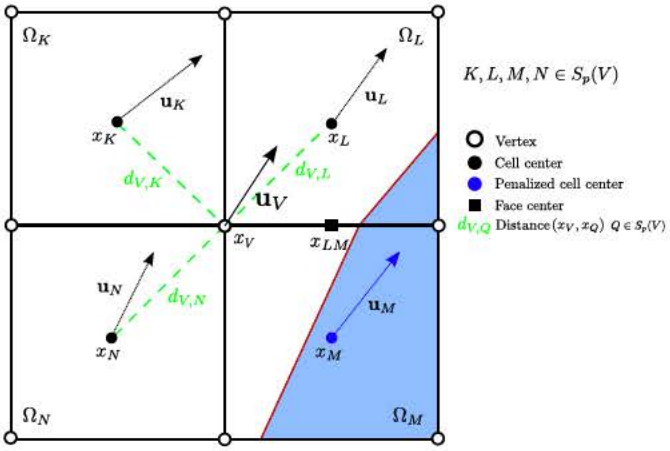


Fig. 8. Definition of the velocity $\mathbf{u}_V$ on the vertex x_V used to calculate the velocity at the face center x_{LM}. In this example, the cell center M is penalized and the set of non-penalized neighboring is $S_p(V) = \{K, L, N\}$.

We now detail the new computation of the velocities defined at the faces center used to evaluate the mass fluxes in relation (13). The proposed formulation has been devised to take into account the presence of the interface of moving penalized objects in the fluid. To describe the method, we consider two cells Ω_I and Ω_J, Ω_J being penalized. The original formulation inherited from NEPTUNE_CFD defines the computation of the velocity $\mathbf{u}_{IJ}$ at the face center as:

$$\frac{\mathbf{u}_{IJ} - \mathbf{u}_I}{d(x_I, x_{IJ})} = \frac{\mathbf{u}_J - \mathbf{u}_{IJ}}{d(x_J, x_{IJ})} \tag{14}$$

where $d(x_I, x_{IJ})$ and $d(x_J, x_{IJ})$ are respectively the distances between the centers x_I, x_{IJ} and x_J, x_{IJ}. Considering a uniform Cartesian grid, we have:

$$\mathbf{u}_{IJ} = \frac{\mathbf{u}_I + \mathbf{u}_J}{2} \tag{15}$$

This formulation shows its limitation when at least one of the two cells is penalized as shown in Fig. 8 where the whole face joining cells L and M is not totally immersed in the penalized domain. The fluid flux at the face F_{LM} is determined using the penalized velocity at x_M and not the correct fluid velocity in the partially penalized cell Ω_M. To improve the mass flux in such situation, we introduce a new vertex-cell connectivity to NEPTUNE_CFD. We define the set $S_p(V)$ defined as the set of non-penalized neighboring cells of a given vertex x_V. An example in shown in Fig. 8 where $S_p(V) = \{K, L, N\}$. The velocities $\mathbf{u}_V$ at the vertex x_V is computed as the average velocity of the non-penalized neighboring cells nodes as:

$$\mathbf{u}_V = \frac{\sum_{Q \in S_p(V)} d(x_V, x_Q)\mathbf{u}_Q}{\sum_{Q \in S_p(V)} d(x_V, x_Q)} \tag{16}$$

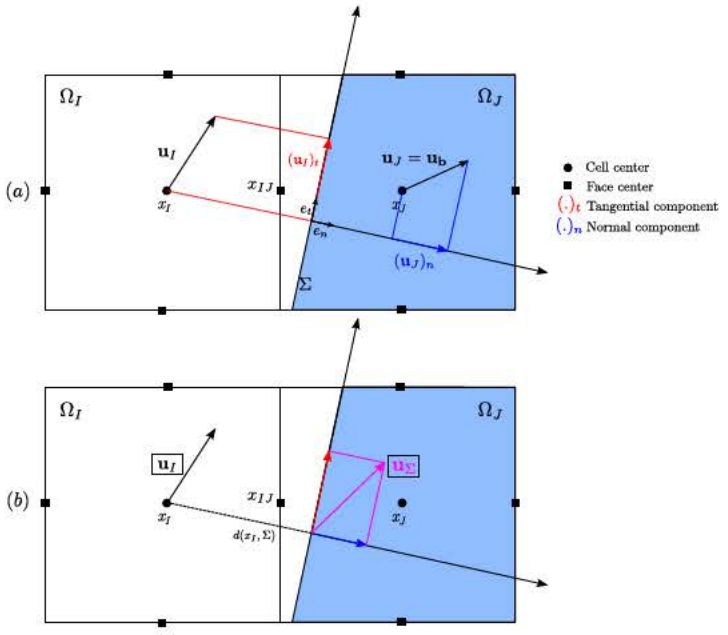


Fig. 9. Computation of the interface velocity $\mathbf{u}_\Sigma$ considering the cells Ω_I and Ω_J, Ω_J containing the interface. (a) The tangential velocity at the interface (red) is given by projection of the fluid velocity in cell Ω_I on the interface Σ while the normal motion of the interface (blue) is given by the velocity in the penalized cell Ω_J. (b) The velocity at the interface $\mathbf{u}_\Sigma$ (magenta) is then assembled with these components.

We introduce the set $N_p(IJ)$ listing the non-penalized neighboring vertices contained on a given face which center is x_{IJ}. The computation of the velocity $\mathbf{u}_{IJ}$ simply writes as the sum of the fluid velocities at the non-penalized neighboring vertices as:

$$\mathbf{u}_{IJ} = \frac{1}{|N_p(IJ)|} \sum_{V \in N_p(IJ)} \mathbf{u}_V \tag{17}$$

where $|N_p(IJ)|$ is the number of elements of $N_p(IJ)$.

2.6. The shear-free condition

We now detail the computation of the viscous flux between two cells Ω_I and Ω_J, when one of them is penalized, in order to ensure the local shear-free condition (9) at the interface. Fig. 9 shows such a situation, Ω_J being in this example the penalized cell. The cell Ω_J is crossed by the interface Σ. The original computation of the viscous shear stress $\tau_{f,IJ}$ used in NEPTUNE_CFD for the fluid at the face center x_{IJ} was the following:

$$\tau_{f,IJ} = \mu_f \frac{u_{Jy} - u_{Iy}}{d(x_I, x_J)} F_{IJ} \tag{18}$$

This formulation does not take into account the presence of the slip condition at the interface Σ, so it needs to be adapted. We introduce $\mathbf{u}_\Sigma = (u_{\Sigma x}, u_{\Sigma y}, u_{\Sigma z})$ the velocity at the interface Σ. $\mathbf{u}_\Sigma$ is decomposed into its normal and tangential components as shown in Fig. 9. The tangential component $(\mathbf{u}_I)_t$ is given by the projection of the fluid velocity $\mathbf{u}_I$ at node I on the tangential direction of the interface. The normal contribution $(\mathbf{u}_J)_n$ results from the normal displacement induced by the motion of the penalized zone so it is given by the normal projection on the interface of the penalized velocity $\mathbf{u_b}$ at node J. $\mathbf{u}_\Sigma$ is then calculated as

$$\mathbf{u}_\Sigma = (\mathbf{u}_I)_t + (\mathbf{u}_J)_n \tag{19}$$

Instead of using the velocity of the cell $\mathbf{u}_J$ in relation (18), the proposed adaptation of the computation of the fluid shear stress at the face center x_{IJ} relies on the use of $\mathbf{u}_\Sigma$. For the situation shown in Fig. 9 it writes:

$$\tau_{f,IJ} = \mu_f \frac{u_{\Sigma y} - u_{Iy}}{d(x_I, \Sigma)} \alpha_{f,IJ} F_{IJ} \tag{20}$$

where $d(x_I, \Sigma)$ is the distance between the cell center x_I and the interface Σ. Thanks to Eq. (19), the shear stress cancels when projected on the surface of the penalization domain and the shear-free condition (9) is then satisfied at the bubble surface.

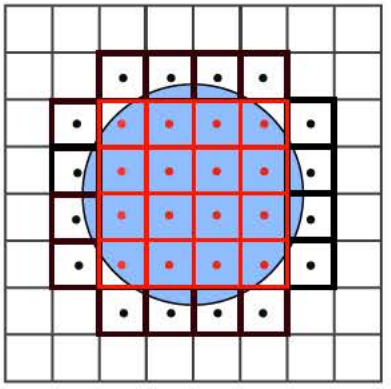


Fig. 10. Pressure regularization: example showing the domain Ω_P (in red) where is solved the pressure for one bubble and (in black) the external nodes used to impose the local Dirichlet boundary condition on $\partial\Omega_P$.

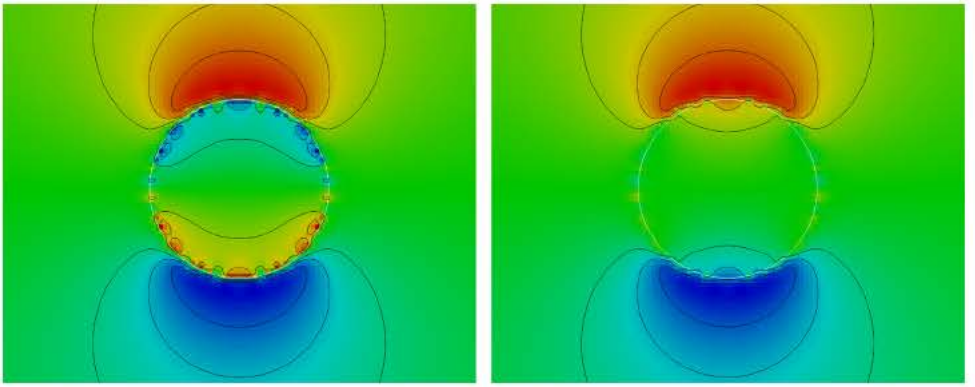


Fig. 11. Bubble translation in a fluid at rest at $Re = 10$. (Left) pressure field without pressure regularization; (right) pressure field with the new method for the pressure regularization.

2.7. Pressure regularization

Due to the penalization of the velocity inside the bubble, non-physical values of the pressure may occur inside the bubble. Such values may affect the pressure solution in the fluid domain because the pressure is solved in the entire domain. To avoid this effect, the pressure is regularized inside each bubble by solving a Laplace equation for the pressure P in the domain Ω_P defined by all the cells whose center is inside the considered bubble (the red nodes in Fig. 10). A Dirichlet boundary condition is imposed on the boundary $\partial\Omega_P$ of Ω_P. The value imposed for the Dirichlet boundary condition P_Σ is set by selecting for each face of $\partial\Omega_P$ the pressure of the closest non-penalized cells in the fluid containing the interface Σ (the black nodes in Fig. 10). The regularization of the pressure is performed by solving the following equation for each bubble:

$$\Delta P = 0 \quad \text{in } \Omega_P \tag{21}$$

$$P = P_\Sigma \quad \text{on } \partial\Omega_P \tag{22}$$

The finite volume method is applied on the system (21)–(22) for each bubble and a unique system of equations is formed. The resulting matrix is solved with a conjugate gradient method. The effect of the pressure regularization is clearly illustrated in Fig. 11. A 2D bubble of diameter $d_b = 1$ mm is moving with a uniform velocity of magnitude $u_{by} = 0.1$ m.s^{-1} in a fluid at rest. The fluid properties are $\rho_f = 1000$ kg.m^{-3} and $\mu_f = 0.01$ Pa.s such that the bubble Reynolds number is $Re = \rho_f d_b u_b / \mu_f = 10$. The computational domain is periodic in each direction. Results are shown at the final time $t_f = 0.2$ s when the flow around the bubble is stabilized.

2.8. Bubble–liquid coupling formulation

We present the formulation used in this study for the momentum coupling between the liquid and the bubbles. Instead of working with the original penalized momentum equation (7), the starting point of the coupling is a formulation similar

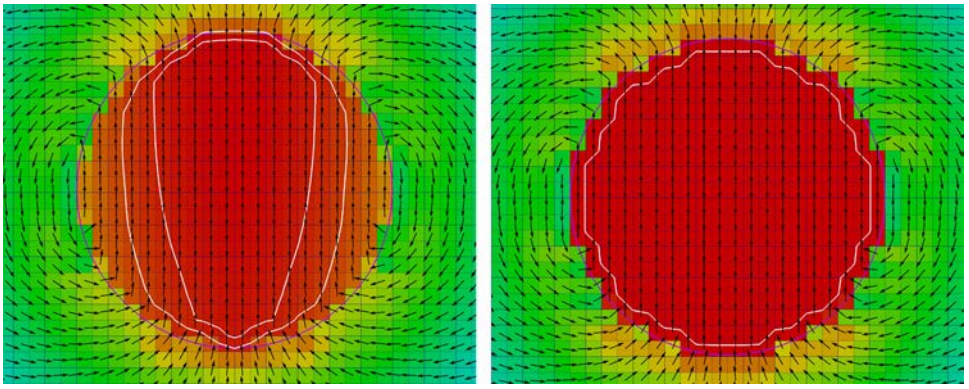


Fig. 12. Effect of the penalization viscosity μ_p on the velocity field inside the bubble; left figure: $\mu_p = 1$ Pa.s, right figure: $\mu_p = 100$ Pa.s. The contours of the velocity magnitude inside the bubble are plotted in white, the contour of the color fraction in purple.

to a 2-Fluid eulerian multiphase frame, in which a momentum equation for each phase, liquid and bubble, is weighted by their respective volume fraction α_f and α_b. The formulation of the system of equation is the following:

$$\begin{aligned}\alpha_f\rho_f\frac{\partial \mathbf{u}}{\partial t} + \alpha_f\rho_f\mathbf{u}.\nabla\mathbf{u} &= -\alpha_f\nabla P + \alpha_f\mu_f\nabla^2\mathbf{u} + \alpha_f\rho_f\mathbf{g} + \alpha_f\alpha_b\rho_b\frac{\mathbf{u_b}-\mathbf{u}}{\tau}\\ \alpha_b\rho_b\frac{\partial \mathbf{u_b}}{\partial t} + \alpha_b\rho_b\mathbf{u_b}.\nabla\mathbf{u_b} &= -\alpha_b\nabla P + \alpha_b\mu_p\nabla^2\mathbf{u_b} + \alpha_b\rho_b\mathbf{g} - \alpha_f\alpha_b\rho_b\frac{\mathbf{u_b}-\mathbf{u}}{\tau}\end{aligned} \tag{23}$$

The velocities appearing in the penalization term are considered at time $n+1$ for a more stable coupling between the velocity of the bubble and the hydrodynamic action of the liquid. The aim is now to express the penalization term $\alpha_b\rho_b(\mathbf{u_b}^{n+1} - \mathbf{u}^{n+1})/\tau$ in equation (7), given the fluid velocity $\mathbf{u}^n$ and $\mathbf{u}^{n+1}$ at time n and $n+1$, respectively. The contribution from the fluid on the bubbles is expressed from the liquid momentum and is injected in the bubble equation. Given that the convective term $\alpha_b\rho_b\mathbf{u_b}.\nabla\mathbf{u_b} = 0$ and the viscous term $\alpha_b\mu_p\nabla^2\mathbf{u_b} = 0$ because $\mathbf{u_b}$ is imposed to be uniform in the penalized bubble, we obtain the following formulation of the penalization term (see Appendix):

$$\begin{aligned}\rho_b\frac{\mathbf{u_b}^{n+1}-\mathbf{u}^{n+1}}{\tau} = \rho_b\frac{\Delta t}{\tau+\Delta t}\frac{\mathbf{u_b}^{n}-\mathbf{u}^{n+1}}{\Delta t} + \rho_f\frac{\Delta t}{\tau+\Delta t}\left[\frac{\mathbf{u}^{n+1}-\mathbf{u}^{n}}{\Delta t} + \mathbf{u}^{n+1}.\nabla\mathbf{u}^n - \mu_f\nabla^2\mathbf{u}^n\right]\\ + (\rho_b-\rho_f)\frac{\Delta t}{\tau+\Delta t}\mathbf{g}\end{aligned} \tag{24}$$

which is injected in the liquid momentum equation to obtain the reverse coupling in the fluid:

$$\begin{aligned}\rho_f\left(1-\alpha_b\frac{\Delta t}{\tau+\Delta t}\right)\left[\frac{\mathbf{u}^{n+1}-\mathbf{u}^{n}}{\Delta t} + \mathbf{u}^{n+1}.\nabla\mathbf{u}^n - \mu_f\nabla^2\mathbf{u}^n\right] = -\nabla P^n + \left(\rho_f + \alpha_b(\rho_b-\rho_f)\frac{\Delta t}{\tau+\Delta t}\right)\mathbf{g}\\ + \alpha_b\rho_b\frac{\mathbf{u_b}^{n}-\mathbf{u}^{n+1}}{\tau+\Delta t}\end{aligned} \tag{25}$$

The coupling method is defined by the semi-implicit system of equations (24) and (25). Solving this system gives the values of $\mathbf{u_b}^{n+1}$ and $\mathbf{u}^{n+1}$. A notable drawback of this coupling method concerns a non-uniform solution for the velocity field inside the bubble as shown in Fig. 12(left) after solving the system. The formulation of the penalized term $(\mathbf{u_b}^{n+1} - \mathbf{u}^{n+1})/\tau$ implies that a uniform velocity should be found inside the penalized region. To ensure that condition, we had to introduce a penalization viscosity μ_p. μ_f is imposed to μ_p in equation (25) inside the penalized domain. μ_p has not a physical meaning since a shear-free condition is imposed for the liquid at the bubble surface (see section 2.6). The role of μ_p is to set the velocity field inside the bubble as uniform as possible as shown in Fig. 12(right). The artificial penalization viscosity μ_p is used for the viscous stress calculation on the faces joining two penalized cells as shown in Fig. 13. We observed no influence of the penalization viscosity μ_p on the stability of the simulations. However the penalization viscosity μ_p was observed to impact the computational cost of the simulations. The CPU time of the simulations is increased when increasing μ_p due to the poor conditioning of the matrix involved in the resolution of the coupling method of NEPTUNE_CFD. The impact of the penalization viscosity μ_p on the results is further discussed in the following sections.

3. Validation of the numerical method

Test cases are now proposed to validate the enhancement of the penalization method for the simulation of bubbly flows. For that purpose, we will consider in this section the flow in a 2D curved channel with fixed boundaries, the flow

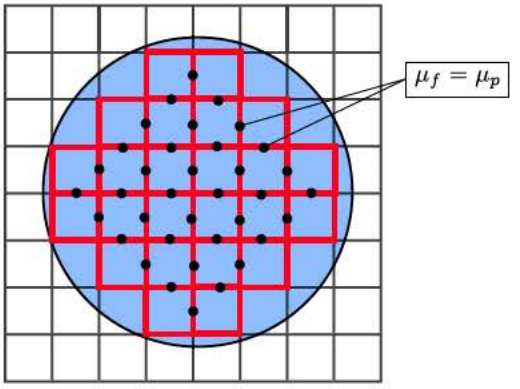


Fig. 13. Cell faces where the penalization viscosity μ_p is applied inside the penalized bubble.

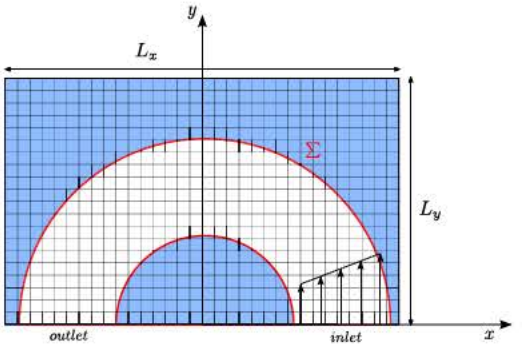


Fig. 14. Sketch of the curved channel test case, the penalized domain being in blue.

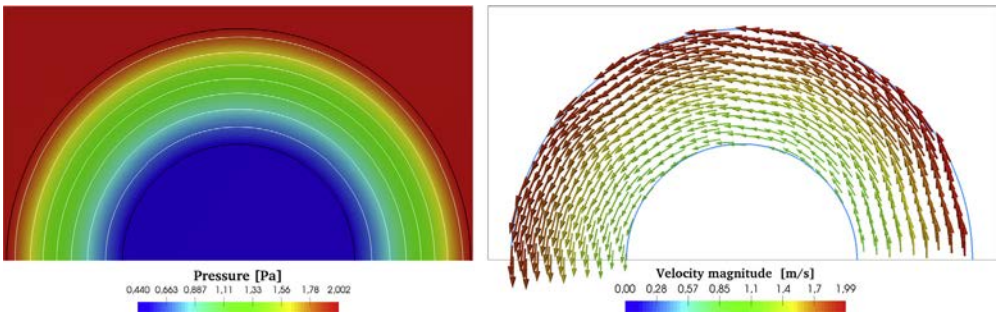
in a 2D inclined channel with moving wall and a 3D spherical bubble moving at the same velocity as the carrying fluid. The presented test-cases have been conducted in a common way. An exact analytical solution is used for the measure of the numerical error so that a convergence study is realized. The shear free condition is imposed at the interface of the penalized domain for all the cases. For some cases, the New Penalization Method (NPM) will be compared to the Original Penalization Method (OPM). The main changes between (NPM) and (OPM) are related to the divergence calculation (section 2.5), the shear free condition at the surface of the penalized objects (section 2.6) and the pressure regularization inside the penalized objects (section 2.7). The simulations have been carried out on regular grids and the grid spacing is noted Δx. Each convergence study has been realized at the constant CFL value $|\mathbf{u}|\Delta t/\Delta x = 0.5$. The penalization parameter value is set with $\tau = 10^{-14}$ s.

3.1. Flow inside a 2D curved channel

We consider a flow inside a 2D curved channel as shown in Fig. 14. Slip conditions are imposed at the surface of the penalized domain Ω_b and the velocity in the penalized domain is set to $\mathbf{u_b} = 0$. The aim of this test-case is to validate the new developments presented above, in particular the shear free condition on an curvilinear surface. The dimensions of the domain are $Lx = 4.4$ m and $L_y = Lx/2$. The radius of the inner circle is $R_1 = 1$ m and the radius of the outer circle is $R_2 = 2$ m so that the gap is $r = R_2 - R_1 = 1$ m. The initial conditions are imposed to $\mathbf{u}(t = 0, (x, y)) = 0$ and $P(x, y) = 0$ inside the fluid region Ω_f. The simulation starts at $t = 0$ by imposing the solid body rotation $u_x(x, y = 0) = 0$ and $u_y(x, y = 0) = Ax$ with $A = 1$ s^{-1} at the inlet of the channel. The pressure $P(x, y = 0) = \rho_f A^2 x^2/2$ is imposed at the outlet. The final steady state solution in the fluid domain is then

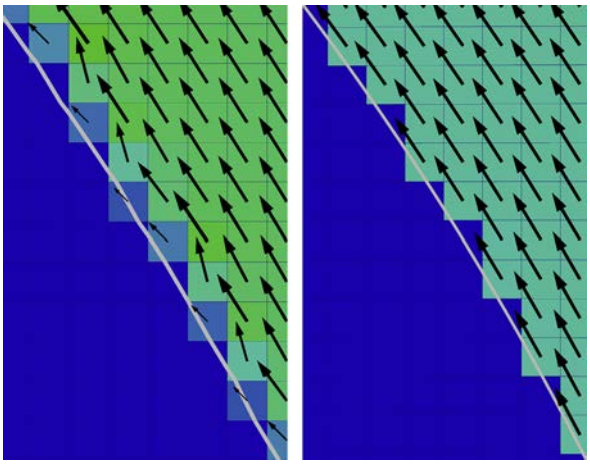
$$u_x = -Ay, \quad \mathbf{u}_y = Ax, \quad P = \rho_f A^2(x^2 + y^2)/2 \tag{26}$$

The fluid properties are $\rho_f = 2$ kg.m^{-3} and $\mu_f = 0.01$ Pa.s. The Reynolds number based on the gap r and the mean velocity is $Re = 300$. The simulations are run until the steady state is reached corresponding to the final time $t = 10$ s. The velocity and the pressure fields are reported in Fig. 15 and they apparently show an excellent agreement with the analytical solution. The role of the slip condition imposed on the interface can be examined more closely zooming on the velocity field near the interface (Fig. 16). The new penalization method is compared to the original one. The velocity field follows remarkably well the curved boundary.



(a) (b)

Fig. 15. Curved channel test case. (a) Pressure field at the end of the simulation. In white: isocontours of the pressure field. (b) Velocity field at the end of the simulation, coloration by magnitude.



(a) (b)

Fig. 16. Curved channel test case. Zoom of the velocity field near the interface; Left: Original method, Right: new penalization method.

In order to determine the convergence order of the method, simulations have been run with six different grids made of 4, 8, 16, 32, 64, and 128 cells in the gap of the channel. The results are reported in Fig. 17. The error on the x- and y-components of the velocity both decrease with an order of 0.75 while the pressure decreases show a first order convergence. The convergence close to first order is attributed to the imposed slip condition based on a first order extrapolation of the fluid velocity at the surface of the penalized domain. The convergence may be improved since other penalization methods offer better convergence rates such as 2 for the velocity and 1 for the pressure [10,37]. Additional numerical developments required to improve the convergence rate may be considered for future developments in NEPTUNE_CFD.

3.2. Flow inside a 2D inclined channel with moving walls

For this second test case, the configuration studied is a 2D inclined channel as shown in Fig. 18a. The sides of the channel (Σ_1 and Σ_2) are two parallel lines defined by the vector $(0.1, 0.8)$. The dimensions are $Lx = 0.1$ m, $Ly = 2 \times Lx$. The gap of the channel is $r = 0.25$ m. Outside of the fluid domain, the penalized domain Ω is set to a constant velocity $\mathbf{u_b} = -\mathbf{u_0} = -(u_{0x}, u_{0y}) = (-0.01, -0.08)$ m.s^{-1} parallel to the wall as shown in Fig. 18b. The slip condition is imposed on both surfaces. The fluid properties are $\rho_f = 1000$ kg.m^{-3} and $\mu_f = 0.1$ Pa.s so that the Reynolds number is $Re = \rho_f r |\mathbf{u_0}| / \mu_f = 20$. The initial conditions are imposed to $\mathbf{u}(t = 0, (x, y)) = 0$ and $P(x, y) = 0$ inside the fluid region Ω_f. At $t = 0$ the fluid velocity is imposed to $\mathbf{u} = \mathbf{u_0}$ at the inlet of the domain while the pressure is $P = 0$ at the outlet. The fluid velocity is thus parallel to the wall and in the opposite direction of the velocity imposed in the penalized domain. Due to the slip condition, the moving walls are expected not to interact with the fluid so that solution $\mathbf{u_b}(x, y) = \mathbf{u_0}$ and $P(x, y) = 0$ is expected in the channel. The simulations are run for $t = 5$ s until the steady state is reached. The velocity fields are presented in Fig. 19 for the original penalization method and the new one. The grid is made of 20×40 cells in each direction. The new method clearly show a uniform velocity field parallel to the wall while some oscillations attributed to the calculation of the divergence in the mass conservation are observed with the original method. Simulations have been

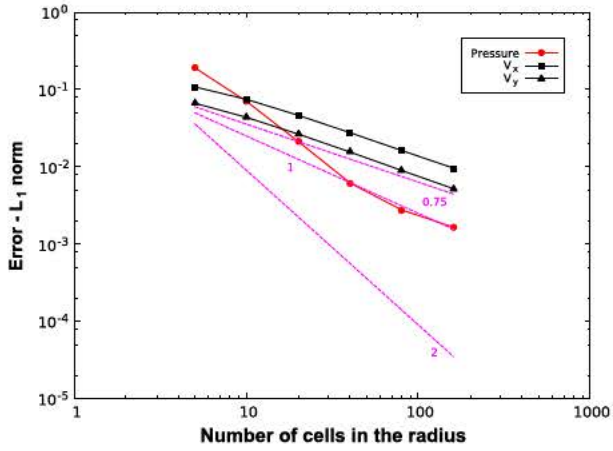


Fig. 17. Curved channel test case. Convergence with the grid size for P, u_x and u_y in L_1 norm.

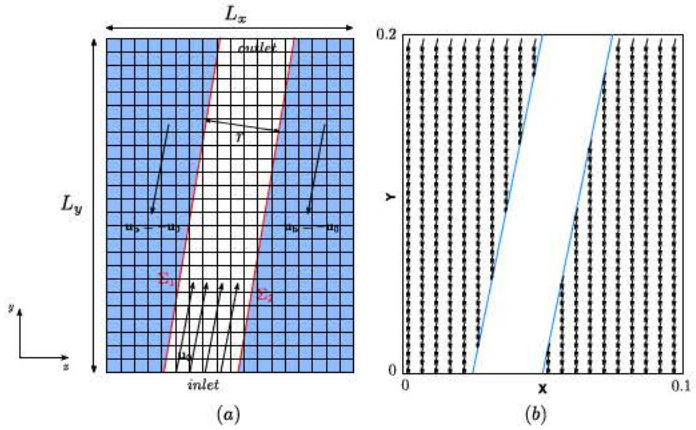


Fig. 18. (a) Sketch of the inclined channel test case, (b) initialization of the velocity field at $t = 0$.

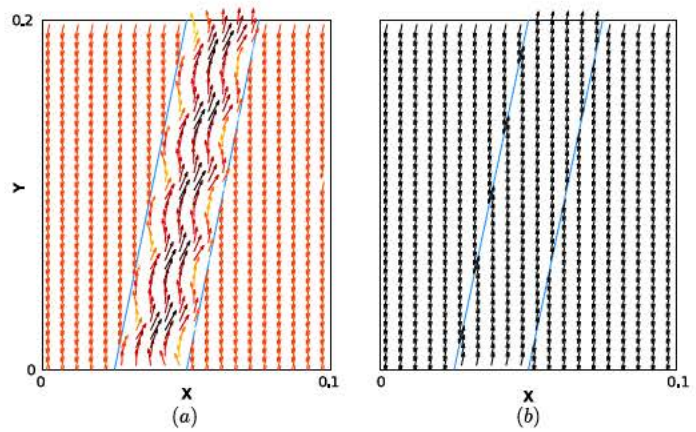


Fig. 19. Inclined channel test case. Velocity field at the end of the simulation, 20×40 grid. (a) Original method, (b) new penalization method.

run for seven different grids with 20, 40, 80, 160, 320, 640 and 1280 cells in the gap r of the channel. The exact solution at the end of the simulation is used to calculate the relative error. The results are shown in Fig. 20. For all the cases, the error is found to be at the level of the machine precision.

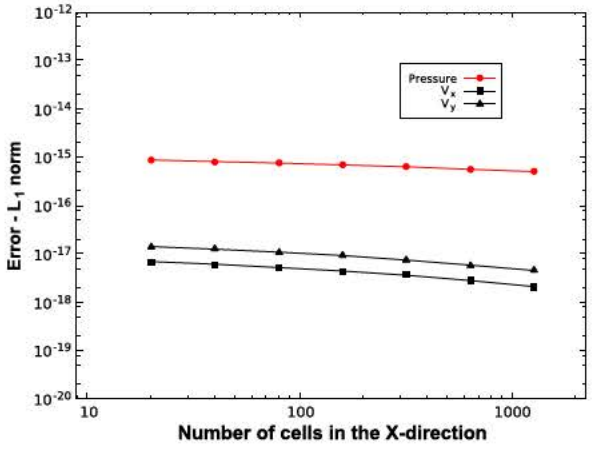
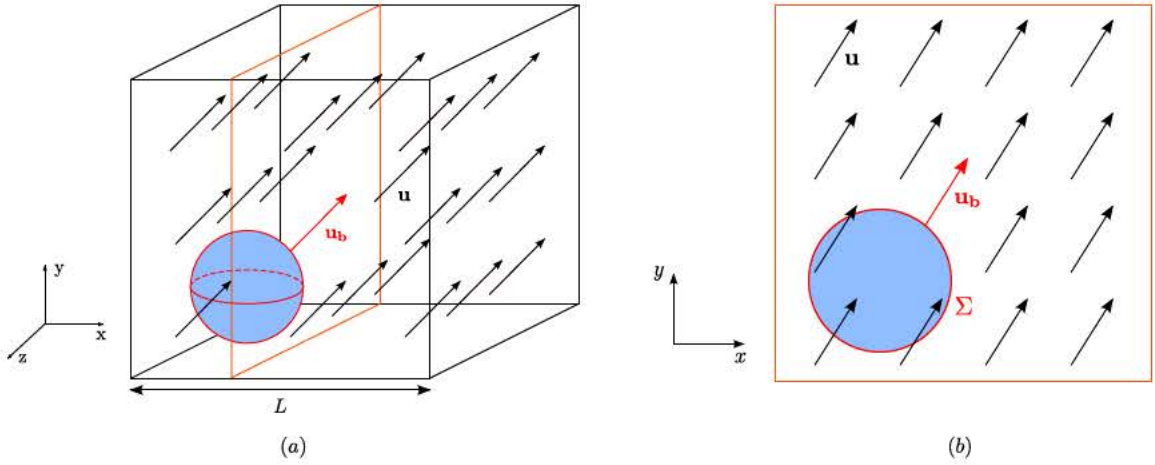


Fig. 20. Inclined channel test case. Convergence study for P, u_x, u_y in L_1 norm for the flow in a inclined channel.



(a) (b)

Fig. 21. 3D translating bubble test case. (a) Initial sketch of the configuration, (b) extracted plane.

3.3. 3D Bubble in an uniform translation with zero-relative velocity with the fluid

We now consider a 3D test-case. A 3D spherical bubble is moving at constant velocity $\mathbf{u_b} = (u_{bx}, u_{by}, u_{bz}) = (0.10, 0.15, 0.20)$ m.s^{-1} in the computational domain $L^3 = 0.01$ m $\times$ 0.01 m $\times$ 0.01 m as shown in Fig. 21a. The fluid properties are $\rho_f = 1000$ kg.m^{-3}, $\mu_f = 0.5$ Pa.s and the bubble radius is $r_b = 0.0016$ m. A zero relative velocity with the fluid is imposed so that the velocity should be everywhere the bubble velocity. The bubble moves from a position $\mathbf{x_0} = (x_0, y_0, z_0)$ to $\mathbf{x_1} = (x_1, y_1, z_1)$ with $|\mathbf{x_0} - \mathbf{x_1}| = 0.75 \times L$. The slip condition is imposed at the bubble surface. Periodic boundary conditions are imposed on each face of the cube. The initial pressure field is set to zero $P(x, t = 0) = 0$. The same initial fluid velocity $\mathbf{u_0}$ is imposed in both the fluid and the penalized domains. Due to the imposed slip condition at the bubble surface the velocity field should remain unchanged. Thus in each cut plane extracted from the cube and parallel to one of the planes XY, YZ, XZ (Fig. 21b), the corresponding 2-D velocity field should be uniform and not influenced by the penalized domain. The velocity fields obtained from the simulations with the new and original penalization methods on a 20^3 grid and extracted from cut planes (XY, YZ, XZ) containing the center of the bubble are shown in Fig. 22. As observed in the previous test cases, the new algorithm proposed here clearly show very good performances compared to the original one. The error with the exact solution calculated for both versions of the penalization method are reported in Table 2. The error is shown for three different times for the 200^3 grid. The error remains stable at the machine precision for the new method, while there is no error stabilization for the original one. The error is now analyzed with simulation carried out on the following grids: 25^3, 50^3, 100^3, 200^3 and 400^3. The relative errors reported in Fig. 23 for the velocity components and the pressure field remain at the machine precision for the grids considered.

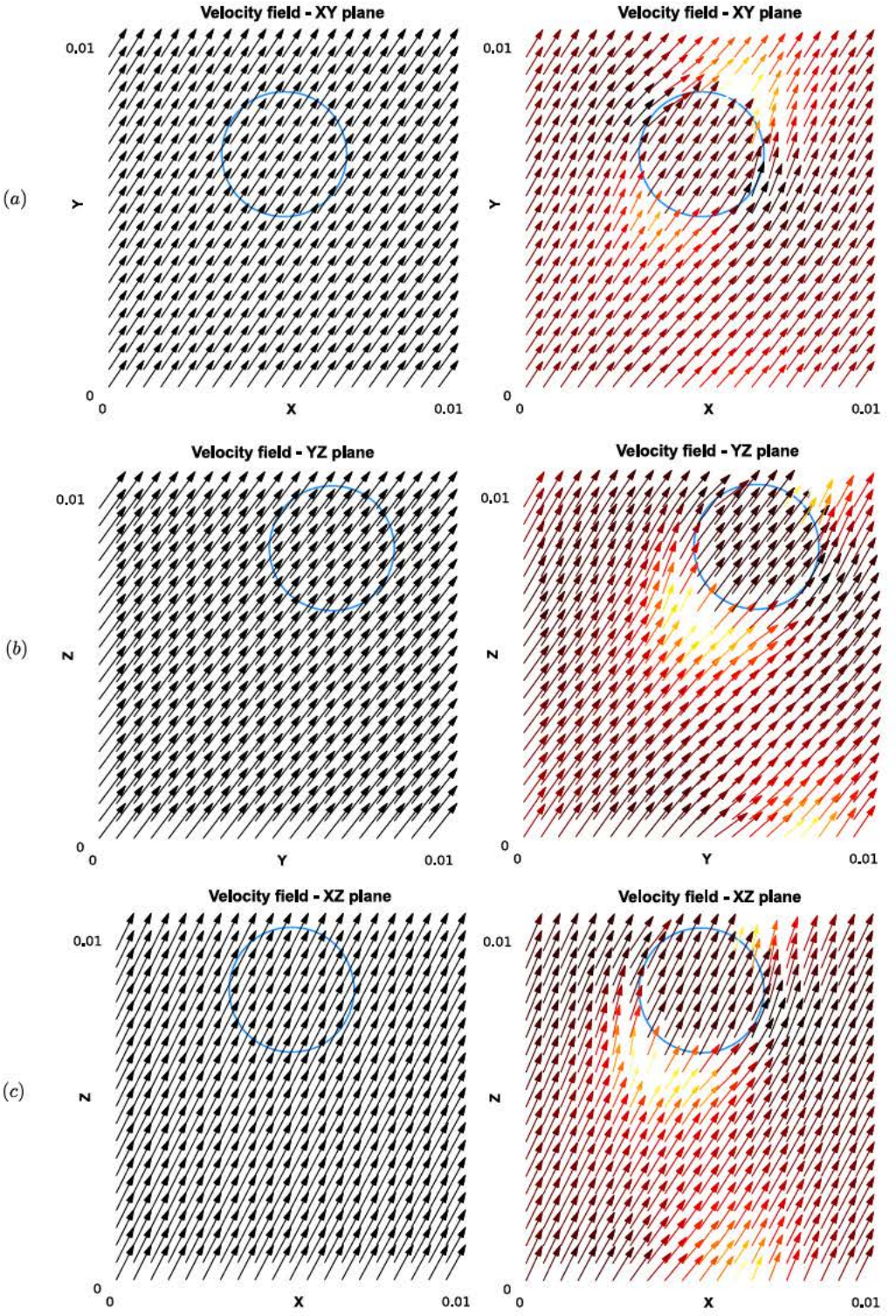


Fig. 22. 3D translating bubble test case. Velocity field at the end of the simulation. New penalization method (left) compared to the original method (right): (a) YZ plane, (b) XZ plane and (c) XY plane.

Table 2
3D translating bubble test case. Evolution of the L_2 error on the velocity through 3 time iterations. The 200^3 grid is used.

	Method	t_1	t_2	t_3
$\frac{\|\|\mathbf{u}-\mathbf{u_0}\|\|_2}{\|\|\mathbf{u_0}\|\|_2}$	CR	4.19×10^{-16}	4.14×10^{-16}	4.24×10^{-16}
	NCR	$1.76 \times 10^{+1}$	$2.61 \times 10^{+1}$	$1.31 \times 10^{+1}$

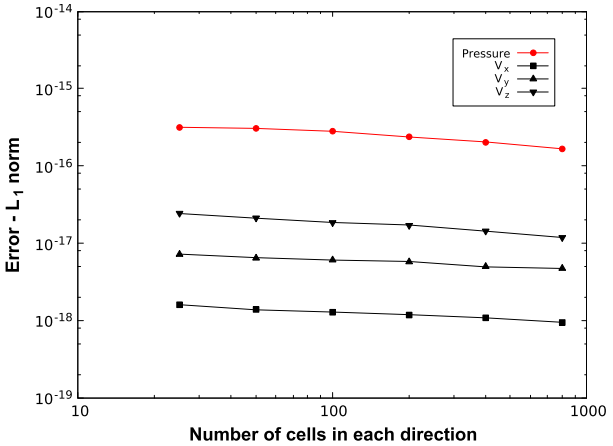


Fig. 23. Convergence study for $P, (V_x, V_y, V_z)$ in L_1 norm for the bubble moving at zero relative velocity in the flow test case.

4. Rise of a single bubble in a quiescent liquid

The aim of this section is to simulate the rise a single bubble in a liquid at rest in order to test the momentum coupling method proposed in section 2.8. The presented formulation makes possible a dynamic interaction between the fluid and the bubble. We consider two different moderate Reynolds numbers $Re = 17$ and $Re = 71$ based on the bubble diameter $d_b = 2r_b$ and u_{bT} the modulus of its terminal velocity $\mathbf{u_b}_T$. For such Reynolds numbers, the bubble is spherical and its trajectory can be obtained with [18]:

$$m_f C_M \frac{d\mathbf{u_b}}{dt} = -\frac{1}{2} C_D \rho_f \pi r_b^2 \mathbf{u_b}^2 - m_f \mathbf{g} \tag{27}$$

where $m_f = \rho_f \frac{4}{3}\pi r_b^3$, $C_M = 1/2$ is the added mass coefficient of a single spherical bubble and the history force can be considered of second order for such Reynolds numbers. The drag coefficient is given through the correlation [20]:

$$C_d = \frac{16}{Re} \quad 1 + (8/Re + 1/2(1 + 3.315/Re^{0.5})^{-1} \tag{28}$$

Equation (27) can be easily solved, leading to the terminal velocity $\mathbf{u_b}_T$ and the transient evolution of the bubble rising velocity. As it will be discussed, the coupling formulation introduces a dependence to the penalization viscosity. In particular, we will focus on the bubble initial acceleration $d\mathbf{u_b}/dt = -2\mathbf{g}$ (balance between bubble acceleration and gravity volume forcing), the terminal velocity $\mathbf{u_b}_T$ (balance between pressure-viscous drag with gravity volume forcing), and the bubble wake shape because of its importance for the development of bubble induced agitation [26].

We consider the configuration described in Fig. 24. A 1 mm spherical bubble is located at the bottom of a tank-like computational domain $(x_b, y_b, z_b) = (0.25d_b, 0.25d_b, 0.2d_b)$. At t_0, the bubble starts rising until reaching its terminal velocity. The simulation is stopped after the steady state is reached, for a final simulation time t_f depending on the considered Reynolds number. Simulations have been carried out for two different fluid viscosities leading to two different terminal velocities and the two Reynolds numbers $Re = 17$ and $Re = 71$. The list of the parameters used is given in Table 3. The penalization parameter τ is set to $\tau = 10^{-14}$ s. We want to consider here the influence of the internal penalization viscosity μ_p on the bubble motion. The time step Δt is imposed following the CFL criteria $|\mathbf{u}|\Delta t/\Delta x = 0.5$. Three grid refinements have been tested: 4, 8 and 16 cells in the radius of the bubble. Note that when considering 4 cells in the bubble radius, the boundary layer thickness estimated as $\delta \sim r_b/Re^{1/2}$ and the grid size have the same order of magnitude for $Re = 71$.

The dimensions of the computational domain shown in Fig. 24 are $5d_b \times 5d_b \times 25d_b$. 48 simulations in total have been carried out for a wide range of internal viscosities, eight for each Reynolds number and grid refinement. The computational resources have been adapted to the refinement of each grid: 6×24, 8×24, 12×24 CPU units were used respectively for 4, 8 and 16 cells in the radius of the bubble, leading to the corresponding CPU times 45 min, 2 hours and 12 hours for a single simulation.

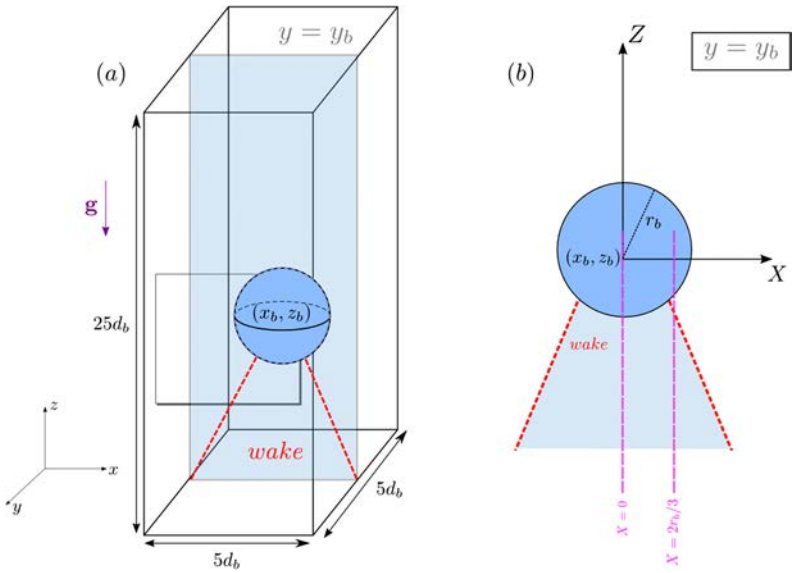


Fig. 24. Single bubble rise. (a) Initial configuration for the simulation. The bubble is located at the bottom of the tank. The reported vertical plane is used for the wake description. (b) Locations of the reported wake profiles.

Table 3
Set of numerical parameters for the simulation of a single bubble rise.

d_b (mm)	ρ_b (kg.m^{-3})	ρ_f (kg.m^{-3})	μ_f (Pa.s)	t_f (s)	$\lvert\mathbf{u}_{\mathbf{b}T}\rvert$ (cm.s^{-1})	Re
1.0	50	1000	0.005	0.06	8.90	17
1.0	50	1000	0.0022	0.12	15.60	71

4.1. Initial acceleration and terminal velocity

The time evolution of the bubble is reported in Fig. 25 for $Re = 17$ and $Re = 71$ and different penalization viscosities μ_p. The evolutions clearly reveal the effect of the penalization viscosity μ_p on the evolution of the bubble velocity. An appropriate penalization viscosity μ_p needs to be selected to reproduce correctly both the initial acceleration (added mass effect) and the terminal velocity of the bubble. For each Reynolds number, and for each ratio of cells per radius, we are always able to reproduce a satisfactory evolution of the bubble rise.

A summary of the tends observed in Fig. 25 is shown in Fig. 26. The optimum value of the penalization viscosity μ_p for a good description of the initial acceleration is compared to the one given the best terminal velocity. They are reported as a function of the grid spacing. Depending on both the grid resolution and the Reynolds number an appropriate penalization viscosity μ_p has to be selected. It is not here possible to propose a clear scaling of μ_p and this aspect of the numerical method needs to be improved for providing predictive simulations of single bubble dynamics. However, the objective is here to provide a method able to describe bubbly flows. As it will be shown in the next section, the impact of the penalization viscosity μ_p is significantly reduced when considering a bubble swarm.

4.2. Maximal vorticity

In addition to the velocity fields, we also computed the maximal vorticity Ω_{max} that develops at the bubble surface for each mesh refinement and each Reynolds number. This quantity is of interest since it controls the bubble drag force and the development of its wake. The maximal vorticity for a spherical bubble moving steadily in a viscous fluid is given by the following expression [17]:

$$\omega_{max} = \frac{|\mathbf{u}_{\mathbf{b}T}|}{r_b} \frac{16 + 3.315Re^{1/2} + 3Re}{16 + 3.315Re^{1/2} + Re} \tag{29}$$

This expression has been established through DNS simulations for Reynolds numbers ranging from 0.1 to 5000. The evolutions are shown in Fig. 27. The maximal vorticity measured in the simulations tends to get closer to the DNS results (Equation (29)) with the mesh refinement. The maximum velocity and vorticity at a bubble surface are directly linked by the relation $u_{max} = r_b\omega_{max}/2$ [17]. The good convergence of the maximal vorticity to the expected value implies that our simulations reproduce a correct fluid velocity close to the bubble surface. The streamlines and the velocity magnitude for

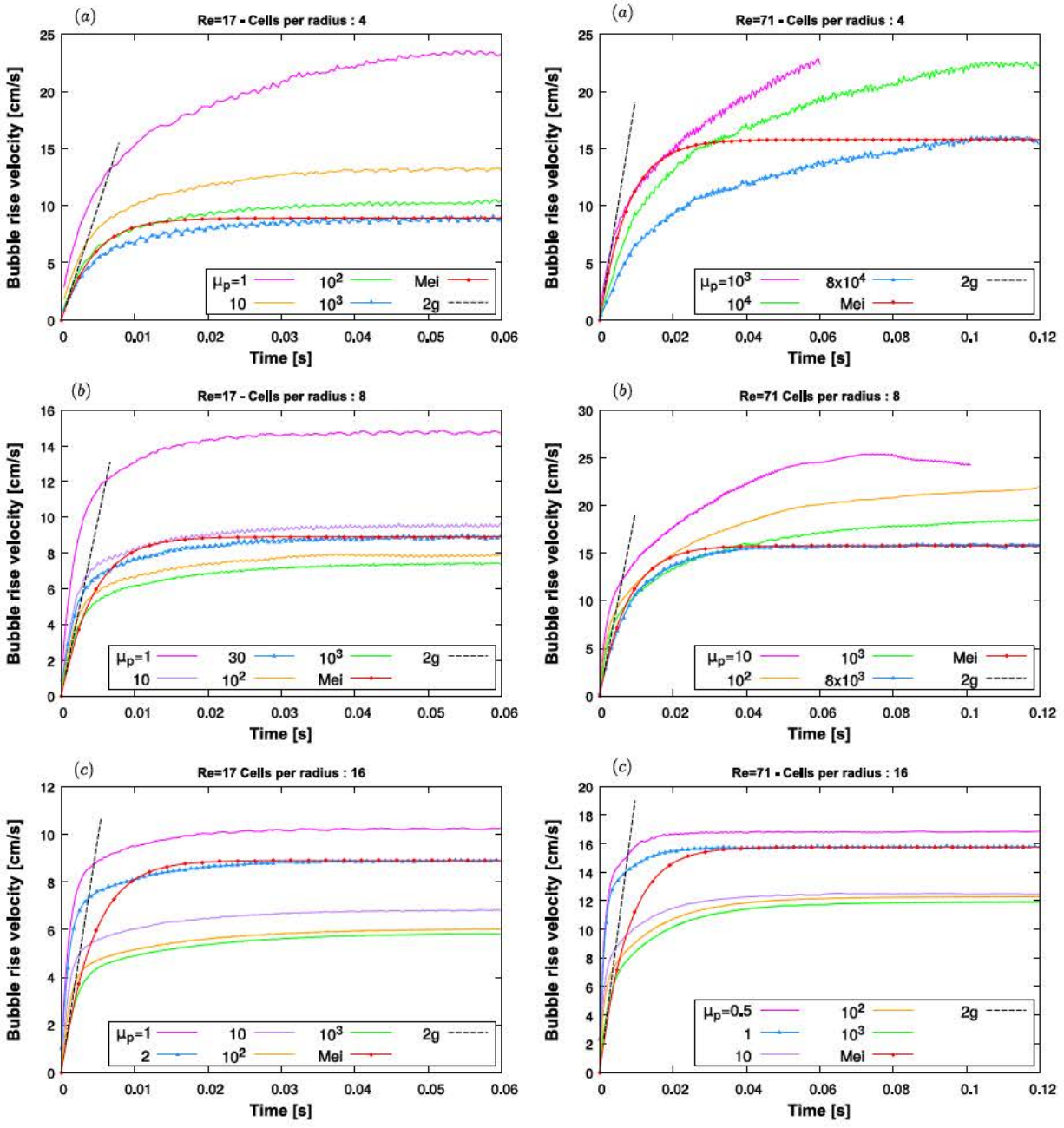


Fig. 25. Single bubble rise. Effect of the penalization viscosity μ_p on the transient motion of a rising bubble at $Re = 17$ (left column) and $Re = 71$ (right column). Grid refinement: (a) 4 cells per radius, (b) 8 cells per radius and (c) 16 cells per radius. On each plot, the blue line with triangles indicates the best solution in term of terminal velocity. The black dashed line represents the theoretical initial acceleration leading to $\mathbf{u_b} = -2gt$.

the two cases are reported in Fig. 28 for the finer mesh (16 cells in the bubble radius). The plots clearly show the location of the velocity in the vicinity of the bubble and the increase of its magnitude when increasing the Reynolds number.

4.3. Bubble wake

We end the inspection of the bubble rising motion by considering the bubble wake. In bubble swarm configurations, the liquid agitation mainly results from nonlinear interactions between bubble wakes [25,26]. For a further use of the method in this kind of configuration, obtaining a good description of the wake behind the bubble ensures that the method may be ready for the simulation of bubbly flows. A detailed description of bubble wakes can be found in [13]. We denote X the radial distance from the wake axis, Z the distance downstream (see Fig. 24). At large distances downstream, the flow

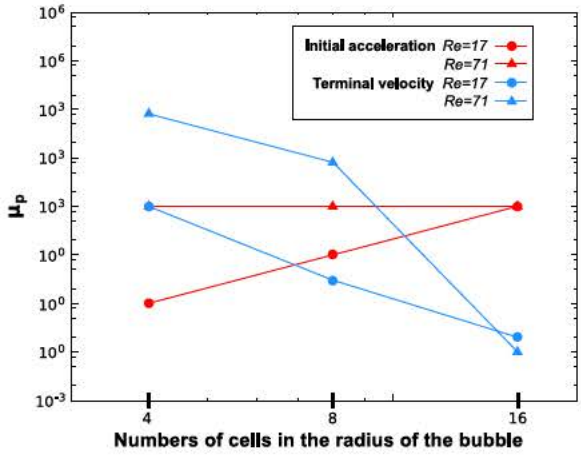


Fig. 26. Single bubble rise. Best penalization viscosity μ_p for the description of the initial acceleration and the terminal velocity as a function of the grid spacing.

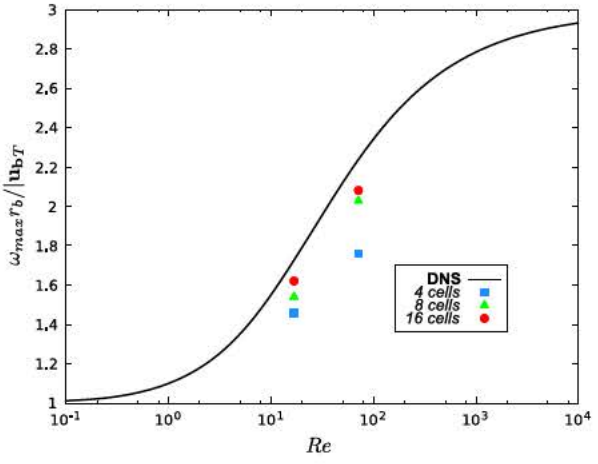


Fig. 27. Single bubble rise. Normalized vorticity $\omega_{max} r_b / |\mathbf{u}_{bT}|$ for $Re = 17$ and $Re = 71$.

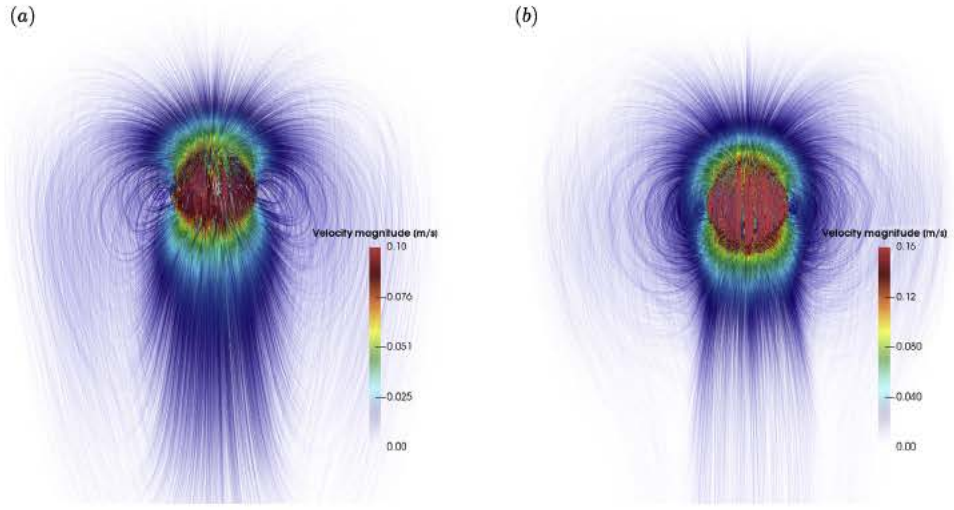


Fig. 28. Single bubble rise. Streamlines and velocity magnitude: (a) $Re = 17$, (b) $Re = 71$.

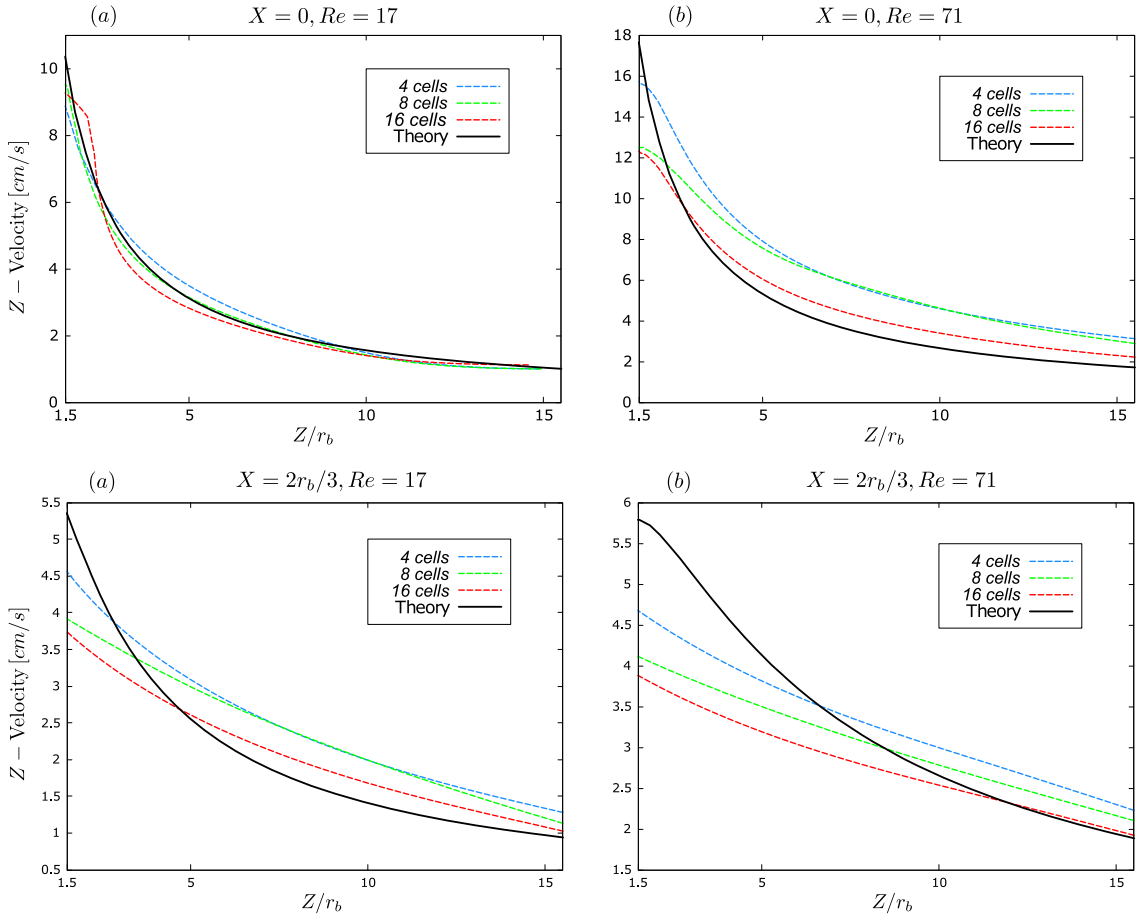


Fig. 29. Single bubble rise. Velocity profile in the wake of a rising bubble as indicated in Fig. 24. (a) $Re = 17$, (b) $Re = 71$. First line $X = 0$, second line $X = 2R/3$.

is known to follow the standard far-wake behavior that will be used here for the comparison. The corresponding velocity deficit profile parallel to the X-direction is given by (Batchelor, 1967 [3]):

$$u_{wake} \sim \frac{u_{bT}\, Q}{4\pi\, \nu_f\, Z} exp\left[-u_{bT}\frac{X^2}{4\nu_f Z}\right] \tag{30}$$

where $\nu_f = \mu_f/\rho_f$ is the cinematic viscosity of the fluid and $Q = F_D/\rho_f u_{bT}$ is determined by integration over the wake and is directly related to the magnitude F_D of the drag force of the body. Given $\mathbf{u_b}_T$ and F_D, we obtain the far wake description. The numerical results are presented for the two cases in Figs. 29 for $X = 0$ and $X = 2r_b/3$ and for the 3 mesh refinements considered: 4, 8 and 16 cells in the bubble radius. The velocity decay in the bubble wake tends to get closer to relation (30) with the mesh refinement for both Reynolds numbers $Re = 17$ and $Re = 71$.

5. Mono-dispersed bubble swarm

The final step of the validation process is the simulation of bubbles swarm for both a high Reynolds numbers and a high number of bubbles. The objective is to verify the whole numerical method, in order to demonstrate the effectiveness of the so-built numerical tool for bubbly flows simulations. The validation strategy is defined as follows. Experimental investigations of the flow generated by a homogeneous population of bubbles rising in water have been carried out by Zenit et al. (2001) [42], Garnier et al. (2002) [11], Riboux et al. (2010) [26] and Colombet et al. (2015) [8]. The experiments provide a rather complete description of both the bubbles motion and the induced agitation in the liquid. The idea here is to reproduce in relative similar conditions those experiments with our numerical tool, and make a direct comparison with the relevant information characterizing a bubble swarm such as the velocity PDFs of both the bubbles and the fluid. The flow is supposed to be fully periodic in each direction. The collisions between the bubbles are assumed fully elastic. The simulation parameters are shown in Table 4. For all the simulations reported in this section, the time step is the time step imposed by the CFL condition divided by 5. The statistical measurement (PDF) is started after reaching a stabilized agitated

Table 4
Set of numerical parameters for the bubble swarm simulations.

d_b (mm)	ρ_b (kg.m^{-3})	ρ_f (kg.m^{-3})	μ_f (Pa.s)	μ_p/μ_f	V_0 (cm.s^{-1})	Re_0	α_0
1.0	50	1000	0.0008	100	22.5	281	0.008

state of the mixture. The simulations are run until the PDFs are converged. We note here α the global void fraction. The analysis of the results requires for normalization purpose the computation of the rise velocity V_0 of a single bubble. We note Re_0 and α_0 the corresponding Reynolds number and reference void fraction, respectively. The values of V_0, Re_0 and α_0 are reported in Table 4.

Before a detailed investigation of the bubble swarm, we present some preliminary simulations to test the mesh refinement, the domain size and the penalization viscosity.

5.1. Influence of the mesh refinement

We first consider the influence of the mesh refinement with a special attention to the liquid and bubble velocity fluctuations. We introduce three grids refinements R_0, R_1 and R_2 corresponding respectively to 3, 6 and 12 cells in the bubble radius. The computational domain is chosen as $\Omega = D_1 = [0; 005] \times [0; 005] \times [0; 025]$ (Fig. 31). Three void fraction are considered here: $\alpha = 5\%, 10\%$ and 15%, corresponding to $N_b = 60, 120$ and 180 bubbles, respectively. The PDFs of the fluid velocity and the bubble velocity for both the horizontal and vertical components are reported in Fig. 30. Statistics on the flow have been measured between $t = 0.2$ s, once the fluid is homogeneously agitated, and $t_f = 2$ s. As shown, the grid has no significant influence on the bubble horizontal velocity for the considered refinements. The other velocity components are more sensitive to the grid but a clear convergence with the grid is observed and close results are found for the refinements R_1 and R_2. The average bubble rising velocity increases as Δx decreases. This is due to an underestimation of the bubbles rising velocity on coarse grids, mostly due to the relative high Reynolds number considered here ($Re \simeq 300$) because of the boundary layer at the bubble surface that decreases as $Re^{-1/2}$. The average rising velocity is evolving in the same way for each void fraction. The variance of the horizontal and vertical fluid velocities increases slightly with the mesh refinement. The simulations performed with the mesh refinement R_0, R_1 and R_2 required respectively 12, 81 and 350 CPU hours of calculation for 6×28 CPU units. In order to save computational resources, the refinement R_1 is used for all the simulations reported in the following sections.

5.2. Influence of the domain size

We want now to consider the influence of the domain size on the results and on the convergence of the statistics. For each simulation, the number of bubbles N_b is deduced from the volume of the computational domain and the imposed void fraction α. If the number of bubbles in the domain reaches a relative low value, the statistical measurements will not be able to describe precisely the bubble-induced agitation of the fluid. The computational domains $D_1 = [0; 5d_b] \times [0; 5d_b] \times [0; 25d_b] = [0; L_x] \times [0; L_y] \times [0; L_z]$, $D_2 = [0; L_x] \times [0; L_y] \times [0; 2L_z]$ and $D_3 = [0; 2L_x] \times [0; 2L_y] \times [0; L_z]$ are considered (see Fig. 31). Compared to $D1$, for $D2$ the length of the domain in the (Oz) direction doubles, while for $D3$ the domain is double in both horizontal directions (Ox) and (Oy). Three void fractions have been studied: $\alpha = 5\%$, 10% and 15%, corresponding respectively to $N_b = 60$, 120 and 180 bubbles for $\Omega = D_1$, $N_b = 120$, 240 and 360 bubbles for $\Omega = D_2$ and $N_b = 240$, 480 and 720 bubbles for $\Omega = D_3$. Statistics on the flow have also been measured between $t = 0.2$ s, once the fluid is homogeneously agitated, and $t_f = 2$ s. The simulations required approximately 4, 6 and 8 *days* of calculation for the domains D_1, D_2, D_3 using 6×28 CPU units. The normalized PDFs plotted in Fig. 32 show very similar results for each component of the liquid and bubble velocity when using the domains D_1, D_2 and D_3 for the void fractions considered ($\alpha = 5\%$, 10% and 15%).

We conclude that performing simulations on a computational domain larger than D_1 in any direction will not provide any additional information on the PDFs. All the simulations presented in the following sections are carried out using the computational domain D_1.

5.3. Influence of the penalization viscosity μ_p

In the previous section, we detailed the influence of the penalization viscosity μ_p on the dynamics of a single bubble rising in a quiescent liquid. For each bubble Reynolds number and mesh refinement considered, an optimum value of μ_p has to be determined for a good description of the bubble velocity evolution. As we deal now with bubble swarms, we want to examine again the influence of μ_p on the results. For this purpose, three simulations have been carried out for $\alpha = 10\%$, corresponding to three different values, $\mu_p = 0.1$, 1 and 10 Pa.s. The mesh refinement is R2 (6 cells in the bubble radius) and the domain is $D1$. The simulations are stopped at $t_f = 4$ s. We extracted the PDFs of the horizontal and vertical components of the fluid and bubble velocities for $\mu_p = 0.1$, 1 and 10 Pa.s (Fig. 33). For each component of the liquid and bubble velocities, the agreement is perfect between the PDFs for different values of μ_p. As a conclusion, the choice of μ_p has no influence on the simulations of bubble swarms. A clear different behavior was observed for the simulation

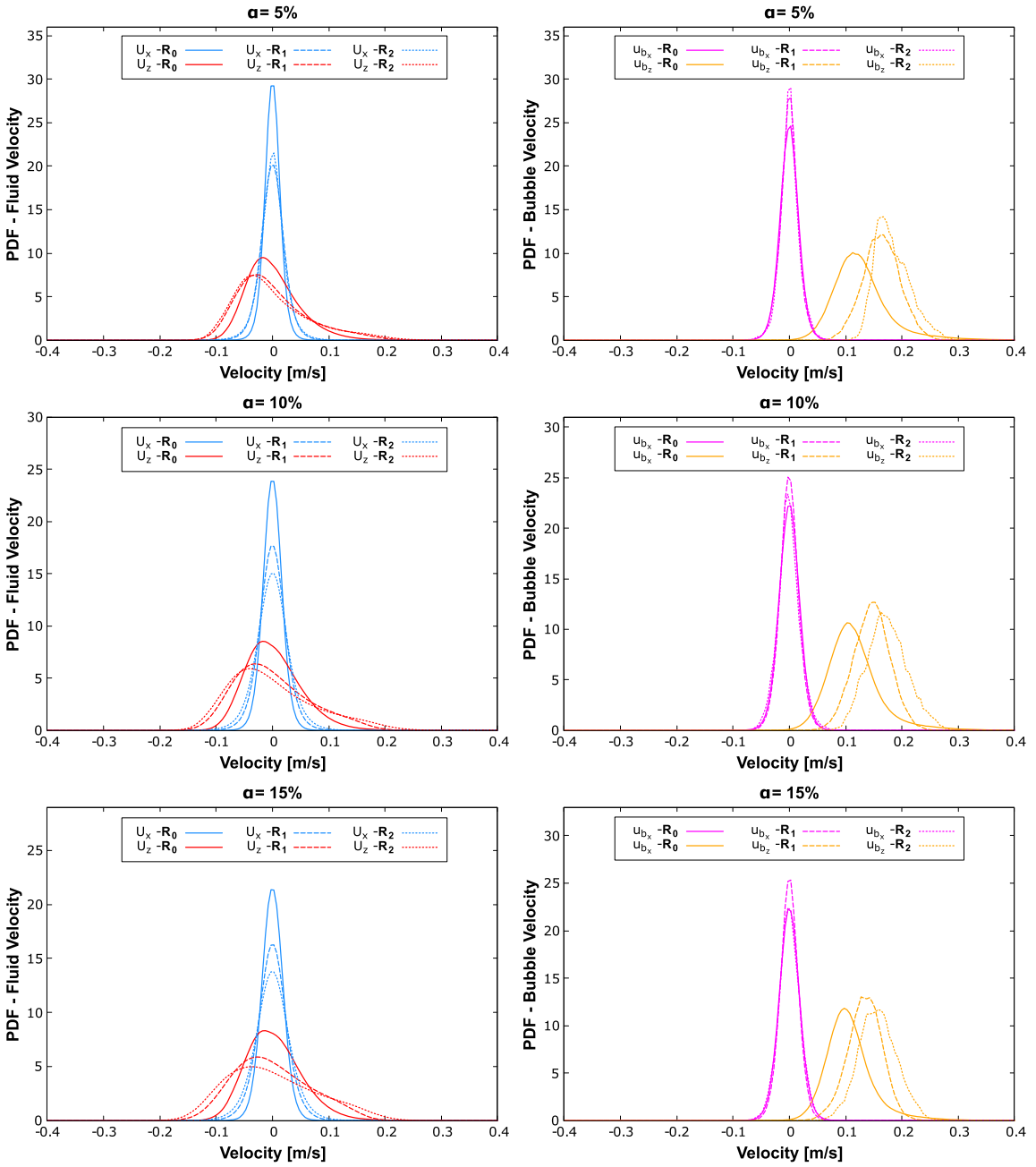


Fig. 30. Bubble swarm. Effect of the mesh refinement on the normalized PDFs of the liquid velocity (left) and the bubbles velocity (right).

of a single bubble where results are sensitive to the choice of the penalization viscosity μ_p. We were not able to clearly determine the origin of the observed difference between the single bubble case and the bubble swarm case. We believe that the main candidate is the solver used in NEPTUNE_CFD for the inversion of the semi-implicit system (24)–(25). In single bubble configurations, we think that the presence of the penalization viscosity causes a singularity in the matrix to inverse while in the bubble swarm configuration, the bubble singularities are distributed in all the domain improving the matrix conditioning, explaining the independence of the results with the viscosity of penalization. A deeper analysis of the solver performances need to be conducted to clearly explain this numerical behavior.

We now discuss the influence of the numerical value of μ_p on the CPU time. We have observed that the CPU time strongly increases when μ_p increases. For each simulation performed ($\mu_p = 0.1$, 1 and 10 Pa.s) we report the final CPU time at $t_f = 2$ s in Table 5. Again, we note the increase of the CPU time as μ_p increases. However the CPU time increase is only around 7% for a change of one order of magnitude for μ_p (0.1 to 1 Pa.s and 1 to 10 Pa.s), which remains acceptable in term of management of computing resources.

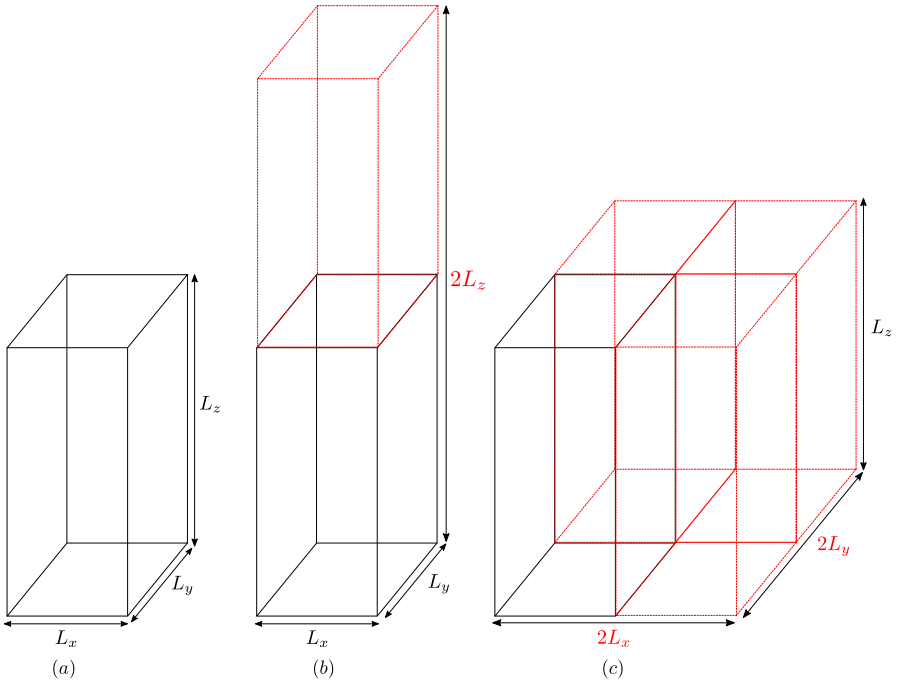


Fig. 31. Bubble swarm. Computational domains used for the study of the domain size effect (a) D_1, (b) D_2, (c) D_3.

Table 5
Bubble swarm. Evolution of the final CPU time for different values of μ_p.

	$\mu_p = 0.1$ Pa.s	$\mu_p = 1$ Pa.s	$\mu_p = 10$ Pa.s
CPU time (hours)	72	77	83

5.4. Bubble and fluid agitation

Simulations have been carried out for different void fractions ranging from $\alpha = 2.5\%$ to 20%, leading to an increasing number of bubbles in the flow as shown in Fig. 34.

The average bubble rising velocity $< u_{bz} >$ is reported in Fig. 35 as a function of the void fraction for $0.002 \leq \alpha \leq 0.20$. The evolution is compared to the experiments. The diameter $d_b = 1$ mm is considered in the present work while the diameters considered in the experiments are $d_b = 1.6$, 2.1 and 2.5 mm in Riboux et al. [26], $d_b = 1.4$ mm in Zenit et al. [42], and $d_b = 3.5$ mm in Garnier et al. [11]. The velocities are normalized by the velocity V_0 of a single rising bubble. The results are found in very good agreement with both the experiments and the decreasing law $V_0(1-\alpha^{0.49})$.

Variances of the bubble rising velocity $< u'^2_{bz} >$ are compared in Fig. 35 with the corresponding data available in the literature. The experimental results highlight the independence of the variance to the void fraction, tendency which is also remarkably observed for the present simulations. This behavior show that the contribution of the path oscillations dominates over bubble interactions, effect that is properly reproduced in our simulations. We also note that the variance in the present work ($< u'^2_{bz} > \approx 2 \times 10^{-3}$ m^2.s^{-2}) significantly differ with the values obtained in the experiments ($< u'^2_{bz} > \approx 1.2 \times 10^{-2}$ m^2.s^{-2}) for the three diameters considered. Clearly we are not able to reproduce the same level of bubble agitation as reported by these experiments. The experiments were performed for higher bubble diameters, corresponding to higher bubble Reynolds numbers and significantly deformed bubbles. For such conditions, single bubble move following zig-zag or helicoidal path. This may be a possible explanation of the lower level of bubble agitation observed with our spherical bubbles. However, more recently Colombet et al. (2015) [8] also obtained in their experiments a lower level for the bubble agitation when the bubble motion is measured using particle tracking velocimetry based on images taken with a fixed focal lens. They report $< u'^2_{bz} > \approx 3 \times 10^{-2}$ m^2.s^{-2} up to $\alpha = 10\%$, a value comparable to our simulations. As discussed in [26] the measure by dual optical probe of the variance of the bubble agitation may be significantly perturbed when bubbles are oblate spheroid moving with oscillating velocity and orientation.

We now focus on the dynamics of the liquid. Fig. 36 shows the normalized PDF of two components of the fluid velocity for the present work ($d_b = 1$ mm), the considered void fractions ranging from $\alpha = 2.5\%$ to $\alpha = 15\%$. The PDFs are scaled following $(\alpha/\alpha_0)/V_0$ initially proposed by Risso [27]. We have also checked (not shown here) that the best superposition of our PDF is also obtained with $\alpha^{0.4}$. The PDFs of the horizontal velocity (right column) are symmetric, meaning that the simulations are able to restore the anisotropy property, the flow being statistically axisymmetric around the bubble, and

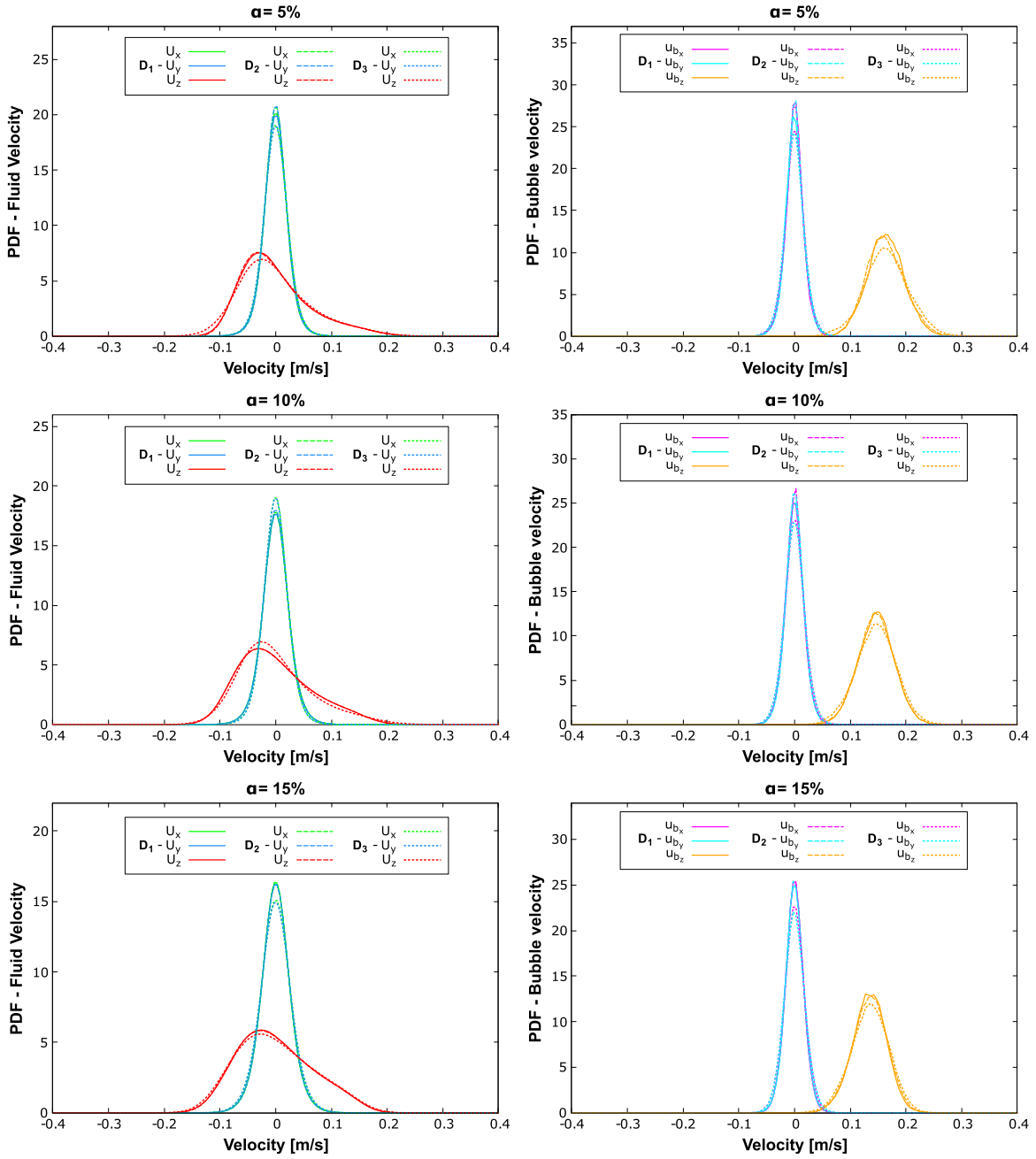


Fig. 32. Bubble swarm. Effect of the domain size. Normalized PDFs of vertical and horizontal liquid (left) and bubble (right) velocity fluctuations.

the distribution of bubbles in the horizontal direction being uniform. A lower liquid agitation is observed in the horizontal direction. However the comparison for the vertical fluid agitation is very good for both the shape and the level of agitation. The shape of the vertical bubble velocity PDFs is clearly non-symmetric due to the entrainment of the fluid in the wake of the bubbles, implying that upward fluctuations are more probable.

6. Conclusions

A new penalization method for the simulation of bubbly flows has been proposed. The originality of the presented work relies on the extension of the classic penalization methods developed for solid objects for the simulation of bubbles. The bubbles are seen as moving objects whose motion is determined by the action of the fluid on them, through the new coupling method presented. A second notable point is the numerical methodology implemented to ensure a shear-free condition at the surface of the penalized domain.

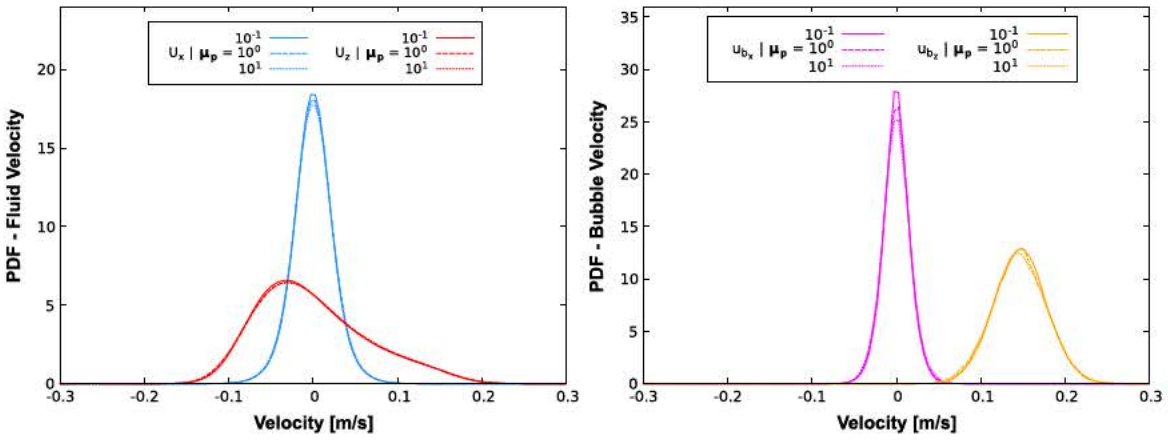


Fig. 33. Bubble swarm. Normalized PDFs of vertical and horizontal components of (left) the liquid velocity and (right) the bubble velocity for $\mu_p = 0.1$, 1 and 10 Pa.s.

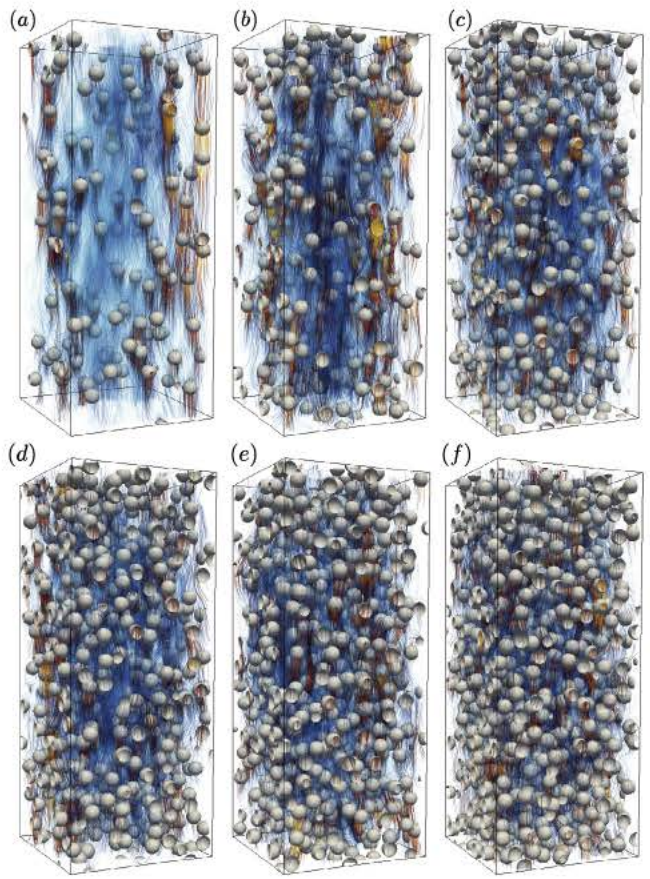


Fig. 34. Bubble swarm. Streamlines of the velocity field for six simulations (a), (b), (c), (d), (e), (f) corresponding to the void fractions $\alpha =$ 2.5%, 5%, 7.5%, 10%, 12.5% and 15%, respectively.

The presented results bring different enlightenments to the efficiency of the method. The first part of the results allows us to establish the convergence of the method and its stability through different test cases on non-coupled objects with different shapes (plane, curve, sphere). The tests performed for a rising bubble show the need of a calibration of the method by adjusting a penalization viscosity, introduced in the coupling formulation to have a uniform velocity inside the penalized domain. Once the penalization viscosity is selected, a good agreement is shown for the terminal velocity, the added mass effect and the wake shape. Finally, the simulations carried out for bubble swarms with a large range of void fraction reveal a very good agreement with experiments making a clear confirmation of major experimental results: the decay of the mean

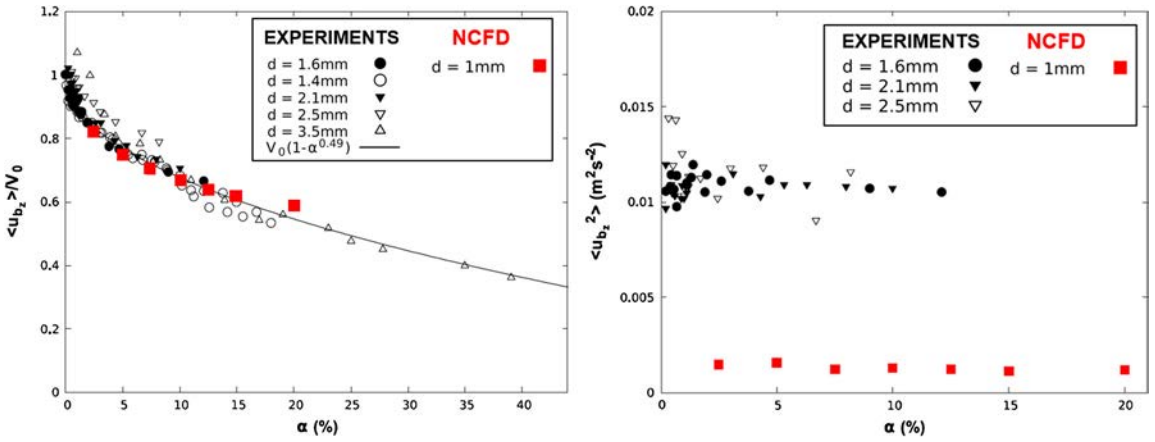


Fig. 35. Bubble swarm. (Left) Average bubble velocity $< u_{bz} >$ and (right) variance $< u'^2_{bz} >$ of the bubble velocity normalized by the velocity V_0 of a single rising bubble as a function of the gas volume fraction α. Red squares, present numerical work. Experiments: • ▼ ▽ Riboux et al. (2010), ○ Zenit et al. (2001), △ Garnier et al. (2002).

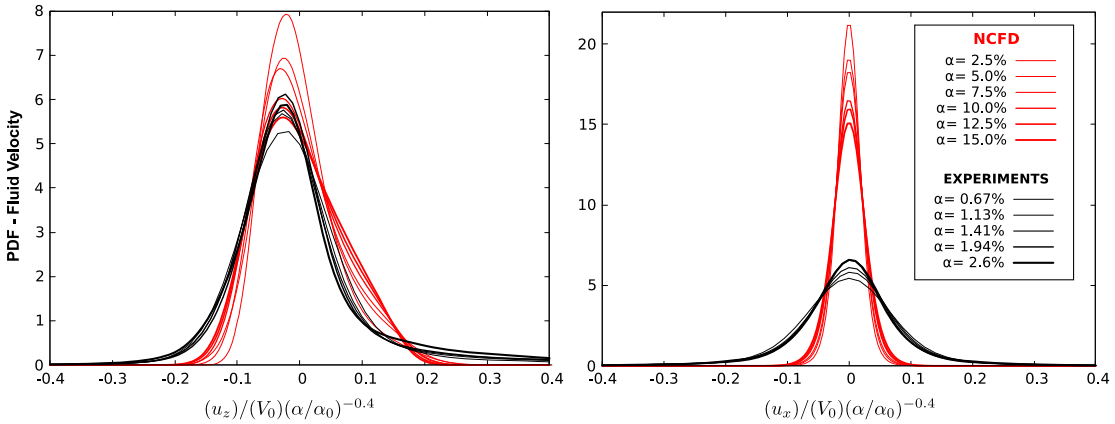


Fig. 36. Bubble swarm. Normalized PDFs of vertical (left) and horizontal (right) liquid velocity fluctuations.

bubble velocity with the void fraction, a bubble agitation independent of the void fraction, and a liquid agitation in both horizontal and vertical directions scaling as $\alpha^{0.4}$.

The results obtained through this paper demonstrate the viability of the numerical tool for the simulations of bubbly flows. Despite the assumptions made on the bubbles (spherical shape, non-deformability, elastic collisions) we were able to reproduce the main characteristic of bubble induced agitation and confirm the origin of the induced agitation resulting from wake interactions. One of the main interest of the method is its capacity to deal easily with a high number of bubbles at moderate and high void fraction. The simulations presented in this paper involve up to 720 bubbles for a bubble Reynolds number of $Re \simeq 300$ and for a void fraction up to 20%. In this paper we show that our numerical approach is able to reproduce the main characteristics of bubbly flows.

This numerical method was originally built to do deal with two kinds of objectives. The short-term objective was to develop a numerical tool allowing us to perform liquid–bubbles simulations in a simplified frame (no break-up, no coalescence, no mass transfer, elastic collisions). We proposed a successful validation strategy to qualify the abilities of our tool and it can now be used to investigate mono-dispersed bubbly flows and to work on closure relations for Euler/Euler simulations. The second objective is to consider bubbly flows relatively close to nuclear applications. First, the method will be extended to consider poly-dispersed bubbly flows. Then deformation and coalescence effects, as well as mass transfer will be introduced in the numerical method with relative ease, meanwhile considering deformation up to breakup events will be more challenging.

Acknowledgements

The authors thank EDF R&D for the computational resources provided for the realization of this work. The computations were performed using ATHOS and PORTHOS supercomputers.

Appendix

We detail in this appendix the derivation of the coupled relations (24) and (25). We start from the two-phase formalism described in equation (23). Due to the uniformity of the penalized velocity $\mathbf{u_b}$ in the bubble, $\nabla \mathbf{u_b} = 0$ and $\nabla^2 \mathbf{u_b} = 0$. Each equation is first divided by α_f and α_b for the liquid and the bubble phase, respectively. The pressure gradient ∇P is then expressed from each equation as:

$$\nabla P = -\rho_f \frac{\partial \mathbf{u}}{\partial t} - \rho_f \mathbf{u}.\nabla \mathbf{u} + \mu_f \nabla^2 \mathbf{u} + \rho_f \mathbf{g} + \alpha_b \rho_b \frac{\mathbf{u_b} - \mathbf{u}}{\tau} \tag{31}$$

$$\nabla P = -\rho_b \frac{\partial \mathbf{u_b}}{\partial t} + \rho_b \mathbf{g} - \alpha_f \rho_b \frac{\mathbf{u_b} - \mathbf{u}}{\tau} \tag{32}$$

These equations are linked through the same pressure gradient. Combining equations (31) and (32) leads to the following relation:

$$(\alpha_b + \alpha_f)\rho_b \frac{\mathbf{u_b} - \mathbf{u}}{\tau} = -\rho_b \frac{\partial \mathbf{u_b}}{\partial t} + \rho_f \frac{\partial \mathbf{u}}{\partial t} + \rho_f \mathbf{u}.\nabla \mathbf{u} - \mu_f \nabla^2 \mathbf{u} + (\rho_b - \rho_f)\mathbf{g} \tag{33}$$

with $\alpha_b + \alpha_f = 1$. An implicit scheme is used for the penalization term and the resulting semi-discrete formulation writes as:

$$\rho_b \frac{{\mathbf{u_b}}^{n+1} - \mathbf{u}^{n+1}}{\tau} = -\rho_b \frac{{\mathbf{u_b}}^{n+1} - {\mathbf{u_b}}^{n}}{\Delta t} + \rho_f \left[\frac{\mathbf{u}^{n+1} - \mathbf{u}^n}{\Delta t} + \mathbf{u}^{n+1}.\nabla \mathbf{u}^n - \frac{\mu_f}{\rho_f} \nabla^2 \mathbf{u}^n \right] + (\rho_b - \rho_f)\mathbf{g} \tag{34}$$

The u_b^{n+1} terms are regrouped on the left side of the equation. Equation (34) becomes:

$$\begin{aligned} \rho_b \frac{{\mathbf{u_b}}^{n+1} - \mathbf{u}^{n+1}}{\tau} = \rho_b \frac{\Delta t}{\tau + \Delta t} \frac{{\mathbf{u_b}}^{n} - \mathbf{u}^{n+1}}{\Delta t} + \rho_f \frac{\Delta t}{\tau + \Delta t} \left[\frac{\mathbf{u}^{n+1} - \mathbf{u}^n}{\Delta t} + \mathbf{u}^{n+1}.\nabla u_f^n - \frac{\mu_f}{\rho_f} \nabla^2 \mathbf{u}^n \right] \\ + (\rho_b - \rho_f) \frac{\Delta t}{\tau + \Delta t} \mathbf{g} \end{aligned} \tag{35}$$

This equation corresponds to the implicit formulation of the penalization velocity. This expression is then injected into the discretized form of the fluid equation to obtain:

$$\begin{aligned} \rho_f \left(1 - \alpha_b \frac{\Delta t}{\tau + \Delta t} \right) \left[\frac{\mathbf{u}^{n+1} - \mathbf{u}^n}{\Delta t} + \mathbf{u}^{n+1}.\nabla \mathbf{u}^n - \frac{\mu_f}{\rho_f} \nabla^2 \mathbf{u}^n \right] = -\nabla P^n + \left(\rho_f + \alpha_b (\rho_b - \rho_f) \frac{\Delta t}{\tau + \Delta t} \right) \mathbf{g} \\ + \alpha_b \rho_b \frac{{\mathbf{u_b}}^{n} - \mathbf{u}^{n+1}}{\tau + \Delta t} \end{aligned} \tag{36}$$

References

[1] B.J. Alder, T.E. Wainwright, Studies in molecular dynamics. I. General method, J. Chem. Phys. 31 (2) (1959) 459–466.
[2] P. Angot, C.H. Bruneau, P. Fabrie, A penalization method to take into account obstacles in incompressible viscous flows, Numer. Math. 81 (4) (1999) 497–520.
[3] G.K. Batchelor, An Introduction to Fluid Dynamics, Cambridge University Press, 1967.
[4] S. Bnà, S. Manservisi, R. Scardovelli, P. Yecko, S. Zaleski, Numerical integration of implicit functions for the initialization of the VOF function, Comput. Fluids 113 (2015) 42–52.
[5] B. Bunner, G. Tryggvason, Dynamics of homogeneous bubbly flows. Part 2. Velocity fluctuations, J. Fluid Mech. 466 (2002) 53–84.
[6] B. Bunner, G. Tryggvason, Effect of bubble deformation on the properties of bubbly flows, J. Fluid Mech. 495 (2003) 77–118.
[7] D.K. Clarke, H.A. Hassan, M.D. Salas, Euler calculations for multielement airfoils using Cartesian grids, AIAA J. 24 (3) (1986) 353–358.
[8] D. Colombet, D. Legendre, F. Risso, A. Cockx, P. Guiraud, Dynamics and mass transfer of rising bubbles in a homogenous swarm at large gas volume fraction, J. Fluid Mech. 763 (2015) 254–285.
[9] A. Esmaeeli, G. Tryggvason, A direct numerical simulation study of the buoyant rise of bubbles at O(100) Reynolds number, Phys. Fluids 17 (9) (2005) 093303.
[10] A. Etcheverlepo, Développement de méthodes de domaines fictifs au second ordre, Doctoral dissertation, Université Sciences et Technologies-Bordeaux I, 2013.
[11] C. Garnier, M. Lance, J.L. Marié, Measurement of local flow characteristics in buoyancy-driven bubbly flow at high void fraction, Exp. Therm. Fluid Sci. 26 (6–7) (2002) 811–815.
[12] A. Guelfi, D. Bestion, M. Boucker, P. Boudier, P. Fillion, M. Grandotto, et al., NEPTUNE: a new software platform for advanced nuclear thermal hydraulics, Nucl. Sci. Eng. 156 (3) (2007) 281–324.
[13] Y. Hallez, D. Legendre, Interaction between two spherical bubbles rising in a viscous liquid, J. Fluid Mech. 673 (2011) 406–431.
[14] T. Kempe, J. Fröhlich, An improved immersed boundary method with direct forcing for the simulation of particle laden flows, J. Comput. Phys. 231 (9) (2012) 3663–3684.
[15] T. Kempe, M. Lennartz, S. Schwarz, J. Fröhlich, Imposing the free-slip condition with a continuous forcing immersed boundary method, J. Comput. Phys. 282 (2015) 183–209.
[16] J. Kim, D. Kim, H. Choi, An immersed-boundary finite-volume method for simulations of flow in complex geometries, J. Comput. Phys. 171 (1) (2001) 132–150.

[17] D. Legendre, On the relation between the drag and the vorticity produced on a clean bubble, Phys. Fluids 19 (1) (2007) 018102.
[18] J. Magnaudet, I. Eames, The motion of high-Reynolds-number bubbles in inhomogeneous flows, Annu. Rev. Fluid Mech. 32 (1) (2000) 659–708.
[19] J.B. Mclaughlin, Numerical simulation of bubble motion in water, J. Colloid Interface Sci. 184 (2) (1996) 614–625.
[20] R. Mei, J.F. Klausner, C.J. Lawrence, A note on the history force on a spherical bubble at finite Reynolds number, Phys. Fluids 6 (1) (1994) 418–420.
[21] R. Mittal, G. Iaccarino, Immersed boundary methods, Annu. Rev. Fluid Mech. 37 (2005) 239–261.
[22] A. Morente, A Penalization Method for the Simulation of Bubbly Flows, Thèse de doctorat, Institut National Polytechnique de Toulouse, 2018.
[23] C.S. Peskin, Flow patterns around heart valves: a numerical method, J. Comput. Phys. 10 (2) (1972) 252–271.
[24] C.S. Peskin, Numerical analysis of blood flow in the heart, J. Comput. Phys. 25 (3) (1977) 220–252.
[25] G. Riboux, D. Legendre, F. Risso, A model of bubble-induced turbulence based on large-scale wake interactions, J. Fluid Mech. 719 (2013) 362–387.
[26] G. Riboux, F. Risso, D. Legendre, Experimental characterization of the agitation generated by bubbles rising at high Reynolds number, J. Fluid Mech. 643 (2010) 509–539.
[27] F. Risso, K. Ellingsen, Velocity fluctuations in a homogeneous dilute dispersion of high-Reynolds-number rising bubbles, J. Fluid Mech. 453 (2002) 395–410.
[28] I. Roghair, Y.M. Lau, N.G. Deen, H.M. Slagter, M.W. Baltussen, M.V.S. Annaland, J.A.M. Kuipers, On the drag force of bubbles in bubble swarms at intermediate and high Reynolds numbers, Chem. Eng. Sci. 66 (14) (2011) 3204–3211.
[29] I. Roghair, J.M. Mercado, M.V.S. Annaland, H. Kuipers, C. Sun, D. Lohse, Energy spectra and bubble velocity distributions in pseudo-turbulence: numerical simulations vs. experiments, Int. J. Multiph. Flow 37 (9) (2011) 1093–1098.
[30] S. Schwarz, T. Kempe, J. Fröhlich, An immersed boundary method for the simulation of bubbles with varying shape, J. Comput. Phys. 315 (2016) 124–149.
[31] H. Sigurgeirsson, A. Stuart, W.L. Wan, Algorithms for particle-field simulations with collisions, J. Comput. Phys. 172 (2) (2001) 766–807.
[32] G. Tryggvason, B. Bunner, A. Esmaeeli, D. Juric, N. Al-Rawahi, W. Tauber, et al., A front-tracking method for the computations of multiphase flow, J. Comput. Phys. 169 (2) (2001) 708–759.
[33] M. Uhlmann, An immersed boundary method with direct forcing for the simulation of particulate flows, J. Comput. Phys. 209 (2) (2005) 448–476.
[34] M. Uhlmann, Interface-resolved direct numerical simulation of vertical particulate channel flow in the turbulent regime, Phys. Fluids 20 (5) (2008) 053305.
[35] S.O. Unverdi, G. Tryggvason, A front-tracking method for viscous, incompressible, multi-fluid flows, J. Comput. Phys. 100 (1) (1992) 25–37.
[36] S.P. Van der Pijl, A. Segal, C. Vuik, P. Wesseling, A mass-conserving level–set method for modelling of multi-phase flows, Int. J. Numer. Methods Fluids 47 (4) (2005) 339–361.
[37] S. Vincent, J.C.B. De Motta, A. Sarthou, J.L. Estivalezes, O. Simonin, E. Climent, A Lagrangian VOF tensorial penalty method for the DNS of resolved particle-laden flows, J. Comput. Phys. 256 (2014) 582–614.
[38] S. Vincent, A. Sarthou, J.P. Caltagirone, F. Sonilhac, P. Février, C. Mignot, G. Pianet, Augmented Lagrangian and penalty methods for the simulation of two-phase flows interacting with moving solids. Application to hydroplaning flows interacting with real tire tread patterns, J. Comput. Phys. 230 (4) (2011) 956–983.
[39] D. Wunsch, Theoretical and Numerical Study of Collision and Coalescence – Statistical Modeling Approaches in Gas-Droplet Turbulent Flows, Doctoral Dissertation, Thèse de doctorat, Institut National Polytechnique de Toulouse, 2009.
[40] T. Ye, R. Mittal, H.S. Udaykumar, W. Shyy, An accurate Cartesian grid method for viscous incompressible flows with complex immersed boundaries, J. Comput. Phys. 156 (2) (1999) 209–240.
[41] X. Yin, D.L. Koch, Lattice-Boltzmann simulation of finite Reynolds number buoyancy-driven bubbly flows in periodic and wall-bounded domains, Phys. Fluids 20 (10) (2008) 103304.
[42] R. Zenit, D.L. Koch, A.S. Sangani, Measurements of the average properties of a suspension of bubbles rising in a vertical channel, J. Fluid Mech. 429 (2001) 307–342.