



# A coupled level-set and reference map method for interface representation with applications to two-phase flows simulation

Thomas Bellotti <sup>a,b</sup>, Maxime Theillard <sup>c,\*</sup>

<sup>a</sup> *Département de Mathématiques Appliquées, Ecole polytechnique, 91128 Palaiseau, France*

<sup>b</sup> *Dipartimento di Matematica, Politecnico di Milano, 20133 Milan, Italy*

<sup>c</sup> *School of Natural Sciences, Applied Mathematics Department, University of California, Merced, CA 95340, USA*

## ARTICLE INFO

### Article history:

Received 20 June 2018

Received in revised form 1 May 2019

Accepted 2 May 2019

Available online 9 May 2019

### Keywords:

Interface problems

Level set

Reference map

Reinitialization

Navier-Stokes

Two-phase flows

## ABSTRACT

We present a novel hybrid methodology combining the reference map theory with the level-set method for tracking moving interfaces. Instead of directly advecting the level-set function, we track the reference map, which maps the deformed state into the original one. We then reconstruct the deformed level-set function from this mapping and the original geometry. Because of the better smoothness of the reference map and the reduced impact of the reinitialization, this new approach grants higher precision. This results in significant improvements regarding interface location precision and mass conservation, especially in situations involving small deformations. Moreover, the implementation of the method is straightforward since it is based on the same numerical techniques used for the level-set method. Our coupled method is extensively validated in the case of externally generated velocity fields and incorporated into our previously introduced two-phase incompressible Navier-Stokes flow solver.

Published by Elsevier Inc.

## 1. Introduction

Deforming interfaces play a crucial role in many real-life applications, among which multiphase flows, combustion, solidification processes or cellular dynamics in biology are some of the most paradigmatic examples. Their modeling and simulation require to capture both the interface and its evolution accurately and efficiently. Consequently, over the last decades, numerous mathematical approaches have been developed to address this issue. These approaches can be classified into two categories. Explicit methods, such as the front tracking, represent the interface by a discrete mesh lying on and evolving with the interface. It was first presented in the context of shock fitting by Richtmyer and Morton [47], later formally developed by Glimm et al. [18,17] and more recently popularized by Tryggvason et al. [63]. With this representation, very high accuracy can be achieved. Nonetheless, the re-meshing procedure becomes challenging and computationally expensive as the interface undergoes large deformations or topological changes.

Implicit representations alleviate the need for re-meshing by capturing the interface through an auxiliary function. In the phase-field method, as described in [3], the so-called order parameter function is taken to be constant in each phase and varies smoothly but rapidly across the interface over a prescribed length-scale.

\* Corresponding author.

E-mail address: [mtheillard@ucmerced.edu](mailto:mtheillard@ucmerced.edu) (M. Theillard).

The significant disadvantage of this approach is the scale separation from the macroscopic scale and that of the phase-transition layer (see [52]), which is a purely mathematical artifact and needs to be kept small for accurate results. As a result, very high spatial resolution is needed close to the interface for the outcomes to be quantitative. Furthermore, [44] points out that the presence of several spatial scales makes the phase-field equations particularly stiff. With the level-set method, introduced in the seminal paper [39] by Osher and Sethian, the interface is defined by the zero level set of a scalar-valued function, advected by the same velocity field. For the problem to be well posed, the function is assumed to be a signed distance function or equivalently reinitialized. In this case, no artificial length scales are introduced and topological changes are automatically handled. The method involves two equations: the level-set equation capturing the deformation of the interface and the reinitialization equation, used to enforce the signed distance condition.

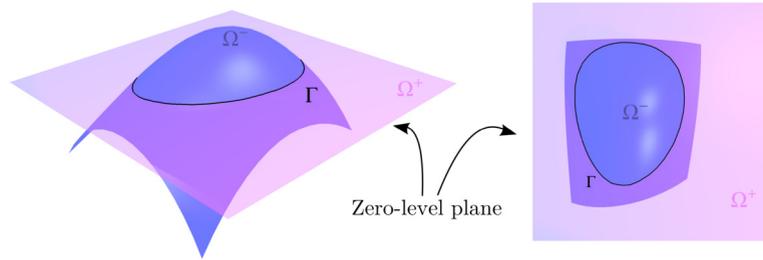
For the solution of the level-set equation, the finite-differences semi-Lagrangian approach is probably the most natural way of proceeding. Consequently, it is used in the present work. Enright et al. [14] claim that the semi-Lagrangian methods are effective and precise even with low-order discretizations, as also done by Strain in [53]. On the other hand, Xiu and Karniadakis [66] showed that, especially when facing turbulence phenomena, a high-order semi-Lagrangian discretization is beneficial. Indeed, the simplest way to reduce the numerical error and thus enhance mass conservation is to use high-order discretizations of the level-set equation. More modern approaches like the extended finite element [23] or the discontinuous Galerkin [45,46,25] methods have been successfully applied to the level set. Another improvement which has been developed is the extension-velocity level-set method, introduced by [1] and consisting in advecting the level-set function with a different velocity field than the fluid velocity. This modified velocity field  $\mathbf{v}_{\text{mod}}$  aims, by setting  $\nabla \mathbf{v}_{\text{ext}} \cdot \nabla \phi = 0$ , at preserving the property of the level set being a signed distance function:  $|\nabla \phi| = 1$ , where  $\phi$  is the level-set function. To this extent, Ovsyannikov et al. [40] introduced a corrected velocity field obtained by adding a source term to the level-set equation. This methodology has the drawback of providing only a first-order approximation of the velocity field close to the interface, leading to some artifacts. More recently, this method has been improved by [49] in order to achieve high-order accuracy. A more radical way of improving mass conservation is the so-called hyperbolic tangent level-set method, firstly presented by Olsson et al. in [36], using a hyperbolic tangent function as level set in lieu of the standard signed distance function. This type of function is a common choice for the phase-field methods. The approach has been tested in several other studies [37,65,11,12,35], showing a general tendency to generate non-physical drops breaking off where the mesh is under-resolved, called “flotsams”. [41] uses this strategy in combination with a discontinuous Galerkin method.

Knowing that the reinitialization of the level-set function is a crucial step, responsible for a large fraction of the numerical error, improvements in the reinitialization technique have been investigated thoroughly. Several ways of proceeding have been indicated in [58,57,48,20,21], principally by Sussman et al. and Hartmann et al., adding some correction terms (called “re-distancing” by Sussman) when discretizing the reinitialization equation (constrained reinitialization). Following this idea, Chang et al. [7] change the reinitialization equation which becomes a perturbed Hamilton-Jacobi equation, which they solve until reaching its steady state. Recently, the level-set method has been associated with other computational techniques in order to enhance its performance. These techniques include the augmented level set introduced in [34], where up to fourth-order accuracy can be achieved by transporting both the level-set function and its gradient. Another improved level-set approach is the so-called Coupled Level Set and Volume of Fluid method from [59,55], developed in the context of two-phase flows. In this approach, the evolution of the level-set function is coupled with the evolution of the volume of fluid in each computational cell, such that the total mass is exactly preserved.

The notion of reference map arises naturally in the context of solid mechanics, where it is the isomorphism mapping the deformed configuration to the reference one. To the best of our knowledge, it was first associated with the level-set method by Pons et al. in [42] and inspired by the work of Adalsteinsson and Xu [2,67], to keep track of quantities defined on a moving interface, using a point correspondence function and projections onto the interface. In their approach, the level-set function is advected in the traditional level-set fashion. It was later used in the field of fluid-structure interactions [9,29] to model isotropic and anisotropic elastic materials, where the coexistence of solid and liquid phases calls for a hybrid Lagrangian/Eulerian framework. More recently, this method was employed again to simulate coupling with elastic materials in [24,64] using finite differences, in both providing extensive validations showing the convergence of the reference map, the velocity field, and the stress tensor, as well as reduced computational times.

In this work, we couple the level-set and reference map methods to model and simulate the general interface evolution under an external velocity field. Unlike [42], where both quantities are independently advected, we only transport the reference map and use it to reconstruct the level set. In place of the reinitialization, we define a restarting procedure, which we only perform when needed. The resulting overall method employs the same techniques and has a similar cost as the classical level-set method, yet it achieves systematic higher accuracy and ensures strong limitations of mass loss. Consequently, our hybrid method outstands for its minimalism and simplicity.

The present paper is structured as follows. In Section 2, we start by introducing the level-set and reference map methods separately before explaining how they are coupled. Our numerical implementation on adaptive quadtree grids, based on a semi-Lagrangian discretization, is detailed in the following section. In Section 4, we consider the motion of an interface under an external analytic velocity field and compare our results against those obtained with the level-set technique developed in [33]. In Sections 5 and 6, we apply our method to simulate two-phase flows and reproduce some canonical examples such as studying parasitic currents, simulating a rising bubble of gas and observing the phenomenon of droplet oscillations induced by the surface tension. We conclude in Section 7.



**Fig. 1.** Level-set representation of a contour  $\Gamma$ , separating two subdomains  $\Omega^+$  and  $\Omega^-$ . The intersection between the level-set surface and the zero-level plane determines  $\Gamma$ .

## 2. Coupled level-set and reference map method

### 2.1. Level-set method

Using an auxiliary scalar-valued function  $\phi$ , the level-set method identifies a closed interface  $\Gamma$  and the two subdomains  $\Omega^-$  and  $\Omega^+$  (with  $\Omega = \Omega^- \cup \Omega^+$ ) that it separates (see Fig. 1) as follows:

$$\begin{aligned} \Gamma &= \{\mathbf{x} \in \Omega : \phi(\mathbf{x}) = 0\}, \\ \Omega^- &= \{\mathbf{x} \in \Omega : \phi(\mathbf{x}) < 0\}, \\ \Omega^+ &= \{\mathbf{x} \in \Omega : \phi(\mathbf{x}) > 0\}. \end{aligned} \tag{1}$$

From the level-set function  $\phi$ , all geometrical quantities involving the interface can be recovered. For example, its normal vector  $\mathbf{n}$  and the mean curvature  $\kappa$  are given by

$$\mathbf{n} = \frac{\nabla\phi}{|\nabla\phi|}, \quad \kappa = \nabla \cdot \mathbf{n} = \nabla \cdot \left( \frac{\nabla\phi}{|\nabla\phi|} \right). \tag{2}$$

Since for any given contour  $\Gamma$ , definition (1) does not uniquely define the function  $\phi$ , we enforce that it is a signed distance function, i.e.

$$|\nabla\phi(\mathbf{x})| = 1 \quad \forall \mathbf{x} \in \Omega. \tag{3}$$

As it is remarked in [38], any level-set function can be reinitialized by solving the following reinitialization equation for a fictitious time  $\tau$

$$\frac{\partial\phi}{\partial\tau} + \text{sign}(\phi)(|\nabla\phi| - 1) = 0 \quad 0 \leq \tau \leq t_f, \quad \forall \mathbf{x} \in \Omega, \tag{4}$$

in order to make it arbitrarily close to be a signed distance function without theoretically moving the interface  $\Gamma$ . As the interface evolves under a known velocity field  $\mathbf{u}$ , the level-set function is the solution of the following advection equation

$$\begin{cases} \frac{\partial\phi}{\partial t} + \mathbf{u} \cdot \nabla\phi = 0 & t \geq 0, \quad \forall \mathbf{x} \in \Omega, \\ \phi(t = 0, \mathbf{x}) = \phi_0(\mathbf{x}) & \forall \mathbf{x} \in \Omega. \end{cases} \tag{5}$$

Even if the initial condition of Eq. (5) is a signed distance function, the future solution may not remain a signed distance function. Therefore, after each advection step, the new solution needs to be systematically reinitialized solving Eq. (4) for a fixed fictitious final time  $\tau_f$ . This is done in accordance with [19,62,56,58,14,37,30,34,41], where the reinitialization is performed at every time step. This choice complies with the observation made by [31]: Eq. (4) will rapidly (for small  $t_f$ ) provide a good signed distance function if  $\phi$  is reinitialized at each discrete instant of time. From a broader point of view, this follows the idea by Osher and Fedkiw in [38], who highlighted the necessity of reinitializing on a periodic basis and more often when dealing with a signed distance function is essential. We will eventually refer to this technique as the “standard level-set approach”, though the frequency of application of the reinitialization procedure is not unanimously defined in the literature (see discussions in [16,41]). The numerical integration of Eqs. (5) and (4) will be detailed in Sections 3.2 and 3.4.

### 2.2. Reference map method

Following [24], we consider a reference domain  $\mathcal{B}_0 \subset \mathbb{R}^d$ , with  $d = 2, 3$ , possibly unbounded, representing the entire domain of interest at the initial time  $t = 0$ . We assume that  $\mathcal{B}_0$  is deformed by a velocity field  $\mathbf{u}$ , and we call  $\mathcal{B}(t) \subset \mathbb{R}^d$  the

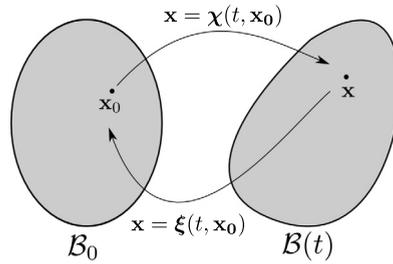


Fig. 2. The motion map  $\chi(t)$  transforms the initial geometry  $\mathcal{B}_0$  into the deformed one  $\mathcal{B}(t)$ : any point  $\mathbf{x}_0$  will be transported to  $\mathbf{x}(t) = \chi(t) \in \mathcal{B}(t)$ . The reference map  $\xi(t)$  is the inverse transformation.

deformed domain at time  $t > 0$ . We introduce the so-called “motion” morphism  $\chi$ , which maps every material point  $\mathbf{x}_0$  on the initial configuration to its corresponding image  $\mathbf{x}$  on the deformed one (see Fig. 2), hence we write:

$$\mathbf{x} = \chi(t, \mathbf{x}_0) \quad t \geq 0, \quad \forall \mathbf{x}_0 \in \mathcal{B}_0. \tag{6}$$

One can easily see that  $\mathcal{B}(t)$  is nothing but the image of  $\mathcal{B}_0$  through  $\chi(t, \cdot)$ . Assuming that  $\chi(t, \cdot)$  is an isomorphism, which is for example the case if  $\det(\nabla \chi) > 0$ , we call the inverse of the motion the reference map  $\xi(t, \mathbf{x})$ :

$$\mathbf{x}_0 = \xi(t, \mathbf{x}) \quad t \geq 0, \quad \forall \mathbf{x} \in \mathcal{B}(t). \tag{7}$$

As Fig. 2 illustrates, in a pure Eulerian fashion, considering a point  $\mathbf{x}$  in the deformed space on which a particle lies at time  $t$ , we obtain, by the action of the function  $\xi$ , the position at which this particle was at  $t = 0$ , in the reference configuration. In order to derive an evolution equation for  $\xi$ , we consider a tracer particle located at  $\mathbf{x}$  at time  $t > 0$ , which has the property that its reference location does not change in time. Hence we can write:

$$\frac{d\mathbf{x}_0}{dt} = \frac{d\xi}{dt}(t, \mathbf{x}) = 0 \quad t \geq 0, \quad \forall \mathbf{x} \in \mathcal{B}(t), \tag{8}$$

which gives the system satisfied by the reference map  $\xi$  by application of the chain rule:

$$\begin{cases} \frac{\partial \xi}{\partial t} + \mathbf{u} \cdot \nabla \xi = 0 & \forall t \geq 0, \quad \forall \mathbf{x} \in \mathcal{B}(t), \\ \xi(t = 0, \mathbf{x}) = \mathbf{x} & \forall \mathbf{x} \in \mathcal{B}_0. \end{cases} \tag{9}$$

### 2.3. Coupled method

In order to advect the interface  $\Gamma$ , we do not advance the level-set function  $\phi$  according to Eq. (5) but instead, we deform the entire space on which  $\phi$  is defined by solving Eq. (9) and then recover the level set from its initial profile and the deformed space.

In this way, we take  $\mathcal{B}_0$  large enough to include the entire computational domain and consider the deforming velocity field  $\mathbf{u}$  to be the one advecting the interface. The level-set reconstruction formula is obtained from Eq. (5), showing that the level-set function is constant along the characteristics, implying that

$$\phi(t, \mathbf{x}) = \phi_0(\xi(t, \mathbf{x})) \quad t \geq 0, \quad \forall \mathbf{x} \in \mathcal{B}(t). \tag{10}$$

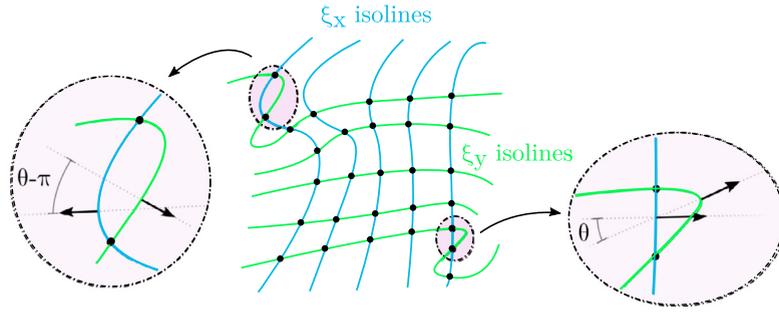
We point out that Eq. (9) could be recovered by injecting Eq. (10) in Eq. (5). One main difference between our coupled method and the standard level set is the regularity of the advected object, and therefore the accuracy of the advection. Assuming that the velocity field is smooth, with our method the reference map will remain smooth. On the other hand, with the level-set method, the reinitialization will always produce discontinuities in the gradient of the level-set function.

### 2.4. Restarting procedure

As we pointed out before, the bijectivity of  $\chi(t, \mathbf{x})$ , and thus of  $\xi(t, \mathbf{x})$ , is essential to the reference map technique. Even if in theory the velocity field  $\mathbf{u}$  can be chosen to ensure that this property is preserved, it can be lost numerically, especially when facing large deformations, as remarked in [24]. Aiming at numerically preserving the bijectivity of  $\xi(t, \mathbf{x})$ , we propose a restarting procedure and a criterion telling us when to use it.

The restarting works as follows. Suppose that our criterion identifies a critical time  $t_c > 0$  at which restarting is needed, then – from that moment on – we take:

$$\begin{aligned} \phi_0(\mathbf{x}) &= \mathcal{R}(\phi_0(\xi(t_c, \mathbf{x}))) & \forall \mathbf{x} \in \Omega, \\ \xi(t_c, \mathbf{x}) &= \mathbf{x} & \forall \mathbf{x} \in \mathcal{B}(t_c), \end{aligned}$$



**Fig. 3.** In these two examples, the contour lines of  $\xi_x$  (blue) and of  $\xi_y$  (green) intersect twice and therefore  $\xi$  is not bijective. These degenerate cases are characterized by a locally small angle  $\theta$  between the normal vectors to the contours. (For interpretation of the colors in the figure(s), the reader is referred to the web version of this article.)

where  $\mathcal{R}(\phi_0(\xi(t_c, \mathbf{x})))$  is the reinitialization of the level set  $\phi_0(\xi(t_c, \mathbf{x}))$ . We note that even if the restarting procedure is systematically applied, our method still differs from the standard level-set method. In particular, the numerical errors due to the reinitializations are propagated differently. With the standard level-set method, the advected object (the level-set function) carries the errors of the reinitialization. With our coupled method, the advected object (the reference map) does not. We also note that because of the restarting criterion, less reinitialization will be performed with our coupled method than with the standard level-set approach, reducing significantly the numerical error.

The restarting criterion is based on the following observations: when  $\xi = (\xi_x, \xi_y)$  (or  $\xi = (\xi_x, \xi_y, \xi_z)$  for  $d = 3$ ) is bijective, all the contours of  $\xi_x$  intersect each contour of  $\xi_y$  at most once. The bijectivity is lost if at least one contour of  $\xi_x$  intersects one contour of  $\xi_y$  at least twice. Starting from a bijective reference map, we observed that multiple intersections take place when the tangential (or normal) vectors to the contour lines of  $\xi_x$  and  $\xi_y$  are close to being collinear (see Fig. 3). In addition, the loss of bijectivity is only critical close to the interface, where the level-set values matter the most. Therefore we define the restarting criterion:

$$\text{restart } \xi \text{ if } : \left\| \frac{\nabla \xi_x}{\|\nabla \xi_x\|} \cdot \frac{\nabla \xi_y}{\|\nabla \xi_y\|} \right\|_{L^\infty(\mathcal{V})} > \cos(\theta_{crit}), \tag{11}$$

where  $\theta_{crit} \in [0, \frac{\pi}{2}]$  is an arbitrary (possibly small) critical angle defined by the user. An excessively small critical angle might prevent from restarting when it is needed and have unwanted effects on the convergence. On the other hand, an unreasonably large angle might restart too often and introduce unnecessary computational costs and numerical errors. The neighborhood close to the interface  $\mathcal{V}$  is defined as:

$$\mathcal{V} = \{\mathbf{x} \in \Omega : |\phi(t, \mathbf{x})| < \epsilon_r\} \quad t \geq 0, \tag{12}$$

where  $\epsilon_r$  is another arbitrary parameter. In this work, we will consistently use the following:

$$\theta_{crit} = \frac{\pi}{9} \ll \frac{\pi}{2}, \quad \epsilon_r = 0.1 \cdot \left( \max_{\mathbf{x} \in \Omega} \phi_0(\mathbf{x}) - \min_{\mathbf{x} \in \Omega} \phi_0(\mathbf{x}) \right),$$

which provide accurate results for all the examples presented in this paper. We note that  $\epsilon_r$  is chosen to scale with the initial range of level-set value. A systematic study could identify optimal values for both parameters.

In the three-dimensional context ( $d = 3$ ), we have to consider the angles between the three normal surfaces, and therefore we restart if:

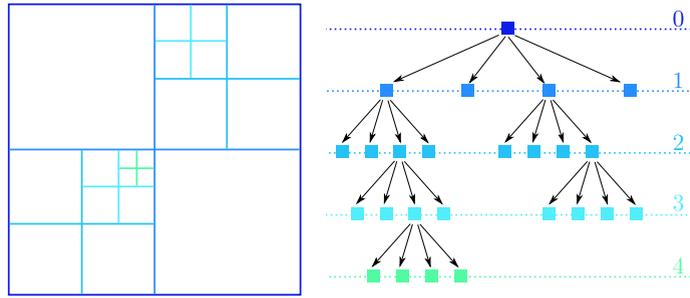
$$\max \left\{ \left\| \frac{\nabla \xi_x}{\|\nabla \xi_x\|} \cdot \frac{\nabla \xi_y}{\|\nabla \xi_y\|} \right\|_{L^\infty(\mathcal{V})}, \left\| \frac{\nabla \xi_x}{\|\nabla \xi_x\|} \cdot \frac{\nabla \xi_z}{\|\nabla \xi_z\|} \right\|_{L^\infty(\mathcal{V})}, \left\| \frac{\nabla \xi_y}{\|\nabla \xi_y\|} \cdot \frac{\nabla \xi_z}{\|\nabla \xi_z\|} \right\|_{L^\infty(\mathcal{V})} \right\} > \cos(\theta_{crit}).$$

Even though with our current definition, restarting and reinitialization are tied together, their purposes are unrelated. The restarting procedure aims at avoiding the degenerate cases where the reference map is no longer bijective. The reinitialization ensures that the level-set function remains a signed distance function. Therefore, it is possible that a restarting is needed while the level-set function is perfectly reinitialized. Our restarting criteria could certainly be refined to detect these cases and decide to reinitialize or not as part of the restarting.

On the other hand, it is also possible that the level-set function deviates from a signed distance function, while the contours of  $\xi_x$  and  $\xi_y$  remain orthogonal and therefore the reference map remains bijective. It is, for example, the case for an initially circular interface, centered at the origin and deformed by the velocity field  $\mathbf{u} = (x, 0)$ .

The deviation of the level-set function from a signed distance function can be estimated by differentiating Eq. (10)

$$\|\nabla \phi(t, \mathbf{x})\| = \|\nabla \phi_0(\mathbf{x}) \cdot \nabla \xi(t, \mathbf{x})\| \quad t \geq 0, \quad \forall \mathbf{x} \in \Omega,$$



**Fig. 4.** Quadtree grids and their representation: starting from the root cell (in dark blue, level 0) representing the entire computational domain, which is split into 4 identical daughter cells (level 1). The refinement strategy then identifies which of these 4 cells should be refined and splits them into 4 new identical cells (level 2). This process is recursively repeated on the newly created cells, until the maximum level is reached (4 in this case).

and since  $\phi_0$  is reinitialized, we obtain the following upper bound for the norm of the gradient of  $\phi$

$$\|\nabla\phi(t, \mathbf{x})\| \leq \|\nabla\xi(t, \mathbf{x})\| \quad t \geq 0, \quad \forall \mathbf{x} \in \Omega,$$

where  $\|\cdot\|$  is the matrix norm induced by any vector norm  $\|\cdot\|$ , namely:

$$\|A\| = \sup_{\substack{\mathbf{u} \in \mathbb{R}^d \\ \|\mathbf{u}\|=1}} \|A\mathbf{u}\| \quad \forall A \in \mathbb{R}^{d \times d}.$$

### 3. Numerical algorithm

#### 3.1. Sampling and data structure

As it was done in previous studies [33,19,62,61,60,10], data are stored at the nodes of a non-graded quadtree grid, whose construction is illustrated in Fig. 4. The mesh is created by mean of a recursive splitting using an arbitrary criterion. Each cell is said to be of level  $N$  if it has been created after splitting the original root cell  $N$  successive times. Hence, if the mother cell has size 1 in one direction, a cell of level  $N$  has size  $2^{-N}$  in this same direction. We introduce upper and lower limitations to the level, which are called respectively  $\max_{\text{level}}$  and  $\min_{\text{level}}$ .

We use the following criterion to refine mesh where it is necessary: we want to correctly take the interface  $\Gamma$  into account. Let us consider a cell  $C$  in the quadtree, then we use (see [33]):

$$\text{split } C \text{ if } : \quad \min_{n \in \text{nodes}(C)} |\phi(n)| \leq \text{Lip}(\phi) \cdot \text{diag}(C) \quad \text{and} \quad \text{level}(C) \leq \max_{\text{level}}, \quad (13)$$

where  $\text{Lip}$  is the Lipschitz constant of  $\phi$  and  $\text{diag}(C)$  is the length of the diagonal of cell  $C$ . We point out that the above criteria can easily be modified to enforce that interface is surrounded by a uniform band of maximum resolution and arbitrary width.

#### 3.2. Advection of the reference map

To evolve the reference map, we follow [33] and use the second-order accurate semi-Lagrangian scheme proposed in [66]. At each iteration, to compute  $\xi^{n+1}$  at any grid point  $\mathbf{x}^{n+1}$ , we follow the characteristic curve passing through  $(t_{n+1}, \mathbf{x}^{n+1})$  to find the departure point  $\mathbf{x}_d^n$  at time  $t_n$ . To this end, we employ the second-order mid-point method (which is a Backward Runge-Kutta method of order two):

$$\hat{\mathbf{x}} = \mathbf{x}^{n+1} - \frac{\Delta t_n}{2} \mathbf{u}^n(\mathbf{x}^{n+1}),$$

$$\mathbf{x}_d^n = \mathbf{x}^{n+1} - \Delta t_n \mathbf{u}^{n+\frac{1}{2}}(\hat{\mathbf{x}}).$$

In these formulæ,  $\Delta t_n$  and  $\Delta t_{n-1}$  are the time steps from  $t_n$  to  $t_{n+1}$  and from  $t_{n-1}$  to  $t_n$ . Once the departure point is known, the new value of  $\xi^{n+1}$  is obtained by interpolating  $\xi^n$  at  $\mathbf{x}_d^n$ , using the third-order accurate interpolations proposed in [33]. Specifically, if  $\mathbf{x}_d^n = (x, y)$  is in cell  $C$ , which we assume to be  $[0, 1]^2$  for the sake of presentation

$$\phi(x, y) = \phi_{00}(1-x)(1-y) + \phi_{01}(1-x)y + \phi_{10}x(1-y) + \phi_{11}xy - \phi_{xx} \frac{x(1-x)}{2} - \phi_{yy} \frac{y(1-y)}{2}, \quad (14)$$

where the second order derivatives are computed as

$$\phi_{xx} = \min_{n \in \text{nodes}(C)} \phi_{xx}^n, \quad \phi_{yy} = \min_{n \in \text{nodes}(C)} \phi_{yy}^n,$$

where  $\phi_{xx}^n, \phi_{yy}^n$  are the discrete second-order derivatives at node  $n$ , computed using the finite-differences method presented in [33].

The intermediate velocity  $\mathbf{u}^{n+\frac{1}{2}}(\hat{\mathbf{x}})$  is computed, when not explicitly known, through interpolation, using:

$$\mathbf{u}^{n+\frac{1}{2}}(\hat{\mathbf{x}}) = \frac{2\Delta t_{n-1} + \Delta t_n}{2\Delta t_{n-1}} \mathbf{u}^n(\hat{\mathbf{x}}) - \frac{\Delta t_n}{2\Delta t_{n-1}} \mathbf{u}^{n-1}(\hat{\mathbf{x}}).$$

It is not difficult to see that this approximation is second-order accurate in time (here the  $\mathbf{u}$ s indicate the exact velocity field, not the discretized one) using formal Taylor expansions:

$$\begin{aligned} \frac{2\Delta t_{n-1} + \Delta t_n}{2\Delta t_{n-1}} \mathbf{u}^n(\hat{\mathbf{x}}) - \frac{\Delta t_n}{2\Delta t_{n-1}} \mathbf{u}^{n-1}(\hat{\mathbf{x}}) &= \frac{2\Delta t_{n-1} + \Delta t_n}{2\Delta t_{n-1}} \left( \mathbf{u}^{n+\frac{1}{2}} - \frac{\Delta t_n}{2} \partial_t \mathbf{u}^{n+\frac{1}{2}} + O(\Delta t_n^2) \right) (\hat{\mathbf{x}}) \\ &\quad - \frac{\Delta t_n}{2\Delta t_{n-1}} \left( \mathbf{u}^{n+\frac{1}{2}} - \left( \frac{\Delta t_n}{2} + \Delta t_{n-1} \right) \partial_t \mathbf{u}^{n+\frac{1}{2}} + O(\Delta t_n^2) \right) (\hat{\mathbf{x}}) \\ &= \mathbf{u}^{n+\frac{1}{2}}(\hat{\mathbf{x}}) + O(\Delta t_n^2) + O(\Delta t_{n-1}^2). \end{aligned}$$

Again, the values of  $\mathbf{u}^n(\hat{\mathbf{x}})$  and  $\mathbf{u}^{n-1}(\hat{\mathbf{x}})$  are recovered using a third-order interpolation. This scheme is convergent as the time step  $\Delta t_n \rightarrow 0$  in virtue of the Lax-Richtmyer theorem, provided that the interpolations are stable.

This numerical method will also be used to advect the level-set function solving Eq. (5), in order to perform numerical comparisons and validations.

### 3.3. Reconstruction of the level set

Once the reference map has been updated as shown in 3.2, the new level-set value at each grid node is computed using Eq. (10) and third-order accurate interpolations.

### 3.4. Reinitialization equation

The reinitialization equation (4) is solved using a Total Variation Diminishing second-order Runge-Kutta scheme (TVD-RK2), introduced for the first time in [51]. It is constructed by first discretizing Eq. (4) in space

$$\frac{\partial \phi}{\partial \tau} + \hat{\mathcal{H}}_{\text{GD}}(\phi_{x^+}, \phi_{x^-}, \phi_{y^+}, \phi_{y^-}) = 0 \quad 0 \leq \tau \leq t_f, \tag{15}$$

defining the Godunov Hamiltonian

$$\hat{\mathcal{H}}_{\text{GD}}(a, b, c, d) = \text{sign}(\phi^0) \cdot \begin{cases} \sqrt{\max(a^+, b^-)^2 + \max(c^+, d^-)^2} - 1 & \text{if } \text{sign}(\phi^0) \geq 0 \\ \sqrt{\max(a^-, b^+)^2 + \max(c^-, d^+)^2} - 1 & \text{if } \text{sign}(\phi^0) < 0, \end{cases}$$

from the first one-sided derivatives  $\phi_{x^\pm}, \phi_{y^\pm}$ , with  $a^+ = \max(a, 0)$  and  $a^- = \min(a, 0)$ . The equation is then solved in fictitious time using a TVD-RK2 scheme: we start by computing  $\tilde{\phi}^{n+1}$  and  $\tilde{\phi}^{n+2}$ , solutions of

$$\begin{aligned} \frac{\tilde{\phi}^{n+1} - \phi^n}{\Delta \tau} + \hat{\mathcal{H}}_{\text{GD}}^n &= 0, \\ \frac{\tilde{\phi}^{n+2} - \tilde{\phi}^{n+1}}{\Delta \tau} + \hat{\mathcal{H}}_{\text{GD}}^n &= 0, \end{aligned}$$

and then constructing  $\phi^{n+1}$  as

$$\phi^{n+1} = \frac{\tilde{\phi}^{n+2} + \phi^n}{2}.$$

### 3.5. Mesh and solution update

At the end of each iteration, the mesh is adapted to the new geometry using the criterion given by Eq. (13). The current solution is then interpolated on this new mesh using the quadratic interpolation (14).

#### 4. Advection under an externally generated velocity field

In this section, we validate our method for analytical external velocity fields. In all the examples, the interface is advected under a constant velocity field until  $t = T/2$ , when the velocity field is reversed and the simulation is carried until  $t = T$ . This way, at the final time, the analytic solution  $\phi(T, \mathbf{x})$  is exactly the initial condition  $\phi_0(\mathbf{x})$ . For the two-dimensional examples, the computational domain is the square  $\Omega = [0, 1]^2$  and the final time is  $T = 2\pi$ .

We compare our coupled method to the standard level-set method, where the solution is reinitialized at each iteration. The advection of  $\xi$  or  $\phi$  is performed using the semi-Lagrangian method presented in Subsection 3.2, and – in the case of  $\phi$  – solving Eq. (4) as hinted in Subsection 3.4. The initial circular contour of center  $(x_c, y_c) \in \Omega$  and radius  $R > 0$  is given by the level set

$$\phi(t = 0, x, y) = \phi_0(x, y) = \sqrt{(x - x_c)^2 + (y - y_c)^2} - R, \tag{16}$$

where for the first three examples the initial contour is a circle of radius  $R = 0.15$  centered at  $(x_c, y_c) = (0.5, 0.75)$ . Naturally, the reference map is initialized as

$$\xi(t = 0, x, y) = (x, y). \tag{17}$$

In all tests, we monitor the  $L^\infty$ -error of the level-set function  $e_{L^\infty}$ , the errors on the interface position at the final time  $T$  and the relative mass loss (expressed in percentage). The  $L^\infty, L^1$  and  $L^2$ -errors of the interface position are defined from the distance to the interface function  $\delta(\mathbf{x})$  as

$$d_{L^\infty}(\phi) = \max_{\mathbf{x} \in N(\phi)} |\delta(\mathbf{x}) - \delta_{\text{exact}}(\mathbf{x})|,$$

$$d_{L^1}(\phi) = \int_{\Gamma_{\text{exact}}} |\delta(\mathbf{x})| dl,$$

$$d_{L^2}(\phi) = \left( \int_{\Gamma_{\text{exact}}} \delta(\mathbf{x})^2 dl \right)^{1/2}.$$

The  $\delta(\mathbf{x})$  function is constructed numerically at the set of nodes  $N(\phi)$  near the interfaces (i.e. at least one of their direct neighbors is across the interface), using standard third-order approximations. The contour integrals in the above definitions and the volume integral intervening in the mass loss calculations are computed using the second-order techniques detailed in [32]. We point out that when using the standard level-set method,  $\delta(\mathbf{x})$  is the level-set function itself since the level-set function is systematically reinitialized. Therefore the errors  $e_{L^\infty}(\phi)$  and  $d_{L^\infty}(\phi)$  are identical. They are not identical when using our coupled method since the level-set function does not remain reinitialized throughout the whole advection process. The time step is defined from the smallest grid resolution  $\Delta x$  as

$$\Delta t = \text{CFL} \cdot \frac{\Delta x}{\|\mathbf{u}\|_{L^\infty(\Omega)}},$$

to ensure that the interface is not moved by more than  $\text{CFL} \cdot \Delta x$  at each iteration. For the two-dimensional example, to ensure that the interface remains highly resolved along all the simulation, we enforce that it is surrounded by a band of finest cells of half-width  $\max(1, \text{CFL}) \cdot \Delta x$ .

In all convergence tables, the notation  $a : b$  in the levels column stands for  $\min_{\text{level}} = a$  and  $\max_{\text{level}} = b$  and we denote our coupled method by the abbreviation CLSRM (Coupled level-set Reference Map method).

##### 4.1. Translation

We start our analysis with the simplest displacement, a translation at constant speed associated with the velocity field

$$\mathbf{u}(x, y) = \left( \frac{1}{4T}, -\frac{3}{4T} \right). \tag{18}$$

Since  $\mathbf{u}$  has rectilinear characteristic lines (as Fig. 5 illustrates), the exact solution for each component of the reference map is a first-order polynomial. Considering that our advection scheme is second-order accurate, we expect the solution from the reference map method to be exact, and in particular to be the identity at the final time step. Based on our criterion (11), we expect no restarting to be performed and therefore the final level set should be exactly the initial one. On the other hand, with the standard level-set method, we expect second-order convergence as it was observed in [33]. We take  $\text{CFL} = 5$ , and test both methods on uniform and adaptive grids. The results in Table 1 and 2 confirm our expectations. The error on the distance to the interface for the CLSRM, is only third-order accurate because of the reconstruction of  $\delta(\mathbf{x})$  which employs third-order interpolations.

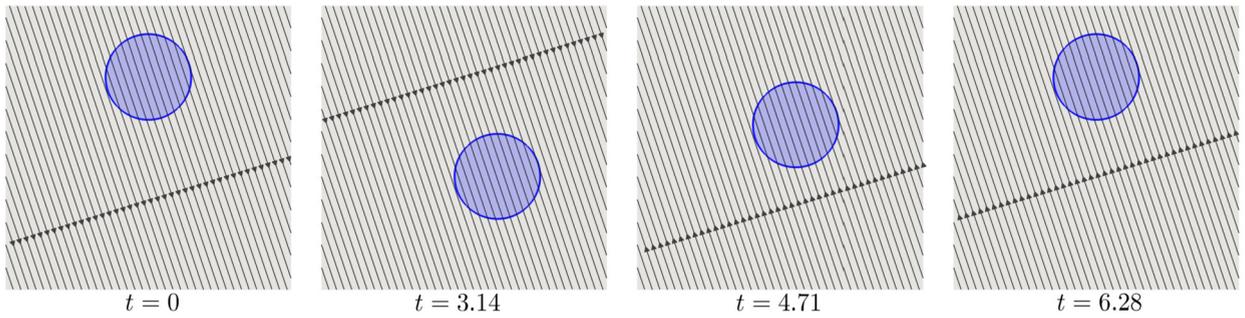


Fig. 5. Translation test, geometry (in blue) and velocity stream lines (in black).

**Table 1**  
Error table for the translation Example 4.1 on uniform grids with CFL = 5.0.

	Levels	$d_{L^\infty}(\phi)$	Order	$e_{L^\infty}(\phi)$	Order	$d_{L^1}(\phi)$	Order	$d_{L^2}(\phi)$	Order	Mass loss (%)
CLSRM	4:4	$5.69 \cdot 10^{-4}$	–	$2.22 \cdot 10^{-16}$	–	$3.38 \cdot 10^{-4}$	–	$4.34 \cdot 10^{-4}$	–	$8.09 \cdot 10^{-14}$
	5:5	$7.51 \cdot 10^{-5}$	2.92	$2.22 \cdot 10^{-16}$	–	$3.62 \cdot 10^{-5}$	3.22	$4.91 \cdot 10^{-5}$	3.14	$5.68 \cdot 10^{-14}$
	6:6	$1.10 \cdot 10^{-5}$	2.77	$2.50 \cdot 10^{-16}$	–	$5.90 \cdot 10^{-6}$	2.62	$7.61 \cdot 10^{-6}$	2.69	$3.93 \cdot 10^{-14}$
	7:7	$1.32 \cdot 10^{-6}$	3.06	$3.33 \cdot 10^{-16}$	–	$4.67 \cdot 10^{-7}$	3.66	$6.31 \cdot 10^{-7}$	3.59	$1.18 \cdot 10^{-13}$
	8:8	$1.62 \cdot 10^{-7}$	3.02	$6.94 \cdot 10^{-16}$	–	$7.45 \cdot 10^{-8}$	2.65	$9.64 \cdot 10^{-8}$	2.71	$3.93 \cdot 10^{-14}$
	9:9	$2.42 \cdot 10^{-8}$	2.74	$1.03 \cdot 10^{-15}$	–	$9.74 \cdot 10^{-9}$	2.94	$1.30 \cdot 10^{-8}$	2.90	$1.57 \cdot 10^{-13}$
10:10	$3.04 \cdot 10^{-9}$	2.99	$1.33 \cdot 10^{-15}$	–	$1.14 \cdot 10^{-9}$	3.10	$1.53 \cdot 10^{-9}$	3.08	$3.93 \cdot 10^{-14}$	
level-set	4:4	$6.89 \cdot 10^{-3}$	–	$9.01 \cdot 10^{-3}$	–	$5.48 \cdot 10^{-3}$	–	$5.84 \cdot 10^{-3}$	–	$8.03 \cdot 10^0$
	5:5	$1.56 \cdot 10^{-3}$	2.14	$1.73 \cdot 10^{-3}$	2.38	$1.11 \cdot 10^{-3}$	2.31	$1.20 \cdot 10^{-3}$	2.28	$1.58 \cdot 10^0$
	6:6	$4.03 \cdot 10^{-4}$	1.95	$4.25 \cdot 10^{-4}$	2.02	$2.76 \cdot 10^{-4}$	2.00	$3.00 \cdot 10^{-4}$	2.00	$3.97 \cdot 10^{-1}$
	7:7	$9.63 \cdot 10^{-5}$	2.07	$9.99 \cdot 10^{-5}$	2.09	$6.62 \cdot 10^{-5}$	2.06	$7.20 \cdot 10^{-5}$	2.06	$9.59 \cdot 10^{-2}$
	8:8	$2.36 \cdot 10^{-5}$	2.03	$2.41 \cdot 10^{-5}$	2.05	$1.58 \cdot 10^{-5}$	2.06	$1.73 \cdot 10^{-5}$	2.06	$2.30 \cdot 10^{-2}$
	9:9	$5.81 \cdot 10^{-6}$	2.02	$5.82 \cdot 10^{-6}$	2.05	$3.90 \cdot 10^{-6}$	2.02	$4.26 \cdot 10^{-6}$	2.02	$5.68 \cdot 10^{-3}$
10:10	$1.44 \cdot 10^{-6}$	2.01	$1.45 \cdot 10^{-6}$	2.01	$9.65 \cdot 10^{-7}$	2.01	$1.06 \cdot 10^{-6}$	2.01	$1.41 \cdot 10^{-3}$	

**Table 2**  
Error table for the translation Example 4.1 on adaptive mesh with CFL = 5.0.

	Levels	$d_{L^\infty}(\phi)$	Order	$e_{L^\infty}(\phi)$	Order	$d_{L^1}(\phi)$	Order	$d_{L^2}(\phi)$	Order	Mass loss (%)
CLSRM	1:4	$5.69 \cdot 10^{-4}$	–	$2.22 \cdot 10^{-16}$	–	$3.39 \cdot 10^{-4}$	–	$4.34 \cdot 10^{-4}$	–	$8.07 \cdot 10^{-14}$
	2:5	$7.51 \cdot 10^{-5}$	2.92	$2.22 \cdot 10^{-16}$	–	$3.62 \cdot 10^{-5}$	3.22	$4.91 \cdot 10^{-5}$	3.14	$4.86 \cdot 10^{-14}$
	3:6	$1.10 \cdot 10^{-5}$	2.77	$2.50 \cdot 10^{-16}$	–	$5.90 \cdot 10^{-6}$	2.62	$7.61 \cdot 10^{-6}$	2.69	$5.90 \cdot 10^{-14}$
	4:7	$1.32 \cdot 10^{-6}$	3.06	$3.33 \cdot 10^{-16}$	–	$4.67 \cdot 10^{-7}$	3.67	$6.31 \cdot 10^{-7}$	3.59	$9.82 \cdot 10^{-14}$
	5:8	$1.62 \cdot 10^{-7}$	3.02	$6.94 \cdot 10^{-16}$	–	$7.45 \cdot 10^{-8}$	2.65	$9.64 \cdot 10^{-8}$	2.71	$1.96 \cdot 10^{-14}$
	6:9	$2.42 \cdot 10^{-8}$	2.74	$8.05 \cdot 10^{-16}$	–	$9.74 \cdot 10^{-9}$	2.94	$1.30 \cdot 10^{-8}$	2.90	$1.77 \cdot 10^{-13}$
	7:10	$3.04 \cdot 10^{-9}$	2.99	$1.14 \cdot 10^{-15}$	–	$1.14 \cdot 10^{-9}$	3.10	$1.53 \cdot 10^{-9}$	3.08	$5.89 \cdot 10^{-14}$
	8:11	$3.77 \cdot 10^{-10}$	3.01	$2.66 \cdot 10^{-15}$	–	$1.44 \cdot 10^{-10}$	2.98	$1.91 \cdot 10^{-10}$	3.00	$5.10 \cdot 10^{-13}$
	9:12	$4.77 \cdot 10^{-11}$	2.98	$3.61 \cdot 10^{-15}$	–	$1.73 \cdot 10^{-11}$	3.06	$2.32 \cdot 10^{-11}$	3.04	$9.22 \cdot 10^{-13}$
	10:13	$5.98 \cdot 10^{-12}$	3.00	$9.35 \cdot 10^{-15}$	–	$2.23 \cdot 10^{-12}$	2.95	$2.99 \cdot 10^{-12}$	2.96	$1.83 \cdot 10^{-12}$
level-set	1:4	$6.89 \cdot 10^{-3}$	–	$9.01 \cdot 10^{-3}$	–	$5.47 \cdot 10^{-3}$	–	$5.83 \cdot 10^{-3}$	–	$8.02 \cdot 10^0$
	2:5	$1.56 \cdot 10^{-3}$	2.15	$1.73 \cdot 10^{-3}$	2.38	$1.10 \cdot 10^{-3}$	2.32	$1.19 \cdot 10^{-3}$	2.29	$1.56 \cdot 10^0$
	3:6	$4.01 \cdot 10^{-4}$	1.96	$4.24 \cdot 10^{-4}$	2.03	$2.76 \cdot 10^{-4}$	1.99	$2.99 \cdot 10^{-4}$	1.99	$3.96 \cdot 10^{-1}$
	4:7	$9.60 \cdot 10^{-5}$	2.06	$9.94 \cdot 10^{-5}$	2.09	$6.61 \cdot 10^{-5}$	2.06	$7.20 \cdot 10^{-5}$	2.06	$9.57 \cdot 10^{-2}$
	5:8	$2.39 \cdot 10^{-5}$	2.01	$2.44 \cdot 10^{-5}$	2.03	$1.58 \cdot 10^{-5}$	2.06	$1.73 \cdot 10^{-5}$	2.06	$2.30 \cdot 10^{-2}$
	6:9	$5.96 \cdot 10^{-6}$	2.01	$5.92 \cdot 10^{-6}$	2.04	$3.90 \cdot 10^{-6}$	2.02	$4.27 \cdot 10^{-6}$	2.02	$5.68 \cdot 10^{-3}$
	7:10	$1.46 \cdot 10^{-6}$	2.01	$1.47 \cdot 10^{-6}$	2.01	$9.66 \cdot 10^{-7}$	2.01	$1.06 \cdot 10^{-6}$	2.01	$1.41 \cdot 10^{-3}$
	8:11	$3.65 \cdot 10^{-7}$	2.00	$3.66 \cdot 10^{-7}$	2.01	$2.42 \cdot 10^{-7}$	2.00	$2.65 \cdot 10^{-7}$	2.00	$3.53 \cdot 10^{-4}$
	9:12	$9.12 \cdot 10^{-8}$	2.00	$9.13 \cdot 10^{-8}$	2.00	$6.04 \cdot 10^{-8}$	2.00	$6.63 \cdot 10^{-8}$	2.00	$8.83 \cdot 10^{-5}$
	10:13	$2.28 \cdot 10^{-8}$	2.00	$2.28 \cdot 10^{-8}$	2.00	$1.51 \cdot 10^{-8}$	2.00	$1.66 \cdot 10^{-8}$	2.00	$2.21 \cdot 10^{-5}$

4.2. Rotation

In this second case, we consider a rotation at a constant angular velocity, setting:

$$\mathbf{u}(x, y) = (-y + 0.5, x - 0.5). \tag{19}$$

As for the previous example, based on the restarting criterion (11), we expect no restarting to be performed. For CFL = 5, the results provided in Tables 3 and 4, clearly show that for this example, our method is one order more accurate than

**Table 3**

Error table for the rotation Example 4.2 on uniform grids with CFL = 5.0.

	Levels	$d_{L^\infty}(\phi)$	Order	$e_{L^\infty}(\phi)$	Order	$d_{L^1}(\phi)$	Order	$d_{L^2}(\phi)$	Order	Mass loss (%)
CLSRM	3:3	$1.24 \cdot 10^{-1}$	-inf	$1.86 \cdot 10^{-1}$	-inf	$1.86 \cdot 10^{-2}$	-inf	$4.71 \cdot 10^{-2}$	-inf	$7.63 \cdot 10^{+01}$
	4:4	$2.91 \cdot 10^{-2}$	2.10	$2.94 \cdot 10^{-2}$	2.66	$1.11 \cdot 10^{-2}$	0.75	$1.50 \cdot 10^{-2}$	1.65	$1.44 \cdot 10^{+01}$
	5:5	$3.32 \cdot 10^{-3}$	3.13	$3.40 \cdot 10^{-3}$	3.11	$1.49 \cdot 10^{-3}$	2.89	$1.90 \cdot 10^{-3}$	2.98	$1.66 \cdot 10^{+00}$
	6:6	$4.47 \cdot 10^{-4}$	2.89	$4.54 \cdot 10^{-4}$	2.91	$1.96 \cdot 10^{-4}$	2.92	$2.50 \cdot 10^{-4}$	2.93	$2.16 \cdot 10^{-1}$
	7:7	$5.41 \cdot 10^{-5}$	3.05	$5.46 \cdot 10^{-5}$	3.06	$2.36 \cdot 10^{-5}$	3.05	$3.02 \cdot 10^{-5}$	3.05	$2.67 \cdot 10^{-2}$
	8:8	$6.83 \cdot 10^{-6}$	2.99	$6.77 \cdot 10^{-6}$	3.01	$2.97 \cdot 10^{-6}$	2.99	$3.80 \cdot 10^{-6}$	2.99	$3.32 \cdot 10^{-3}$
	9:9	$8.27 \cdot 10^{-7}$	3.05	$8.27 \cdot 10^{-7}$	3.03	$3.63 \cdot 10^{-7}$	3.03	$4.66 \cdot 10^{-7}$	3.03	$4.13 \cdot 10^{-4}$
	10:10	$1.04 \cdot 10^{-7}$	2.99	$1.04 \cdot 10^{-7}$	3.00	$4.54 \cdot 10^{-8}$	3.00	$5.82 \cdot 10^{-8}$	3.00	$5.17 \cdot 10^{-5}$
level-set	4:4	$4.94 \cdot 10^{-2}$	-	$4.99 \cdot 10^{-2}$	-	$2.12 \cdot 10^{-2}$	-	$2.82 \cdot 10^{-2}$	-	$3.83 \cdot 10^1$
	5:5	$8.22 \cdot 10^{-3}$	2.58	$8.47 \cdot 10^{-3}$	2.55	$5.61 \cdot 10^{-3}$	1.92	$6.07 \cdot 10^{-3}$	2.21	$7.99 \cdot 10^0$
	6:6	$1.73 \cdot 10^{-3}$	2.24	$1.75 \cdot 10^{-3}$	2.26	$1.27 \cdot 10^{-3}$	2.13	$1.35 \cdot 10^{-3}$	2.16	$1.84 \cdot 10^0$
	7:7	$3.84 \cdot 10^{-4}$	2.17	$3.87 \cdot 10^{-4}$	2.18	$3.01 \cdot 10^{-4}$	2.09	$3.16 \cdot 10^{-4}$	2.09	$4.36 \cdot 10^{-1}$
	8:8	$9.27 \cdot 10^{-5}$	2.05	$9.27 \cdot 10^{-5}$	2.06	$7.43 \cdot 10^{-5}$	2.01	$7.78 \cdot 10^{-5}$	2.02	$1.07 \cdot 10^{-1}$
	9:9	$2.38 \cdot 10^{-5}$	1.95	$2.38 \cdot 10^{-5}$	1.95	$1.84 \cdot 10^{-5}$	2.01	$1.93 \cdot 10^{-5}$	2.01	$2.66 \cdot 10^{-2}$

**Table 4**

Error table for the rotation Example 4.2 on adaptive grids with CFL = 5.0.

	Levels	$d_{L^\infty}(\phi)$	Order	$e_{L^\infty}(\phi)$	Order	$d_{L^1}(\phi)$	Order	$d_{L^2}(\phi)$	Order	Mass loss (%)
CLSRM	1:4	$2.74 \cdot 10^{-2}$	-	$2.77 \cdot 10^{-2}$	-	$1.05 \cdot 10^{-2}$	-	$1.42 \cdot 10^{-2}$	-	$1.36 \cdot 10^1$
	2:5	$4.55 \cdot 10^{-3}$	2.59	$4.65 \cdot 10^{-3}$	2.58	$2.01 \cdot 10^{-3}$	2.39	$2.50 \cdot 10^{-3}$	2.50	$2.00 \cdot 10^0$
	3:6	$7.56 \cdot 10^{-4}$	2.59	$7.59 \cdot 10^{-4}$	2.61	$3.48 \cdot 10^{-4}$	2.53	$4.21 \cdot 10^{-4}$	2.58	$2.57 \cdot 10^{-1}$
	4:7	$6.34 \cdot 10^{-5}$	3.58	$6.44 \cdot 10^{-5}$	3.56	$2.78 \cdot 10^{-5}$	3.65	$3.56 \cdot 10^{-5}$	3.57	$3.17 \cdot 10^{-2}$
	5:8	$1.06 \cdot 10^{-5}$	2.58	$1.05 \cdot 10^{-5}$	2.62	$4.92 \cdot 10^{-6}$	2.50	$5.95 \cdot 10^{-6}$	2.58	$3.46 \cdot 10^{-3}$
	6:9	$8.27 \cdot 10^{-7}$	3.68	$8.27 \cdot 10^{-7}$	3.66	$3.63 \cdot 10^{-7}$	3.76	$4.66 \cdot 10^{-7}$	3.67	$4.13 \cdot 10^{-4}$
	7:10	$1.04 \cdot 10^{-7}$	2.99	$1.06 \cdot 10^{-7}$	3.00	$4.54 \cdot 10^{-8}$	3.00	$5.82 \cdot 10^{-8}$	3.00	$5.17 \cdot 10^{-5}$
	8:11	$1.30 \cdot 10^{-8}$	3.00	$1.29 \cdot 10^{-8}$	3.00	$5.67 \cdot 10^{-9}$	3.00	$7.28 \cdot 10^{-9}$	3.00	$6.46 \cdot 10^{-6}$
	9:12	$1.62 \cdot 10^{-9}$	3.00	$1.62 \cdot 10^{-9}$	3.00	$7.09 \cdot 10^{-10}$	3.00	$9.09 \cdot 10^{-10}$	3.00	$8.08 \cdot 10^{-7}$
level-set	1:4	$4.84 \cdot 10^{-2}$	-	$4.88 \cdot 10^{-2}$	-	$2.16 \cdot 10^{-2}$	-	$2.83 \cdot 10^{-2}$	-	$3.88 \cdot 10^1$
	2:5	$8.70 \cdot 10^{-3}$	2.48	$8.71 \cdot 10^{-3}$	2.49	$5.64 \cdot 10^{-3}$	1.93	$6.20 \cdot 10^{-3}$	2.19	$8.04 \cdot 10^0$
	3:6	$2.18 \cdot 10^{-3}$	2.00	$2.21 \cdot 10^{-3}$	1.98	$1.20 \cdot 10^{-3}$	2.23	$1.39 \cdot 10^{-3}$	2.16	$1.73 \cdot 10^0$
	4:7	$4.14 \cdot 10^{-4}$	2.40	$4.15 \cdot 10^{-4}$	2.41	$2.96 \cdot 10^{-4}$	2.02	$3.13 \cdot 10^{-4}$	2.15	$4.30 \cdot 10^{-1}$
	5:8	$9.45 \cdot 10^{-5}$	2.13	$9.48 \cdot 10^{-5}$	2.13	$7.35 \cdot 10^{-5}$	2.01	$7.72 \cdot 10^{-5}$	2.02	$1.07 \cdot 10^{-1}$
	6:9	$2.38 \cdot 10^{-5}$	1.99	$2.38 \cdot 10^{-5}$	1.99	$1.83 \cdot 10^{-5}$	2.01	$1.92 \cdot 10^{-5}$	2.01	$2.65 \cdot 10^{-2}$
	7:10	$5.99 \cdot 10^{-6}$	1.99	$5.99 \cdot 10^{-6}$	1.99	$4.57 \cdot 10^{-6}$	2.00	$4.79 \cdot 10^{-6}$	2.00	$6.64 \cdot 10^{-3}$
	8:11	$1.51 \cdot 10^{-6}$	1.99	$1.51 \cdot 10^{-6}$	1.99	$1.14 \cdot 10^{-6}$	2.00	$1.19 \cdot 10^{-6}$	2.00	$1.66 \cdot 10^{-3}$
	9:12	$3.76 \cdot 10^{-7}$	2.00	$3.76 \cdot 10^{-7}$	2.00	$2.84 \cdot 10^{-7}$	2.00	$2.98 \cdot 10^{-7}$	2.00	$4.13 \cdot 10^{-4}$

the standard level-set method, both on uniform and adaptive grids. At fixed maximum grid level and for either method, the uniform and adaptive errors (in any norm) are almost identical. It illustrates the efficiency of our refinement strategy at reducing the computational cost while preserving the same accuracy. For the rest of the article, we will focus on adaptive grids only.

The third-order convergence of our coupled method can be proven analytically by comparing the computed x-coordinate of the departure point

$$x_d^n = x^{n+1} + \Delta t_n y^{n+1} - \frac{\Delta t_n}{2} - \frac{\Delta t_n^2}{2} x^{n+1} + \frac{\Delta t_n^2}{4},$$

with the exact one, obtained by integrating the x-velocity along the characteristic line and performing one formal Taylor expansion:

$$\begin{aligned} \tilde{x}_d^n &= x^{n+1} - \int_{t_n}^{t_n + \Delta t_n} \left[ \frac{1}{2} - y(s) \right] ds \\ &= x^{n+1} - \frac{\Delta t_n}{2} + \int_{t_n}^{t_n + \Delta t_n} \left[ y^{n+1} + y'(t_{n+1})(s - t_{n+1}) + \frac{y''(t_{n+1})}{2}(s - t_{n+1})^2 + O(\Delta t_n^3) \right] ds, \end{aligned}$$

and using  $y'(t) = x(t) - 1/2$ . This yields:

$$= x^{n+1} + \Delta t_n y^{n+1} - \frac{\Delta t_n}{2} + \frac{\Delta t_n^2}{2} x^{n+1} - \frac{\Delta t_n^2}{4} + O(\Delta t_n^3)$$

**Table 5**  
Error table for the rotation Example 4.2 on adaptive grids with CFL = 0.8.

	Levels	$d_{L^\infty}(\phi)$	Order	$e_{L^\infty}(\phi)$	Order	$d_{L^1}(\phi)$	Order	$d_{L^2}(\phi)$	Order	Mass loss (%)
CLSRM	1:4	$7.24 \cdot 10^{-4}$	–	$1.67 \cdot 10^{-4}$	–	$3.89 \cdot 10^{-4}$	–	$4.77 \cdot 10^{-4}$	–	$6.32 \cdot 10^{-2}$
	2:5	$1.25 \cdot 10^{-4}$	2.53	$5.31 \cdot 10^{-5}$	1.66	$4.94 \cdot 10^{-5}$	2.97	$6.27 \cdot 10^{-5}$	2.92	$8.48 \cdot 10^{-3}$
	3:6	$2.21 \cdot 10^{-5}$	2.50	$1.24 \cdot 10^{-5}$	2.09	$9.11 \cdot 10^{-6}$	2.44	$1.15 \cdot 10^{-5}$	2.44	$1.06 \cdot 10^{-3}$
	4:7	$2.62 \cdot 10^{-6}$	3.07	$1.46 \cdot 10^{-6}$	3.08	$9.58 \cdot 10^{-7}$	3.24	$1.19 \cdot 10^{-6}$	3.27	$1.19 \cdot 10^{-4}$
	5:8	$3.27 \cdot 10^{-7}$	3.00	$1.77 \cdot 10^{-7}$	3.04	$1.24 \cdot 10^{-7}$	2.94	$1.57 \cdot 10^{-7}$	2.92	$1.42 \cdot 10^{-5}$
	6:9	$2.72 \cdot 10^{-8}$	3.58	$3.39 \cdot 10^{-9}$	5.70	$1.09 \cdot 10^{-8}$	3.50	$1.40 \cdot 10^{-8}$	3.48	$1.69 \cdot 10^{-6}$
	7:10	$3.42 \cdot 10^{-9}$	2.99	$4.24 \cdot 10^{-10}$	2.99	$1.29 \cdot 10^{-9}$	3.08	$1.66 \cdot 10^{-9}$	3.07	$2.11 \cdot 10^{-7}$
	8:11	$6.80 \cdot 10^{-10}$	2.32	$3.33 \cdot 10^{-10}$	0.34	$2.38 \cdot 10^{-10}$	2.43	$3.01 \cdot 10^{-10}$	2.46	$2.66 \cdot 10^{-8}$
	9:12	$5.35 \cdot 10^{-11}$	3.66	$6.62 \cdot 10^{-12}$	5.65	$1.96 \cdot 10^{-11}$	3.60	$2.52 \cdot 10^{-11}$	3.57	$3.30 \cdot 10^{-9}$
level-set	2:5	$3.80 \cdot 10^{-2}$	–	$3.80 \cdot 10^{-2}$	–	$1.22 \cdot 10^{-2}$	–	$1.90 \cdot 10^{-2}$	–	$3.56 \cdot 10^1$
	3:6	$8.62 \cdot 10^{-3}$	2.13	$8.63 \cdot 10^{-3}$	2.13	$5.14 \cdot 10^{-3}$	1.25	$5.93 \cdot 10^{-3}$	1.68	$8.68 \cdot 10^0$
	4:7	$2.18 \cdot 10^{-3}$	1.98	$2.18 \cdot 10^{-3}$	1.98	$1.45 \cdot 10^{-3}$	1.81	$1.60 \cdot 10^{-3}$	1.89	$2.21 \cdot 10^0$
	5:8	$5.67 \cdot 10^{-4}$	1.94	$5.67 \cdot 10^{-4}$	1.94	$3.89 \cdot 10^{-4}$	1.90	$4.19 \cdot 10^{-4}$	1.93	$5.69 \cdot 10^{-1}$
	6:9	$1.43 \cdot 10^{-4}$	1.98	$1.43 \cdot 10^{-4}$	1.98	$9.89 \cdot 10^{-5}$	1.97	$1.05 \cdot 10^{-4}$	1.98	$1.43 \cdot 10^{-1}$
	7:10	$3.57 \cdot 10^{-5}$	2.00	$3.57 \cdot 10^{-5}$	2.00	$2.46 \cdot 10^{-5}$	2.00	$2.64 \cdot 10^{-5}$	2.00	$3.59 \cdot 10^{-2}$
	8:11	$8.99 \cdot 10^{-6}$	1.98	$8.99 \cdot 10^{-6}$	1.98	$6.20 \cdot 10^{-6}$	1.99	$6.65 \cdot 10^{-6}$	1.99	$9.01 \cdot 10^{-3}$
	9:12	$2.24 \cdot 10^{-6}$	2.00	$2.24 \cdot 10^{-6}$	2.00	$1.55 \cdot 10^{-6}$	1.99	$1.66 \cdot 10^{-6}$	1.99	$2.25 \cdot 10^{-3}$

**Table 6**  
Convergence of the curvature for the rotation test 4.2 with CFL = 5.0.

	Levels	$e_{L^\infty}(\kappa)$	Order	$e_{L^1}(\kappa)$	Order	$e_{L^2}(\kappa)$	Order
CLSRM	1:4	$2.33 \cdot 10^0$	–	$9.41 \cdot 10^{-1}$	–	$1.20 \cdot 10^0$	–
	2:5	$4.62 \cdot 10^{-1}$	2.34	$1.98 \cdot 10^{-1}$	2.24	$2.41 \cdot 10^{-1}$	2.31
	3:6	$9.90 \cdot 10^{-2}$	2.22	$4.85 \cdot 10^{-2}$	2.03	$5.71 \cdot 10^{-2}$	2.08
	4:7	$2.38 \cdot 10^{-2}$	2.05	$1.22 \cdot 10^{-2}$	1.98	$1.42 \cdot 10^{-2}$	2.00
	5:8	$5.77 \cdot 10^{-3}$	2.04	$3.05 \cdot 10^{-3}$	2.00	$3.53 \cdot 10^{-3}$	2.01
	6:9	$1.42 \cdot 10^{-3}$	2.01	$7.58 \cdot 10^{-4}$	2.00	$8.79 \cdot 10^{-4}$	2.00
	7:10	$3.55 \cdot 10^{-4}$	2.00	$1.90 \cdot 10^{-4}$	1.99	$2.20 \cdot 10^{-4}$	1.99
	8:11	$8.85 \cdot 10^{-5}$	2.00	$4.75 \cdot 10^{-5}$	2.00	$5.49 \cdot 10^{-5}$	2.00
	9:12	$2.21 \cdot 10^{-5}$	2.00	$1.18 \cdot 10^{-5}$	2.00	$1.37 \cdot 10^{-5}$	2.00
level-set	1:4	$3.72 \cdot 10^0$	–	$1.63 \cdot 10^0$	–	$2.08 \cdot 10^0$	–
	2:5	$6.49 \cdot 10^{-1}$	2.52	$3.09 \cdot 10^{-1}$	2.39	$3.82 \cdot 10^{-1}$	2.44
	3:6	$2.34 \cdot 10^{-1}$	1.46	$8.17 \cdot 10^{-2}$	1.92	$9.92 \cdot 10^{-2}$	1.94
	4:7	$1.19 \cdot 10^{-1}$	0.97	$2.86 \cdot 10^{-2}$	1.51	$3.89 \cdot 10^{-2}$	1.34
	5:8	$6.73 \cdot 10^{-2}$	0.82	$9.30 \cdot 10^{-3}$	1.62	$1.35 \cdot 10^{-2}$	1.53
	6:9	$2.46 \cdot 10^{-2}$	1.45	$3.48 \cdot 10^{-3}$	1.41	$4.96 \cdot 10^{-3}$	1.44
	7:10	$1.41 \cdot 10^{-2}$	0.80	$1.61 \cdot 10^{-3}$	1.10	$2.31 \cdot 10^{-3}$	1.09
	8:11	$7.79 \cdot 10^{-2}$	0.85	$8.97 \cdot 10^{-4}$	0.85	$1.23 \cdot 10^{-3}$	0.91
	9:12	$5.70 \cdot 10^{-2}$	0.45	$4.28 \cdot 10^{-4}$	1.06	$6.18 \cdot 10^{-3}$	0.99

$$= x_d^n + O(\Delta t_n^3).$$

Since the interpolations are also third-order accurate, the solution retains the same accuracy. If restarting had been performed, the accuracy would have been bounded by that of the reinitialization and would have dropped to second order. For CFL = 0.8, the orders of convergence on adaptive grids (given in Table 5) are again 3 for the CLSRM and 2 for the level set. The error of the coupled method is systematically at least an order of magnitude smaller than for CFL = 5. We point out that we could not predict it since the truncation error of the semi-Lagrangian method is  $O(\Delta t^2 + \frac{\Delta x^3}{\Delta t})$ , and therefore can potentially increase as the time step is reduced (keeping the spatial resolution fixed). For the level-set method, the error is systematically increased by a factor close to 6. This is due to the reinitialization procedure, which at each time step introduces a  $O(\Delta x^3)$  error, and therefore a total reinitialization error behaving as  $\sim \frac{\|u\|_\infty \Delta x^2}{T \cdot \text{CFL}}$ . Going from CFL = 5 to CFL = 0.8, this total error is multiplied by  $\frac{5}{0.8} = 6.25$ .

Convergence results for the curvature are presented in Table 6. We observe that with our coupled method the curvature converges with second-order accuracy while with the level-set method it only is first order. It is unsurprising since the interface representation is one order more accurate with our method.

### 4.3. Vortex

In this test, we consider the divergence-free velocity field (introduced in [4] for the first time) illustrated on Fig. 6 and defined by

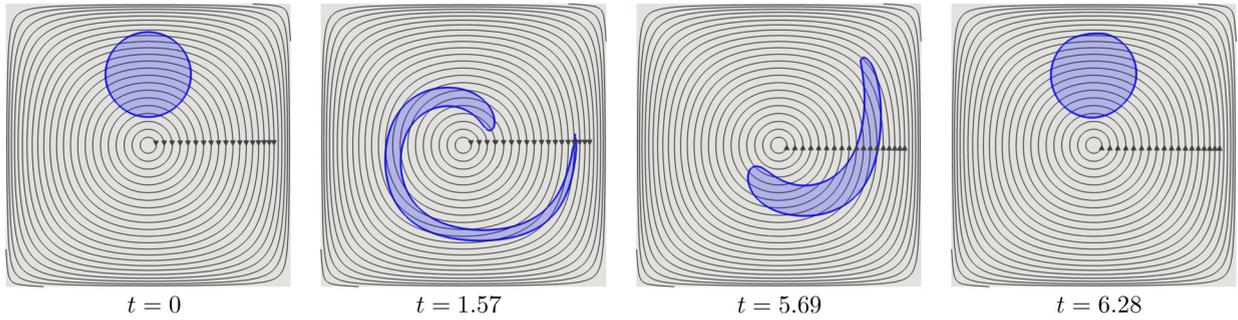


Fig. 6. Vortex test, geometry (in blue) and velocity stream lines (in black).

Table 7

Error table for the vortex Example 4.3 with CFL = 5.0 and using the systematic restarting.

	Levels	$d_{L^\infty}(\phi)$	Order	$e_{L^\infty}(\phi)$	Order	$d_{L^1}(\phi)$	Order	$d_{L^2}(\phi)$	Order	Mass loss (%)
CLSRM	3:6	$1.10 \cdot 10^{-1}$	–	$1.10 \cdot 10^{-1}$	–	$5.34 \cdot 10^{-4}$	–	$2.71 \cdot 10^{-3}$	–	$8.38 \cdot 10^1$
	4:7	$7.15 \cdot 10^{-2}$	0.63	$7.15 \cdot 10^{-2}$	0.63	$1.01 \cdot 10^{-3}$	–.92	$2.70 \cdot 10^{-3}$	0.00	$2.42 \cdot 10^1$
	5:8	$2.77 \cdot 10^{-2}$	1.37	$2.77 \cdot 10^{-2}$	1.37	$6.45 \cdot 10^{-4}$	0.65	$1.10 \cdot 10^{-3}$	1.30	$6.17 \cdot 10^0$
	6:9	$7.42 \cdot 10^{-3}$	1.90	$7.43 \cdot 10^{-3}$	1.90	$1.69 \cdot 10^{-4}$	1.93	$3.65 \cdot 10^{-4}$	1.59	$1.03 \cdot 10^0$
	7:10	$3.20 \cdot 10^{-3}$	1.21	$3.20 \cdot 10^{-3}$	1.21	$5.25 \cdot 10^{-5}$	1.69	$1.59 \cdot 10^{-4}$	1.20	$3.12 \cdot 10^{-1}$
	8:11	$1.62 \cdot 10^{-3}$	0.98	$1.62 \cdot 10^{-3}$	0.98	$1.76 \cdot 10^{-5}$	1.57	$6.63 \cdot 10^{-5}$	1.26	$1.13 \cdot 10^{-1}$
	9:12	$8.22 \cdot 10^{-4}$	0.98	$8.22 \cdot 10^{-4}$	0.98	$5.18 \cdot 10^{-6}$	1.77	$2.20 \cdot 10^{-5}$	1.59	$4.12 \cdot 10^{-2}$

Table 8

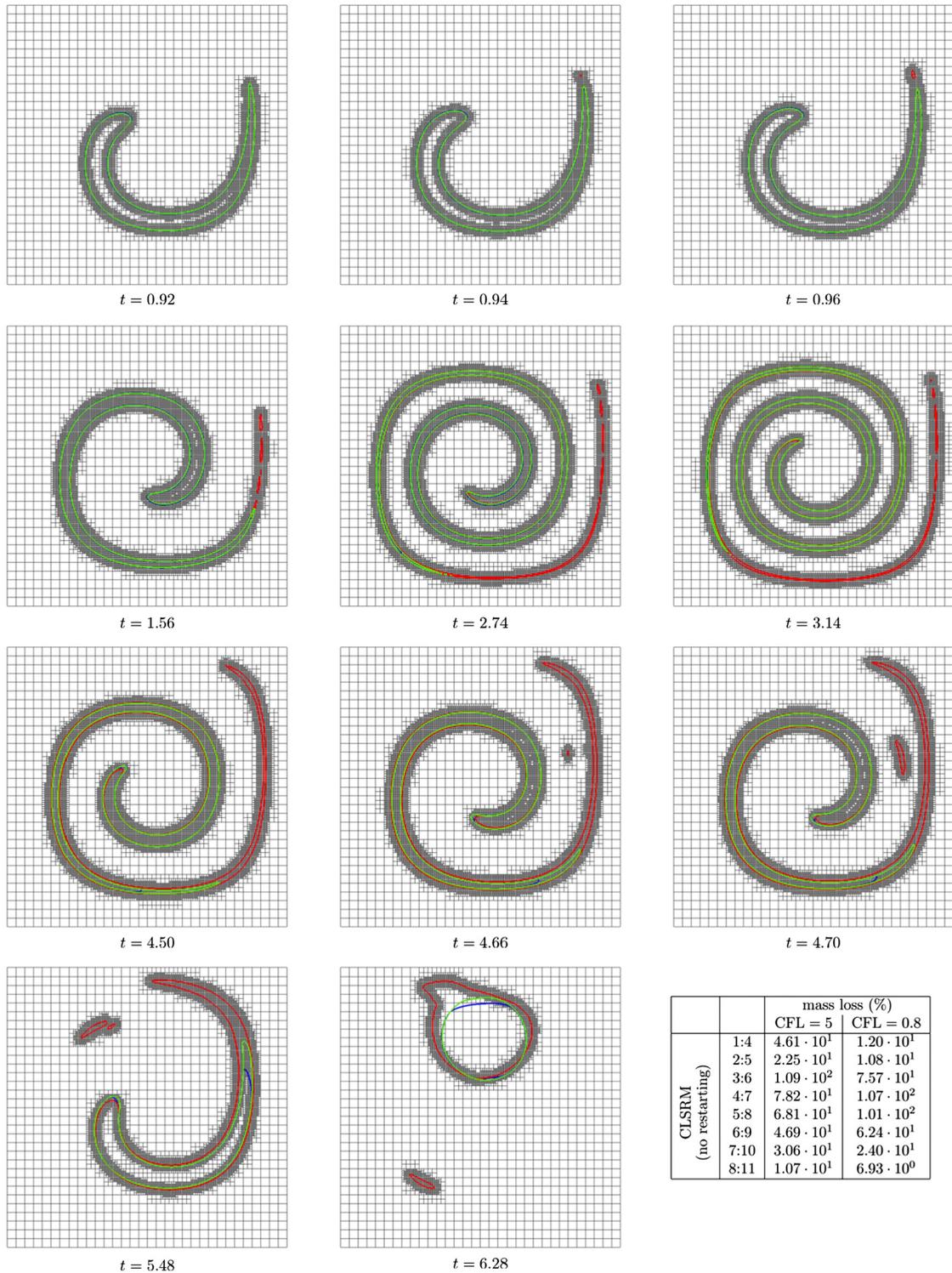
Error table for the vortex Example 4.3 with CFL = 5.0 and using selective restarting.

	Levels	$d_{L^\infty}(\phi)$	Order	$e_{L^\infty}(\phi)$	Order	$d_{L^1}(\phi)$	Order	$d_{L^2}(\phi)$	Order	Mass loss (%)
CLSRM	3:6	$1.21 \cdot 10^{-1}$	–	$1.46 \cdot 10^{-1}$	–	$5.52 \cdot 10^{-4}$	–	$2.58 \cdot 10^{-3}$	–	$6.91 \cdot 10^1$
	4:7	$6.30 \cdot 10^{-2}$	0.94	$6.28 \cdot 10^{-2}$	1.21	$1.31 \cdot 10^{-3}$	–1.25	$2.90 \cdot 10^{-3}$	–.17	$1.86 \cdot 10^1$
	5:8	$1.53 \cdot 10^{-2}$	2.04	$1.55 \cdot 10^{-2}$	2.01	$5.86 \cdot 10^{-4}$	1.16	$9.24 \cdot 10^{-4}$	1.65	$2.93 \cdot 10^0$
	6:9	$3.04 \cdot 10^{-3}$	2.33	$3.14 \cdot 10^{-3}$	2.31	$1.33 \cdot 10^{-4}$	2.14	$3.35 \cdot 10^{-4}$	1.46	$2.28 \cdot 10^{-1}$
	7:10	$1.02 \cdot 10^{-3}$	1.58	$1.02 \cdot 10^{-3}$	1.62	$2.96 \cdot 10^{-5}$	2.16	$1.19 \cdot 10^{-4}$	1.49	$4.68 \cdot 10^{-2}$
	8:11	$2.01 \cdot 10^{-4}$	2.34	$2.05 \cdot 10^{-4}$	2.32	$6.20 \cdot 10^{-6}$	2.26	$2.31 \cdot 10^{-5}$	2.37	$5.00 \cdot 10^{-3}$
	9:12	$4.71 \cdot 10^{-5}$	2.10	$4.78 \cdot 10^{-5}$	2.10	$1.41 \cdot 10^{-6}$	2.13	$5.19 \cdot 10^{-6}$	2.15	$2.37 \cdot 10^{-4}$
level-set	3:6	$1.10 \cdot 10^{-1}$	–	$1.10 \cdot 10^{-1}$	–	$5.32 \cdot 10^{-4}$	–	$2.70 \cdot 10^{-3}$	–	$8.39 \cdot 10^1$
	4:7	$7.08 \cdot 10^{-2}$	0.63	$7.08 \cdot 10^{-2}$	0.63	$1.02 \cdot 10^{-3}$	–.94	$2.73 \cdot 10^{-3}$	–.02	$2.39 \cdot 10^1$
	5:8	$2.77 \cdot 10^{-2}$	1.35	$2.77 \cdot 10^{-2}$	1.35	$6.47 \cdot 10^{-4}$	0.66	$1.10 \cdot 10^{-3}$	1.31	$6.18 \cdot 10^0$
	6:9	$7.44 \cdot 10^{-3}$	1.90	$7.44 \cdot 10^{-3}$	1.90	$1.69 \cdot 10^{-4}$	1.93	$3.65 \cdot 10^{-4}$	1.59	$1.04 \cdot 10^0$
	7:10	$3.19 \cdot 10^{-3}$	1.22	$3.19 \cdot 10^{-3}$	1.22	$5.24 \cdot 10^{-5}$	1.69	$1.58 \cdot 10^{-4}$	1.21	$3.11 \cdot 10^{-1}$
	8:11	$1.62 \cdot 10^{-3}$	0.98	$1.62 \cdot 10^{-3}$	0.98	$1.76 \cdot 10^{-5}$	1.57	$6.63 \cdot 10^{-5}$	1.26	$1.13 \cdot 10^{-1}$
	9:12	$8.22 \cdot 10^{-4}$	0.98	$8.22 \cdot 10^{-4}$	0.98	$5.18 \cdot 10^{-6}$	1.77	$2.20 \cdot 10^{-5}$	1.59	$4.12 \cdot 10^{-2}$

$$\mathbf{u}(x, y) = \left( -\sin^2(\pi x) \sin(2\pi y), \sin^2(\pi y) \sin(2\pi x) \right). \tag{20}$$

First, we would like to justify the necessity of restarting. Fig. 7 illustrates the typical behavior of our method when the restarting procedure is deactivated. Artificial nucleations occur during the simulation, which we understand as a consequence of the non-bijectionality of the reference map in coarsely resolved computational areas close to the interface. These artifacts have catastrophic consequences on the rest of the simulation and thus on the convergence of all interface-related quantities, such as the mass loss (see bottom right of Fig. 7). We conclude that the restarting procedure must be activated to ensure convergence in the general case.

Table 7 and 8 compare the effects of systematically restarting versus restarting only when the criterion (11) is met and using CFL = 5. With the systematic restarting, our method produces (almost) the same errors as the level-set method, and converges with first-order accuracy. Using the selective restarting, we achieve second order. Using selective restarting and reducing the time step to CFL = 0.8 (see Table 9), the order of convergence of both methods remains unchanged. The errors for the coupled method are almost unaffected, which probably indicates that in this regime the spatial error dominates. On the other hand, the errors of the level set slightly increase, which we suspect to be due to the increased number of performed reinitializations.

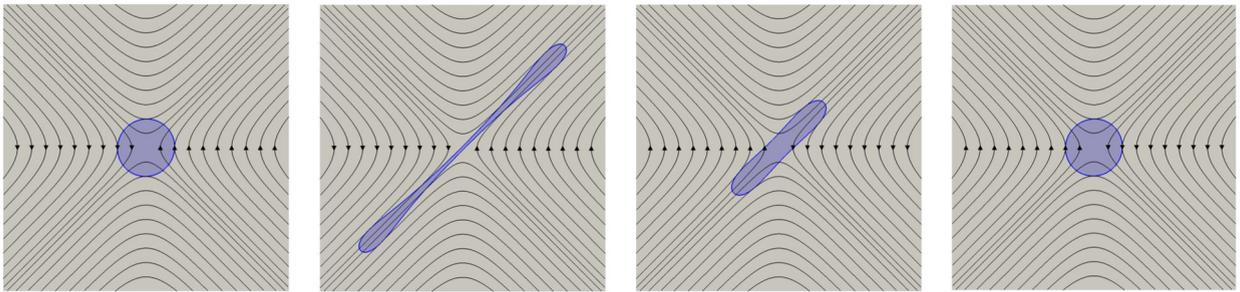


**Fig. 7.** (Vortex Example 4.3) Without restarting (red), our method produces artificial contours that dramatically degrade the convergence, as the mass loss illustrates (bottom right) for different CFL values. These numerical artifacts are not present when the restarting is used (green), or with the standard level-set method (blue). The mesh used for the simulation without restarting is represented in black.

**Table 9**

Error table for the vortex Example 4.3 with CFL = 0.8 and using selective restarting.

	Levels	$d_{L^\infty}(\phi)$	Order	$e_{L^\infty}(\phi)$	Order	$d_{L^1}(\phi)$	Order	$d_{L^2}(\phi)$	Order	Mass loss (%)
CLSRM	3:6	$1.20 \cdot 10^{-1}$	–	$1.12 \cdot 10^{-1}$	–	$2.02 \cdot 10^{-3}$	–	$4.83 \cdot 10^{-3}$	–	$5.08 \cdot 10^1$
	4:7	$2.36 \cdot 10^{-2}$	2.35	$2.41 \cdot 10^{-2}$	2.32	$1.04 \cdot 10^{-3}$	0.95	$1.78 \cdot 10^{-3}$	1.44	$5.38 \cdot 10^0$
	5:8	$6.83 \cdot 10^{-3}$	1.79	$6.83 \cdot 10^{-2}$	1.82	$4.95 \cdot 10^{-4}$	1.07	$7.86 \cdot 10^{-4}$	1.17	$8.40 \cdot 10^{-1}$
	6:9	$1.98 \cdot 10^{-3}$	1.78	$3.33 \cdot 10^{-3}$	1.03	$1.20 \cdot 10^{-4}$	2.04	$2.41 \cdot 10^{-4}$	1.70	$1.44 \cdot 10^{-1}$
	7:10	$5.31 \cdot 10^{-4}$	1.90	$5.31 \cdot 10^{-4}$	2.64	$3.07 \cdot 10^{-5}$	1.96	$7.42 \cdot 10^{-5}$	1.70	$2.28 \cdot 10^{-2}$
	8:11	$1.31 \cdot 10^{-4}$	2.01	$1.32 \cdot 10^{-4}$	2.00	$1.02 \cdot 10^{-5}$	1.58	$1.94 \cdot 10^{-5}$	1.93	$3.83 \cdot 10^{-3}$
	9:12	$3.52 \cdot 10^{-5}$	1.90	$3.57 \cdot 10^{-5}$	1.88	$5.06 \cdot 10^{-6}$	1.01	$7.51 \cdot 10^{-6}$	1.37	$9.90 \cdot 10^{-4}$
level-set	4:7	$1.05 \cdot 10^{-1}$	–	$1.05 \cdot 10^{-1}$	–	$3.07 \cdot 10^{-4}$	–	$1.56 \cdot 10^{-3}$	–	$4.86 \cdot 10^1$
	5:8	$3.94 \cdot 10^{-2}$	1.42	$3.94 \cdot 10^{-2}$	1.42	$7.46 \cdot 10^{-4}$	–1.28	$1.28 \cdot 10^{-3}$	0.28	$1.32 \cdot 10^1$
	6:9	$1.68 \cdot 10^{-2}$	1.23	$1.68 \cdot 10^{-2}$	1.23	$3.12 \cdot 10^{-4}$	1.26	$4.63 \cdot 10^{-4}$	1.47	$3.92 \cdot 10^0$
	7:10	$8.50 \cdot 10^{-3}$	0.98	$8.50 \cdot 10^{-3}$	0.98	$1.02 \cdot 10^{-4}$	1.62	$1.69 \cdot 10^{-4}$	1.46	$1.38 \cdot 10^0$
	8:11	$4.30 \cdot 10^{-3}$	0.98	$4.30 \cdot 10^{-3}$	0.98	$3.09 \cdot 10^{-5}$	1.72	$6.23 \cdot 10^{-5}$	1.44	$4.90 \cdot 10^{-1}$
	9:12	$2.16 \cdot 10^{-3}$	1.00	$2.16 \cdot 10^{-3}$	1.00	$1.05 \cdot 10^{-5}$	1.55	$2.88 \cdot 10^{-5}$	1.11	$1.73 \cdot 10^{-1}$



**Fig. 8.** Stretching test, geometry (in blue) and velocity stream lines (in black).

**Table 10**

Error table for the stretching Example 4.4 with CFL = 5.0.

	Levels	$d_{L^\infty}(\phi)$	Order	$e_{L^\infty}(\phi)$	Order	$d_{L^1}(\phi)$	Order	$d_{L^2}(\phi)$	Order	Mass loss (%)
CLSRM	1:4	$5.39 \cdot 10^{-2}$	–	$7.98 \cdot 10^{-2}$	–	$1.05 \cdot 10^{-2}$	–	$2.23 \cdot 10^{-2}$	–	$6.05 \cdot 10^1$
	2:5	$3.60 \cdot 10^{-2}$	0.58	$7.60 \cdot 10^{-2}$	0.07	$2.34 \cdot 10^{-3}$	2.16	$4.03 \cdot 10^{-3}$	2.46	$3.13 \cdot 10^1$
	3:6	$7.02 \cdot 10^{-3}$	2.35	$2.19 \cdot 10^{-2}$	1.79	$1.25 \cdot 10^{-3}$	0.89	$2.29 \cdot 10^{-3}$	0.81	$4.01 \cdot 10^0$
	4:7	$1.72 \cdot 10^{-4}$	2.02	$9.63 \cdot 10^{-3}$	1.18	$1.94 \cdot 10^{-4}$	2.69	$3.51 \cdot 10^{-4}$	2.70	$7.83 \cdot 10^{-1}$
	5:8	$1.23 \cdot 10^{-5}$	3.80	$4.13 \cdot 10^{-3}$	1.22	$2.62 \cdot 10^{-5}$	2.89	$4.08 \cdot 10^{-5}$	3.10	$1.18 \cdot 10^{-1}$
	6:9	$2.24 \cdot 10^{-5}$	2.46	$6.27 \cdot 10^{-5}$	6.04	$3.38 \cdot 10^{-6}$	2.95	$5.50 \cdot 10^{-6}$	2.89	$1.09 \cdot 10^{-2}$
	7:10	$3.36 \cdot 10^{-6}$	2.73	$2.16 \cdot 10^{-5}$	1.53	$3.83 \cdot 10^{-7}$	3.14	$6.69 \cdot 10^{-7}$	3.04	$1.23 \cdot 10^{-3}$
	8:11	$7.24 \cdot 10^{-7}$	2.21	$5.53 \cdot 10^{-6}$	1.96	$4.74 \cdot 10^{-8}$	3.01	$1.08 \cdot 10^{-7}$	2.62	$1.06 \cdot 10^{-4}$
level-set	1:4	$9.72 \cdot 10^{-2}$	–	$9.73 \cdot 10^{-2}$	–	$1.27 \cdot 10^{-2}$	–	$3.52 \cdot 10^{-2}$	–	$9.94 \cdot 10^1$
	2:5	$6.98 \cdot 10^{-2}$	0.47	$8.18 \cdot 10^{-2}$	0.25	$2.84 \cdot 10^{-3}$	2.16	$5.62 \cdot 10^{-3}$	2.64	$5.13 \cdot 10^1$
	3:6	$8.27 \cdot 10^{-2}$	–0.24	$8.48 \cdot 10^{-2}$	–0.05	$2.51 \cdot 10^{-3}$	0.17	$4.15 \cdot 10^{-3}$	0.43	$1.63 \cdot 10^1$
	4:7	$1.98 \cdot 10^{-3}$	5.38	$2.03 \cdot 10^{-3}$	5.37	$4.43 \cdot 10^{-4}$	2.50	$7.23 \cdot 10^{-4}$	2.52	$1.49 \cdot 10^0$
	5:8	$6.08 \cdot 10^{-4}$	1.70	$6.08 \cdot 10^{-4}$	1.74	$8.34 \cdot 10^{-5}$	2.41	$1.56 \cdot 10^{-4}$	2.21	$2.99 \cdot 10^{-1}$
	6:9	$1.72 \cdot 10^{-4}$	1.81	$1.73 \cdot 10^{-4}$	1.81	$2.11 \cdot 10^{-5}$	1.98	$4.26 \cdot 10^{-5}$	1.87	$7.15 \cdot 10^{-2}$
	7:10	$4.68 \cdot 10^{-5}$	1.88	$4.68 \cdot 10^{-5}$	1.88	$4.98 \cdot 10^{-6}$	2.08	$1.07 \cdot 10^{-5}$	1.99	$1.72 \cdot 10^{-2}$
	8:11	$1.20 \cdot 10^{-5}$	1.95	$1.20 \cdot 10^{-5}$	1.95	$1.22 \cdot 10^{-6}$	2.02	$2.70 \cdot 10^{-6}$	1.98	$4.22 \cdot 10^{-3}$

4.4. Stretching

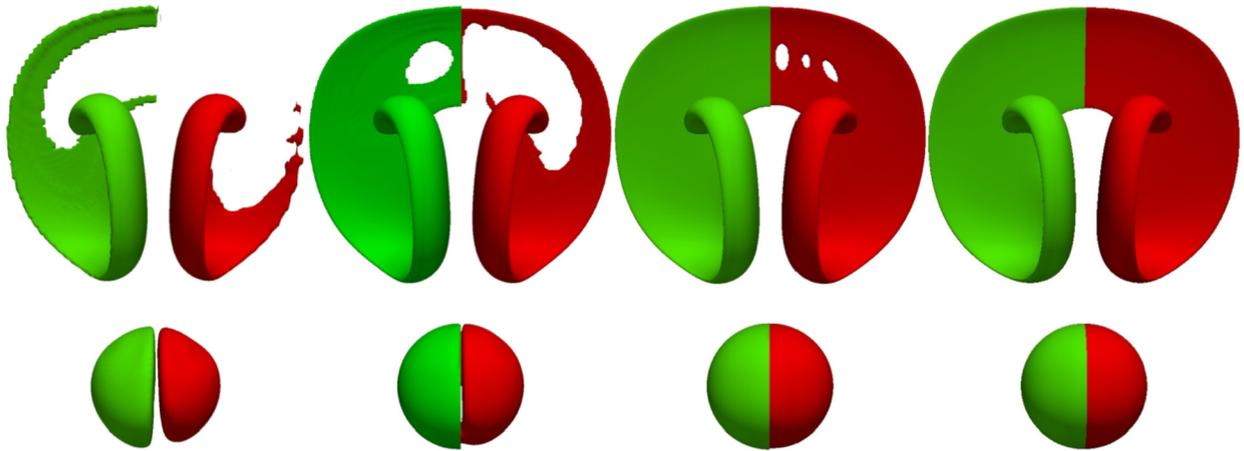
For this example we consider the non divergence-free velocity field

$$\mathbf{u}(x, y) = (0.1 \tanh(10(y - 0.5)), 0.1 \tanh(10(x - 0.5)))$$

The initial geometry is a circle of radius  $R = 0.1$ , centered at  $(0.5, 0.5)$  and we take  $CFL = 5$ . The velocity field and the evolution of the interface are depicted in Fig. 8. We observe second-order convergence for the level-set method (see Table 10) and close to third-order convergence for the coupled method. On the coarsest grids, for both methods, the elongation at the center of the domain is poorly resolved and eventually causes the contour to separate into two disconnected contours. These artificial topological changes explain the poor convergence rates at low resolutions.

**Table 11**  
Convergence analysis for the 3D Enright’s test 4.5.

	Levels	$d_{L^\infty}(\phi)$	Order	$e_{L^\infty}(\phi)$	Order	$d_{L^1}(\phi)$	Order	$d_{L^2}(\phi)$	Order	Mass loss (%)	Order
CLSRM	1:7	$1.46 \cdot 10^{-1}$	–	$1.48 \cdot 10^{-1}$	–	$5.54 \cdot 10^{-4}$	–	$1.56 \cdot 10^{-3}$	–	$1.11 \cdot 10^1$	–
	2:8	$9.70 \cdot 10^{-2}$	0.59	$1.46 \cdot 10^{-1}$	0.03	$1.54 \cdot 10^{-4}$	1.84	$4.22 \cdot 10^{-4}$	1.12	$1.12 \cdot 10^0$	3.31
	3:9	$3.13 \cdot 10^{-3}$	4.95	$9.00 \cdot 10^{-3}$	4.02	$2.57 \cdot 10^{-5}$	2.58	$7.41 \cdot 10^{-5}$	2.51	$1.06 \cdot 10^{-1}$	3.39
	4:10	$2.76 \cdot 10^{-4}$	3.49	$2.76 \cdot 10^{-4}$	5.02	$5.91 \cdot 10^{-6}$	2.12	$1.63 \cdot 10^{-5}$	2.18	$2.86 \cdot 10^{-2}$	1.89
level-set	1:7	$1.41 \cdot 10^{-1}$	–	$1.48 \cdot 10^{-1}$	–	$4.96 \cdot 10^{-4}$	–	$1.72 \cdot 10^{-3}$	–	$2.46 \cdot 10^1$	–
	2:8	$1.45 \cdot 10^{-1}$	–0.04	$1.50 \cdot 10^{-1}$	–0.01	$2.39 \cdot 10^{-4}$	1.05	$7.16 \cdot 10^{-4}$	1.26	$1.03 \cdot 10^1$	1.25
	3:9	$1.32 \cdot 10^{-1}$	0.13	$1.32 \cdot 10^{-1}$	0.18	$9.54 \cdot 10^{-5}$	1.32	$2.82 \cdot 10^{-4}$	1.34	$1.93 \cdot 10^0$	2.41
	4:10	$4.70 \cdot 10^{-3}$	4.81	$4.70 \cdot 10^{-3}$	4.81	$3.63 \cdot 10^{-5}$	1.39	$1.09 \cdot 10^{-4}$	1.37	$3.99 \cdot 10^{-1}$	2.27



**Fig. 9.** Intermediate ( $t = 1.1$ , top) and final interfaces ( $t = 2.2$ , bottom) computed using the CLSRM (green) and standard level-set methods (red) for the Enright’s sphere test. For visual purposes only half of the interface is depicted for each method.

4.5. Three-dimensional Enright’s test

For this example, proposed by Enright et al. [13], the computational domain is  $\Omega = [0, 1]^3$ , the initial interface is a sphere of radius 0.15 centered at (0.35, 0.35, 0.35). The deforming velocity is

$$\mathbf{u}(x, y, z) = \left( 2 \sin^2(\pi x) \sin(2\pi y) \sin(2\pi z), -\sin^2(\pi y) \sin(2\pi x) \sin(2\pi z), -\sin^2(\pi z) \sin(2\pi x) \sin(2\pi y) \right).$$

We take CFL = 10 and we only impose a uniform band of  $\Delta x$  on each side of the interface to speed up computations. Convergence results are given in Table 11. The intermediate and final interfaces are depicted in green in Fig. 9. As for the two-dimensional examples, our new coupled method performs systematically better. The  $L^1$  and  $L^2$  order of convergence are consistent with our previous observations. At low resolutions, the extreme stretching of the interface is under-resolved, causing the topological changes observed in Fig. 9, which induced large  $L^\infty$  errors impacting the overall convergence. Nonetheless, the average orders (2.66 for the CLSRM and 1.63 for the level-set) are in good agreement with our previous observations.

5. Simulation of two-phase flows – method

For this application, we employ our method to simulate two-phase incompressible, non-miscible and Newtonian fluids. We incorporate our new interface representation in our existing two-phase flow solver [62]. The overall method is described below, and we refer the interested reader to [62] for a detailed description.

5.1. Governing equations

5.1.1. Incompressible Navier-Stokes equations

Under the influence of a gravity field  $\mathbf{g}$  and the assumptions that the viscosities  $\mu^+, \mu^-$  and densities  $\rho^+, \rho^-$  are constant depending on the phase, the general form of the Navier-Stokes equations is

$$\begin{cases} \rho^\pm \left( \frac{\partial \mathbf{u}^\pm}{\partial t} + \mathbf{u}^\pm \cdot \nabla \mathbf{u}^\pm \right) = -\nabla p^\pm + \mu^\pm \Delta \mathbf{u}^\pm + \rho^\pm \mathbf{g} & \text{in } \Omega^\pm \setminus \Gamma \\ \nabla \cdot \mathbf{u}^\pm = 0 & \text{in } \Omega^\pm \setminus \Gamma, \end{cases} \tag{21}$$

where the upper script  $\pm$  refers to quantities defined in phases  $\Omega^+$  or in  $\Omega^-$  respectively.

5.1.2. Interface and boundary conditions

The system (21) is completed by enforcing the continuity of velocity and stress across the interface

$$\begin{cases} \llbracket \mathbf{u} \rrbracket = 0 & \text{on } \Gamma \\ \llbracket \sigma \mathbf{n} - p \mathbf{n} \rrbracket = \gamma \kappa \mathbf{n} & \text{on } \Gamma, \end{cases} \tag{22}$$

where  $\llbracket \psi \rrbracket = \psi^+ - \psi^-$  is the jump of the quantity  $\psi$  across the interface  $\Gamma$ ,  $\sigma$  is the viscous stress tensor that writes, in this case,  $\sigma = \mu (\nabla \mathbf{u} + \nabla \mathbf{u}^T)$ ,  $\gamma$  is the constant surface tension and  $\mathbf{n}$  and  $\kappa$  are respectively normal vector and curvature of the interface as previously defined. The boundary conditions on  $\partial\Omega$  will be specified for each problem.

5.2. Projection method

We follow the approach presented in [62], itself based on [19]. For the sake of simplicity, we drop the  $\pm$  sign and let  $\mathbf{u}, \mu, \rho$  be single fields defined in both phases. The solver is designed as a projection method, using the Helmholtz-Hodge decomposition as it was first introduced by Chorin (see [8]). It is composed of two main steps. First, an intermediate velocity field  $\mathbf{u}^*$ , is computed as the solution of

$$\rho \left( \frac{\mathbf{u}^* - \mathbf{u}^n}{\Delta t} + \mathbf{u}^n \cdot \nabla \mathbf{u}^n \right) = -\nabla \tilde{p} + \mu \Delta \mathbf{u}^* + \rho \mathbf{g}, \tag{23}$$

where  $\tilde{p}$  is a pressure guess that will be defined in subsection 5.2.1. Writing the Helmholtz-Hodge decomposition for  $\mathbf{u}^{n+1}$ , we obtain

$$\mathbf{u}^{n+1} = \mathbf{u}^* - \nabla \Phi, \tag{24}$$

where  $\Phi$  is the Hodge variable. Taking the divergence of Eq. (24) and enforcing the incompressibility condition on  $\mathbf{u}^{n+1}$ ,  $\Phi$  is shown to be the solution of

$$\Delta \Phi = \nabla \cdot \mathbf{u}^*. \tag{25}$$

Specifying the appropriate boundary and interface conditions,  $\Phi$  is constructed as the numerical solution of the above Poisson problem and used to reconstruct  $\mathbf{u}^{n+1}$  according to Eq. (24). As it is commonly done, the solution components are stored according to the Marker And Cell (MAC) layout, the pressure (and the Hodge variable) being stored at the cell centers, while the velocity is stored at the cell faces.

5.2.1. Overall description of the Navier-Stokes solver

At each iteration, the new velocity field and level set are computed following these main five steps

1. **Initialization** of the corrective terms  $\mathbf{X}_0$  and  $\Sigma_0$ , which will be used to enforce the jump conditions given by Eq. (22).
2. **Pressure guess**: We construct the pressure guess  $\tilde{p}$ , as the solution of

$$\begin{cases} \Delta \tilde{p} = 0 & \text{in } \Omega \setminus \Gamma, \\ \llbracket \tilde{p} \rrbracket = -\gamma \kappa - \mathbf{g} \cdot \mathbf{n} & \text{on } \Gamma, \\ \llbracket \frac{1}{\rho} \nabla \tilde{p} \cdot \mathbf{n} \rrbracket = 0 & \text{on } \Gamma, \end{cases} \tag{26}$$

using a second-order cell-based finite-volume solver.

3. **Repeat until convergence**

- **Viscosity step**: compute  $\mathbf{u}^*$  by discretizing Eq. (23)

$$\begin{cases} \rho \frac{D\mathbf{u}^*}{Dt} = \mu \Delta \mathbf{u}^* - \nabla \tilde{p} + \rho \mathbf{g} & \text{in } \Omega \setminus \Gamma \\ \llbracket \mathbf{u}^* \rrbracket = \mathbf{X}_k & \text{on } \Gamma \\ \llbracket \mu \nabla \mathbf{u}^* \cdot \mathbf{n} \rrbracket = \mathbf{g} - \mathbf{n}(\mathbf{g} \cdot \mathbf{n}) + \Sigma_k & \text{on } \Gamma. \end{cases} \tag{27}$$

The time discretization of this system is made using a semi-Lagrangian approach similar to that developed in Section 3.2, incorporating a second-order Backward Difference Formula (BDF). The space discretization is achieved using a finite-volume Poisson solver based on Voronoi's partitions.

- **Projection step**: compute  $\Phi$ , the Hodge variable

$$\begin{cases} \Delta \Phi = \nabla \cdot \mathbf{u}^* & \text{in } \Omega \setminus \Gamma \\ \llbracket \rho \Phi \rrbracket = 0 & \text{on } \Gamma \\ \llbracket \nabla \Phi \cdot \mathbf{n} \rrbracket = 0 & \text{on } \Gamma, \end{cases} \tag{28}$$

**Table 12**  
Parasitic currents: error on the velocity field and interface location measured at the final time  $T$ .

max <sub>level</sub>	CLSRM			Level-set		
	$e_{L^\infty}(u)$	$e_{L^\infty}(\phi)$	Order	$e_{L^\infty}(u)$	$e_{L^\infty}(\phi)$	Order
4	$4.04 \cdot 10^{-2}$	$4.16 \cdot 10^{-3}$	–	$6.54 \cdot 10^{-2}$	$1.72 \cdot 10^{-2}$	–
5	$2.97 \cdot 10^{-2}$	$4.09 \cdot 10^{-4}$	3.34	$3.68 \cdot 10^{-2}$	$2.27 \cdot 10^{-3}$	2.92
6	$9.81 \cdot 10^{-4}$	$7.03 \cdot 10^{-5}$	2.54	$7.08 \cdot 10^{-2}$	$1.09 \cdot 10^{-3}$	1.06
7	$3.65 \cdot 10^{-2}$	$1.30 \cdot 10^{-5}$	2.43	$6.55 \cdot 10^{-2}$	$4.78 \cdot 10^{-4}$	1.18
8	$3.63 \cdot 10^{-3}$	$1.69 \cdot 10^{-6}$	2.94	$8.59 \cdot 10^{-2}$	$2.74 \cdot 10^{-4}$	0.80
9	$8.91 \cdot 10^{-4}$	$2.52 \cdot 10^{-7}$	2.74	$4.64 \cdot 10^{-2}$	$7.49 \cdot 10^{-5}$	1.87
10	$1.49 \cdot 10^{-3}$	$3.15 \cdot 10^{-8}$	2.99	$4.17 \cdot 10^{-2}$	$2.81 \cdot 10^{-5}$	1.41

using the same approach as for Eq. (26).

4. **Reference map advection:**  $\xi^{n+1}$  is computed by advecting  $\xi^n$  using  $\mathbf{u}^{n+1}$ , as we fully detail in Section 3.2.

5. **Final update:** we update the mesh and all the quantities defined on it, interpolating when necessary.

Again, we refer the interested reader to [62] for a detailed description of steps 1, 2 and 3 and the numerical techniques they involve.

### 5.2.2. Temporal and spatial resolution

As it was observed in [62], the previous the method is stable under the time step restriction

$$\Delta t < \min \left( \frac{\Delta x}{\|\mathbf{u}^\pm\|_{L^\infty}}, \frac{\min(\mu^-, \mu^+) \Delta x}{\gamma} + \sqrt{\left( \frac{\min(\mu^-, \mu^+) \Delta x}{\gamma} \right)^2 + \frac{(\rho^+ + \rho^-) \Delta x^3}{4\pi \gamma}} \right),$$

where  $\Delta x$  is the smallest grid size. We note that the above condition, inspired by the work of Galusinski et al. [16], is always less restrictive than the classical one proposed by Brackbill in [6].

On top of the meshing criteria (13), we continuously adapt the mesh to the spatial variation of the velocity field, splitting any cell  $\mathcal{C}$  as follows:

$$\text{split } \mathcal{C} \text{ if } : \max_{n \in \text{nodes}(\mathcal{C})} \frac{|\nabla \mathbf{u}(n)|}{\|\mathbf{u}\|_\infty} \geq T_V \quad \text{and} \quad \text{level}(\mathcal{C}) \leq \max_V \leq \max_{\text{level}}, \quad (29)$$

where  $T_V$  is an arbitrary threshold and  $\max_V$  is the maximum level allowed for this criterion.

## 6. Simulation of two-phase flows – examples

### 6.1. Parasitic or spurious currents

In this section, we study the resilience of our numerical method against the well-known and widely studied parasitic or spurious currents (see [62,54,43,22,15] for example). They consist in purely artificial flows generated by the capillary forces at the interface  $\Gamma$ . The setting is the following: we consider an initially circular drop in two dimensions having radius  $R > 0$  in absence of gravity ( $\mathbf{g} = 0$ ) and we measure the interface velocity and displacement, knowing that they should – theoretically – vanish. We use these parameters:

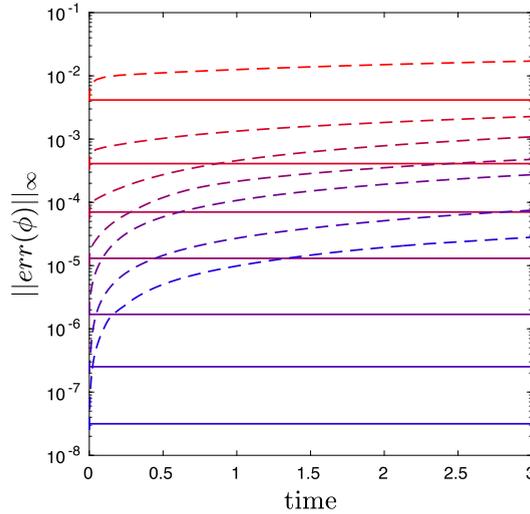
$$\Omega = [-1.25, 1.25]^2 \quad R = 0.5 \quad \rho^- = 1 \quad \rho^+ = 10^{-3} \quad \mu^- = 10^{-2} \quad \mu^+ = 10^{-5} \quad \gamma = 0.5 \quad T = 3,$$

and enforce homogeneous Dirichlet boundary condition for the velocity fields on the left and right computational walls, and homogeneous Neumann boundary conditions on the top and bottom walls. Alternated boundary conditions are considered for the pressure. To verify the convergence of our method, we consider a fixed minimum level of 4 and gradually increase the maximum level. The mesh is only refined close to the interface, using the criterion provided in Eq. (13).

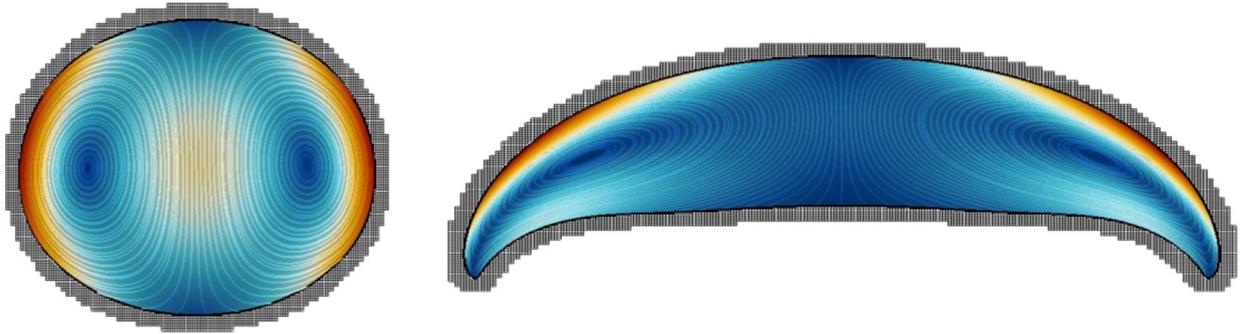
Errors in the  $L^\infty$ -norm and at the final time are reported in Table 12, using both our level-set/reference map method or the standard level set to capture the interface. We remark that for both methods, the parasitic velocity remains small and in particular a few orders smaller than the characteristic velocity

$$\|\mathbf{u}\|_{L^\infty} \ll \frac{\gamma}{\mu^-} = 50,$$

as it was previously observed by [26,68]. The parasitic velocity with our coupled method is significantly smaller, and possibly converging. The error on the interface position is converging in both cases, being close to second-order convergent for the level-set method, as reported in [62], and third-order convergent for the level-set/reference map method. As for the



**Fig. 10.** Parasitic currents: interface location error plotted against time for both the coupled method (solid lines) and the standard level-set method (dash lines) and increasing maximum resolutions (coarsest in red, finest in blue).



**Fig. 11.** Rising bubble: terminal shape, stream lines of the relative velocity and finest mesh near the interface: Example 6.2.1 (left) and 6.2.2 (right). Color refers to the magnitude of the relative velocity. Dark is associated with high velocity, blue indicates small velocity.

translation 4.1 and rotation 4.2 tests, the absence of restarting – and thus of reinitialization – decided by our restarting criterion (11) explains these accuracy improvements. The impact of the reinitialization on the simulation is well illustrated in Fig. 10, representing the temporal evolution of the interface errors. With the level-set method, the systematic reinitialization generates numerical errors adding up over time. With the coupled method, the error appears to be constant over time.

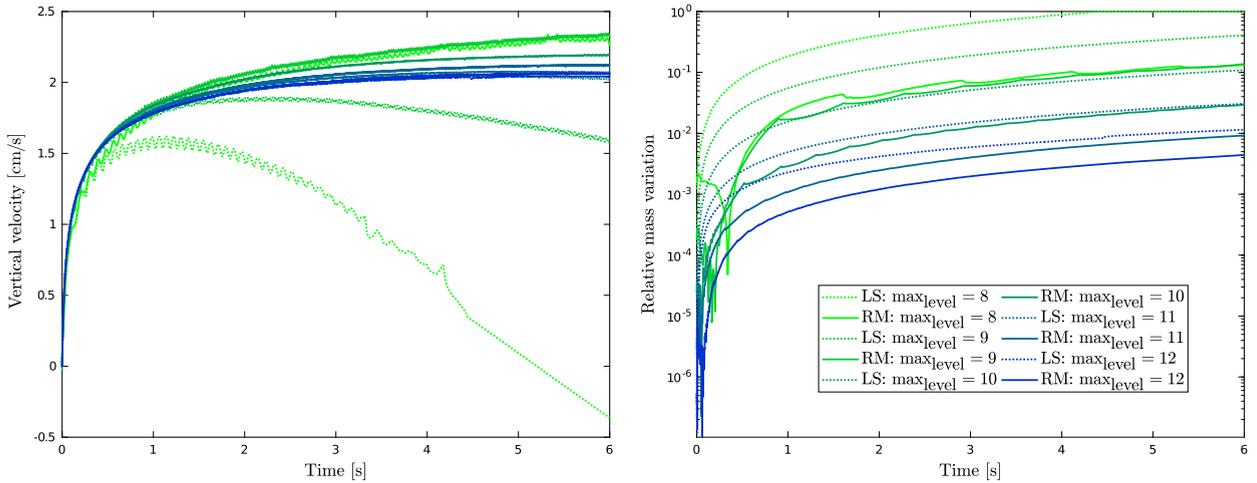
6.2. Rising bubble

Here we study the dynamics of a rising bubble in two dimensions, reproducing the validations exposed in [62] and inspired by the experimental observations of Bhaga et al. [5]. Because of the differences between two and three-dimensional behaviors, and in light of recent studies [28], we expect slight discrepancies between our computational results and the experimental observations [5].

For our analysis, we consider an initially spherical (gas) bubble placed in a heavier and more viscous fluid (liquid), subject to the gravitational force. The buoyancy force exerted on the bubble induces its rise. The dimensional analysis conducted in [5] identifies three dimensionless quantities: the Morton, Eötvös and Reynolds numbers

$$Mo = \frac{g\mu_+^4}{\rho_+\gamma^3}, \quad Eo = \frac{gd^2\rho_+}{\gamma}, \quad Re = \frac{\rho_+Ud}{\mu_+}, \tag{30}$$

where the subscript + refers to the liquid phase, while – is for the gas (the bubble),  $d$  is the bubble diameter and  $U$  the asymptotic rising velocity. The Morton number is the ratio between viscous and capillary effects, the Eötvös number measures the influence of the gravitational force over the surface tension, and finally the Reynolds number is the ratio between inertial and viscous effects. In our simulation, all parameters are constructed from these three numbers setting the rising velocity to  $U = 1$ , yielding:



**Fig. 12.** Bubble facing moderate deformations: comparison between our coupled method (solid lines) and the standard level-set method (dash lines). Drop velocity (left) and relative mass loss as functions of time and for increasing maximum resolution.

$$\rho_+ = 1, \quad \frac{\rho_+}{\rho_-} = 10^3, \quad \mu_+ = \frac{\rho_+ d}{Re} U, \quad \frac{\mu_+}{\mu_-} = 10^2, \quad \gamma = \frac{\mu_+^2}{d \rho_+} \sqrt{\frac{Eo}{Mo}}, \quad g = \frac{\rho_+ \gamma^3}{\mu_+^4} Mo.$$

The viscosity and density ratios are chosen to be close to the typical air-water ones. We consider an initial drop of diameter 1 placed in a square domain of side length  $L = 32$  at an initial high of  $H = 8$  from the bottom. By taking such a large domain, we aim to annihilate the influence of the wall boundary conditions on the overall dynamics. We impose no-slip boundary conditions for the velocity field on every wall except for the upper one, where a no-flux boundary condition is enforced. We set the pressure to be equal to the hydrostatic (up to an additive constant) pressure:

$$p = -g \rho_+ y \quad \text{on } \partial\Omega. \tag{31}$$

As for the previous example, we set the minimum level for the mesh as  $\text{min}_{\text{level}} = 4$ , and we study the convergence of our algorithm by varying the maximum level of the quadtree. The mesh is refined both where the interface and where high variations of velocity occur using

$$T_V = 0.01, \quad \text{max}_V = \text{max}_{\text{level}} - 1,$$

in criterion (29).

6.2.1. Bubble facing moderate deformations

We first consider the case (b) taken from [5], associated with the following set of dimensionless parameters:

$$Mo = 711, \quad Eo = 17.7, \quad Re = 0.232. \tag{32}$$

For this configuration, the Weber number is  $We = Re^2 \sqrt{\frac{Mo}{Eo}} \approx 0.34$ , indicating that moderate deviations of the bubble from the initial circular shape are expected, and that therefore, our method should perform significantly better than the standard level set. The final shape and relative flow are depicted in Fig. 11. The resulting rising velocities (measured at the tip of the bubble) and mass losses are shown in Fig. 12. The final position of the bubble for both methods and various maximum resolutions is depicted in Fig. 13.

The mass losses with our coupled method are typically a few times smaller and are particularly noticeable on the coarsest resolutions. In fact, for the lowest resolution ( $\text{max}_{\text{level}} = 8$ ), with the level-set method, the bubble loses its entire mass right before the final time, while by then the same bubble, simulated using our new method has retained 90% of its mass. Both methods are converging towards the same terminal velocity, approximately 2.1, which is arguably close to the experimental value (being 1). The final shapes are in good agreement with the observations of Bhaga et al. [5] and the three-dimensional simulations in [62]. With the level-set/reference map method, the rising velocity appears to be always larger than with the level-set method. We understand it as a consequence of the improved mass conservation: a larger bubble will rise faster.

6.2.2. Bubble facing strong deformations

For this example, we look at a more extreme situation, corresponding to the case (d) in [5] and involving the following dimensionless parameters

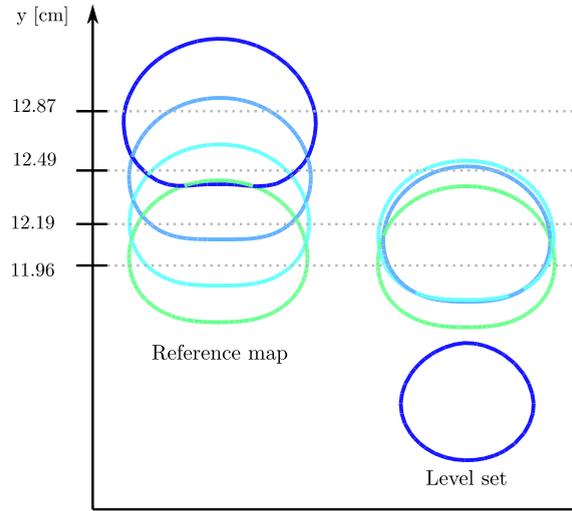


Fig. 13. Bubble facing moderate deformations: drop position at final time  $T$ . Blue for  $\max_{\text{level}} = 9$ , light-blue for  $\max_{\text{level}} = 10$ , pale light-blue for  $\max_{\text{level}} = 11$  and finally pale green for  $\max_{\text{level}} = 12$ . The dotted line means that the two point are at the same high.

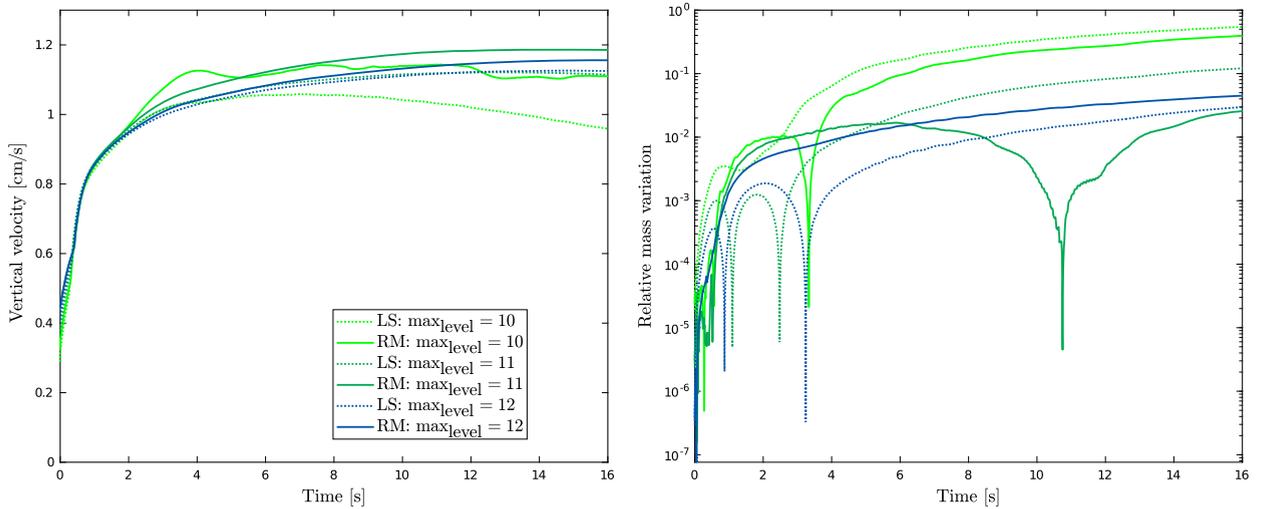


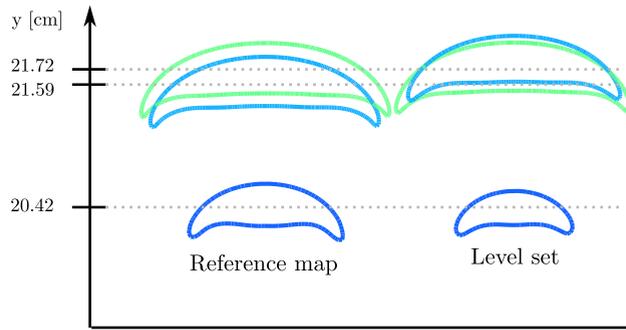
Fig. 14. Bubble facing strong deformations: solid line is for the reference map method, dotted line is for the level-set procedure. On the left: velocity of the bubble with respect to time. On the right: relative mass variation (without sign) as function of time.

$$Mo = 266, \quad Eo = 243, \quad Re = 7.77. \tag{33}$$

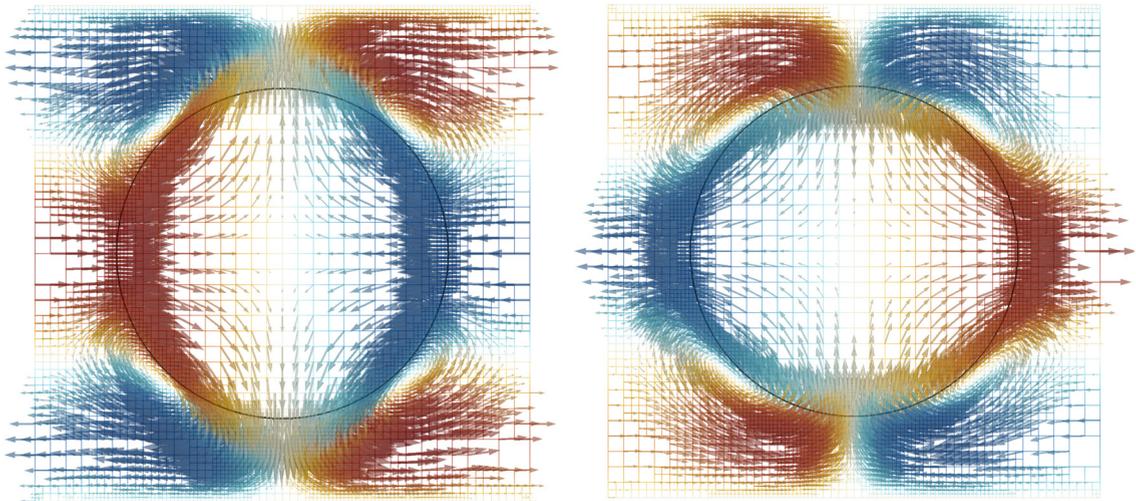
In this case, the Weber number is approximately  $We \approx 66$ , justifying the strong deviations from the initial spherical shape observed in Fig. 11. Looking at Figs. 14 and 15, we observe both methods converging to the same terminal rising velocity, shape and position. Again the mass loss is reduced using our method, even though the reduction is slightly less important than in the previous case. It is possible that the larger deformations of the reference map introduce more frequent restarting and thus additional numerical errors.

### 6.3. Oscillating droplet

For this example, we test our new method against the oscillations of a droplet around its equilibrium spherical shape. The initial perturbation of the droplet shape generates non-uniform capillary forces. Their interaction with inertial forces generates oscillations of the droplet shape and a fluid flow (see Fig. 16). Eventually, these oscillations are damped by the viscous effects. This problem was first theoretically studied by Lamb in [27], where it is shown that in the limit of infinitesimal deformations, the drop radius  $R(\theta, t)$  can be decomposed on a basis of eigenmodes, plus an exponential decay:



**Fig. 15.** Bubble facing strong deformations: bubble position at final time  $T$ . Blue for  $\max_{\text{level}} = 10$ , light-blue for  $\max_{\text{level}} = 11$  and finally pale green for  $\max_{\text{level}} = 12$ .



**Fig. 16.** Oscillating droplet: contraction and expansion along the  $x$ -axis induced by the capillary forces. The arrows are parallel to the velocity field and their sizes are proportional to its local magnitude. They are colored by the  $x$ -velocity, blue and red being associated large positive and negative values. The droplet interface is represented in black.

$$R(t, \theta) = R_\infty + \sum_{l=2}^{+\infty} \epsilon_l \cos(\omega_l t) P_l(\cos \theta) e^{-\frac{t}{\tau_l}} \quad t \geq 0, \quad \theta \in [0, 2\pi), \tag{34}$$

where  $\theta$  is the polar angle,  $R_\infty$  is the asymptotic radius of the drop,  $\epsilon_l$  is the amplitude of the  $l$ -th mode and  $P_l$  is the  $l$ -th Legendre polynomial, proportional to the spherical harmonics  $Y_l$  up to a factor depending on  $l$ , included in the definition of each  $\epsilon_l$ . Moreover, the pulsation and the characteristic damping time for each eigenmode are given by:

$$\omega_l = \sqrt{\frac{\gamma}{R_\infty^3} \frac{l(l-1)(l+1)(l+2)}{(l+1)\rho^- + l\rho^+}}, \quad \tau_l = \frac{R_\infty^2 \rho^-}{\mu^-(2l+1)(l-1)} \quad \text{with } l \in \{2, 3, \dots\}. \tag{35}$$

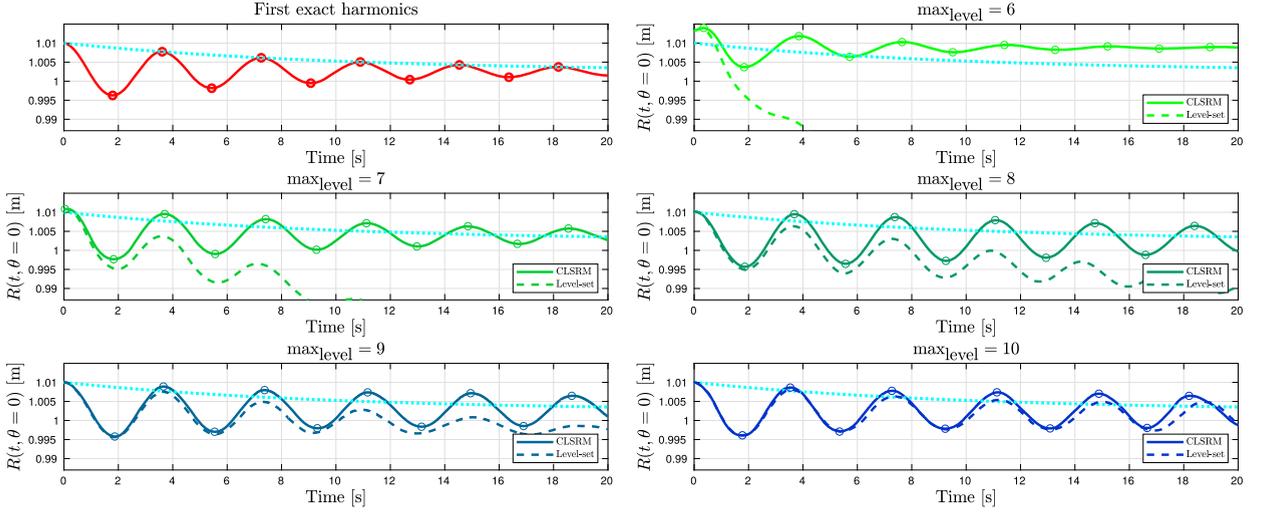
Limiting our study to the two-dimensional problem, we use the two-dimensional expression for the pulsation taken from [50]

$$\omega_l^{2D} = \sqrt{\frac{(l^3 - l)\gamma}{(\rho^+ + \rho^-)R_\infty^3}} \quad \text{with } l \in \{2, 3, \dots\}. \tag{36}$$

Unfortunately, we were unable to find a similar result for the damping coefficient, so we conduct our analysis using the above three-dimensional one.

For our simulation, we consider a circular bubble and we deform its shape along the second mode. This corresponds to the initial level-set function:

$$\phi(t = 0, r, \theta) = \phi_0(r, \theta) = r - (R + \epsilon P_2(\cos \theta)) = r - \left( R + \frac{\epsilon}{2} (3 \cos^2 \theta - 1) \right), \tag{37}$$



**Fig. 17.** Bubble radius as a function of time for different  $\max_{\text{level}}$ . The dotted light-blue line is the theoretical three-dimensional exponential decay. CLSRM (solid lines) and classical level-set (dash lines) methods.

where  $r$  is the radial coordinate and  $\epsilon = \epsilon_2 \ll 1$ . The asymptotic radius  $R_\infty$  toward which the shape is expected to converge is not exactly  $R$ , but the equivalent radius i.e. the radius of the perfectly round disk of area  $A = |\mathcal{D}|$ , taking  $\mathcal{D} = \{(r, \theta) : \phi_0(r, \theta) < 0\}$ . In other words:

$$R_\infty = \sqrt{\frac{A}{\pi}}. \tag{38}$$

To find  $A$ , we compute the following integral:

$$A = \int_{\mathcal{D}} dx dy = \int_0^{2\pi} \int_0^{R + \frac{\epsilon}{2}(3 \cos^2 \theta - 1)} r dr d\theta = \frac{1}{2} \int_0^{2\pi} \left[ R + \frac{\epsilon}{2}(3 \cos^2 \theta - 1) \right]^2 d\theta = \pi R^2 + \epsilon \frac{\pi R}{2} + \epsilon^2 \frac{11\pi}{32},$$

yielding:

$$R_\infty = \sqrt{R^2 + \epsilon \frac{R}{2} + \epsilon^2 \frac{11}{32}}. \tag{39}$$

Computations are performed using the following parameters:

$$\begin{aligned} \Omega &= [-1.5, 1.5]^2, & T &= 20, & \min_{\text{level}} &= 4, & \epsilon &= 0.01 \ll 1, \\ R &= 1, & \rho^- &= 1, & \mu^- &= 0.02, & \rho^+ &= 10^{-3} \cdot \rho^-, & \mu^+ &= 10^{-3} \cdot \mu^-, & \gamma &= 0.5, \end{aligned}$$

and consider homogeneous Neumann boundary condition for the velocity fields on the left and right computational walls, and homogeneous Dirichlet boundary conditions on the top and bottom walls. Alternated boundary conditions are enforced on the pressure. For the above parameters, the first expected eigenfrequencies, periods and damping time are

$$\begin{aligned} \omega_2 &= 2.4385 \text{ s}^{-1} & \omega_2^{2D} &= 1.7247 \text{ s}^{-1} \\ T_2 &= 2.5767 \text{ s} & T_2^{2D} &= 3.6431 \text{ s}, \\ \tau_2 &= 10 \text{ s}. \end{aligned}$$

Results provided in Fig. 17 and in Table 13 show that the expected oscillation period is very well matched by both methods. Yet, the coupled level-set/reference map approach consistently performs better than the classical level-set approach. Focusing on mass conservation, our new method performs outstandingly better. The difference is visually noticeable for the low resolution (6, 7, 8) in Fig. 17, where the drop radius decays below the range of plotted values. The viscous damping agrees fairly well with the three-dimensional predictions.

**Table 13**

Oscillating bubble: evolution of the measured period (and relative difference to the reference value) as well as the mass loss with mesh refinement.

max <sub>level</sub>	CLSRM		Level-set	
	T [s]	Mass loss	T [s]	Mass loss
6	3.7725 (+2.18%)	$1.16 \cdot 10^{-2}$	–	$1.68 \cdot 10^1$
7	3.7011 (+1.59%)	$3.08 \cdot 10^{-2}$	3.4315 (–5.18%)	$7.50 \cdot 10^0$
8	3.6743 (+0.86%)	$4.96 \cdot 10^{-3}$	3.4892 (–4.22%)	$2.42 \cdot 10^0$
9	3.7392 (+2.50%)	$3.40 \cdot 10^{-3}$	3.7592 (+3.19%)	$1.02 \cdot 10^0$
10	3.6495 (+0.18%)	$6.54 \cdot 10^{-4}$	3.7744 (+3.60%)	$3.44 \cdot 10^{-1}$

## 7. Conclusions

We have presented a coupled level-set and reference map method for the general interface advection problem. The central idea relies on advecting the entire computational space (represented by the reference map) and using this deformed space and the knowledge of the initial interface to compute the current one, instead of advecting the interface directly, as it is done with the level-set method. In lieu of the level-set reinitialization, we have constructed a restarting procedure and proposed a criterion to decide when it should be performed. Mathematically, it ensures that the reference map remains bijective at all times. In practice, it prevents the appearance of artificial contours.

Although our new method uses exactly the same numerical techniques as the traditional level-set method and has almost the same computational cost, it systematically grants higher accuracy. This improvement in accuracy has two main explanations. First, the advected function is smoother: while the level-set function is not continuously differentiable, the reference map is infinitely differentiable. Second, the numerical errors introduced by the reinitialization are reduced: while with the standard level-set method, the reinitialization must be performed at each iteration, with our coupled method, it is only performed when restarting is needed.

In the last section, we show that our coupled method can indeed be implemented to improve any level-set based method without having to design or implement new computational techniques. For the application we considered, the simulation of two-phase flows, our coupled level-set/reference map method makes a substantial difference, in particular at reducing the mass loss and minimizing parasitic waves. We want to emphasize our method's simplicity as its key feature, especially compared to other improved level-set frameworks.

## Acknowledgements

The authors would like to deeply thank Chris Rycroft for stimulating discussion at UC Merced and the two anonymous reviewers for their valuable comments and suggestions. M.T. gratefully acknowledge support through startup funds from the University of California, Merced and computing time on the Multi-Environment Computer for Exploration and Discovery (MERCED) cluster at the University of California, Merced, which was funded by National Science Foundation Grant No. ACI-1429783.

## References

- [1] D. Adalsteinsson, J.A. Sethian, The fast construction of extension velocities in level set methods, *J. Comput. Phys.* 148 (1) (1999) 2–22.
- [2] David Adalsteinsson, James A. Sethian, Transport and diffusion of material quantities on propagating interfaces via level set methods, *J. Comput. Phys.* 185 (1) (2003) 271–288.
- [3] D.M. Anderson, G.B. McFadden, A.A. Wheeler, Diffuse-interface methods in fluid mechanics, *Annu. Rev. Fluid Mech.* 30 (1) (1998) 139–165.
- [4] John B. Bell, Phillip Colella, Harland M. Glaz, A second-order projection method for the incompressible Navier-Stokes equations, *J. Comput. Phys.* 85 (2) (1989) 257–283.
- [5] Dahya Bhaga, M.E. Weber, Bubbles in viscous liquids: shapes, wakes and velocities, *J. Fluid Mech.* 105 (1981) 61–85.
- [6] J.U. Brackbill, Douglas B. Kothe, Charles Zemach, A continuum method for modeling surface tension, *J. Comput. Phys.* 100 (2) (1992) 335–354.
- [7] Y.C. Chang, T.Y. Hou, B. Merriman, S. Osher, A level set formulation of Eulerian interface capturing methods for incompressible fluid flows, *J. Comput. Phys.* 124 (2) (1996) 449–464.
- [8] Alexandre Joel Chorin, Numerical solution of the Navier-Stokes equations, *Math. Comput.* 22 (1968) 745–762.
- [9] Georges-Henri Cottet, Emmanuel Maitre, Thomas Milcent, Eulerian formulation and level set models for incompressible fluid-structure interaction, *ESAIM: Math. Model. Numer. Anal.* 42 (3) (2008) 471–492.
- [10] Charles Cleret de Langavant, Arthur Guittet, Maxime Theillard, Fernando Temprano-Coleto, Frédéric Gibou, Level-set simulations of soluble surfactant driven flows, *J. Comput. Phys.* 348 (2017) 271–297.
- [11] Olivier Desjardins, Vincent Moureau, Heinz Pitsch, An accurate conservative level set/ghost fluid method for simulating turbulent atomization, *J. Comput. Phys.* 227 (18) (2008) 8395–8416.
- [12] Olivier Desjardins, Heinz Pitsch, A spectrally refined interface approach for simulating multiphase flows, *J. Comput. Phys.* 228 (5) (2009) 1658–1677.
- [13] Douglas Enright, Ronald Fedkiw, Joel Ferziger, Ian Mitchell, A hybrid particle level set method for improved interface capturing, *J. Comput. Phys.* 183 (1) (2002) 83–116.
- [14] Douglas Enright, Frank Losasso, Ronald Fedkiw, A fast and accurate semi-Lagrangian particle level set method, in: *Frontier of Multi-Phase Flow Analysis and Fluid-Structure*, *Comput. Struct.* 83 (6) (2005) 479–490.
- [15] Marianne M. Francois, Sharen J. Cummins, Edward D. Dendy, Douglas B. Kothe, James M. Sicilian, Matthew W. Williams, A balanced-force algorithm for continuous and sharp interfacial surface tension models within a volume tracking framework, *J. Comput. Phys.* 213 (1) (2006) 141–173.

- [16] Cédric Galusinski, Paul Vigneaux, On stability condition for bifluid flows with surface tension: application to microfluidics, *J. Comput. Phys.* 227 (12) (2008) 6140–6164.
- [17] J. Glimm, J. Grove, X. Li, K. Shyue, Y. Zeng, Q. Zhang, Three-dimensional front tracking, *SIAM J. Sci. Comput.* 19 (3) (1998) 703–727.
- [18] J. Glimm, E. Isaacson, D. Marchesin, O. McBryan, Front tracking for hyperbolic systems, *Adv. Appl. Math.* 2 (1) (1981) 91–119.
- [19] Arthur Guittet, Maxime Theillard, Frédéric Gibou, A stable projection method for the incompressible Navier–Stokes equations on arbitrary geometries and adaptive quad/octrees, *J. Comput. Phys.* 292 (2015) 215–238.
- [20] Daniel Hartmann, Matthias Meinke, Wolfgang Schröder, Differential equation based constrained reinitialization for level set methods, *J. Comput. Phys.* 227 (14) (2008) 6821–6845.
- [21] Daniel Hartmann, Matthias Meinke, Wolfgang Schröder, The constrained reinitialization equation for level set methods, *J. Comput. Phys.* 229 (5) (2010) 1514–1535.
- [22] Dalton J.E. Harvie, M.R. Davidson, Murray Rudman, An analysis of parasitic current generation in volume of fluid simulations, *Appl. Math. Model.* 30 (10) (2006) 1056–1066.
- [23] H. Ji, D. Chopp, J.E. Dolbow, A hybrid extended finite element/level set method for modeling phase transformations, *Int. J. Numer. Methods Eng.* 54 (8) (2002) 1209–1233.
- [24] Ken Kamrin, Chris H. Rycroft, Jean-Christophe Nave, Reference map technique for finite-strain elasticity and fluid–solid interaction, *J. Mech. Phys. Solids* 60 (11) (2012) 1952–1969.
- [25] A. Karakus, T. Warburton, M.H. Aksel, C. Sert, A gpu accelerated level set reinitialization for an adaptive discontinuous galerkin method, *Comput. Math. Appl.* 72 (3) (2016) 755–767.
- [26] Bruno Lafaurie, Carlo Nardone, Ruben Scardovelli, Stéphane Zaleski, Gianluigi Zanetti, Modelling merging and fragmentation in multiphase flows with surfer, *J. Comput. Phys.* 113 (1) (1994) 134–147.
- [27] Horace Lamb, *Hydrodynamics*, Cambridge University Press, 1932.
- [28] Liu Liu, Hongjie Yan, Guojian Zhao, Jiakai Zhuang, Experimental studies on the terminal velocity of air bubbles in water and glycerol aqueous solution, *Exp. Therm. Fluid Sci.* 78 (2016) 254–265.
- [29] Emmanuel Maitre, Thomas Milcent, Georges-Henri Cottet, Annie Raoult, Yves Usson, Applications of level set methods in computational biophysics, *Math. Comput. Model.* 49 (11–12) (2009) 2161–2169.
- [30] S. Marella, S.L.H.H. Krishnan, H. Liu, H.S. Udaykumar, Sharp interface cartesian grid method i: an easily implemented technique for 3d moving boundary computations, *J. Comput. Phys.* 210 (1) (2005) 1–31.
- [31] Chohong Min, On reinitializing level set functions, *J. Comput. Phys.* 229 (8) (2010) 2764–2772.
- [32] Chohong Min, Frédéric Gibou, Geometric integration over irregular domains with application to level-set methods, *J. Comput. Phys.* 226 (2007) 1432–1443.
- [33] Chohong Min, Frédéric Gibou, A second order accurate level set method on non-graded adaptive Cartesian grids, *J. Comput. Phys.* 225 (1) (2007) 300–321.
- [34] Jean-Christophe Nave, Rodolfo Ruben Rosales, Benjamin Seibold, A gradient-augmented level set method with an optimally local, coherent advection scheme, *J. Comput. Phys.* 229 (10) (2010) 3802–3827.
- [35] Taku Nonomura, Keiichi Kitamura, Kozo Fujii, A simple interface sharpening technique with a hyperbolic tangent function applied to compressible two-fluid modeling, *J. Comput. Phys.* 258 (2014) 95–117.
- [36] Elin Olsson, Gunilla Kreiss, A conservative level set method for two phase flow, *J. Comput. Phys.* 210 (1) (2005) 225–246.
- [37] Elin Olsson, Gunilla Kreiss, Sara Zahedi, A conservative level set method for two phase flow ii, *J. Comput. Phys.* 225 (1) (2007) 785–807.
- [38] Stanley Osher, Ronald Fedkiw, *Implicit functions*, in: *Level Set Methods and Dynamic Implicit Surfaces*, Springer, 2003, pp. 3–16.
- [39] Stanley Osher, James A. Sethian, Fronts propagating with curvature-dependent speed: algorithms based on Hamilton–Jacobi formulations, *J. Comput. Phys.* 79 (1) (1988) 12–49.
- [40] A. Ovsyannikov, V. Sabel’nikov, M. Gorokhovski, A new level set equation and its numerical assessments, in: *Proceedings of the Summer Program, 2012*, p. 315.
- [41] Mark Owkes, Olivier Desjardins, A discontinuous Galerkin conservative level set scheme for interface capturing in multiphase flows, *J. Comput. Phys.* 249 (2013) 275–302.
- [42] J-P. Pons, Gerardo Hermosillo, Renaud Keriven, O. Faugeras, Maintaining the point correspondence in the level set framework, *J. Comput. Phys.* 220 (1) (2006) 339–354.
- [43] Stéphane Popinet, Stéphane Zaleski, A front-tracking algorithm for accurate representation of surface tension, *Int. J. Numer. Methods Fluids* 30 (6) (1999) 775–793.
- [44] Nikolas Provatas, Ken Elder, *Phase-Field Methods in Materials Science and Engineering*, John Wiley & Sons, 2011.
- [45] Patrick Rasetarinera, M.Y. Hussaini, An efficient implicit discontinuous spectral Galerkin method, *J. Comput. Phys.* 172 (2) (2001) 718–738.
- [46] Jean-François Remacle, Nicolas Chevaugnon, Emilie Marchandise, Christophe Geuzaine, Efficient visualization of high-order finite elements, *Int. J. Numer. Methods Eng.* 69 (4) (2007) 750–771.
- [47] Robert D. Richtmyer, K.W. Morton, *Difference methods for initial-value problems*, *SIAM Rev.* 10 (3) (1968) 381–383.
- [48] Giovanni Russo, Peter Smereka, A remark on computing distance functions, *J. Comput. Phys.* 163 (1) (2000) 51–67.
- [49] Vladimir Sabelnikov, Andrey Yu. Ovsyannikov, Mikhael Gorokhovski, Modified level set equation and its numerical assessment, *J. Comput. Phys.* 278 (2014) 1–30.
- [50] Seungwon Shin, Damir Juric, Modeling three-dimensional multiphase flow using a level contour reconstruction method for front tracking without connectivity, *J. Comput. Phys.* 180 (2) (2002) 427–470.
- [51] Chi-Wang Shu, Stanley Osher, Efficient implementation of essentially non-oscillatory shock-capturing schemes, *J. Comput. Phys.* 77 (2) (1988) 439–471.
- [52] Irina Singer-Loginova, H.M. Singer, The phase field technique for modeling multiphase materials, *Rep. Prog. Phys.* 71 (10) (2008) 106501.
- [53] John Strain, Semi-Lagrangian methods for level set equations, *J. Comput. Phys.* 151 (2) (1999) 498–533.
- [54] M. Sussman, K.M. Smith, M.Y. Hussaini, M. Ohta, R. Zhi-Wei, A sharp interface method for incompressible two-phase flows, *J. Comput. Phys.* 221 (2) (2007) 469–505.
- [55] Mark Sussman, A second order coupled level set and volume-of-fluid method for computing growth and collapse of vapor bubbles, *J. Comput. Phys.* 187 (1) (2003) 110–136.
- [56] Mark Sussman, Ann S. Almgren, John B. Bell, Phillip Colella, Louis H. Howell, Michael L. Welcome, An adaptive level set approach for incompressible two-phase flows, *J. Comput. Phys.* 148 (1) (1999) 81–124.
- [57] Mark Sussman, Emad Fatemi, An efficient, interface-preserving level set redistancing algorithm and its application to interfacial incompressible fluid flow, *SIAM J. Sci. Comput.* 20 (4) (1999) 1165–1191.
- [58] Mark Sussman, Emad Fatemi, Peter Smereka, Stanley Osher, An improved level set method for incompressible two-phase flows, *Comput. Fluids* 27 (5) (1998) 663–680.
- [59] Mark Sussman, Elbridge Gerry Puckett, A coupled level set and volume-of-fluid method for computing 3d and axisymmetric incompressible two-phase flows, *J. Comput. Phys.* 162 (2) (2000) 301–337.

- [60] Maxime Theillard, Landry Fokoua Djodom, Jean-Léopold Vié, Frédéric Gibou, A second-order sharp numerical method for solving the linear elasticity equations on irregular domains and adaptive grids—application to shape optimization, *J. Comput. Phys.* 233 (2013) 430–448.
- [61] Maxime Theillard, Frédéric Gibou, Tresa Pollock, A sharp computational method for the simulation of the solidification of binary alloys, *J. Sci. Comput.* 63 (2) (2015) 330–354.
- [62] Maxime Theillard, Frédéric Gibou, David Saintillan, Sharp numerical simulation of incompressible two-phase flows, *J. Comput. Phys.* 391 (2019) 91–118.
- [63] G. Tryggvason, B. Bunner, A. Esmaeeli, D. Juric, N. Al-Rawahi, W. Tauber, J. Han, S. Nas, Y.-J. Jan, A front-tracking method for the computations of multiphase flow, *J. Comput. Phys.* 169 (2) (2001) 708–759.
- [64] Boris Valkov, Chris H. Rycroft, Ken Kamrin, Eulerian method for fluid-structure interaction and submerged solid-solid contact problems, 2015.
- [65] F. Xiao, Y. Honma, T. Kono, A simple algebraic interface capturing scheme using hyperbolic tangent function, *Int. J. Numer. Methods Fluids* 48 (9) (2005) 1023–1040.
- [66] Dongbin Xiu, George Em Karniadakis, A semi-Lagrangian high-order method for Navier–Stokes equations, *J. Comput. Phys.* 172 (2) (2001) 658–684.
- [67] Jian-Jun Xu, Hong-Kai Zhao, An Eulerian formulation for solving partial differential equations along a moving interface, *J. Sci. Comput.* 19 (1) (Dec 2003) 573–594.
- [68] Sara Zahedi, Martin Kronbichler, Gunilla Kreiss, Spurious currents in finite element based level set methods for two-phase flow, *Int. J. Numer. Methods Fluids* 69 (9) (2012) 1433–1456.