

It takes three to tango: 1. Simulating buoyancy-driven flow in the presence of large viscosity contrasts

Jenny Suckale

Jean-Christophe Nave

Bradford H. Hager

Abstract. Buoyancy-driven flow is of fundamental importance for numerous geodynamic phenomena. Since the equations of motion governing multi-phase flow are rarely amenable to analytical solutions, numerical simulations provide a compelling alternative. They offer the ability to carefully analyze flow phenomena under differing regimes, initial conditions, and flow dynamics. The three key challenges in these computations are (1) the accurate solution of the equations of motion in the presence of large viscosity contrasts, (2) the representation of strongly deforming interfaces between different fluids and (3) the accurate coupling of fluid- and interface solver. In three dimensions, these challenges become even more intricate, and the appropriate choice of numerical scheme has a profound influence on the tractability, accuracy, robustness, and efficiency of the computational simulation.

This is the first paper in a two-part series that examines numerical simulations of buoyancy-driven flow in the presence of large viscosity contrasts. In this paper, we present our numerical approach which tackles the above three main challenges through a combination of three numerical methods, namely (1) an extended ghost fluid discretization which we developed specifically for the Stokes regime, (2) the level set method, and (3) the extension velocity technique. We find that all three components are crucial to obtain a versatile numerical tool for simulating complex structures in evolving flow. We validate our code by reproducing four benchmark problems in two and three dimensions. We devote special attention to comparing our method to other existing techniques, detailing the advantages of this approach. Finally, we highlight several types of geophysical flow problems for which we believe our method to be well suited.

1. Introduction

Numerical modeling is an indispensable tool for understanding geophysical processes and an important complement to analytical models, which are inevitably simplified versions of the original problem. Although numerical models often approximate the original problem directly, each specific method comes with its own limitations and advantages. Thus, numerical benchmark studies are an important tool to evaluate and compare the performance of various approaches. The goal of this paper is to develop and test a numerical tool that correctly captures the complex flow dynamics of systems with large viscosity contrasts and strongly deforming interfaces.

The emphasis of this paper is on buoyancy-driven flow. This exclusive focus on buoyancy is justified by the fact that it is a common element in numerous geodynamical processes including salt diapirs (e.g. *Woidt* [1978]; *Schmeling* [1987]; *Römer and Neugebauer* [1991]; *Ismail-Zadeh et al.* [2004]), lithospheric instabilities (e.g. *Houseman et al.* [1981]; *Fleitout and Froidevaux* [1982]; *Conrad and Molnar* [1997]; *Schott et al.* [2000]; *Hoogenboom and Houseman* [2006]; *Elkins-Tanton* [2007]), interactions between compositional plumes (e.g. *Schaeffer and Manga* [2001]), trench motion in the context of self-consistent subduction models (e.g. *Royden and Husson* [2006]; *Schmeling et al.* [2008]), magma mixing (e.g. *Marsh* [1988]; *Bergantz and Ni* [1999]),

lava dome growth (e.g. *Bourgouin et al.* [2007]; *Hale et al.* [2007]; *Hale* [2008]), and bubble dynamics in magmatic flow (e.g. *Manga and Stone* [1995b]). Although temperature plays an important role in many of these processes, isothermal models are valuable as long as the time scale of thermal interaction is large compared to the time scale of buoyant ascent.

From a numerical perspective, simulations of buoyancy-driven flow pose three main challenges: (1) The computation of the flow field is complicated by the jump conditions at the interface. The material parameters (i.e., viscosity and density) jump discontinuously across the interface. These discontinuities in the coefficients of the equations of motion are paralleled by jumps in their solution (i.e., pressure and stresses). Since all of these jumps occur directly at the interface, numerical errors associated with the jump computation will translate quickly into less accurate interface dynamics. (2) The time-evolution of the interface needs to be tracked without restricting the deformability of the interface in the flow field and without violating mass conservation in either of the two fluids or affecting the discontinuous jumps in fluid properties across the interface. (3) The two solvers, fluid and interface solver, need to be fully and accurately coupled. We stress that this requirement is of special significance for buoyancy-driven flow in which motion is driven actively by the interface as compared to, for example, passive advection of an interface in a thermally-driven convection cell.

In this paper, we combine three tools, a ghost fluid type approach [*Fedkiw et al.*, 1999; *Kang et al.*, 2000; *Liu et al.*, 2000], the level set method [*Osher and Sethian*, 1988], and the extension velocity technique [*Malladi et al.*, 1995; *Adalsteinsson and Sethian*, 1999] to tackle each of the above

three challenges. We deal with the first challenge, the incorporation of the jump conditions in a sharp manner, by developing an extended discretization scheme for the multi-phase Stokes equation based on a ghost fluid type approach. Our implementation was inspired by *Kang et al.* [2000] and *Liu et al.* [2000]. To our knowledge, it provides the first application of ghost fluids to the Stokes equation. We address the second challenge by representing the interface through a level set function. The third challenge, the accurate coupling of flow and interface solver, is tackled through the velocity-extension procedure pioneered by *Malladi et al.* [1995] and later improved upon by *Adalsteinsson and Sethian* [1999] and *Chopp* [2001, 2009]. As discussed in detail elsewhere [*Sussman et al.*, 1998; *Adalsteinsson and Sethian*, 1999; *Chopp*, 2009; *Keck*, 2007] the construction of extension velocities ensures that our method is not prone to problems such as mass loss and spurious interface repositioning, which commonly plague codes relying on iterative re-initialization [*Sussman et al.*, 1994, 1998; *Sussman and Fatemi*, 1999].

Together, these three tools equip us with an accurate method of simulating buoyancy-driven flow in the presence of large viscosity contrasts. We have selected four benchmark problems to validate its different aspects and to compare its performance to other approaches: (1) The pressure jump due to surface tension at the interface of a spherical drop, (2) the jump in pressure and normal stresses at the interface of a circular inclusion which is several orders of magnitude less viscous than the surrounding rock matrix [*Schmid and Podladchikov*, 2003; *Deubelbeiss and Kaus*, 2008], (3) the isothermal Rayleigh-Taylor instability as specified by *Van Keken et al.* [1997], and (4) a compositional plume rising from a free-slip surface as investigated by *Manga et al.* [1993]. The first two benchmark problems serve as a verification for the accuracy of our multi-phase Stokes solver and its ability to resolve the various types of jump conditions that might arise in complex flow fields. The third benchmark problem, the isothermal Rayleigh-Taylor instability specified by *Van Keken et al.* [1997], allows us to study the relative advantages and disadvantages of our interface tracking technique in comparison to four other approaches for representing interfaces. Finally, we choose a compositional plume as our fourth and only three-dimensional benchmark problem in order to validate our computations with respect to experimental [*Manga et al.*, 1993] and numerical data [*Manga et al.*, 1993; *Schmalzl and Loddoch*, 2003].

This paper is structured in the following way: Section 2 reviews both the equation of motion and the jump conditions for buoyancy-driven flow. Section 3 discusses the basic idea behind and the implementation of the three main components of our code. Section 4 presents the four benchmark problems we have selected to verify our method, and section 5 the computations for each of these. Section 6 discusses our results with particular emphasis on comparing our approach to interface dynamics with previous methods, i.e., the tracer, marker-chain, and field approaches. Finally, section 7 briefly summarizes the main conclusions and identifies several geophysical problems for which we consider our method to be particularly promising.

2. Governing equations

The dynamics of buoyancy-driven flow involving two incompressible fluids with zero chemical diffusivity and negligible inertia are described by the Stokes equation

$$-\nabla p + \nabla \cdot \left(\mu(\Gamma) \left(\nabla \mathbf{v} + (\nabla \mathbf{v})^T \right) \right) + \mathbf{g}\rho(\Gamma) = 0, \quad (1)$$

with the incompressibility constraint,

$$\nabla \cdot \mathbf{v} = 0, \quad (2)$$

and the advection equation for composition

$$\frac{\partial \Gamma}{\partial t} + (\mathbf{v} \cdot \nabla) \Gamma = 0. \quad (3)$$

In these equations p denotes the pressure, μ the viscosity, ρ the density, \mathbf{v} the velocity field, \mathbf{g} the gravitational acceleration and Γ the composition, which captures the spatial dependence of ρ and μ in the presence of multiple fluid phases.

Additionally, the jump conditions at the interface due to the combined effect of surface tension and stresses need to be fulfilled:

$$\left[\begin{pmatrix} \mathbf{n} \\ \mathbf{t}_1 \\ \mathbf{t}_2 \end{pmatrix} (pI - \tau) \mathbf{n}^T \right] = \begin{pmatrix} \sigma \kappa \\ 0 \\ 0 \end{pmatrix}, \quad (4)$$

where I is the identity matrix, σ the surface tension coefficient, τ the stress tensor, κ the curvature of the interface, and $\mathbf{n}, \mathbf{t}_1, \mathbf{t}_2$ the normal and two tangential vectors to the interface. Note that we use square brackets to denote a jump in a given quantity throughout the paper.

Instead of tracking the discontinuous composition function Γ , we introduce an auxiliary function called the level set function ϕ . The level set function ϕ will be properly defined

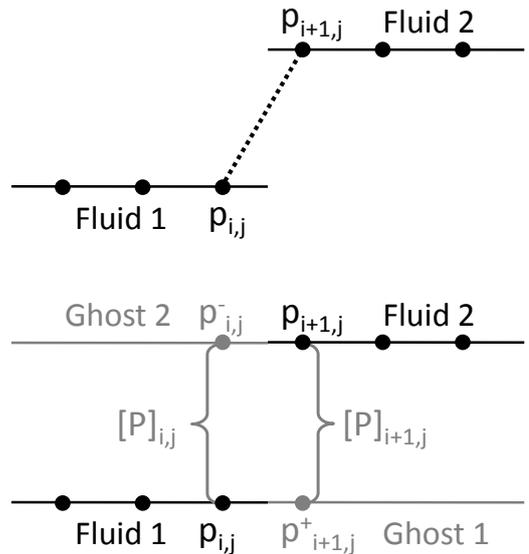


Figure 1. Comparison of the standard (top) and ghost fluid (bottom) construction of finite-difference stencils for computing the pressure gradient in the vicinity of the interface. The interface between fluid 1 and 2 is located between grid points (i, j) and $(i + 1, j)$. If the existence of the interface is not taken into account (top), the finite-difference approximation of the derivative $\partial P / \partial x \approx (p_{i+1,j} - p_{i,j}) / \Delta x$ will only be $O(1)$ accurate. Ghost fluid methods fictitiously extend each fluid into the domain of the other yielding two 'ghost' phases (bottom). After this extension, two values for pressure are associated with grid point $(i + 1, j)$, the physical $p_{i+1,j}$ and the ghost value $p_{i+1,j}^+$. The ghost phases fulfill the additional purpose of enforcing the jump conditions at the interface. The jump conditions $[P]_{i,j}$ and $[P]_{i+1,j}$ are computed on both sides of the interface and then interpolated to reflect the sub-grid position of the interface. This yields the jump at the interface denoted as $[P]_{I,j}$. The resulting finite-difference stencil $\partial P / \partial x \approx (p_{i+1,j} - p_{i,j}^- + [P]_{I,j}) / \Delta x$ is now order $O(\Delta x)$ accurate.

in section 3.2. At this point, we simply note that the spatial variation of material parameters can be written in the form $\rho(\Gamma) = \rho(\phi)$ and $\mu(\Gamma) = \mu(\phi)$.

In the absence of surface tension, the dynamics of the system are fully described by two non-dimensional numbers,

$$\Pi_1 = \frac{\mu_1}{\mu_2} \quad \text{and} \quad (5)$$

$$\Pi_2 = \frac{gd^3 \Delta\rho^2}{\mu_{\text{ref}}^2} \quad (6)$$

where $\Delta\rho$ represents the density difference between the two fluid phases and either of the two viscosities can be chosen for μ_{ref} . Π_2 represents the ratio of buoyancy forces to viscous drag. Introducing the characteristic length d , we substitute the following characteristic quantities $x = dx'$, $\mathbf{v} = \Delta\rho g d^2 / \mu_{\text{ref}} \mathbf{v}'$, $\mu = \mu_{\text{ref}} \mu'$, $\rho = \Delta\rho \rho'$, and $p = \mu_{\text{ref}}^2 / (\Delta\rho d^2) p'$ into eq. 1, drop the primes, and obtain the non-dimensional Stokes equation

$$-\nabla p + \Pi_2 \nabla \cdot \left(\mu(\phi) (\nabla \mathbf{v} + (\nabla \mathbf{v})^T) \right) + \Pi_2 \rho(\phi) \hat{\mathbf{z}} = 0, \quad (7)$$

where $\hat{\mathbf{z}}$ is the unit vector in vertical direction and

$$\rho(\phi) = \begin{cases} \rho_1 / \Delta\rho & \text{for } \phi < 0 \\ \rho_2 / \Delta\rho & \text{for } \phi \geq 0, \end{cases} \quad (8)$$

and

$$\mu(\phi) = \begin{cases} \Pi_1 & \text{for } \phi < 0 \\ 1 & \text{for } \phi \geq 0. \end{cases} \quad (9)$$

In the presence of surface tension, we additionally introduce the Bond number,

$$\Pi_3 = \frac{\Delta\rho g d^2}{\sigma} \quad (10)$$

which represents the ratio of buoyancy to surface tension forces.

3. Numerical method

3.1. Ghost-fluid-type fluid solver

3.1.1. Ghost fluid methods

Considering only finite-difference approaches on structured grids, the different numerical strategies that exist for capturing the jump conditions at dynamic interfaces can be grouped into three categories [Chern and Shu, 2007]: regularization (i.e., discontinuities in the coefficients and in singular sources are smeared out over one or more grid points), dimension-un-splitting (i.e., discontinuities are represented in a sharp manner based on a local Taylor-series expansion in multiple dimensions), and dimension-splitting techniques (i.e., discontinuities are represented in a sharp manner based on multiple local Taylor-series in one dimension thus representing the original problem as a combination of one-dimensional problems). The disadvantages of regularization approaches include artificial smearing of the jump conditions, formation of spurious oscillations for large jump magnitudes, and their limitation to first order accuracy in two or higher dimensions [Tornberg and Engquist, 2003]. Discretizations based on dimension un-splitting, on the other hand, typically match the jump conditions to second order at the interface (e.g. the immersed-interface method [Leveque and Li, 1994]). We note that an adaptation of the immersed-interface method to Stokes flow exists [Leveque and Li, 1997]. One drawback of dimension-un-splitting is that the construction of the finite-difference stencils is not a trivial undertaking, particularly in three dimensions. Ghost-fluid methods are part of the third category, dimension-splitting techniques. Since they rely on a combination of local Taylor-series expansions in one dimension, stencils typically consist of only three points in

each dimension (e.g. Kang et al. [2000]; Liu et al. [2000]) instead of 3^d grid points in d dimensions as required for dimension-unsplitting. The main drawback is that the splitting of dimensions requires projecting the jump conditions in the derivatives onto the direction normal to the interface. As a consequence, the jumps in the normal derivative are captured correctly and in a sharp manner, but those in the tangential derivative are still smeared out (e.g. [Liu et al., 2000; Chern and Shu, 2007]). However, since the tangential component of the derivative jump condition typically contributes little to the overall jump in buoyancy-driven flow, we suggest that ghost fluid methods provide a reasonable compromise between accuracy and computational effort for this specific class of problems.

The key idea behind ghost-fluid techniques – and the origin of their name – is the fictitious extension of the physical domain of each fluid into the physical domain of the other fluid (e.g. Glimm et al. [1981]; Mayo [1984, 1985]; Fedkiw et al. [1999]; Kang et al. [2000]; Liu et al. [2000]). We build the ghost phases through linear extrapolation of the physical phase [Kang et al., 2000; Liu et al., 2000]. They fulfill two purposes (see Figure 1): (1) They are used to construct more accurate stencils in the vicinity of the interface, which consist of a mixture of ghost and physical points. The gain in accuracy is related to the fact that the interface was essentially removed from the stencil by constructing the linear extension of the fluid. (2) They are used to enforce the jump conditions at the interface, which are determined by harmonic interpolation taking the subgrid location of the interface into account. Thus, ghost-fluid methods locally decouple a single two-phase flow problem into two single-phase flow problems that are then merged back together based on the jump conditions at the interface.

Figure 1 illustrates the construction of finite-difference stencils for the pressure gradient (eq. 7). Although the strategy for building the stencils for the viscous term in equation 7 is similar in essence, we use asymmetric stencils for additional accuracy around the interface [Kang et al., 2000; Liu et al., 2000]. Figure 2 explains the construction of the asymmetric three-point stencils approximating the second derivative of the velocity. The central point $u_{i,j}$ of the standard symmetric stencil is shifted to coincide exactly with the interface $u_{I,j}$. The viscosity of the ‘asymmetric’ stencil point $u_{I,j}$ is computed through harmonic interpolation [Kang et al., 2000; Liu et al., 2000]

$$\hat{\mu} = \frac{\mu_1 \mu_2}{\mu_2 \theta + \mu_1 (1 - \theta)}, \quad (11)$$

where θ denotes the subgrid position of the interface.

3.1.2. Jump conditions

Evidently, the accuracy of ghost-fluid methods hinges critically on the accuracy with which the jump conditions are computed. Three types of jump conditions need to be considered (eq. 4): (1) the jump in pressure due to surface tension, (2) the jump in pressure due to viscous stresses, and (3) the jump in stresses. As noted by Kang et al. [2000], the pressure jump in equation 4 can be rephrased more simply as

$$[p] = \sigma \kappa + 2[\mu](\nabla u \cdot \mathbf{n}, \nabla v \cdot \mathbf{n}, \nabla w \cdot \mathbf{n}) \cdot \mathbf{n}. \quad (12)$$

The first term in equation 12 is the surface-tension-related contribution to the pressure jump and the second term represents the viscous contribution. In addition to the pressure, stresses may also be discontinuous across the interface, implying that jump conditions need to be considered when approximating the second derivative of the velocity field. The computation of the stress jumps is based on the insight that the discontinuity of normal stresses can be avoided through

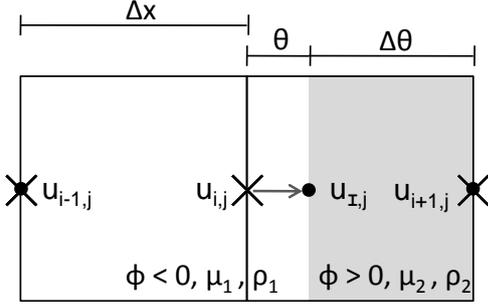


Figure 2. Illustration of the construction of the asymmetric finite-difference stencil for the second derivative of the velocity field in one dimension. Two neighboring grid cells of size Δx are shown. The two fluids are shaded in grey and white, respectively, implying that the interface crosses only the grid cell on the right. The symmetric three-point stencil commonly used to compute the second derivative of the velocity at point $u_{i,j}$ is indicated by crosses. In order to take the subgrid position of the interface explicitly into account, we shift the central point in the symmetric stencil $u_{i,j}$ to coincide with the interface $u_{I,j}$. This leads to an asymmetric stencil indicated by black dots and spanned by the points $u_{i-1,j}$, $u_{i,j}$ and $u_{i+1,j}$.

a coordinate transform [Leveque and Li, 1994]. Hence, we compute the stress jump in the coordinate system spanned by the normal and the two tangential vectors at each point of the interface, yielding a matrix of jump conditions that is then rotated back into the computational domain. This strategy leads to the following expression [Kang et al., 2000]

$$\begin{pmatrix} [\mu u_x] & [\mu u_y] & [\mu u_z] \\ [\mu v_x] & [\mu v_y] & [\mu v_z] \\ [\mu w_x] & [\mu w_y] & [\mu w_z] \end{pmatrix} = [\mu] \begin{pmatrix} \nabla u \\ \nabla v \\ \nabla w \end{pmatrix} \begin{pmatrix} \mathbf{0} \\ \mathbf{t}_1 \\ \mathbf{t}_2 \end{pmatrix}^T \begin{pmatrix} \mathbf{0} \\ \mathbf{t}_1 \\ \mathbf{t}_2 \end{pmatrix} + [\mu] \mathbf{n}^T \mathbf{n} \begin{pmatrix} \nabla u \\ \nabla v \\ \nabla w \end{pmatrix} \mathbf{n}^T \mathbf{n} - [\mu] \begin{pmatrix} \mathbf{0} \\ \mathbf{t}_1 \\ \mathbf{t}_2 \end{pmatrix}^T \begin{pmatrix} \mathbf{0} \\ \mathbf{t}_1 \\ \mathbf{t}_2 \end{pmatrix} \begin{pmatrix} \nabla u \\ \nabla v \\ \nabla w \end{pmatrix}^T \mathbf{n}^T \mathbf{n} \quad (13)$$

We note that contrary to the discontinuous derivatives on the left, the derivatives on the right hand side of equation 13 are spatially continuous and thus amenable to a finite-difference approximation [Kang et al., 2000]. For a detailed derivation of equations 12 and 13, please refer to the on-line supplement or to Kang et al. [2000]. As is evident from equations 12 and 13, the jump conditions themselves depend on the velocity field. Therefore, it is not obvious how to compute the jump conditions without knowing the solution to the equations of motion beforehand. In their ghost-fluid method for solving the multi-phase Navier-Stokes equation, Kang et al. [2000] suggest using the numerical estimate of the velocity field from the previous time step to compute the jump conditions at the current time step. This strategy is not an option for the Stokes equation, because it conflicts with the assumption of quasi-stationarity inherent to Stokes flow. Instead, we solve for the jump conditions at each time step as explained next.

3.1.3. Discretization of the multi-phase Stokes equation in a ghost framework

We use finite differences on a staggered grid in Cartesian coordinates to approximate the non-dimensionalized multi-phase Stokes equation (eq. 7). The coefficients of the Stokes equation (i.e., density and viscosity) are assigned at each time step based on the new position of the interface, which yields a sharp representation of the discontinuity in material properties without introducing artificial smearing (eqs. 8 and 9).

As discussed in section 3.1.1, the basic idea of ghost-fluid methods is to locally decouple a single multi-phase flow computation into multiple single-phase computations. One key advantage of this strategy is that the incompressibility constraint (eq. 2) can be used to simplify the viscous term

$$\nabla \cdot (\mu(\phi)(\nabla \mathbf{v} + (\nabla \mathbf{v})^T)) = \mu(\phi) \nabla^2 \mathbf{v} \quad (14)$$

which leads to a standard saddle-point problem

$$\begin{pmatrix} -\mu(\phi) \nabla^2 & \text{grad} \\ -\text{div} & 0 \end{pmatrix} \begin{pmatrix} \mathbf{v} \\ p \end{pmatrix} = \begin{pmatrix} \mathbf{g}\rho(\phi) \\ 0 \end{pmatrix} \quad (15)$$

when ignoring the jump conditions. The three jump conditions, the pressure jump due to surface tension $[p]_{st}$, the viscous contribution to the pressure jump $[p]_v$, and the jump in the stresses $[\nabla \mathbf{v}]$, scale differently (see eqs. 12 and 13)

$$[p]_{st} \sim \kappa, \quad (16)$$

$$[p]_v \sim \nabla \mathbf{v}, \quad \text{and} \quad (17)$$

$$[\nabla \mathbf{v}] \sim \nabla \mathbf{v}, \quad (18)$$

where κ is the curvature of the interface. Since $[p]_{st}$ does not depend on the velocity field, it can be incorporated into the linear system 15 as a source term on the right hand side [Kang et al., 2000; Liu et al., 2000]. The other two jumps, $[p]_v$ and $[\nabla \mathbf{v}]$, however, need to be solved for at each time step (sec. 3.1.2) yielding the linear system

$$\left(\begin{pmatrix} -\mu(\phi) \nabla^2 & \text{grad} \\ -\text{div} & 0 \end{pmatrix} + \begin{pmatrix} [\nabla \mathbf{v}] & [p]_v \\ 0 & 0 \end{pmatrix} \right) \begin{pmatrix} \mathbf{v} \\ p \end{pmatrix} = \begin{pmatrix} \mathbf{g}\rho(\phi) \\ 0 \end{pmatrix} - \begin{pmatrix} [p]_{st} \\ 0 \end{pmatrix}. \quad (19)$$

It is worth mentioning that the three jump conditions are not equally challenging to resolve numerically. The easiest jump to compute is $[p]_{st} = \sigma \kappa$ because it does not depend on the velocity field and because the accurate computation of the curvature κ is ensured by representing the interface through a level set function (sec. 3.2). The viscous contribution to the pressure jump

$$[p]_v = 2[\mu](\nabla u \cdot \mathbf{n}, \nabla v \cdot \mathbf{n}, \nabla w \cdot \mathbf{n}) \cdot \mathbf{n}, \quad (20)$$

and the stress jumps $[\nabla \mathbf{v}]$ (eq. 13) are more challenging.

We note that in the presence of viscosity contrasts of several orders of magnitudes and/or jump conditions of several orders of magnitude, the condition number of the extended Stokes matrix in the linear system (eq. 18) can become a concern. Most computations in this paper are based either on backslash or the direct solver PARDISO [Schenk and Gärtner, 2004, 2006]. In terms of iterative solvers, GMRES or stabilized BICG in combination with an incomplete-LU preconditioner seem to be reliable alternatives.

3.2. Level-set-based Interface solver

Interface-tracking methods can be divided into two classes, explicit and implicit. Before delving into the details, we attempt to offer an intuitive motivation for these two types of approaches following Sethian [1997]: Suppose

our task were to track the coastline of a lake that experiences strong variations with water level. One possibility would be to monitor the driftwood along the lake shore. However, if water-level variations were large and rapid and the lake-bed topography was rough, the driftwood would partly wash ashore and a large amount of driftwood would be necessary to obtain an accurate estimate of the coastline. The level set method takes a different approach. Instead of tracking the shore of the lake directly, we draw a topographic map of the lake bed. We adjust the datum of this map such that the coastline always corresponds exactly to the zero-elevation contour. In order to find out the precise location of the interface at a given time, we simply plot the zero contour of the corresponding level set function.

This example illustrates that instead of finding an explicit representation of the interface (i.e., placing markers along it), the level set method takes an implicit approach and embeds the interface into a higher-dimensional function (i.e., the topographic map) such that it corresponds to the zero-contour of that function. The main advantage of this implicit formulation is that geometric complexities of highly convoluted interfaces or topological changes resulting from merging or ripping of interfaces can be captured easily. Also, the approach generalizes in a straightforward manner to higher dimensions, contrary to some explicit front representations (e.g. marker chains) for which three dimensions represent a substantial challenge.

In a static problem, the only property required for the auxiliary higher-dimensional function is that its zero-contour must correspond to the real interface. In a seminal paper, *Osher and Sethian* [1988] developed a framework for adding dynamics to implicit surfaces. Dynamics pose new constraints on the auxiliary higher-dimensional function. In particular, it turns out to be advantageous if the level set function is constructed as a signed distance function. This entails that it is negative in one fluid, positive in the other, and zero exactly at the interface. Furthermore, its absolute value at any point corresponds to the minimum distance of that point to the interface. Functions that fulfill these properties will be referred to as level set functions denoted by ϕ . From this definition it follows that a level set function ϕ is normalized

$$|\nabla\phi| = 1. \quad (21)$$

Examples of level set functions are plotted in the online supplement.

Given a velocity field \mathbf{v} , the propagation of the level set function ϕ is described by the advection equation:

$$\frac{\partial\phi}{\partial t} + \mathbf{v} \cdot \nabla\phi = 0, \quad (22)$$

also referred to as the level set equation [*Osher and Sethian*, 1988]. Equation 22 shows that the level set method turns a Lagrangian front propagation problem into an Eulerian initial value partial differential equation. At this point we have intentionally not specified the nature of the velocity field \mathbf{v} in equation 22. One of our goals will be to construct an appropriate velocity field that maintains the signed distance property (eq. 21) as the level set function moves (sec. 3.3), while continuing to link the zero level set to the proper interface motion.

The numerical solution of equation 22 requires high-order schemes. For the spatial discretization we implement the 5th-order-accurate WENO scheme [*Liu et al.*, 1994; *Jiang and Shu*, 1996; *Jiang and Peng*, 2000] and for the temporal discretization the 3rd-order-accurate, Total-Variation-Diminishing (TVD) Runge-Kutta scheme [*Shu and Osher*, 1988].

The time step used in the computation is restricted through the Courant-Friedrichs-Lewy (CFL) criterion associated with equation 22

$$\Delta t < \frac{\Delta x}{|\mathbf{v}|_{\max}}, \quad (23)$$

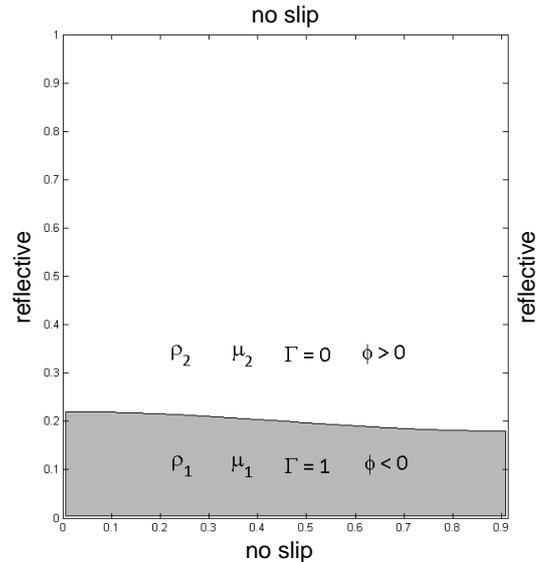


Figure 3. Initial and boundary conditions for the Rayleigh-Taylor instability as specified by *Van Keken et al.* [1997]. The fluids are characterized by different densities ρ_i and viscosities μ_i . The level set function is constructed such that it is negative within the buoyant fluid and positive outside.

where Δt and Δx are the temporal and spatial grid spacing and $|\mathbf{v}|_{\max}$ is the maximum velocity. In the computations we typically use $\Delta t = 0.5 \Delta x / |\mathbf{v}|_{\max}$ unless convergence requires higher temporal resolution.

3.3. Coupling of the two solvers

Both the fluid (sec. 3.1) and the interface (sec. 3.2) solver are Eulerian and each is based on a Cartesian grid with the same grid resolution Δx . Thus, one might expect that the coupling of the two solvers should be straight-forward, which is only partially true. While the forward coupling of the new interface position into the fluid solver works in the obvious way (i.e. by updating eqs. 8 and 9), the coupling of the flow field back to the interface solver requires more subtlety in the context of level set methods. Let us return to the question of which velocity field \mathbf{v} to use in equation 22. Intuitively, the most obvious choice might be the physical velocity field resulting from the solution of the Stokes equation. However, as pointed out by *Sethian* [1999a], using the fluid velocity itself to update the level set function is often a poor choice, since it necessarily creates shearing and distortion of the level set function unless the strain tensor of the flow field vanishes. While possible, continuous restoration of the level set function through an iterative re-initialization procedure such as described by *Sussman et al.* [1994] leads to several numerical problems such as spurious mass loss and artificial front repositioning [*Adalsteinsson and Sethian*, 1999; *Sethian*, 1999a; *Chopp*, 2009; *Keck*, 2007]. A more detailed study of the potential bias introduced through iterative re-initialization is available in the online supplement.

The need to re-initialize the level set function is a consequence of choosing the physical velocity field for solving the level set equation 22, but this is certainly not the only valid choice. In fact, the velocity field used to advect the level set function need be that given by the physics of the problem only for the zero contour – not for any other contour. Thus, analogous to the procedure of embedding the interface into the higher-dimensional level set function, we are free to embed the velocity of the interface into a higher-dimensional velocity function, called the extension velocity

field. It is constructed based on two criteria [Adalsteinsson and Sethian, 1999]: (1) The extension velocity at the zero contour has to be identical to the physical velocity, and (2) the extension velocity for all other points is chosen such that it ensures maintenance of the normalization criterion (eq. 21). It is straight-forward to show (e.g. Sethian [1999a]) that the normalization condition will be maintained if

$$\nabla\phi \cdot \nabla\mathbf{v}_{\text{ext}} = 0. \quad (24)$$

Numerically, we compute the extension velocity function as recommended by Adalsteinsson and Sethian [1999] through the fast-marching method [Sethian, 1996, 1999b].

4. Benchmark problems

4.1. Pressure jump at the interface of a viscous drop

Surface tension causes pressure to jump at the interface of a spherical drop as can be derived either from thermodynamic free energy considerations or from the Young-Laplace equation. For a static drop of radius a and surface tension σ the pressure jump is given by

$$[p]_{st} = \frac{2\sigma}{a}. \quad (25)$$

This expression highlights that the effect of surface tension is small for medium-sized drops, but may become substantial as the drop radius decreases.

We investigate the pressure jump (eq. 25) both for the case of a static drop and that of a dynamic drop rising under its own buoyancy in a hydrostatic ambient pressure field. In the first case, the pressure jump should be given by a discontinuous step function; in the second case, there will be an additional contribution from the ambient pressure field. We use 'periodic' boundary conditions on the sides of the computational domain and 'no slip' on the top and bottom walls. The viscosity contrast is set to $\Pi_1 = 10^{-6}$ and the Bond number to $\Pi_3 = 10^{-3}$ to ensure sphericity.

4.2. Pressure jump for weak inclusions in pure shear

Schmid and Podladchikov [2003] derived analytical solutions for the pressure and velocity fields of elliptical inclusions that are subject to shearing. A viscosity contrast of several orders of magnitude is assumed between the inclusion and the surrounding rock matrix. The shear boundary condition in combination with the viscosity contrast creates a discontinuous jump in stresses $[\nabla\mathbf{v}]$ and pressure $[p]_v$ as derived in eqs. 13 and 20.

We compute the pressure and velocity field around a circular inclusion of non-dimensional radius $r = 0.1$ inside a 1×1 computational domain. The boundary conditions in the far-field are set to pure shear with a strain rate of $\dot{\epsilon} = 1$. The velocity at the four edges of the computational domain is computed analytically based on equations 23-26 in Schmid and Podladchikov [2003]. We investigate viscosity contrasts of $\Pi_1 = 10^{-3} - 10^{-6}$, where the inclusions is assumed to have the lower viscosity. The density is constant in both domains. We compare our computations both to the analytical solution and to the numerical results by Deubelbeiss and Kaus [2008].

4.3. Rayleigh-Taylor instability

Van Keken et al. [1997] performed a careful benchmark study of the Rayleigh-Taylor instability. The dynamics are specified through the two non-dimensional numbers Π_1 and Π_2 derived in section 2. In this paper, we study both the isoviscous case ($\Pi_1 = 1$) and that of a viscosity contrast of $\Pi_1 = 10$. The dimensions of the computational domain are

$[0, \lambda] \times [0, 1]$ where $\lambda = 0.9142$ is chosen such that a harmonic perturbation with wavelength 2λ yields the largest growth rate. The thickness of the buoyant layer is $d = 0.2$ and the initial deflection of the interface between the two layers is $w = 0.02 \cos(\pi x/\lambda)$. The boundary conditions of the box are assumed to be 'no slip' on the top and bottom of the box and 'reflective' on the side walls. These initial and boundary conditions for the Rayleigh-Taylor instability are illustrated in Figure 3.

Van Keken et al. [1997] defined four quantitative parameters characterizing the dynamics of the Rayleigh-Taylor instability: (1) the initial growth rate γ , which is computed based on the growth of the amplitude h of the interface

$$h(t) = h(0)e^{\gamma t}, \quad (26)$$

evaluated at $t \simeq 0$, (2) the maximum of the root-mean-square velocity

$$v_{\text{rms}} = \sqrt{\frac{1}{V} \int_V \|\mathbf{v}\|^2}, \quad (27)$$

where V is the area of the computational domain, (3) the time $t(v_{\text{max}})$ at which the maximum root-mean-square velocity v_{rms} is achieved, and (4) the relative entrainment e of the source layer above a specified height $d = 0.2$

$$e = \frac{1}{\lambda d} \int_d^1 \Gamma \, dV. \quad (28)$$

4.4. Compositional plume

Manga et al. [1993] studied the rise of a compositional plume from a free-slip surface both experimentally and through two-dimensional boundary-element simulations. The experiments were performed with colored corn syrup in colorless glycerine and are reported in terms of non-dimensional times [Manga et al., 1993]. The initial condition is a semi-spherical blob resting on the free-slip surface. In addition to the experimental and numerical data by Manga et al. [1993], Schmalzl and Loddoch [2003] presented a numerical solution to this problem in three dimensions.

The computational domain in our simulation is a rectangular box of non-dimensional size $4 \times 4 \times 5$. The initial condition is a semi-spherical blob of radius 1 placed in the center of the box. The density and viscosity contrasts between the two fluids are set to 0.15 and 1, respectively. The boundary conditions are 'periodic' on all four sides of the box and 'free slip' on the top and bottom wall. The 'free-slip' condition is implemented as $\mathbf{v} \cdot \mathbf{n} = 0$ and $\nabla\mathbf{v} \cdot \mathbf{t}_1 = \nabla\mathbf{v} \cdot \mathbf{t}_2 = 0$.

5. Results

5.1. Pressure jump at the interface of a viscous drop

The linear dependence of the pressure jump $[p]_{st}$ on drop radius (eq. 25) is straight-forward to reproduce numerically. Table 1 details a dimensional comparison of the analytical pressure jump based on equation 25 and our numerical results. An example computation of a static drop with Bond number $\Pi_3 = 10^{-3}$ and viscosity ratio $\Pi_1 = 10^{-6}$ in a constant ambient pressure field is shown in Figure 4. Despite the relatively low resolution of the computation (51×51), the pressure jump is resolved as a sharp discontinuity without artificial smearing or spurious oscillations. A more detailed convergence test for this case is included in the online supplement.

Table 1. Analytical versus numerical results for the pressure jump due to surface tension at the interface of a static drop. This computation was done dimensionally with $\sigma = 0.0728$ kg/s² – a representative value for the air/water interface.

	r = 1dm	r = 1cm	r = 1mm	r = 0.1mm
$[p]_{st}$ in Pa (analyt.)	1.456	14.56	145.6	1456
$[p]_{st}$ in Pa (numer.)	1.458	14.59	146.0	1462
Deviation	< 1%	< 1%	< 1%	< 1%

Figure 5 shows the equivalent computation for a dynamic drop rising under its own buoyancy. The pressure field reflects the hydrostatic and dynamic contributions to pressure as well as the effect of surface tension. Contrary to Figure 4, we now not only observe a jump in pressure itself, but also a jump in the first derivative of pressure. We note that the dynamic case is more challenging than the static case, because it requires both approximating the static pressure jump correctly and maintaining it as a sharp discontinuity over time.

5.2. Pressure jump for weak inclusions in pure shear

Figure 6 shows the pressure field for a circular inclusion with a viscosity contrast of $\Pi_1 = 10^{-3}$ compared to the surrounding rock matrix. The left panel represents the numerical solution and the middle panel the analytical solution by *Schmid and Podladchikov* [2003]. Visually, there is no discernable difference between the numerical and the analytical solution. The right panel in Figure 6 shows the percentage error of the numerical with respect to the analytical solution. Not surprisingly, the highest error occurs at the interface, more specifically at finite inclinations with respect to the axes $x = 0$ and $y = 0$. This error is not unexpected given that the tangential component of the stress

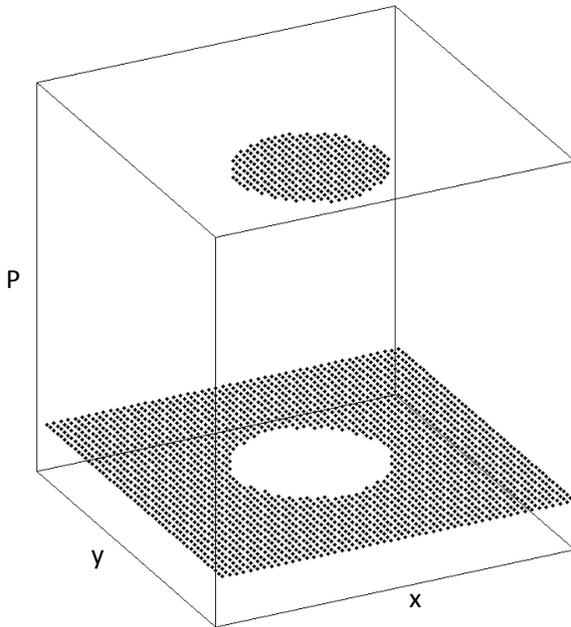


Figure 4. Computation of the pressure jump inside a static drop as a consequence of surface tension (benchmark problem 1). The Bond number (eq. 10) is set to $\Pi_3 = 10^{-3}$ to ensure sphericity of the drop. The viscosity contrast is $\Pi_1 = 10^{-6}$. The point-wise visualization of the pressure field illustrates the original resolution of the simulation (51×51). The computational domain is a square box of aspect ratio 1.

jump is not captured in a sharp manner, but instead smeared over the length of a grid cell (see section 6.1). At the given grid resolution of 280×280 , the percentage error drops to below 1% in locations where only the normal component contributes to the stress jump as compared to a maximum percentage error of 3.7 % where the tangential component contributes significantly. The average percentage error in the computational domain is $< 1\%$. In comparison to a previous study of the inclusion problem [*Deubelbeiss and Kaus, 2008*], three aspects of our numerical solution merit attention: (1) Contrary to *Deubelbeiss and Kaus* [2008], there are no spurious pressure oscillations at the interface of the inclusion, not even at minimal resolution (see online appendix). This aspect is of greatest importance for dynamic problems as spurious oscillations tend to build up over time. (2) We are able to reproduce a sharp transition from a highly variable pressure field in the matrix to a constant pressure field inside the inclusion. We do not observe any artificial distortions to the constant pressure field inside the inclusion as is the case in *Deubelbeiss and Kaus* [2008]. (3) The overall error is lower in our computation implying that less resolution is required to accurately resolve the pressure field. In fact, a grid resolution of 80×80 is sufficient to achieve a lower overall L2-error than the best performing method by *Deubelbeiss and Kaus* [2008] at a grid resolution of 280×280 .

In comparison to benchmark problem 1 (sec. 5.1), our simulations confirm that the jump conditions $[p]_v$ and $[\nabla \mathbf{v}]$ are more challenging to resolve than $[p]_{st}$ (sec. 3.1.2). As detailed in the online appendix, the additional challenge lies in computing the local magnitude of the jump along the interface correctly.

5.3. Rayleigh-Taylor instability

We were able to reproduce the evolution of the Rayleigh-Taylor instability as specified by *Van Keken et al.* [1997] to high precision. Figure 7 shows a comparison of the isoviscous

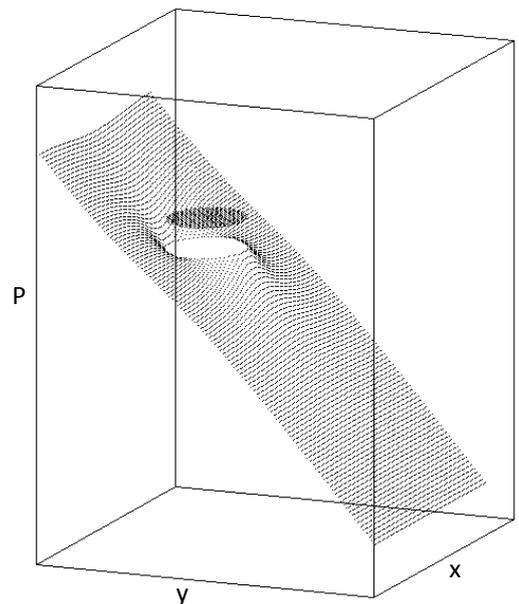


Figure 5. Computation of the pressure field associated with a dynamic drop rising under its own buoyancy. The Bond number (eq. 10) is set to $\Pi_3 = 10^{-3}$ to ensure sphericity of the drop. The point-wise visualization of the pressure field highlights the original resolution of the simulation (81×121). The computational domain is a rectangular box of aspect ratio 2×3 . The pressure field is a combination of the discontinuous jump at the interface and the hydrostatic and dynamic contributions outside.

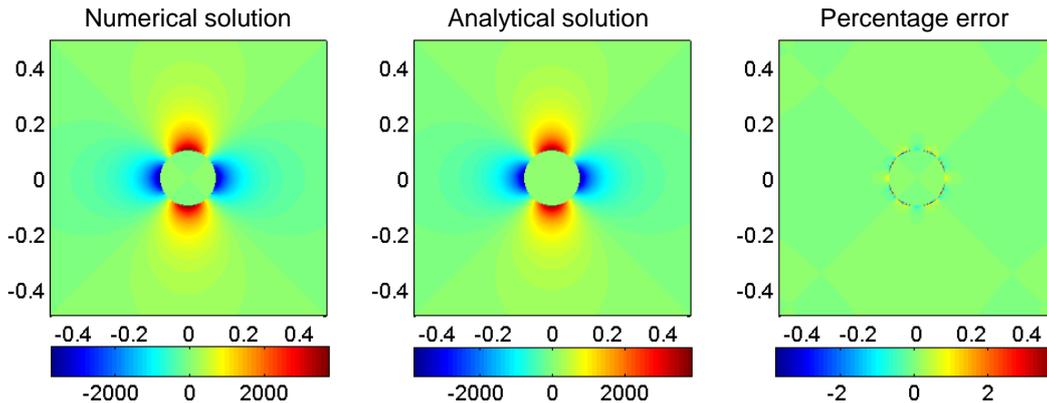


Figure 6. Comparison of the pressure field obtained numerically (left), computed analytically (middle) and the percentage error of the numerical solution (right) for benchmark problem 2. The average percentage error is $< 1\%$, the maximum is 3.76% . For easier comparison with *Deubelbeiss and Kaus* [2008], the grid resolution in the computation is set to 280×280 and the viscosity contrast between inclusion μ_{in} and surrounding matrix μ_m to $\Pi_1 = \mu_{in}/\mu_m = 10^{-3}$.

Rayleigh-Taylor instability computed by five different codes. The top plot is the level-set-based code presented in this paper at a grid resolution of 300×330 . A detailed convergence study is available in the online supplement. The other four plots are reconstructed based on the work published in *Van Keken et al.* [1997]. The authors kindly made the data available online. We observe the closest agreement of our results with those computed by the marker-chain method. A detailed comparison of the two interfaces is given in Figure 8. The temporal evolution of the root-mean-velocity v_{rms} and the entrainment e (Figs. 9 and 10) confirms the visual impression that the two methods match well. Table 2 provides an overview of the quantitative parameters characterizing the evolution of the instability, i.e. initial growth rate γ , maximum root-mean-square velocity, and the time $t(v_{max})$ at which it is achieved. The cited values are from Table 1 in *Van Keken et al.* [1997].

The long-term evolution of the isothermal Rayleigh-Taylor instability in the non-isoviscous case is more challenging to resolve, not so much because of the viscosity contrast itself, but rather because of the fact that the interface thins out rapidly. Simulations of buoyancy-driven flow require a full two-way coupling of fluid and interface solver, implying that the minimal resolution required is determined by the thickness of the fluid film. Thus, the computational expense scales inversely with the minimal thickness of the fluid film. For additional details, please refer to the online appendix.

5.4. Compositional plume

Figure 11 gives snapshots of our three-dimensional simulation of an initially semi-spherical compositional plume rising from a free-slip surface. The computation was performed on a $40 \times 40 \times 50$ grid. The experimental results by *Manga et al.* [1993] are included for comparison purposes in the background. Qualitatively, experimental and numerical results match well, with the possible exception of the initial condition. This slight deviation resulting from non-zero surface tension in the experiment has little consequence for the long-term evolution of the instability [*Manga et al.*, 1993]. Our computations also agree well with prior numerical results by *Manga et al.* [1993] and *Schmalzl and Loddoch* [2003].

6. Discussion

6.1. Evaluation of the fluid solver

To our knowledge, the extended ghost-fluid-type discretization of the multi-phase Stokes equation presented in

this paper provides the first adaptation of a ghost-fluid-type scheme to the multi-phase Stokes equation. Benchmark problem 1 (sec. 4.1 and 5.1) demonstrates that our method is able to resolve the pressure jump resulting from surface tension acting at the interface of a spherical drop in both a static (Fig. 4) and a dynamic (Fig. 5) setting even at low grid resolutions (see online supplement).

Benchmark problem 2 (sec. 4.2 and 5.2) establishes the ability of our ghost-fluid-type fluid solver to resolve a pressure jump of multiple orders of magnitude at the interface of a circular inclusion. Contrary to previous methods [*Deubelbeiss and Kaus*, 2008], our numerical solution does not suffer from spurious pressure oscillations and introduces neither smearing in the pressure jump nor erroneous deviations from the constant pressure field inside the inclusion. In addition, the overall numerical error is lower implying that less resolution is required to obtain a satisfactory level of accuracy.

Another important difference between the regularization-type methods discussed by *Deubelbeiss and Kaus* [2008] and the ghost-fluid-type method developed in this paper is how error scales with increasing viscosity contrast. While the accuracy of regularization methods typically decreases with increasing viscosity contrast, ghost fluid methods can handle viscosity contrasts of numerous orders of magnitude. The reason for this favourable scaling lies in the discretization of the jump conditions (eqs. 13 and 12): In both of these equations, the viscosity jump $[\mu]$ is separated out from the velocity-dependent contribution to the jump. Since the velocity-dependent contribution is not discontinuous at the interface, it can be approximated through finite differences and a larger viscosity contrast will not affect the accuracy with which this term is resolved. Consequently, the maximum percentage error of the computed pressure field is largely independent of the viscosity contrast so long as the condition number of the associated linear system (eq. 18) does not render it un-solvable (please refer to the online appendix for more details).

The absence of any signs of numerical diffusion and/or spurious oscillations is notable in Figure 5, Figure 6 and also in benchmark problems 3 and 4. The absence of numerical diffusion is largely related to the level set representation of the interface and will be discussed in that context (sec. 6.2). The formation and possible build-up of spurious oscillations as a consequence of dispersive errors is avoided through a combination of three tools: (1) the construction of more accurate, ghost-fluid-based stencils in the immediate vicinity of the interface (sec. 3.1.1), which reduce dispersive error and implicitly enforce the jump conditions in the flow field, (2) the level set representation of the interface in

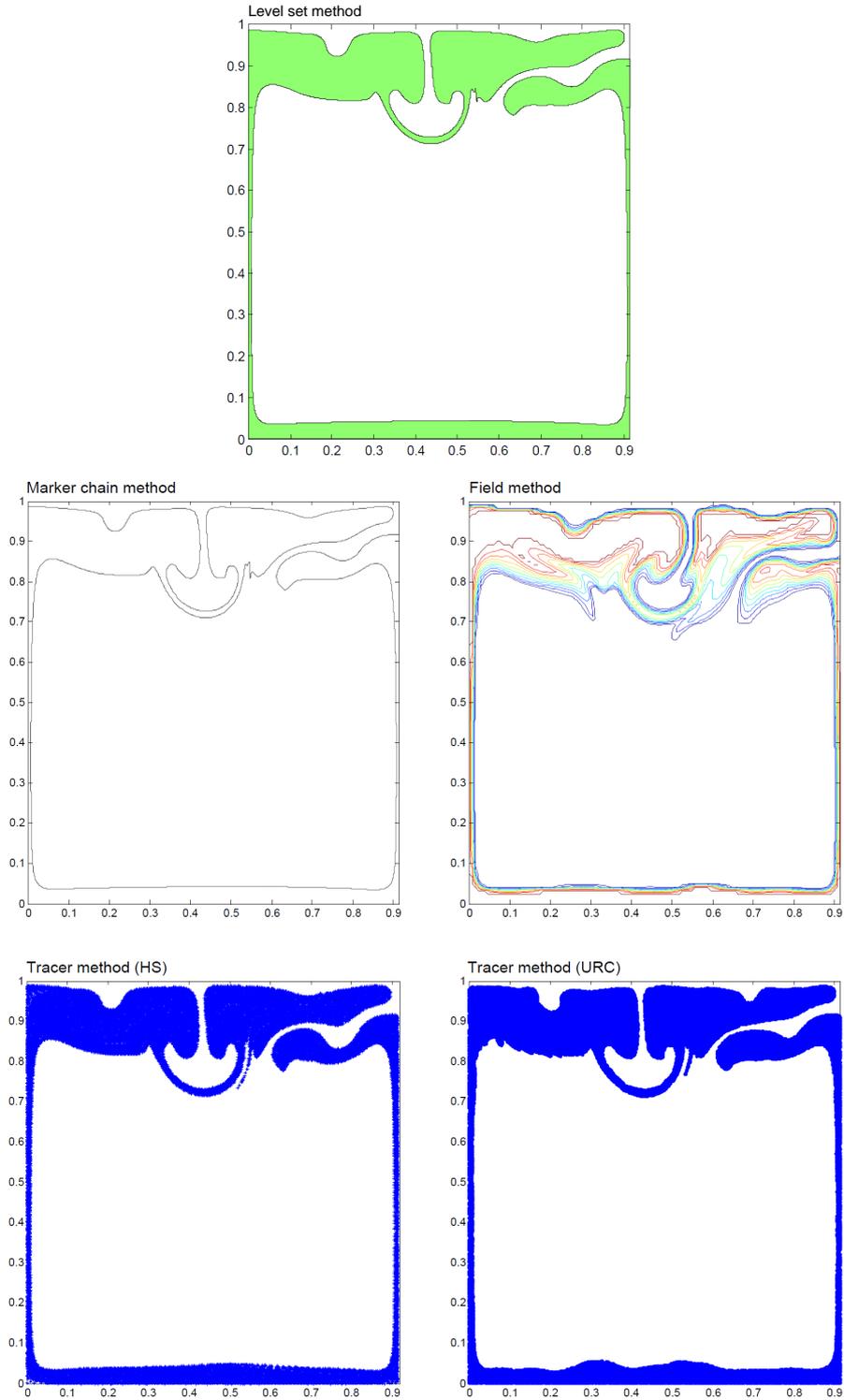


Figure 7. The Rayleigh-Taylor instability at $t = 1500$ computed by the level set method on a 300×330 grid compared to the best results of the four codes compared in *Van Keken et al.* [1997].

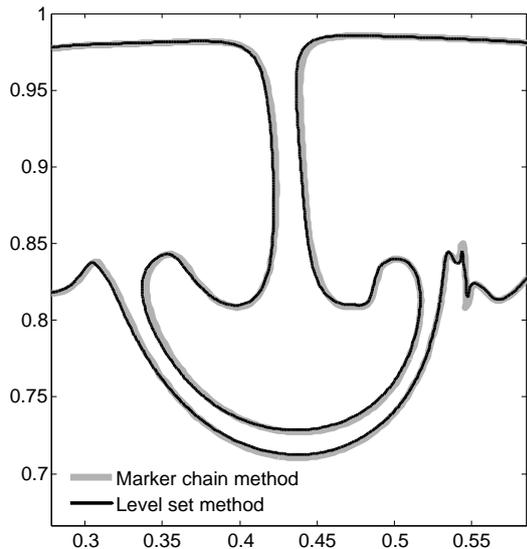


Figure 8. Detailed comparison of the level set (thin black line) and the marker chain approach (thick grey line) for the isothermal and isoviscous Rayleigh-Taylor instability at nondimensional time $t = 1500$. The plotted interfaces represent a zoom onto the instability descending from the top downwards in the middle of the box. The two methods yield an almost identical interface.

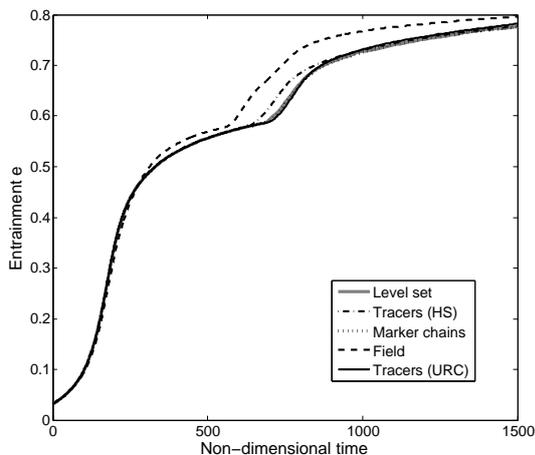


Figure 9. Evolution of the entrainment of the buoyant fluid over time as computed by the five different codes. The level set computation was done on a 160×176 grid.

combination with the extension-velocity-based advection of the level set function (sec. 6.2 for more details) and (3) the usage of Total-Variations-Diminishing (TVD) discretization schemes for the level set advection (eq. 22).

Overall, we conclude that our extended ghost-fluid-type method is a useful new tool for geodynamics, specifically for problems involving large viscosity contrasts and interfaces at which pressure and stresses may jump discontinuously. We note that the tangential component in the stress jump is not resolved equally well as

the normal component, but the error introduced by that inaccuracy is typically small to negligible, particularly in the context of buoyancy-driven flow.

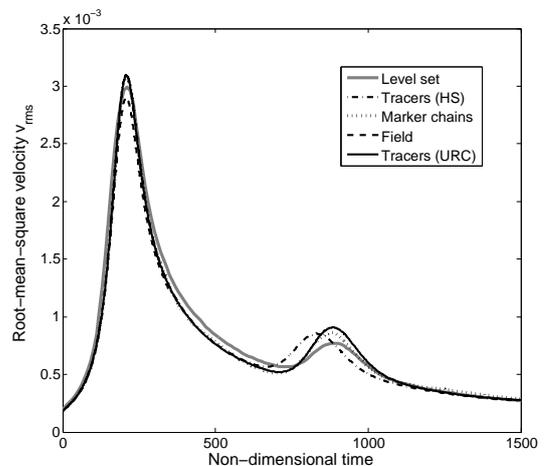


Figure 10. Evolution of the root mean square velocity of the interface over time as computed by the five different codes. The level set computation was done on a 160×176 grid.

6.2. Evaluation of the interface solver

The appeal of benchmark problem 3, the isothermal Rayleigh-Taylor instability by *Van Keken et al.* [1997] (sec. 4.3 and 5.3), is that it allows for a direct comparison of the various strategies for representing interfaces (sec. 3.2). Out of the four other approaches, we find the poorest agreement with the results obtained by the field approach (Fig. 7). The field method works by tracking a characteristic function Γ , usually referred to as the composition function, which is zero on one side of the interface and one on the other. The advective-transport equation is solved for the time evolution of this composition function. This is a straightforward method, in part because it is naturally adaptive – the front is defined on the same Eulerian mesh used in the computation – and also because it easily extends to three dimensions. However, the advective update of a discontinuous function can be problematic since considerable artificial smoothing can occur due to numerical diffusion in the composition function update (e.g. the supposedly sharp interface has spread out into a fictitiously continuous transition between the two fluid phases in Fig. 7). Furthermore, the advection of a discontinuous function is prone to dispersive errors, which result in spurious oscillations. Consequently, most implementations smear out the interface over several grid cells, thereby compromising the accuracy with which the fluid dynamics in the vicinity of the interface are resolved. Thus, one is often led to a higher grid resolution than desired in order to ameliorate the smoothing. Nonetheless, dispersive errors commonly prevail even for smeared interfaces

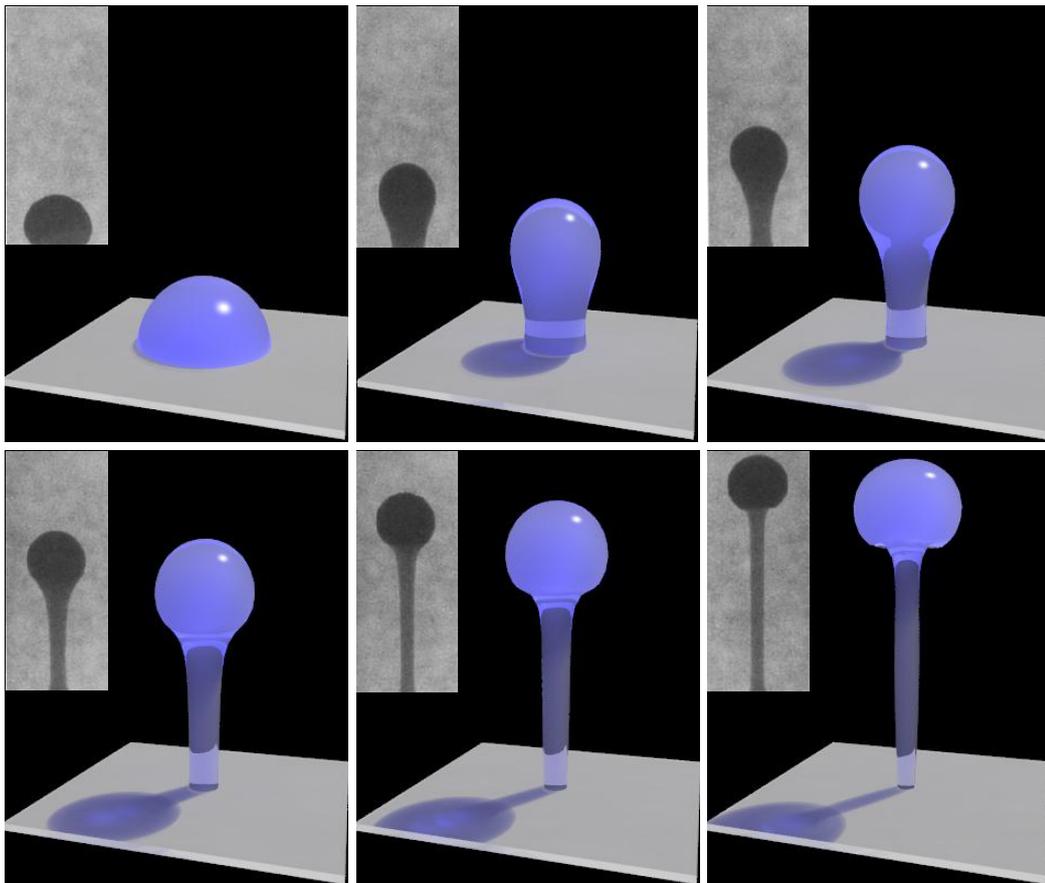


Figure 11. Three-dimensional benchmark computation for problem 3, a compositional plume rising from a free-slip surface. The grid resolution is $40 \times 40 \times 50$. The six snapshots of the dynamic evolution of the plume are shown for non-dimensional times 0, 8.4, 16.8, 25.2, 33.6, and 42. The experimental results by *Manga et al.* [1993] are included as black and white reproduction in the background .

(e.g. *Christensen* [1992], *Van Keken* [1993]). While remedies correcting the consequences of numerical diffusion (e.g. *Alley and Parmentier* [1998]) and dispersion exist (e.g. *Lenardic and Kaula* [1993]), we choose to focus on and construct numerical approaches which keep the interface sharp, thus avoiding both the smearing and artificial attempts to 're-sharpen' it. The level set method (sec. 3.2) avoids numerical diffusion of discontinuities, because it supersedes the advection of a discontinuous function. Instead, the advection equation (eq. 22) is solved for the higher-dimensional level set function, which will not result in numerical diffusion because the level set function has slope one by construction (eq. 21).

The evaluation of the differences between the interface as computed through level sets and that obtained

from the two particle or tracer methods (see Fig. 7) is subtle. Tracer approaches differentiate between two fluids by placing a large number of Lagrangian particles in one or both of the phases (e.g. *Moresi et al.* [2003]; *Gerya and Yuen* [2003, 2007]). While this Lagrangian representation of compositional differences is attractive

Table 2. Comparison of the quantitative parameters characterizing the dynamics of the isoviscous Rayleigh-Taylor instability.

Code	Grid	Growth Rate	$t(\max v)$	$\max v$
Level sets	120x132	0.01252	211.2	0.00301
Marker chains	80x80	0.01225	207.05	0.00309
Marker chains	80x80	0.01207	210.75	0.00305
Tracers HS	81x81	0.01118	208.99	0.00309
Tracers CND	48x48	0.01106	208.5	0.00309
Field SK	160x160	0.01179	207.84	0.00289

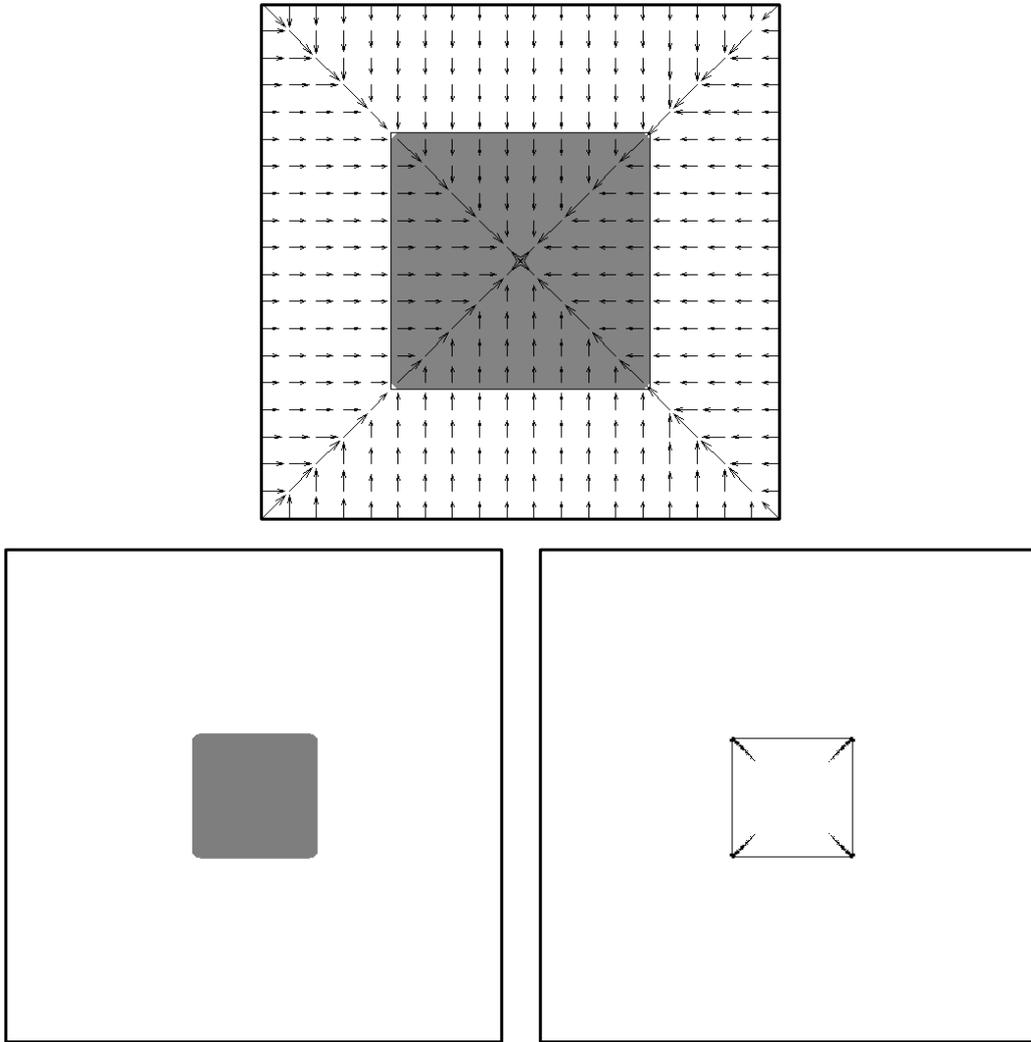


Figure 12. Three-dimensional benchmark computation for problem 3, a compositional plume rising from a free-slip surface. The grid resolution is $40 \times 40 \times 50$. The six snapshots of the dynamic evolution of the plume are shown for non-dimensional times 0, 8.4, 16.8, 25.2, 33.6, and 42. The experimental results by *Manga et al.* [1993] are included as black and white reproduction in the background.

for its avoidance of the numerical-diffusion problems of field approaches and its maintenance of mass conservation, the tracer method is prone to 'sampling errors': Although the fluid velocity is typically computed on an underlying mesh, evolving marker particles can create unreal and non-physical subgrid results, far below the actual accuracy of the computation, and these can create misleading and non-existent fingering, as well as thin, sub-cell structures. These structures result from the use of interpolation functions to move the sub-grid

markers. An additional difficulty is that there is no natural adaptivity to this approach. Particles must be added and subtracted as the interface stretches, shrinks, and contorts. Finally, three-dimensional implementations become computationally very expensive, because a large number of particles is required to obtain sufficient accuracy. For example, *Schmalzl and Loddoch* [2003] estimated that an increase of tracer particles on the order of 10^3 is necessary when adding another dimension to a previously two-dimensional problem. As a remedy to this problem, *Tackley and King* [2003] suggested using a so-called 'tracer ratio' method, but com-

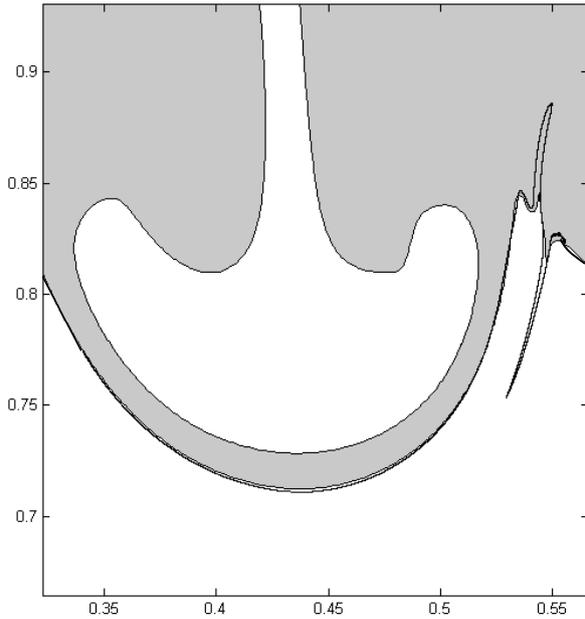


Figure 13. Zoom onto the isothermal and isoviscous Rayleigh-Taylor instability specified by *Van Keken et al. [1997]* at time $t = 1500$. In this computation, the interface is tracked simultaneously by a level set function (in grey/white) and by tracers (black line). We only plot the tracers for the lower segment of the interface to highlight the difference between the two interface-representation techniques. In the tracer-based computation, we observe the formation of a thin, elongated peak, reminiscent of the tentacles observed at the edges of the collapsing square in Fig. 12.

computational efficiency remains a concern even then [*Lin and van Keken, 2006*].

It is often regarded as one of the advantages of particle approaches that they can provide sub-grid resolution. After all, both implementations of the particle method presented in *Van Keken et al. [1997]* find an elongated fine-scale structure accompanying the instability sinking in the center of the computational domain. However, there is also ample fine-scale structure that differs for the two approaches (see Fig. 7). An example illustrating how tracer-based interface tracking can lead to the formation of spurious fine-scale structures at the interface is shown in Figure 12. A collapsing square is tracked simultaneously through a level set function (displayed in grey/white) and 10 000 tracers located on the initial interface (Fig. 12, top). While the level set function yields a correct approximate solution, the tracers have formed long tentacles at the edges of the square (Fig. 12, bottom). This well-known behavior of tracer approaches is related to the fact that tracers move with the local velocity while imbedded surfaces (e.g. fluid interfaces) move with the local fluid velocity in the normal direction. Despite the tracer method’s appeal to indicate possible fine-scale structures, subgrid resolution comes at the risk of not being able to distinguish a geophysical structure from a numerical artifact. Fig-

ure 13 highlights the challenges associated with tracking deforming interfaces accurately. It shows a computation in which we track the evolution of the isoviscous Rayleigh-Taylor instability [*Van Keken et al., 1997*] simultaneously via level sets and tracers. In order to highlight the part of the interface where the two solutions differ, we only plot the tracers on the lower segment of the interface. Although the underlying resolution at which the equations of motion are solved is identical by construction, the tracer-based interface shows a thin tentacle, while the same structure on the level-set-based interface remains much less elongated. For a more detailed discussion of subgrid structures, please refer to the online supplement.

Finally, marker-chain approaches track the interface by linking together a chain of particles. Contrary to the tracer method, these particles are only placed directly on the interface. This reduces computational cost, while keeping the advantage of the tracer method, namely its relative non-diffusivity. We note that the marker-chain solution to benchmark problem 3 is almost identical to our level-set-based results (Figs. 7 and 8) and that the thin tentacle accompanying the center instability is smaller than for both tracer approaches. The main challenges of the marker-chain method are the geometric intricacies related to its three-dimensional implementation, in which one is required to track an evolving two-dimensional mesh of linked fluid particles. While possible, adaptive algorithms for grid refinement and simplifications are required [*Schmalzl and Loddoch, 2003*].

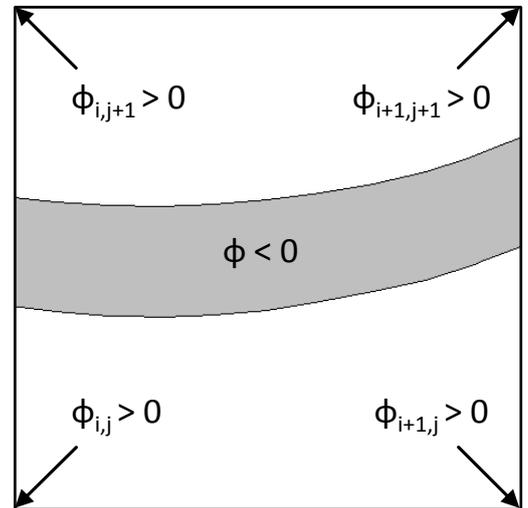


Figure 14. Illustration of the mass loss problem associated with interfaces that are entirely below grid resolution. Shown is a single computational cell spanned by the four grid points at the corners. One fluid (shaded in grey) is surrounded by the other fluid (white) such that the interface crosses the cell without interacting with the grid points. Because the level set function is positive at all grid points, the piece of grey fluid will be added to the white phase in the next computational step, leading to mass loss in the grey phase.

Overall, we conclude that the three main advantages of an interface solver based on the level set method are that: (1) It is naturally adaptive and able to handle strongly deforming interfaces and topological changes in the interface (i.e. rupturing or merging) without compromising accuracy or increasing computational cost. (2) It is computationally efficient and maintains this efficiency even for very complex interfaces and in three dimensions (Fig. 11). In the computations for benchmark problems 3 and 4, only about 5% of the computational time at each time step is spent on advecting the interface. Since less computational effort is required for the interface solver, more resources are available for the more accurate solution of the equations of motion. (3) Level sets propagate like surfaces with the local fluid velocity normal to the interface and avoid many problems commonly encountered in explicit approaches as illustrated in Figure 12.

6.3. Coupling of the solvers

The level set method avoids many problems commonly associated with tracking dynamic interfaces (sec. 6.2) as long as the normalization of the level set function (eq. 21) is maintained throughout the computation. In our implementation, the coupling between fluid and interface solver is facilitated by the fact that both solvers are based on the same grid. As a consequence, we will not resolve any subgrid features in our computations. In fact, if one of the fluids is thinned out to below grid resolution, mass loss will occur. This is illustrated in Figure 14. It shows a single computational cell spanned by four grid points located in the corners. In this constellation the level set function is positive $\phi > 0$ at all grid points. Thus, the piece of grey fluid shown will be added to the white fluid in the next computational step, resulting in mass loss in the grey phase. We stress that this mass loss is an indication of insufficient resolution in the computation. In fact, mass loss is closely related to the accumulation of numerical error and can thus be used to evaluate numerical error over the course of the simulation. We monitor mass balance throughout our computations. For an example of the evolution of the mass balance over time, please refer to the online supplement. We note that the level set method has been extended for problems in which the passive tracking of subgrid features might be desirable [Enright *et al.*, 2002, 2005], but since these are not relevant for buoyancy-driven problems, we will not discuss them further.

7. Conclusion

The combination of three complementary methods, the ghost-fluid, level set, and extension velocity methods yields a versatile and accurate method for simulating buoyancy-driven flow in the presence of large viscosity contrasts. Each of the three components targets one of the main challenges associated with the numer-

ical description of complex flows, namely (1) the solution of the multi-phase Stokes equation in the presence of discontinuities in the coefficients, solution, and source terms, (2) the accurate advection of a dynamically deforming interface such that no restrictions on interface geometry are imposed, and (3) the full and accurate coupling of fluid and interface solver while ensuring that discontinuities remain sharp over time and that mass is conserved in both phases. We validate our approach through four carefully selected benchmark problems in both two and three dimensions. We find excellent agreement in all four cases. Deviations from results obtained with different interface-representation techniques, as for the isothermal Rayleigh-Taylor instability by *Van Keken et al.* [1997], are not unexpected and explained by methodological differences.

Based on its methodology and performance for the selected benchmarks, we argue that our method is particularly interesting for the following types of geodynamical problems: (1) Problems that are characterized by sharp discontinuities in material parameters, pressure, stresses, or a source term. An example could be compositional plumes impinging on the lithosphere or a compositional discontinuity in the Earth's mantle. (2) Flow for which surface tension needs to be taken into account, such as magma bubbles (e.g. *Manga and Stone* [1993, 1995a]). (3) Problems with strongly and/or rapidly deforming interfaces such as multiple plume overturn in the Earth's mantle or lava-dome formation [Bourgoin *et al.*, 2007].

Acknowledgments. This work was supported by the Director, Office of Science, Computational and Technology Research, U.S. Department of Energy under Contract No. DE-AC02-05CH11231 and by Leigh Royden through the NSF-grant on Continental Dynamics EAR-0409373. Partial support for J.-C. Nave was provided through NSF grant DMS-0813648. We are indebted to James Sethian for generous advice on the implementation of extension velocities and to Benjamin Seibold for his interest and helpful comments. We thank Peter van Keken and Yolanda Deubelbeiss for sharing their numerical results from previous benchmark studies and Michael Manga for his consent to use photos from *Manga et al.* [1993] in Fig. 11. Finally, we would like to acknowledge Taras Gerya, Peter van Keken, Linda Elkins-Tanton, and James Sethian for their instructive reviews of this manuscript.

References

- Adalsteinsson, D., and J. A. Sethian (1995), A Fast Level Set Method for Propagating Interfaces, *J. Comput. Phys.*, 118(2), 269–277.
- Adalsteinsson, D., and J. A. Sethian (1999), The Fast Construction of Extension Velocities in Level Set Methods, *J. Comput. Phys.*, 148(1), 2–22.
- Alley, K. M., and E. M. Parmentier (1998), Numerical experiments on thermal convection in a chemically stratified viscous fluid heated from below: implications for a model of lunar evolution, *Phys. Earth Planet. Int.*, 108(1), 15–32.
- Bergantz, G. W., and J. Ni (1999), A numerical study of sedimentation by dripping instabilities in viscous fluids, *Int. J. Multiphase Flow*, 25(2), 307–320.
- Bourgoin, L., H. B. Mülhaus, A. J. Hale, and A. Arzac (2007), Studying the influence of a solid shell on lava dome growth and evolution using the level set method, *Geophys. J. Int.*, 170(3), 1431–1438.

- Chern, I. L., and Y. C. Shu (2007), A coupling interface method for elliptic interface problems, *J. Comput. Phys.*, *225*(2), 2138–2174.
- Chopp, D. L. (1993), Computing Minimal Surfaces via Level Set Curvature Flow, *J. Comput. Phys.*, *106*(1), 77–91.
- Chopp, D. L. (2001), Some Improvements of the Fast Marching Method, *SIAM J. Sci. Stat. Comput.*, *23*(1), 230–244.
- Chopp, D. L. (2009), Another look at velocity extensions in the level set method, *SIAM J. Sci. Comput.*, to appear, *31*(5), 3255–3273.
- Christensen, U. R. (1992), An Eulerian technique for thermomechanical modeling of lithospheric extension, *J. Geophys. Res.*, *97*(B2), 2015–2036.
- Conrad, C. P., and P. Molnar (1997), The growth of Rayleigh-Taylor-type instabilities in the lithosphere for various rheological and density structures, *Geophys. J. Int.*, *129*(1), 95–112.
- Deubelbeiss, Y., and B. J. P. Kaus (2008), Comparison of Eulerian and Lagrangian numerical techniques for the Stokes equations in the presence of strongly varying viscosity, *Phys. Earth Planet. Int.*, *171*(1–4), 92–111.
- Elkins-Tanton, L. T. (2007), Continental magmatism, volatile recycling, and a heterogeneous mantle caused by lithospheric gravitational instabilities, *J. Geophys. Res.*, *112*(B3), B03,405.
- Enright, D., R. Fedkiw, J. Ferziger, and I. Mitchell (2002), A Hybrid Particle Level Set Method for Improved Interface Capturing, *J. Comput. Phys.*, *183*(1), 83–116.
- Enright, D., F. Losasso, and R. Fedkiw (2005), A fast and accurate semi-Lagrangian particle level set method, *Comput. Struct.*, *83*(6–7), 479–490.
- Fedkiw, R. P., T. Aslam, B. Merriman, and S. Osher (1999), A Non-oscillatory Eulerian Approach to Interfaces in Multi-material Flows (the Ghost Fluid Method), *J. Comput. Phys.*, *152*(2), 457–492.
- Fleitout, L., and C. Froidevaux (1982), Tectonics and topography for a lithosphere containing density heterogeneities, *Tectonics*, *1*(1), 21–56.
- Gerya, T. V., and D. A. Yuen (2003), Characteristics-based marker-in-cell method with conservative finite-differences schemes for modeling geological flows with strongly variable transport properties, *Phys. Earth Planet. Int.*, *140*(4), 293–318.
- Gerya, T. V., and D. A. Yuen (2007), Robust characteristics method for modelling multiphase visco-elasto-plastic thermo-mechanical problems, *Phys. Earth Planet. Int.*, *163*(1–4), 83–105.
- Glimm, J., D. Marchesin, and O. McBryan (1981), A numerical method for two phase flow with an unstable interface, *J. Comput. Phys.*, *39*(1), 179–200.
- Gross, L., L. Bourgooin, A. J. Hale, and H. B. Mühlhaus (2007), Interface modeling in incompressible media using level sets in Escript, *Phys. Earth Planet. Int.*, *163*(1–4), 23–34.
- Hale, A. J. (2008), Lava dome growth and evolution with an independently deformable talus, *Geophys. J. Int.*, *174*(1), 391–417.
- Hale, A. J., L. Bourgooin, and H. B. Mühlhaus (2007), Using the Level-Set Method to Model Endogenous Lava Dome Growth, *J. Geophys. Res.*, *112*(B3), B03,213.
- Hoogenboom, T., and G. A. Houseman (2006), Rayleigh–Taylor instability as a mechanism for corona formation on Venus, *Icarus*, *180*(2), 292–307.
- Houseman, G. A., D. P. McKenzie, and P. Molnar (1981), Convective instability of a thickened boundary layer and its relevance for the thermal evolution of continental convergent belts, *J. Geophys. Res.*, *86*(NB7), 6115–6132.
- Ismail-Zadeh, A., I. Tsepelev, C. Talbot, and A. Korotkii (2004), Three-dimensional forward and backward modelling of diapirism: numerical approach and its applicability to the evolution of salt structures in the Pricaspian basin, *Tectonophysics*, *387*(1–4), 81–103.
- Jiang, G.-S., and D. Peng (2000), Weighted ENO Schemes for Hamilton Jacobi Equations, *SIAM J. Sci. Comput.*, *21*(6), 2126–2143.
- Jiang, G.-S., and C.-W. Shu (1996), Efficient Implementation of Weighted ENO Schemes, *J. Comput. Phys.*, *126*(1), 202–228.
- Kang, M., R. P. Fedkiw, and X. D. Liu (2000), A boundary condition capturing method for multiphase incompressible flow, *J. Sci. Comput.*, *15*(3), 323–360.
- Keck, R. (2007), Reinitialization for Level Set Methods, Master’s thesis, University of Kaiserslautern.
- Lenardic, A., and W. M. Kaula (1993), A Numerical Treatment of Geodynamic Viscous Flow Problems Involving the Advection of Material Interfaces, *J. Geophys. Res.*, *98*(B5), 8243–8260.
- Leveque, R. J., and Z. Li (1994), The immersed interface method for elliptic equations with discontinuous coefficients and singular sources, *SIAM J. Num. Ana.*, *31*(4), 1019–1044.
- Leveque, R. J., and Z. Li (1997), Immersed interface methods for Stokes flow with elastic boundaries or surface tension, *SIAM J. Sci. Comput.*, *18*(3), 709–735.
- Lin, S. C., and P. E. van Keken (2006), Dynamics of thermochemical plumes: 1. Plume formation and entrainment of a dense layer, *Geochem. Geophys. Geosyst.*, *7*(2), Q02,006.
- Liu, X. D., and T. C. Sideris (2003), Convergence of the ghost fluid method for elliptic equations with interfaces, *Math. Comput.*, *72*(244), 1731–1746.
- Liu, X.-D., S. Osher, and T. Chan (1994), Weighted Essentially Non-Oscillatory Schemes, *J. Comput. Phys.*, *115*(1), 200–212.
- Liu, X.-D., R. P. Fedkiw, and M. Kang (2000), A boundary condition capturing method for Poisson’s equation on irregular domains, *J. Comput. Phys.*, *160*(1), 151–178.
- Malladi, R., J. A. Sethian, and B. C. Vemuri (1995), Shape Modeling with Front Propagation: A Level Set Approach, in *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 17, pp. 158–175.
- Manga, M., and H. A. Stone (1993), Buoyancy-driven interactions between two deformable viscous drops, *J. Fluid Mech.*, *256*, 647–683.
- Manga, M., and H. A. Stone (1995a), Collective hydrodynamics of deformable drops and bubbles in dilute low Reynolds number suspensions, *J. Fluid Mech.*, *300*, 231–263.
- Manga, M., and H. A. Stone (1995b), Interactions between bubbles in magmas and lavas: effects of bubble deformation, *J. Volcanol. Geotherm. Res.*, *63*(3), 267–280.
- Manga, M., H. A. Stone, and R. J. O’Connell (1993), The interaction of plume heads with compositional discontinuities in the Earth’s mantle, *J. Geophys. Res.*, *98*(B11), 19,979–19,990.
- Marsh, B. D. (1988), Crystal capture, sorting, and retention in convecting magma, *Bull. Geol. Soc. Am.*, *100*(11), 1720–1737.
- Mayo, A. (1984), Fast solution of Poisson’s and the biharmonic equations on irregular regions., *SIAM J. Numer. Anal.*, *21*(2), 285–299.
- Mayo, A. (1985), Fast High Order Accurate Solution of Laplaces Equation on Irregular Regions, *SIAM J. Scient. Stat. Comput.*, *6*(1), 144–157.
- Moresi, L., F. Dufour, and H. B. Mühlhaus (2003), A Lagrangian integration point finite element method for large deformation modeling of viscoelastic geomaterials, *J. Comput. Phys.*, *184*(2), 476–497.
- Osher, S., and J. A. Sethian (1988), Fronts Propagating with Curvature-Dependent Speed: Algorithms Based on Hamilton-Jacobi Formulations, *J. Comput. Phys.*, *79*(1), 12–49.
- Römer, M. M., and H. J. Neugebauer (1991), The salt dome problem: a multilayered approach, *J. Geophys. Res.*, *96*(B2), 2389–2396.
- Rouy, E., and A. Tourin (1992), A viscosity solution approach to shape-from-shading, *SIAM J. Numer. Anal.*, *29*(3), 867–884.
- Royden, L. H., and L. Husson (2006), Trench motion, slab geometry and viscous stresses in subduction systems, *Geophys. J. Int.*, *167*(2), 881–905.
- Schaeffer, N., and M. Manga (2001), Interaction of rising and sinking mantle plumes, *Geophys. Res. Lett.*, *28*(3), 455–458.
- Schenk, O., and K. Gärtner (2004), Solving Unsymmetric Sparse Systems of Linear Equations with PARDISO, *J. Future Gen. Comp. Syst.*, *20*(3), 475–487.
- Schenk, O., and K. Gärtner (2006), On fast factorization pivoting methods for symmetric indefinite systems, *Elec. Trans. Numer. Anal.*, *23*, 158–179.
- Schmalzl, J., and A. Loddock (2003), Using subdivision surfaces and adaptive surface simplification algorithms for modeling chemical heterogeneities in geophysical flows, *Geochem. Geophys. Geosyst.*, *4*(9), 8303, doi:10.129/2003GC000,578.
- Schmeling, H. (1987), On the relation between initial conditions and late stages of Rayleigh-Taylor instabilities, *Tectonophysics*, *133*(1–2), 65–80.
- Schmeling, H., et al. (2008), A benchmark comparison of spontaneous subduction models Towards a free surface, *Phys. Earth Planet. Int.*, *171*(1–4), 198–223.

- Schmid, D., and Y. Podladchikov (2003), Analytical solutions for deformable elliptical inclusions in general shear, *Geophys. J. Int.*, *155*(1), 269–288.
- Schott, B., D. A. Yuen, and H. Schmeling (2000), The significance of shear heating in continental delamination, *Phys. Earth Planet. Int.*, *118*(3-4), 273–290.
- Sethian, J. A. (1996), A fast marching level set method for monotonically advancing fronts, *Proc. Nat. Acad. Sci.*, *93*(4), 1591–1595.
- Sethian, J. A. (1997), Level Set Methods: An Act of Violence, *American Scientist*, *85*(3), 1–23.
- Sethian, J. A. (1999a), *Level Set Methods and Fast Marching Methods*, 2nd ed., Cambridge University Press.
- Sethian, J. A. (1999b), Fast marching methods, *SIAM review*, *41*(2), 199–235.
- Shu, C. W., and S. Osher (1988), Efficient Implementation of Essentially Non-Oscillatory Shock Capturing Schemes, *J. Comput. Phys.*, *77*(2), 439–471.
- Sussman, M., and E. Fatemi (1999), An efficient, interface-preserving level set redistancing algorithm and its application to interfacial incompressible fluid flow, *SIAM J. Sci. Comput.*, *20*(4), 1165–1191.
- Sussman, M., P. Smereka, and S. Osher (1994), A level set approach for computing solutions to incompressible two-phase flow, *J. Comput. Phys.*, *114*(1), 146–159.
- Sussman, M., E. Fatemi, P. Smereka, and S. Osher (1998), An improved level set method for incompressible two-phase flows, *Com. Fluids*, *27*(5-6), 663–680.
- Tackley, P. J., and S. D. King (2003), Testing the tracer ratio method for modeling active compositional fields in mantle convection simulations, *Geochem. Geophys. Geosyst.*, *4*, 8302.
- Tornberg, A. K., and B. Engquist (2003), Regularization techniques for numerical approximation of PDEs with singularities, *J. Scient. Comput.*, *19*(1-3), 527–552.
- Van Keken, P. E. (1993), Numerical modeling of thermochemically driven fluid flow, applied to the Earth’s lithosphere and mantle. , Ph.D. thesis, Geologica Ultraiectina, 107, University of Utrecht, The Netherlands.
- Van Keken, P. E., S. D. King, H. Schmeling, U. R. Christensen, D. Neumeister, and M.-P. Doin (1997), A comparison of methods for the modeling of thermochemical convection, *J. Geophys. Res.*, *102*(B10), 22,477–22,495.
- Woidt, W. D. (1978), Finite element calculations applied to salt dome analysis, *Tectonophys.*, *50*(2-3), 369–386.

J. Suckale, Department of Earth, Atmospheric, and Planetary Sciences, Massachusetts Institute of Technology, 77 Massachusetts Ave, Cambridge, MA-02139, USA, suckale@mit.edu

J.-C. Nave, Department of Mathematics, Massachusetts Institute of Technology, 77 Massachusetts Ave, Cambridge, MA-02139, USA

B.H. Hager, Department of Earth, Atmospheric, and Planetary Sciences, Massachusetts Institute of Technology, 77 Massachusetts Ave, Cambridge, MA-02139, USA

Text S1: Online supplement

This online supplement provides: (1) An abbreviated derivation of the jump conditions (eqs. 12 and 13) following *Kang et al.* [2000]; (2) Examples of the level set function for the initial conditions of benchmark problems 1 and 3; (3) Additional results for benchmark problem 2 for viscosity contrasts of several orders of magnitude; (4) Additional results for the non-isoviscous Rayleigh-Taylor instability (sec. 5.3); (5) The convergence tests for benchmark problems 1, 2, and 3 including a detailed plot of mass fluctuations for the isoviscous Rayleigh-Taylor instability as specified by *Van Keken et al.* [1997]; (6) An example computation highlighting possible biases related to the usage of iterative reinitialization (e.g. *Sussman et al.* [1994]); (7) A more detailed discussion of what we mean by subgrid features in the context of benchmark problem 3; and (8) some background on the efficiency of the level set method and how to optimize it.

Abbreviated derivation of the jump conditions

In this section we summarize the derivation of the jump conditions for pressure (eq. 12) and stresses (eq. 13) closely following the discussion in *Kang et al.* [2000].

As discussed in section 2, the jump conditions of an interface moving with the local fluid velocity are given by

$$\left[\begin{pmatrix} \mathbf{n} \\ \mathbf{t}_1 \\ \mathbf{t}_2 \end{pmatrix} (pI - \tau)\mathbf{n}^T \right] = \begin{pmatrix} \sigma\kappa \\ 0 \\ 0 \end{pmatrix}, \quad (29)$$

where the square brackets denote the jump across the interface. Using the definition of the stress tensor

$$\tau = \mu \begin{pmatrix} \nabla u \\ \nabla v \\ \nabla w \end{pmatrix} + \mu \begin{pmatrix} \nabla u \\ \nabla v \\ \nabla w \end{pmatrix}^T \quad (30)$$

in eq. 29 yields

$$\left[\begin{pmatrix} p \\ 0 \\ 0 \end{pmatrix} - \mu \begin{pmatrix} \mathbf{n} \\ \mathbf{t}_1 \\ \mathbf{t}_2 \end{pmatrix} \begin{pmatrix} \nabla u \cdot \mathbf{n} \\ \nabla v \cdot \mathbf{n} \\ \nabla w \cdot \mathbf{n} \end{pmatrix} - \mu \begin{pmatrix} \nabla u \cdot \mathbf{n} & \nabla v \cdot \mathbf{n} & \nabla w \cdot \mathbf{n} \\ \nabla u \cdot \mathbf{t}_1 & \nabla v \cdot \mathbf{t}_1 & \nabla w \cdot \mathbf{t}_1 \\ \nabla u \cdot \mathbf{t}_2 & \nabla v \cdot \mathbf{t}_2 & \nabla w \cdot \mathbf{t}_2 \end{pmatrix} \cdot \mathbf{n} \right] = \begin{pmatrix} \sigma\kappa \\ 0 \\ 0 \end{pmatrix}. \quad (31)$$

Evidently, eq. 31 can also be written as three separate jump conditions:

$$[p - 2\mu(\nabla u \cdot \mathbf{n}, \nabla v \cdot \mathbf{n}, \nabla w \cdot \mathbf{n}) \cdot \mathbf{n}] = \sigma\kappa \quad (32)$$

$$[\mu(\nabla u \cdot \mathbf{n}, \nabla v \cdot \mathbf{n}, \nabla w \cdot \mathbf{n}) \cdot \mathbf{t}_1 + \mu(\nabla u \cdot \mathbf{t}_1, \nabla v \cdot \mathbf{t}_1, \nabla w \cdot \mathbf{t}_1) \cdot \mathbf{n}] = 0 \quad (33)$$

$$[\mu(\nabla u \cdot \mathbf{n}, \nabla v \cdot \mathbf{n}, \nabla w \cdot \mathbf{n}) \cdot \mathbf{t}_2 + \mu(\nabla u \cdot \mathbf{t}_2, \nabla v \cdot \mathbf{t}_2, \nabla w \cdot \mathbf{t}_2) \cdot \mathbf{n}] = 0. \quad (34)$$

In viscous flows, the velocities across fluid interfaces are continuous

$$[u] = [v] = [w] = 0. \quad (35)$$

Furthermore, fluid interfaces cannot support shear stresses implying that the tangential velocity derivatives are continuous as well

$$[\nabla u \cdot \mathbf{t}_1] = [\nabla v \cdot \mathbf{t}_1] = [\nabla w \cdot \mathbf{t}_1] = 0 \quad (36)$$

$$[\nabla u \cdot \mathbf{t}_2] = [\nabla v \cdot \mathbf{t}_2] = [\nabla w \cdot \mathbf{t}_2] = 0. \quad (37)$$

Rewriting the incompressibility condition $\nabla \cdot \mathbf{v} = 0$ as

$$(\nabla u \cdot \mathbf{n}, \nabla v \cdot \mathbf{n}, \nabla w \cdot \mathbf{n}) \cdot \mathbf{n} + (\nabla u \cdot \mathbf{t}_1, \nabla v \cdot \mathbf{t}_1, \nabla w \cdot \mathbf{t}_1) \cdot \mathbf{t}_1 \quad (38)$$

$$+ (\nabla u \cdot \mathbf{t}_2, \nabla v \cdot \mathbf{t}_2, \nabla w \cdot \mathbf{t}_2) \cdot \mathbf{t}_2 = \nabla \cdot \mathbf{v} = 0, \quad (39)$$

it follows from eqs. 36 and 37 that

$$[(\nabla u \cdot \mathbf{n}, \nabla v \cdot \mathbf{n}, \nabla w \cdot \mathbf{n}) \cdot \mathbf{n}] = 0. \quad (40)$$

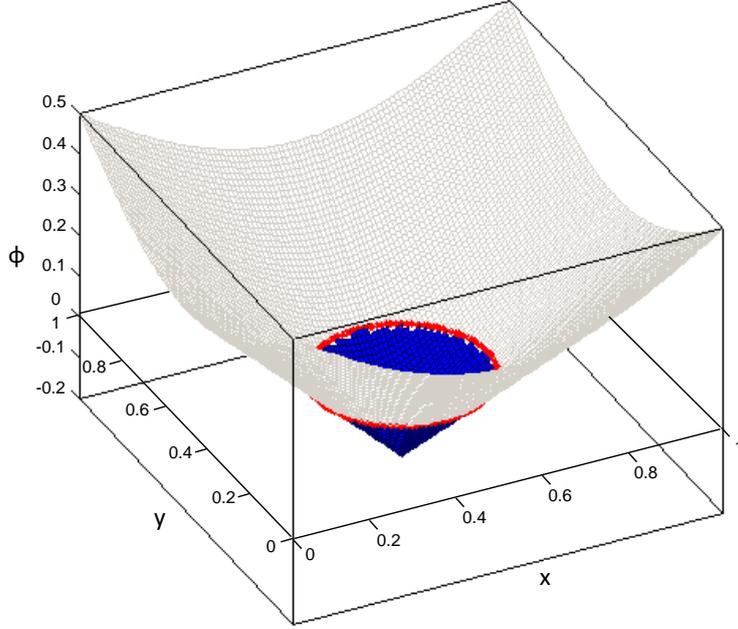


Figure S 1. Plot of the level set function representing a spherical drop, benchmark problem 1. The level set function ϕ is plotted on the vertical axis. The level set function intersects with the physical domain only at $\phi = 0$. The x- and y-axis of the physical domain are displayed at the $\phi = 0$ -level to highlight this fact. The interface between the viscous drop and the surrounding fluid coincides with the zero level set and is highlighted in red. The two fluids inside and outside the drop are indicated in blue and grey, respectively. Note that the blue phase corresponds to the domain for which $\phi < 0$ and the grey phase to the domain for which $\phi > 0$.

Thus, from combining eqs. 40 and 32 we obtain

$$[p] = \sigma\kappa + 2[\mu](\nabla u \cdot \mathbf{n}, \nabla v \cdot \mathbf{n}, \nabla w \cdot \mathbf{n}) \cdot \mathbf{n}, \quad (41)$$

which is the jump condition for pressure (eq. 12) discussed in section 3.1.2 in the paper. For the derivation of the jump in normal-stresses, eqs. 33, 33, 36, 37, and 40 can be compiled to obtain

$$\begin{aligned} \begin{pmatrix} \mathbf{n} \\ \mathbf{t}_1 \\ \mathbf{t}_2 \end{pmatrix} \begin{pmatrix} [\mu \nabla u] \\ [\mu \nabla v] \\ [\mu \nabla w] \end{pmatrix} \begin{pmatrix} \mathbf{n} \\ \mathbf{t}_1 \\ \mathbf{t}_2 \end{pmatrix}^T &= [\mu] \begin{pmatrix} \mathbf{n} \\ \mathbf{t}_1 \\ \mathbf{t}_2 \end{pmatrix} \begin{pmatrix} \nabla u \\ \nabla v \\ \nabla w \end{pmatrix} \begin{pmatrix} \mathbf{0} \\ \mathbf{t}_1 \\ \mathbf{t}_2 \end{pmatrix}^T + [\mu] \begin{pmatrix} \mathbf{n} \\ \mathbf{0} \\ \mathbf{0} \end{pmatrix} \begin{pmatrix} \nabla u \\ \nabla v \\ \nabla w \end{pmatrix} \begin{pmatrix} \mathbf{n} \\ \mathbf{0} \\ \mathbf{0} \end{pmatrix}^T \\ &\quad - [\mu] \begin{pmatrix} \mathbf{0} \\ \mathbf{t}_1 \\ \mathbf{t}_2 \end{pmatrix} \begin{pmatrix} \nabla u \\ \nabla v \\ \nabla w \end{pmatrix} \begin{pmatrix} \mathbf{n} \\ \mathbf{0} \\ \mathbf{0} \end{pmatrix}^T, \end{aligned} \quad (42)$$

which can be rewritten to yield the expression for the stress-jump used in the paper (eq. 13, sec. 3.1.2):

$$\begin{aligned} \begin{pmatrix} [\mu u_x] & [\mu u_y] & [\mu u_z] \\ [\mu v_x] & [\mu v_y] & [\mu v_z] \\ [\mu w_x] & [\mu w_x] & [\mu w_x] \end{pmatrix} &= [\mu] \begin{pmatrix} \nabla u \\ \nabla v \\ \nabla w \end{pmatrix} \begin{pmatrix} \mathbf{0} \\ \mathbf{t}_1 \\ \mathbf{t}_2 \end{pmatrix}^T \begin{pmatrix} \mathbf{0} \\ \mathbf{t}_1 \\ \mathbf{t}_2 \end{pmatrix} + [\mu] \mathbf{n}^T \mathbf{n} \begin{pmatrix} \nabla u \\ \nabla v \\ \nabla w \end{pmatrix} \mathbf{n}^T \mathbf{n} \\ &\quad - [\mu] \begin{pmatrix} \mathbf{0} \\ \mathbf{t}_1 \\ \mathbf{t}_2 \end{pmatrix}^T \begin{pmatrix} \mathbf{0} \\ \mathbf{t}_1 \\ \mathbf{t}_2 \end{pmatrix} \begin{pmatrix} \nabla u \\ \nabla v \\ \nabla w \end{pmatrix}^T \mathbf{n}^T \mathbf{n}. \end{aligned} \quad (43)$$

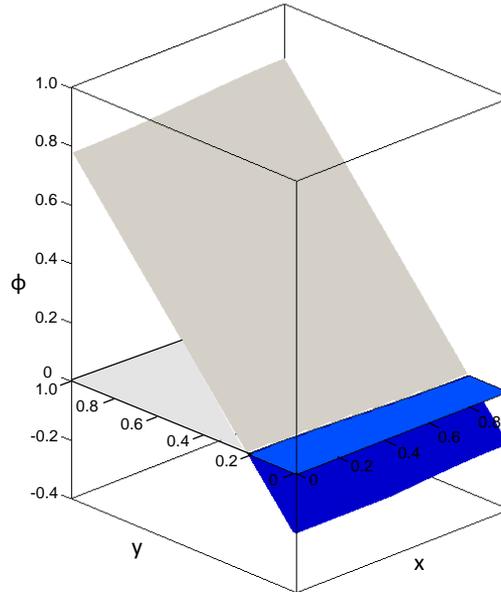


Figure S 2. Illustration of the connection between the physical domain and the level set function for the initial condition of the isothermal Rayleigh Taylor instability, benchmark problem 3 (sec. 4.3). The level set function ϕ is plotted on the vertical axis. It intersects with the physical domain only at $\phi = 0$. The x- and y-axis of the physical domain are displayed at the $\phi = 0$ -level to highlight this fact. Note that the interface in the physical domain coincides with the zero level set. The two fluid phases are shown in blue and grey, respectively. The blue phase is represented by the domain $\phi < 0$ and the grey phase by the domain $\phi > 0$.

Recasting the jump conditions in this way (eq. 43) has the advantage that the jumps in the stresses (left hand side of eq. 43) are reduced to the jump in the viscosity combined with various components of the velocity derivatives which are continuous and can be approximated through finite differences (right hand side of eq. 43).

Level-set functions for problems 1 and 3

Instead of tracking specific coordinates on the interface (typically stored in a vector, a 1D object), level set methods work by placing a grid on the interface and defining an auxiliary function everywhere on that grid (typically stored as a matrix, a two-dimensional object). This leads to the common notion that level set methods add an additional dimension to the problem. The auxiliary function, or level set function, is constructed such that it has negative values for grid points located in fluid phase 1, positive values for grid points located in fluid phase 2, and zero values directly at the interface, allowing for an easy identification of which phase a grid point is located in. In a dynamic setting, it turns out to be advantageous to construct the level set function as a signed distance function (sec. 3.2), meaning that for each grid point the shortest distance to the interface is assigned as the function value. The sign is determined based on the fluid phase the grid point is located in. Examples of the level set functions representing the initial conditions for both benchmark problems 1 and 3 are shown in Figures S1 and S2.

Pressure jump for weak inclusions at extremely large viscosity contrasts

When presenting our results for benchmark problem 2 (see sec. 5.2), the viscosity contrast is set to $\Pi_1 = 10^{-3}$ to allow for an easy comparison with previous numerical results by *Deubelbeiss and Kaus* [2008]. Our ghost-fluid-type method can also handle viscosity contrasts of numerous orders of magnitude as illustrated in Figure S3 for $\Pi_1 = 10^{-6}$ and $\Pi_1 = 10^{-10}$. Note that the percentage error in these computations is approximately the same despite the higher viscosity contrast. The reason is the discretization of the jump conditions (see eqs. 4 and 13), where the viscosity jump multiplies the viscous contribution to the jump which is not discontinuous and can thus be discretized without difficulties. In all of these computations, the grid resolution is set to 150×150 to highlight that the treatment of larger viscosity contrasts does not require higher grid resolutions.

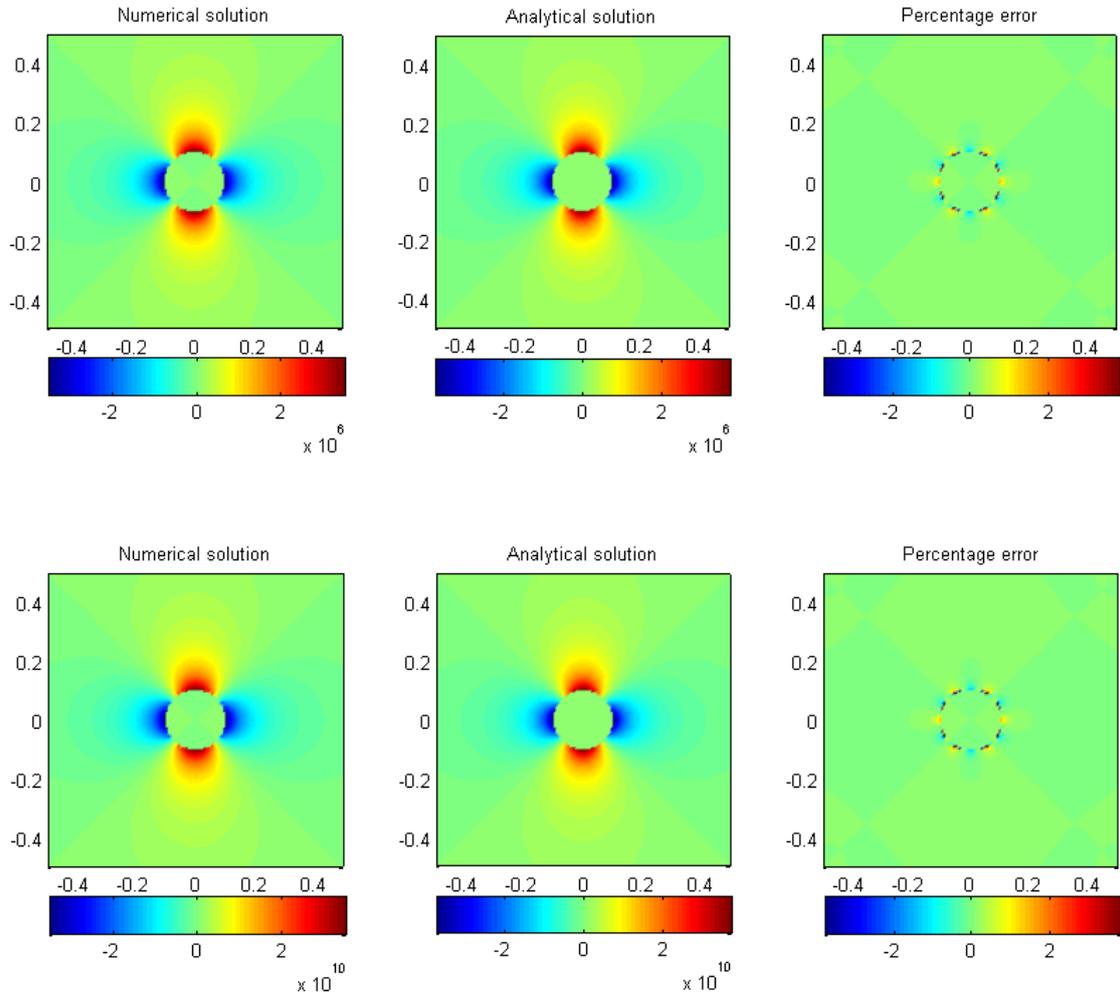


Figure S 3. Numerical (left) and analytical (middle) solution for the pressure field for benchmark problem 2. The viscosity contrast between inclusion and surrounding matrix is 6 (top row) and 10 (bottom row) orders of magnitude. The right panels shows the percentage error for both computations. The grid resolution is 150×150 in both cases.

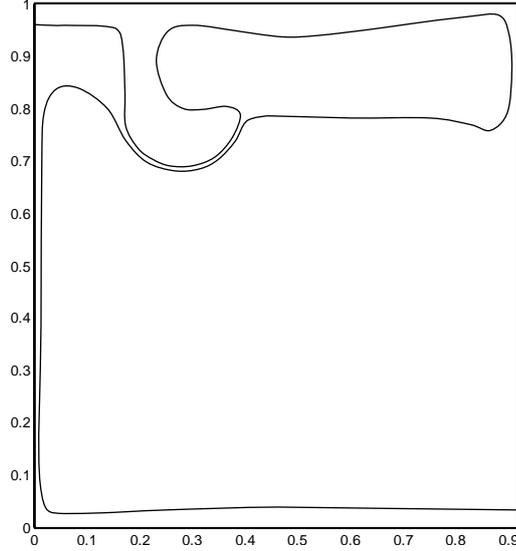


Figure S 4. The isothermal Rayleigh-Taylor instability with viscosity contrast $\Pi_2 = 10$ at non-dimensional time $t=500$. The computation was done with a grid resolution of 250×275 .

Non-isoviscous Rayleigh-Taylor instability

We compute the evolution of the Rayleigh-Taylor instability characterized by $\Pi_2 = 1$ and a viscosity ratio $\Pi_1 = 10$ until non-dimensional time step $t = 500$. Similar to the isoviscous case, we observe a very close correspondence between the level-set-based computation and the marker-chain simulations by *Van Keken et al.* [1997] (see Fig. S4). The initial growth rate of the instability is $\gamma \approx 0.04809$ compared to the marker-chain value of 0.04815. It is difficult to evaluate which one of these two values is more accurate, since the growth rate is known analytically only for infinitesimal initial perturbations of the interface. We note that the deviation between the two values is comparable to the deviation of growth rates observed in the isoviscous case (see Table 2).

Convergence tests for benchmark problems 1, 2, and 3

Benchmark problem 1 is targeted specifically at verifying whether the jump conditions at the interface are resolved accurately and at identifying the minimum grid resolution required for that purpose. Although only three points are necessary to build ghost-fluid-type stencils (sec. 3.1.1), this does not guarantee convergence. We

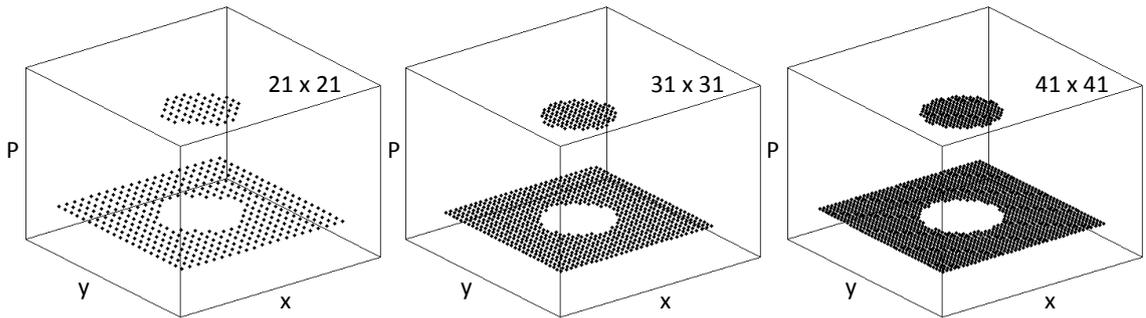


Figure S 5. Convergence test for benchmark problem 3. Shown are the numerical solutions for the pressure jump due to surface tension at the interface between a viscous drop and the surrounding fluid at grid resolutions 21×21 (left), 31×31 (center), and 41×41 (right). Each dot represents one grid point. Although the jump as such is resolved sharply without artificial smoothing at all of these resolutions, a minimum of ≈ 15 grid points in both the x- and y-direction are required inside of the drop to resolve its spherical shape.

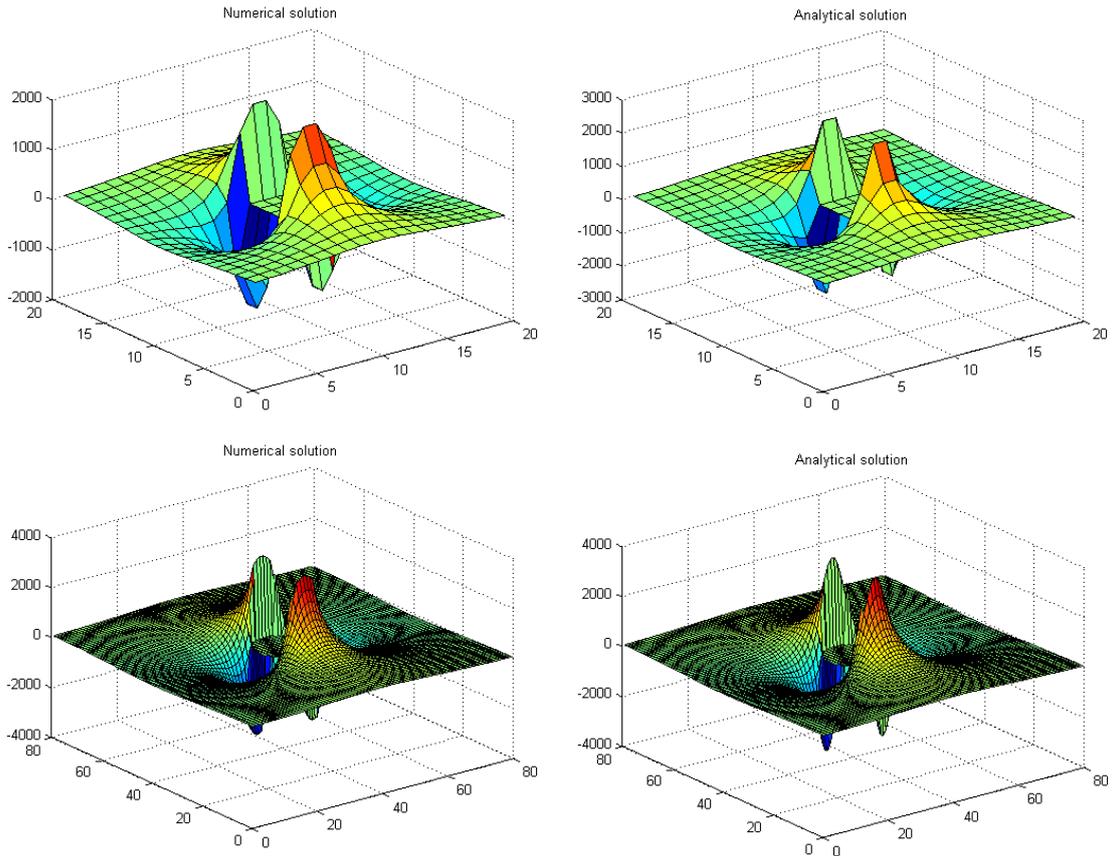


Figure S 6. Convergence test for benchmark problem 2. The viscosity contrast is $\Pi_1 = 10^{-3}$ as discussed in the paper, sec. 5.2. The shown grid resolutions are 20×20 (top) and 80×80 (bottom). In both cases, we contrast numerical (left) and analytical solution (right). The comparison shows that the additional challenge of benchmark problem 2 as compared to benchmark problem 1 lies in fully resolve the magnitude of the pressure jump, for which high resolution is required.

note that a convergence proof for ghost-fluid methods exists [Liu and Sideris, 2003], but for practical purposes it is often a concern whether the required resolution can be reached given available computational resources. Figure S5 shows that although the magnitude of the jump is approximated correctly at all grid sizes, resolving the circular shape of the bubble requires a minimum of ≈ 15 grid points in both x- and y-direction inside of the bubble. This is an encouraging result, since resolutions in this range are feasible even in three dimensions. Although the jump conditions in benchmark problem 2 are more challenging to resolve numerically than the surface-tension-related jump in pressure in benchmark problem 1, we observe that the pressure jump is captured sharply and without spurious oscillations even at very low resolution (see Fig. S6). The additional challenge as compared to benchmark problem 1, is to accurately resolve the magnitude of the jump which could be computed accurately even at minimal grid resolutions for benchmark problem 1. Figure S6 shows two example computations at a grid resolution of 20×20 and 80×80 . The percentage error in estimating the maximum pressure jump is 19 % and 8%, respectively.

The dynamics of benchmark problem 3 are not dominated by the jump conditions. Thus, it serves primarily as a verification for the computation of the rise speed of the instability and the advection of the interface. In addition to the convergence test for grid resolution (Fig. S7), we also performed a convergence test for the required temporal resolution (Fig. S8). We observe that the calculation is well converged for grid sizes exceeding 100×110 and incremental time steps of $\Delta t \leq 25 \Delta x$. In order to quantify the accumulation of numerical error – an important concern in dynamic problems – we plot the mass fluctuations in the buoyant phase in Figure S9. Although we observe notable fluctuations, particularly towards the end of the computation, these fluctuations remain $< 1\%$ error, indicating that the numerical solution is still reliable.

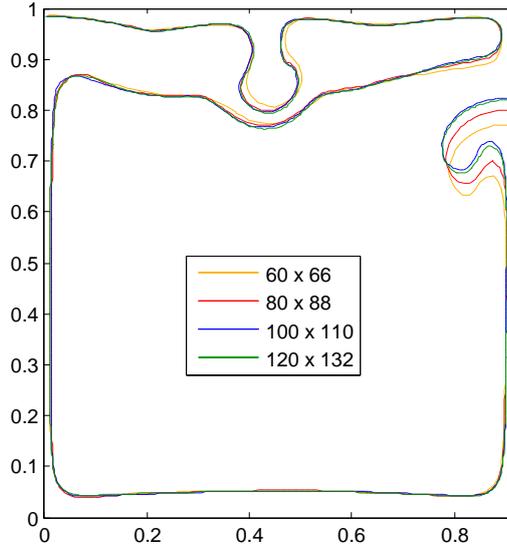


Figure S 7. Convergence test for the isothermal and isoviscous Rayleigh-Taylor instability, benchmark problem 3. A lack of convergence is easiest to identify during the phases of rapid rise of an instability. We illustrate this for the rise of the secondary instability on the right side of the box at time $t=1000$ and four different grid sizes: 60×66 , 80×88 , 100×110 , and 120×132 . We observe convergence for grid sizes above 100×110 .

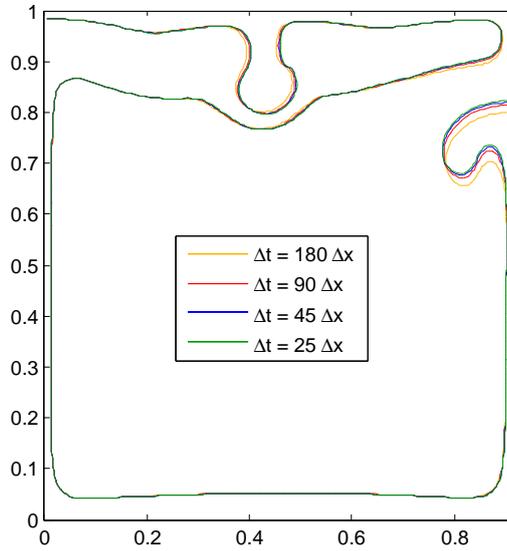


Figure S 8. Convergence test for the isothermal and isoviscous Rayleigh-Taylor instability, benchmark problem 3. Analogous to Fig. S7, we illustrate this convergence test for the rise of the secondary instability at time $t=1000$. The four interfaces were computed based on the time steps: $\Delta t = 180\Delta x$, $\Delta t = 90\Delta x$, $\Delta t = 45\Delta x$, and $\Delta t = 25\Delta x$. We observe convergence for time steps $\Delta t \leq 25\Delta x$.

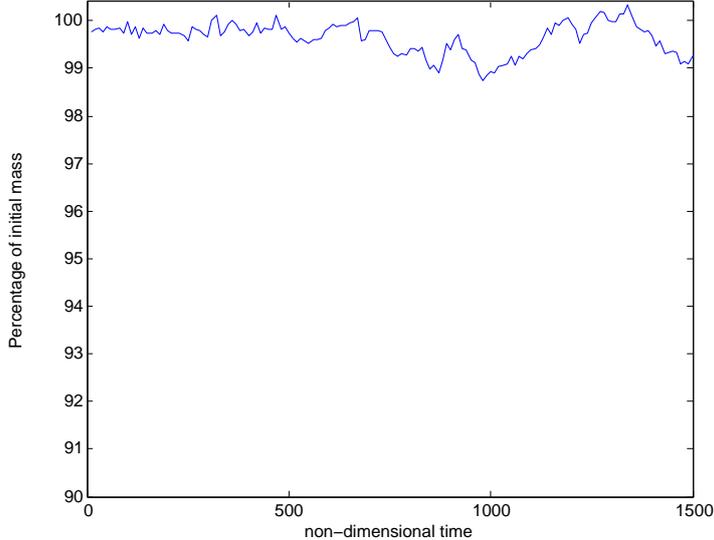


Figure S 9. Illustration of the accumulation of numerical error over time reflected in mass fluctuations. The plot shows the mass of the buoyant phase as a percentage of its initial mass. This plot compares to a similar plot presented by *Schmalzl and Loddoch*, [2003]. We note that fluctuations $< 1\%$ are not unexpected in complex fluid dynamical simulations. Overall, the mass conservation is satisfactory.

Possible biases related to iterative reinitialization techniques

Iterative reinitialization relies on a fundamentally different approach to coupling the fluid and the interface solver than the one adopted in our method (sec. 3.3). Instead of constructing an extension velocity function, the physical velocity field is used to advect the level set function (eq. 22). As a consequence, the normalization of the level set function needs to be restored continuously - a procedure known as re-initialization. The need to re-initialize was first noted by *Chopp* [1993]. The approach most commonly used to accomplish re-initialization was developed by *Sussman et al.* [1994] drawing on prior work by *Rouy and Tourin* [1992]. The idea is to iteratively find a solution to the so-called re-initialization equation

$$\frac{\partial \phi}{\partial \tau} + \text{sign}(\phi_0)(1 - |\nabla \phi|) = 0, \quad (44)$$

where τ is the pseudo-time and $\text{sign}(\phi_0)$ the signum function referring to $\phi_0 = \phi(\tau = 0)$. We refer to τ as a pseudo-time scale, because it does not correspond to an actual time scale in the physical problem.

Unfortunately, the iterative solution to the re-initialization is known to be prone to spurious mass loss and artificial front repositioning (e.g. *Adalsteinsson and Sethian* [1999], *Sethian* [1999a], *Sussman et al.* [1998], *Chopp* [2009], *Keck* [2007]). We implemented the iterative re-initialization procedure by *Sussman et al.* [1994] to test its performance for the isothermal Rayleigh-Taylor instability (benchmark problem 3) and indeed observed both substantial mass loss and significant variations in the final position of the interface. Figure S10 illustrates our findings. The instability on the left (Fig. S10 A) is computed based on the extension velocity approach adopted in this paper. The other two (Fig. S10 B and C) are both based on the iterative reinitialization by *Sussman et al.* [1994], but reinitialize more (C) or less (B) frequently and use different iterative parameters to solve equation 44. This comparison highlights that while it is certainly possible to obtain the 'correct' interface based on iterative reinitialization (Fig. S10 B), the accuracy of the solution depends sensitively on the details of the reinitialization procedure and the resulting biases can be substantial. Since iterative reinitialization is still commonly used (e.g. *Gross et al.* [2007]; *Bourgouin et al.* [2007]), it is important to be aware of these potential problems.

Subgrid features

In section 6.2, we discuss the role of subgrid features in the context of comparing the level set method to tracer approaches. We consider a certain feature of the interface to be subgrid, if the length scale associated with it is smaller than the resolution afforded by the fluid solver. An example is shown in Figure S11. Plotted is the numerical solution of the isothermal and isoviscous Rayleigh-Taylor instability at $t=1500$ computed by the

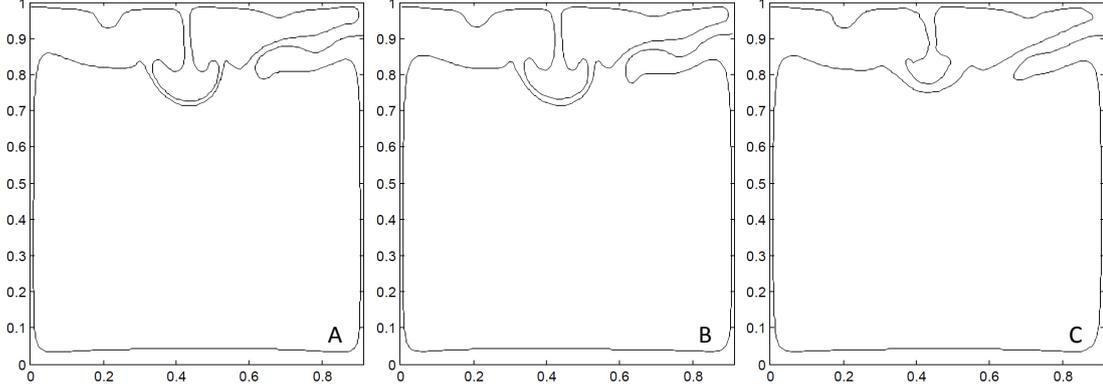


Figure S 10. Illustration of the potential bias introduced through an iterative reinitialization procedure. The left figure (A) shows the isothermal and isoviscous Rayleigh-Taylor instability at $t=1500$ computed with extension velocities. The other two figures (B) and (C) are based on an iterative reinitialization procedure, but different parameters are used in the iteration. For case (B) a single reinitialization iteration is performed at each computational time step Δt and $\Delta\tau = 0.9\Delta x$ is used in the numerical solution of equation 44. For case (C) 20 iterative reinitialization steps were taken at each physical time step Δt with $\Delta\tau = 0.9\Delta x/20$. All computations were performed on a 120×132 grid.

tracer-based finite-difference method by Schmeling. For the numerical details please refer to *Van Keken et al.* [1997]. The descending instability in the center is accompanied by a thin tentacle, which is observed to various degrees in all of the explicit interface representation techniques (i.e. both tracer computations and, to a much smaller degree, also for the marker chains). In the computation shown in Figure S11, the Stokes equation is solved on a structured finite-difference grid of size 81×81 . To visualize the resolution afforded by this choice of grid size, we overlay the thin tentacles located to the right of the descending instability on a grid of approximately that size. The length scale at which the Stokes equation is approximated numerically is given by the size of these grid cells. Evidently, the thickness of the tentacle falls well below the resolution limit of the fluid solver. In this case, it becomes a non-trivial challenge to accurately solve the equations of motion.

Efficiency considerations for the level set method

The most substantial gain in efficiency from using a level set representation of a dynamic interface instead of an explicit method (sec. 3.2) will occur in flow problems with strongly deforming interfaces such as benchmark problem 3. The level set method is based on an implicit view of the interface, which means that the computational effort required to represent it is essentially independent of its complexity. However, the details of the level set implementation also affect efficiency sensitively and special techniques have been developed to optimize the method, most notably the narrow-band method [*Adalsteinsson and Sethian, 1995*]. We also note that the computation of extension velocities based on the fast-marching method is computationally more efficient than the iterative reinitialization approach [*Adalsteinsson and Sethian, 1999*].

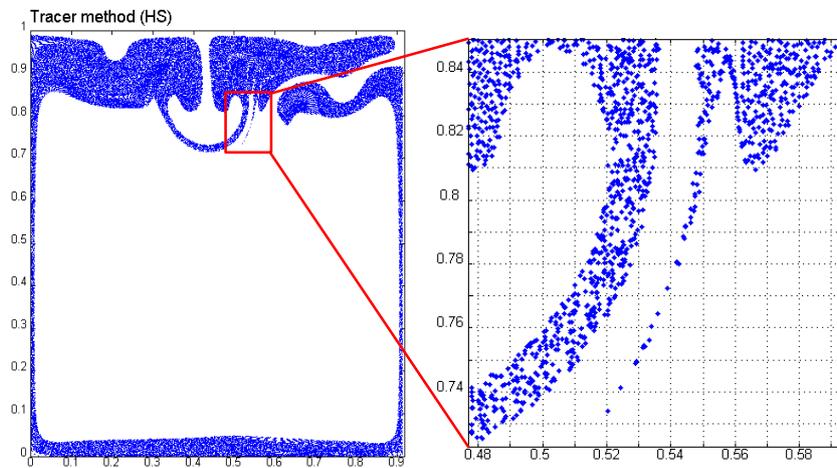


Figure S 11. The Rayleigh-Taylor instability as computed by the HS-tracer method at time $t=1500$. The equations of motion for this simulation were solved on an 81×81 grid. The right panel is a zoom onto the peak located left of the descending instability. Each blue dot represents one particle and the grid represents a rough estimate of the scale at which the flow field is approximated correctly.