An unstructured conservative level-set algorithm coupled with dynamic mesh adaptation for the computation of liquid-gas flows

Romain Janodet, Geoffroy Vaudor, Ghislain Lartigue, Pierre Bénard, Vincent Moureau, Renaud Mercier

An unstructured conservative level-set algorithm coupled with dynamic mesh adaptation for the computation of liquid-gas flows

Romain Janodet*[1], Geoffroy Vaudor[1], Ghislain Lartigue[1], Pierre Bénard[1], Vincent Moureau[1], Renaud Mercier[2]
[1]CORIA, CNRS UMR6614, INSA and University of Rouen, F-76000 Rouen, France
[2]Safran Tech, Modelling & Simulation, Rue des Jeunes Bois, Châteaufort, 78114 Magny-Les-Hameaux, France
*Corresponding author: romain.janodet@coria.fr

Abstract
Accurate and efficient simulations of 3D liquid-gas flows are of first importance in many industrial applications, such as fuel injection in aeronautical combustion chambers. In this context, it is mandatory to handle complex geometries. The use of unstructured grids for two-phase flow modeling fulfills this requirement and paves the way to isotropic adaptive mesh refinement. This work presents a narrow-band conservative level-set algorithm implemented in the YALES2 incompressible flow solver, which is combined to dynamic mesh adaptation. This strategy enables resolving the small physical scales at the liquid-gas interface at a moderate cost. It is applied to predicting the outcome of a droplet collision with reflexive separation.
In the accurate conservative level set framework, the interface is represented using a hyperbolic tangent profile, which is advected by the fluid, and then reshaped using a reinitialization equation. The classical signed-distance function is reconstructed at nodes in the narrow band around the interface using a geometric projection/marker method (GPMM), to calculate the smallest distance to the interface. The interface normal and curvature are computed using this signed-distance function. Within a mesh cell, the interface is approximated by a segment (2D) or one or several triangles (3D). The distance at the nodes is simply obtained by projection to the closest surface elements. If a node is connected to n elements containing interface fragments, it has a n-marker list (a marker contains the coordinates of the crossing points and the distance). To speed-up the algorithm, the markers stored at each node are sorted based on their distance. Markers are propagated from one band to another: each node compares its markers to its neighbors' and keeps the closest only.
The GPMM approach for the reconstruction of the level-set signed-distance function used in conjunction with the reinitialization of Chiodi et al. (2017) leads to significant improvement in the interface quality and overall accuracy compared to the reinitialization of Desjardins et al. (2008) in the calculations performed on unstructured grids. Since the accuracy of the interface normal and curvature directly depends on the signed-distance function reconstruction, less spurious currents occur on the implicit surface. The improved level-set algorithm leads to accurate predictions of the outcome of a droplet collision with reflexive separation, and is validated against the experimental results of Ashgriz et al. (1990).

Keywords
Conservative Level Set; Adaptive Mesh Refinement; Unstructured grids; Droplet collision

Introduction

Two-phase flows are ubiquitous in nature and in industrial systems. The understanding of the various phenomena occurring in liquid-gas flows is crucial for aeronautical combustors, in which a fuel is injected in liquid form, goes under an atomization process, evaporation, mixing with air and eventually combustion. Understanding the atomization process and the resulting droplet distribution is of first importance for aircraft engine performance and operability. The prediction of the atomization process is complex, due to many non-linear phenomena such as interface break-up, droplet convection, or droplet collision. Atomization also involves a wide range of time and space scales, which leads to important calculation costs. Thus, the use of dynamic mesh adaptation for unstructured meshes is particularly helpful for simulating industrial liquid-gas flow problems, as it allows implicit interface dynamics calculation in complex geometries at a reasonable cost [1]. To capture the interface, the conservative level set method is used, which accurately predicts the interface dynamics while conserving liquid mass [2].
This article presents a method to compute the signed-distance function on unstructured grids, and an implementation of the reinitialization of [3], adapted to unstructured meshes. Classic test cases are run to check the overall accuracy and robustness of the method, and a droplet collision case is simulated to validate the global algorithm with a front merging scenario against the experimental results of [4].

Accurate Conservative Level Set framework (ACLS)

In the ACLS framework [5], the interface Γ is represented using a hyperbolic tangent profile:

$$\psi(\mathbf{x},t) = \frac{1}{2}\left(\tanh\left(\frac{\phi(\mathbf{x},t)}{2\varepsilon}\right)+1\right) \tag{1}$$

where 2ϵ is the thickness of the profile, and $\phi = \pm|\mathbf{x} - \mathbf{x}_\Gamma|$ is the signed-distance function.
The interface is located at the iso-level 0.5:

$$\Gamma = \{\mathbf{x} \in \mathrm{IR}^3 / \psi(\mathbf{x}, t) = 0.5\} \tag{2}$$

Assuming the flow velocity **u** is divergence free, the scalar ψ is advected by the fluid:

$$\frac{\partial \psi}{\partial t} + \boldsymbol{\nabla} \cdot (\psi \mathbf{u}) = 0 \tag{3}$$

and then reshaped using the reinitialization equation of [5]:

$$\frac{\partial \psi}{\partial \tau} = \boldsymbol{\nabla} \cdot \Big(\underbrace{\varepsilon (\boldsymbol{\nabla}\psi \cdot \mathbf{n})\mathbf{n}}_{\text{Diffusion}} - \underbrace{\psi(1-\psi)\mathbf{n}}_{\text{Resharpening}} \Big) \tag{4}$$

where τ is a pseudo-time, and $\mathbf{n} = \dfrac{\boldsymbol{\nabla}\phi}{|\boldsymbol{\nabla}\phi|}$ is the interface normal.
A schematic view of the ACLS framework is given in Fig. 1.

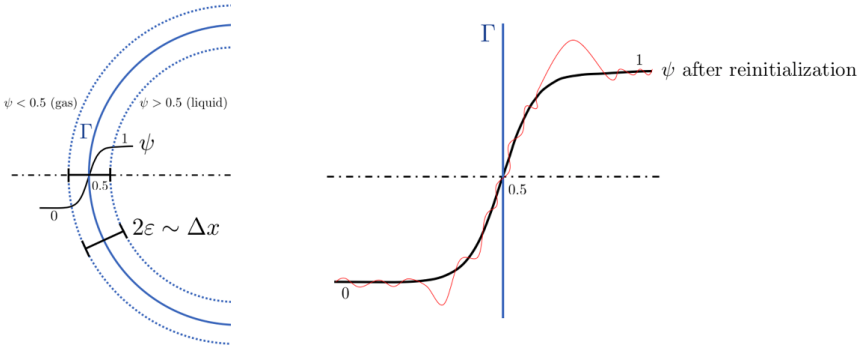


Figure 1. Interface representation (left) and reinitialization (right) in the ACLS framework

A new form of the reinitialization equation was recently proposed by Chiodi et al. in [3]:

$$\frac{\partial \psi}{\partial \tau} = \boldsymbol{\nabla} \cdot \left(\frac{1}{4\cosh^2\left(\frac{\phi_{\mathrm{map}}}{2\epsilon}\right)} \left(|\boldsymbol{\nabla}\phi_{\mathrm{map}} \cdot \mathbf{n}| - 1\right) \mathbf{n} \right) \tag{5}$$

where $\phi_{\mathrm{map}} = \epsilon \ln\left(\psi/(1-\psi)\right)$ is an analytical signed-distance function, mapped for $\psi \in]0;1[$.
The signed-distance function ϕ is reconstructed at nodes in the narrow band around the interface using a geometric projection/marker method (GPMM), to estimate the smallest distance to the interface.
The interface mean curvature κ is computed directly from ϕ using Goldman's formula [6]:

$$\kappa = \frac{\mathrm{Tr}\left(\boldsymbol{\mathcal{H}}(\phi)\right) - \frac{\boldsymbol{\nabla}\phi^T}{|\boldsymbol{\nabla}\phi|} \cdot \boldsymbol{\mathcal{H}}\left(\phi\right) \cdot \frac{\boldsymbol{\nabla}\phi}{|\boldsymbol{\nabla}\phi|}}{(n_{\mathrm{dim}} - 1)|\boldsymbol{\nabla}\phi|} \tag{6}$$

where $\boldsymbol{\mathcal{H}}$ is the hessian matrix of the signed-distance function.
The main motivation for using the ACLS method is to better conserve the liquid volume. The approximate and exact enclosed volumes of liquid are respectively:

$$\widetilde{V}_l = \int_V \psi \, dV \quad \text{and} \quad V_l = \int_V H(\psi - 0.5) \, dV \tag{7}$$

where H is the Heaviside function (0 in gas, 1 in liquid).
In the ACLS method, the approximate volume $\widetilde{V}_l$ is conserved, but $\widetilde{V}_l \neq V_l$ due to the interface curvature. Another reason for losing mass on irregular meshes is the variation of the mesh resolution from fine to coarse, which leads to the numerical diffusion of ψ, which may imply a loss of the $\psi = 0.5$ iso-surface. In this case, the interface is not detected anymore and gas can be set instead of liquid. Therefore, special attention is needed when coupling with dynamic mesh adaptation, so that Γ never encounters cell-size gradients.
In this framework, surface tension σ is embedded in the pressure jump $[p]_\Gamma$:

$$[p]_\Gamma = p_{l,\Gamma} - p_{g,\Gamma} = \sigma\kappa + 2[\mu]_\Gamma \mathbf{n}^{\mathrm{T}} \cdot \boldsymbol{\nabla}\mathbf{u} \cdot \mathbf{n} \tag{8}$$

and is treated with the Ghost-Fluid Method (GFM) [7], which is a sharp-interface method dealing explictly with discontinuities at Γ. In Eq. 8, μ is the dynamic viscosity, $p_{l,\Gamma}$ is the liquid pressure at the interface, and $p_{g,\Gamma}$ the gas pressure at the interface.

Geometric projection multiple marker method (GPMM) for signed-distance function reconstruction

In order to compute interface normal and mean curvature, it is needed to reconstruct the signed-distance function in the narrow band around the front. The method presented here is fully geometric and uses a fast marching method towards the farthest nodes of the band. In the present work, the band width is generally between 8 to 12 cells on each side of the interface.

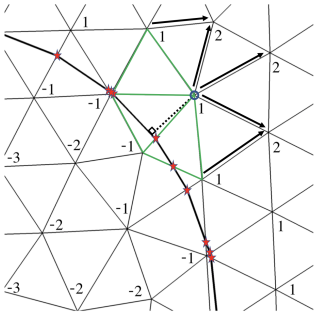


Figure 2. GPMM method on a triangular grid

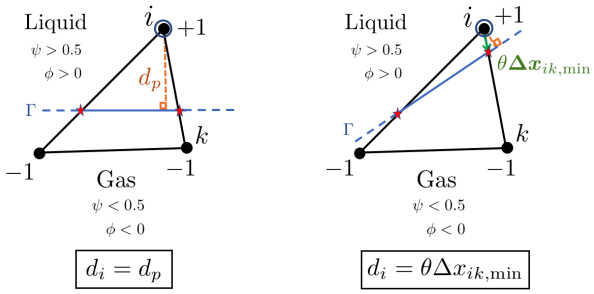


Figure 3. Projection algorithm illustrated in the first band level: the projected point is either inside the element (left, the projected distance d_p is retained), or outside the element (right, the minimum distance in a pair direction $\theta\Delta x_{ik,\min}$ is retained)

An interpolation is first performed to compute the interface position $\boldsymbol{x}_\Gamma$ on each crossing pair of nodes (red stars in Fig. 2). Given the conservative level-set functions ψ_i and ψ_k at the nodes i and k belonging to a crossing pair ik, one can invert these functions to build two distance functions to the interface $\phi_{inv,i}$ and $\phi_{inv,k}$:

$$\begin{cases} \phi_{inv,i} = 2\varepsilon_i \text{atanh}\,(2\psi_i - 1) \\ \phi_{inv,k} = 2\varepsilon_k \text{atanh}\,(2\psi_k - 1) \end{cases} \tag{9}$$

The level-set position θ on a pair crossing the interface is then calculated as:

$$\theta = \frac{\phi_{inv,i}}{\phi_{inv,i} - \phi_{inv,k}} \tag{10}$$

which means that $\theta = 0$ corresponds to Γ is at node i. The interface position is then calculated as:

$$\boldsymbol{x}_\Gamma = \boldsymbol{x}_i + \theta(\boldsymbol{x}_k - \boldsymbol{x}_i) = \boldsymbol{x}_i + \theta\boldsymbol{\Delta x}_{ik} \tag{11}$$

Within a cell, the interface is approximated by a segment (2D) or a set of triangles (3D). These geometric entities are defined from the intersections of the interface with the edges of the mesh. While the 2D case is trivial, the 3D case is more complex. Indeed, a tetrahedron may have 3 or 4 edges crossed by the interface. When three edges are crossed, the interface inside the cell can be described by a single triangle. When four edges are crossed, the interface in the cell is a quadrilateral shape. This latter is triangulated in the proposed approach into four triangles to ease the storing of the surface elements.

At a given node away from the interface, the distance to the interface is simply calculated as the closest distance to an interface segment or triangle. To this aim, a geometric projection has to be performed as illustrated in 2D in Fig. 3. This projection involves a lot of geometric tests as the closest distance to a segment can be the orthogonal projection of the node onto the segment or any of the nodes of the segment. This is even more complex in 3D for a triangle, where the closest distance may also be to an edge of the interface triangle.

In the proposed GPMM, each node in the band around the interface has a list of interface segments or triangles that are supposedly the closest to the node. To build this list, a fast marching method is employed. The node with the blue circle in Fig. 2 is connected to three elements containing interface fragments (green triangles in Fig. 2), thus it starts with a three-marker list. Each marker contains the coordinates of the segment or triangle points and the distance to the node. The three markers are sorted based on their distance. This node then compares its marker list with its neighbors of same or lower level. Its list will be enriched until it reaches its maximum length, which is user-defined. When the list is full, only the closest markers are retained. Based on numerical experiments, a maximum list length of 3 in 2D and 10 in 3D is sufficient to have an accurate distance function as it will be demonstrated hereafter.

Numerical discretization of the new reinitialization equation

YALES2 is a pair-based finite-volume low-Mach flow solver, which uses a vertex-centered variable arrangement on unstructured meshes. The grid generation is based on Delaunay triangulation. 4th-order finite-volume schemes are written on the dual mesh constructed from edge and element centers. Fig. 4 shows the control volume used for spatial discretization.

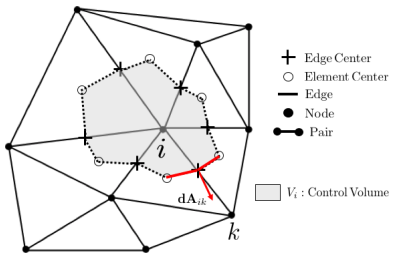


Figure 4. Control volume used in YALES2 for the finite-volume spatial discretization (adapted from [8])

After integrating Eq. 5 over a control volume and applying the divergence theorem, the reinitialization of [3] is discretized as follow:

$$\frac{\psi_i^{n^*+1} - \psi_i^{n^*}}{\Delta\tau} = \frac{1}{V_i} \sum_{ik \text{ pairs of i}} \left(\frac{1}{4\cosh^2\left(\frac{\phi_{\text{map},ik}}{2\varepsilon_{ik}}\right)} \left(\boldsymbol{\nabla}\phi_{\text{map}}|_{ik} \cdot \mathbf{n}_{ik} - \mathbf{n}_{ik} \cdot \mathbf{n}_{ik}\right) \right) \mathbf{n}_{ik} \cdot \mathbf{dA}_{ik} \tag{12}$$

where $\mathbf{n}_{ik} = \dfrac{\boldsymbol{\nabla}\phi|_i^{4\text{th}} + \boldsymbol{\nabla}\phi|_k^{4\text{th}}}{2}$ is the pair-based interface normal (interpolation of fourth-order nodal gradients), $\phi_{\text{map},ik} = \dfrac{\phi_{\text{map},i} + \phi_{\text{map},k}}{2}$ and $\varepsilon_{ik} = \dfrac{\varepsilon_i + \varepsilon_k}{2}$ are linearly interpolated interface distance and thickness, respectively, and $\mathbf{dA}_{ik}$ is the surface vector of the small piece of control volume surface associated to the node pair ik. The distance gradient $\boldsymbol{\nabla}\phi_{\text{map}}|_{ik}$ is computed with the standard 4th-order operator of the flow solver.
Special attention has to be paid in regions of the narrow band where ψ is close to 0 and 1. In these regions, ϕ_{map} is difficult to estimate and two modifications of the reinitialization equations are done: i) ϕ_{map} is computed using $\psi = 10^{-15}$ and $\psi = 10^{+15}$ in the case of undershoots and overshoots, respectively, ii) the resharpening term is removed so that the reinitialization becomes purely diffusive. Once the overshoots or undershoots of ψ disappear, the reinitialization turns back to its original form. The pseudo timestep of each reinitialization step is simply determined by a Fourier condition on the diffusive part of the reinitialization. This stability condition leads to:

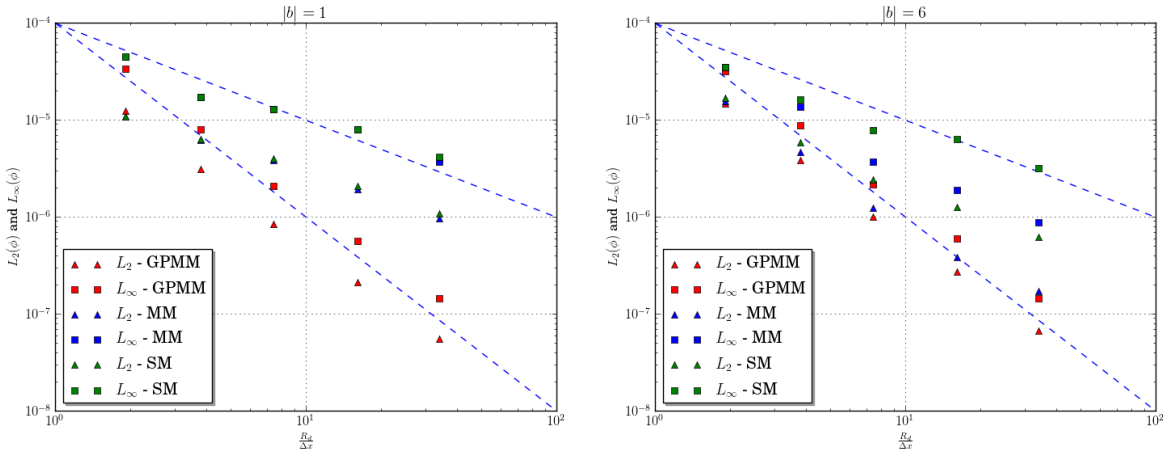
$$\Delta\tau = \min_{\text{pairs } ik} \left(\frac{\text{Fo}.\Delta x_{ik}^2}{\varepsilon_{ik}} \right) \tag{13}$$

where Fo is the Fourier number set to 0.5.

2D validation: results and discussion
Two-dimensional test cases are run on homogeneous triangular grids both with the GPMM signed-distance and the reinitialization of Eq. 5). First, a 2D static droplet test case (water in air, droplet of radius $R_d = 1\text{mm}$) is used to assess the accuracy of the GPMM for the signed-distance reconstruction. Errors are calculated as follows:

$$L_2(\phi) = \sqrt{\frac{1}{N_{i\in|b|}} \sum_{i=1}^{N_{i\in|b|}} \left(\phi_{\text{th},i} - \phi_{\text{GPMM},i}\right)^2} \quad \text{and} \quad L_\infty(\phi) = \max_{i\in|b|} \left|\phi_{\text{th},i} - \phi_{\text{GPMM},i}\right| \tag{14}$$

where $|b|$ is the absolute band level, $\phi_{\text{th},i}$ is the theoretical distance at node i for a circular interface, and $N_{i\in|b|}$ is the total number of nodes in the corresponding band level. The calculations of L_2 and L_∞ are performed for nodes belonging to edges crossed by the interface ($|b| = 1$) and in the sixth band level ($|b| = 6$, far from the interface). The results are validated against two other marker algorithms for the distance calculation. The first, denoted Simple Marker (SM) limits the marker list to a single geometrical element per node (one segment or one triangle) and no projection is performed, i.e. that the distance is computed simply the distance to the closest node of the geometrical element. The second, denoted Multiple Marker (MM) is the same as the SM method except that it increases the size of the marker list to the same length as the GPMM. These two methods are expected to give larger errors than the GPMM. Results are shown in Figs. 5a and 5b.
It can be seen in Fig. 5a that in the first band level, i.e. close to the interface, the results are logically identical whether SM or MM is used at each node. Both SM and MM methods exhibit first-order convergence with mesh resolution (L_2 and L_∞), whereas the GPMM method converges with second-order accuracy (L_2 and L_∞), with significant decrease in error values at a given resolution compared to the two other methods.
Fig. 5b shows the results for the sixth band level, i.e. far from the interface. A notable difference is seen between SM and MM methods as resolution of the droplet increases. As several markers are propagated (3 in 2D) in the MM method, more data are available to compute the smallest distance at the nodes far from the interface than with the SM method. Both L_2 and L_∞ for the GPMM method exhibit second-order convergence for $|b| = 6$.
Zalesak's notched disk [9] is then used to assess if the new algorithms are robust with sharp edges and corners. The circle is 0.15m in radius R, with a notch of width 0.05m and height 0.25m. It is centered initially at $(0.5\text{m}, 0.75\text{m})$ in a $[-0.5\text{m}; 0.5\text{m}]^2$ domain. The parameters of the run are given in Table 1.



(a) First band level

(b) Sixth band level

Figure 5. L_2 and L_∞ of the reconstructed signed-distance function with different methods (Geometric Projection Multiple Markers GPMM, Multiple Markers MM, Simple Marker SM)

Table 1. Simulation parameters used in the two-dimensional test cases

	CFL	Number of elements	Number of reinit. steps / iter.	Physical duration
Zalesak's disk	0.48	230912	5	62.8s
Serpentine	0.64 (initial)	230912	3	8s

Fig. 6 shows the ψ field. It is seen that the GPMM method in combination with the new reinitialization are robust for treating corners: the interface deformations remain small with five iterations of reinitialization per time step. The final interface shape of the disk is then computed at various mesh resolutions. The results are depicted in Fig. 7: excellent convergence properties are demonstrated for this test case.

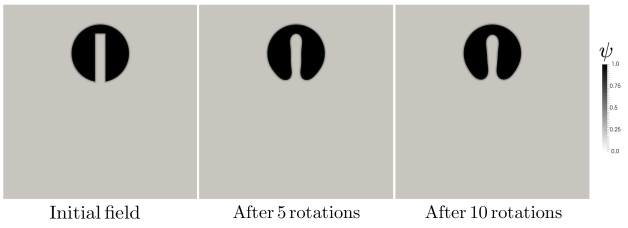


Figure 6. Bidimensional notched disk test, resolved with 50 points within the disk radius

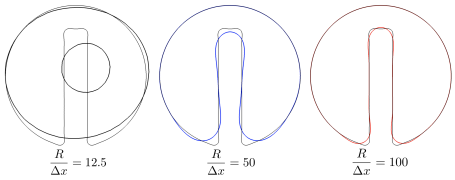


Figure 7. Interface shape after 10 rotations for different triangular grid resolutions, superimposed on a simulation performed with 400 mesh points within the disk radius

The 2D circle deformation (serpentine) test case is then simulated [10] to check the accuracy of the methods when the structures are close to the resolution limit. An initial circle of radius $R = 0.15\text{m}$ with a theoretical area of $A_0 = 0.07\text{m}^2$ is placed in a square domain of dimensions $[-0.5\text{m}; 0.5\text{m}]^2$, in a rotating deformation flow field:

$$\mathbf{u}(x,y,t) = \begin{pmatrix} u(x,y,t) \\ v(x,y,t) \end{pmatrix} = \begin{pmatrix} -\sin^2(\pi x)\sin(2\pi y)\cos\left(\frac{\pi t}{T}\right) \\ \sin^2(\pi y)\sin(2\pi x)\cos\left(\frac{\pi t}{T}\right) \end{pmatrix} \tag{15}$$

The serpentine is reversed at $t = T/2$ and is supposed to go back to its original shape at $t = T$. The parameters of the run are given in Table 1. Results are seen in Figs. 8 (ψ field) and 10 (temporal evolution of approximated liquid volume $\widetilde{V}_l$). Small perturbations of the serpentine are seen when the thickness of the tail is comparable to the cell size. However, these errors remain limited and do not lead to significant errors in the final shape. The distance and reinitialization algorithms used in combination with the central transport scheme [11] lead to accurate liquid volume conservation. The interface shape of the serpentine is then computed at $t = 4\text{s}$ and $t = 8\text{s}$, at various mesh resolutions. As shown in Fig. 9, excellent convergence properties are demonstrated here: the theoretical trajectory is very well reproduced using 100 points within the circle radius.

Finally, a spurious current analysis is performed on the 2D static droplet case. A static droplet of diameter $D_d = 2\text{mm}$, resolved with 33 mesh points within its diameter, is initialized in a square domain of dimensions $[-5\text{mm}, 5\text{mm}]^2$ on a homogeneous unstructured grid of 230912 elements. Results are shown in Figs. 11a and 11b. The maximum

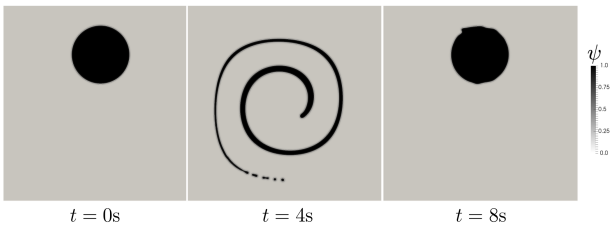


Figure 8. Bidimensional circle deformation test, resolved with 50 points within the circle radius

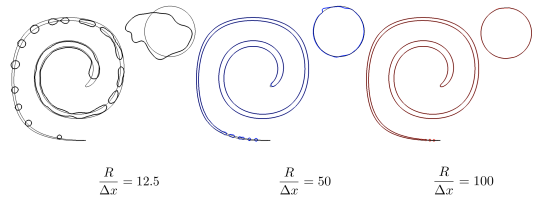


Figure 9. Interface shape at $t = 4\text{s}$ and $t = 8\text{s}$ for different triangular grid resolutions, superimposed on a simulation performed with 400 mesh points within the circle radius

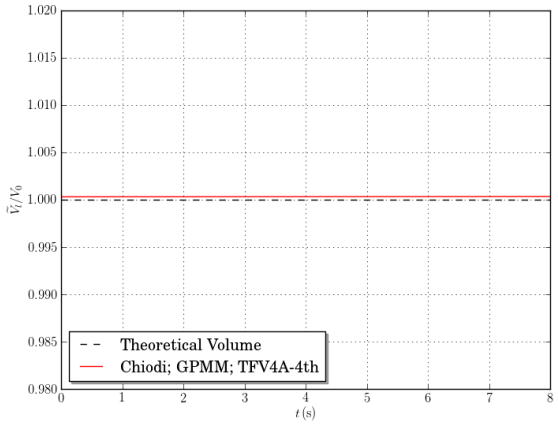
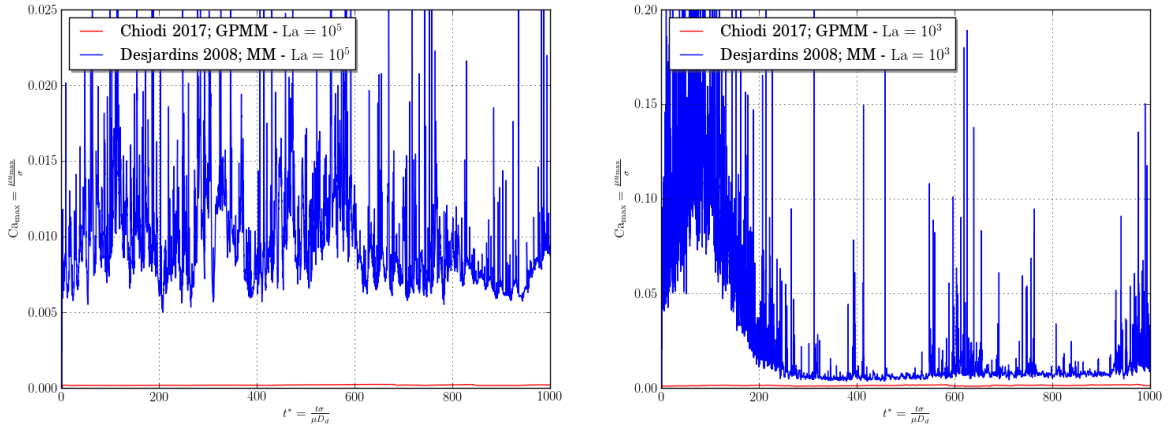


Figure 10. Temporal evolution of $\widetilde{V}_l = \int_V \psi \, dV$ with the new methods; V_0 is the theoretical volume



(a) $\text{La} = 10^5$ (b) $\text{La} = 10^3$

Figure 11. Temporal evolution of the parasitic currents, represented by $\text{Ca}_{\max}$

Capillary number $\text{Ca}_{\max} = \mu u_{\max}/\sigma$ is plotted for different Laplace numbers $\text{La} = \rho\sigma D_d/\mu^2$ varied by changing simultaneously the liquid and gas densities, as a function of a non-dimensional time $t^* = t\sigma/\mu D_d$. Five iterations of reinitialization are performed at each time step. The physical parameters used in the simulations are presented in Table 2. Results over $t^* = 1000$ with the GPMM method and the new reinitialization (Eq. 5) are compared with the MM method in combination with the reinitialization of [5]. A significant decrease in spurious currents is seen, thus less deformation occur on the implicit surface. It confirms that the new reinitialization equation is better in capillary-driven cases, even for unstructured grids. Moreover, steady state is also reached very early with the new algorithms, compared to the former ones which don't reach steady state at $t^* = 1000$: steady state is longer to reach as La increases, which leads to accumulation of errors with the old reinitialization equation. In the case of $\text{La} = 10^5$, the limiting time step for stability is the capillary time step, calculated from the dimensionless number SN

(surface tension number) set to 0.5 in all liquid-gas simulations of this article:

$$\mathrm{SN} = \sqrt{\frac{\Delta t^2 \sigma \kappa}{\min(\rho_l, \rho_g)\Delta x^2}} \tag{16}$$

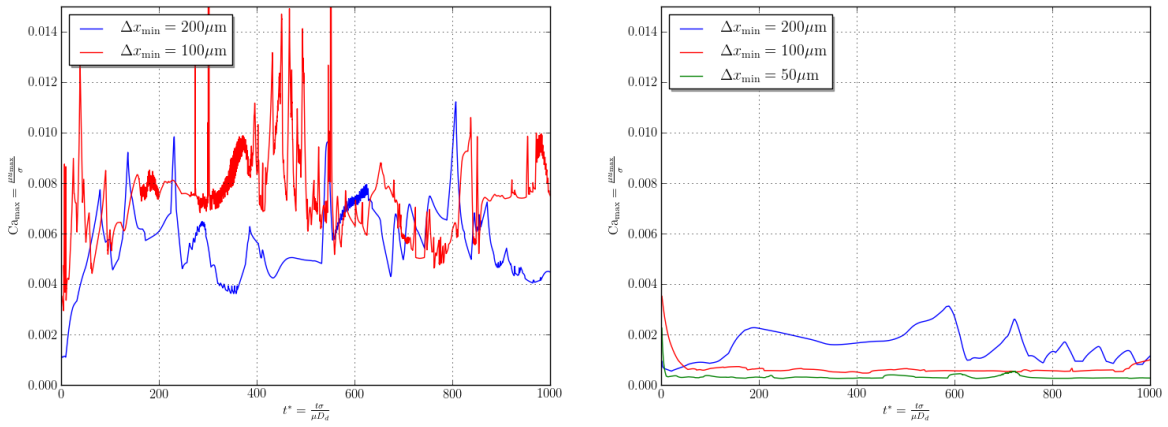
whereas in the case $\mathrm{La} = 10^3$, the viscous time step is limiting for stability.

Table 2. Simulation parameters used for the study of 2D spurious currents

	Δt_{ini} s	ρ_l kg.m^{-3}	ρ_g kg.m^{-3}	μ_l kg.m^{-1}.s^{-1}	μ_g kg.m^{-1}.s^{-1}	σ N.m^{-1}
$\mathrm{La} = 10^5$	$1.8.10^{-6}$	998.2	1.204	0.001137	$1.8265.10^{-5}$	$64.76.10^{-3}$
$\mathrm{La} = 10^3$	$5.5.10^{-8}$	9.982	0.01204	0.001137	$1.8265.10^{-5}$	$64.76.10^{-3}$

3D validation: results and discussion

A mesh convergence study of 3D spurious currents is performed for the MM/Desjardins 2008 and GPMM/Chiodi 2017 algorithms. The mesh is refined around the interface with adaptive mesh refinement. The droplet is of diameter $D_d = 2\mathrm{mm}$, and the chosen resolutions are $200\mu\mathrm{m}$, $100\mu\mathrm{m}$, and $50\mu\mathrm{m}$ at the interface. The Laplace number is $\mathrm{La} = 111188$, which is representative of a static water droplet in air. The surface tension number is set to $\mathrm{SN} = 0.5$ in both runs, and 5 iterations of reinitialization are performed per time step. Results are shown in Fig. 12a and 12b. A good mesh convergence and reduced errors are obtained with the GPMM method used in combination with the reinitialization of [3], whereas the MM method used with the reinitialization of [5] exhibits a chaotic behaviour without mesh convergence. The results corresponding to $\Delta x_{\min} = 50\mu m$ for the MM/Desjardins 2008 algorithms are not shown here due to higher error levels and no mesh convergence.



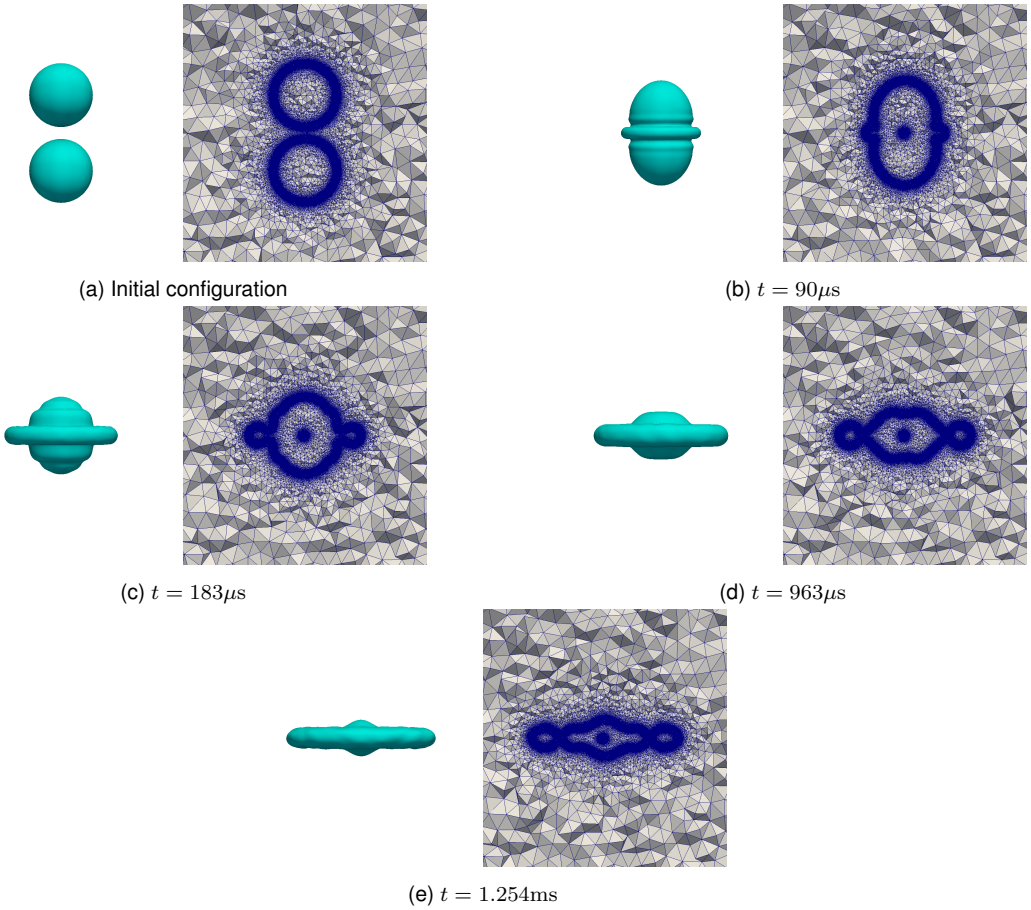
(a) Desjardins 2008; MM

(b) Chiodi 2017; GPMM

Figure 12. Temporal evolution of 3D spurious currents at various mesh resolutions, for the MM/Desjardins 2008 and GPMM/Chiodi 2017 algorithms, $\mathrm{La} = 111188$

Application to 3D complex dynamics

To demonstrate the ability of the methods, namely the GPMM and Chiodi's reinitialization, to be coupled with dynamic mesh adaptation, a simulation of a head-on water droplet collision with reflexive separation in air is run. Two droplets of equal size (diameter $D_d = 800\mu\mathrm{m}$) are initialized in a $[-5\mathrm{mm}, 5\mathrm{mm}]^3$ cubic domain (Fig. 13a). To compare with experiments, the same parameters than in [4] are chosen. The Weber number is $\mathrm{We} = \rho_l D_d u_r^2/\sigma = 23$, with u_r the relative velocity ($u_r = 1.44\mathrm{m/s}$ here), and the Ohnesorge number is $\mathrm{Oh} = \mu/\sqrt{\sigma \rho_l D_d} = 0.0047$. The narrow-band width is set to 8 to avoid undershoots/overshoots of ψ, and 3 steps of reinitialization are performed per time step. The minimum cell size is imposed in a band around the interface of half-width $8\Delta x_{\min} = 48\mu\mathrm{m}$, then the cell size is linearly increased until reaching a maximum coarse cell size. Dynamic mesh adaptation is performed every $0.7\mu\mathrm{s}$ to accurately follow the interface dynamics of the droplets and to avoid any cell size gradient in the ψ profile. The main parameters of the run are summarized in Fig. 13. Instantaneous interface positions and meshes are presented in Figs. 13a to 13d, showing good results in terms of interface quality and accuracy in the case of a complex dynamics, during a very large number of iterations ($\Delta t \sim 10^{-8}\mathrm{s}$ for $\mathrm{SN} = 0.5$). Moreover, interface topology is correctly reproduced compared to the experimental results of [4]. The overall algorithm used for this simulation perfectly ensures mass conservation, however kinetic energy is not strictly conserved.



(a) Initial configuration

(b) $t = 90\mu\mathrm{s}$

(c) $t = 183\mu\mathrm{s}$

(d) $t = 963\mu\mathrm{s}$

(e) $t = 1.254\mathrm{ms}$

Figure 13. Head-on droplet collision: $\mathrm{We} = 23$; $D_d = 800\mu\mathrm{m}$; $\mathrm{Oh} = 0.0047$; $\Delta x_{\min} = 6\mu\mathrm{m}$; 25 million tetrahedra.

Conclusions

This article introduces a novel method based on geometric markers to compute the signed-distance to a front on unstructured grids. The calculation of the signed-distance function is accurate and converges at second-order. It also enables to achieve convergence of the curvature, which is very difficult to obtain on unstructured meshes. An implementation of the reinitialization equation of Chiodi et al. [3] for unstructured meshes has also been presented, leading to a well-preserved interface shape, especially in capillary-driven cases, and to a correct front topology. The combination of all these methods has been validated on various cases and it has been shown that it drastically reduces spurious currents in the 2D and 3D static droplet cases. Further validation on more complex cases is still in progress but very promising results have been obtained for the head-on droplet collision case with dynamic mesh adaptation.

Acknowledgements

Computer resources for this project have been provided by CRIANN, France, under the allocation 2012006 and at CINES by GENCI, France, under the allocation A0052B06880.

References

[1] Leparoux, J., Mercier, R., Moureau, V., Musaefendic, H., July 22-26. 2018, 14th Triennal International Conference on Liquid Atomization and Spray Systems.
[2] Olsson, E., and Kreiss, G., 2005, *Journal of Computational Physics*, 210, pp. 225-246.
[3] Chiodi, R., Desjardins, O., 2017, *Journal of Computational Physics*, 343, pp. 186-200.
[4] Ashgriz, N., Poo, J.Y., 1990, *Journal of Fluid Mechanics*, 221, pp. 183-204.
[5] Desjardins, O., Moureau, V., Pitsch, H., 2008, *Journal of Computational Physics*, 227, pp. 8395-8416.
[6] Goldman, R., 2005, *Computer Aided Geometric Design*, 22, pp. 632-658.
[7] Fedkiw, R., Aslam, T., Merriman, B., Osher, S., 1999, *Journal of Computational Physics*, 152, pp. 457-492.
[8] Wikipedia: Delaunay triangulation, https://en.wikipedia.org/wiki/Delaunay_triangulation
[9] Zalesak, S.T., 1979, *Journal of Computational Physics*, 362, pp. 335-362.
[10] LeVeque, R.J., 1996, *SIAM Journal on Numerical Analysis*, 33(2), pp. 627-665.
[11] Kraushaar, M., 2011, *PhD Thesis, Institut National Polytechnique de Toulouse*.