Available online at www.sciencedirect.com

ScienceDirect

Journal of Computational Physics 227 (2008) 3940–3970

JOURNAL OF COMPUTATIONAL PHYSICS

www.elsevier.com/locate/jcp

Numerical prediction of interfacial instabilities: Sharp interface method (SIM)

R.R. Nourgaliev [a,*], M.-S. Liou [b], T.G. Theofanous [a]

[a] Center for Risk Studies and Safety, University of California, Santa Barbara, USA
[b] NASA Glenn Research Center, Aeropropulsion Division, Cleveland, USA

Received 12 February 2007; received in revised form 5 December 2007; accepted 6 December 2007
Available online 23 December 2007

Abstract

We introduce a sharp interface method (SIM) for the direct numerical simulation of unstable fluid–fluid interfaces. The method is based on the level set approach and the structured adaptive mesh refinement technology, endowed with a corridor of irregular, cut-cell grids that resolve the interfacial region to third-order spatial accuracy. Key in that regard are avoidance of numerical mixing, and a least-squares interpolation method that is supported by irregular datasets distinctly on each side of the interface. Results on test problems show our method to be free of the spurious current problem of the continuous surface force method and to converge, on grid refinement, at near-theoretical rates. Simulations of unstable Rayleigh–Taylor and viscous Kelvin–Helmholtz flows are found to converge at near-theoretical rates to the exact results over a wide range of conditions. Further, we show predictions of neutral-stability maps of the viscous Kelvin–Helmholtz flows (Yih instability), as well as self-selection of the most unstable wave-number in multimode simulations of Rayleigh–Taylor instability. All these results were obtained with a simple seeding of random infinitesimal disturbances of interface-shape, as opposed to seeding by a complete eigenmode. For other than elementary flows the latter would normally not be available, and extremely difficult to obtain if at all. Sample comparisons with our code adapted to mimic typical diffuse interface treatments were not satisfactory for shear-dominated flows. On the other hand the sharp dynamics of our method would appear to be compatible and possibly advantageous to any interfacial flow algorithm in which the interface is represented as a discrete Heaviside function.
Published by Elsevier Inc.

Keywords: Interfacial instability; Interfacial flows; Gas–liquid interfaces; Sharp-interface treatment; Viscous Kelvin–Helmholtz instability; Yih instability; Rayleigh–Taylor instability; Instability-seeding in numerical simulations

[*] Corresponding author. Present address: Idaho National Laboratory, Advanced Nuclear Energy Systems, Idaho Falls, USA. Tel.: +1 208 526 8495; fax: +1 208 526 2930.

E-mail addresses: robert.nourgaliev@inl.gov (R.R. Nourgaliev), meng-sing.liou@grc.nasa.gov (M.-S. Liou), theo@engr.ucsb.edu (T.G. Theofanous).

0021-9991/$ - see front matter Published by Elsevier Inc.
doi:10.1016/j.jcp.2007.12.008

1. Introduction

Free (fluid–fluid) interfaces in the presence of body forces and/or differential velocities are subject to deformation and breakup – processes that are principally responsible for flow regimes development, and thus for the macroscopic features in all multi-fluid systems. Depending on the direction of the acceleration vector, body forces normal to an interface can be stabilizing or destabilizing, differential velocities parallel to the interface are always destabilizing, and for development of instability such driving forces must be sufficient to overcome the force due to interfacial tension (always stabilizing). Under unstable conditions, early growth of an interfacial disturbance is exponential in time, and the theory for understanding this regime, based on the linearized Navier–Stokes equations, rests on firm grounds. At amplitudes that are a significant fraction of the wavelength, this theory breaks down, non-linear analysis becomes scarcely feasible, and numerical simulation is the key to further progress. In this paper, we address ourselves to a preliminary step in this quest – that of simulating interfacial instabilities at their inception and early growth. For this, linear regime, as noted above, there is a rigorous methodology and a body of work that make available a solid foundation for measuring success in a numerical endeavor that is, by its very nature, fraught with difficulty.

More specifically, we are concerned with numerical prediction of the classical *Rayleigh–Taylor* (R–T) and *Kelvin–Helmholtz* (K–H) instabilities at clean interfaces of viscous immiscible fluids [12]. They are the two canonical problems of the subject, as they capture the essential physics in all manifestations of interfacial instability in nature.[1] Correspondingly, we will refer to R–T and vK–H flows, thus emphasizing the fundamental significance of viscosity in the later case (arising due to viscosity discontinuity, this is known as the Yih instability [72]), as distinct from the original K–H instability (ideal fluids) [9]. Our immediate practical interest derives from the simultaneous action of these instabilities, at curved interfaces, along with transient mean flow development, as found in aero-breakup (Fig. 1 and [64,65]). The experiments only hint at the rich physics yet to be found in complex, non-linear interactions that generate, depending on fluid properties and flow conditions, a multitude of breakup regimes (morphologies), and the problem could well be regarded as the next canonical step towards understanding, a step for which direct numerical simulations can be seen to be clearly indispensable.

A key requirement in this practical perspective is that the numerical simulation be capable of self-selecting (for dominant growth) the naturally-preferred wavelength(s) as appropriate to any particular flow realization. Thus, it is necessary that the numerical simulation method is capable of recovering the dispersion relationships (growth rates as function of wave-length) found from linear stability analysis, and that this is done naturally as a part of the solution evolving from random, infinitesimally small interface-shape perturbations, as opposed to seeding of the complete flow field derived from eigensolutions. In the presence of viscosity, the generation of such eigensolutions for the vK–H problem involves a significant numerical effort [66,68]. Moreover in the presence of superposition of multiple types of instabilities, perhaps even with a developing base flow (as in many practical problems, including such as shown in Fig. 1) the eigensolutions would be extremely difficult to obtain by any means, if at all. Meeting this practical requirement is a special, and so far unique, aim of this paper.

Our method is focused on the singular nature of real, immiscible, fluid–fluid interfaces – boundaries of discontinuity in material properties, velocity derivatives and normal stresses. In particular, the solution is obtained by coupling the two flows on either side through a rigorous application of the exact boundary conditions at the interface (SIM, for *Sharp Interface Method*). While much more elaborate (and expensive) compared to the standard approach in the past 15 years, an approach that embeds the boundary in the solution of the two bulk flows through a narrow, constant-width interfacial region of variable (transitional) properties, we show that for shear-dominated flows the additional effort may be well deserved. On the other hand, we will also show that the diffuse interface method (DIM) is quite adequate for problems which are not dominated by shear, as is the case for R–T flows, and of course as in the numerous cases involving capillary-dominated flows (bubbles, drops at mild conditions of deformation and/or splitting/coalescence) that have been well documented in the literature.

[1] Not included in the above general statement are capillary instabilities and the effects of interfacial tension gradients (Marangoni instabilities) and of shock waves (Richtmyer–Meshkov instabilities).

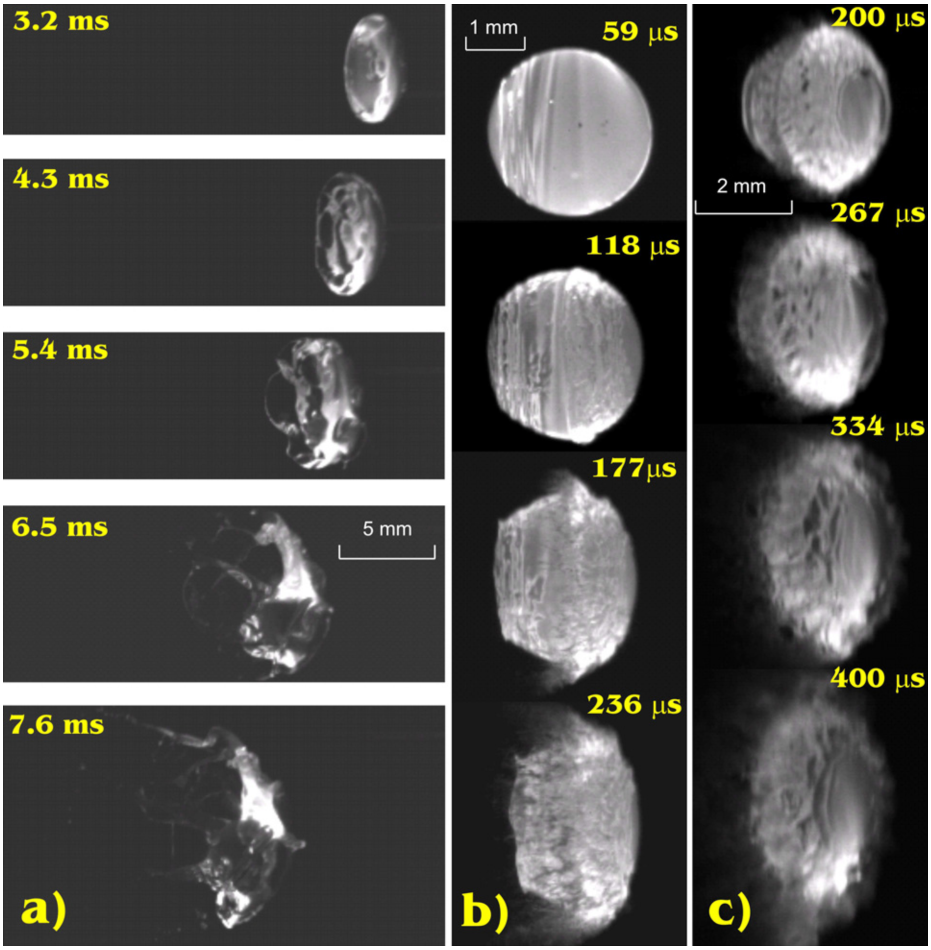

Fig. 1. Sequences of experimental LIF (laser-induced fluorescence) images during breakup of TBP (tributyl phosphate) drops in a high-speed gas flow [64]. (a) $M = 3$, $We = 75$, $Oh = 0.013$, $R = 500$. (b,c) $M = 0.87$, $We = 5400$, $Oh = 0.013$, $R = 15,000$. Images in (c) are taken by the camera 60° oblique to the gas flow. M is the flow Mach number; We, Oh, and R are the Weber, Ohnesorge, and Reynolds numbers, respectively. Initial drop size is 3.5 mm.

It is noteworthy that concerns about bypassing an explicitly-sharp treatment have been raised previously [27,39], but mildly and broadly, and they were not resolved. On the other hand, Shyy et al. [57,71], LeVeque et al. [34,35] and Sussman et al. [63] offered sharp treatments, but without addressing the particular fluid dynamic and interfacial instability context considered here.

The alternative noted above (based on the immersed boundary method [45,46]) is embodied in two classes of approaches, distinguished by the manner in which the transition in properties, most notably for our present purposes that of viscosity, is implemented. One, the *Diffuse Interface Method* (DIM), smears the discontinuity over an interfacial region of constant-width, made up by a few numerical cells (typically 2–8, on each side), according to a mollifying function chosen among a variety that have been used for this purpose [60,61,70]. The other, called the *Volume of Fluid method* (VOF), maintains the interface kinematically sharp, but it is dynamically diffuse in that the interfacial cell is occupied by a mixture of the two fluids, with one pressure, velocity and viscosity, the latter defined by some prescribed interpolation scheme from the pure-fluid viscosities weighted by the volume fractions. In both the DIM and VOF the interfacial tension is represented as a body force in the momentum equation [11,16,48,70]– it is applied within a mollified zone (in both DIM and VOF).[2] Using the VOF method, Coward et al. [13,36] (Couette flow) and Boeck et al. [10] (free shear layer) have shown good comparisons to linear stability theory with

[2] Recently there has been progress in imposing the interfacial tension sharply in level set [27,63] and VOF [16] based methods.

eigenfunction-seeded calculations for Reynolds numbers of 800 and 8000, respectively, and they highlighted the importance of dealing with the viscosity jump. No valid comparisons are available for the DIM [66].

In this paper, we introduce the SIM, and we present results of its implementation into our MuSiC (*Multi-scale Simulation Code*) code. This is a critical step in developing the simulation capability sought for problems such as shown in Fig. 1. Previously developed/presented key features of the code include (a) the characteristics-based matching for consistent interfacing between fluids (or fluids/solids) of high acoustic impedance mismatch at all flow speeds [40]; (b) an anchored, unlimited level set that provides high-fidelity performance in capturing interfaces [41], and (c) all of the above implemented in a structured adaptive mesh refinement (SAMR) environment for the high-resolutions needed in interfacial breakup [42]. The SIM is implemented in a front-tracking-like manner, using a band of unstructured-grid (cut-cells around the interface) incorporated into the SAMR mesh, and a treatment of fluxes at the interface that is based on one-sided calculation of derivatives which are also matched to satisfy the set of interface jump conditions, as appropriate for a boundary of discontinuity.[3] The method is third-order accurate both in the bulk fluids and near the interface, as appropriate for DNS of viscous flows. It will become apparent that the sharp dynamics of our method would appear to be compatible and possibly advantageous to any interfacial flow algorithm in which the interface is represented as a discrete Heaviside function.

In particular, we demonstrate second-order convergence in stresses, and thus with MuSiC–SIM we are able to recover complete neutral-stability maps of linear stability theory for viscous shear flows, as well as growth factors, all with the simple (primitive) seeding as discussed above.

As noted already, the evaluation basis of simulations presented here is linear stability analysis carried out with our *All-Regime Orr-Sommerfeld* (AROS) code. It involves domain-decomposition and Chebyshev collocation methods applied to the linearized Navier–Stokes equations. We use high-order Chebyshev polynomials and quadruple precision to permit convergent eigensystem analysis (even for large density/viscosity ratios) with the QZ algorithm. The code has been extensively verified with problems of various types as summarized in [59,66–68]. These works also address: (a) what is the relation/distinction between true interfacial shear instabilities of immiscible–fluid interfaces (*Yih-type*), and *interface-like* instabilities found in physically-diffuse interfaces under certain conditions; (b) whether an immiscible–fluid behavior can be recovered in the limit of a miscible layer thickness being let go to zero; and (c) the effect of mass diffusion (the Schmidt number) in the above questions.

A technical overview of our method in provided in Section 2. In Section 3 we provide the gridding strategy for the adaptive, sharp treatment. Then the heart of our numerical approach (SIM), that is the calculation of fluxes at interfacial cell edges, is given in Section 4. The performance in basic tests can be found in Section 5, and several applications to simulation of instabilities, both in R–T and vK–H flows, are in Section 6. Here we include comparison exercises with DIM. Conclusions are summarized in Section 7.

2. Basic considerations

As noted above, our *Sharp Interface Method* (SIM) is focused on satisfying the ***jump conditions*** at the interface exactly (within one cell)

$$\begin{aligned}
&\underline{Stresses}:\\
&\quad \mathbf{JC1}:\ [\mathcal{T}_{\xi\xi}] = [P] + \sigma\kappa\\
&\quad \mathbf{JC2}:\ [\mathcal{T}_{\xi\eta}] = 0\\
&\underline{Velocitycontinuity}:\\
&\quad \mathbf{JC3}:\ [u] = 0\\
&\quad \mathbf{JC4}:\ [v] = 0\\
&\underline{Propertyjump}:\\
&\quad \mathbf{JC5}:\ [\rho] = \text{given} \neq 0\\
&\quad \mathbf{JC6}:\ [\mu] = \text{given} \neq 0
\end{aligned} \tag{1}$$

[3] Two-sided treatment of interface in SIM is to be contrasted to the one-velocity/pressure treatment found in interfacial Cartesian cells of embedding-based approaches (DIM, VOF).

where $\mathcal{T}_{\xi\xi}$ and $\mathcal{T}_{\xi\eta}$ are the normal and shear components of the viscous stress tensor, respectively; $\vec{\xi}$ is the unit normal-to-the-interface vector; $\vec{\eta}$ is the unit tangential-to-the-interface vector; P is fluid pressure; $\mathbf{u} = (u, v)$ is the velocity vector; (μ, ρ, σ) are the fluid (viscosity, density, surface tension); κ is the interface curvature; and the brackets denote the jump of the bracketed quantity across the interface, i.e. $[\Psi] = \Psi^+ - \Psi^-$.

Note in particular that because of JC2 and JC6, the tangential velocity derivatives are discontinuous at the interface, and that because of JC1, the normal velocity derivatives on a curved interface become discontinuous too. As one could deduce from Yih's theory [72], the former discontinuity plays the crucial role in interfacial instability of shear flows, and it is therefore the *sine qua non* in the numerical formulation addressed in this paper. This is what sets apart this work from that by Kang et al. [27], as they assumed continuity in both these derivatives. On the other hand, it is the treatment of this same discontinuity that sets our method apart from the DIM, as the whole basis of it is to smooth (or mollify) the discontinuity. In Section 6, we show that this can prove problematic to fidelity for unstable viscous Kelvin–Helmholtz flows.

Respect for the jump conditions in numerical terms requires convergence in shear stresses, which translates to a requirement for a minimum of second-order accuracy in primitive variables – in fact, due to other potential losses we aim for third-order accuracy. The essential idea in achieving this goal is avoidance of numerical mixing, and this in turn is obtained by a combination of the following key features of our method:

(a) A sharp, piecewise-linear definition of the interface, Section 3.
(b) A third-order-accurate mapping between cell-average and (centroid) point values of all interfacial cells, Section 4.1.
(c) A flux treatment on the interface-defining edges of the interfacial cells that incorporates the jump conditions at all interfacial vertices to the third-order accuracy, Section 4.2.
(d) A third-order accurate least-squares interpolation procedure that is based on irregular data-sets, distinctly on each side of the interface (Appendix B).

All this is implemented within the Structured Adaptive Mesh Refinement (SAMR) framework. A corridor of irregular cut-cells (C_2-grid) is imbedded on the Cartesian (C_1-grid) of the finest SAMR level (Fig. 2). The

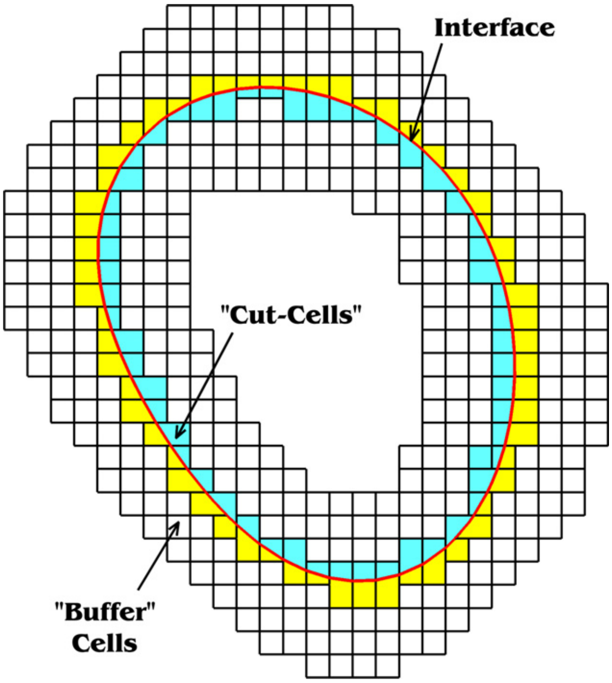

Fig. 2. Illustration of the C_2-grid corridor on top of the C_1-grid of the finest SAMR level.

flow field near the interface is obtained from finite-volume updates on C_2-grid, respecting interfacial jump conditions enforced to the third-order spatial accuracy at interfacial vertices. Solution in bulk fluids is obtained by a third-order-accurate conservative finite difference method on C_1-grid. These two solutions are properly synchronized in the frame of the third-order-accurate Runge–Kutta time updates.

This approach sets us apart from the Ghost Fluid Method whose underlying philosophy is based on extrapolation to populate the Ghost cells. In the original approach [15,27] this was done on a first-order basis with attendant difficulties. More recently [17], high-order techniques were demonstrated and accurate solutions to the Stefan problem were obtained. Further developments along these lines towards addressing the kinds of problems considered here would appear feasible.

The notion of enforcing jump conditions at an interface has also been suggested by LeVeque et al. [34,35] – they have developed a second-order accurate, discrete approximation of elliptic equations within a uniform Cartesian-grid. Shyy et al. [57,71] used cut-cells to handle the interface on a Cartesian-grid and demonstrated second-order accuracy (both locally and globally) for solving the incompressible Navier–Stokes equations. Most recently, Sussman et al. [63] have also shown the importance of honoring jump conditions – they worked on incompressible flow with a combination of the level set, volume tracking, and projection methods. The present paper differs from existing approaches, including the ones cited above, not only in methodology, but also in demonstrating the value of sharp treatment for the class of physical problems involving unstable, sheared interfaces.

3. Adaptive, Cartesian, Cut-Cell Mesh Refinement (AC^{3}MR)

3.1. SAMR, C_1-grid

As the means to more effectively resolve the interfacial region, we employ parallel, structured AMR. Developed in late 1980s by Berger and co-authors [5–7] for gas dynamics, it has been recently combined with the level set method [41] and characteristics-based matching (CBM) to simulate high-speed, multi-fluid flows [40]. SAMR is a multi-level, dynamic hierarchy of nested, structured-grid patches (Fig. 3) [3]. Our SAMR-code is implemented within the SAMRAI framework [26,51,69] and combines the following elements:

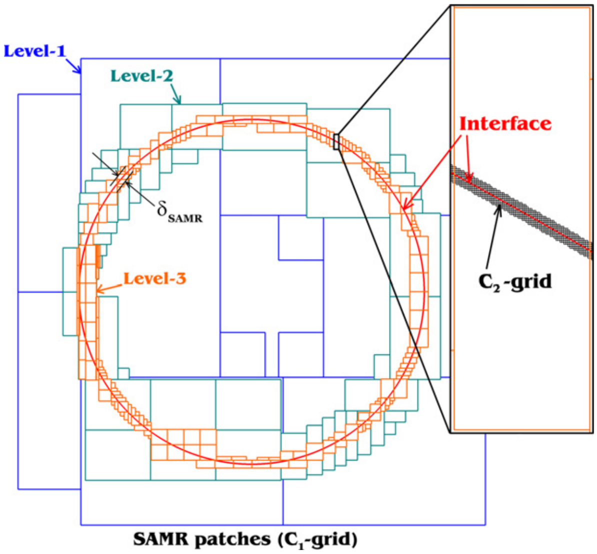

Fig. 3. Illustration of the C_1 and C_2-grids and the δ_{SAMR}-corridor, for a 4-level SAMR with a refinement ratio of 4 (SAMR$_{\mathrm{L4\text{-}R4}}$). SAMR patches on different levels are shown in different color. As discussed in [40], SAMR grid $\Omega^{\{h_0,h_1,\ldots,h_k\}|_{k=0,\ldots,L-1}}$ is a nested hierarchy of L grid levels $\Omega^{h_0} \supset \Omega^{h_1} \supset \cdots \supset \Omega^{h_{L-1}}$, where the coarsest grid Ω^{h_0} covers the entire computational domain. Each level Ω^{h_k} consists of a union of M_k logically rectangular regions, or patches, at the same grid resolution h_k and of generally different size. Note that the levels are nested, but the patches on different levels are not, and the patches on the same level may overlap. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

(a) *Algorithms to dynamically create patches of various (grid) refinements and to communicate solutions among them*. This includes specification of tagging criteria (refinement of δ_{SAMR}-corridor around the zero-contour of level set [41]); generation/disposal of SAMR patches according to the Berger–Rigoutsos algorithm [7] and inter-level prolongation/restriction operators needed to communicate/interpolate solutions between patches (the third-order-accurate, least-squares algorithm developed for this purpose is described in Appendix B).
(b) *Patch-update solver*. We use a conservative, finite difference, pseudo-compressibility method [14], based on a third-order-accurate, piecewise-parabolic spatial dicretization [38,39].
(c) *Time-integrator with dynamic time-stepping*. This is for observing multiple CFL limits (level-dependent), and for synchronizing the solutions among different patches. Details of the third-order-accurate, explicit Runge–Kutta, hyperbolic SAMR time-integrator can be found in [40].

3.2. Cut-Cell, unstructured, C_2-grid

The interface position is obtained by solving the level set (φ) equation with the unsteady velocity field [41]. Hence, a new C_2-grid must be regenerated at the beginning of each time step. The procedure involves the following steps:

(1) Creating markers defined by the intersections of the level set's zero-contour with the gridlines, Fig. 4(a). The piecewise-cubic positioning algorithm is given in Appendix A.
(2) Connecting these markers piecewise-linearly, to create a set of polygons that surround the interface, Figs. 2 and 4(b). Depending on their position relative to the interface, these polygons are called *positive* and *negative* cut-cells. Those signed-cut-cells that are sufficiently small (with areas $A_{\text{small}} < 0.4h^2$), are joined with same-sign, immediate neighbors – the ones with the maximum cell-cell contact interface, Fig. 4(c).
(3) Adding a *buffer* layer of regular cells so as to create a narrow-band of cells around the cut-cells, Figs. 2 and 4(d). The width of this band (measured normal to the interface) is selected $\sim (2\ st\ h)$, where st is the spatial-discretization stencil used on the C_1-grid. Thus, the C_2-grid is generated so as to adapt to the C_1-grid smoothly while providing an effective *barrier* between the two bulk-fluid solutions.

The flow solution is updated on the C_2-grid (Section 4), and restricted to the underlying C_1-grid, using a least-squares interpolation on an appropriate irregular dataset (Section 4.2). The level set function is then updated to provide the new configuration of the interface at the next time level $t^{(n+1)}$. Before their disposal, the data structure and solution on the $C_2^{(n)}$-grid are used to initialize the next-time-level $C_2^{(n+1)}$-grid (Section 4.2).

From the mesh generation point of view [4,20,52], our meshing strategy can be viewed as a combination of h- and p-refinements. Some polygons generated by the above algorithm might be skew and/or concave, but this is mitigated to a significant degree by our least-squares interpolation algorithm. The robustness and accuracy of meshing could be further improved by r-refinement (e.g. using the *Laplace–Beltrami Target Metric (LBTM) adaptation method* [21]), at the cost, of course, of added complication in programming as well as expense.

4. The Sharp Interface Method (SIM)

On the C_1-grid, the variables are defined at the cell centers and updated by a rather standard conservative finite difference formulation, as used in our previous work [38,39].

On the C_2-grid, the finite-volume update of the mth cut-cell is obtained by time integration of cell-averaged, conservative variables, $\overline{\mathbf{U}}_{(m)}$, using

$$\frac{\partial}{\partial t}\int_{\Omega_{(m)}(t)} \mathbf{U}_m(\mathbf{x},t)\mathrm{d}\Omega_{(m)} + \int_{\Gamma_{(m)}(t)} \mathbf{F}\mathrm{d}\Gamma_{(m)} = \int_{\Omega_{(m)}(t)} \mathbf{S}\mathrm{d}\Omega_{(m)} \tag{2}$$

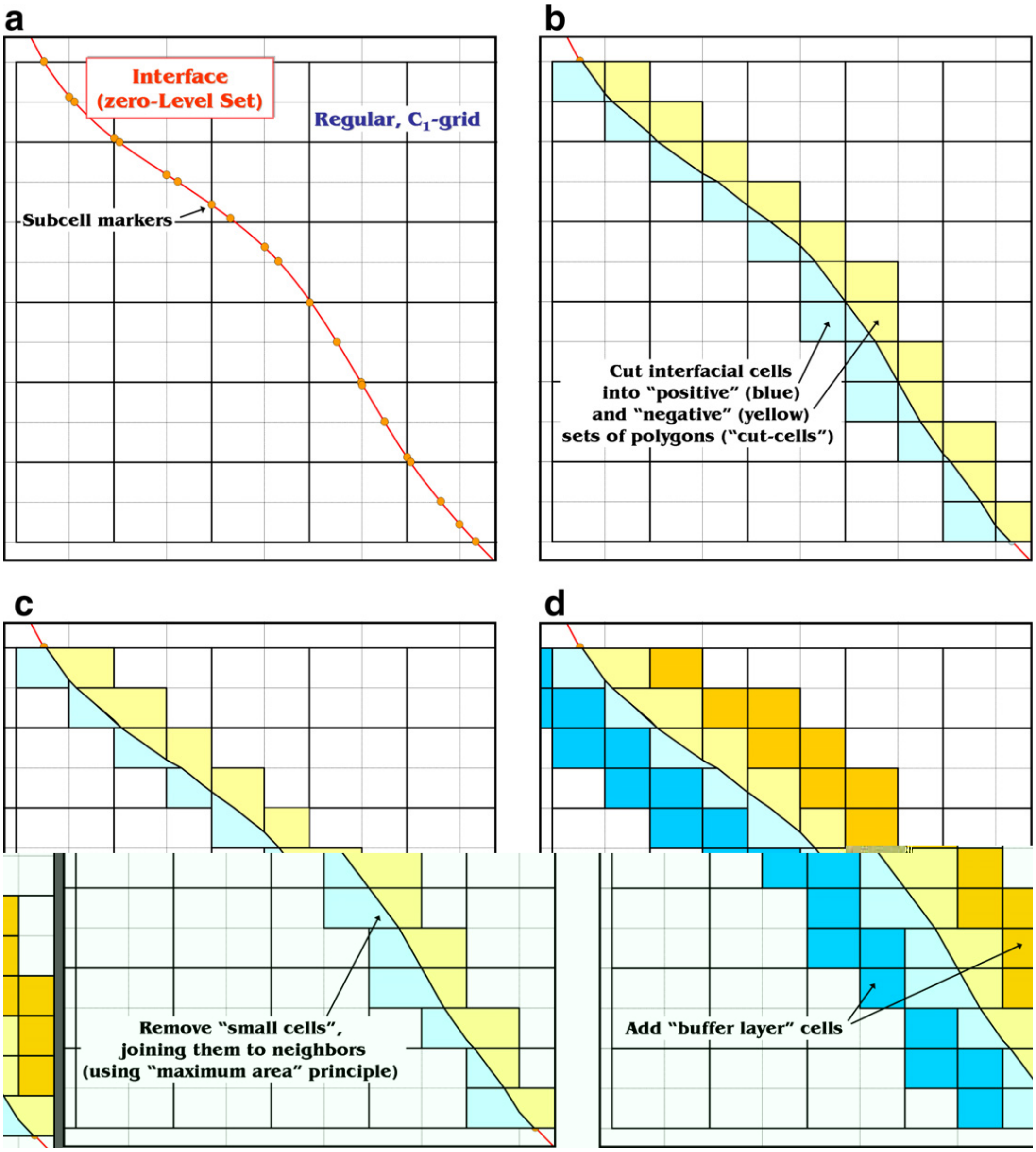

Fig. 4. Illustration of the C_2-grid generation procedure.

where $\Omega_{(m)}$ and $\Gamma_{(m)}$ are the volume and surface area, respectively, of the mth irregular, C_2-cell. The vector of cell-average, conservative variables is

$$\overline{\mathbf{U}}_{(m)} = \begin{bmatrix} \bar{\rho} \\ \overline{\rho u} \\ \overline{\rho v} \end{bmatrix}_{(m)} = \frac{1}{\Omega_{(m)}} \int_{\Omega_{(m)}} \mathbf{U}_{\mathrm{m}}(\mathbf{x}, t) \mathrm{d}\Omega \tag{3}$$

The vectors of *point values*, $\mathbf{U}_{(m)}$, are defined at cell's centroids; and the *in-cell polynomials*, $\mathbf{U}_m(\mathbf{x}, t)$, represent third-order-accurate distributions. These distributions are constructed by least-squares interpolation (Section 4.1 and Appendix B), from point values and flow variables at interfacial vertices. The latter variables are double-valued, $\mathbf{U}_{(i)}^{+/-}$ (see Fig. 5), and computed with the sharp reconstruction algorithm described in Section 4.2. The vertices of cut-cells are either *internal* or *interfacial*. The latter ones are defined as those at the two ends of each *interfacial edge* (Fig. 5).

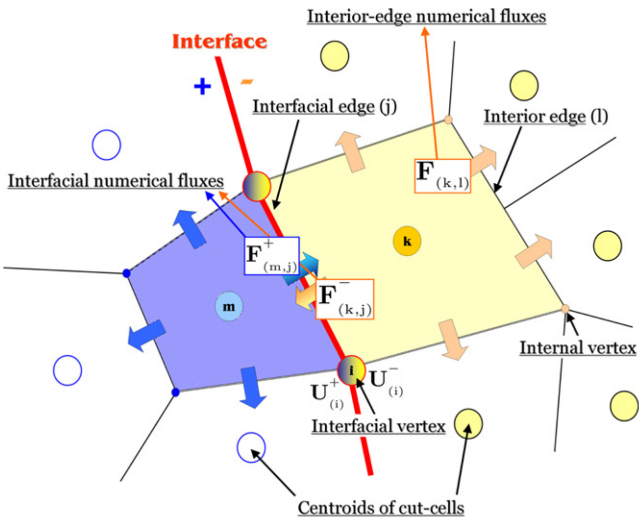

Fig. 5. The terminology and nomenclature employed in SIM. Explanations are given in the text.

Fluxes at each edge of C_2-cells are defined as

$$\mathbf{F} = \begin{bmatrix} \rho u_n \\ \rho u u_n + (P - \tau_{xx})n_x - \tau_{xy}n_y \\ \rho v u_n - \tau_{xy}n_x + (P - \tau_{yy})n_y \end{bmatrix} - \dot{\mathbf{x}} \cdot \mathbf{n} \begin{bmatrix} \rho \\ \rho u \\ \rho v \end{bmatrix} \tag{4}$$

where P and τ are the fluid pressure and viscous stress tensor, respectively; $\mathbf{n} = (n_x, n_y)$ is the outwards-pointing normal-to-the-edge unit vector; and $\dot{\mathbf{x}}$ is the cell-edge velocity (non-zero only for interfacial edges). The last term on the right-hand-side of Eq. (4) represents the *dynamic flux treatment* of interfacial cells [19,25]. Body forces are accounted for by the following source term

$$\mathbf{S} = \begin{bmatrix} 0 \\ \rho g_x \\ \rho g_y \end{bmatrix} \tag{5}$$

The semi-discrete representation of Eq. (2) is

$$\frac{\partial}{\partial t}\overline{\mathbf{U}}_{(m)}(t) + \frac{1}{\Omega_{(m)}}\sum_{e=1}^{N_{\mathrm{E}}} \Gamma_e \mathbf{F}_e = \overline{\mathbf{S}}_{(\mathbf{m})} \tag{6}$$

where N_{E} is the number of edges for the mth irregular cell, and Γ_e is the area of the eth-edge. Computation of numerical fluxes $\mathbf{F}_e$ requires in-cell polynomials, constructed from variable values at cell centroids and interfacial vertices. The former ones are recovered from cell averages by introducing a third-order correction as follows:

$$\begin{aligned} \mathbf{U}_{(m)} &= \overline{\mathbf{U}}_{(m)} - \delta^{(3)}\mathbf{U}_{(m)} + O(h^4) \\ \delta^{(3)}\mathbf{U}_{(m)} &= \frac{1}{\Omega_{(m)}}\left(\frac{C_{xx}^{(m)}}{2}\partial_{xx}\overline{\mathbf{U}}_{(m)} + C_{xy}^{(m)}\partial_{xy}\overline{\mathbf{U}}_{(m)} + \frac{C_{yy}^{(m)}}{2}\partial_{yy}\overline{\mathbf{U}}_{(m)}\right) \end{aligned} \tag{7}$$

where the moments of inertia

$$C_{xx}^{(m)} = \int_{\Omega_{(m)}} x^2 \mathrm{d}\Omega; \quad C_{xy}^{(m)} = \int_{\Omega_{(m)}} xy\,\mathrm{d}\Omega; \quad \text{and} \quad C_{yy}^{(m)} = \int_{\Omega_{(m)}} y^2 \mathrm{d}\Omega$$

are computed numerically for each irregular cell.

Mapping by Eq. (7) is used to (a) populate the $C_2(t^{(n)})$-grid upon its creation, on the basis of point values from regular C_1-grid and previous $C_2(t^{(n-1)})$-grid, Fig. 6, and (b) map new updates for cell-average variables to point values, as needed in the evaluation of fluxes. This is done at the end of each Runge–Kutta stage.

The evaluation of fluxes at the cell-edges adjacent to cells of same fluid, referred to as the *interior* cell-edges, is described in Section 4.1. The fluxes at the *interface-defining* edges require special treatment by incorporating the jump conditions. This, the essence of our method, is detailed in Section 4.2.

The overall time discretization is carried out with a three-stage Runge–Kutta TVD scheme [56], with an appropriate synchronization between C_2-grid solution and multi-level C_1-grid solution within the SAMR framework. After completing each time step $[t^{(n)}; t^{(n+1)}]$ on the finest SAMR level, the material velocity becomes available for moving the Level Set function and constructing the new C_2-grid, Fig. 6. Thus, the cycle repeats itself and the flow evolves accordingly.

4.1. Flux calculation on the interior cell-edges

Fluxes $\mathbf{F}_e$ at the edges between cells $_{(m)}$ and $_{(k)}$ are computed using the *Local-Lax-Friedrichs (LLF) flux splitting* algorithm

$$\mathbf{F}_e = \frac{1}{2}\left[\mathbf{F}\left(\mathbf{U}_e^{(m)}\right) + \mathbf{F}\left(\mathbf{U}_e^{(k)}\right) + \widehat{\lambda}_e \mathcal{S}\left(\mathbf{U}_e^{(m)} - \mathbf{U}_e^{(k)}\right)\right] \tag{8}$$

where the sign $\mathcal{S} = \pm 1$ depends on the convention used for the unit normal at the cell-edges. As demonstrated in [39], the slope for information flow in this upwinding scheme can be set to

$$\widehat{\lambda}_e = \alpha \mathrm{MAX}\left(\left|\lambda_s\left(\mathbf{U}_e^{(m)}\right)\right|, \left|\lambda_s\left(\mathbf{U}_e^{(k)}\right)\right|\right)_{s=1,\dots,3} \tag{9}$$

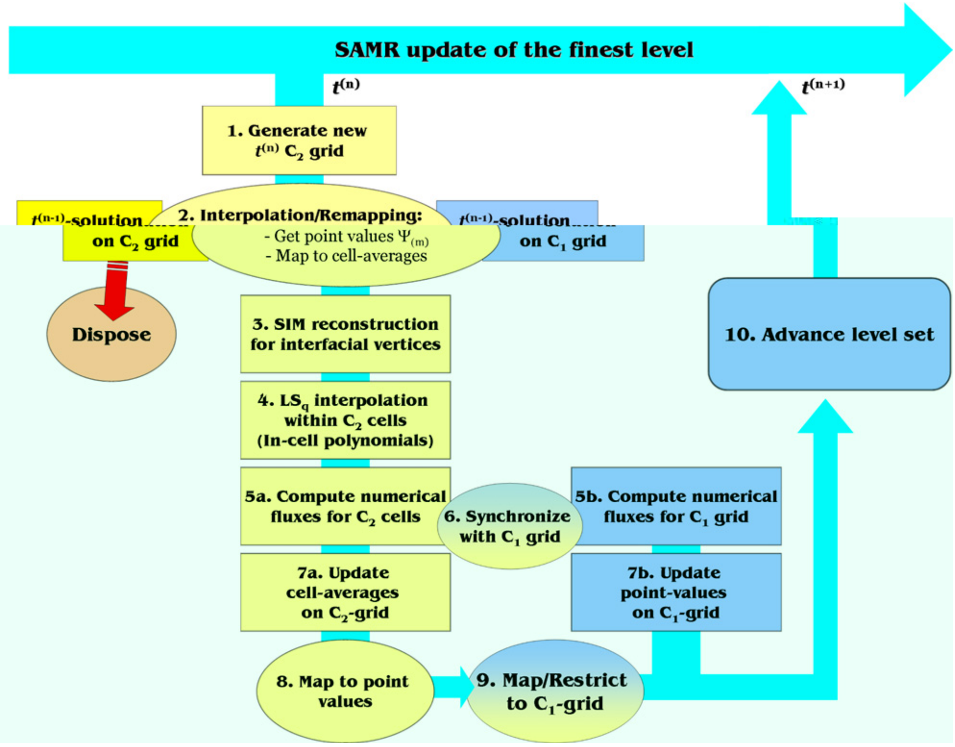
SAMR update of the finest level
t(n)
t(n+1)
1. Generate new t(n) C2 grid
t(n-1)-solution on C2 grid
2. Interpolation/Remapping:
- Get point values Ψ(m)
- Map to cell-averages
t(n-1)-solution on C1 grid
Dispose
3. SIM reconstruction for interfacial vertices
10. Advance level set
4. LSq interpolation within C2 cells (In-cell polynomials)
5a. Compute numerical fluxes for C2 cells
6. Synchronize with C1 grid
5b. Compute numerical fluxes for C1 grid
7a. Update cell-averages on C2-grid
7b. Update point-values on C1-grid
8. Map to point values
9. Map/Restrict to C1-grid

Fig. 6. Outline of the C_1–C_2-grid synchronization algorithm. The main steps involved in the generation/update/disposal of the C_2-grid are shown on the left (in yellow), while the main steps for time update of the C_1-grid are shown on the right (in blue). Interpolation, remapping and flux synchronization are steps 2, 6 and 9.

where λ_s is the sth-component of the eigenvalue vector of the convective flux, $[u+c, c, u-c]^{\mathrm{T}}$, and c is the numerical sound speed chosen to maintain numerical Mach number below 10^{-3}. Nominally, for compressible flow α is set to be O(1). In the nearly-incompressible regime, the value $\alpha = 0.05$ is sufficient to avoid central-differencing for pressure gradient operator and the associated decoupling on odd and even grid points, while providing a low level of numerical diffusion. More about pseudocompressible methods for incompressible flows and comparison with projection and LBE methods can be found in [14,37–39].

Edge values $\mathbf{U}_e^{(m)}$ and $\mathbf{U}_e^{(k)}$ for the above flux integration are provided by the third-order-accurate Least-Squares interpolation (described in Appendix B and denoted hereafter as LSq$_3$), using support datasets sampled in the following manner[4]:

(a) First, we take data from the *centroids of the nearest irregular C_2 cells*, and *centers of regular C_1 cells*, all belonging to the same fluid as the treated cell.

(b) Second, we check the vertices of the neighbor cut-cells. If any of these *vertices* are *interfacial*, then same-fluid values are taken into the interpolation dataset.

With this sampling we avoid numerical mixing across the interface.

It is important to emphasize that the numerical fluxes must be synchronized at the boundary between regular (C_1) and irregular (C_2)-grids (Fig. 6), to ensure conservation. This is implemented in a manner similar to that employed by Berger and Collela [5] for SAMR-synchronization of fluxes between cells with different grid resolutions. More specifically, at the edges interfacing C_1 and C_2 cells, we use the single numerical flux taken from the flux calculation on unstructured C_2-grid, thus preventing mass and momentum losses at the C_1–C_2-grid boundary.

4.2. Flux calculation on the interface-defining cell-edges

The treatment of numerical fluxes on interfacial edges is *non-conservative*, as necessary for multimaterial interfaces [1,2,15,40]. Using the notation of Fig. 5, these fluxes are $\mathbf{F}^{+}_{(m,j)} = \mathbf{F}(\mathbf{U}_e^{(m)})$ and $\mathbf{F}^{-}_{(k,j)} = \mathbf{F}(\mathbf{U}_e^{(k)})$, where the edge values $\mathbf{U}_e^{(m)}$ and $\mathbf{U}_e^{(k)}$ are provided by the LSq$_3$ using support datasets sampled as described in the end of Section 4.1. Thus, the principal task is to obtain a sharp reconstruction at the interfacial vertices – this is done by incorporating the jump conditions at the interface through a *third-order-accurate, sided,*[4] *least-squares interpolation*, carried out as follows.

Flow fields for primitive variables around each interfacial vertex are approximated with the following sided[5] multi-dimensional Taylor expansions

$$\vec{\Psi}^{+}(x,y) = \begin{bmatrix} u_{\mathrm{I}} + (\partial_x^{+} u)x + (\partial_y^{+} u)y + a_1^{+}x^2 + a_2^{+}xy + a_3^{+}y^2 + \mathrm{O}(h^3) \\ v_{\mathrm{I}} + (\partial_x^{+} v)x + (\partial_y^{+} v)y + b_1^{+}x^2 + b_2^{+}xy + b_3^{+}y^2 + \mathrm{O}(h^3) \\ P_{\mathrm{I}}^{+} + (\partial_x^{+} P)x + (\partial_y^{+} P)y + \mathrm{O}(h^2) \end{bmatrix} \tag{10}$$

and

$$\vec{\Psi}^{-}(x,y) = \begin{bmatrix} u_{\mathrm{I}} + (\partial_x^{-} u)x + (\partial_y^{-} u)y + a_1^{-}x^2 + a_2^{-}xy + a_3^{-}y^2 + \mathrm{O}(h^3) \\ v_{\mathrm{I}} + (\partial_x^{-} v)x + (\partial_y^{-} v)y + b_1^{-}x^2 + b_2^{-}xy + b_3^{-}y^2 + \mathrm{O}(h^3) \\ P_{\mathrm{I}}^{-} + (\partial_x^{-} P)x + (\partial_y^{-} P)y + \mathrm{O}(h^2) \end{bmatrix} \tag{11}$$

Note that *velocity continuity* is already accounted for. Similar to the LSq$_3$ interpolation discussed in Appendix B, we solve for the following vector of unknowns:

$$\mathbf{X} = (u_{\mathrm{I}}, v_{\mathrm{I}}, P_{\mathrm{I}}^{+}, P^{-}{}_{\mathrm{I}}, \partial_x^{+}u, \partial_x^{+}v, \partial_x^{+}P, \partial_x^{-}u, \partial_x^{-}v, \partial_x^{-}P, \partial_y^{+}u, \partial_y^{+}v, \ldots \text{ etc } \ldots)^{\mathrm{T}} \tag{12}$$

[4] As described in Appendix B, the LSq$_3$ provides also second-order-accurate approximations of spatial derivatives, which are used for computation of viscous stresses.

[5] We use *sided* instead of *one-sided* to avoid potential confusion by the presence of two sides on which this process is implemented.

corresponding to 28 variables in total per interfacial vertex. Truncating expansions Eqs. (10) and (11) to the second-order, the number of unknowns can be reduced to 16. Hereafter, the third-order and second-order sided interpolations are denoted as SIM$_3$ and SIM$_2$, respectively.

To close the problem, we (a) use the two stress jump conditions, Eq. (1) and (b) require the polynomials of Eqs. (10) and (11) to pass through the gravity centers of the nearest irregular cells (as close as possible, in the Least-Squares sense). The various pieces of information developed for these purposes may be listed as follows:

(**a**) Normal and shear components of the viscous stress tensor are expressed by:

$$\mathcal{T}_{\xi\xi} = \frac{2}{3}\mu(e_1\partial_x u + e_2\partial_x v + e_3\partial_y u + e_4\partial_y v + e_5 u + e_6 v) \tag{13}$$

and

$$\mathcal{T}_{\xi\eta} = \mu(f_1\partial_x u + f_2\partial_x v + f_3\partial_y u + f_4\partial_y v + f_5 u + f_6 v) \tag{14}$$

where the coefficients of transformation to the *normal-to-interface* coordinate system are computed as:

$$\begin{aligned}
&e_1 = 2x_\xi\xi_x - x_\eta\eta_x \; e_2 = 2x_\xi\xi_y - x_\eta\eta_y \\
&e_3 = 2y_\xi\xi_x - y_\eta\eta_x \; e_4 = 2y_\xi\xi_y - y_\eta\eta_y \\
&e_5 = 2(x_\xi\partial_x\xi_x + y_\xi\partial_y\xi_x) - x_\eta\partial_x\eta_x - y_\eta\partial_y\eta_x \\
&e_6 = 2(x_\xi\partial_x\xi_y + y_\xi\partial_y\xi_y) - x_\eta\partial_x\eta_y - y_\eta\partial_y\eta_y - \frac{\varsigma}{y}
\end{aligned} \tag{15}$$

and

$$\begin{aligned}
&f_1 = x_\xi\eta_x + x_\eta\xi_x \; f_2 = x_\xi\eta_y + x_\eta\xi_y \\
&f_3 = y_\xi\eta_x + y_\eta\xi_x \; f_4 = y_\xi\eta_y + y_\eta\xi_y \\
&f_5 = x_\xi\partial_x\eta_x + x_\eta\partial_x\xi_x + y_\xi\partial_y\eta_x + y_\eta\partial_y\xi_x \\
&f_6 = x_\xi\partial_x\eta_y + x_\eta\partial_x\xi_y + y_\xi\partial_y\eta_y + y_\eta\partial_y\xi_y
\end{aligned} \tag{16}$$

(**b**) The interfacial geometry is expressed by:

$$\begin{aligned}
&\text{Unit normal}: \mathbf{n} = \frac{\nabla\varphi}{|\nabla\varphi|} = (\partial_x\varphi, \partial_y\varphi) = (\xi_x, \xi_y) \\
&\text{Unit tangential}: \; \mathbf{t} = (\eta_x, \eta_y) = (\xi_y, -\xi_x) \\
&\text{Curvature}: \; \kappa = \nabla\cdot\left(\frac{\nabla\varphi}{|\nabla\varphi|}\right) = \partial_x\xi_x + \partial_y\xi_y + \varsigma\frac{\xi_y}{y}
\end{aligned} \tag{17}$$

where $\varsigma = 0$ and 1 for Cartesian and cylindrical coordinate systems, respectively.

(**c**) The *normal* and *tangential* components of the velocity vector are obtained from

$$\begin{bmatrix} V_\xi \\ V_\eta \end{bmatrix} = \mathbb{J}\begin{bmatrix} u \\ v \end{bmatrix} \tag{18}$$

where the Jacobian matrix for coordinate transformation and its inverse are defined as

$$\mathbb{J} = \begin{bmatrix} \xi_x & \xi_y \\ \eta_x & \eta_y \end{bmatrix}, \mathbb{J}^{-1} = \begin{bmatrix} x_\xi & x_\eta \\ y_\xi & y_\eta \end{bmatrix} = \begin{bmatrix} \xi_x & \xi_y \\ \xi_y & -\xi_x \end{bmatrix} \tag{19}$$

Importantly, the Level Set-based computation of viscous stresses allows to account for interfacial curvature (with $e_{5,6}$ and $f_{5,6}$).

(**d**) Derivatives involved in Eqs. (15)–(17) are evaluated in the following way:

- *First*, we compute regular-grid point values of the Level Set derivatives, $(\partial_x\varphi, \partial_y\varphi) = (\xi_x, \xi_y)$, using the fourth-order central differencing.

- *Second*, using the above-computed regular-grid values of ξ_x, ξ_y, their spatial derivatives are calculated at the regular-grid, again with the fourth-order central differencing.
- *Finally*, for each interfacial vertex, we reconstruct normals, tangentials and their spatial derivatives using the fourth-order cubic interpolation along the corresponding gridline.

(e) Now, the stress jump conditions (JC1 and JC2 in Eq. (1)) are written as:

$$\frac{\mu^+}{\mu^-}\left(f_1\partial_x^+u+f_2\partial_x^+v+f_3\partial_y^+u+f_4\partial_y^+v+f_5u_{\mathrm{I}}+f_6v_{\mathrm{I}}\right)=f_1\partial_x^-u+f_2\partial_x^-v+f_3\partial_y^-u+f_4\partial_y^-v+f_5u_{\mathrm{I}}+f_6v_{\mathrm{I}} \tag{20}$$

and

$$\begin{aligned}&\frac{2\mu^+}{3}\left(e_1\partial_x^+u+e_2\partial_x^+v+e_3\partial_y^+u+e_4\partial_y^+v+e_5u_{\mathrm{I}}+e_6v_{\mathrm{I}}\right)-P_{\mathrm{I}}^+\\&\quad=\frac{2\mu^-}{3}\left(e_1\partial_x^-u+e_2\partial_x^-v+e_3\partial_y^-u+e_4\partial_y^-v+e_5u_{\mathrm{I}}+e_6v_{\mathrm{I}}\right)-P_{\mathrm{I}}^-+\sigma\kappa\end{aligned} \tag{21}$$

which provide two of the equations needed for closure of the SIM interpolation. The rest 26 (SIM$_3$), or 14 (SIM$_2$) closing equations are derived from the following minimization problem. For example in SIM$_3$ this amounts to

$$\mathrm{MIN}\left\{\begin{aligned}&\mathcal{E}_{|\mathbf{u}|}\equiv\sum_{i=1}^{N_+}\omega_i^+\left[\left(u_i^{\left(\mathrm{LSq}_3^+\right)}-u_i\right)^2+\left(v_i^{\left(\mathrm{LSq}_3^+\right)}-v_i\right)^2\right]\\&\qquad+\sum_{i=1}^{N_-}\omega_i^-\left[\left(u_i^{\left(\mathrm{LSq}_3^-\right)}-u_i\right)^2+\left(v_i^{\left(\mathrm{LSq}_3^-\right)}-v_i\right)^2\right]\\&\mathcal{E}_{\mathrm{P}}\equiv\sum_{i=1}^{N_+}\omega_i^+\left(P_i^{\left(\mathrm{LSq}_3^+\right)}-P_i\right)^2+\sum_{i=1}^{N_-}\omega_i^-\left(P_i^{\left(\mathrm{LSq}_3^-\right)}-P_i\right)^2\end{aligned}\right\} \tag{22}$$

where $\vec{\Psi}_i=(u_i,v_i,P_i)$ is the known solution at the ith datapoint with coordinate $\mathbf{x}_i$ and $\vec{\Psi}_i^{\left(\mathrm{LSq}_3^\pm\right)}$ is the corresponding LSq$_3$ interpolation by Eqs. (10) and (11). This results in[6] 26 algebraic equations

$$\begin{aligned}&\frac{\partial\mathcal{E}_{|\mathbf{u}|}}{\partial(\partial_x^+u)}=\frac{\partial\mathcal{E}_{|\mathbf{u}|}}{\partial\left(\partial_y^+u\right)}=\frac{\partial\mathcal{E}_{|\mathbf{u}|}}{\partial(\partial_x^-u)}=\frac{\partial\mathcal{E}_{|\mathbf{u}|}}{\partial\left(\partial_y^-u\right)}=\frac{\partial\mathcal{E}_{|\mathbf{u}|}}{\partial(\partial_x^+v)}=\frac{\partial\mathcal{E}_{|\mathbf{u}|}}{\partial\left(\partial_y^+v\right)}=\frac{\partial\mathcal{E}_{|\mathbf{u}|}}{\partial(\partial_x^-v)}=\frac{\partial\mathcal{E}_{|\mathbf{u}|}}{\partial\left(\partial_y^-v\right)}=\frac{\partial\mathcal{E}_{|\mathbf{u}|}}{\partial(\partial_{xx}^+u)}\\&=\frac{\partial\mathcal{E}_{|\mathbf{u}|}}{\partial\left(\partial_{xy}^+u\right)}=\frac{\partial\mathcal{E}_{|\mathbf{u}|}}{\partial\left(\partial_{yy}^+u\right)}=\frac{\partial\mathcal{E}_{|\mathbf{u}|}}{\partial(\partial_{xx}^-u)}=\frac{\partial\mathcal{E}_{|\mathbf{u}|}}{\partial\left(\partial_{xy}^-u\right)}=\frac{\partial\mathcal{E}_{|\mathbf{u}|}}{\partial\left(\partial_{yy}^-u\right)}=\frac{\partial\mathcal{E}_{|\mathbf{u}|}}{\partial(\partial_{xx}^+v)}=\frac{\partial\mathcal{E}_{|\mathbf{u}|}}{\partial\left(\partial_{xy}^+v\right)}=\frac{\partial\mathcal{E}_{|\mathbf{u}|}}{\partial\left(\partial_{yy}^+v\right)}\\&=\frac{\partial\mathcal{E}_{|\mathbf{u}|}}{\partial(\partial_{xx}^-v)}=\frac{\partial\mathcal{E}_{|\mathbf{u}|}}{\partial\left(\partial_{xy}^-v\right)}=\frac{\partial\mathcal{E}_{|\mathbf{u}|}}{\partial\left(\partial_{yy}^-v\right)}=\frac{\partial\mathcal{E}_{\mathrm{P}}}{\partial P_{\mathrm{I}}^+}=\frac{\partial\mathcal{E}_{\mathrm{P}}}{\partial P_{\mathrm{I}}^-}=\frac{\partial\mathcal{E}_{\mathrm{P}}}{\partial(\partial_x^+P)}=\frac{\partial\mathcal{E}_{\mathrm{P}}}{\partial\left(\partial_y^+P\right)}=\frac{\partial\mathcal{E}_{\mathrm{P}}}{\partial(\partial_x^-P)}=\frac{\partial\mathcal{E}_{\mathrm{P}}}{\partial\left(\partial_y^-P\right)}\\&=0\end{aligned} \tag{23}$$

(f) The rest of the SIM reconstruction is similar to the LSq$_3$ interpolation described in Appendix B – i.e. (Eqs. (20),(21),(23)) are converted into the linear algebra

$$\mathbb{A}_{\mathrm{SIM}}\mathbf{X}=\mathbf{B}_{\mathrm{SIM}} \tag{24}$$

For ω_i^- and ω_i^+, the weights in Eq. (B.7) are utilized. The support for SIM interpolation is sampled from the nearest cut-cells on the *positive* and *negative* sides of the interface, respectively, taking point values of velocity and pressure from the corresponding gravity centers. To ensure non-degeneracy of the linear algebra, datasets

[6] In difference to LSq$_3$, the minimization is constrained by requiring to satisfy stress jump conditions JC1 and JC2. Therefore, two minimization conditions in (B.4) must be dropped and substituted by Eqs. (20) and (21). We chose to drop $\frac{\partial\mathcal{E}_{|\mathbf{u}|}}{\partial u_{\mathrm{I}}}=0$ and $\frac{\partial\mathcal{E}_{|\mathbf{u}|}}{\partial v_{\mathrm{I}}}=0$, since careful examination of Eq. (23) reveals that dropping any other couple of minimization conditions would involve giving preference to one of the spatial directions.

are oversampled ($N_+ > 4$ and $N_- > 4$). Finally, the solution $\mathbf{X}$ is obtained by the standard LU decomposition of $\mathbb{A}_{\text{SIM}}$.

5. Basic tests

In preparation for the applications of principal interest to this work, given in Section 6, here we show results for a couple of basic convergence tests. One is the Laplace-law test, namely the simulation of a gas bubble in equilibrium with a stagnant liquid (no body forces). It provides the most sensitive diagnostic of the spurious current problem, as shown by previous works that addressed themselves to potential remedies. One result of these works is that the amplitude of the spurious current scales as $U_{\text{sp}} \sim C_{\text{sp}} \frac{\sigma}{\mu}$, where C_{sp} varies between 10^{-2} and 10^{-5} depending on the choice of mollifying function used [33,53]. More recently, in-depth assessments have been provided by Francois et al. [16] & Renardy and Renardy [48]. With their VOF codes, they apply interfacial re-balancing and parabolic reconstruction in computing curvature and show good results. We show that the sharp treatment can provide a very effective remedy too.

The second test, based on the method of “manufactured” solutions [31], is constructed as an analytical representation of two idealized, flow fields that are made to satisfy the interfacial jump conditions. The principal purpose of the test is to verify grid convergence rates of our SIM interface reconstruction procedure. To this end, interfacial values of primitive variables and stresses are recovered at interfacial vertices by SIM_2 and SIM_3 from point values on C_1 and C_2 meshes, and they are compared to the exact results.

5.1. The Laplace-law test

Sample computations were carried out in the computational domain $[1 \times 1]$ with a static bubble of radius $\frac{1}{2}$ placed at the center. Surface tension σ was set to 1, density ratio to 10, and viscosities equal to each other.

The results are summarized in Fig. 7. Fig. 7(a) shows the steady-state pressure distribution. Fig. 7(b) depicts reduction of discretization errors with grid refinement. The errors are measured by the $\mathcal{L}_1$-norm defined by

$$\mathcal{L}_1(\Phi) \equiv \frac{\sum_{m=1,\dots,N} \left| \Phi_m^{\text{exact}} - \Phi_m^{\text{computed}} \right|}{N}$$

where N is the total number of cells.

SIM_3 converges with the fourth-order, corresponding to the leading order of errors in computation of curvature.

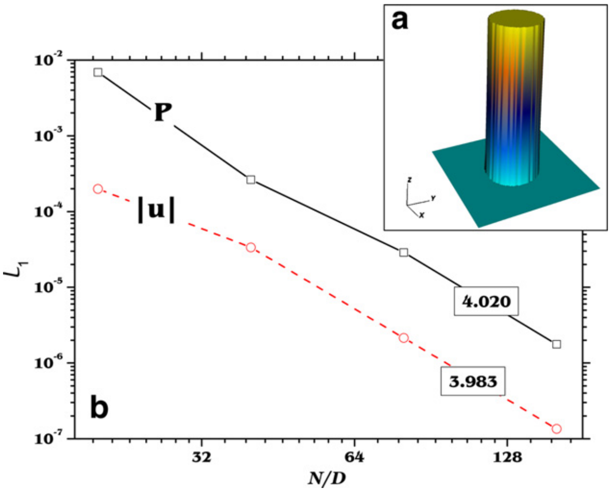

Fig. 7. Error in pressure and velocity fields for the Laplace-law test, using SIM_3. (a) Pressure distribution. (b) Grid convergence (rates are shown in boxes; grid resolution is defined as a number of nodes per bubble’s diameter).

5.2. The spinning bubble test

A bubble of radius $\mathcal{R} = 0.4$ is placed in the center of the computational domain $[1 \times 1]$. The pressure (P) and velocity ($\mathbf{u} = (V_\xi, V_\eta)$) (see Eq. (18)) fields inside ($\varphi < 0$) and outside ($\varphi > 0$) the bubble are set to (Fig. 8)

$$\begin{aligned} P^-(r) &= P_0 + (P_\mathrm{I}^- - P_0)\left(\frac{r}{\mathcal{R}}\right)^3 \\ V_\xi^-(r) &= V_{\xi_\mathrm{I}}\left(\frac{r}{\mathcal{R}}\right)^2 \\ V_\eta^-(r) &= V_{\eta_\mathrm{I}}\left(\frac{r}{\mathcal{R}}\right)^3 \end{aligned} \tag{25}$$

and

$$\begin{aligned} P^+(r) &= \left(P_\mathrm{I}^- - \frac{\sigma}{\mathcal{R}}\right)\frac{\mathcal{R}}{r} \\ V_\xi^+(r) &= V_{\xi_\mathrm{I}}\left(1 - 2m\frac{r}{\mathcal{R}}\left(1 - \frac{r}{R}\right)\right) \\ V_\eta^+(r) &= V_{\eta_\mathrm{I}}\left(1 - 3m\left(1 - \frac{r}{\mathcal{R}}\right)\right) \end{aligned} \tag{26}$$

with $P_0 = 20$, $P_\mathrm{I}^+ = 10$, $V_{\xi_\mathrm{I}} = 1$, $V_{\eta_\mathrm{I}} = 1$; $\sigma = 1$ and $m = \frac{\mu^+}{\mu^-} = 10^{-2}$, which yield at the interface

$$V_{\xi_\mathrm{I}} = 1, \quad V_{\eta_\mathrm{I}} = 1, \quad \mathcal{T}_{\xi\xi_\mathrm{I}} = \frac{8\mu^+}{3\mathcal{R}} \quad \text{and} \quad \mathcal{T}_{\xi\eta_\mathrm{I}} = \frac{3\mu^+}{\mathcal{R}} \tag{27}$$

Results are summarized in Fig. 9 in terms of the $\mathcal{L}_1$-norm of errors defined above. We can see that the SIM converges with nearly-theoretical rates, i.e. with second (SIM$_2$) and third (SIM$_3$) orders for velocity and with first (SIM$_2$) and second (SIM$_3$) orders for shear stresses. Moreover, SIM$_3$ is orders of magnitude more accurate than SIM$_2$.

Notably, even though we treat pressure piecewise-linearly (see Eqs. (10) and (11)), it converges with only first-order in SIM$_2$. This is a reflection of the fact that we do not operator-split pressure and viscous stresses as done in DIMs, for which JC1 is never explicitly satisfied. Rather in SIM pressure and viscous stresses are tightly coupled through Eqs. (20) and (21) (to satisfy both JC1 and JC2), and thusly the apparent convergence

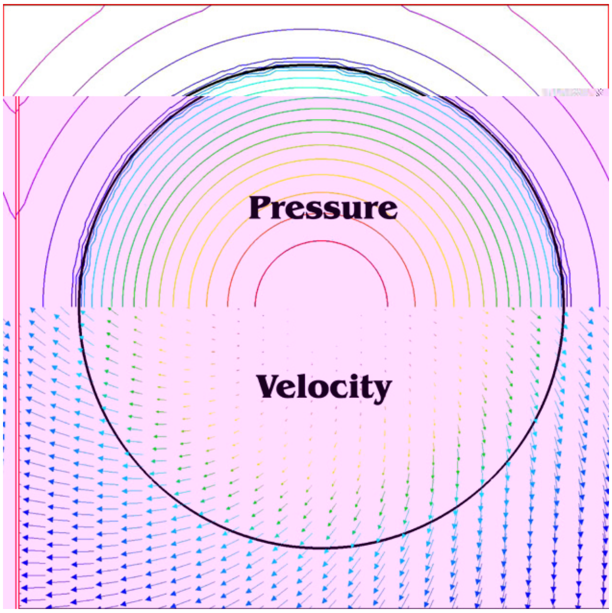

Fig. 8. Illustration of pressure and velocity fields in the “Spinning-Bubble” problem.

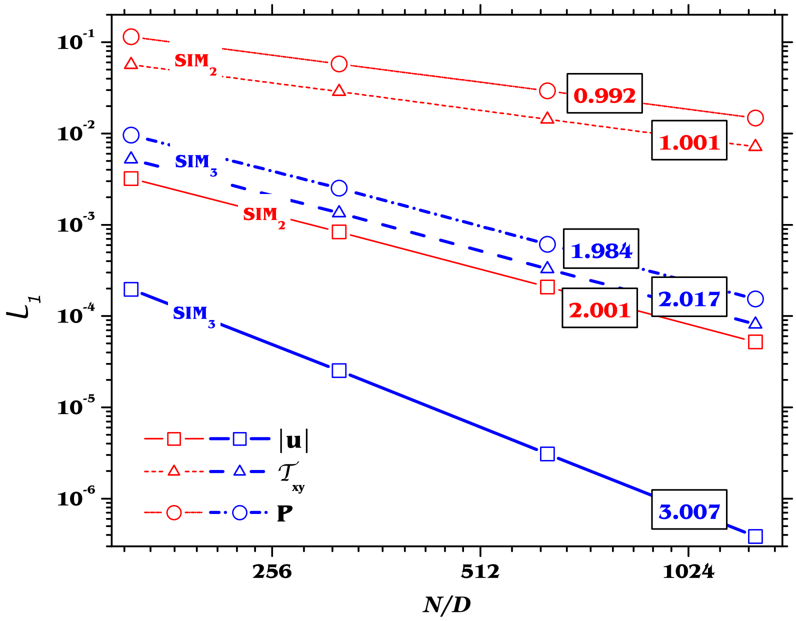

Fig. 9. Convergence rates of the SIM, applied to the “Spinning-Bubble” problem. The grid-resolution is given by the number of nodes per bubble diameter.

rate is dictated by the leading-order errors due to discretization of viscous normal stresses. These are the first-order in SIM$_2$, making the pressure converge with only first-order (note that pressure does converges with the second-order when the discretization of the normal stresses is second-order in SIM$_3$). This tight coupling of viscous normal stresses and pressure terms is especially important for low-R-number flows, for which the viscous operator is more significant.

6. Numerical prediction of interfacial instabilities

In this Section we show that with MuSiC–SIM we can attain direct numerical simulations of *Rayleigh–Taylor* (R–T) and *viscous-Kelvin–Helmholtz* (vK–H) instabilities in the linear (exponential growth) regime. In particular, we demonstrate higher-than-first-order convergence to the results of the linear stability theory over a broad range of conditions, including capability to handle density/viscosity ratios typical of gas–liquid systems. Also, we demonstrate self-selection of the most dangerous wave in R–T, and capability to predict neutral-stability maps in vK–H flows. In all cases, the simulations start with a simple seeding an interfacial shape disturbance thereby also capturing the early acceleration regime, a key aspect of the behavior not available by linear theory. Sample results for vK–H reveal transition to non-linear growth of surprisingly limited extent (duration, amplitude, see Fig. 16).

6.1. Rayleigh–Taylor flows

We consider here the classical Rayleigh–Taylor problem – *two fluids of unequal densities subject to a body force field (or equivalent acceleration) directed normally to their common, planar interface*. When the direction of acceleration is from the light to the heavy fluid, the interface is unstable to small perturbations, waves grow exponentially in time (linear regime), and eventually the interface breaks up into a two-phase mixing zone that grows algebraically in time (non-linear regime). Such a problem is specified completely by the Atwood number (At), the Reynolds number (R), the surface tension parameter (S), and the viscosity ratio (m) defined by

$$At = \frac{\rho_h - \rho_l}{\rho_h + \rho_l}; \quad R_k = \frac{\lambda\sqrt{\lambda g}}{v_k}; \quad S_k = \frac{\sigma}{(\rho_h + \rho_l)\sqrt[3]{g v_k^4}}; \quad m = \frac{\mu_h}{\mu_l} \tag{28}$$

where the light ($k = l$) and heavy ($k = h$) fluids are taken so as to satisfy the condition of instability noted above. In defining these parameters, the natural length and velocity scales were taken as wave-length λ and $\vartheta = \sqrt{\lambda g}$, which yield a characteristic time of $\tau_{\rm RT} = \sqrt{\frac{\lambda}{g}}$, where g is the applied acceleration. In all simulation shown here, kinematic viscosities of fluids set to be are equal, $\nu_l = \nu_h = \nu$.

In presenting the results, we will refer to scaled time $\left(t^\star = \frac{t}{\tau_{\rm RT}}\right)$, maximum interface displacement $\left(\varepsilon^\star_{\rm RT} = \frac{\varepsilon}{\lambda}\right)$, and dimensionless growth factor $\left(K^\star_{\rm RT}\right)$ defined by

$$K^\star_{\rm RT} = \frac{\ln\frac{\varepsilon(t)}{\varepsilon(0)}}{t}\frac{1}{\sqrt[3]{g^2/\nu}} \tag{29}$$

The wave-number α is defined as $\frac{2\pi}{\lambda}$ and scaled as $\alpha^\star_{\rm RT} = \frac{\alpha}{\sqrt[3]{g/\nu^2}}$ [12]. It is generally thought that in the absence of a particular (imposed) initial wave (the mode), the instability will self-select the fastest growing, or so-called *most dangerous* wave. In our sample simulations, we are interested to show convergence to the exact analytic results, over a wide range of conditions, including in particular: (a) At numbers that span typical density values in gas–gas (or liquid–liquid) and gas–liquid systems and (b) the presence or absence of interfacial tension. From a numerical prowess standpoint, we wish to also demonstrate capability to handle arbitrarily large density and viscosity ratio systems.

All simulations were carried out with SIM$_3$, and some included grid adaptation. The DIM calculations were carried out by using the mollifying function defined by Eqs. (C.1)–(C.4), applied in the corridor $\delta = 2h$ (two nodes on each side of the interface) on a Cartesian grid (Appendix C).

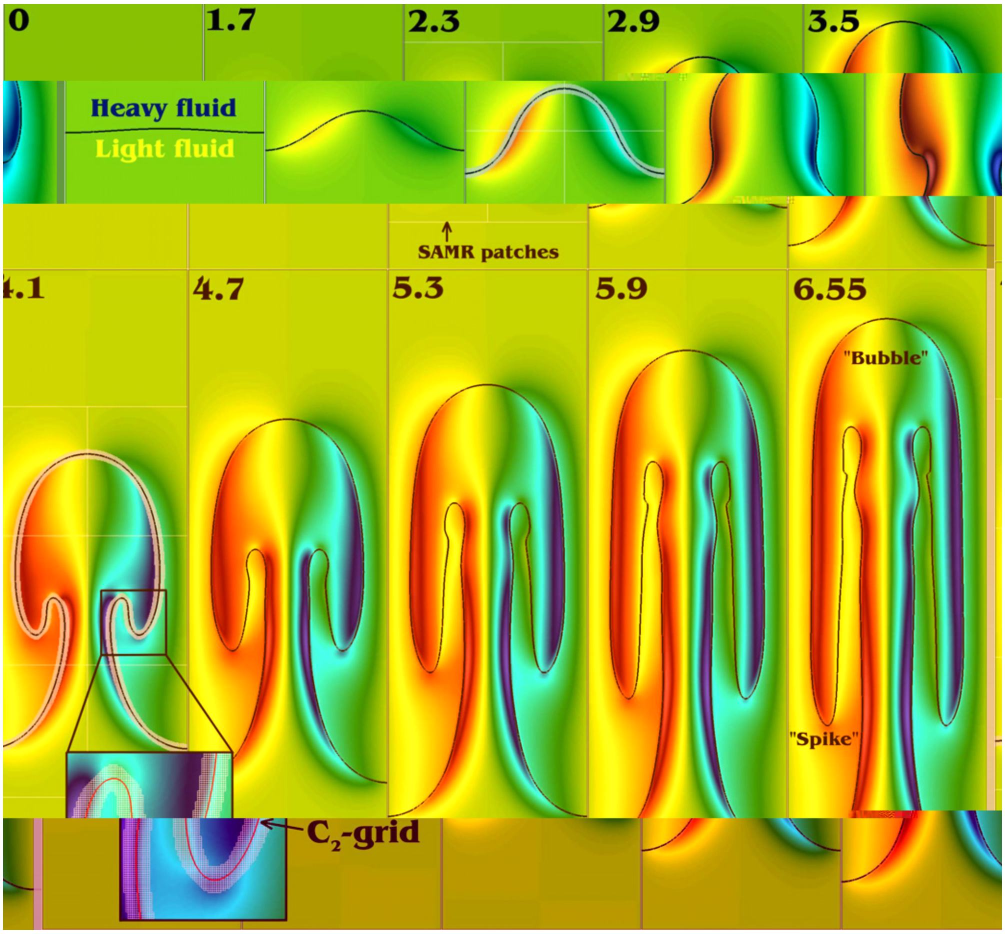

Fig. 10. Dynamics of interfacial shape and vorticity field for single-mode Rayleigh–Taylor instability. $At = 1/2$, $R = 76$, $m = 3$, $S = 0$. Structure of the aC^{3}MR grid is shown for $t^\star = 2.3$ and 4.1. Effective grid resolution by SAMR$_{L2\text{-}R4}$ is 256 nodes/λ.

In Fig. 10 we illustrate the overall growth dynamics and the well-known bubble-spike pattern that develops well into the non-linear regime. The At number value is relatively low, and the density ratio (3) is modest as is typical of same-phase systems. As shown in the insets, this kind of simulation is greatly benefited by grid adaptation.

Comparisons with the linear theory [12] for small and large values of density ratios are shown in Figs. 11 and 12, respectively. From these we can also visualize the effects of surface tension and viscosity, as well as appreciate that the full-range of interest in density and viscosity ratios is tractable by the SIM.

Besides the excellent fidelity of SIM, one could see also the remarkably good performance of DIM. From our overall experience, this is a general result for this class of (R–T) problems in the linear regime, at not-too-small values of R considered (>5), where shear effects are not too strong even in the presence of large viscosities.

Amplitude data for the case $At = 1/2$ are shown, along with deduced history of the bubble acceleration constant taken to characterize experiments in the non-linear regime [18], in Fig. 13. Past experience with such simulations, and related experimental evidence are summarized in Table 1.

Notably, an acceleration time of $t^\star \sim 1$ is needed for the growth to become exponential; a useful result in providing physical connection to real-time behavior of instability in practical settings, not available from

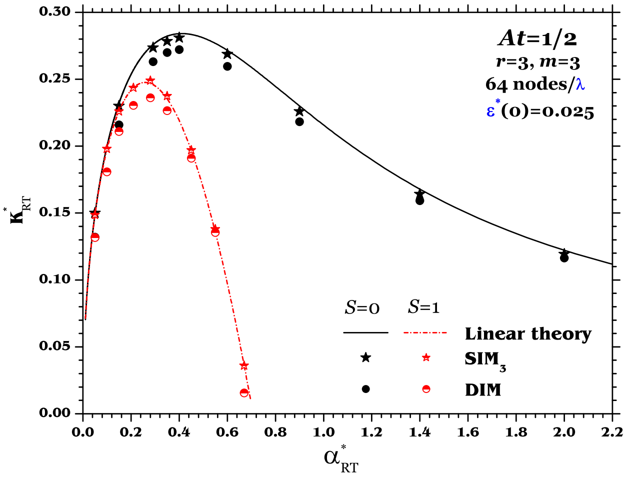

Fig. 11. Dependence of the disturbance growth factor $K^\star_{RT}$ on wave-number $\alpha^\star_{RT}$, for small-density/viscosity-ratio configurations.

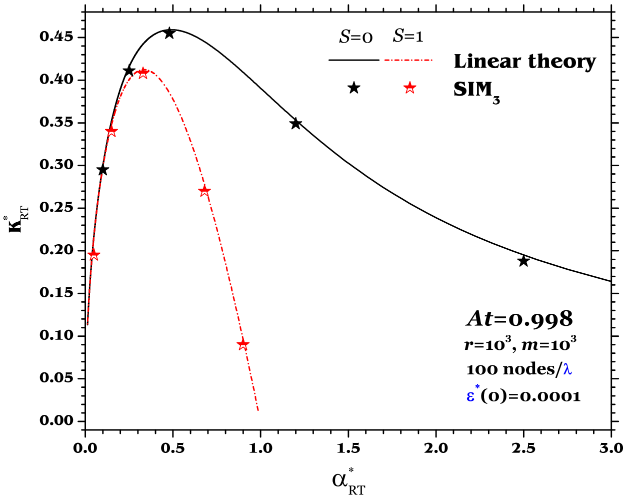

Fig. 12. Dependence of the disturbance growth factor $K^\star_{RT}$ on wave-number $\alpha^\star_{RT}$, for large-density/viscosity-ratio configurations.

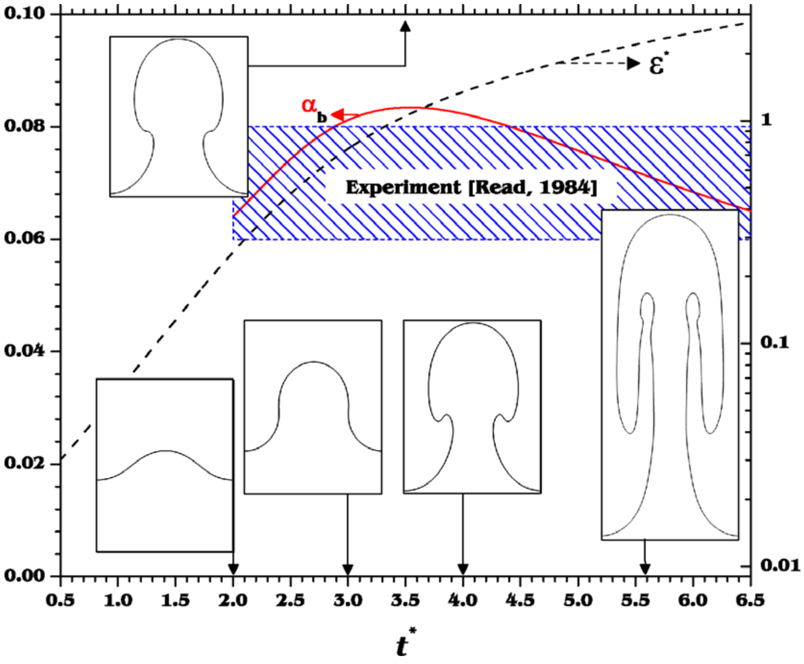

Fig. 13. Illustration of transition to non-linear growth and comparison to experimental data (in the non-linear regime). Bubble acceleration constant is defined as $\alpha_b = \frac{\varepsilon(t)}{At\ g\ t^2}$. $At = 1/2$, $R = 76$, $m = 3$, $S = 0$. Effective grid resolution by SAMR$_{L2\text{-}R4}$ is 256 nodes/λ.

Table 1
Summary of the experimental and simulation data for α_b

Experiments			
Read [47]		1984	$\alpha_b \sim 0.063$–0.077
Kucherenko et al. [32]		1991	$\alpha_b \sim 0.07$
Snider and Andrews [58]		1994	$\alpha_b \sim 0.07$
Simulations			
Youngs [73,74]	(VOF)	1984	$\alpha_b \sim 0.04$–0.05
He et al. [24]	(LBGK)	1999	$\alpha_b \sim 0.04$
Glimm et al. [18]	(FT-DIM)	2001	$\alpha_b \sim 0.07$
Present study	(LS-SIM$_3$)		$\alpha_b \sim 0.06$–0.08

linear theory. At the other end, transition to non-linear growth is seen to occur at $t^\star \sim 2$ ($\varepsilon^\star \sim 0.3$). As we will see in the next section, this relatively narrow lifespan of the linear regime can be even smaller in vK–H flows, a matter of considerable significance in validation exercises of numerical simulations.

Finally, a sample of multimode growth and self-selection is illustrated in Fig. 14. This simulation was run with an initial disturbance on the level set function (φ) given by

$$\varphi(\mathbf{x}) = y - \left[H_0 + \varepsilon^\star(0)\sum_{i=0}^{7}\cos\left(\frac{\pi\alpha_i}{W}x\right)\right] \tag{30}$$

where $W = 1$ is the width of the computational domain, H_0 is the position of the interface, $\varepsilon^\star(0) = 1.25 \times 10^{-4}$ is the amplitude of initial disturbance and α_i are the wave-numbers (4, 14, 22, 28, 34, 42, 52, 58), selected at random. In the results we can see the complex manner in which the dominant mode emerges, and also that, as expected, this mode has a wave-number of 5–6, in a good agreement with the theoretical prediction for the *most-dangerous* wave (5.767).

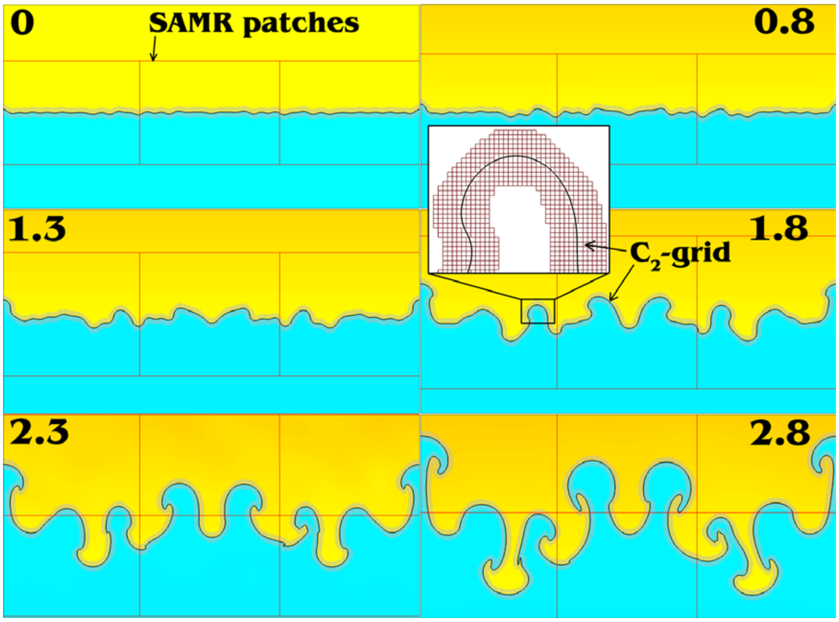

Fig. 14. Interfacial shape evolution in multi-mode Rayleigh–Taylor instability. $At = 1/3$, $R = 103$, $m = 2$, $r = 2$, $S = 0$. Effective grid resolution by SAMR$_{L2\text{-}R4}$ is 512 nodes/W.

6.2. Viscous Kelvin–Helmholtz flows

The classical viscous Kelvin–Helmholtz problem involves *flows of two superimposed real fluids parallel to their common interface*. The remarkable result of Yih [72] is that the interface will have a range of unstable wave-numbers, as long as $m \neq 1$, even as $R \sim 0$. As an example, we will consider the case of two-dimensional channel flow under a pressure gradient (Poiseuille) as illustrated in Fig. 15. The problem is completely defined by the fluid-height ratio (n), density ratio (r), viscosity ratio (m), Reynolds number (R) and surface tension parameter (S), all defined in Fig. 15.

The characteristic velocity is that of the interface, U_i (it can be obtained from the imposed pressure gradient), and the characteristic length scale is taken as the depth of the upper fluid layer ($H = h_1$), so that the char-

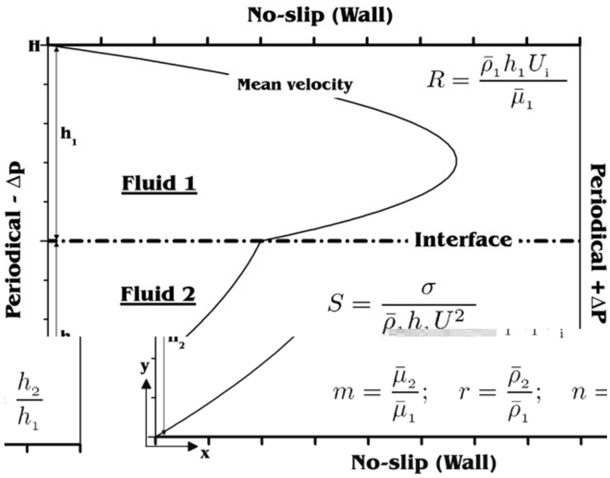
No-slip (Wall)
H
Mean velocity
$R = \frac{\bar{\rho}_1 h_1 U_i}{\bar{\mu}_1}$
h_1
Periodical - Δp
Fluid 1
Interface
Periodical +ΔP
Fluid 2
$S = \frac{\sigma}{\bar{\rho}_1 h_1 U^2}$
h_2
$\frac{h_2}{h_1}$
y
$m = \frac{\bar{\mu}_2}{\bar{\mu}_1}$; $r = \frac{\bar{\rho}_2}{\bar{\rho}_1}$; $n =$
x
No-slip (Wall)

Fig. 15. Flow configuration and definitions for viscous Kelvin–Helmholtz instability.

acteristic time is $\tau = \frac{H}{U_i}$. Displacements, wave-numbers and times in the numerical results below are scaled by these quantities accordingly, and we use the starred-quantities nomenclature as in Section 6.1.

Simulations were started from base-flow velocity profiles

$$U(\mathbf{x}) = \begin{cases} U_{\mathrm{i}}\left(1 + a_1\frac{y-h_1}{h_1} + b_1\left(\frac{y-h_1}{h_1}\right)^2\right) & \text{if } y > h_2 \\ U_{\mathrm{i}}\left(1 + \frac{a_1}{m}\frac{y-h_1}{h_1} + \frac{b_1}{m}\left(\frac{y-h_1}{h_1}\right)^2\right) & \text{otherwise} \end{cases} \tag{31}$$

where $a_1 = \frac{m-n^2}{n(n+1)}$ and $b_1 = -\frac{m+n}{n(n+1)}$. The pressure gradient is

$$\frac{\mathrm{d}P}{\mathrm{d}x} = -\frac{2U_{\mathrm{i}}\mu_1(m+n)}{h_1h_2(n+1)} \tag{32}$$

and in the simulation it is supported by imposing periodic boundary conditions at the channel inlet/exit, Fig. 15. Instabilities were seeded by imposing a small disturbance $\varepsilon^\star(0)$ on the interfacial shape – we will specify the initial amplitude in each case as needed.

An illustration of interfacial dynamics, starting from an exceedingly small initial amplitude of 2.5×10^{-4} with wave-number $\alpha^\star_{\mathrm{KH}} = 2.5133$, is given in Fig. 16. The symmetric early (linear regime) growth is easily discernable. So is also the remarkably early departure at amplitude of 1.4×10^{-3}, where the wave steepening is first observed on the frontal face of the wave-form (at $t^\star \sim 30$). Such remarkably early transitions in viscous shear flows have been noted also by Coward et al. [13]; as noted already, such early transitions make the window of opportunity for testing simulations against linear theory rather narrow – with attendant practical difficulties.

An illustration of the mechanics involved in energy transfer between wave and mean flow is given in Fig. 17. The energy is extracted from the mean flow of the lower fluid, and dissipated into the upper fluid; the difference is going to wave energy. This figure was obtained from linear stability theory (AROS code) and it also shows the stringent grid requirements for an accurate simulation, as the vicinity of the *critical layer* (where the mean flow velocity is equal to the phase speed of the wave) must be fully resolved.

Fidelity and convergence properties of our code are illustrated now by three sample simulations. Since viscosity jump is the central issue, we focus on density-matched fluids with no interfacial tension. One case is for an exceedingly low $R = 0.009$ ($m = 0.203$, and $n = 4.875$), for which it happens that there are high-quality experimental data for comparison (Khomami and Su [29]); the second is for $R = 0.47$ (with $\alpha^\star_{\mathrm{KH}} = 5.66$, a

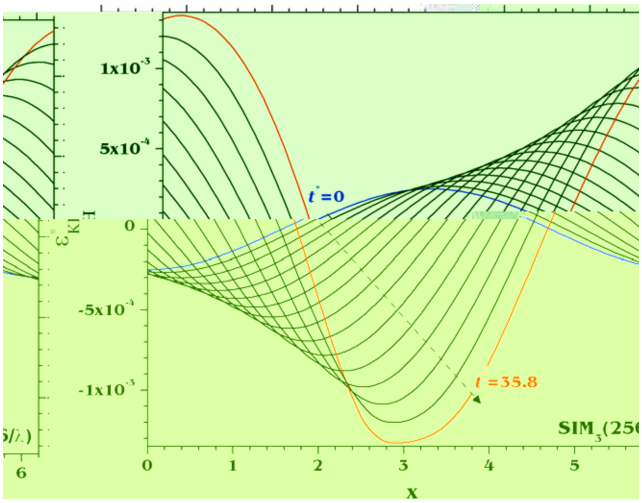

Fig. 16. Illustration of interfacial shape evolution and transition to non-linear behavior ($t^* \sim 30$). $R = 7.1$, $m = 0.5$, $n = 4$, $r = 1$, $\alpha^\star_{\mathrm{KH}} = 2.5133$, and $\varepsilon^\star_{\mathrm{KH}}(0) = 2.5 \times 10^{-4}$.

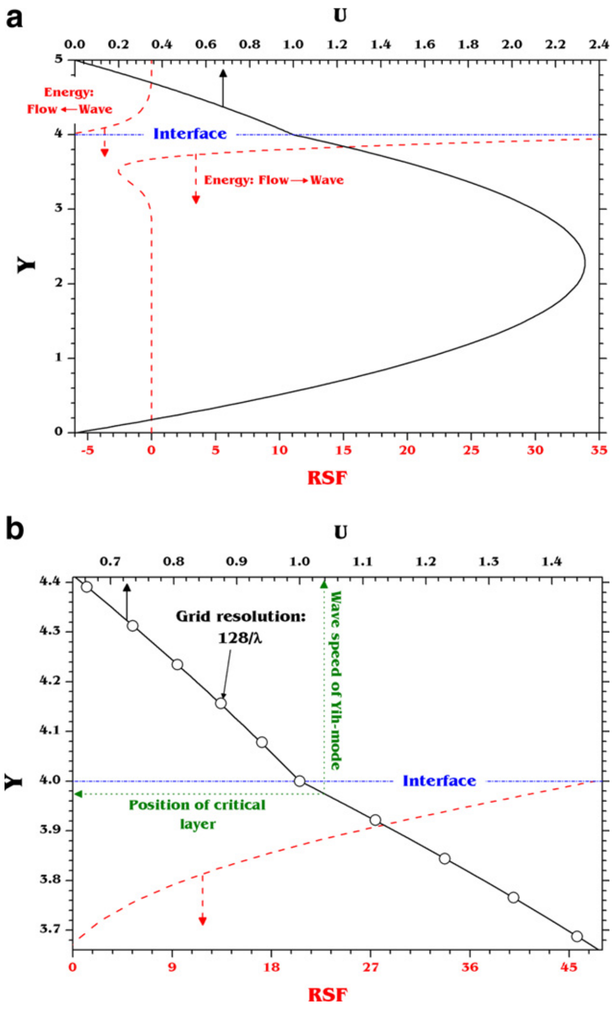

Fig. 17. (a) The base flow field and Reynolds Stress Function (RSF) from AROS computation of the case considered in Fig. 20. $R = 7.1$, $m = 0.5$, $r = 1$, $S = 0$, $n = 4$, $\alpha^{\star}_{\mathrm{KH}} = 2.5133$. The magnified view (b) shows the detail of the critical layer, and indicates a minimum grid necessary for resolution in DNS.

stable mode; and $\alpha^{\star}_{\mathrm{KH}} = 45.24$, an unstable mode; $m = 0.5$, and $n = 0.111$); and the third for a moderate value of $R = 7.1$ (with $\alpha^{\star}_{\mathrm{KH}} = 2.5133$, $m = 0.5$, and $n = 4$).

The results are summarized in Figs. 18–20. The latter includes several DIM simulations with various smearing-widths, mollifying functions (harmonic- and arithmetic-mean, Eqs. (C.1) and (C.2)) and resolutions. The convergence obtained in the MuSiC–SIM calculations for the case $R = 7.1$ is shown in Tables 2 and 3. We can see that in agreement with the suggestion of Fig. 17, the 256×256 grid provides the appropriate resolution of the critical layer at these conditions, resulting in the asymptotic grid convergence of SIM. The point of Fig. 21 is that an excellent agreement with experimental data testifies as to the intrinsic merit of such simulations [67].

In connection with Fig. 20, several salient points that need to be emphasized

(1) There is an acceleration time needed for the instability to enter the linear growth, and this is made available only by the simple seeding employed here in SIM. This physical mechanism removes arbitrariness in time origin, or initial magnitude of disturbance, in making predictions for real flows.
(2) The harmonic-mean approach seems to show less variability to the size of the mollifying zone in comparison to that found by arithmetic-mean.

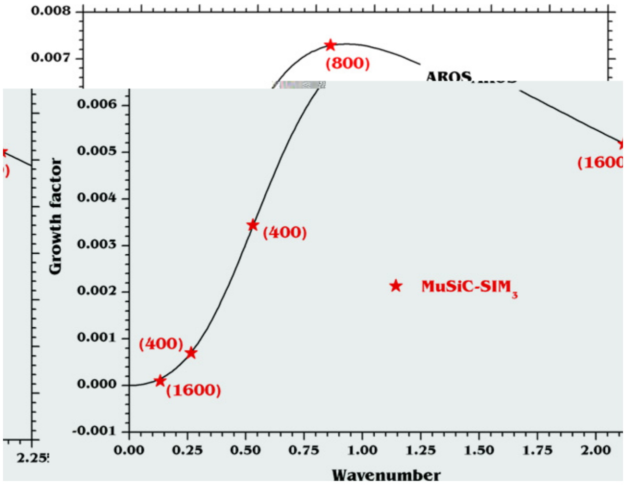

Fig. 18. Recovery of linear stability theory results (AROS) with MuSiC-SIM. $R = 0.009$, $m = 0.203$, $r = 1$, $S = 0$, $n = 4.875$, $\varepsilon^{\star}_{\mathrm{KH}}(0) = 1.175 \times 10^{-3}$. SIM's effective grid resolutions (# of nodes per λ) are shown in parentheses.

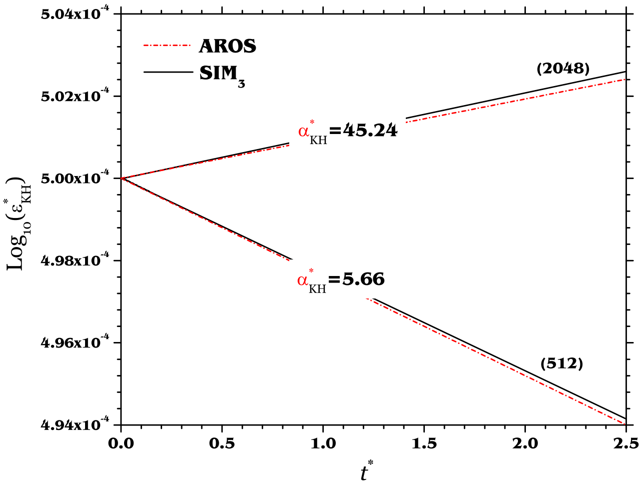

Fig. 19. Recovery of linear stability theory results (AROS) with MuSiC-SIM. $R = 0.47$, $m = 0.5$, $r = 1$, $S = 0$, $n = 0.111$, $\varepsilon^{\star}_{\mathrm{KH}}(0) = 5 \times 10^{-4}$. SIM's effective grid resolutions (# of nodes per λ) are shown in parentheses.

(3) At the limit of mollifying over two cells all results agree, but to a growth rate that is much greater than the correct one.
(4) Increasing the size of the mollifying zone results in less unstable behaviors, even to stability as the case $\delta = 8h$ shows (arithmetic-mean); this is consistent with results on real diffuse interfaces [68].

Finally in Fig. 22 we show the prediction of the neutral stability map[7] using MuSiC–SIM for the case of $R = 10$ and a relatively high viscosity ratio of $m = 10$, in comparison to the stability boundaries found by AROS. In all cases, the simulations were started from an amplitude of 0.00025 and the stable/unstable characterization was assigned according to the decay/growth of this initial disturbance, both of them occurring exponentially, after an initial acceleration period as noted above, in agreement with linear stability theory.

[7] To construct the map in Fig. 22, we used relatively coarse grids, which were apparently insufficient to accurately predict transition from stable to unstable interfaces in the very small sensitive region around $n \sim 4$ and wavenumber $\sim$1, requiring significant grid resolution in the critical layer.

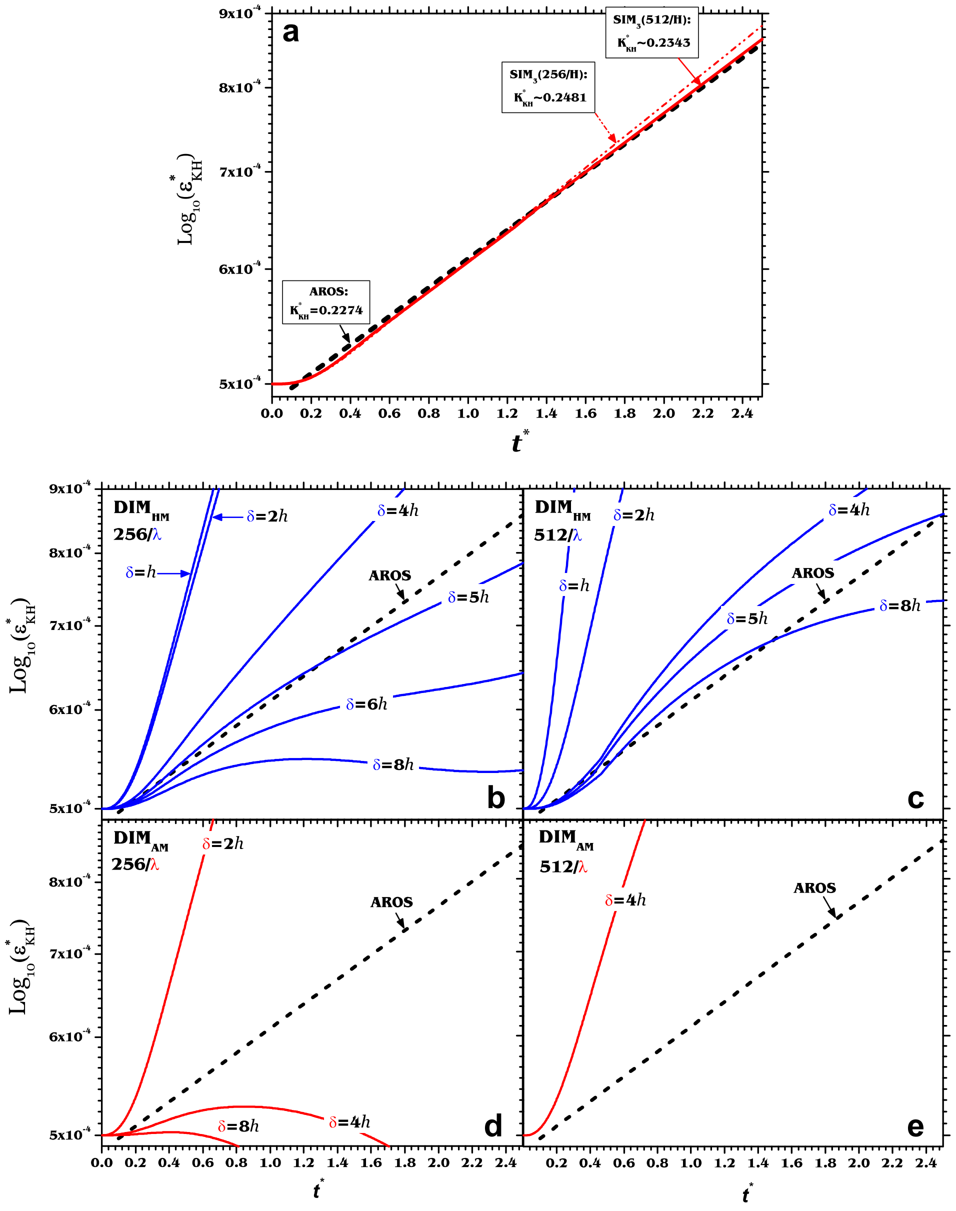

Fig. 20. Performance of MuSiC-SIM (a) and various implementations of DIM (b–e) against linear stability theory (AROS). Subscripts HM and AM are harmonic and arithmetic-mean interpolation at the interfacial cells. Growth factors $K^{\star}_{\mathrm{KH}}$ are measured as $\frac{\ln\left(\varepsilon^{\star}(t_1)-\varepsilon^{\star}(t_0)\right)}{t_1^{\star}-t_0^{\star}}$ at the "measurement" intervals $[t_0;t_1]$ chosen within the exponential growth regime. Scaled perturbations and time are defined as in [66]. $R=7.1$, $m=0.5$, $r=1$, $S=0$, $n=4$, $\alpha^{\star}_{\mathrm{KH}}=2.5133$, and $\varepsilon^{\star}_{\mathrm{KH}}(0)=5\times10^{-4}$.

Table 2
Grid convergence for growth factors $K^{\star}_{\mathrm{KH}}$ using SIM$_3$

Grid resolution	$K^{\star}_{\mathrm{KH}}$	$\mathcal{L}_1$-norm
$256/\lambda$	0.2481	0.0207
$512/\lambda$	0.2343	0.0069
Rate		*1.585*

$R=7.1$, $m=0.5$, $r=1$, $S=0$, $n=4$, $\alpha^{\star}_{\mathrm{KH}}=2.5133$, $\varepsilon^{\star}_{\mathrm{KH}}(0)=5\times10^{-4}$. Exact (Orr-Sommerfeld based, AROS) $K^{\star}_{\mathrm{KH}}$ is 0.2274.

Table 3
Grid convergence for wave speed C_r using SIM$_3$

Grid resolution	C_r	$\mathcal{L}_1$-norm
$256/\lambda$	1.03761	0.00019
$512/\lambda$	1.0377513	0.0000487
Rate		*1.96*

$R = 7.1$, $m = 0.5$, $r = 1$, $S = 0$, $n = 4$, $\alpha^{\star}_{\mathrm{KH}} = 2.5133$, $\varepsilon^{\star}_{\mathrm{KH}}(0) = 5 \times 10^{-4}$. Exact (Orr-Sommerfeld based, AROS) C_r is 1.0378.

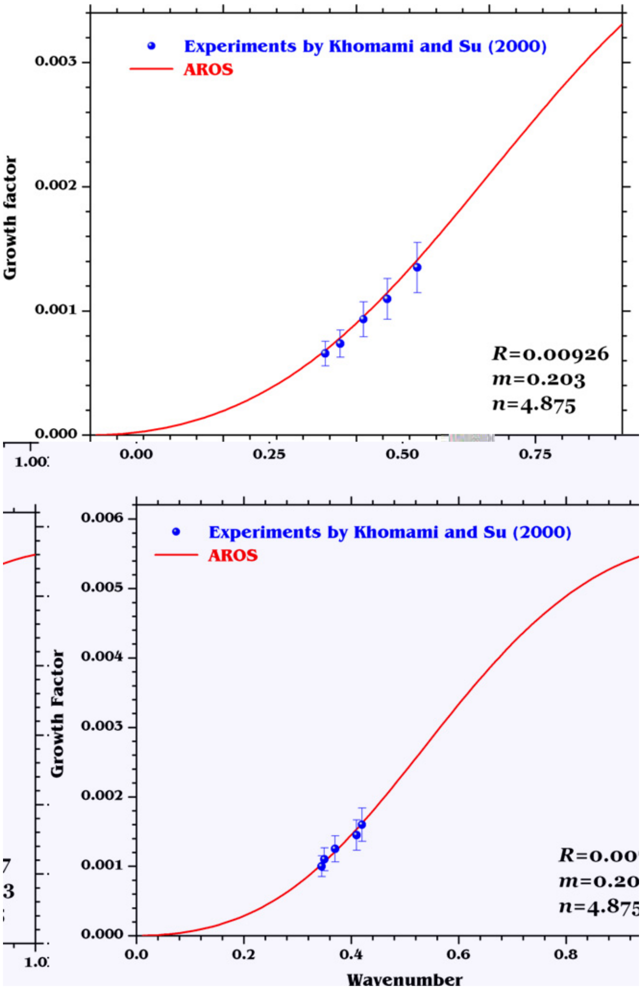

Fig. 21. A comparison of experiment [29] and linear stability theory (AROS [67]).

7. Conclusions

Results of sample numerical simulations (with both the SIM and DIM approaches) on test problems and real flows involving Rayleigh–Taylor (R–T) and viscous Kelvin–Helmholtz (vK–H) instabilities allow us to conclude the following:

1) The SIM$_3$ allows convergence to exact results with near-theoretical rates (third-order).
2) The SIM$_3$ allows prediction of growth factors in the linear regime for both R–T and vK–H flows, of neutral-stability maps for the Yih instability in vK–H flows, and self-selects the most unstable wave in multimode R–T flows, all without *a priori* knowledge or use of the eigenmodes.

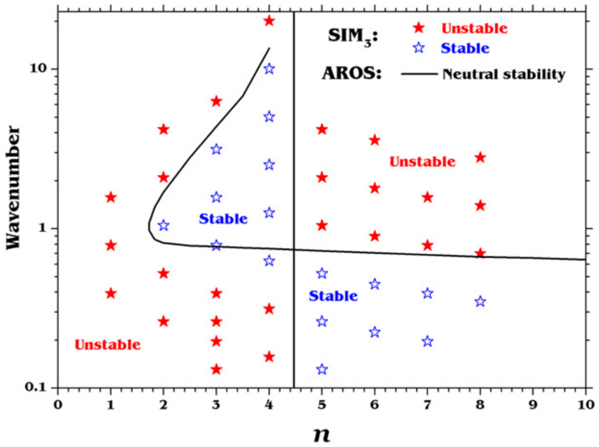

Fig. 22. Neutral-stability map for the Yih-mode. $R = 10$, $m = 20$, $r = 1$, $S = 0$. Comparison of SIM with linear theory (AROS [59,66]).

3) SIM$_3$ provides yet another way to eliminate the spurious currents that arise in DIM.
4) The DIM performs well when instability is controlled by normal stresses as in R–T flows; but when shear stresses dominate, performance appears to be problematic, and worthy of further investigation. Our limited tests suggest that depending on the type and degree of mollifying employed, results can vary from stable to unstable behaviors with widely-varying growth factors.

Future development of the SIM will be focused on two challenging issues: (a) extension of the cut-cell grid generation to three dimensions and (b) high-order implicit treatment of acoustic waves and diffusion operator, which will be implemented along the lines of the third-order implicit Runge–Kutta [8] and Jacobian-Free Newton Krylov [28,30,50] methods, with physics-based (all-speed ICE [22,23,43]) preconditioning. On the other hand, it would appear that the sharp fluid dynamics treatment of our method is compatible and possibly advantageous to any interfacial flow algorithm in which the interface is represented as a discrete Heaviside function.

Acknowledgments

This work was sponsored by the Joint Science and Technology Center of the Defense Threat Reduction Agency (JSTO/DTRA), Lawrence Livermore National Laboratory (the HOPS program), and the National Ground Intelligence Center (NGIC). The support and collaboration of Drs. C. Fromer, J. Hanna, F. Handler, R. Babarsky and G. Nakafuji is gratefully acknowledged. We are also grateful to Drs. Doug Kothe (ORNL) and Marianne Francois (LANL) for reading the near-final manuscript and for their many insightful comments and suggestions.

Appendix A. Positioning algorithm for sub-cell markers

Subcell markers $_{(\mathrm{sm})}$ are placed at intersections of the level set's zero-contour with the gridlines. The positions, $\theta = \frac{\mathbf{x}_{\mathrm{sm}} - \mathbf{x}_{(i,j)}}{h}$, of subcell markers are computed in a coordinate-by-coordinate fashion, using a piecewise-cubic (fourth-order) interpolation procedure. For example, for markers found between cells $[_{(\mathrm{i,j})}$ and $_{(\mathrm{i+1,j})}]$ (if $\varphi_{(i,j)} \cdot \varphi_{(i+1,j)} \leqslant 0$) or $[_{(\mathrm{i,j})}$ and $_{(\mathrm{i,j+1})}]$ (if $\varphi_{(i,j)} \cdot \varphi_{(i,j+1)} \leqslant 0$), we have (shown for the $[_{(\mathrm{i,j})}$ and $_{(\mathrm{i+1,j})}]$-case only)

$$\varphi_{(i,j)} - \frac{3\varphi_{(i,j)} + 2\varphi_{(i-1,j)} - 6\varphi_{(i+1,j)} + \varphi_{(i+2,j)}}{6}\theta + \frac{\varphi_{(i-1,j)} - 2\varphi_{(i,j)} + \varphi_{(i+1,j)}}{2}\theta^2 + \frac{3(\varphi_{(i,j)} - \varphi_{(i+1,j)}) - \varphi_{(i-1,\mathrm{j})} + \varphi_{(i+2,j)}}{6}\theta^3 = 0 \quad (A.1)$$

which is solved for θ iteratively by the Newton–Raphson method.

Appendix B. Least-squares interpolation

The following algorithm is used for interpolation of a scalar field on an irregular data set of N points. To the third-order, the two-dimensional Taylor expansion is

$$\Psi(x,y) = \Psi_0 + x\Psi_x + y\Psi_y + \frac{x^2}{2}\Psi_{xx} + xy\Psi_{xy} + \frac{y^2}{2}\Psi_{yy} + \mathrm{O}(h^3) \tag{B.1}$$

where the center of coordinate system is chosen to be at the interpolated point. Introducing the vector of unknowns as $\mathbf{X} = \left(\Psi_0, \Psi_x, \Psi_y, \frac{\Psi_{xx}}{2}, \Psi_{xy}, \frac{\Psi_{yy}}{2}\right)^{\mathrm{T}}$, interpolation is completely defined with $N = 6$ (distinct points) by

$$\mathbf{X} = \mathbb{A}^{-1} \cdot \mathbf{B} \tag{B.2}$$

where

$$\mathbf{B} = \begin{bmatrix} \Psi_1 \\ \Psi_2 \\ \Psi_3 \\ \Psi_4 \\ \Psi_5 \\ \Psi_6 \end{bmatrix} \quad \text{and} \quad \mathbb{A} = \begin{bmatrix} 1 & x_1 & y_1 & x_1^2 & x_1y_1 & y_1^2 \\ 1 & x_2 & y_2 & x_2^2 & x_2y_2 & y_2^2 \\ 1 & x_3 & y_3 & x_3^2 & x_3y_3 & y_3^2 \\ 1 & x_4 & y_4 & x_4^2 & x_4y_4 & y_4^2 \\ 1 & x_5 & y_5 & x_5^2 & x_5y_5 & y_5^2 \\ 1 & x_6 & y_6 & x_6^2 & x_6y_6 & y_6^2 \end{bmatrix} \tag{B.3}$$

To avoid singularities in matrix $\mathbb{A}$ (typical for irregular-grid), the dataset for interpolation is overspecified by selecting $N > 6$. In this case, the interpolation is cast into the minimization problem

$$\begin{gathered} \mathrm{MIN}\left\{ \mathcal{E} \equiv \sum_{i=1}^{N>6} \omega_i \left(\Psi_i^{(\mathrm{LSq}_3)} - \Psi_i \right)^2 \right\} \\ \frac{\partial\mathcal{E}}{\partial\Psi_0} = \frac{\partial\mathcal{E}}{\partial\Psi_x} = \frac{\partial\mathcal{E}}{\partial\Psi_y} = \frac{\partial\mathcal{E}}{\partial\Psi_{xx}} = \frac{\partial\mathcal{E}}{\partial\Psi_{xy}} = \frac{\partial\mathcal{E}}{\partial\Psi_{yy}} = 0 \end{gathered} \tag{B.4}$$

resulting in the following vector and matrix of the LSq_3 interpolation

$$\widetilde{\mathbf{B}} = \left[\sum_{i=1}^N \omega_i\Psi_i,\ \sum_{i=1}^N \omega_i x_i \Psi_i,\ \sum_{i=1}^N \omega_i y_i \Psi_i,\ \sum_{i=1}^N \omega_i x_i^2 \Psi_i,\ \sum_{i=1}^N \omega_i x_i y_i \Psi_i,\ \sum_{i=1}^N \omega_i y_i^2 \Psi_i \right]^{\mathrm{T}} \tag{B.5}$$

$$\widetilde{\mathbb{A}} = \begin{bmatrix} \sum_{i=1}^N \omega_i & \sum_{i=1}^N \omega_i x_i & \sum_{i=1}^N \omega_i y_i & \sum_{i=1}^N \omega_i x_i^2 & \sum_{i=1}^N \omega_i x_i y_i & \sum_{i=1}^N \omega_i y_i^2 \\ \sum_{i=1}^N \omega_i x_i & \sum_{i=1}^N \omega_i x_i^2 & \sum_{i=1}^N \omega_i x_i y_i & \sum_{i=1}^N \omega_i x_i^3 & \sum_{i=1}^N \omega_i x_i^2 y_i & \sum_{i=1}^N \omega_i x_i y_i^2 \\ \sum_{i=1}^N \omega_i y_i & \sum_{i=1}^N \omega_i x_i y_i & \sum_{i=1}^N \omega_i y_i^2 & \sum_{i=1}^N \omega_i x_i^2 y_i & \sum_{i=1}^N \omega_i x_i y_i^2 & \sum_{i=1}^N \omega_i y_i^3 \\ \sum_{i=1}^N \omega_i x_i^2 & \sum_{i=1}^N \omega_i x_i^3 & \sum_{i=1}^N \omega_i x_i^2 y_i & \sum_{i=1}^N \omega_i x_i^4 & \sum_{i=1}^N \omega_i x_i^3 y_i & \sum_{i=1}^N \omega_i x_i^2 y_i^2 \\ \sum_{i=1}^N \omega_i x_i y_i & \sum_{i=1}^N \omega_i x_i^2 y_i & \sum_{i=1}^N \omega_i x_i y_i^2 & \sum_{i=1}^N \omega_i x_i^3 y_i & \sum_{i=1}^N \omega_i x_i^2 y_i^2 & \sum_{i=1}^N \omega_i x_i y_i^3 \\ \sum_{i=1}^N \omega_i y_i^2 & \sum_{i=1}^N \omega_i x_i y_i^2 & \sum_{i=1}^N \omega_i y_i^3 & \sum_{i=1}^N \omega_i x_i^2 y_i^2 & \sum_{i=1}^N \omega_i x_i y_i^3 & \sum_{i=1}^N \omega_i y_i^4 \end{bmatrix} \tag{B.6}$$

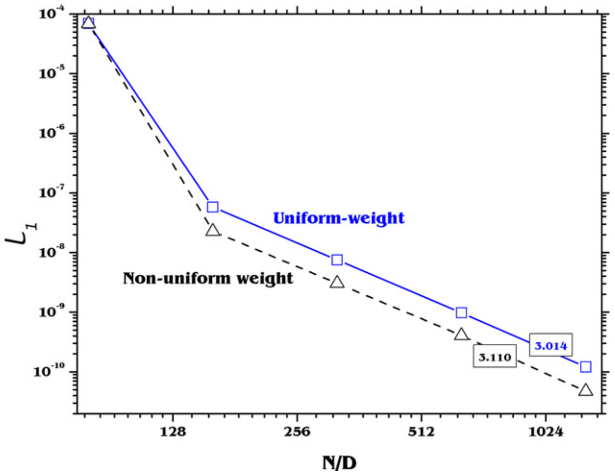

Fig. 23. Convergence of the third-order-accurate least-squares interpolation for a smooth scalar field ($\mathcal{L}_1$-norm of errors).

For efficiency, the analytical form of the inverse of this matrix is computed and stored for multiple uses. The weight functions are defined as[8]

$$\tilde{\omega}_i\left(\frac{d_i}{d_{\max}}\right)=\frac{4}{9}+\frac{d_i/d_{\max}}{4\left(1-\sqrt{2}\right)}+\frac{8-5\sqrt{2}}{12\left(\sqrt{2}-1\right)}\frac{d_i^2}{d_{\max}^2};\quad \omega_i=\frac{\tilde{\omega}_i}{\sum_{i=1}^{N}\tilde{\omega}_i} \tag{B.7}$$

where d_i and $d_{\max}=\mathrm{MAX}\{d_i\}_{i=1,\ldots,N}$ are the distances-to-interpolation-points and the "radius" of the irregular datasets, respectively. Using non-uniform weights $\left(\omega_i\neq\frac{1}{N}\right)$ not only increases the accuracy of the interpolation,[9] but also ensures the robustness of the finite-volume-based integration on irregular cells. The solution to Eq. (B.2) provides directly the derivatives needed to compute the viscous stresses.

For smooth fields, this least-squares interpolation is third-order accurate in primitive variables, and second-order accurate in stresses. For example let us take

$$\varphi(\mathbf{x})=\frac{D}{2}-\sqrt{\left(x-\frac{1}{2}\right)^2+\left(y-\frac{1}{2}\right)^2}$$

in the domain $[1\times 1]$ with $D=0.8$. Subcell markers are generated as described in Appendix A. The Level Set is computed at these markers using the LSq$_3$, by sampling data from gravity centers of the nearest 16 irregular (C_2-grid) cells. As seen in Fig. 23, both uniform-weights $\left(\omega_i=\frac{1}{N}\right)$ and those from Eq. (B.7) exhibit near-third-order convergence in the grid-asymptotic range, however, the uniform-weight approach lags slightly in accuracy. More importantly, we found that the use of uniform weights in finite-volume integration can cause stability problems.

Appendix C. The DIM as used for the sample comparisons

Various implementations of DIM can be found in the literature along with the various choices available for representing the interface, namely, the level set method (LS) [44,54], volume tracking (VT) [49], or front tracking (FT) [70].

[8] Here, these weights are prescribed heuristically. Note that on regular datasets, they are consistent with the weights of the quadratures used for momentum-space integration in the LBGK models [37].

[9] In principle, one should be able to compute "optimal" weights $\omega_i^{(\mathrm{opt})}$ corresponding to the maximum possible accuracy $\sim \mathrm{O}(h^{N/2})$ on the chosen N-point irregular dataset.

Performance of a particular DIM is defined by one's choice of mollifying function (the prescription of smearing that allows gradual transition from one fluid's properties to the other's) and of the body force term in the momentum equation used to model the interfacial tension. Following common use [60,61] for the former, we make use of the level set function (φ), which allows an explicit and convenient "mollification" as follows[10]:

$$\underline{\text{Arithmetic-Mean, AM}}: \quad \psi(\varphi) = \psi^- + (\psi^+ - \psi^-)\mathfrak{H}_\delta(\varphi), \quad \psi = \rho, \mu \tag{C.1}$$

$$\underline{\text{Harmonic-Mean, HM}}: \quad \psi(\varphi) = \frac{1}{\frac{\mathfrak{H}_\delta(\varphi)}{\psi^+} + \frac{1-\mathfrak{H}_\delta(\varphi)}{\psi^-}} \tag{C.2}$$

where the superscripts "+/−" denote the bulk properties of the two fluids involved, and the approximation to the Heaviside function is defined as

$$\mathfrak{H}_\delta(\varphi) = \begin{cases} 0 & \text{if } \varphi < -\delta \\ \frac{1}{2}\left[1 + \frac{\varphi}{\delta} + \frac{\sin\frac{\pi\varphi}{\delta}}{\pi}\right] & \text{if } |\varphi| \leqslant \delta \\ 1 & \text{otherwise} \end{cases} \tag{C.3}$$

For surface tension, the body force $\mathbf{F}_\sigma(\mathbf{x})$ in the mollifying corridor [11,70] is defined by

$$\mathbf{F}_\sigma(\varphi(\mathbf{x})) = \mathfrak{D}_\delta \sigma \kappa \mathbf{n} \tag{C.4}$$

where

$$\mathfrak{D}_\delta(\varphi(\mathbf{x})) = \begin{cases} \frac{1+\cos\frac{\pi\varphi}{\delta}}{2\delta} & \text{if } |\varphi| \leqslant \delta \\ 0 & \text{otherwise} \end{cases}$$

where σ, κ and $\mathbf{n}$ are the coefficient of surface tension, interfacial curvature and unit normal-to-the-interface vector, respectively.

References

[1] R. Abgrall, Generalization of Roe scheme for the computation of mixture of perfect gases, Recherche Aerospatiale 6 (1988) 31.

[2] R. Abgrall, S. Karni, Computations of compressible multifluids, Journal of Computational Physics 169 (2001) 594–623.

[3] S.B. Baden, N.P. Chrisochides, D.B. Gannon, Structured Adaptive Mesh Refinement (SAMR) Grid Methods, Springer, Berlin, Heidelberg, New York, 1999.

[4] W. Bangerth, R. Rannacher, Adaptive Finite Element Methods for Solving Differential Equations, Birkhäuser, 2003.

[5] M.J. Berger, P. Colella, Local adaptive mesh refinement for shock hydrodynamics, Journal of Computational Physics 82 (1989) 67–84.

[6] M.J. Berger, J. Oliger, Adaptive mesh refinement for hyperbolic partial differential equations, Journal of Computational Physics 53 (1984) 482–512.

[7] M.J. Berger, I. Rigoutsos, An algorithm for point clustering and grid generation, IEEE Transactions on Systems, Man, and Cybernetics 21 (5) (1991) 1278–1286.

[8] H. Bijl, M.H. Carpenter, V.N. Vatsa, C.A. Kennedy, Implicit time integration schemes for the unsteady compressible Navier–Stokes equations: laminar flow, Journal of Computational Physics 179 (2002) 313–329.

[9] T. Boeck, S. Zaleski, Viscous versus inviscid instability of two-phase mixing layers with continuous velocity profile, Physics of Fluids 17 (3) (2005) 032106.

[10] T. Boeck, J. Li, E. López-Pagés, P. Yecko, S. Zaleski, Ligament formation in sheared liquid–gas layers, Theoretical Computational Fluid Dynamics 21 (2007) 59–76.

[11] J.U. Brackbill, D.B. Kothe, C. Zemach, A continuum method for modeling surface tension, Journal of Computational Physics 100 (1992) 335–354.

[12] S. Chandrasekhar, Hydrodynamic and Hydromagnetic Stability, Dover Publications, Inc., New York, 1981.

[10] In difference to *explicit* smearing Eqs. (C.1) and (C.2) in the LS formulation, VT and FT methods require additional procedures to construct *support* for mollification. For example, FT utilizes a fast Poisson solver for elliptic equation $\nabla^2\mathcal{I} = \nabla \cdot \mathcal{G}$, in order to find the *indicator function* $\mathcal{I}$, which is a counterpart of the Heaviside function $\mathfrak{H}_\delta(\varphi)$ used here. The gradient of the indicator function $\mathcal{G}(\mathcal{D})$ depends on the choice of the kernel $\mathcal{D}$, which determines what fraction of the interface quantity should contribute to each node in the *mollifying* corridor. More recent tendency is to combine VT and FT with the LS-related techniques [55,62,63], which allows to benefit from an ability to both explicitly *mollify* the interface and easily compute interfacial curvature in three dimensions.

[13] A.V. Coward, Y.Y. Renardy, M. Renardy, J.R. Richards, Temporal evolution of periodic disturbances in two-layer Couette flow, Journal of Computational Physics 132 (1997) 346–361.

[14] D. Drikakis, W. Rider, High-Resolution Methods for Incompressible and Low-Speed Flows, Springer, Berlin, Heidelberg, New York, 2005.

[15] R.P. Fedkiw, T. Aslam, B. Merriman, S. Osher, A non-oscillatory Eulerian approach to interfaces in multimaterial flows (the Ghost fluid method), Journal of Computational Physics 152 (1999) 457–492.

[16] M.M. Francois, S.J. Cummins, E.D. Dendy, D.B. Kothe, J.M. Sicilian, M.W. Williams, A balanced-force algorithm for continuous and sharp interfacial surface tension models within a volume tracking framework, Journal of Computational Physics 213 (2006) 141–173.

[17] F. Gibou, R. Fedkiw, A fourth order accurate discretization for the Laplace and heat equations on arbitrary domains, with applications to the Stefan problem, Journal of Computational Physics 202 (2005) 577–601.

[18] J. Glimm, J.W. Grove, X.L. Li, W. Oh, D.H. Sharp, A critical analysis of Rayleigh–Taylor growth rates, Journal of Computational Physics 169 (2001) 652–677.

[19] J. Glimm, X.L. Li, Y.J. Liu, Z.L. Xu, N. Zhao, Conservative front tracking with improved accuracy, SIAM Journal on Numerical Analysis 41 (5) (2003) 1926–1947.

[20] G. Hansen, R. Douglass, A. Zardecki, Mesh Enhancement: Selected Elliptic Methods, Foundations and Applications, Imperial College Press, London, 2005.

[21] G. Hansen, A. Zardecki, D. Greening, R. Bos, A finite element method for three-dimensional unstructured grid smoothing, Journal of Computational Physics 202 (2005) 281–297.

[22] F.H. Harlow, A.A. Amsden, Numerical calculation of almost incompressible flow, Journal of Computational Physics 3 (1968) 80–93.

[23] F.H. Harlow, A.A. Amsden, A numerical fluid dynamics calculation method for all flow speeds, Journal of Computational Physics 8 (1971) 197–213.

[24] X. He, S. Chen, R. Zhang, A lattice Boltzmann scheme for incompressible multiphase flow and its application in simulation of Rayleigh–Taylor instability, Journal of Computational Physics 152 (1999) 642–663.

[25] C.W. Hirt, An arbitrary Lagrangian–Eulerian computing technique, in: Proceedings of the International Conference of Numerical Methods for Fluid Dynamics, 2nd. Berkeley, California, September 15–19, 1970.

[26] R.D. Hornung, S.R. Kohn, Managing application complexity in the SAMRAI object-oriented framework, in: Concurrency and Computation: Practice and Experience (Special Issue), 14, 2002, pp. 347–368.

[27] M. Kang, R.P. Fedkiw, X.-D. Liu, A boundary condition capturing method for multiphase incompressible flow, Journal of Scientific Computing 15 (2) (2000) 323–360.

[28] C.T. Kelley, Solving nonlinear equations with Newton's method, Fundamentals of Algorithms, SIAM, 2003.

[29] B. Khomami, K.C. Su, An experimental/theoretical investigation of interfacial instabilities in superposed pressure-driven channel flow of Newtonian and well characterized viscoelastic fluids: part I: linear stability and encapsulation effects, Journal of Non-Newtonian Fluid Mechanics 91 (2000) 59–84.

[30] D.A. Knoll, D. Keyes, Jacobian-free Newton–Krylov methods: a survey of approaches and applications, Journal of Computational Physics 193 (2003) 357–397.

[31] P.M. Knupp, K. Salari, S.G. Krantz, Verification of Computer Codes in Computational Science and Engineering, CRC Press, 2003.

[32] Ya.I. Kucherenko, L.I. Shibarshev, V.I. Chitaikin, S.A. Balabin, A.P. Palaev, Experimental investigation into the self-similar mode of the gravitational turbulent mixing, in: Proceedings of the 3rd International Workshop on the Physics of Compressible Turbulent Mixing, 1991.

[33] B. Lafaurie, C. Nardone, R. Scardovelli, S. Zaleski, G. Zanetti, Modelling merging and fragmentation in multiphase flows with SURFER, Journal of Computational Physics 113 (1994) 134–147.

[34] R.J. LeVeque, Z. Li, An immersed interface method for elliptic equations with discontinuous coefficients and singular sources, SIAM Journal of Numerical Analysis 31 (4) (1994) 1019–1044.

[35] L. Lee, R.J. LeVeque, An immersed interface method for incompressible Navier–Stokes equations, SIAM Journal on Scientific Computing 25 (3) (2003) 832–856.

[36] J. Li, Y.Y. Renardy, M. Renardy, A numerical study of periodic disturbances on two-layer Couette flow, Physics of Fluids 10 (1998) 12.

[37] R.R. Nourgaliev, T.N. Dinh, T.G. Theofanous, D. Joseph, The Lattice Boltzmann equation method: theoretical interpretation, numerics and implications, International Journal of Multiphase Flow 29 (1) (2003) 117–169.

[38] R.R. Nourgaliev, T.N. Dinh, T.G. Theofanous, A pseudo-compressibility method for the numerical simulation of incompressible multifluid flows, International Journal of Multiphase Flow 30 (7–8) (2004) 901–937.

[39] R.R. Nourgaliev, T.N. Dinh, T.G. Theofanous, Sharp treatment of surface tension and viscous stresses in multifluid dynamics, AIAA 2005-5349, in: 17th AIAA Computational Fluid Dynamics Conference, Toronto, Canada, June 6–9, 2005.

[40] R.R. Nourgaliev, T.N. Dinh, T.G. Theofanous, Adaptive characteristics-based matching for compressible multifluid dynamics, Journal of Computational Physics 213 (2) (2006) 220–252.

[41] R.R. Nourgaliev, T.G. Theofanous, High-fidelity interface tracking in compressible flows: unlimited anchored adaptive level set, Journal of Computational Physics 224 (2) (2007) 836–866.

[42] R.R. Nourgaliev, M.-S. Liou, T.G. Theofanous, Numerical prediction of interfacial instability, in: Proceedings of the 4th International Conference on Computational Fluid Dynamics, ICCFD'4, Ghent, Belgium, July 10–14, 2006.

[43] R.R. Nourgaliev, D. Knoll, V. Mousseau, R. Berry, Direct numerical simulation of boiling multiphase flows: state-of-the-art, modeling, algorithmic and computer needs, in: Joint International Topical Meeting on Mathematics & Computation and

Supercomputing in Nuclear Applications (M&C + SNA 2007), Monterey, California, April 15–19, 2007, on CD-ROM, American Nuclear Society, LaGrange Park, IL, 2007.

[44] S. Osher, R. Fedkiw, Level set methods and dynamic implicit surfaces, Applied Mathematical Sciences, vol. 153, Springer-Verlag, New York, Berlin, Heidelberg, 2003.

[45] C. Peskin, Flow patterns around heart valves: a digital computer method for solving the equations of motion, Ph.D. Thesis, Physiol., Albert Einstein Coll. Med., Univ. Microfilms 378 (1972) 72–30.

[46] C. Peskin, Numerical analysis of blood flow in the heart, Journal of Computational Physics 25 (1977) 220–252.

[47] K.I. Read, Experimental investigation of turbulent mixing by Rayleigh–Taylor instability, Physica D 12 (1984) 45.

[48] Y. Renardy, M. Renardy, PROST: a parabolic reconstruction of surface tension for the volume-of-fluid method, Journal of Computational Physics 183 (2003) 400–421.

[49] W.J. Rider, D.B. Kothe, Reconstructing volume tracking, Journal of Computational Physics 141 (1998) 112–152.

[50] Y. Saad, Iterative Methods for Sparse Linear Systems, second ed., SIAM, 2003.

[51] SAMRAI: Structured Adaptive Mesh Refinement Application Infrastructure, <http://www.llnl.gov/CASC/SAMRAI>.

[52] P. Solin, K. Segeth, I. Dolezel, Higher-Order Finite Element Methods, Studies in Advanced Mathematics, Chapman & Hall/CRC, 2004.

[53] R. Scardovelli, S. Zaleski, Direct numerical simulations of free-surface and interfacial flow, Annual Review of Fluid Mechanics 31 (1999) 567–603.

[54] J.A. Sethian, Level Set Methods and Fast Marching Methods, Cambridge University Press, 1999.

[55] S. Shin, D. Juric, Modeling three-dimensional multiphase flow using a level contour reconstruction method for front tracking without connectivity, Journal of Computational Physics 180 (2002) 427–470.

[56] C.-W. Shu, S. Osher, Efficient implementation of essentially non-oscillatory shock-capturing schemes, Journal of Computational Physics 77 (1989) 439–471.

[57] W. Shyy, Multiphase computations using sharp and continuous interface techniques for micro-gravity applications, Comptes Rendus Mechanique 332 (2004) 375–386.

[58] D.M. Snider, M.J. Andrews, Rayleigh–Taylor and shear driven mixing with an unstable thermal stratification, Physics of Fluids 6 (10) (1994) 3324.

[59] S.Yu. Sushchikh, R.R. Nourgaliev, T.G. Theofanous, Linear instability of diffuse interfaces under shear, International Journal of Multiphase Flow, 2007, under review.

[60] M. Sussman, P. Smereka, S. Osher, A level set method for computing solutions to incompressible two-phase flow, Journal of Computational Physics 119 (1994) 146.

[61] M. Sussman, A.S. Almgren, J.B. Bell, P. Colella, L.H. Howell, M.L. Welcome, An adaptive level set approach for incompressible two-phase flows, Journal of Computational Physics 148 (1999) 81–124.

[62] M. Sussman, E.G. Puckett, A coupled level set and volume-of fluid method for computing 3D and axisymmetric incompressible two-phase flows, Journal of Computational Physics 162 (2000) 301–337.

[63] M. Sussman, K.M. Smith, M.Y. Hussaini, M. Ohta, R. Zhi-Wei, Sharp interface method for incompressible two-phase flows, Journal of Computational Physics 221 (2007) 469–505.

[64] T.G. Theofanous, G.J. Li, On the physics of aero-breakup, Physics of Fluids, in press.

[65] T.G. Theofanous, G.J. Li, T.N. Dinh, C.-H. Chang, Aerobreakup in disturbed subsonic and supersonic flow fields, Journal of Fluid Mechanics 593 (2007) 131–170.

[66] T.G. Theofanous, R.R. Nourgaliev, S. Wiri, Short Communication: Direct numerical simulations of two-layer viscosity-stratified flow by Qing Cao, Kausik Sarkar, Ajay K. Prasad, International Journal of Multiphase Flow 33 (7) (2007) 789–796.

[67] T.G. Theofanous, R.R. Nourgaliev, B. Khomami, Short Communication: An experimental/theoretical investigation of interfacial instabilities in superposed pressure-driven channel flow of newtonian and well characterized viscoelastic fluids: part I: linear stability and encapsulation effects, by Bamin Khomami and Kuan C. Su, Journal of Non-Newtonian Fluid Mechanics 143 (2007) 131–132.

[68] T.G. Theofanous, S. Sushchikh, R.R. Nourgaliev, Linear stability of sharp and diffuse interfaces under shear, in: FEDSM07 5th Joint ASME/JSME Fluids Engineering Summer Conference, San Diego, California, USA, July 30–August 2, 2007.

[69] A. Wissink, D. Hysom, R. Hornung, Enhancing scalability of parallel structured amr calculations, in: Proceedings of the 17th ACM International Conference on Supercomputing (ICS03), San Francisco, CA, June 23–26, 2003, pp. 336–347.

[70] S.O. Unverdi, G. Tryggvason, A front-tracking method for viscous, incompressible, multi-fluid flows, Journal of Computational Physics 100 (1992) 25.

[71] T. Ye, W. Shyy, J.N. Chung, A fixed-grid sharp-interface method for bubble dynamics and phase change, Journal of Computational Physics 174 (2001) 781–815.

[72] C.-S. Yih, Instability due to viscosity stratification, Journal of Fluid Mechanics 27 (1967) 337–352.

[73] D.L. Youngs, Numerical simulation of turbulent mixing by Rayleigh–Taylor instability, Physica D 12 (1984) 32–44.

[74] D.L. Youngs, Three-dimensional numerical simulation of turbulent mixing by Rayleigh–Taylor instability, Physics of Fluids A 3 (1991) 1312.