Available online at www.sciencedirect.com
ScienceDirect
Journal of Computational Physics 219 (2006) 780–800
JOURNAL OF COMPUTATIONAL PHYSICS
www.elsevier.com/locate/jcp

A stabilized finite element method using a discontinuous level set approach for solving two phase incompressible flows

Emilie Marchandise [a,c,*], Jean-François Remacle [a,b]

[a] Université catholique de Louvain, Department of Civil Engineering, Place du Levant 1, 1348 Louvain-la-Neuve, Belgium
[b] Center for Systems Engineering and Applied Mechanics (CESAME), Université catholique de Louvain, 1348 Louvain-la-Neuve, Belgium
[c] Fonds National, de la Recherche Scientifique, rue d'Egmont 5, 1000 Bruxelles, Belgium

Received 31 October 2005; received in revised form 18 April 2006; accepted 24 April 2006
Available online 15 June 2006

Abstract

A numerical method for the simulation of three-dimensional incompressible two-phase flows is presented. The proposed algorithm combines an implicit pressure stabilized finite element method for the solution of incompressible two-phase flow problems with a level set method implemented with a quadrature-free discontinuous Galerkin (DG) method [E. Marchandise, J.-F. Remacle, N. Chevaugeon, A quadrature free discontinuous Galerkin method for the level set equation, Journal of Computational Physics 212 (2006) 338–357]. The use of a fast contouring algorithm [N. Chevaugeon, E. Marchandise, C. Geuzaine, J.-F. Remacle, Efficient visualization of high order finite elements, International Journal for Numerical Methods in Engineering] permits us to localize the interface accurately. By doing so, we can compute the discontinuous integrals without neither introducing an interface thickness nor reinitializing the level set.

The capability of the resulting algorithm is demonstrated with "large scale" numerical examples (free surface flows: dam break, sloshing) and "small scale" ones (two phase Poiseuille, Rayleigh–Taylor instability).

Keywords: Two-phase flow model; Free surface; Level set; 3D incompressible Navier–Stokes; Discontinuous Galerkin

1. Introduction

The study of two phase flows covers a wide range of engineering and environmental flows, including small-scale bubble dynamics, wave mechanics, open channel flows, flows around a ship or structure. The main challenge for solving time-dependent two-phase flow problems in three dimensions is to provide an accurate representation of the interface that separates the two different fluids. This involves the tracking of a discontinuity in the material properties like density and viscosity.

The principal computational methods used to solve incompressible two-phase flows are the front tracking methods [3–7], and the front capturing methods (volume of fluid [8,9] and level set [10,11]).

DOI of original article: 10.1016/j.jcp.2005.07.006.
* Corresponding author. Tel.: +32 1047 2061; fax: +32 1047 2179.
E-mail address: marchandise@gce.ucl.ac.be (E. Marchandise).

0021-9991/$ - see front matter
doi:10.1016/j.jcp.2006.04.015

A successful approach to deal with two phase flows, especially in the presence of topological changes, is the level set method [10]. Application of level sets in two-phase flow calculations have been extensively described by Sussman, Smereka and Osher in [11–13] and used by among others [14–17]. The level set function is able to represent an arbitrary number of bubbles or drops interfaces and complex changes of topology are naturally taken into account by the method. The level set function $\phi(\vec{x},t)$ is defined to be a smooth function that is positive in one region and negative in the other. The implicit surface $\phi(\vec{x},t)=0$ represents the current position of an interface. This interface is advected by a vector field $\mathbf{u}(\vec{x},t)$ that is, in case of two-phase flows, the solution of the Navier–Stokes equations. The elementary advection equation for interface evolution is:

$$\partial_t\phi+\mathbf{u}\cdot\nabla\phi=0. \tag{1}$$

In [1], we have developed a high order quadrature free Runge–Kutta discontinuous Galerkin (DG) method to solve the level set equation (1) in space and time. The method was compared with classical Hamilton–Jacobi ENO/WENO methods [13,18–20] and showed to be computationally effective and mass conservative. Besides, we showed that there was no need to reinitialize the level set.

Level sets are representing a fluid interface in an implicit manner. The main advantage of this approach is that the underlying computational mesh does not conform to the interface. Hence, discontinuous integrals have to be computed in the fluid formulation because both viscosities and densities are discontinuous in all the elements crossed by the interface. The most common approach is to define a zone of thickness 2ϵ in the vicinity of the interface ($|\phi|<\epsilon$) and to smooth the discontinuous density and viscosity over this thickness [11,17,21–23]. Smoothing physical parameters in the interfacial zone may be the cause of two problems. The first one is the introduction of non-physical densities and viscosities in the smoothed region, leading to possible thermodynamical aberrations [24]. The second problem is the obligation to keep the interface thickness constant in time. For ensuring that the smoothed region has a constant thickness, one has to reinitialize the level set so that it remains a distance function. In this work, we rather adopt a discontinuous approach [25,26] to compute the discontinuous integrals. The use of a recursive contouring algorithm [2] allows to localize the interface accurately. Consequently, we are able to compute the discontinuous integrals with a very high level of accuracy.

For the computation of the incompressible two-phase Navier–Stokes equations, various numerical methods have been developed. Among them are the projection methods [27–29], stabilized finite element methods [17,30,31] and artificial compressibility methods [32,33]. A key feature of stabilized methods is that they have proved to be LBB stable and to have good convergence properties [34,35].

In this work, we present a stabilized finite element method for computing flows in both phases and combine it with a discontinuous Galerkin level set method for computing the interface motion. The overall algorithm avoids the cost of the renormalization of the level set as well as the introduction of a non-physical interface thickness and exhibits good mass conservation properties.

The outline of this paper is as follows: we first present the governing equations in Section 2. Section 3 is devoted to the description of our computational method. We present the Navier–Stokes solver and the coupling with the discontinuous Galerkin method for the level set equation. Section 4 gives numerical examples to verify accuracy, stability and convergence properties.

2. Governing equation

In the present work, the three-dimensional flow field of two non-miscible laminar incompressible fluids is calculated. The two fluids are denoted respectively by (+) and (−) and have distinct viscosity and density (ρ_+,μ_+) and (ρ_-,μ_-). Fig. 1 shows an illustration of a configuration with two fluids.

The solution in both phases, denoted as phase (+) and phase (−), are obtained simultaneously. The non-dimensional equations are given by the incompressible Navier–Stokes equations:

$$\frac{\mathrm{D}\mathbf{u}}{\mathrm{D}t}=-\frac{\nabla p}{\rho(\phi)}+\frac{1}{\rho(\phi)}\frac{1}{Re}\nabla\cdot(2\mu(\phi)S)+\frac{\mathbf{e_g}}{Fr^2}+\frac{\kappa\mathbf{n}}{We}, \tag{2}$$

$$\nabla\cdot\mathbf{u}=0. \tag{3}$$

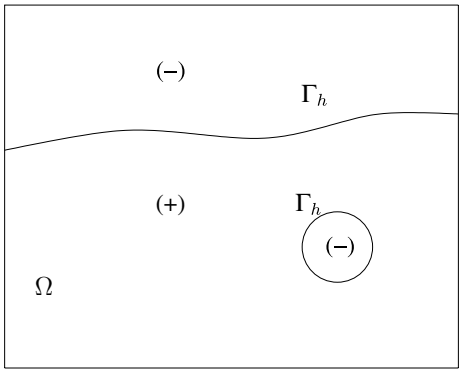


Fig. 1. Sketch of a two-fluid flow configuration.

Here the fluid velocity is denoted by $\mathbf{u}$, p is the static pressure, $S = \frac{1}{2}(\nabla\mathbf{u} + \nabla\mathbf{u}^{\mathrm{T}})$ is the deformation rate tensor, ρ is the fluid density, μ is the dynamic fluid viscosity, $\mathbf{n}$ is the unit normal vector along the interface, κ is the curvature of the interface and $\mathbf{e}_{\mathrm{g}}$ is the direction in which the gravity ($\vec{g}$) acts.

The above non-dimensional physical quantities are defined as:

$$\begin{aligned} &\mathbf{u} = \frac{\mathbf{u}^*}{U_{\mathrm{R}}}, \quad t = \frac{t^*}{L_{\mathrm{R}}/U_{\mathrm{R}}}, \quad p = \frac{p^*}{\rho_{\mathrm{R}} U_{\mathrm{R}}^2}, \\ &\rho = \frac{\rho^*}{\rho_{\mathrm{R}}}, \quad \mu = \frac{\mu^*}{\mu_{\mathrm{R}}}, \quad \mathbf{e}_{\mathbf{g}} = \vec{g}/g, \end{aligned} \tag{4}$$

where $*$ denotes the dimensional quantity, the subscript 'R' denotes the reference value, and g is the acceleration due to gravity.

The key flow parameters are the ratios of density and viscosity ($\frac{\rho_-}{\rho_+}, \frac{\mu_-}{\mu_+}$), the Reynolds number $Re = \frac{\rho_{\mathrm{R}} U_{\mathrm{R}} L_{\mathrm{R}}}{\mu_{\mathrm{R}}}$, the Froude number $Fr = \frac{U_{\mathrm{R}}}{\sqrt{g L_{\mathrm{R}}}}$ and the Weber number $We = \frac{\rho_{\mathrm{R}} U_{\mathrm{R}}^2 L}{\sigma_{\mathrm{R}}}$.

The fluid interface, denoted Γ_{h}, is impermeable. Assuming no mass transfer between the two fields yields a continuous velocity condition at the interface:

$$[u] = 0, \tag{5}$$

where the brackets denotes jumps across the interface. The jump in normal stresses along the fluid interface is balanced with the surface tension. Neglecting the variations of the surface tension coefficient σ gives the following boundary condition for momentum conservation at the interface:

$$[-pI + 2\mu S] \cdot \mathbf{n} = \sigma\kappa\mathbf{n} = \bar{\theta}. \tag{6}$$

It should be noted that for the flows computed in this paper, the surface tension can be neglected ($\bar{\theta} = 0$) and the non-dimensional Weber number set to zero.

The interface is represented by the zero level set of the function ϕ. This level set function ϕ is defined to be a smooth function which is positive in one fluid and negative in the other. The level set equation describes the evolution in time of the level set in particular of the interface which is the iso-zero level set.

Let us rewrite the level set equation (1) in a *conservative form*:

$$\partial_t\phi + \nabla \cdot (\mathbf{u}\phi) = \phi.\nabla \cdot \mathbf{u}. \tag{7}$$

Since we want to deal with incompressible flows, we have that $\nabla \cdot \mathbf{u} = 0$ and the conservation law simplifies in

$$\partial_t\phi + \nabla \cdot (\mathbf{u}\phi) = 0. \tag{8}$$

Within this framework, we do not need to evaluate $\nabla\phi$ anymore. Remember that poor accuracy of the evaluation of $\nabla\phi$ leads to instabilities in the absence of renormalization for the ENO method cited above [13,18–20]. Our formulation, in case of incompressible flows, has the advantage to avoid the cost of the renormalization step.

With this level set approach, the non-dimensional values of density are easily defined as:

$$\rho = \begin{cases} \rho_+ = \frac{\rho_+^*}{\rho_R}, & \phi > 0, \\ \rho_- = \frac{\rho_-^*}{\rho_R}, & \phi \leqslant 0. \end{cases} \tag{9}$$

The non-dimensional values of viscosity are defined the same way.

3. Numerical method

This section describes our numerical implementation for the computation of two-phase incompressible flows. As we have chosen to work with standard finite elements (for its ease of treatment of complex geometries), the computation of incompressible flows will involve two sources of potential numerical instabilities. One source is due to the presence of convective terms in the momentum equations. The other potential source of instability may be due to an inappropriate combination of interpolation functions to represent the velocity and pressure fields (violation of the LBB condition [36–38]). We will show within this section how to prevent those instabilities. We subsequently present the coupling between this flow solver and our interface solver [1]. We then describe a recursive contouring algorithm that allows an accurate localization of the interface. Discontinuous integrals are therefore computed accurately. At the end, we summarize the overall computational algorithm.

3.1. Nodal discretization

In this work, tetrahedral meshes are considered exclusively because they offer the maximum flexibility and robustness when dealing with complex geometries and mesh adaptation [39]. In tetrahedral meshes, there are 6 times more tetrahedrons than nodes and 12 times more faces than nodes [40]. In consequence, we have chosen to locate the unknown at the nodes rather than on the tetrahedrons (e.g. cell-centered finite volume schemes) or on the faces (e.g. staggered schemes).

3.2. Pressure stabilization

The choice of a fully nodal discretization introduces the well-known issue of pressure modes [41,42]. The pressure stabilized Petrov–Galerkin (PSPG) method introduced by Hughes and Franca [30] circumvents the Babuska–Brezzi condition and allows the use of equal-order P1–P1 velocity–pressure interpolation.

In order to introduce this method, let $\mathscr{T}_h$ be a partition of the domain Ω into tetrahedral elements Ω_e. The boundary Γ consists of two complementary subsets Γ_d and Γ_n on which given Dirichlet-type and Neumann-type boundary conditions apply, respectively.

Be Ω_e an element with boundary Γ_e and outer radius h_e. Let $H^1(\Omega)$ be the Hilbert space of square integrable functions with square integrable first order derivatives.

To derive the finite element discretization of the weak form of the equations of motion (2), we first introduce the trial and weight function spaces for the semi-discrete formulation:

$$\mathbf{S}^h = \{\mathbf{u}^h | \mathbf{u}^h \in H^1(\Omega),\ \mathbf{u}^h = \bar{\mathbf{u}} \text{ on } \Gamma_d\}, \tag{10}$$

$$\mathbf{V}^h = \{\mathbf{v}^h | \mathbf{v}^h \in H^1(\Omega),\ \mathbf{v}^h = 0 \text{ on } \Gamma_d\}. \tag{11}$$

This PSPG method is a full Petrov–Galerkin formulation in which a weight function q^h is applied to the term of the continuity equation (3) and a perturbed weight function $\tilde{\mathbf{v}}^h$

$$\tilde{\mathbf{v}}^h = \mathbf{v}^h + \tau_\epsilon \nabla q^h \tag{12}$$

is applied to all the terms of the momentum equation (2). By subsequently integrating those equations over the computational domain and by using the divergence theorem, we obtain the PSPG formulation. This formulation reads as follows:

Find $(\mathbf{u}^h, p^h) \in \mathbf{S}_\mathbf{u}^h \times \mathbf{S}_\mathbf{p}^h$ such that $\forall (\mathbf{v}^h, q^h) \in \mathbf{V}_\mathbf{v}^h \times \mathbf{V}_\mathbf{q}^h$

$$\int_\Omega \partial_t \mathbf{u}^h \mathbf{v}^h \, \mathrm{d}v + \int_\Omega \mathbf{v}^h \mathbf{u}^h \cdot \nabla \mathbf{u}^h \, \mathrm{d}v + \int_\Omega \nabla \mathbf{v}^h \frac{\mu}{\rho Re} \nabla \mathbf{u}^h \, \mathrm{d}v + \int_\Omega \mathbf{v}^h \frac{\nabla p^h}{\rho} \, \mathrm{d}v = \int_\Omega \mathbf{v}^h \frac{\mathbf{e_g}}{Fr^2} \, \mathrm{d}v, \tag{13}$$

$$\int_\Omega q^h \nabla \cdot \mathbf{u}^h \, \mathrm{d}v + ST = 0. \tag{14}$$

The stabilization term ST

$$ST = \sum_e \int_{\Omega_e} \tau_e \nabla q^h \mathbf{R}(p^h, \mathbf{u}^h) \, \mathrm{d}v \tag{15}$$

contains the residual of the momentum equation

$$\mathbf{R}(p^h, \mathbf{u}^h) = \partial_t \mathbf{u}^h + (\mathbf{u}^h \cdot \nabla)\mathbf{u}^h - \frac{\mu}{\rho Re} \nabla^2 \mathbf{u}^h + \frac{1}{\rho} \nabla p^h - \frac{\mathbf{e_g}}{Fr^2}. \tag{16}$$

The stabilization parameter τ_ϵ is of order $\mathcal{O}(h_e^2/\nu)$ in the diffusion dominated case and of order of $\mathcal{O}(h_e)$ in the advection dominated case [43,44].

3.3. Convective stabilization

It is well known that in the context of finite elements, there is a need to stabilize the advective term. One popular possibility is to use the streamline upwind/Petrov–Galerkin (SUPG) formulation. In this work, we have rather chosen to use an upwind finite volume stabilization with the finite volumes being the median cells. This work is based on the one of Barth and Selmin [45,46] in which they show the link between the standard finite element Galerkin formulation on tetrahedral meshes and the finite volume formulation with control volumes being the median cells. Those median cells are obtained by connecting the centroid of each face of the surrounding tetrahedrons to the midpoint of the edges of the tetrahedron. We have chosen to use a full upwind method when evaluating the convective flux on the faces of this control volume. Those fluxes at the faces of the dual cell are computed using a linear approximation of the variables:

$$\mathbf{u}_{\text{face}} = \mathbf{u}_{\text{node}} + \Delta x^{\mathrm{T}} \nabla \mathbf{u}_{\text{node}}, \tag{17}$$

where the nodal gradients $\nabla \mathbf{u}_{\text{node}}$ are computed with a least square reconstruction method [47].

3.4. Coupling between the two solvers

In this section, we describe how we have coupled our two-phase incompressible flow solver with the interface solver that is the RK-DG method for the level set equation. First, we have used the same computational mesh for both solvers.

The flow solver uses continuous linear approximations (N^1) for the velocity $\mathbf{u}$ while the interface DG solver uses piecewise continuous high order (p) approximations (N^p) for the level set ϕ

$$\mathbf{u} = \sum_{i=1}^{4} \mathbf{u}_i N_i^1 \quad \text{and} \quad \phi = \sum_{i=1}^{n_p} \phi_i N_i^p.$$

The reason why higher order polynomials are used for discretizing the level set is that Eq. (1) involves $\nabla\phi$ and $\mathbf{u}$. We choose a level set for which the gradient is at least in the space of the velocity.

Projection operators are used to project the velocity space to the level set space and conversely. The elementary projection operator $\mathbb{P}$ that projects the velocity variable from a polynomial space of order $p = a$ to a space of order $p = b$ is given by:

$$u^b = \mathbb{P}u^a = M^{-b} M^{ab} u^a, \tag{18}$$

where the mass matrix for the element Ω_e is given by:

$$M_{ij}^{ab} = \int_{\Omega_e} N_i^a N_j^b \, \mathrm{d}v. \tag{19}$$

On the other way, to project the ϕ unknowns to the flows dofs, we take the average of ϕ at the nodes of the tetrahedron. Fig. 2 illustrates the coupling described above.

3.5. Evaluation of the discontinuous integrals

Since the interface is represented implicitly by the zero level set, the computational mesh does not conform with the interface. It follows that if the density and viscosity are discontinuous across the interface, the two following integrals of Eq. (13) are discontinuous:

$$\int_{\Omega_e} \mathbf{v}^h \frac{\nabla p^h}{\rho} \, dv \quad \text{and} \quad \int_{\Omega_e} \nabla \mathbf{v}^h \frac{\mu}{\rho Re} \nabla \mathbf{u}^h \, dv. \tag{20}$$

An accurate localization of the interface enables us to divide the discontinuous integrals into two continuous integrals and to compute those exactly by numerical integration:

$$\int_{\Omega_e} \mathbf{v}^h \frac{\nabla p^h}{\rho} \, dv = \int_{V_+} \mathbf{v}^h \frac{\nabla p^h}{\rho_+} \, dv + \int_{V_-} \mathbf{v}^h \frac{\nabla p^h}{\rho_-} \, dv, \tag{21}$$

$$\int_{\Omega_e} \nabla \mathbf{v}^h \frac{\mu}{\rho Re} \nabla \mathbf{u}^h \, dv = \int_{V_+} \nabla \mathbf{v}^h \frac{\mu_+}{\rho_+ Re} \nabla \mathbf{u}^h \, dv + \int_{V_-} \nabla \mathbf{v}^h \frac{\mu_-}{\rho_- Re} \nabla \mathbf{u}^h \, dv. \tag{22}$$

The question is how accurately can we determine the shape of the interface? To answer this question, we consider a triangular element of coordinates $(0,0)$ $(1,0)$ and $(0,1)$ that is crossed by the interface. Consider the following level set function:

$$\phi(x,y) = (x-1)^2 + y^2 - (0.5)^2$$

for which the iso-zero level set (interface) is the circle of radius $(1,0)$. Remember that we use high order polynomials to represent the level set within the interface solver, while within the fluid solver we only keep in memory the nodal values of the level set.

The first idea to localize the interface is to interpolate linearly the iso-zero level set (interface) from the nodal values of the level set. Unfortunately, since the level set (even if initially defined as a distance function) does not remain a distance function throughout the computation, this approach leads to a significant localization error of the interface. Fig. 3 illustrates the error made with the first approach.

A more accurate approach (illustrated in Fig. 4) is to use the recursive contouring algorithm described in [2] combined with a fast search tree method.

The approach is quite simple. It consists of dividing the elements recursively into sub-elements and performs a linear approximation on every edge of those sub-elements to find the points on the interface (points

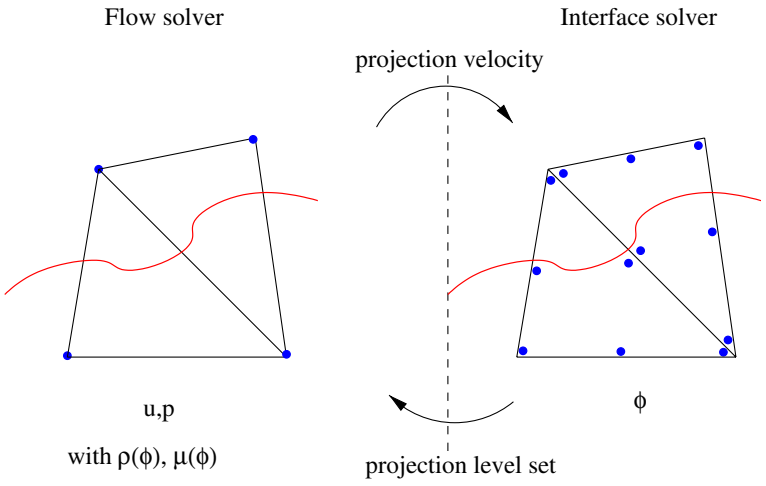


Fig. 2. Coupling between flow solver and the interface solver.

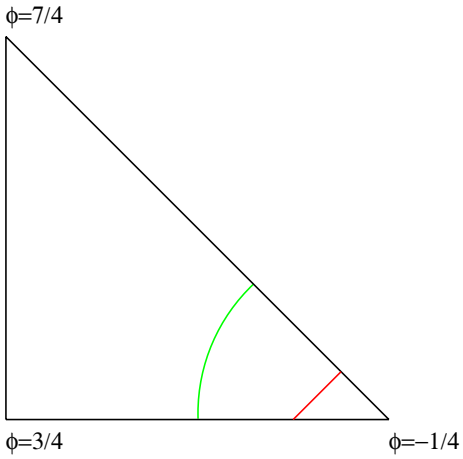


Fig. 3. Error made on the localization of the interface with the simple linear approximation. The level set function is defined as $\phi(x,y) = (x-1)^2 + y^2 - (0.5)^2$. The green curve is the exact interface and the red line is the localized interface. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

with a zero value of the level set). Those points (blue dots in Fig. 4) are then stored in a search tree (approximate nearest neighbor (ANN) [48,49]). Next, using the search tree, we compute the smallest distance (denoted d) from fluid mesh vertexes to the tree points. The sign (+/−) is kept, so we have a signed distance. We see in Fig. 4, that with those new signed distance values, the linearly interpolated interface (red line) is much more accurate. We also see that the recursive algorithm is only performed if the sub-element is crossed by the interface. The level of recursion is a given value for the numerical simulation.

Fig. 5 shows an example of the recursive algorithm of level 4 to localize the interface.

Our approach is trivially extended in three dimensions for our tetrahedral meshes. The localized interface is then a planar surface which can be either a triangle or a quadrangle (see Fig. 6). The first way divides the tetrahedron in a smaller tetrahedron and a prism, while the latter cuts the tetrahedron into two prisms. Volumes and therefore discontinuous integrals can then be easily evaluated.

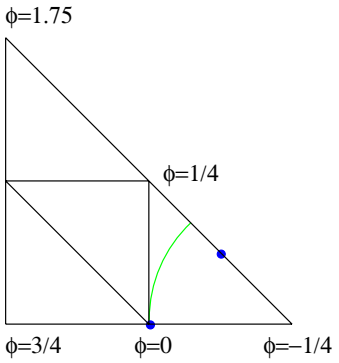
The first integral (Eq. (21)) contains a linear term (∇p^h is constant inside each element Ω_e and $\mathbf{v}^h$ is a linear weight function). It follows that one integration point is sufficient for exact integration in a tetrahedron and the point is located at the center of gravity (denoted cg) of the tetrahedron. Consider that the interface cuts the tetrahedron as in Fig. 6(a), the integral can then be computed as follows:

$$\begin{aligned}
\int_{\Omega_e} \mathbf{v}^h \frac{\nabla p^h}{\rho}\,\mathrm{d}v &= \nabla p^h \left(\int_{V_+} \mathbf{v}^h \frac{1}{\rho_+}\,\mathrm{d}v + \int_{V_-} \mathbf{v}^h \frac{1}{\rho_-}\,\mathrm{d}v \right) \\
&= \nabla p^h \left(\int_{V_+} \mathbf{v}^h \frac{1}{\rho_+}\,\mathrm{d}v + \int_{V/V_+} \mathbf{v}^h \frac{1}{\rho_-}\,\mathrm{d}v \right) \\
&= \nabla p^h \left(\frac{1}{\rho_+} \mathbf{v}^h(cg_{V_+}) V_+ - \frac{1}{\rho_-} \mathbf{v}^h(cg_{V_+}) V_+ + \frac{1}{\rho_-} \mathbf{v}^h(cg_{\Omega_e}) \Omega_e \right) \\
&= \nabla p^h \left(\left(\frac{1}{\rho_+} - \frac{1}{\rho_-} \right) \mathbf{v}^h(cg_{V_+}) V_+ + \frac{1}{4\rho_-} \Omega_e \right).
\end{aligned} \tag{23}$$

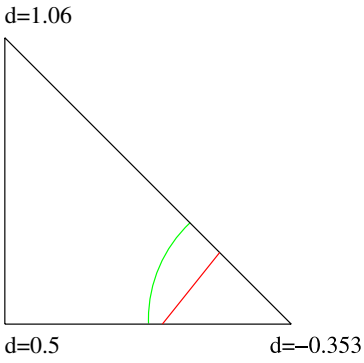
The second integral (Eq. (21)) contains a constant terms ($\nabla \mathbf{v}^h$ and $\nabla \mathbf{u}^h$ are constant within each element) and can be integrated exactly as follows:

$$\int_{\Omega_e} \nabla \mathbf{v}^h \frac{\mu}{\rho Re} \nabla \mathbf{u}^h\,\mathrm{d}v = \int_{V_+} \nabla \mathbf{v}^h \frac{\mu_+}{\rho_+ Re} \nabla \mathbf{u}^h\,\mathrm{d}v + \int_{V_-} \nabla \mathbf{v}^h \frac{\mu_-}{\rho_- Re} \nabla \mathbf{u}^h\,\mathrm{d}v = \frac{\nabla \mathbf{v}^h \nabla \mathbf{u}^h}{Re} \left(\frac{\mu_+}{\rho_+} V_+ + \frac{\mu_-}{\rho_-} V_- \right). \tag{24}$$

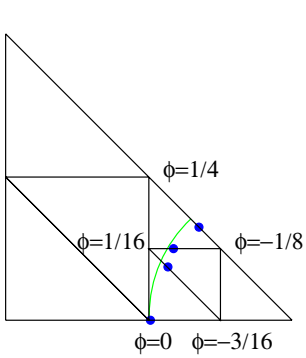
Our discontinuous integration approach based on a combination of a recursive contouring algorithm and an ANN Kd-tree presents three advantages:



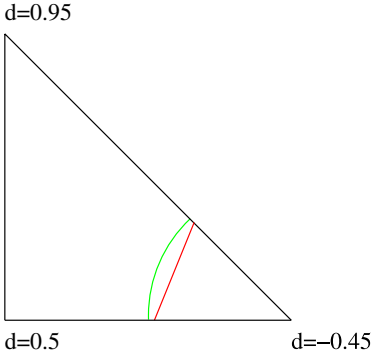
(a) Recursive algorithm of level 1



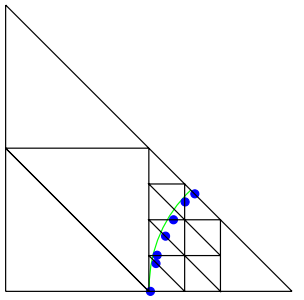
(b) Localized interface.



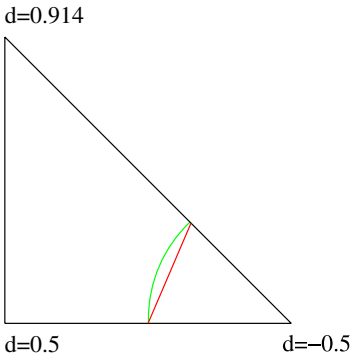
(a) Recursive algorithm of level 2



(b) Localized interface.



(a) Recursive algorithm of level 3



(b) Localized interface.

Fig. 4. Error made on the localization of the interface with the recursive contouring algorithm. The level set function is defined as $\phi(x,y) = (x-1)^2 + y^2 - (0.5)^2$. The green curve is the exact interface. The set of points are marked by a blue dot and the localized interface is the red line. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

- We do not introduce non-physical parameters.
- This approach is computationally more efficient in computational time compared to renormalization procedures.
- As we do not reinitialize, we are not faced with the problem of not moving the interface during the reinitialization procedure, which leads to mass losses.

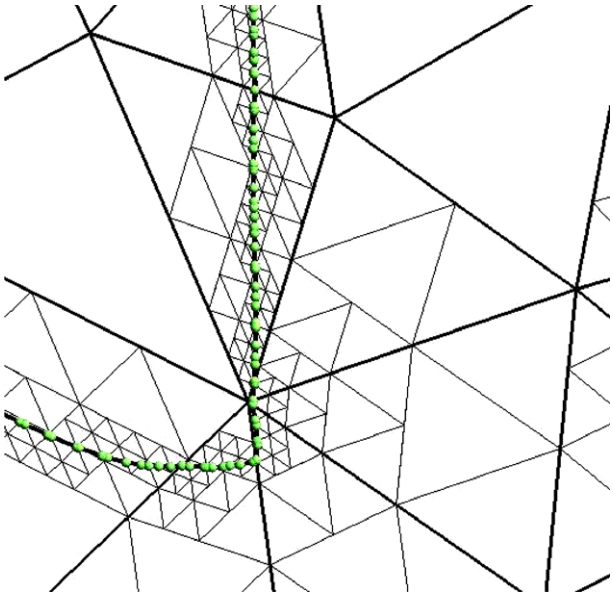


Fig. 5. Contouring algorithm to accurately localize the interface. In black, the computational mesh. In grey, the 4 level refined elements. In green, a set of points on the interface. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

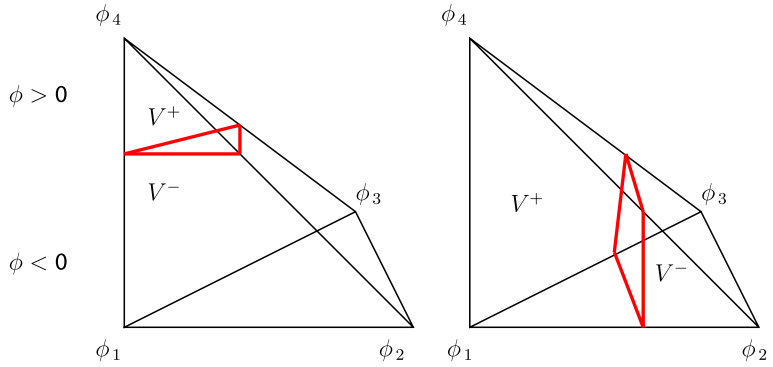


Fig. 6. Tetrahedron intersected by the interface.

The errors we make with this approach are the follows:

- The first one comes from the linear interface approximation by a triangle or a quadrangle, and yields an error of $\mathcal{O}(h^2)$ [26].
- The second one comes from the P1/P1 choice of finite element discretization that suppose a linear velocity/pressure field and thus a constant gradient of velocity/pressure inside an element. As a consequence we cannot satisfy exactly the boundary conditions at the interface (6). We will show in section 4 that this error can be significantly reduced by using a refined mesh near the interface.

3.6. Time integration

For time integration a second-order three-point backward difference scheme is employed.

At each time step, solving (13), results is resolving a system of nonlinear equations. To solve this system, an inexact Newton method based on a finite difference Newton–Krylov algorithm [50] is used. The iterative solu-

tion of the large sparse linear system of equations that arises at each Newton iteration is solved by the GMRES method preconditioned by the RAS [51] algorithm.

3.7. Summary of the algorithm

Our computational approach for numerical modeling of two-phase flows can be summarized as follows. First we initialize all the variables (velocity $\mathbf{u}$, pressure p, level set ϕ). Then,

For each time t^n, $n = 0, 1, 2 \ldots$:

- First solve the Navier–Stokes equation from t^{n-1} until t^n and find a velocity $\mathbf{u}^n$. This equation is solved implicitly for the velocity and pressure variables but uses the previous position of the level set ϕ^{n-1} to identify the fluid variable such as viscosity and density. To go to step 2, project the velocity field $\mathbf{u}$ onto the dofs of the interface solver.
- Second advance explicitly the level set (Eq. (8)) from t^{n-1} until t^n using sub-time steps with a linear velocity varying in time between $\mathbf{u}^{n-1}$ until $\mathbf{u}^n$ and find a new position for the level set ϕ^n. Then project the level set function ϕ onto the dofs of the fluid solver.
- Third go back to step 2, with this new position ϕ^n and repeat those steps until $\|\phi^{n-1} - \phi^n\| < eps$.

4. Numerical results and discussion

In this section, the coupling of the incompressible two-phase flow solver and the interface solver described in the previous section is tested and applied to several 2D and 3D two-phase flows problems. As we are only working with tetrahedral meshes, we use for the computation of the 2D flows a 2D mesh that is extruded over a thin layer in the third dimension.

Our methods were implemented in Standard C++, compiled with GNU g++ v3.3, and run on one CPU of a 2.4 GHz Intel(R) Pentium(R) 4 with 512 KB of L2 cache.

4.1. Poiseuille two-phase flow

The horizontal stratified flow of two fluids between parallel walls is first considered. For long times a steady solution is obtained for the two-phase flow problem which can be described by an analytical solution. This simple test case does not include any deformation of the interface but is interesting anyway because it allows to characterize the error due to the evaluation of the discontinuous integrals.

The height of the channel is $H = 2$ m and the length $L = 4$ m. The kinematic viscosities of the two fluids are, respectively, $\nu_1 = 0.1$ and $\nu_2 = 0.02$ m/s^2. The two fluids are assumed to have the same density. The pressures are imposed on the left and right limit of the calculation domain to ensure a pressure difference of $\Delta p = -2.0$ Pa. The interface between the two phases is located at the half height of the channel. The numerical solution was obtained within one iteration with a huge time step of 10^4 s. The residual decreased of 6 orders of magnitude.

The comparison between the analytical and numerical velocity field is depicted at Fig. 8. The solutions have been computed for three different structured and unstructured meshes (see Fig. 7) of mesh size $h = 0.1$. The first mesh 7(a) is structured and build such that the interface between the two phases corresponds with the mesh. The second 7(b) is a structured mesh in which the interface cuts the tetrahedron and the third one 7(c) is an unstructured mesh that is refined near the interface ($h = 0.01$).

Table 1 shows the L_2 error on the velocity field for those three meshes. We see that, with mesh a, we capture well the analytical solution while with mesh b, errors are significant. The error comes from the evaluation of integral (22). Indeed the boundary conditions are not exactly satisfied. We should have $[p] = 0$ and $[\mu \frac{\partial u}{\partial n}] = 0$, where the brackets denote the jump at the interface. The first condition is satisfied while the second cannot be satisfied since we have $[\frac{\partial u}{\partial n}] = 0$ and $[\mu] \neq 0$. This error can be reduced by refining the mesh near the interface. This is done in mesh c.

Another way to reduce the error is to enrich the velocity field with an extended basis (XFEM) whose gradient is discontinuous across the interface. This has been done by Chessa and Belytschko in [14,15]. The only

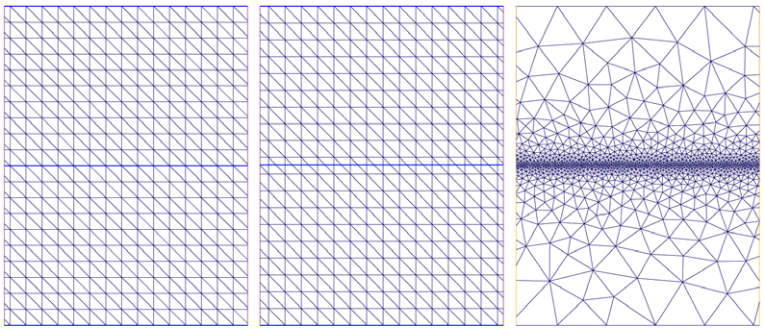
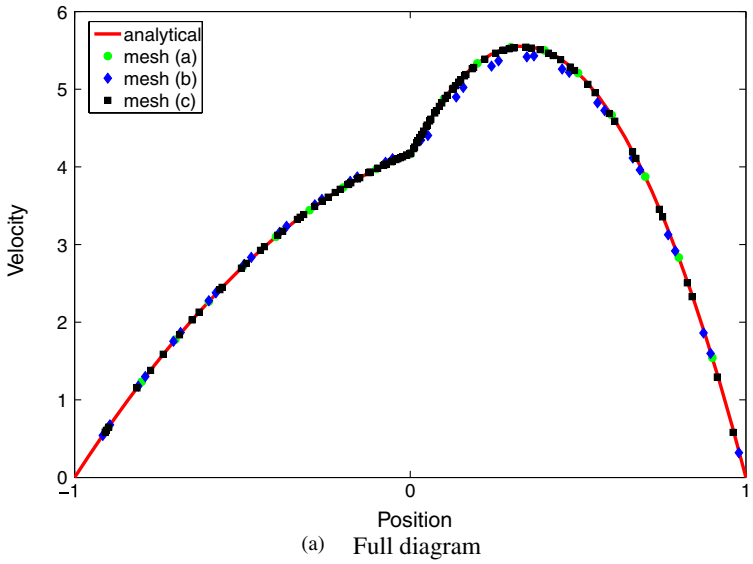
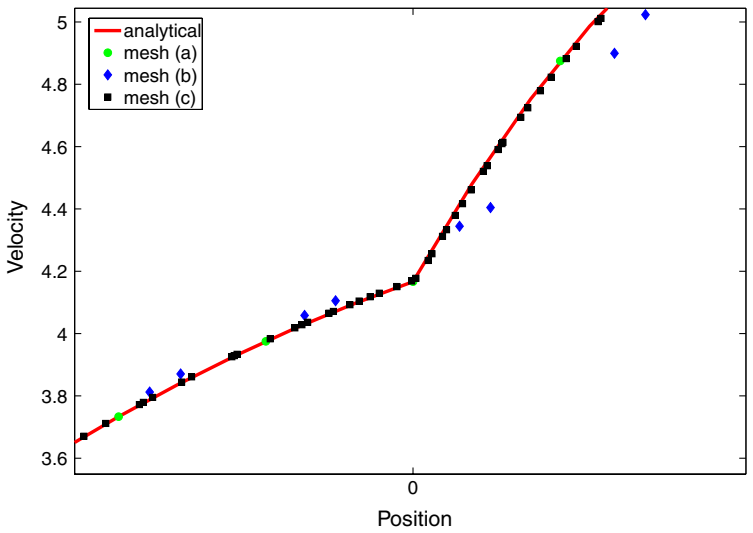


Fig. 7. Two-phase Poiseuille: snapshots of the three computational meshes. The interface between the two phases is located in the middle of the channel height.



(a) Full diagram



(b) Zoom near the interface

Fig. 8. Two-phase Poiseuille: comparison between numerical and experimental velocity field for different meshes.

Table 1
L_2 norm of the error on the velocity field for different meshes

Mesh	L_2 error
(a)	0.0064
(b)	0.0321
(c)	0.0049

drawback of this method is the additional computational cost and the loss of simplicity. Indeed at each time step, the number of enrichments varies and so the size of the vectors and matrices of the system to be solved.

4.2. Sloshing in a rectangular tank

This test case consists at looking at the free oscillations of a liquid contained in a two-dimensional tank of length d (Fig. 9(a)).

The oscillation is caused by the following sinusoidal free-surface set-up at time $t = 0$:

$$\eta = d + a_0 \sin(\pi(0.5 - x)), \tag{25}$$

where d is the mean water depth set to unity and a_0 is the amplitude of the wave equal to 0.01. For this small amplitude wave test case, Prosperetti [52] and Wu et al. [53] have proposed an analytical solution to the problem based on the linearized Navier–Stokes equations with linear free surface boundary conditions.

The computational domain is a rectangle of dimensions $[0, d] \times [0, 1.3d]$ in the XZ plane that is extruded over a length 0.1 in the Y-direction. The side walls are assumed to be slip walls, while the bottom wall is a no-slip wall and the upper wall is assigned a zero pressure boundary condition. The level set is initialized as the distance function to the water elevation. We take as initial pressure field an hydrostatic pressure distribution.

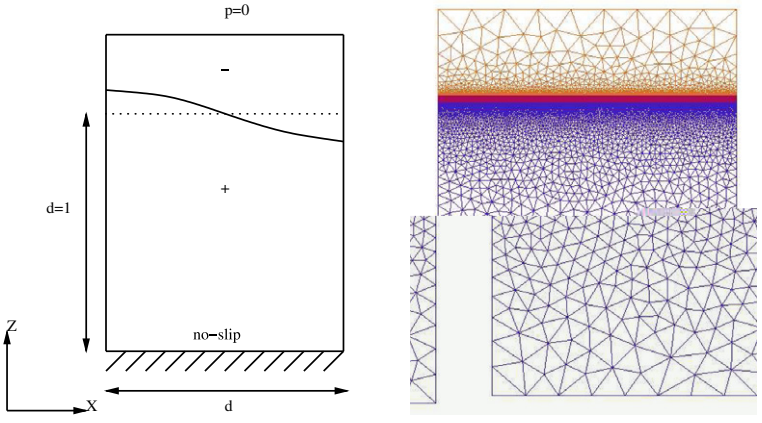
For this flow $U_R = \sqrt{gd}$ is assumed as a reference value for the velocity, $L_R = d$ is the reference length. The non-dimensional numbers are the Froude number $Fr = 1$ and the Reynolds number $Re = 100$ and the two superimposed fluids are assumed to have a viscosity ratio of 1/100 and a density ratio of 1/100. With this assumption we can assume that the upper fluid has negligible dynamical effects.

The grid used is made of 24,648 tetrahedrons refined near the interface and is depicted at Fig. 9(b).

Fig. 10 shows the non-dimensional time evolution ($t = t^*\sqrt{g}d$) of the numerical solution compared with the analytical solution of Prosperetti [52,53]. The physical parameters for the analytical solution are the wave number $k = \frac{2\pi}{2d} = \pi$ and the viscosity $\nu = \frac{U_R L}{Re} = \frac{\sqrt{g}}{Re}$. Analytical and numerical solutions are in very good agreement.

4.3. Dam break problem

The dam break problem consists in the sudden collapse of a rectangular column of fluid onto a horizontal surface and it is used to model the sudden failure of a Dam. This problem has been widely studied in the



(a) Schematic description (b) Mesh

Fig. 9. Sloshing in a rectangular tank.

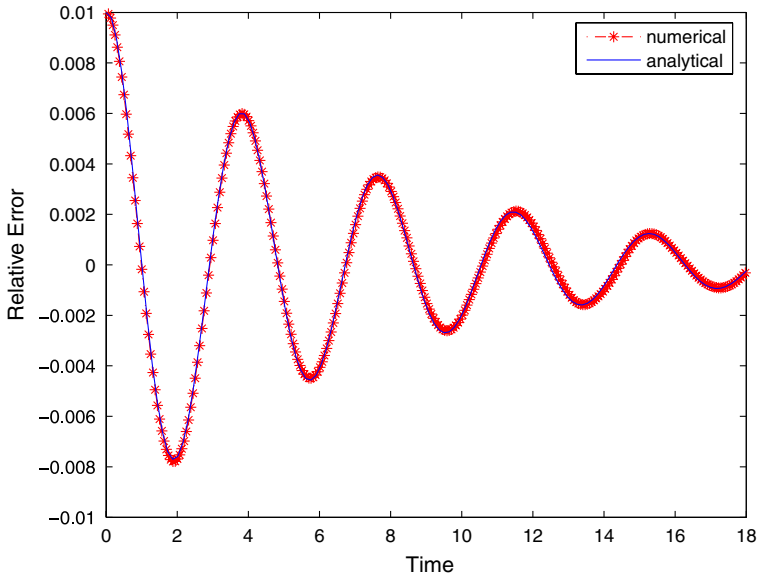


Fig. 10. $Re = 100$.

literature by many different experimental [54], theoretical [55] and numerical methods [56,57]. Through the Saint Venant or Shallow water model, the dam break flow admits analytical solutions under the hydrostatic pressure distribution and perfect fluid assumption.

The problem emphasizes the influence of gravity and viscosity. The strong deformation of the interface and the unsteady character of the flow confer on this test case a reference point of view to validate the model.

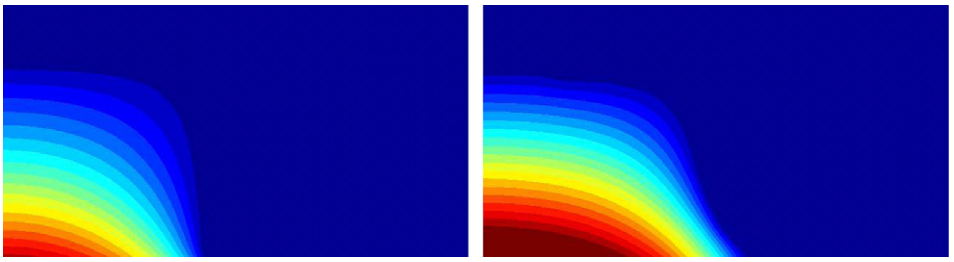
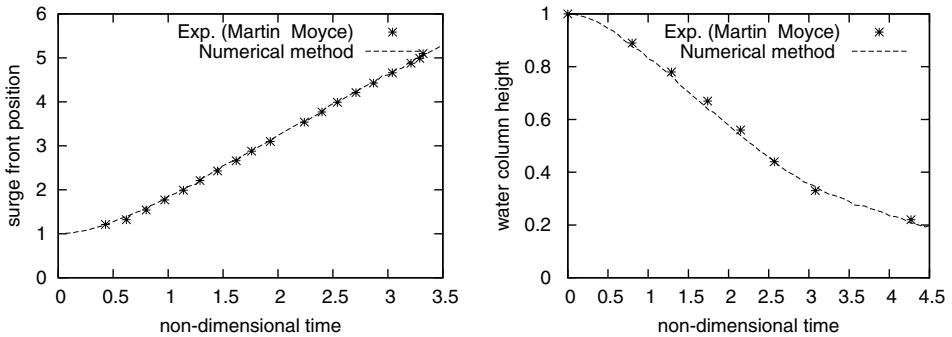


Fig. 11. Dam break problem: pressure field at the initial stages of the dam break at the non-dimensional times $t = 0.1$ s, 0.3 s.



(a) Surge front position (b) Remaining water column height

Fig. 12. Dam break problem: comparison between present method and experimental results.

The fluid characteristics (water and air referred by the subscripts l and g for liquid and gas) are the follows:

- density $\rho_l = 1000\ \mathrm{kg\ m^{-3}}$, $\rho_g = 1\ \mathrm{kg\ m^{-3}}$;
- dynamic viscosity $\mu_l = 10^{-3}$ Pa s, $\mu_g = 10^{-5}$ Pa s.

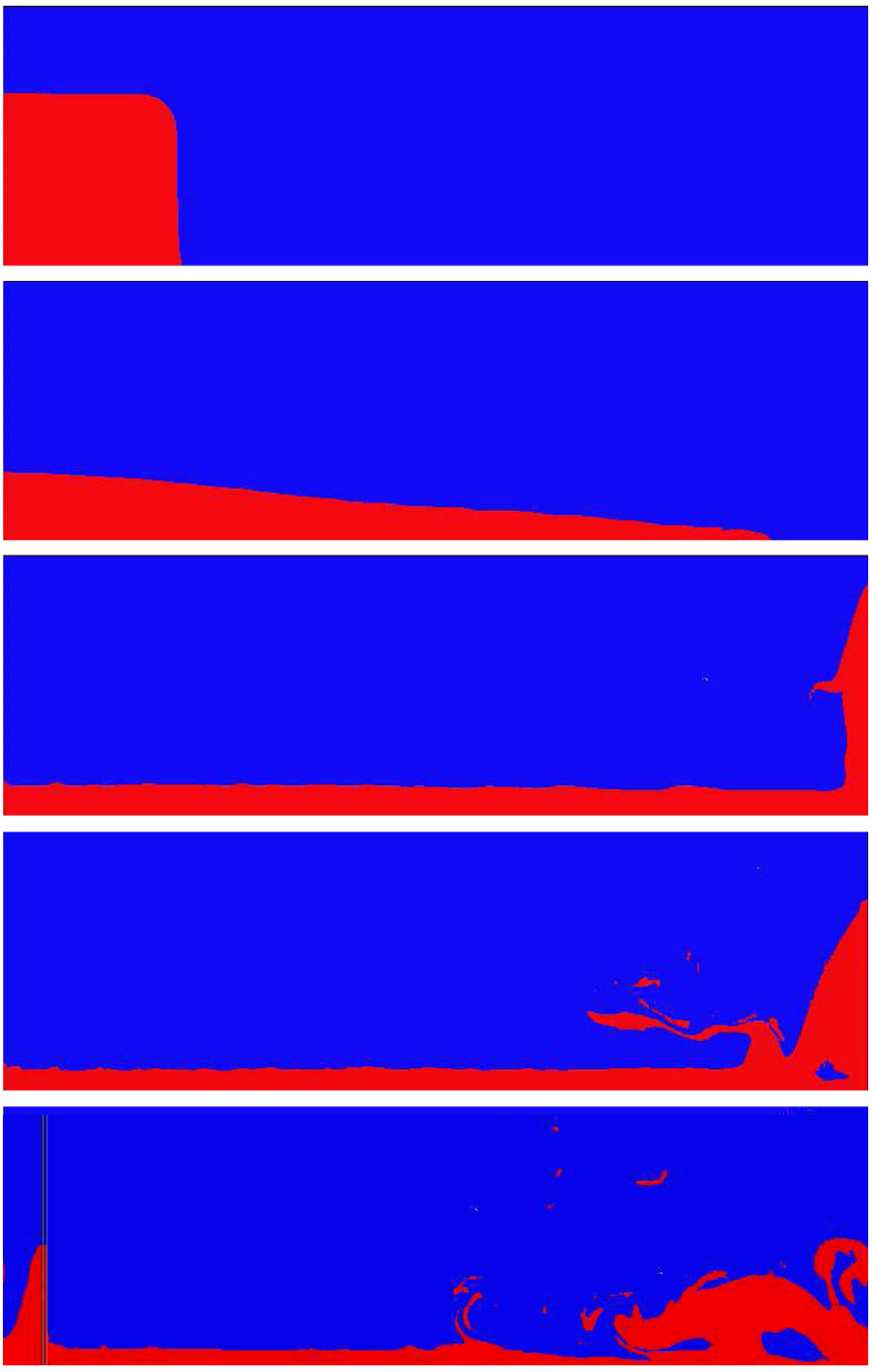


Fig. 13. Dam break problem: free surface position at the selected non-dimensional times ($t = 0.15, 2.81, 5.01, 6.26$, and 9.08 s).

The Reynolds number is 40,000. Non-slip boundary conditions are applied to all the walls, therefore the water column collapses under gravity.

The calculation domain is described by the length $L = 6$ m and the height $H = 1.5$ m. The height of the water column is $h_l = 1$ m. The mesh is unstructured and made of 10,218 nodes.

Fig. 11 depicts the pressure field at the initial stages of the dam break.

The history of the dimensionless horizontal displacement of the water front is shown in Fig. 12. For comparison the experimental values from Martin and Moyce [54] are added to the diagram. In this diagram, the time is non-dimensionalized by $t = \sqrt{h_l/g}$.

Finally, Fig. 13 displays snapshots of the free surface position at selected times.

4.4. 3D dam break with a cylindrical obstacle

To show the ability to simulate 3D free surface flows, we consider the breaking of a cubic water column in a domain containing a cylindrical obstacle.

The computational domain is described by the length $L = 6$ m and the height $H = 1.5$ m. The height and the width of the water column are $h_l = 1$ m and $h_w = 1.5$ m. The cylinder is located 1.3 m downstream the water column and has a diameter of 0.4 m. The mesh depicted at Fig. 14 is unstructured and made of 39,421 nodes.

Fig. 15 and 16 show snapshots of the water surface position at selected times. We see that from the time the water front reaches the cylinder, the flow shows clearly three dimensionality.

Table 2 compares the relative error of mass conservation at $t = 1$ s and $t = 2$ s for different meshes and different order of polynomials for the level set. The mass errors are calculated by:

$$\epsilon_A = \frac{A(t_f) - A(0)}{A(0)}, \tag{26}$$

where $A(t_f)$ is the total area of the liquid at the final time t_f.

From Table 2 we see that mesh refinement as well as increasing the order p improves the mass conservation. We show that those results are obtained within reasonable computational time. The computational time may however be dramatically reduced by performing mesh adaptation near the interface. We are currently working on mesh adaptivity.

4.5. Rayleigh–Taylor instability

The single-mode Rayleigh–Taylor instability is a classical benchmark for the validation of an interface capturing code. The instability is associated with the acceleration of a heavy fluid into a light one under the action of a gravitational field and is generic to a wide range of physical phenomena.

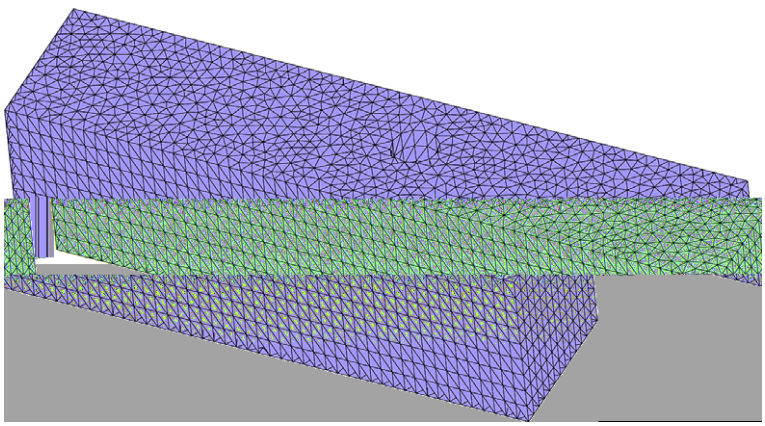


Fig. 14. Computational mesh made of 30.820 nodes.

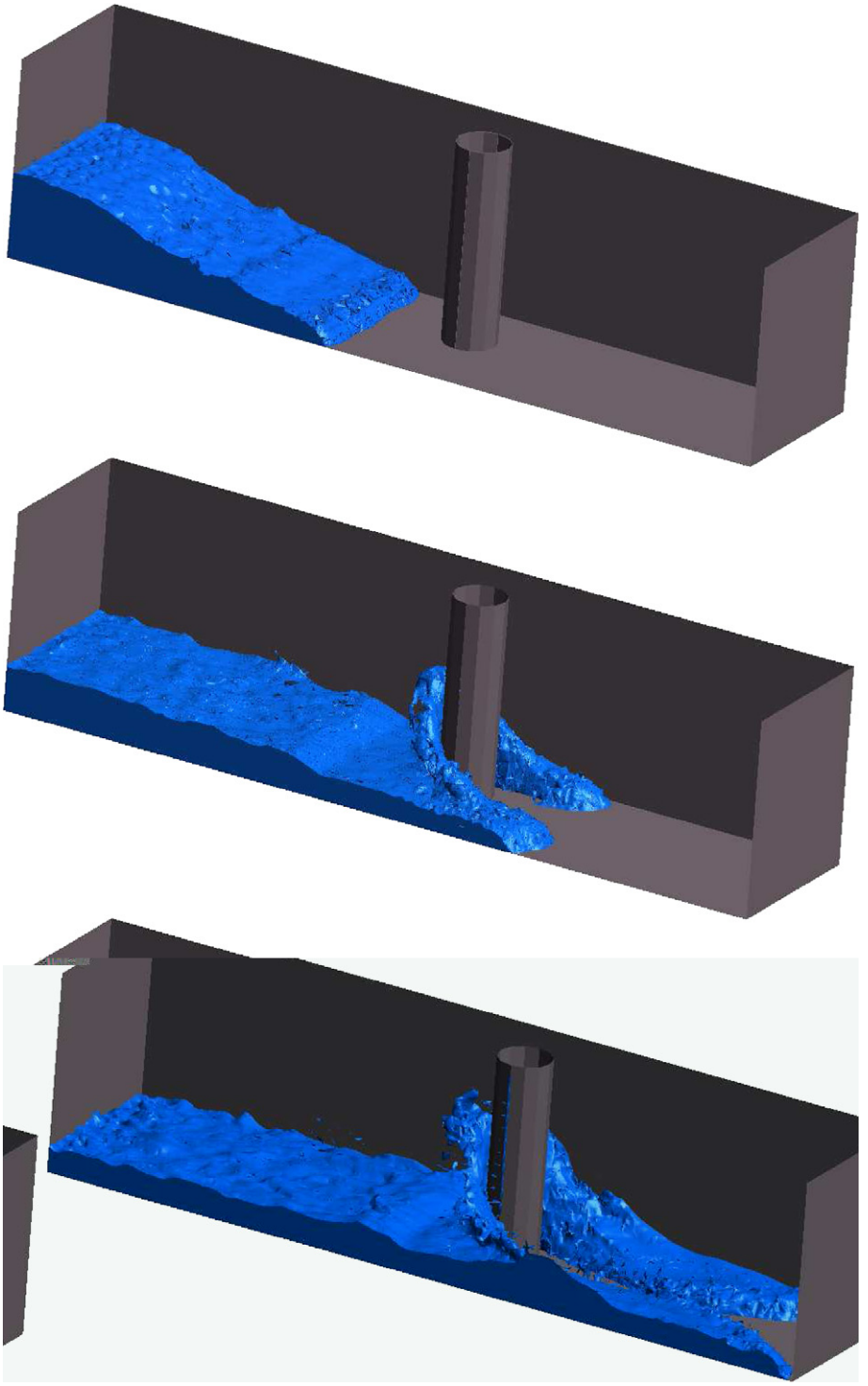


Fig. 15. Perspective view of the free surface for the dam break with a cylindrical obstacle. Free surface position at the selected times: $t = 0.5$ s, 1 s and $t = 2.5$ s.

A 0.5 m wide, 4 m high rectangular domain extruded by 0.001 in the third direction is discretized with $32 \times 265 \times 2$ grid points. Our non-dimensional parameters are the Reynolds number and the Atwood number. The Atwood number is defined as $A = (\rho_h - \rho_l)/(\rho_h + \rho_l)$.

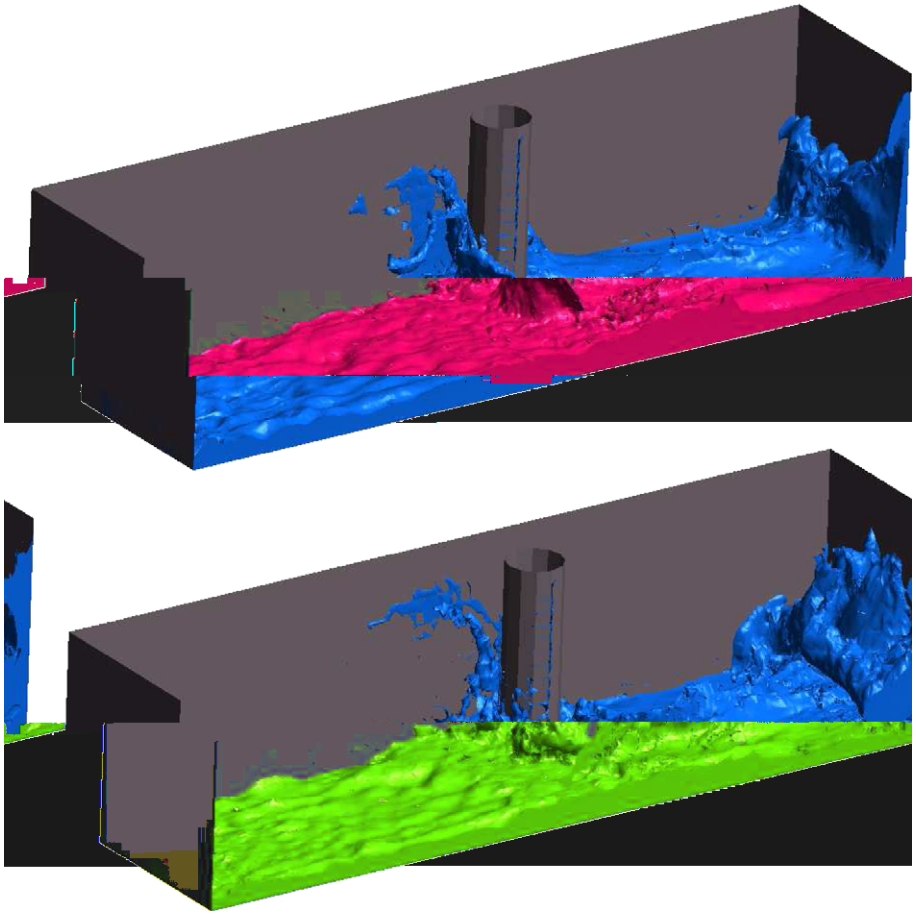


Fig. 16. Perspective view of the free surface for the dam break with a cylindrical obstacle. Free surface position at the selected times: $t = 2$ s and $t = 2.64$ s.

Table 2
Mass fluctuation of the method for the 3D dam break with cylindrical obstacle computed at time $t = 1$ s and $t = 2$ s on different computational meshes and with different polynomial orders p to approximate the level set

Mesh (nb nodes)	Order p	Mass loss ($t = 1$ s) (%)	Mass loss ($t = 2$ s) (%)	CPU (nb)	Comp. time (h)
12.500	1	7.4	14.3	2	1h30
12.500	2	3.4	6.2	2	3h10
12.500	3	2.1	3.1	4	15h10
30.820	2	1.16	0.86	8	13h10
74.016	2	0.62	0.11	24	6h45

The initial shape is initially a cosine function $z = 2.0 + 0.05\cos(2\pi x)$, the initial velocity field is zero, pressure field is hydrostatic, and the boundary conditions are no-slip on the bottom wall, slip on the side walls, and prescribed zero pressure at the upper wall. We use different orders of polynomials to describe the level set within the discontinuous Galerkin method and compare the maximum mass fluctuation for the different solutions obtained. Our results are compared with those obtained by various authors using different numerical methods. Table 3 compares the maximum mass fluctuation. The error is measured according to Eq. (8).

Fig. 17 shows the evolution of the interface at times 5, 8, 11 and 15 s for a Reynolds and Atwood number of $Re = 500$, $A = 0.5$.

Table 3
Maximum mass fluctuation of the method for the Rayleigh–Taylor instability computed on the same computational grid 32×265

Method	Max. mass fluctuation (%)
VOF [58]	0.01
Front tracking [59]	0.14
Level set FEM [25]	0.06
Level set *RK* (order3) *WENO* (order 7–) [60]	0.15
Presented method $p = 1$	0.17
Presented method $p = 2$	0.07
Presented method $p = 3$	0.002
Presented method $p = 4$	0.0004

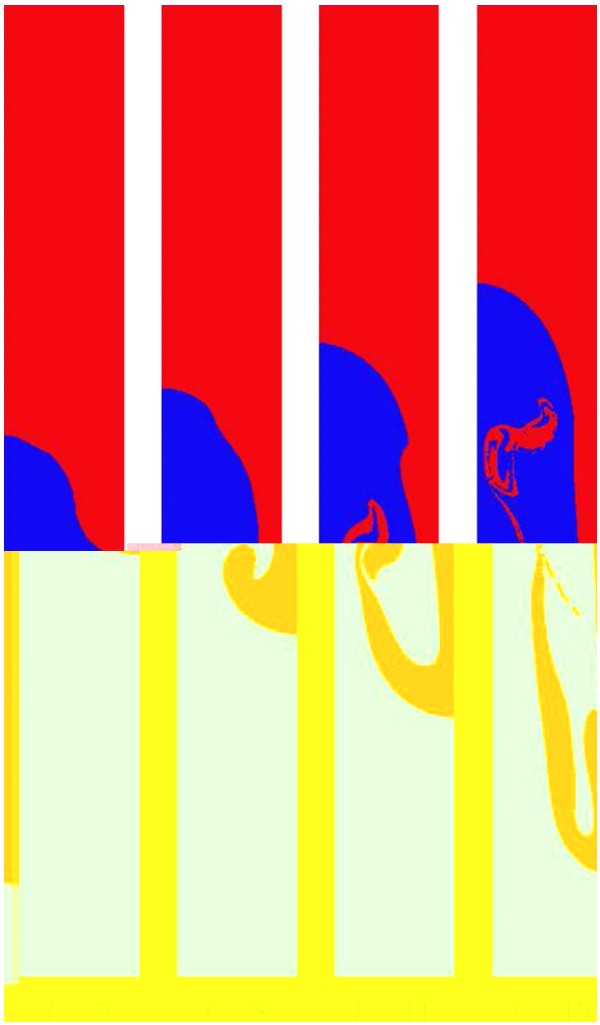


Fig. 17. Evolution of the fluid interface for $p = 3$. The Atwood number is 0.5 and the Reynolds number is 500.

Our numerical simulations compare well with those of Puckett et al. [58] and of Popinet and Zaleski [59] and He [61].

Next, we examine the results against the prediction by the linear stability theory. During the initial stages of the Rayleigh–Taylor instability, the perturbation amplitude is significantly smaller than the wavelength, the equations can then be linearized, and the perturbation of the fluid interface has an exponential growth [62]:

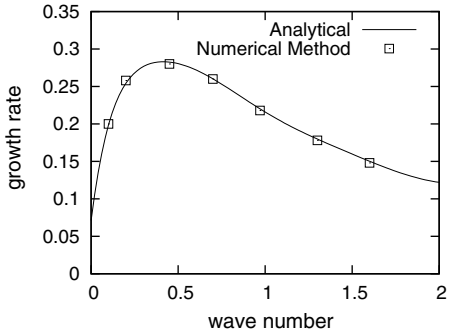


Fig. 18. Dependence of the linear non-dimensional growth rate of a disturbance (α) for different initial wave numbers κ. The Atwood number is $A = 0.5$.

$$h = h_0 e^{\hat{\alpha} t}, \tag{27}$$

where h is the amplitude at time t, h_0 is the initial amplitude and $\hat{\alpha}$ is the growth rate. Fig. 18 shows an analytical solution obtained by Chandrasekhar [62] for the dependence of the dimensionless growth rate $\alpha = \frac{\hat{\alpha}}{(g^2/\nu)^{1/3}}$ on the disturbance's wavenumber $\kappa = \frac{2\pi}{h_0(g/\nu^2)^{1/3}}$. Numerically, we can vary κ by changing the fluid kinematic viscosity ν in the simulations.

5. Conclusion

A unified approach for the numerical simulation of three-dimensional two-phase flows has been presented. The approach relies on an implicit stabilized finite element approximation for the Navier–Stokes equations and discontinuous Galerkin method for the level set method (DG-LSM).

Such a combination of those two numerical methods results in a simple and effective algorithm that allows to simulate diverse flow regimes (ranging from stokes flow to highly convective flows), presenting also large density and viscosity ratios (up to 1000).

Three advantages of the method are:

- *Simplicity and flexibility*: The stabilized method utilizes simple linear elements for the unknowns of velocity and pressure and a DG method of higher order for the level set unknown. The tetrahedral meshes can be structured or unstructured;
- *Accuracy*: The overall scheme is second order accurate in space and time. Besides we do not need to introduce any artificial parameter as the interface thickness. The interface can be localized precisely which enables us to compute accurately the integrals with a discontinuous density and/or viscosity. Furthermore, the method exhibits excellent conservation properties using high order polynomials for the level set (see Table 3).

For our future applications we will focus on spatially adaptive grids to achieve higher resolution of the interface while reducing memory and CPU time. The nested multilevel hierarchy of the tetrahedral meshes we have chosen to work with will allow us to use very simple refining and coarsening routines.

References

[1] E. Marchandise, J.-F. Remacle, N. Chevaugeon, A quadrature free discontinuous Galerkin method for the level set equation, Journal of Computational Physics 212 (2006) 338–357.
[2] J.-F. Remacle, N. Chevaugeon, E. Marchandise, C. Geuzaine, Efficient visualization of high order finite elements, International Journal for Numerical Methods in Engineering (2006).
[3] C.W. Hirt, A.A. Amsden, J.L. Cook, An arbitrary Lagrangian–Eulerian computing method for all flow speeds, Journal of Computational Physics 135 (2) (1997), in press.

[4] T. Hughes, W. Liu, T. Zimmermann, Lagrangian–Eulerian finite element formulation for incompressible viscous flows, Computer Methods in Applied Mechanics and Engineering 29 (1981) 239–349.
[5] J. Donea, Arbitrary Lagrangian–Eulerian finite element methods, Computational Methods for Transient Analysis 1 (1983) 473–516.
[6] W. Dettmer, P. Saksono, D. Peric, On a finite element formulation for incompressible Newtonian fluid flows on moving domains in the presence of surface tension, Computer Methods in Applied Mechanics and Engineering 19 (2003) 659–668.
[7] F. Harlow, J. Welch, Volume tracking methods for interfacial flow calculations, Physics of Fluids 8 (1965) 21–82.
[8] C. Hirt, B. Nichols, Volume of fluid method (VOF) for the dynamics of free boundaries, Journal of Computational Physics 39 (1981) 201–225.
[9] J. Pilliot, E. Puckett, Second order accurate volume-of-fluid algorithms for tracking material interfaces, Journal of Computational Physics 199 (2004) 465–502.
[10] S. Osher, J.A. Sethian, Fronts propagating with curvature dependent speed: algorithms based on Hamilton–Jacobi formulations, Journal of Computational Physics 79 (1988) 12–49.
[11] M. Sussman, E. Fatemi, An efficient interface preserving level set redistancing algorithm and its application to interfacial incompressible fluid flow, SIAM Journal on Scientific Computing 20 (1999) 1165–1191.
[12] M. Sussman, P. Smereka, S. Osher, A level set approach for computing solutions to incompressible two-phase flow, Journal of Computational Physics 114 (1994) 146–159.
[13] M. Sussman, E. Fatemi, P. Smereka, S. Osher, An improved level set method for incompressible two-fluid flows, Computers and Fluids 27 (1998) 663–680.
[14] J. Chessa, T. Belytschko, An extended finite element method for two-phase fluids, Journal of Applied Mechanics (Transactions of the ASME) 70 (2003) 10–17.
[15] J. Chessa, T. Belytschko, An enriched finite element method and level sets for axisymmetric two-phase flow with surface tension, International Journal for Numerical Methods in Engineering 58 (2003) 2041–2064.
[16] C. Norman, M. Miksis, Non-linear dynamics of thin films and fluid interfaces, Physica D: Nonlinear Phenomena 209 (2005) 191–204.
[17] S. Nagrath, K. Jansen, R. Lahey, Computation of incompressible bubble dynamics with a stabilized finite element level set method, Computer Methods in Applied Mechanics and Engineering 194 (42–44) (2005) 4565–4587.
[18] S. Osher, C.-W. Shu, High order essentially non-oscillatory schemes for Hamilton–Jacobi equations, SIAM Journal on Numerical Analysis 728 (1991) 902–921.
[19] C. Hu, C.-W. Shu, Weighted ENO schemes for Hamilton–Jacobi equations, SIAM Journal on Scientific Computing 21 (6) (1999) 2126–2143.
[20] D. Enright, R. Fedkiw, J. Ferziger, I. Mitchell, A hybrid particle level set method for improved interface capturing, Journal of Computational Physics 183 (2002) 83–116.
[21] S. Nagrath, K. Jansen, R. Lahey, Computation of incompressible bubble dynamics with a stabilized finite element level set method, Computer Methods in Applied Mechanics and Engineering 194 (2005) 4565–4587.
[22] A.-K. Tornberg, B. Engquist, A finite element based level set method for multiphase flow applications, Computing Visualization in Science 3 (2000) 93–101.
[23] S. van der Pijl, A. Segal, C. Vuik, P. Wesseling, A mass-conserving levelset method for modeling of multiphase flows, International Journal for Numerical Methods in Fluids 47 (2004) 339–361.
[24] J. Ferziger, Interfacial transfer in Tryggvason's method, International Journal for Numerical Methods in Fluids 41 (5) (2003) 551–560.
[25] A. Smoliansky, Numerical modeling of two-fluid interfacial flows, Ph.D. Thesis, University of Jyvaskyla, Finland, 2001.
[26] A.-K. Tornberg, Interface tracking methods with applications to multiphase flows, Ph.D. Thesis, Royal Institute of Technology (KTH), Finland, 2000.
[27] A. Chorin, Numerical solution of the Navier–Stokes equations, Mathematical Computation 22 (1968) 745–762.
[28] Y. Kallinderis, K. Nakajima, Finite element method for incompressible viscous flow with adaptative grids, AIAA Journal 8 (32) (1994), in press.
[29] Y. Kallinderis, W. Schulz, Unsteady flow structure interaction for incompressible flows using deformable grids, Journal of Computational Physics 143 (2) (1998) 569–597.
[30] T.J.R. Hughes, L.P. Franca, A new finite element formulation for computational fluid dynamics: V. circumventing the Babuska–Brezzi condition: A stable Petrov–Galerkin formulation of the stokes problem accommodating equal-order interpolations, Computer Methods in Applied Mechanics and Engineering 59 (1986) 85–99.
[31] T. De Mulder, Stabilized finite elements methods for turbulent incompressible single-phase and dispersed two-phase flows, Ph.D. Thesis, von Karman Institute For Fluid Dynamics, Belgium, 1997.
[32] A. Chorin, A numerical method for solving incompressible viscous flow problems, Journal of Computational Physics 135 (2) (1997) 118–125.
[33] S. Rogers, D. Kwak, C. Kiris, Steady and unsteady solutions of the incompressible Navier–Stokes equations, AIAA Journal 29 (1991) 306–610.
[34] T. Hughes, L. Franca, G. Hulbert, A new finite element formulation for fluid dynamics: VIII. The Galerkin/least-squares method for advective–diffusive equations, Computer Methods in Applied Mechanics and Engineering 73 (1989) 173–189.
[35] L. Franca, S. Frey, Stabilized finite element methods: II. The incompressible Navier Stokes equations, Computer Methods in Applied Mechanics and Engineering 59 (1992) 209–233.
[36] O. Ladyshenskaya, The Mathematical Theory of Viscous Incompressible Flow, Gordon and Breach, New York, 1969.
[37] I. Babuska, The finite element method with Lagrangian multipliers, Numerical Mathematics 179 (1971), in press.
[38] F. Brezzi, M. Fortin, Mixed and Hybrid Finite Element Methods, Springer, New York, 1991.

[39] J.-F. Remacle, J. Flaherty, M. Shephard, An adaptive discontinuous Galerkin technique with an orthogonal basis applied to compressible flow problems, SIAM Review 45 (2003) 53–72.
[40] J.-F. Remacle, M.S. Shephard, An algorithm oriented mesh database, International Journal for Numerical Methods in Engineering 58 (2) (2003) 349–374.
[41] I. Babuska, The finite element method with Lagrangian multipliers, Numerische Mathematik 20 (1971) 179–192.
[42] F. Brezzi, On the existence uniqueness and approximation of saddle-point problems arising from Lagrange multipliers, R.A.I.R.O., Série Rouge Analyse Numérique 8 (1974) 129–151.
[43] T. Tezduyar, Y. Osawa, Finite element stabilization parameters computed from element matrices and vectors, Computer Methods in Applied Mechanics and Engineering 190 (2000) 411–430.
[44] F. Shakib, T. Hughes, Z. Johan, A new finite element formulation for computational fluid dynamics: the compressible Euler and Navier–Stokes equations, Computer Methods in Applied Mechanics and Engineering 89 (1991) 141–219.
[45] T.J. Barth, Aspects of unstructured grids and finite-volume solvers for the Euler and Navier–Stokes equations, Special course on unstructured grid methods for advection dominated flows, AGARD R-787, 1992 (Chapter 4).
[46] V. Selmin, L. Formaggia, Unified construction of finite element and finite volume discretizations for compressible flows, International Journal for Numerical Methods in Fluids 39 (1996) 1–32.
[47] P. Geuzaine, An implicit upwind finite volume method for compressible turbulent flows on unstructured meshes, Ph.D. Thesis, Université de Liège, Belgium, 1999.
[48] M. Mount, N.S. Arya, R. Netanyahu, Silverman, An optimal algorithm for approximate nearest neighbor searching, Journal of the ACM 45 (1998), in press.
[49] ANN library. Available from: URL <http://www.cs.umd.edu/~mount/ANN/>.
[50] P. Geuzaine, Newton–Krylov strategy for compressible turbulent flows on unstructured meshes, AIAA Journal 39 (2001) 528–531.
[51] X.C. Cai, C. Farhat, M. Sarkis, A minimum overlap restricted additive Schwarz preconditioner and applications in 3D flow simulations, Contemporary Mathematics 218 (1998) 478–484.
[52] A. Prosperetti, Motion of two superimposed viscous fluids, Physics of Fluids 24 (1981) 1217–1223.
[53] G. Wu, R. Eatock Taylor, D. Greaves, The effect of viscosity on the transient free-surface waves in a two-dimensional tank, Journal of Engineering Mathematics 40 (2001) 77–90.
[54] J. Martin, W. Moyce, An experimental study of the collapse of liquid columns on a rigid horizontal plane, Philosophical Transactions A 244 (1952) 312–324.
[55] A. Ritter, Die fortpflanzung der wasserwellen., Z. Ver. deut. Ing. 36 (1982), in press.
[56] M. Quecedo, M. Pastor, M. Herreros, Comparison of two mathematical models for solving the dam break problem using the FEM method, Computer Methods in Applied Mechanics and Engineering (in press).
[57] T. Shigematsu, P. Liu, K. Oda, Numerical modeling of the initial stages of dam-break waves, Journal of Hydraulic Research 42 (2) (2004) 183–195.
[58] E. Puckett, A. Almgren, J. Bell, D. Marcus, W. Rider, A high-order projection method for tracking fluid interfaces in variable density incompressible flows, Journal of Computational Physics 130 (1997) 269–282.
[59] S. Popinet, S. Zaleski, A front-tracking algorithm for accurate representation of surface tension, International Journal for Numerical Methods in Fluids 30 (1999) 775–793.
[60] R.R. Nourgaliev, T.N. Dinh, T.G. Theofanous, A pseudocompressibility method for the numerical simulation of incompressible multifluid flows, International Journal of Multiphase Flow 30 (2004) 6001–6937.
[61] M. Mount, N.S. Arya, R. Netanyahu, Silverman, A lattice Boltzmann scheme for incompressible multiphase flow and its application in simulation of Rayleigh, Journal of Computational Physics 152 (2) (1999) 642–663.
[62] S. Chandrasekhar, Hydrodynamic and Hydromagnetic Stability, Clarendon Press, Oxford, 1961.