Journal of Computational Physics **163,** 172–215 (2000)
doi:10.1006/jcph.2000.6566, available online at http://www.idealibrary.com on IDEAL®

A One-Cell Local Multigrid Method for Solving Unsteady Incompressible Multiphase Flows

Stéphane Vincent and Jean-Paul Caltagirone

Modélisation Avancée des Systèmes Thermiques et Écoulements Réels (MASTER), MASTER-ENSCPB, Université Bordeaux I, Avenue Pey-Berland, B.P. 108, 33402 Talence Cedex, France
E-mail: vincent@lmaster.u-bordeaux.fr, calta@lmaster.u-bordeaux.fr

Received January 12, 1999; revised May 12, 2000

An original local multigrid method for solving incompressible two-phase flow with surface tension is described. The dynamics of the interface are resolved on a hierarchy of structured and uniform grids (orthogonal Cartesian meshes). A new type of composite boundary condition is proposed to solve the dynamics of the multigrid calculation domains. The interface tracking is described by a TVD VOF algorithm and the equations of motion are solved using an augmented Lagrangian method. The surface tension is calculated using a continuous surface force method. The one-cell local multigrid method is compared to relevant analytical scalar advection tests. Several classical two-phase flow problems, including nonlinear drop oscillations, Rayleigh–Taylor instabilities, and the drop impact on liquid film, have also been considered. The local character of the method and the differences between a single-grid and a multigrid solution are discussed. For unsteady problems, such as the Rayleigh–Taylor instability, the memory costs and the computational time have been reduced by up to 50%. © 2000 Academic Press

Key Words: local multigrid method; numerical simulation; multiphase flow; interfaces; TVD schemes; augmented Lagrangian; Rayleigh–Taylor instability.

1. INTRODUCTION

Experimental and theoretical studies of incompressible multiphase viscous flows, involving three-dimensional free surface instabilities or very strong interface tearing and stretching, are difficult to perform. However, free surface flows clearly need to be better understood to develop improved models for industrial applications. Numerical methods have begun to be used to simulate the flow dynamics of the problem. Numerous research projects are now under way to improve both the modelling and the understanding of free surface flows.

Two approaches have generally been used to perform direct numerical simulations of two-phase flows. In the Lagrangian approach, the grid follows the motion of the interface. An

0021-9991/00 $35.00
Copyright © 2000 by Academic Press
All rights of reproduction in any form reserved.

adaptative grid is first built to fit the interface between the two phases. The conservation equations are then solved for the "discretized fluid." Finally a remodelling of the grid is carried out with the grid being moved and deformed to take into account the Lagrangian evolution of the free surface. This technique, which uses finite element or curvilinear finite volume discretizations, has been adapted to thin process studies, occurring near slightly deforming interfaces (Magnaudet *et al.* [27]). However, in three-dimensional unsteady flows with strong interface deformations, the Lagrangian approach proves too expensive in calculation time. When the free surface stretching is too high, it becomes very difficult to implement due to the frequent remeshing required. In the Eulerian approach, a single Navier–Stokes equation system is solved on a fixed Cartesian grid over the whole computational domain, with variable density and viscosity. Different techniques are used to take into account the evolution of the phase distribution, for example, the marker method (Daly [8, 9], Rider and Kothe [31], Popinet and Zaleski [30]), the level-set method (Sussman and Smereka [35]), or the volume of fluid method (Hirt and Nichols [18], Youngs [52], or Rudman [32]). At each calculation time, they all allow the free surface evolution associated with the velocity field to be precisely predicted. The physical properties of the unique mixed fluid are then redistributed according to the new phase arrangement. Finite volumes on a MAC (marker and cell) grid are commonly used to discretize the Navier–Stokes equation system. Even if the Eulerian approach is less precise than the Lagrangian one, it allows the direct numerical simulation of three-dimensional multiphase flow problems to be easily programmed when the interface is strongly distorted.

In spite of the methods described above, significant unsolvable problems still exist in fluid mechanics (Scardovelli and Zaleski [33]). Difficulties arise when large three-dimensional simulations are performed at high Reynolds numbers because thin boundary layers need to be accurately resolved. For example, the flattening of a water droplet on a water film in a gaseous medium or a liquid metal particle crush on a solid substrate is still unsolved. Owing to computer memory limits, the existing surface- or volume-tracking methods do not provide suitable solutions for three-dimensional nonsymmetric problems. On the one hand, the gigantic grids generated cannot be stored on most computers (even those with parallel architectures), and on the other hand, the extensive computational time makes the computations impracticable. The only suitable solution is to consider symmetries in the problem in question and to couple the numerical solver with parallel algorithms. With such an approach, Gueyffier and Zaleski [12, 13] were able to simulate the splashing of a drop on a liquid film in three dimensions.

To reduce the number of computational nodes several authors have suggested the idea of refining the grid near the free surface, since most of the phenomena are concentrated near the interface. Lock *et al.* [25] proposed an Eulerian unstructured finite element method, which locally adapts the mesh near the interface at each iterative time step. However, for three-dimensional simulations, the problem of lack of memory remains because the linear system generated is still very large. A full approximation storage (FAS) multigrid technique was introduced by Thompson and Lezeau [39] for two-dimensional, steady, incompressible, viscous multiphase flows. However, this method is neither local in space nor adaptive over time. It is clearly applicable to convergence acceleration of local iterative solvers, but it does not provide any memory or calculation time improvements. An AMR (adaptive mesh refinement) multigrid method was also presented by Sussman *et al.* [36] to handle two-phase flows with a level-set method. Even if the AMR technique limits the number of calculation nodes and allows the flow resolution scales to change, this method is not

local enough to meet the memory requirements described above. Strain [34] implemented a semi-Lagrangian level-set method on a quadtree mesh. In his tree method the computational effort is concentrated on the interface and the accuracy is comparable to that of a uniform mesh method, while time and memory costs are lower. The results are very promising for scalar velocity field problems. However, no coupling to a Navier–Stokes solution is presented.

Building an algorithm that refines the grid near the interface is fundamental to any improved method and has to be developed by means of a multigrid method, in order to reduce the linear system size. This method must be able to change the scale of the solution on only one or two cells if necessary. In this way, a three-dimensional calculation algorithm can be obtained that is able to solve, for example, the "crownlike" instability of a droplet splashing on a thin liquid film (Yarin and Weiss [51]), where the length-scale ratio between the initial droplet diameter and the finger diameter can be as high as 50.

In the present paper, a new local multigrid method is presented, which offers the possibility of zooming onto one calculation node and obtaining a solution using a refined grid around this node. The OCLM (one-cell local multigrid) technique is a general algorithm independent of the discretization schemes and the Navier–Stokes solver. When the refinement criterion is fulfilled on any one node, a small calculation domain (nine cells) is built over this node. The solution for the multigrid domains is calculated by the same procedure as that used on the original coarser grid. An original augmented Lagrangian method, coupled to a BiCGSTAB solver, is investigated to numerically solve the motion equation system, and a VOF-like algorithm, based on total variation diminishing (TVD) schemes, is developed to handle the interface tracking procedure. The Navier–Strokes solver is chosen for its robustness and the TVD method for its easy programming. A new type of composite boundary conditions is developed to solve the flow at the cell scale. Finally, this robust and flexible OCLM method can act as a zoom, able to solve the multiphase flow at different space scales.

After a complete presentation of the solution to a multiphase flow using a coarse grid, the OCLM method is described and validated with different scalar front tracking tests. Classical two-phase flow problems such as nonlinear drop oscillations, Rayleigh–Taylor instabilities, and drop impacts on liquid films are simulated in two dimensions. The capacity of this method to accurately track surface deformation and stretching is demonstrated, as well as its ability to save computer time and memory storage.

2. GOVERNING EQUATIONS AND RESOLUTION ALGORITHM FOR A SINGLE GRID

2.1. *One-Fluid Model*

A single set of two-dimensional and incompressible Navier–Stokes equations with variable density and viscosity is solved over the entire domain. The conservation equations are convolved with an indicator function C, which is taken to be zero in one fluid and one in the other fluid. Let $\mathbf{u}$ be the velocity field, $\mathbf{g}$ the gravity vector, p the pressure, σ the surface tension, κ the curvature, μ the viscosity, and ρ the density. If the sliding between the phases is taken as being negligible, the velocity field can be assumed to be continuous through the free surface. Without any phase changes the incompressible flow can then be considered locally isovolume. In a uniform Cartesian coordinate system (x, y), associated

with a bounded domain Ω, the one-fluid model can be expressed as

$$\rho = \rho_0 + (\rho_1 - \rho_0)C \tag{1}$$

$$\mu = \mu_0 + (\mu_1 - \mu_0)C \tag{2}$$

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

$$\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla)\mathbf{u} = \mathbf{g} - \frac{1}{\rho}\boldsymbol{\nabla} p + \frac{1}{\rho}\nabla \cdot [\mu(\nabla \mathbf{u} + \nabla^T \mathbf{u})] + \frac{1}{\rho}\sigma\kappa\delta_i \mathbf{n}_i \tag{4}$$

$$\frac{\partial C}{\partial t} + \mathbf{u} \cdot \boldsymbol{\nabla} C = 0, \tag{5}$$

where δ_i is a Dirac function indicating the interface, $\mathbf{n}_i$ is the unit normal to the interface, and ρ_0, ρ_1, μ_0, and μ_1 are the respective densities and viscosities in each phase.

The dimensionless parameters characterizing the flow are the Atwood number $A = (\rho_1 - \rho_0)/(\rho_1 + \rho_0)$, the Reynolds number $\mathrm{Re} = \rho L u/\mu$, and the Weber number $\mathrm{We} = \rho L u^2/\sigma$. L is a characteristic length scale of the flow and u is the intensity of the velocity.

The advection equation (5) of the phase function C, or "colour function," describes the changes of the free surface and simultaneously characterises the evolution of the physical characteristics of the fluids by means of (1) and (2). The interface between the fluids is defined as the discontinuity of C. In practical terms, it is the line defined by C equal to 0.5. The two-phase flow is analysed in terms of an equivalent single fluid whose properties ρ and μ are related to ρ_0, ρ_1, μ_0, and μ_1 of the original two phases by the colour function C. The model can be easily extended to flows dealing with more than two phases, by modifying the phase function C. The numerical solution of (1–5) requires special attention because of (i) the hyperbolic character of Eq. (5) and (ii) the discontinuity of the density and viscosity fields in Eq. (4).

2.2. *Numerical Solution for a Single Grid*

The quality of the interface tracking procedure is strongly dependent on the precision with which the velocity field is calculated. In particular, the incompressibility condition must be satisfied to a good approximation, even with high density or viscosity ratios. Appropriate schemes must be developed to discretize the phase function equation (5) in such a way as to retain the discontinuous character of C. The algorithms used in this section were developed and described by Vincent and Caltagirone [43] in a previous article.

2.2.1. Interface capturing method. The classical methods for solving the advection equation of a discontinuous phase function C are derived from reformulating the problem with a smooth function, based on different criteria such as the volume fraction (VOF-like methods [18, 52]) or the distance function (level-set methods [8], [9, 35]). It was decided to directly compute Eq. (5) for the sake of simplicity. Classical schemes are not efficient enough to treat the hyperbolic character of (5) and the advection of shocks [23, 17]. High-order schemes, such as the Lax–Wendroff, Beam–Warming, or QUICK ones, create spurious oscillations leading to nonphysical solutions, and first-order monotonic schemes are very diffusive. High-resolution numerical methods exist to solve scalar hyperbolic equations (Sweby [37] and LeVeque [23]). These methods must satisfy the property of total variation diminishing. This implies that in the regular zones of the solution high-order resolution is obtained, whereas near the discontinuities or the strong variations of the solution, the

order of the method is decreased to ensure the TVD property leading to a monotonic solution.

An explicit Lax–Wendroff TVD (LWT) time-stepping scheme is used for the advection of the phase indicator function C (Eq. (5)). The stability conditions for this scheme are

$$\begin{aligned} 0 < N^x_{\mathrm{CFL}} < 1, \\ 0 < N^y_{\mathrm{CFI}} < 1, \end{aligned} \tag{6}$$

where $\mathbf{u} = (u_x, u_y)$ is the velocity, $N^x_{\mathrm{CFL}} = u_x \frac{\Delta t}{\Delta x}$, $N^y_{\mathrm{CFL}} = u_y \frac{\Delta t}{\Delta y}$, and Δ_t, Δ_x, and Δy are the time and grid steps. With $\sigma_x = \frac{\Delta t}{\Delta x}$ and $\sigma_y = \frac{\Delta t}{\Delta y}$, the Lax–Wendroff TVD superbee scheme can be expressed as

$$\begin{aligned} C^{n+1/2}_{i,j} &= C^n_{i,j} - \sigma_x \max(0, u_x)\big(F^{n,+}_{i,j} - F^{n,+}_{i-1,j}\big) - \sigma_x \min(0, u_x)\big(F^{n,-}_{i+1,j} - F^{n,-}_{i,j}\big), \\ C^{n+1}_{i,j} &= C^{n+1/2}_{i,j} - \sigma_y \max(0, u_y)\big(F^{n+1/2,+}_{i,j} - F^{n+1/2,+}_{i,j-1}\big) \\ &\quad - \sigma_y \min(0, u_y)\big(F^{n+1/2,-}_{i,j+1} - F^{n+1/2,-}_{i,j}\big), \end{aligned} \tag{7}$$

where $F^{n,+}_{i,j} = C^n_{i,j} - \frac{\Delta x}{2}(u_x\sigma_x - 1)\gamma^n_{i,j}$ and $F^{n,-}_{i,j} = C^n_{i,j} - \frac{\Delta x}{2}(u_x\sigma_x + 1)\gamma^n_{i,j}$ for the x-direction. In these numerical flux expressions, $\gamma^n_{i,j} = \max(0, \min(1, 2\theta^n_{i,j}), \min(2, \theta^n_{i,j}))$ $(C^n_{i+1,j} - C^n_{i,j})/\Delta x$. The ratio of slopes in the upwind direction is $\theta^n_{i,j} = (C^n_{i,j} - C^n_{i-1,j})/(C^n_{i+1,j} - C^n_{i,j})$. The y-direction numerical fluxes are the same as the previous ones, but u_x becomes u_y, Δ_y replaces Δx, and the spatial direction subscripts are inverted.

Applying the LWT scheme for interface tracking problems (Vincent and Caltagirone [43]) presents many advantages over classical methods: the computation is extremely easy in two and three dimensions, the computational time is low (Rider and Kothe [31]) when compared to that for efficient VOF methods or Marker techniques, the extension to multiphase problems (more than two fluids) is easy, and the mass conservation is checked with a second-order convergence rate. In most of the cases tested, the results were shown to be similar to those from classical interface-tracking methods. However, significant differences were observed in strongly sheared problems where the free surface was strongly stretched and local resolution only covered one or two cells. These cases are also difficult to resolve by the classical methods because the VOF technique cuts the interface due to artificial numerical surface tension, the Level–Set model induces a considerable loss of mass, and the Marker method involves complex calculation times and computing. The LWT interface capturing step (7) applied at time $(n\Delta t)$ is referred to as IC^n in the rest of the paper.

2.2.2. Navier–Stokes solver. The Navier–Stokes equations are discretized by a finite-volume method on a staggered mesh (MAC, Harlow and Welsh [16]). An augmented Lagrangian procedure (Fortin and Glowinsky [11], Temam [38], Nicolas *et al.* [28]) is investigated to solve the coupling between the pressure and the velocity in the equations of motion (3–4). Assuming the flow to be artificially weakly compressible until $\nabla \cdot \mathbf{u} = 0$ is numerically satisfied and using the mass balance equation $d\rho/dt + \rho\nabla \cdot \mathbf{u} = 0$, a relation linking the velocity field and the volume force exerted by the pressure can be expressed as

$$\frac{\partial p}{\partial t} + r\nabla \cdot \mathbf{u} = 0, \tag{8}$$

where r is a positive constant.

An implicit resolution algorithm is computed to numerically translate the strong coupling between $\mathbf{u}$ and p. The augmented Lagrangian method consists of adding the term $r\nabla\cdot\mathbf{u}$ in the Navier–Stokes equations. A first-order Euler scheme is used to discretize $\partial p/\partial t$ so that expression (8) can be rewritten as follows:

$$p^{n+1} = p^n - r\,\Delta t\nabla\cdot\mathbf{u}. \tag{9}$$

The implicit pressure term $\boldsymbol{\nabla} p^{n+1}$ in (4) can be rewritten as $\boldsymbol{\nabla} p^n - r'\nabla(\nabla\cdot\mathbf{u})$, using expression (9). Here r' is a numerical parameter that is approximately equal to the ratio between Δt and the isothermal compressibility. In the problems studied the range of r' was $100 < r' < 10{,}000$. Finally, the implicit equations of motion were computed to solve the two-phase flow,

$$\begin{aligned}&\frac{\partial\mathbf{u}}{\partial t} + \mathbf{u}^n\cdot\nabla\mathbf{u}^{n+1} - \frac{r'}{\rho}\nabla(\nabla\cdot\mathbf{u}^{n+1})\\ &\quad = -\frac{1}{\rho}\boldsymbol{\nabla} p^n + \mathbf{g} - \frac{1}{\rho}\nabla\cdot[\mu(\nabla\mathbf{u}^{n+1} + \nabla^T\mathbf{u}^{n+1})] + F_{\mathrm{ST}}^{n+1},\end{aligned} \tag{10}$$

where F_{ST}^{n+1} is a volume force due to surface tension and is described in the next section.

The augmented Lagrangian term $\frac{r'}{\rho}\nabla(\nabla\cdot\mathbf{u}^{n+1})$ ensures an implicit reformulation of the coupling between pressure and velocity; it suppresses the complex definition of boundary conditions on the pressure, and above all it acts as a constraint in (10) to enforce the divergence-free condition $\nabla\cdot\mathbf{u} = 0$. In solving Eq. (10), the pressure is updated using the explicit expression (9). The augmented Lagrangian approach (9)–(10) is a reformulation of the motion equations (3)–(4) in terms of an optimisation problem in which a velocity–pressure saddle point must be determined by an Uzawa algorithm [42]. The augmented Lagrangian algorithm can be considered a variant of the artificial compressibility method (Peyret and Taylor [29]), with a divergence term in the momentum equation that implicitly accounts for the incompressibility constraint. This approach is simpler than a projection method, where pressure boundary counditions can sometimes be difficult to impose (Peyret and Taylor [29]). In contrast, Dirichlet and Neumann boundaries can easily be imposed in the augmented Lagrangian Navier–Stokes solver.

Following the work of Angot [2] and Caltagirone *et al.* [5], volume and surface penalty terms were introduced into the Navier–Stokes equations. In this way velocities were implicitly imposed at the boundaries of Ω and in the inner domain if needed. Then, Eq. (10) reads

$$\begin{cases}\dfrac{\partial\mathbf{u}}{\partial t} + \mathbf{u}^n\cdot\nabla\mathbf{u}^{n+1} + \mathbf{B}_{\mathbf{u}}^{\mathrm{v}}(\mathbf{u}^{n+1} - \mathbf{u}_\infty) - \dfrac{r'}{\rho}\nabla(\nabla\cdot\mathbf{u}^{n+1})\\ \quad = -\dfrac{1}{\rho}\boldsymbol{\nabla} p^n + \mathbf{g} - \dfrac{1}{\rho}\nabla\cdot[\mu(\nabla\mathbf{u}^{n+1} + \nabla^T\mathbf{u}^{n+1})] + F_{\mathrm{ST}}^{n+1},\\ \dfrac{\partial\mathbf{u}^{n+1}}{\partial\mathbf{n}} = \mathbf{B}_{\mathbf{u}}^{\mathrm{s}}(\mathbf{u}^{n+1} - \mathbf{u}_\infty)\cdot\mathbf{n} \text{ on the boundary } \Gamma \text{ of the physical domain,}\end{cases} \tag{11}$$

where $\mathbf{n}$ is a normal to the boundaries of Ω and $\mathbf{u}_\infty$ is a reference velocity. $\mathbf{B}_{\mathbf{u}}^{\mathrm{v}}$ and $\mathbf{B}_{\mathbf{u}}^{\mathrm{s}}$ are diagonal matrices. Their components $B_{\mathbf{u}}^{\mathrm{v},l}$ and $B_{\mathbf{u}}^{\mathrm{s},l}$ are respectively volume control parameters which are used to impose $\mathbf{u}_\infty$ in a velocity control volume and surface control parameters enforcing the boundary conditions on Γ. When $B_{\mathbf{u}}^{\mathrm{v},l}$ is set equal to infinity, the implicit solution of expression (11) involves $\mathbf{u}^{n+1} = \mathbf{u}_\infty$. For $0 < B_{\mathbf{u}}^{\mathrm{v},l} < +\infty$, the numerical solution corresponds to a hybrid value deduced from the equilibrium between

the mechanical constraints and the reference velocity. The penalty term is inactive when $B_{\mathbf{u}}^{\mathrm{v},l}$ is set equal to zero. Dirichlet boundary conditions are easily imposed by setting $B_{\mathbf{u}}^{\mathrm{s},l} = +\infty$, whereas Neumann boundary conditions are obtained by setting $B_{\mathbf{u}}^{\mathrm{s},l} = 0$. By analogy with the theory of thermal exchanges, $0 < B_{\mathbf{u}}^{\mathrm{s},l} < +\infty$ is associated with Fourrier-like boundary conditions.

To approximate the differential form of the equation of motion (10) a second-order Euler scheme, or GEAR scheme, was used on the time derivatives while a third-order QUICK scheme (Leonard [22]) was applied to the nonlinear convective term and a second-order centred scheme was chosen to discretize the viscous and the augmented Lagrangian terms. An iterative BiCGSTAB (Bi-Conjugate Gradient Stabilised) algorithm (Van Der Vorst [50]) was chosen to solve the linear system generated by the discretization of (10). In free surface flow problems the spatial discontinuity of the characteristics through the interface induces large off-diagonal terms in the linear system. Because of this, when the ratios ρ_1/ρ_0 or μ_1/μ_0 assume values much larger than 1, efficient preconditioning is required to ensure the convergence of the BiCGSTAB iterative method. The Modified Incomplete LU (MILU) algorithm of Gustafsson [14] has been used in this study. Even with strongly discontinuous characteristics at the free surface ($\rho_1/\rho_0 > 10^4$ or $\mu_1/\mu_0 > 10^7$), the Navier–Stokes solver provides precise solution, with a divergence-free velocity field. Different solutions of the Rayleigh–Taylor instability problem are presented in Fig. 1. For strong viscosity ratios, a hybrid augmented Lagrangian/projection method can provide more accurate solutions

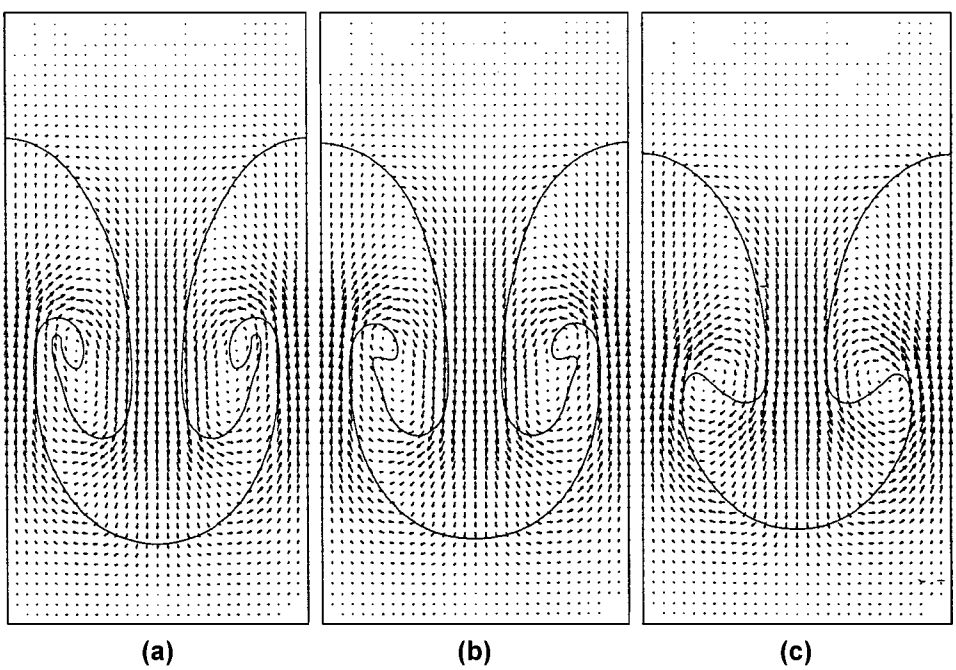

FIG. 1. Numerical simulation of a two-dimensional Rayleigh–Taylor instability on a 120×240 grid. The initial perturbation is 10% of the domain height. The viscosity is the same in the two fluids. The ratio between the heavy and lighter fluid densities ρ_1/ρ_0 is 2 ($\rho_0 = 500\ \mathrm{kg \cdot m^{-3}}$ and $\rho_1 = 1000\ \mathrm{kg \cdot m^{-3}}$) and the Atwood number is $A = 0.33$. The velocity field and the free surface ($C = 0.5$) are presented with $\mathrm{We} = +\infty$, 1000, and 100 in (a), (b), and (c) respectively. As predicted by several studies (Elgowainy and Ashgriz [10]), the surface tension stabilises the interface and limits the instability growth.

(Vincent and Caltagirone [43]). In the following sections, the augmented Lagrangian solver (9–10) at time $(n\Delta t)$ is referred to as AL^n.

2.2.3. Surface tension discretization. Following the CSF method of Brackbill *et al.* [4] and Lafaurie *et al.* [20], we can write the surface tension as a volume force according to the density,

$$\begin{aligned} C^* &= \frac{1}{2}\rho_{i,j} + \frac{1}{8}(\rho_{i+1,j} + \rho_{i-1,j} + \rho_{i,j+1} + \rho_{i,j-1}), \\ F_{\mathrm{TS}} &= -\sigma \frac{\rho}{\rho_{\mathrm{m}}} \frac{\boldsymbol{\nabla} C^*}{[\rho]} \nabla \cdot \left(\frac{\boldsymbol{\nabla} C^*}{\|\boldsymbol{\nabla} C^*\|} \right), \end{aligned} \tag{12}$$

where ρ_{m} is the arithmetic average between ρ_0 and ρ_1, $[\rho]$ is the density jump across the free surface, and $\|C^*\|$ is the L^2-norm of C^*.

This formulation is very convenient because, on a fixed Cartesian grid, the two-phase flow is only characterised by the colour function as a unique fluid with variable physical characteristics and the free surface location is not tracked explicitly by such a method. The volume surface tension method (12) avoids calculating the geometrical properties of the interface such as the curvature κ. The main drawback of the Continuum Surface Force (CSF) method of Brackbill *et al.* [4] is the generation of spurious or parasite currents near the free surface. When either viscosity or gravity dominates the dynamics, the surface tension effects are negligible and the parasite currents are insignificant. However, in certain situations, such as bubble instabilities, the capillary effects are dominant and the spurious currents generated can destroy the numerical solution in near-equilibrium configurations. The discretization of the surface tension (12) can be directly applied to the phase function C. However, to limit the production of parasite currents near the interface (Lafaurie *et al.* [20]), a smoothed function C^* is used instead of C in expression (12). The local character of the method is extended over four or five cells near the free surface by repeating the smoothing operation four times.

Contrary to Brackbill *et al.* [4], who use finite differences to discretize the surface tension force on a MAC grid, Vincent *et al.* [44] use a mixed finite differences/finite volumes discretization to obtain an approximate expression of (12). Working on a velocity control volume V_{c}, they consider $-\sigma \frac{\boldsymbol{\nabla} C^*}{[C^*]}$ to be constant on V_{c} whereas the divergence term $\nabla \cdot (\frac{\boldsymbol{\nabla} C^*}{\|\boldsymbol{\nabla} C^*\|})$ is integrated on the surface of V_{c} using the divergence theorem of Green and Ostrogradsky. In summary, for the x-component, the discretization of F_{ST} is expressed as follows:

$$\begin{aligned} \kappa = -\nabla \cdot \frac{\boldsymbol{\nabla} C}{\|\boldsymbol{\nabla} C\|} = -\nabla \cdot \mathbf{n_i} &= \frac{N_{x_{\mathrm{right}}}}{\Delta x_i \sqrt{N_{x^2_{\mathrm{right}}} + N_{y^2_{\mathrm{right}}}}} - \frac{N_{x_{\mathrm{left}}}}{\Delta x_i \sqrt{N_{x^2_{\mathrm{left}}} + N_{y^2_{\mathrm{left}}}}} \\ &\quad + \frac{N_{y_{\mathrm{up}}}}{\Delta y_j \sqrt{N_{x^2_{\mathrm{up}}} + N_{y^2_{\mathrm{up}}}}} - \frac{N_{y_{\mathrm{down}}}}{\Delta y_j \sqrt{N_{x^2_{\mathrm{down}}} + N_{y^2_{\mathrm{down}}}}}, \\ F_{\mathrm{ST}} &= -\frac{\sigma}{[\rho]} \frac{\rho_{i-1/2,j}}{\rho_m} \frac{(C^*_{i,j} - C^*_{i-1,j})}{\Delta x_i} \kappa_{i,j}. \end{aligned} \tag{13}$$

The two components $(N_x,\ N_y)$ of the normal to the free surface are expressed on the right, left, up, and down interfaces of each discretization control volume. On the control volume

(i, j), they are defined as

$$
\begin{aligned}
N_{x_{\text{right}}} &= \frac{C^*_{i+1,j} - C^*_{i,j}}{2\Delta x_{i+1}} + \frac{C^*_{i,j} - C^*_{i-1,j}}{2\Delta x_i},\\
N_{x_{\text{up}}} &= \frac{C^*_{i,j+1} - C^*_{i-1,j+1}}{2\Delta x_i} + \frac{C^*_{i,j} - C^*_{i-1,j}}{2\Delta x_i},\\
N_{x_{\text{left}}} &= \frac{C^*_{i,j} - C^*_{i-1,j}}{2\Delta x_i} + \frac{C^*_{i-1,j} - C^*_{i-2,j}}{2\Delta x_{i-1}},\\
N_{x_{\text{down}}} &= \frac{C^*_{i,j} - C^*_{i-1,j}}{2\Delta x_i} + \frac{C^*_{i,j-1} - C^*_{i-1,j-1}}{2\Delta x_i},\\
N_{y_{\text{right}}} &= \frac{C^*_{i,j+1} - C^*_{i,j}}{2\Delta y_{j+1}} + \frac{C^*_{i,j} - C^*_{i,j-1}}{2\Delta y_j},\\
N_{y_{\text{up}}} &= \frac{C^*_{i,j+1} - C^*_{i,j}}{2\Delta y_{j+1}} + \frac{C^*_{i-1,j+1} - C^*_{i-1,j}}{2\Delta y_{j+1}},\\
N_{y_{\text{left}}} &= \frac{C^*_{i-1,j+1} - C^*_{i-1,j}}{2\Delta y_{j+1}} + \frac{C^*_{i-1,j} - C^*_{i-1,j-1}}{2\Delta y_j},\\
N_{y_{\text{down}}} &= \frac{C^*_{i,j} - C^*_{i,j-1}}{2\Delta y_j} + \frac{C^*_{i-1,j} - C^*_{i-1,j-1}}{2\Delta y_j}.
\end{aligned}
\tag{14}
$$

According to the second component of **u**, the expression of the surface tension force (12) is directly obtained by a linear combination of the subscripts in Eqs. (13) and (14). This formulation was successfully used on several three-dimensional low Weber number problems such as the liquid droplet impact on a liquid or a solid substrate (Vincent *et al.* [45]). The influence of the surface tension on Rayleigh–Taylor instability is presented in Fig. 1 as an example of a two-phase flow simulation.

3. ONE CELL LOCAL MULTIGRID METHOD

3.1. *OCLM: Local Mesh Refinement Algorithm on One Cell*

The One Cell Local Multigrid (OCLM) method is constructed with the Navier–Stokes solver and the interface-tracking algorithm $\text{AL}^n - \text{IT}^n$. However, it can be used to solve any equation with any solver [46–48]. A natural way to improve the accuracy of free surface problem simulations, while also limiting the computational time and memory costs, is to refine the grid near the interface. Nonuniform meshes can produce inaccurate solutions or generate large algebraic systems. Using the local character of multigrid methods, we have developed an adaptative local mesh refinement algorithm. At each grid level l ($l = 0, 1, \ldots, l_{\max}$), $s_{\max}$ subgrids $G_{l,s}$($s = 0, 1, \ldots, s_{\max}$) are generated with mesh spacing h_l. The coarsest grid is G_0. It initially contains all the necessary information and represents the whole calculating domain. As illustrated in Fig. 2, a fundamental property of the local mesh refinement is the following:

$$\forall s,\ G_{l,s} \subset G_{l-1}. \tag{15}$$

In previous studies of Berger and Collela [3], Caltagirone *et al.* [5], Khadra *et al.* [19], or Angot *et al.* [1] on local mesh refinement methods, rectangular parcels containing many

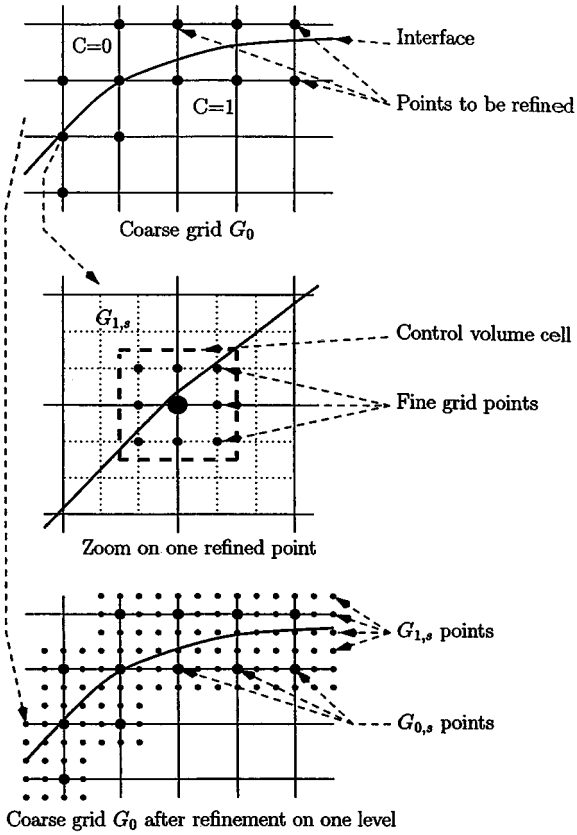

FIG. 2. Example of local mesh refinement with the OCLM method.

coarse grid points were refined. In contrast, in the OCLM method, local refinement occurs in a control volume around a single coarse grid point verifying a refinement criterion described below (see Fig. 2). This technique minimizes the number of refined points and the memory cost. Moreover, as explained by Berger and Collela [3], the use of rectangular grids is very convenient because the calculations can be carried out with an identical solver at each of the different grid levels.

A criterion related to the distance of a point from the free surface is used to build the multigrid architecture:

$$\mathrm{Cr}_{\mathrm{phys}} = \|\nabla C\|. \tag{16}$$

If $\mathrm{Cr}_{\mathrm{phys}} = 0$ on each side of coarse cell $V_c^{0,s}$ centred at a pressure node, no refinement is carried out in this cell. If $\mathrm{Cr}_{\mathrm{phys}} > 0$ on one or more sides, the free surface crosses the cell and a 9-point refinement, illustrated in Fig. 2, is performed in this cell. The coarse grid G_0 is scanned and a set of 3×3 subgrids $G_{1,s}$ is built with a grid step $h_1 = h_0/3$. An odd cutting of the coarser cells in each space direction, 3×3 for example, is essential because it allows a natural connection between the subgrids, as shown in Fig. 2. Thanks to the perfect joining of the refinement cells and the explicit character of the LWT algorithm, the evolutions of the free surface can be described at each of the different multigrid levels maintaining the stability and the precision of the interface capturing method. A local refinement procedure is then

performed on each $G_{1,s}$ subgrid in the same manner as on G_0. The subgrids are thus refined recursively until the grid step becomes approximately equal to the smallest lengthscale of the problem considered, corresponding to a "precision criterion" Cp. Initially, the whole multigrid structure is built according to the interface position on G_0 and the exact position of C is imposed at each multigrid level. By knowing the distribution of C on $G_{l,s}$ and by calculating the velocity field on this grid, the interface tracking algorithm can be directly applied on the 3×3 fine grids $G_{l,s}$ to obtain the new distribution of the phase function C. This procedure is possible thanks to the explicit character of the LWT algorithm. As the dynamics and the geometry of the interface change, the multigrid solver can track the interface in time and space by refining the grid on several levels when either the physical criterion $\mathrm{Cr}_{\mathrm{phys}}$ or the precision criterion Cp have been verified at coarse points. With this step undertaken, the phase function C is initialised on the subgrid l by projecting the finer values of C onto each $G_{l,s}$.

Because of the incompressibility condition, the fine grids need to be coupled with the coarse grids to achieve the implicit solving of the motion equation. The Navier–Stokes equation system is first solved on G_0. Next, p and $\mathbf{u}$ are initialised on each $G_{1,s}$ by means of a classic Q1 interpolation operator (see Appendix A). Composite boundary conditions, described in Section 3.2, are used on the boundaries of $G_{1,s}$ to ensure the conservation of the mass and momentum fluxes. The Navier–Stokes equations are then solved on the $G_{1,s}$ subgrids. This procedure is repeated on all of the multigrid levels.

The solution from a fine grid solution can be transferred back to the previous coarse grid (G_l to G_{l-1}) using a direct injection procedure,

$$\forall (x, y) \in G_l \cap G_{l-1}, \quad \phi_{l-1}(x, y) = \phi_l(x, y), \tag{17}$$

where ϕ represents an unknown variable of the problem such as C, p, or one of the components of $\mathbf{u}$. This reverse procedure is referred to as a restriction step. In this way the problem can be solved again on G_0 to take into account the corrections brought about by the restriction step on the whole coarse grid solution.

Another procedure, which has been used in this article, is the full weighting interface control volume restriction FWICV (Hackbush [15], Laugier *et al.* [21]). It consists of estimating the phase function distribution on a volume $V_{\mathrm{c}}^{l-1,s}$ as the sum of C on the $G_{l,s'}$ subgrids,

$$C^{l-1,s}(x, y) = \frac{1}{S\left(V_{\mathrm{c}}^{l-1,s}\right)} \sum_{s' \in R_l^{l-1,s}} \int_{V_{\mathrm{c}}^{l,s'}} C(x, y)\, dv, \tag{18}$$

where $R_l^{l-1,s} = \{s' / V_{\mathrm{c}}^{l-1,s} \cap V_{\mathrm{c}}^{l,s'} \neq \emptyset\}$ is the restriction space between G_l and G_{l-1}, $C^{l-1,s}(x, y)$ is the projection of the phase function on a calculation domain $V_c^{l-1,s}$, and $S(V_c^{l-1,s})$ is the surface of a control volume of G_{l-1}. Thanks to (18), the fluid rearrangement is conservative at each multigrid level and the resulting dynamics on the coarser grids are coupled to the dynamics on the finer ones.

3.2. *Composite Boundary Conditions (CBC)*

The computation of the discrete Navier–Stokes problem (11) relative to each fine grid $G_{l,s}$ requires the definition of boundary conditions which are compatible with the solution

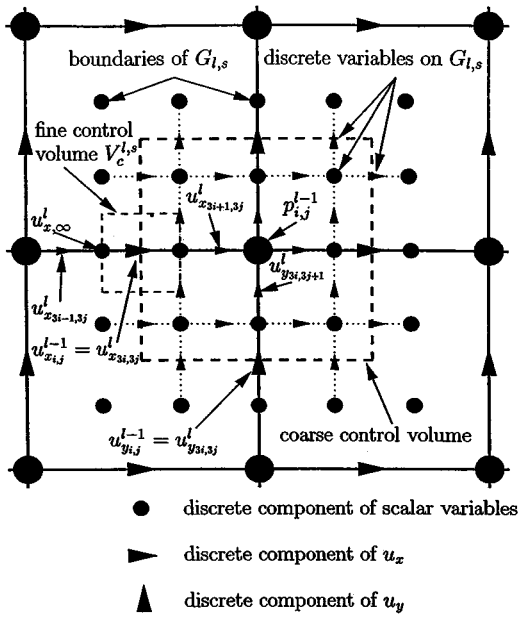

FIG. 3. Discrete unknown variables in a control volume $V_c^{l,s}$ located at a boundary of $G_{l,s}$.

on G_{l-1}. The 3×3 fine discrete domains are extended to a 5×5 mesh to deal with the boundary conditions, as shown in Fig. 3. A classical technique applied in several multigrid methods (FIC [1], FAS [39], or AMR [36]) consists of imposing nonhomogeneous Dirichlet boundary conditions derived from the coarse grid solution ($\mathbf{u}^{l-1}$, p^{l-1}),

$$\mathbf{u}^{l,s} \cdot \mathbf{n} = \begin{cases} u_x^{l-1,\text{prol}} & \text{if } \mathbf{n} = \mathbf{n}_x, \\ u_y^{l-1,\text{prol}} & \text{if } \mathbf{n} = \mathbf{n}_y, \end{cases} \tag{19}$$

where $\mathbf{n}$ is the unit normal to a boundary $\Gamma_{l,s}$ of $G_{l,s}$, $\mathbf{n}_x$ is a unit normal in the x-direction, and $\mathbf{n}_y$ is a unit normal in the y-direction. The velocities $u_x^{l-1,\text{prol}}$ and $u_y^{l-1,\text{prol}}$ are respectively the interpolated values of $\mathbf{u}_x^{l-1}$ and $\mathbf{u}_y^{l-1}$. In the OCLM method Dirichlet boundary conditions ($B_{\mathbf{u}}^{s,l} = +\infty$) were imposed using a Q1 interpolation procedure. However, the small size of the 5×5 subgrids leads to incorrect solutions if this method is used. In particular, the interpolation operator does not preserve the divergence-free property (see Appendix B).

Choosing values of surface control parameters varying from 0 to $+\infty$ and coupling these values with volume control parameters of the same order, one notices that for average values of $B_{\mathbf{u}}^{\text{s},l}$ and $B_{\mathbf{u}}^{\text{v},l}$ (from 10^2 to 10^6), the numerical solution is not restricted to adopting a direction normal to $\Gamma_{l,s}$ and therefore is greatly improved. In this way, to relax the constraints at the boundaries of each $G_{l,s}$, the numerical idea is to introduce a new type of boundary conditions, called composite boundary conditions (CBC), into the Navier–Stokes equation system. The CBC couple in the discrete motion equation system (11) Fourier-like conditions in the direction normal to the boundary $\Gamma_{l,s}$ of $G_{l,s}$ with hybrid volume conditions on the velocity component which is tangential to $\Gamma_{l,s}$. The principle of the CBC is to be able to impose each component of the velocity on the boundaries in a coupled way. The CBC suggest the direction of the flow at the boundaries of the calculation domain. This contrasts with the classical boundary conditions that impose either the flux (Neumann condition) or

one component of the velocity (Dirichlet condition), which are scalar components of the flow.

On the fine grids $G_{l,s}$, solving the discrete Navier–Stokes equations (11) with the CBC requires the definition of control parameters $B_{\mathbf{u}}^{\mathrm{v},l}$ and $B_{\mathbf{u}}^{\mathrm{s},l}$ as well as reference velocities $\mathbf{u}_{\infty}^{l}$ associated with them. Each component of $\mathbf{u}_{\infty}$ on G_l is defined as the average interpolation of the coarse grid values to maintain the coupling between the multigrid levels. The control parameters are estimated by relating the concept of vectorial boundary conditions to the physics included in the Navier–Stokes model, by means of the rate of deformation tensor D and the viscous stress tensor

$$\tau = \begin{pmatrix} \tau_{x,x} & \tau_{x,y} \\ \tau_{y,x} & \tau_{y,y} \end{pmatrix}.$$

They are defined as follows:

$$\begin{aligned} D &= \frac{1}{2}[\nabla \mathbf{u} + \nabla^T \mathbf{u}], \\ \tau &= 2\mu D. \end{aligned} \tag{20}$$

If we assume that the symmetrical components of the viscous stress tensor are preserved between two multigrid levels, we can estimate $B_{\mathbf{u}}^{\mathrm{v},l}$ and $B_{\mathbf{u}}^{\mathrm{s},l}$ on the boundaries of each $G_{l,s}$. The finite volume approximation of $\nabla \cdot \tau$ is developed on a fine control volume $V_{\mathrm{c}}^{l,s}$ around point $(3i, 3j)$ (see Fig. 3). The normal contribution $T_{x_{3i,3j}}$ of the viscous stress tensor in the Navier–Stokes equations is expressed on side $(3i - \frac{1}{2}, 3j)$ of $V_{\mathrm{c}}^{l,s}$, for the first velocity component:

$$T_{x_{3i,3j}} = \int_{V_{\mathrm{c}}^{l,s}} \nabla \cdot \tau \, dv = \int_{\Gamma_{l,s}} \tau_{x,x} \cdot \mathbf{n} \, d\gamma = \left[-\mu \frac{\partial u_x}{\partial x} \right]_{3i-1/2,3j}^{3i+1/2,3j}. \tag{21}$$

From the coarse grid, the interpolation operator leads to

$$\begin{cases} u^l_{x_{3i+1,3j}} = \frac{1}{3} u^l_{x_{i+1,j}} + \frac{2}{3} u^l_{x_{i,j}}, \\ u^l_{x_{3i,3j}} = u^l_{x_{i,j}}, \\ u^l_{x_{3i-1,3j}} = \frac{2}{3} u^l_{x_{i,j}} + \frac{1}{3} u^l_{x_{i-1,j}}. \end{cases} \tag{22}$$

In a discrete form, the following is obtained:

$$\begin{aligned} T_{x_{3i,3j}} &= -\mu \left[\frac{u^l_{x_{3i+1,3j}} - 2u^l_{x_{3i,3j}} + u^l_{x_{3i-1,3j}}}{h_l} \right] = -\mu \left[\frac{\frac{1}{3}u^{l-1}_{x_{i+1,j}} - \frac{2}{3}u^{l-1}_{x_{i,j}} + \frac{1}{3}u^{l-1}_{x_{i-1,j}}}{h_l} \right] \\ &= -\mu \left[\frac{\frac{1}{3}u^{l-1}_{x_{i+1,j}} - \frac{1}{3}u^{l-1}_{x_{i,j}} - u^l_{x_{3i-1,3j}} + u^l_{x_{3i-2,3j}}}{3h_l} \right] = -\mu \left[\frac{\frac{1}{9}u^{l-1}_{x_{i+1,j}} - \frac{2}{9}u^{l-1}_{x_{i-1,j}} - \frac{1}{3}u^l_{x_{3i-1,3j}}}{h_l} \right]. \end{aligned} \tag{23}$$

In Eq. (23), the velocity $u^l_{x_{3i-1,3j}}$, corresponding to side $(3i - \frac{1}{2}, 3j)$ of $V_{\mathrm{c}}^{l,s}$, is exterior to $G_{l,s}$ (Fig. 3). To determine this velocity, the viscous stress is modelled as a flux between the

fine grid and the outside region. As was explained in Section 2.2.2, this method consists of estimating a flux based on a Fourier-like control parameter and a reference velocity,

$$\begin{aligned} -\mu\frac{\partial u_x}{\partial x} &= B_{\mathbf{u}}^{\mathrm{s},*}(u_x - u_{x,\infty}),\\ -\frac{\partial u_x}{\partial x} &= B_{\mathbf{u}}^{\mathrm{s},1}(u_x - u_{x,\infty}), \end{aligned} \tag{24}$$

where $B_{\mathbf{u}}^{\mathrm{s},*}$ and $B_{\mathbf{u}}^{\mathrm{s},1}$ are control parameters and $B_{\mathbf{u}}^{\mathrm{s},1} = B_{\mathbf{u}}^{\mathrm{s},*}/\mu$. On side $(3i - \frac{1}{2}, 3j)$, the discrete expression of (24) is given by

$$-\left(\frac{u^l_{x_{3i,3j}} - u^l_{x_{3i-1,3j}}}{h_l}\right) = B_{\mathbf{u}}^{\mathrm{s},1}\left(u^l_{x_{3i+1/2,3j}} - u_{x,\infty}\right) = B_{\mathbf{u}}^{\mathrm{s},1}\left(\frac{u^l_{x_{3i,3j}} + u^l_{x_{3i-1,3j}}}{2} - u_{x,\infty}\right). \tag{25}$$

Estimating the reference velocity $u_{x,\infty}$ according to coarse values

$$u^l_{x_{3i-1,3j}} = \frac{5}{6}u^{l-1}_{x_{i,j}} + \frac{1}{6}u^{l-1}_{x_{i-1,j}}, \tag{26}$$

using Eq. (25), and interpolating $\mathbf{u}^l$ by the Q1 operator leads to an approximation of the outside velocity $u^l_{x_{3i-1,3j}}$ according to the coarse velocity field $\mathbf{u}^{l-1}$:

$$u^l_{x_{3i-1,3j}}\left(\frac{B_{\mathbf{u}}^{\mathrm{s},1}}{2} - \frac{1}{h_l}\right) = u^{l-1}_{x_{i,j}}\left(\frac{2B_{\mathbf{u}}^{\mathrm{s},1}}{6} - \frac{1}{h_l}\right) + u^{l-1}_{x_{i-1,j}}\left(\frac{B_{\mathbf{u}}^{\mathrm{s},1}}{6}\right). \tag{27}$$

Combining Eqs. (23) and (27), we obtain an explicit expression binding the surface control parameter $B_{\mathbf{u}}^{\mathrm{s},1}$ on the fine grid and $\mathbf{u}^{l-1}$:

$$\begin{aligned} &T_{x_{3i,3j}}\\ &= -\mu\left[\frac{\frac{1}{9}u^{l-1}_{x_{i-1,j}} + \frac{2}{9}u^{l-1}_{x_{i-1,j}} - \frac{1}{3}\left(2B_{\mathbf{u}}^{\mathrm{s},1}h_l - 6/3B_{\mathbf{u}}^{\mathrm{s},1}h_l - 6\right)u^{l-1}_{x_{i,j}} - \frac{1}{3}\left(B_{\mathbf{u}}^{\mathrm{s},1}h_l/\left[3B_{\mathbf{u}}^{\mathrm{s},1}h_l - 6\right]\right)u^{l-1}_{x_{i-1,j}}}{h_l}\right]\\ &= -\mu\left[\frac{\frac{1}{3}u^{l-1}_{x_{i+1,j}} - \frac{2}{3}u^{l-1}_{x_{i,j}} + \frac{1}{3}u^{l-1}_{x_{i-1,j}}}{h_l}\right]. \end{aligned} \tag{28}$$

By identifying the velocities in expression (28), the following is obtained:

$$\begin{aligned} B_{\mathbf{u}}^{\mathrm{s},1} &= \frac{1}{h_l}\frac{\left(3u^{l-1}_{x_{i,j}} - 2u^{l-1}_{x_{i+1,j}} - u^{l-1}_{x_{i-1,j}}\right)}{\left(2u^{l-1}_{x_{i,j}} - u^{l-1}_{x_{i+1,j}} - u^{l-1}_{x_{i-1,j}}\right)} = \frac{1}{h_l}\frac{\left(3 - 2\,u^{l-1}_{x_{i+1,j}}/u^{l-1}_{x_{i,j}} - u^{l-1}_{x_{i-1,j}}/u^{l-1}_{x_{i,j}}\right)}{\left(2 - u^{l-1}_{x_{i+1,j}}/u^{l-1}_{x_{i,j}} - u^{l-1}_{x_{i-1,j}}/u^{l-1}_{x_{i,j}}\right)},\\ B_{\mathbf{u}}^{\mathrm{s},1} &\neq \frac{2}{h_l}. \end{aligned} \tag{29}$$

Assuming $u^{l-1}_{x_{i+1,j}}$, $u^{l-1}_{x_{i,j}}$, and $u^{l-1}_{x_{i-1,j}}$ to be of the same order,

$$\begin{cases} 0.5 \le \frac{u^{l-1}_{x_{i+1,j}}}{u^{l-1}_{x_{i,j}}} \le 1.5,\\ 0.5 \le \frac{u^{l-1}_{x_{i-1,j}}}{u^{l-1}_{x_{i,j}}} \le 1.5, \end{cases} \tag{30}$$

the surface control parameter is estimated to be

$$B_{\mathbf{u}}^{\mathrm{s},1} \approx \frac{3}{2h_l}. \tag{31}$$

The demonstration developed in expressions (21)–(31) can be carried out for $\tau_{x,x}$ or $\tau_{y,y}$ on any fine control volume on the boundaries of $G^{l,s}$, with identical conclusions.

To couple the velocity components in the Navier–Stokes solution a volume control parameter $B_{\mathbf{u}}^{\mathrm{v},2}$ must be defined in the tangent direction to the boundary of $V_{\mathrm{c}}^{l,s}$. According to the second component of the velocity, a Fourier-like flux condition gives rises to the following expression:

$$-\mu \frac{\partial u_y}{\partial y} = B_{\mathbf{u}}^{\mathrm{s},2}(u_y - u_{y,\infty}). \tag{32}$$

In a dimensionless form, this reduces to

$$-\frac{\mu u_0}{L'} \frac{\partial u_y}{\partial y} = B_{\mathbf{u}}^{\mathrm{s},2}\, u_0(u_y - u_{y,\infty}), \tag{33}$$

where L' is a characteristic length, u_o is a reference velocity, and $B_{\mathbf{u}}^{\mathrm{s},2} \approx 3/2h_l$.

The symmetrical component $\tau_{y,y}$ of the viscous stress tensor has to be corrected in the momentum equations (4) with a control volume parameter $B_{\mathbf{u}}^{\mathrm{v},2}$ to check its compatibility with (33). In the Navier–Stokes equations, the second symmetrical component of the viscous stress tensor is related to $B_{\mathbf{u}}^{\mathrm{v},2}$ as follows:

$$-\mu \frac{\partial^2 u_y}{\partial y^2} = B_{\mathbf{u}}^{\mathrm{v},2}(u_y - u_{y,\infty}). \tag{34}$$

The dimensionless form of expression (34) is

$$-\frac{\mu u_0}{L^2} \frac{\partial^2 u_y}{\partial y^2} = B_{\mathbf{u}}^{\mathrm{v},2} u_0(u_y - u_{y,\infty}). \tag{35}$$

Identifying the fluxes in (33) and (35), we relate the volume and surface control parameters by

$$B_{\mathbf{u}}^{\mathrm{v},2} = \frac{L' B_{\mathbf{u}}^{\mathrm{s},2}}{L^2}. \tag{36}$$

In order to be consistent with the metrics on the solving grid $G_{l,s}$, the characteristic length L is chosen to be equal to the width h_{l-1} of $G_{l,s}$. The centred discretisation of $\partial u_y/\partial y$ induces $L' = 2L$. To summarise, assuming the conservation of symmetrical components of the viscous stress tensor between two multigrid levels, surface and volume control parameters can be estimated independent of the solution as follows:

$$\begin{aligned} B_{\mathbf{u}}^{\mathrm{s},l} &\approx \frac{3}{2h_l}, \\ B_{\mathbf{u}}^{\mathrm{v},l} &\approx \frac{2B_{\mathbf{u}}^{\mathrm{s},l}}{h_{l-1}}. \end{aligned} \tag{37}$$

In the present approach, thanks to scalar control parameters (37) acting in all the space directions, vectorial characteristics of the flow are imposed on the boundaries of the fine calculation domains $G_{l,s}$. In the following sections, the validity of the CBC and their superiority over the classical Dirichlet or Neumann conditions will be demonstrated.

3.3. *Convergence Checks on the Self-Similar Vortex Problem*

The self-similar vortex problem consists of adding a source term $S(x, y) = (S_x(x, y), S_y(x, y))$ into the Navier–Stokes equations:

$$\begin{aligned} S_x(x, y) &= -\left(\frac{\pi^2\mu}{2}\right)\cos\left(\frac{\pi x}{2}\right)\sin\left(\frac{\pi y}{2}\right), \\ S_y(x, y) &= \left(\frac{\pi^2\mu}{2}\right)\sin\left(\frac{\pi x}{2}\right)\cos\left(\frac{\pi y}{2}\right). \end{aligned} \tag{38}$$

The velocity field is initially zero and evolves toward a vortex due to the addition of the source term (38). An analytical solution is obtained for the set of equations (3), (4), and (38):

$$\begin{aligned} u_x(x, y, t) &= -\cos\left(\frac{\pi}{2}x\right)\sin\left(\frac{\pi}{2}y\right)\left[1 - e^{-\pi^2\mu t/2\rho}\right], \\ u_y(x, y, z) &= -\sin\left(\frac{\pi}{2}x\right)\cos\left(\frac{\pi}{2}y\right)\left[1 - e^{-\pi^2\mu t/2\rho}\right], \\ p(x, y, z) &= -\frac{\rho}{2}\left[\cos\left(\frac{\pi}{2}x\right) + \cos\left(\frac{\pi}{2}y\right)\right]\left[1 - 2e^{-\pi^2\mu t/2\rho} + e^{-\pi^2\mu t/\rho}\right]. \end{aligned} \tag{39}$$

Assuming the calculation domain to be a 1-m square box, the self-similar vortex problem has been restated by introducing an artificial interface, defined by the following rectangular colour function distribution:

$$C(x, y) = \begin{cases} 1 & \text{if } 0.275 \le x \le 0.725 \text{ and } 0.375 \le y \le 0.625, \\ 0 & \text{else.} \end{cases} \tag{40}$$

Expression (40) is used to evaluate the OCLM method on an interface tracking test, by refining the grid in the regions where $\nabla C \neq 0$. On one side, the colour (40) is advected with the exact velocity field. On the other side, the Navier–Stokes equations and the interface tracking are solved using the local mesh refinement. Comparisons are presented on the velocity field and the interface position. The exact position of the phase function shape is deduced from a Lagrangian advection of particles, placed on the interface.

The absolute error E_V is the L_∞ norm of the difference between the calculated and exact velocity field. Figure 4 shows how E_V changes with the number of calculation points N on the coarsest grid. The calculations illustrate second-order convergence of the solver on the coarse grid, whereas convergence rates of 1.5 and 1.8 are found respectively on G_1 and G_2. These were computed at each grid level with a fixed value of the surface control parameter $B_{\mathbf{u}}^{s,l}$ equal to $3/2h_l$. The differences in the convergence rates can be explained by the very local character of the method, whose convergence is only conditioned by the interpolation of solutions from coarse grids. However, the behaviour of the multiscale OCLM solver is

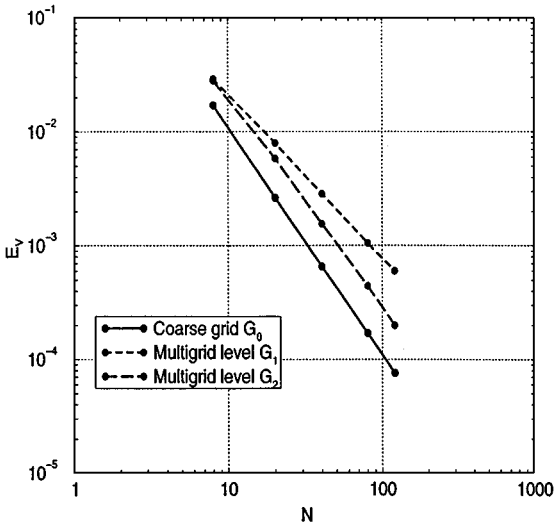

FIG. 4. Behaviour of the absolute error E_V on the velocity for several coarse grids: 8^2, 20^2, 40^2, 80^2, and 120^2.

satisfying. By starting on a coarse grid, it is able to refine the flow resolution at the cell scale maintaining a convergence rate of up to 1.5. In the present problem, in spite of the interface tracking, the dynamic generation and removal of fine cells does not affect the quality of the solver. Moreover, thanks to composite boundary conditions (CBC), the divergence-free property is fulfilled with a 10^{-13} computer error at all grid levels and the solution converges to the analytical solution.

To verify the effects of the boundary conditions on the multigrid solution, additional tests were carried out on the single-phase vortex flow problem. First, the behaviour of E_V was examined while the control parameters were modified. Parameter B_{ND} was defined as the ratio between the surface control parameter B_{comp} used in the computation and the theoretical one $B_{\mathbf{u}}^{s,l}$. According to expression (37), it is estimated by

$$B_{ND} = \frac{2B_{comp}h_l}{3}. \tag{41}$$

An example considered was the solution of the self-similar vortex on two multigrid levels with a 40×40 coarse grid. Figure 5 shows that the maximum error corresponds to Dirichlet ($B_{ND} = +\infty$) and Neumann boundary conditions ($B_{ND} = 0$). The OCLM method reaches its best performance when B_{ND} is more or less equal to 1, which confirms the theoretical results developed in the previous section.

The level of accuracy of the OCLM method is measured by comparing the evolution of the volume repaired by the phase function while the grid is being refined. Let E_S be the difference between the initial volume and the volume provided by the numerical solution. The convergence of the OCLM method to the theoretical volume is illustrated in Fig. 6. A second-order convergence in space is reached on all the multigrid levels. The local mesh refinement ensures better volume conservation from a coarse grid to a finer one. At the same time, the geometrical description of the interface is improved. Starting with a 20×20 coarse grid and carrying out the calculations on two multigrid levels (Fig. 7), one can reach a very accurate solution, being almost equal to the analytical one.

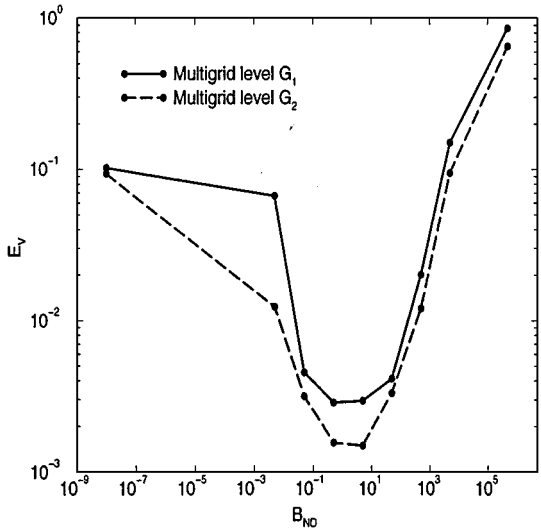

FIG. 5. Behaviour of the absolute error E_V on the velocity for several values of the surface control ratio B_{ND}. The simulation starts on a 40×40 coarse grid.

The number of time steps is defined to be N_t and the number of calculation points at a grid level as N_{CP}. Figure 8 shows that the multigrid solution presented in Fig. 7e is 80% less expensive in computational nodes and thus also in memory, compared to the equivalent single-grid solution.

3.4. *Validation of the OCLM Method on Two-Dimensional Scalar Interface Advection Tests*

This section examines the OCLM method using several advection tests where no Navier–Stokes solution is required. When the velocity field is defined analytically, the difficulty

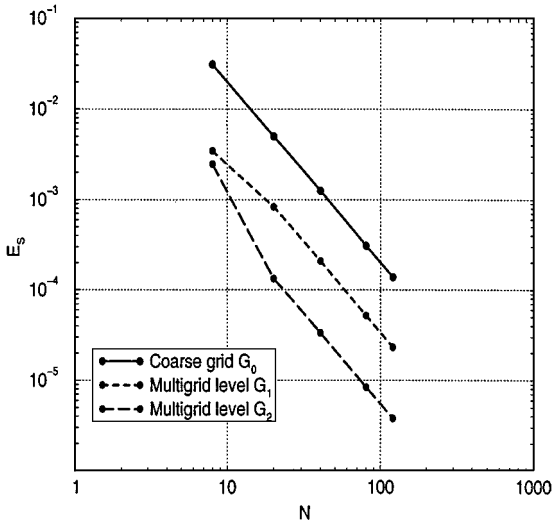

FIG. 6. Convergence study on the volume conservation. E_S is presented for several coarse grids: 8^2, 20^2, 40^2, 80^2, and 120^2.

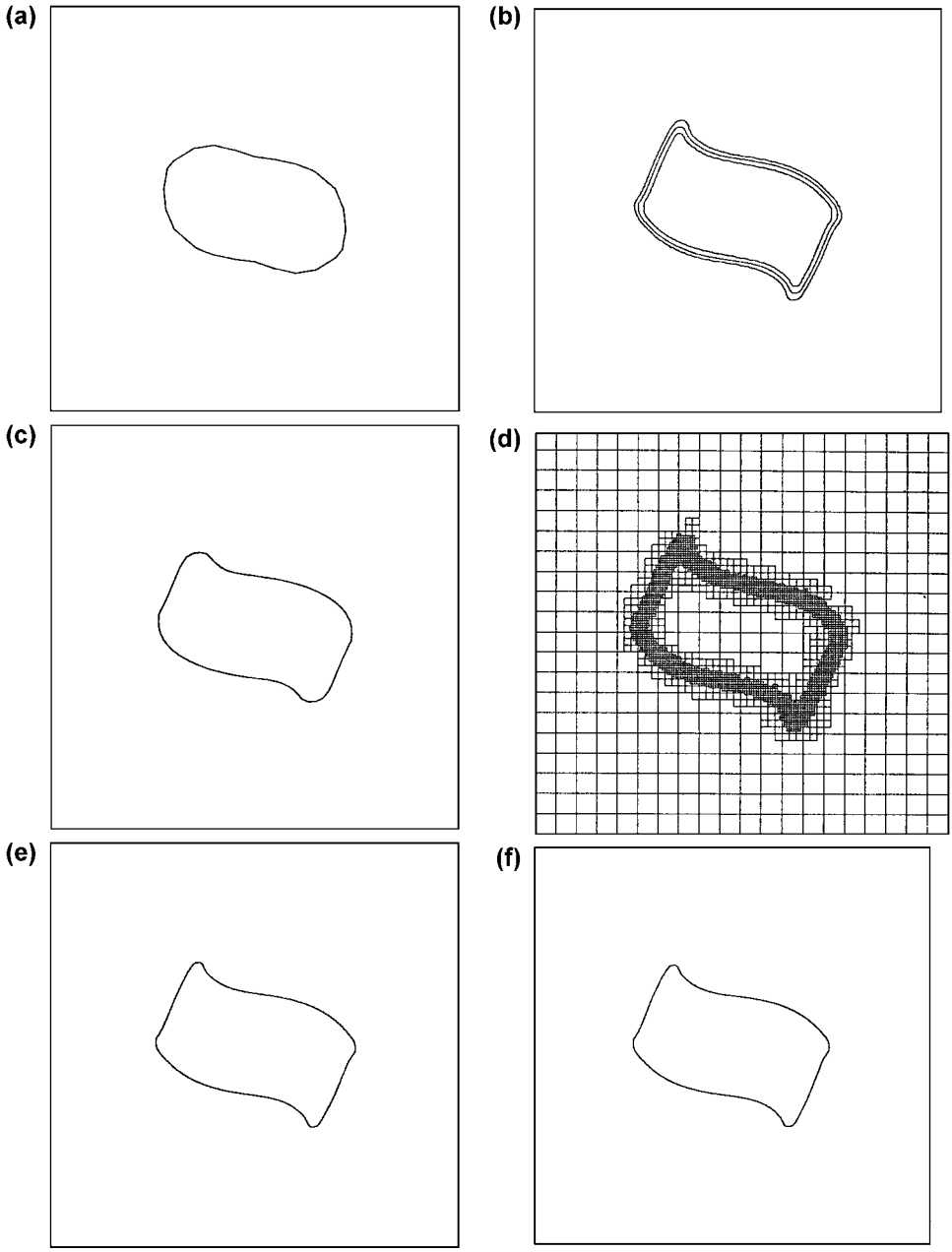

FIG. 7. Multigrid simulation of the self-similar vortex problem. G_0 is a 20×20 grid. A three-level solution is presented after large deformations have been induced for $t = 0.2$ s. (a), (c), and (e) solutions respectively on G_0, G_1, and G_2; (b) diffusion of the multigrid solution on G_2 (contours 0.01, 0.5, and 0.99 are presented); (d) local mesh refinement structure at the final calculation step; and (f) analytical solution.

of the motion equation resolution is avoided and the intrinsic improvements on the interface tracking, provided by the local multigrid treatment, can be highlighted. Two different scalar velocity fields were considered: Zalezak's problem and the vortex flow problem. The domain length L of the square calculation domain is 1 m. In the first test, **u** is defined

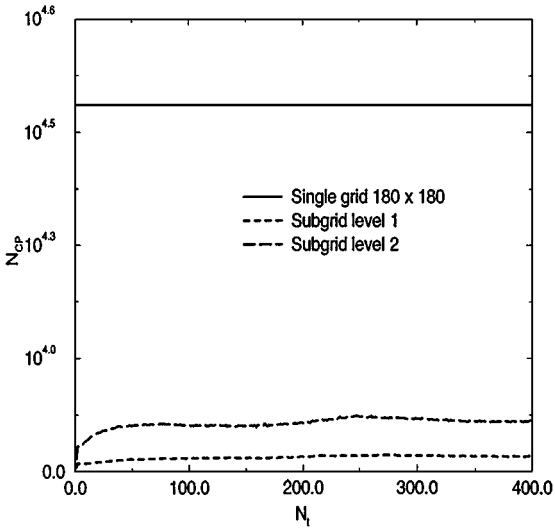

FIG. 8. Number of calculation points at each multigrid level. A 20×20 coarse grid and two multigrid levels are considered.

as

$$u_x = -\frac{\pi}{2}\left(y - \frac{L}{2}\right),$$
$$u_y = \frac{\pi}{2}\left(x - \frac{L}{2}\right), \tag{42}$$

whereas in the second one, the velocity field is chosen to be as in the paper of Rider and Kothe [31],

$$u_x = \cos(4\pi x)\cos(4\pi y),$$
$$u_y = \sin(4\pi x)\sin(4\pi y). \tag{43}$$

The velocity field on the fine grids is deduced from the calculation of the divergence-free solutions (42) and (43) at each fine grid level G_l.

3.4.1. Zalezak's test. A constant turning velocity field advects a slotted circle of a radius of 0.2 m, centred in a l-m-long square calculation domain. With a 30×30 coarse grid G_0 it is clear that the resolution provided by G_0 is too weak to ensure that the problem is precisely solved. Indeed, the circle definition is very approximate, in particular near the sharp corners. At these points, the interface-tracking algorithm requires at least two definition cells to work effectively, unless some diffusion is generated by the TVD scheme. The OCLM method was carried out on two multigrid levels (Fig. 9) to remedy the previous numerical drawbacks. The refinement levels were defined from the beginning of the calculation using criterion (16) and the exact phase repartition was initially enforced on G_1 and G_2, as illustrated in Fig. 9a. The differences among the three phase distributions are quite marked at $t = 0$. The first advantage of the OCLM method is that it takes the initial distribution of C more effectively into account. After one turn of the slotted circle, the interface returns to its initial position. The numerical solution provided by the coarse grid is far removed from the analytical one

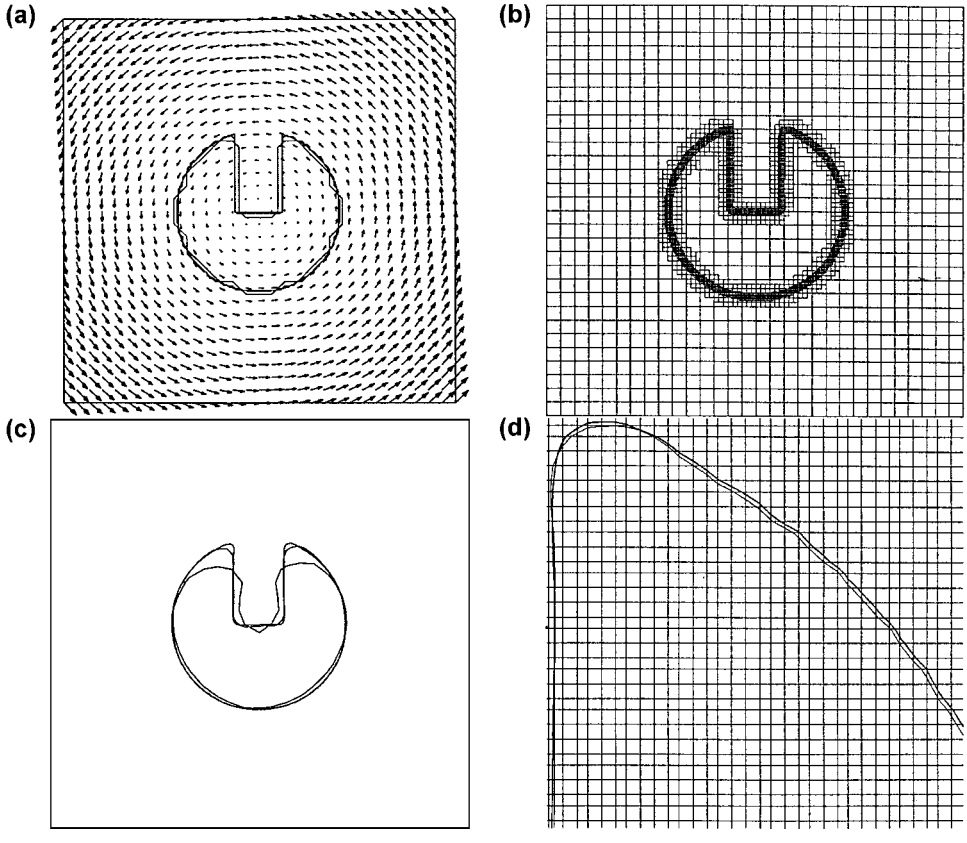

FIG. 9. Multigrid simulation of Zalezak's problem. G_0 is a 30×30 grid. Presentation of a three-level solution. (a) Initial conditions on the different subgrids and velocity field on G_0. (b) Local mesh refinement on the three grid levels after one turn of the slotted circle. (c) Interface position ($C = 0.5$) on each grid level after one turn. (d) Zoom on a sharp corner; visualisation of the OCLM solution on G_2 and of the equivalent single-grid solution on a 270×270 grid.

(Fig. 9c). However, the multigrid solutions on G_1 and G_2 are satisfying. They converge to the exact solution of the problem. If the fine grid solution at level 2 is closely examined, the difference between this solution and the equivalent solution, deduced from a single 270×270 grid calculation, is less than half a fine mesh (Fig. 9d). The local character of the OCLM method is demonstrated in Fig. 9b).

A geometrical improvement ratio R_{GI} is defined as the ratio of the number of calculation points $N_{\mathrm{CP}}^{l_{\max}}$ on the multigrid levels G_0 through $G_{l_{\max}}$ to the number of computation nodes $N_{\mathrm{CP}}^{\mathrm{SG}}$ of an equivalent complete single-grid solution. In the present test (Fig. 10a), the average R_{GI} is equal to 0.28. The best improvement would correspond to R_{GI} being equal to 0, whereas R_{GI} greater than or equal to 1 would represent an increase in the number of computational nodes to be solved. The ratio of the multigrid memory cost Mm to that of the single grid Ms is less than 1 in all the multigrid simulations computed. This represents a significant gain which can be fully exploited in real two-phase flow simulations. Owing to the explicit character of interface tracking and the low time cost of the IT^n algorithm, the multigrid solution proves more expensive in time. As described in Fig. 10b, the ratio of

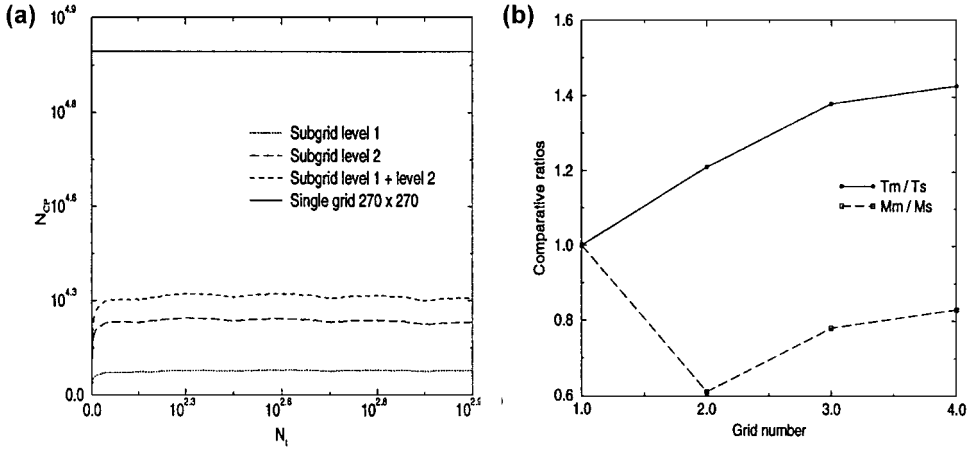

FIG. 10. Comparisons between the multigrid and the single-grid computations of Zalezak's problem. (a) Evolution of the number of calculation points N_{CP} corresponding to a three-grid-level solution. (b) Normalised calculation time and memory costs for several multigrid simulations.

the multigrid computational time Tm to the single grid one Ts is greater than 1 in all the Zalezak's problems solved.

3.4.2. The vortex test. By the same calculation domain as in the previous test, a concentration circle having a radius of 0.15 m, initially centred at the point (0.5, 0.75), was strongly stretched in a complex symmetrical vortex velocity field defined in (43). Outlet boundary conditions were computed on G_0. An almost complete analytical solution of the vortex problem has been calculated by Rider and Kothe [31] using a Marker method with a great number of particles. They demonstrated that the best-performing interface tracking algorithms, such as the level-set technique or the PLIC VOF methods, can only accurately reproduce the larger space scales of the solution. Artificial numerical surface tension makes the VOF method tear the fine scales of the interface, whereas in the level-set technique, the mass loss is very high and the numerical diffusion involves the splitting of the fine-scale features. In the same way a TVD-like method is shown to be poor in representing the fine parts of the interface. On a reasonable single grid, Rider and Kothe explain that only a powerful particle method can deal with strong stretching problems. However, a precise solution is expensive for a scalar test in two dimensions and becomes unrealistic to implement in three dimensions, where the number of particles is very high and the interpolation procedures difficult to carry out.

The OCLM method, thanks to its local and multiscale character, can improve the handling of strongly sheared interfaces. A three-level multigrid solution is presented in Fig. 11. Here G_0 is chosen to be a 70×70 grid. The initial condition and the velocity field on G_0 are presented in Fig. 11a. As shown in Fig. 11b, the resolution provided by the coarse grid is insufficient to accurately solve the problem presented in [31]. In contrast, the multigrid solution on G_1 and G_2 (see Figs. 11d and 11f) successively captures all the features described by the analytical solution of Rider and Kothe. The first multigrid level, G_1, makes the solution more precise in its upper part, but the lower part of the solution is not captured. One additional refinement level, G_2, supplements the numerical solution where very fine structures are captured in its lower part. A comparison between the multigrid solution

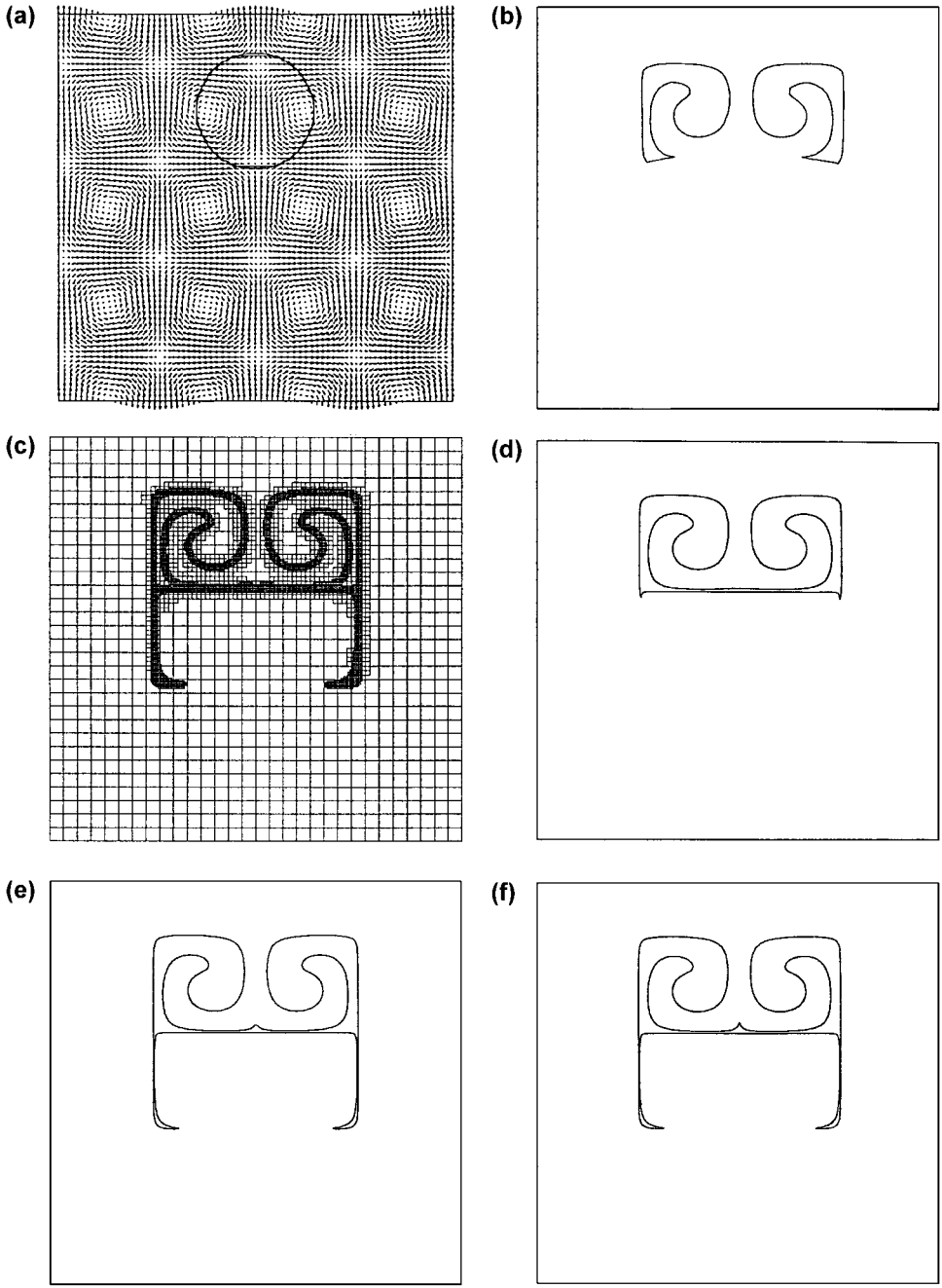

FIG. 11. Multigrid simulation of the vortex problem. G_0 is a 70×70 grid. A three-level solution is presented after large deformations have been induced. (a) Initial conditions on the different subgrids and velocity field on G_0. (b) Interface position ($C = 0.5$) on G_0 after deformation. (c) Local mesh refinement for the final time step (only 40% of the total cells are presented for convenience). (d) Interface position ($C = 0.5$) on G_1 after deformation. (e) Single-grid solution on a 630×630 grid. (f) Interface position ($C = 0.5$) on G_2.

(Fig. 11f) and the equivalent solution on a single 630×630 grid (Fig. 11e) shows almost identical results. The difference between the single-grid and multigrid interface positions is defined as $E_{S,M}$. It is written as a function of linear interpolations of free surface positions on each grid segment which is cut by the interface:

$$\begin{aligned} E_{S,M} &= \max_{i,j}\left(\left|P^0_{x,i,j} - P^{l_{\max}}_{x,3l_{\max}i,3l_{\max}j}\right| + \left|P^0_{y,i,j} - P^{l_{\max}}_{y,3l_{\max}i,3l_{\max}j}\right|\right), \quad \forall l_{\max} > 0 \\ &= \left\|P^0_{x,i,j} - P^{l_{\max}}_{x,3l_{\max}i,3l_{\max}j}\right\|_\infty + \left\|P^0_{y,i,j} - P^{l_{\max}}_{y,3l_{\max}i,3l_{\max}j}\right\|_\infty, \end{aligned} \tag{44}$$

where $P^0_{x,i,j}$ is the linear interpolation of the interface position on the segment (i, j) of the single grid in the x-direction. The superscript $l_{\max}$ refers to the finest grid level in the multigrid architecture, whereas the subscripts x and y refer to the segments slanted in the x- and y-directions.

Figure 12c shows the variations of $E_{S,M}$ according to the number of multigrid levels. The coarse grid and the calculation time are the same for all the solutions in this figure. $E_{S,M}$

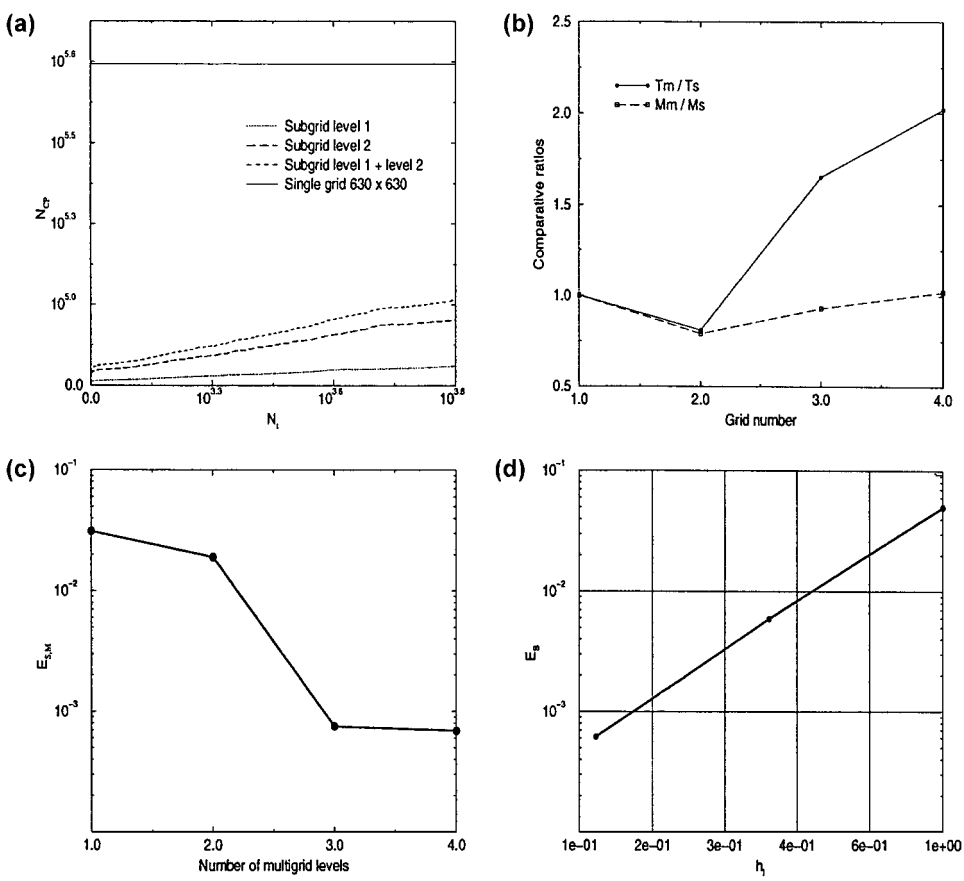

FIG. 12. Comparisons between the multigrid and the single-grid computations of the vortex flow problem presented in Fig. 8. (a) Evolution of the number of calculation points N_{CP} corresponding to a three-grid-level solution. (b) Normalised calculation time and memory costs for several multigrid simulations. (c) Local difference on interface position for several multigrid simulations. (d) Behaviour of the volume conservation E_S according to the nondimensional space scale h_1/h_0, with G_0 being a 70×70 grid.

is high on G_0 and G_1, owing to the poor precision provided by the corresponding grids. However, since the grid level is sufficient, the maximum local difference is very small and the OCLM solution is comparable to the reference computation. The maximum difference between interface positions in single and multigrid solutions is less than half a cell. Beyond a certain number of multigrid levels, weak improvements in $E_{\mathrm{S,M}}$ are obtained owing to the projection step between a coarse and a fine grid level. A higher order projection algorithm must be tested for comparison. However, it would imply a larger discretization stencil and a loss of the local character of the refinement technique.

The convergence of the OCLM method with respect to volume conservation is plotted in Fig. 12d. A second-order convergence rate was obtained where E_{S} was decreased from 5% on G_0 to approximately 0.06% on G_2.

As observed in the previous test and in Fig. 12a, where the average R_{GI} is 0.24, the maximum number of calculation points required in the multigrid solution is four times less than in the equivalent single-grid simulation. However, the memory and the calculation time can be higher with a multigrid calculus than with a single-grid calculus, because of the cost of the local refinement algorithm (interpolation procedures and refinement criterion estimation) in relation to the explicit solution of a scalar problem on G_0.

4. EXAMPLE OF NUMERICAL SIMULATION ON CLASSICAL TWO-PHASE FLOWS

In the following sections, the Navier–Stokes equations (1–5) are solved using the OCLM method. Composite boundary conditions were computed for each problem. According to expressions (37), the surface control parameters $B_{\mathbf{u}}^{s,l}$ were chosen equal to $3/2h_l$. Moreover, the convergence of the motion equation solver were controlled by a divergence threshold ϵ_{NS}, as follows:

$$\sum_{s=1,s_{\max}} \int_{V_c^{l,s}} |\nabla \cdot \mathbf{u}| \, dv < \epsilon_{\mathrm{NS}}. \tag{45}$$

At each grid level G_l, the BiCGSTAB solver iterates until (45) is verified. To satisfy the mass balance with the weakest numerical error, $\epsilon_{\mathrm{NS}} = 10^{-13}$ was chosen.

The gradient of the phase function C was estimated on each side of the pressure control volumes to detect the volumes $V_c^{l,s}$ requiring refining. Then, in keeping with expression (16), a cell was refined if $\mathrm{Cr}_{\mathrm{phys}} \neq 0$. However, this property needs to be numerically verified. A nearly zero interface detection threshold called ϵ_{I} was defined, so that

$$\mathrm{Cr}_{\mathrm{phys}} > \epsilon_{\mathrm{I}}. \tag{46}$$

In this way, a local mesh refinement procedure was implemented on each control volume $V_c^{l,s}$ where (46) was verified.

A conservative procedure of phase function restriction (18) was implemented in all the simulations of two-phase flows presented in this section. In addition an incomplete LU preconditioner ILUD (Chapman *et al.* [7]), which was more powerful than a MILU one but very expensive in calculation time since the grid size increases, was used to solve the reduced linear systems of the multigrid levels G_l, $0 < l < l_{\max}$. The MILU preconditioning is efficient on the coarse grid with suitable time costs. However, its efficiency is insufficient on the 5×5 multigrid calculation domains which are forced by the CBC at the boundaries.

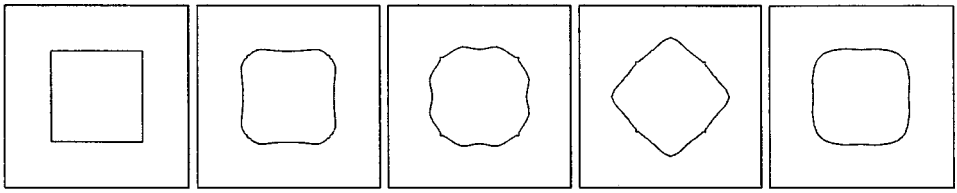

FIG. 13. Multigrid simulation of the oscillation of an initially square drop in zero gravity. The calculation starts with a 30×30 coarse grid G_0 and two local refinement levels. The interface is shown on G_2 at times 0.0, 0.05, 0.1, 0.2, and 1 s from left to right. The amplitude of oscillation is decreasing with time under the action of viscosity in such a way that the drop approaches a spherical equilibrium shape.

4.1. *Nonlinear Oscillation of a Two-Dimensional Drop*

The deformation of a rod, or two-dimensional (2D) drop, induced by unbalanced surface tension forces was then computed. In a zero-gravity field, the surface of the 2D drop is perturbed and the free surface oscillates around its circular equilibrium shape. For our first test, the dynamics of an initially square drop with $\rho_1/\rho_0 = 797.88$, $\mu_1/\mu_0 = 10^2$, and $\sigma = 0.02361\ \mathrm{N \cdot m^{-1}}$ was studied. Two- and three-grid-level simulations were investigated. In Fig. 13, the drop shape on each grid level for $t = 0.0$, $t = 0.05$, $t = 0.1$, $t = 0.2$, and $t = 1.0$ is displayed. A 30×30 coarse grid was chosen in this case with $h_0 = 2.5$ mm. The results correspond closely to Brackbill's [4]. The drop oscillates with a period of approximately 0.4 s. With time, the oscillations become softened by viscous dissipation and the drop shape tends to become circular. The small space scale on G_2 allows a thin description of the interface in the sharp corner zones to be reached.

The second simulation presented in this section is of the oscillation of a cylindrical drop with an initial radius varying as a Legendre polynomial of order 4 (see Lundgren and Mansour [26] and Sussman and Smereka [35]). The initial interfacial position of the 2D drop is defined by the radius

$$R(\theta) = R_0 + \frac{R_0}{100} P_4(\cos(\theta)) \sin(\omega_4 \theta), \tag{47}$$

where

$$\omega_4^2 = \frac{360}{5.04\, R_0^2\, \mathrm{We}}.$$

In Eq. (47), θ is an Euler angle, P_4 is the Legendre polynomial of order 4, and R_0 is the reference radius of the droplet. The density ratio is 100 : 1, the viscosity ratio is 100 : 1, and the surface tension is $0.5\ \mathrm{N \cdot m^{-1}}$. Two and three grid levels were computed with a 50×50 coarse mesh and $h_0 = 0.8$ mm. Figure 14 displays the motion of the viscous drop obtained using two multigrid levels every 0.01 s. Owing to the action of surface tension forces, the perturbed drop oscillates in a nonlinear way with a 0.08-s period. The viscosity action causes the kinetic energy to be dissipated with time. The behaviour of the drop is similar to that computed in axisymmetric problems by Lundgren and Mansour [26] using a boundary integral method and by Sussman and Smereka [35] with a level-set method.

Figures 15(a) and is 15c show that the diffusion of the Lax–Wendroff TVD treatment in the interface capturing step I C^n remains compact on four on five cells even after hundreds of time steps. The local space and time adaptive character of the OCLM method is demonstrated

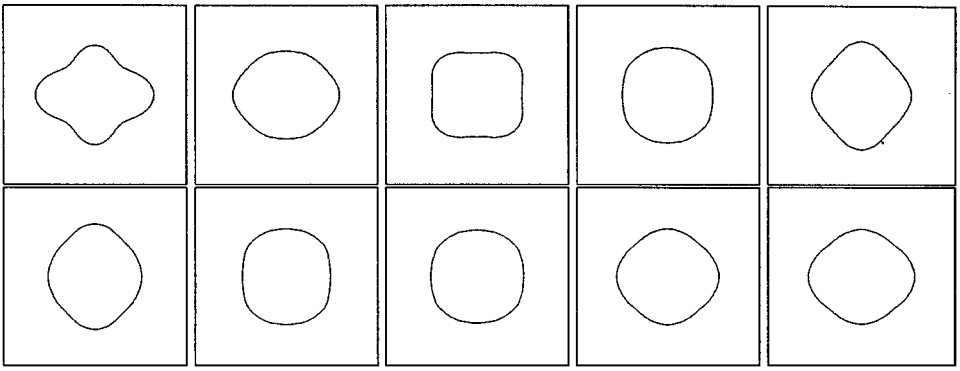

FIG. 14. Multigrid simulation of the dynamics of a 2D viscous drop initialised with a P_4 shape. The computation starts with a 50×50 coarse grid and one multigrid level. The solution is presented each 0.01 s from left to right and from top to bottom.

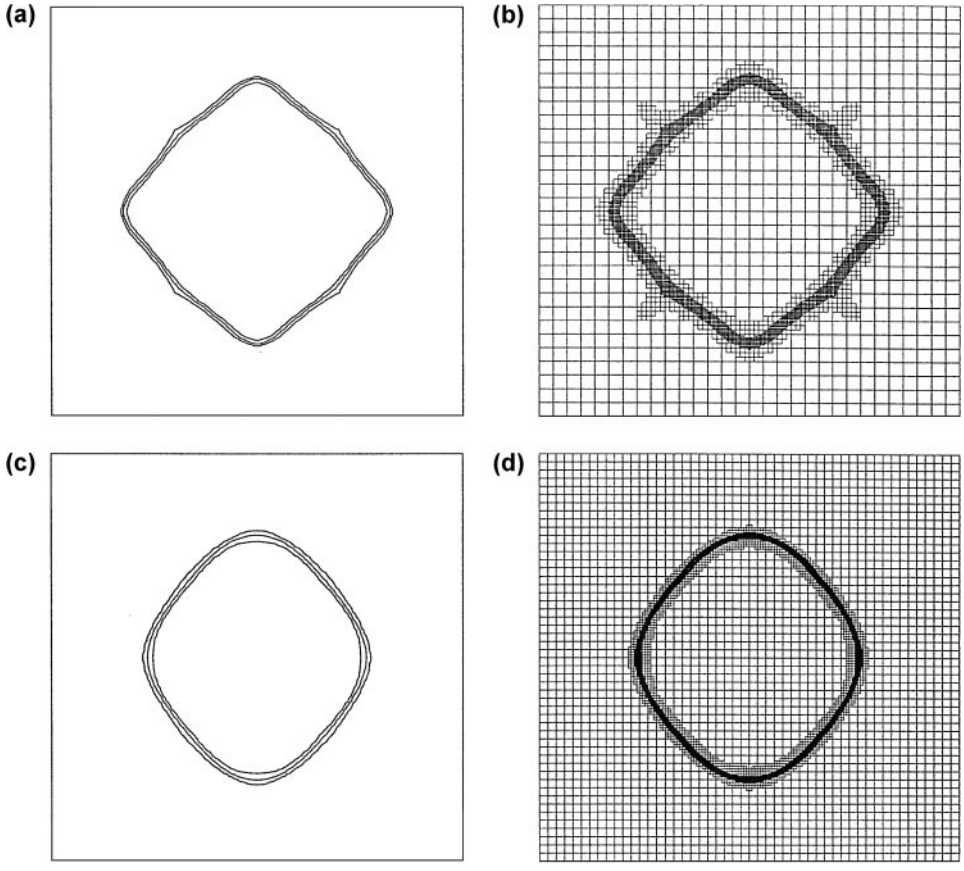

FIG. 15. Drop motion after a half period. The diffusion of the multigrid solution is presented respectively for the first test on G_3 after 0.2 s in (a) and for the second test on G_2 after 0.04 s in (c) (contours 0.01, 0.5, and 0.99 are plotted). The local mesh refinement structure is shown for three-level multigrid simulations of the two nonlinear oscillation drop problems in (b) and (d).

TABLE I
Convergence Study for Nonlinear Drop Oscillation Problems

	Square drop test ($t = 2.0$ s)			Legendre perturbed drop test ($t = 0.09$ s)		
Grid	Surface (cm^2)	E_S (cm^2)	Order	Surface (cm^2)	E_S (cm^2)	Order
G_0	13.9470	0.1130	N/A	3.4401	0.0309	N/A
G_1	14.0472	0.0128	2.0	3.4675	0.0035	2.0
G_2	14.0586	0.0014	2.0	3.4706	0.0004	2.0

Note. The volume conservation E_S is achieved with second-order convergence rate from a coarse grid to a finer one.

in Figs. 15b and 15d. In Table I, convergence results for the two-drop oscillation tests are illustrated when progressively finer grid levels are added. The initial surfaces of the square rod and of the Legrendre perturbed drop are 14.06 cm^2 and 3.471 cm^2. A second-order convergence rate was detected on volume conservation in both problems. At each grid level, the difference $E_{S,M}$ between the multigrid and the single-grid interface position was always less than half a cell. The sensitivity of the solution to the values of $\mathbf{B}_\mathbf{u}^s$ and $\mathbf{B}_\mathbf{u}^v$ is similar to the behaviour observed in Fig. 5. Maximum values of 0.31 and 0.2 were measured respectively in the square rod and in the Legendre perturbed drop tests for the geometrical improvement R_{GI}. In this way, it was verified that the multigrid computation requires 70 to 80% less memory than the single-grid simulation.

The surface tension is immediately taken into account in the OCLM method by means of source terms which are added into the motion equations. The order of magnitude of the parasite currents is not emphasised by the OCLM method.

4.2. *Solving Two-Dimensional Rayleigh–Taylor Instabilities with the OCLM Method*

The Rayleigh–Taylor instability is a classical and widely studied interfacial problem that underlines the competition between the viscous terms and the surface tension force. If we analyse a horizontal fluid stratification, with the heavy fluid lying above the lighter one, any disturbance of the interface between these fluids becomes amplified due to gravity, but the surface tension force tends to counteract this and to minimise the deformation of the free surface. This problem is unstable for any fluid and for any perturbation. It was decided to simulate this problem with the OCLM method in order to emphasise the competitiveness of the local adaptative multigrid architecture in solving multiphase flows, when strong stretching of the free surface occurs.

Three multigrid simulations were carried out on two and three grid levels (Fig. 16). The viscosity is the same in the two fluids. The characteristics of the three problems are the following: We $= +\infty$, $A = 0.11$ and Re $= 200$ in case (a), We $= +\infty$, $A = 0.33$, and Re $= 500$ in case (b), and We $= +\infty$, $A = 0.5$ and Re $= 700$ in case (c). The results presented correspond to all the literature dedicated to Rayleigh–Taylor instabilities (Tryggvason [40] or Unverdi and Tryggvason [41], for example). In the different proposed simulations, the computations consider various Atwood numbers, which correspond to small as well as relatively large differences between ρ_1 and ρ_0. For any Atwood number, non symmetry of the flow appears between the upper and the lower part of the calculation domain, owing to the increased influence of the gravity on the heavy fluid. This phenomenon can be observed

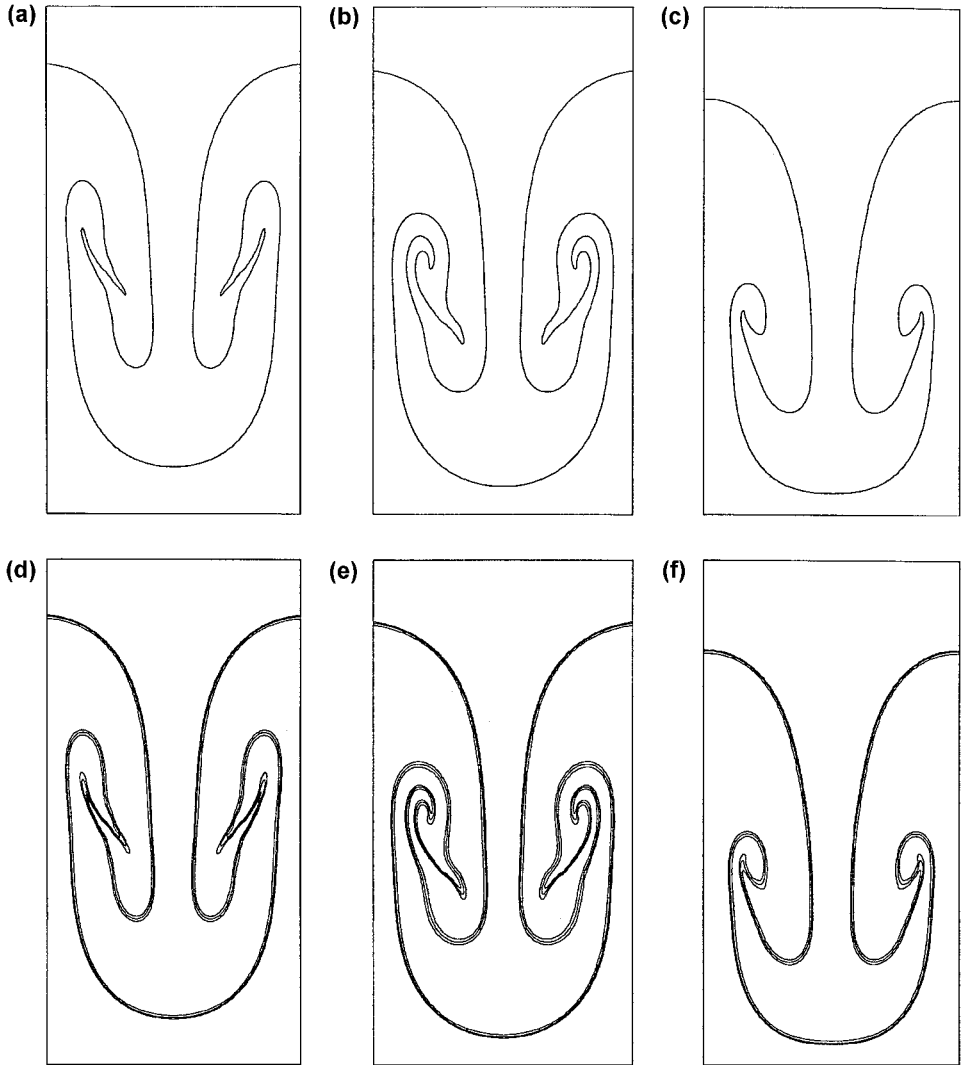

FIG. 16. Numerical simulation of several two-dimensional Rayleigh–Taylor instabilities using the OCLM method with three grid levels. The computations start on a 30×60 coarse grid. The initial perturbations are 10% of the domain height. The viscosities are the same in the two fluids. The results are presented respectively at time 1.5 s for (a) and (d), 1 s for (b) and (e), and 0.75 s for (c) and (f). The interface solution ($C = 0.5$) resulting from a three-grid-level simulation (upper pictures) and the diffusion of the phase function on G_2 (lower graphs) are presented. (a) and (d) We $= +\infty$, $A = 0.11$, and Re $= 200$; (b) and (e) We $= +\infty$, $A = 0.33$, and Re $= 500$; (c) and (f) We $= +\infty$, $A = 0.50$, and Re $= 700$.

in [40, 41]. All the computations presented in this section were carried out on a MIPS R10000 processor with a maximum CPU speed of 180 MHz.

As previously observed in the scalar tests, the local character of the OLCM method is perfectly controlled by means of the physical criterion $\mathrm{Cr}_{\mathrm{phys}}$. The quality of the local grid refinement near the free surface is directly bound to the reduced diffusion of the TVD interface-tracking algorithm (see Figs. 16d and 16e) so that (16) is verified on a small number of cells. The almost discontinuous solution provided by the Lax–Wendroff TVD scheme (7) allows a very local mesh refinement structure concentrated on the interface to

be obtained. The results presented in Fig. 16 (upper part) are almost comparable to the solutions obtained on an equivalent unique grid. For instance, a maximum local error $E_{S,M}$ of 0.00053 m was found between the single-grid and multigrid solutions on G_2.

In local multigrid computations of interfacial problems, the calculation time and the memory costs arise from the solution of the model on a coarse grid, where the memory and the calculation time are more or less constant during the simulations, and from the calculation of the numerical solution on the multigrid levels. If we characterise the repartition of the fine cells obtained with the physical criterion $\mathrm{Cr}_{\mathrm{phys}}$ for several computations presented in this article (Figs. 7, 9, 11, 15, and 16), it is seen that the width of the band of fine cells surrounding the interface represents four or five cells in the whole solution on every multigrid level. As the width of the band of cells wrapping the free surface is not dependent on the coarse grid resolution nor on the calculation time, for a chosen problem, the cost of the OCLM method will only be dependent on the interface length. This assumption can be verified in Zalesak's test (Fig. 9), where the number of multigrid cells is almost constant with time on each multigrid level. As the surface between two states of the phase function is not deformed, the length of the interface remains constant during the computations. Figure 17

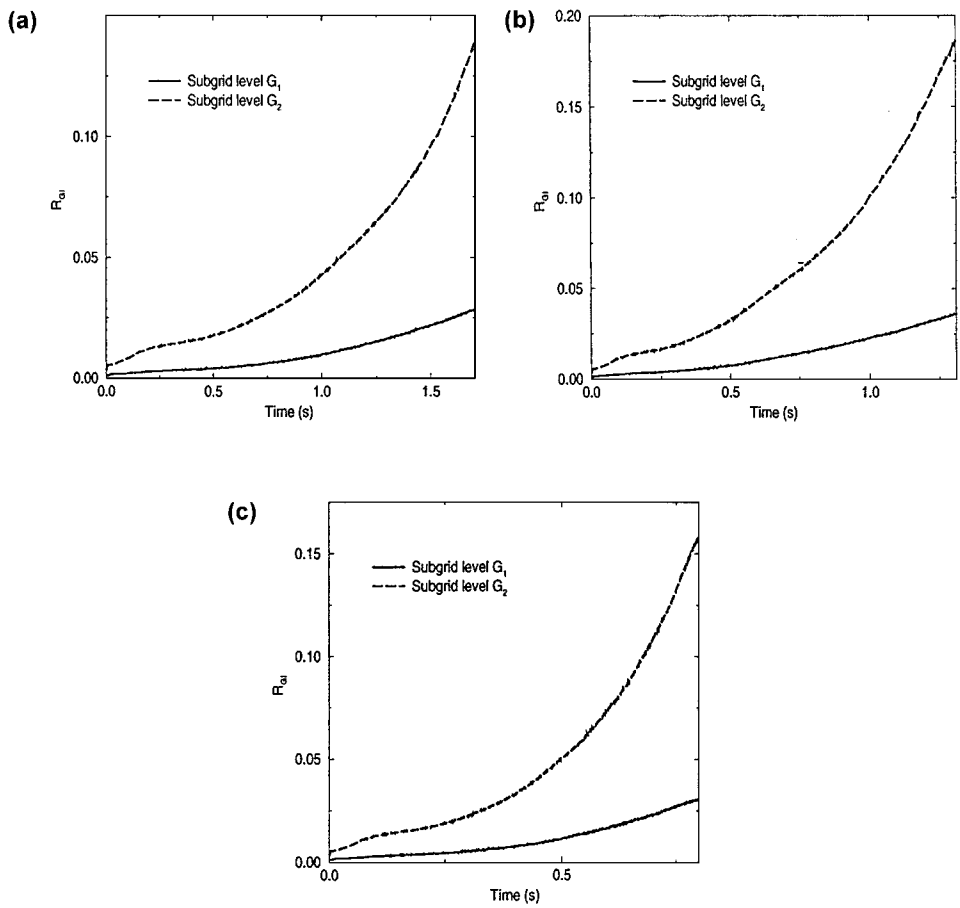

FIG. 17. Evolution of the calculation point rates R_{GI} on the multigrid level $l = 1, 2$. The results correspond to the computations presented in Fig. 16. (a) $A = 0.11$ and $\mathrm{Re} = 200$, (b) $A = 0.33$ and $\mathrm{Re} = 500$, (c) $A = 0.50$ and $\mathrm{Re} = 700$.

describes the evolutions of the calculation point rate R_{GI} with time in the problems presented in Fig. 16. In the first phase, up to $t = 0.1$ s, a rapid increase of the calculation point rate arises from the appearance of the numerical diffusion in the solution, whereas after $t = 0.1$ s, R_{GI} increases regularly. As described previously, this increase is directly due to the increase of the interface length in the Rayleigh–Taylor instability. R_{GI} can be considered as a measure of the free surface stretching. If it is constant, the interface evolutions are slight (approximately the lengthscale of the finer grid level), whereas the deformations are considerable if it increases. In all cases, important gains in memory can be measured (50 to 80%).

To study the sensitivity of the OCLM method in relation to the criterion of interface detection ϵ_I and the convergence threshold of the Navier–Stokes iterative solver $\epsilon_{N,S}$, a representative Rayleigh–Taylor problem ($A = 0.33,\ \mathrm{Re} = 500$) was computed for several values of the numerical parameters ϵ_I and $\epsilon_{N,S}$. A reference numerical solution was calculated on a 270×540 single grid. Several multigrid simulation, corresponding respectively to $10^{-2} \leq \epsilon_I \leq 2 \cdot 10^{-1}$, $10^{-2} \leq \epsilon_{N,S} \leq 10^{-13}$, and $1 \leq l_{max} \leq 2$, were compared to the assumed reference solution.

OCLM method remains slightly dependent on the interface detection threshold ϵ_I (Table II). The difference between the multigrid and the reference numerical solution are almost identical when ϵ_I varies and $0 < \epsilon_I < 0.2$. Even for high values of ϵ_I, 0.2 for example, the obtained solution is satisfying. However, when $\epsilon_I > 0.1$, a small quantity of residual points very near the interface are not dealt with by the OCLM method. These residues can lead to nonnegligible errors in certain cases, when the free surface covers the same region several times during the calculations. The difference E_S between the initial volume and the volume provided by the numerical solution remains unaffected by the value of ϵ_I. Only the number of discretisation points is important to ensure suitable volume conservation.

The convergence threshold of the motion equation solution controls the precision with which the incompressibility constraint is verified. Contrary to ϵ_I, the disparities between solutions can be high when ϵ_{NS} varies (Table II). For small values of ϵ_{NS}, the results obtained are almost identical and the local differences evolve only slightly. However, since $\epsilon_{NS} > 10^{-4}$, $E_{S,M}$ admits nonacceptable values. In this case, relatively important differences were observed between the multigrid solution and the reference one. Owing

TABLE II
Behaviour of Multigrid Solutions According to the Interface Detection Threshold ϵ_I and the Divergence Threshold ϵ_{NS}

Convergence thresholds	l_{max}	Fine grid	Memory (Mo)	Time (h)	$E_{S,M}$ (10^{-3} m)	E_S (10^{-3} m^2)
$\epsilon_I = 10^{-2}$, $\epsilon_{NS} = 10^{-13}$	3	270×540	4.17	63.42	0.50	0.055
$\epsilon_I = 10^{-1}$, $\epsilon_{NS} = 10^{-13}$	3	270×540	4.17	60.94	0.69	0.056
$\epsilon_I = 2 \cdot 10^{-1}$, $\epsilon_{NS} = 10^{-13}$	3	270×540	4.17	58.22	0.86	0.060
$\epsilon_I = 10^{-1}$, $\epsilon_{NS} = 10^{-8}$	3	270×540	4.17	55.73	0.70	0.056
$\epsilon_I = 10^{-1}$, $\epsilon_{NS} = 10^{-2}$	3	270×540	4.17	24.91	1.2	0.10
$\epsilon_I = 10^{-1}$, $\epsilon_{NS} = 10^{-13}$	2	150×300	1.81	15.36	0.47	0.078

Note. Raylaigh–Taylor instabilities corresponding to $A = 0.33$ and $\mathrm{Re} = 500$ are studied. Comparisons are presented on the computational time and memory, the difference $E_{S,M}$ between a single- and a multigrid solution, and the volume conservation E_S.

to the hypothesis of locally isovolume flow, obtaining a satisfying solution near the interface requires defining a very small convergence threshold, leading to an almost perfect divergence-free- velocity field. The error, due to motion equation solving, induces a more or less approximate redistribution of phase function C. For complex multiphase flows, the resulting error on the update of the physical characteristics can lead to unphysical solutions. In the same way, the volume conservation is damaged when ϵ_{NS} is increased.

Starting on a 50×100 grid with one multigrid level offers better results than choosing a 30×60 coarse grid with $l_{max} = 2$, as shown in Table II. On the very coarse 30×60 grid, the interface location is approximate and the resulting Navier–Stokes solution on G_0 is worse than the one on the 50×100 coarse grid. When studying multiphase flow problems, as for all problems of strongly unsteady flows (turbulence for example), a minimum number of meshes is necessary to accurately solve the problem. This observation applies to the multigrid solver. However, several restriction procedures, which are based upon flux conservation properties, are currently being implemented to improve the results on the coarse grid and to limit the dependency of the accuracy of the solution on the fine grid $G_{l_{max}}$ according to the precision of G_0.

In the Rayleigh–Taylor instability considered in Table II ($A = 0.33$ and $\mathrm{Re} = 500$), the calculation point rate logically increases when ϵ_I increases. The behaviour of R_{GI} is similar to that described in Fig. 17. The increase in the number of calculation points when ϵ_{NS} varies ($10^{-2} < \epsilon_{NS} < 10^{-13}$) is not presented in this article because the gaps between the different R_{GI} are less than 1% in these cases.

As observed in the previous sections, a second-order convergence rate of the OCLM method was measured (Table III and Fig. 18) with respect to volume conservation and

TABLE III

Effect of the Grid Size and the Number of Multigrid Levels l_{max} on the Calculation Time, the Memory Costs, the Difference $E_{S,M}$ between a Single- and a Multigrid Simulation, and the Volume Conservation E_S for Three Rayleigh–Taylor Problems

	Problem 1				Problem 2				Problem 3			
	$A = 0.11,\ \mathrm{Re} = 200$				$A = 0.33,\ \mathrm{Re} = 500$				$A = 0.5,\ \mathrm{Re} = 700$			
Number of multigrid levels l_{max}	0	0	1	2	0	0	1	2	0	0	1	2
Resolution of the equivalent single-grid solution in the x-direction	90	270	90	270	90	270	90	270	90	270	90	270
Resolution of the equivalent single-grid solution in the y-direction	180	540	180	540	180	540	180	540	180	540	180	540
Memory (M_0)	3.07	26.02	1.35	4.88	3.07	26.02	1.29	4.17	3.07	26.02	1.20	3.54
Time (h)	3.11	125.94	9.25	72.26	2.12	107.25	7.49	63.42	1.88	94.30	6.81	61.83
$E_{S,M}$ (10^{-3} m)	0	0	2.7	0.33	0	0	2.5	0.30	0	0	2.4	0.27
E_S (10^{-3} m^2)	2.48	0.28	0.52	0.058	2.44	0.27	0.49	0.055	2.42	0.27	0.48	0.053

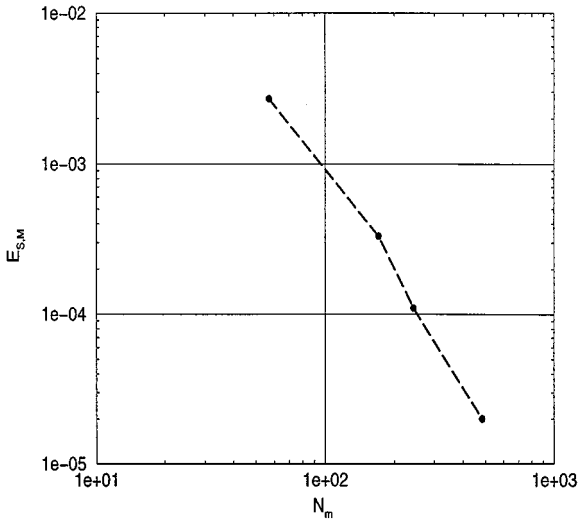

FIG. 18. Evolution of the difference between a multigrid and a single-grid solution $E_{S,M}$ according to the average number of multigrid calculation points N_m. N_m is defined as the square root of the number of multigrid calculation points on G_l, $0 \leq l \leq l_{max}$.

interface position. Table III illustrates a decrease in memory costs of 50 to 85%. The gains in memory are greater when the size of the coarse grid is increased. The calculation time of the OCLM method is four to five times higher than the computational time of a single-grid simulation when N_{CP}^{SG} is weak. In such cases, the multigrid architecture (detection procedure, projection procedure, etc.) hugely penalises the computational time of the method. However, when N_{CP}^{SG} increases, the preconditioning of the linear system, resulting from the discretisation of the motion equations on a single grid, becomes very expensive and the OCLM method, thanks to it local character (5×5 multigrid calculation domains), becomes less costly in calculation time than the single-grid approach.

Finally, the results of the OCLM method are compared to the analytical solution of the instability amplitude evolution proposed by Chandrasekhar [6]. From the linear theory, the amplitude A of the initial perturbation of the free surface is found to increase as

$$A = A_0 \cosh(nt), \tag{48}$$

where A_0 is the amplitude of the initial perturbation,

$$n = \left[\frac{2\pi(\rho_2 - \rho_1)}{\lambda(\rho_2 + \rho_1)}\right]^{1/2},$$

and

$$\lambda = 4\pi\left[\frac{4\mu^2}{g\left(\rho_2^2 + \rho_1^2\right)}\right]^{1/3}.$$

The multigrid solution shows a marked improvement, as presented in Fig. 19. The gap between the solutions is small in comparison to the space scale of the coarse grid. It should be noticed that the differences between the numerical results on G_0 and G_2 quickly stabilise

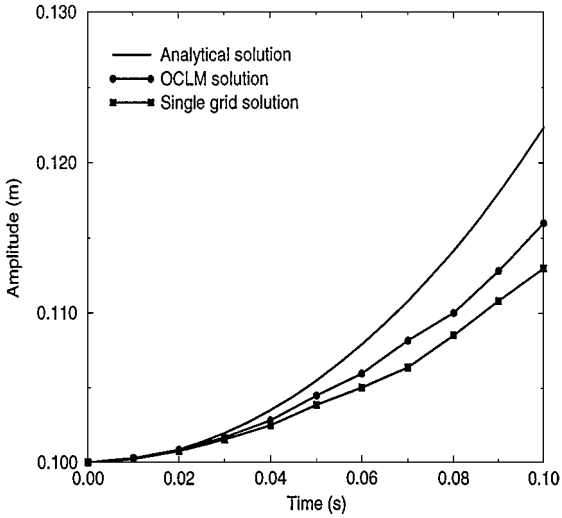

FIG. 19. Amplitude evolution of a Rayleigh–Taylor instability when We $= +\infty$, Re $= 50$, and $A = 0.33$. Comparison among the linear theory, the multigrid solution with $l_{\max} = 2$, and the results on the 20×40 coarse grid.

in time and the various solutions remain, overall, very close, as demonstrated in all the previous Rayleigh–Taylor simulations presented in this article.

4.3. *Droplet Impact on a Liquid Film*

The impact of a cylindrical rod on a liquid film was computed in a box, open in its upper limit. The OLCM method allows us to illustrate the spreading of the 2D drop and the "splash" phenomenon that proceeds this. The simulations were started when the drop came into contact with the liquid film. Initially, a constant impact velocity $\mathbf{u}_0$ was imposed on the rod. The same liquid was considered in the drop and in the film. Let H be the height of the film, L the width of the box, and R the radius of the cylindrical rod. Figure 20 illustrates simulations using $H = 1.6$ mm, $R = 1.2$ mm, $L = 10.66$ mm, and $\mathbf{u}_0 = (0, u_{0,y})$ with $u_{0,y} = 5\ \mathrm{m \cdot s^{-1}}$. Characteristics of $\sigma = 0.075\ \mathrm{N \cdot m^{-1}}$, $\rho_1/\rho_0 = 10^3$, and $\mu_1/\mu_0 = 10^3$ were chosen. Just after the impact, a strong pressure was generated at the impact point which was then transmitted to the bottom of the film. In the initial stages of the spreading of the drop (0.16 ms), the free surface evolution was unsteady (see Fig. 20) and drops were ejected from the liquid film to the top of the cavity. Then, the liquid rising from the bottom of the film created a nonlinear wave at the periphery of the spreading drop (0.32 ms). Finally, the splash phenomenon was characterised by the formation of a liquid lamella (1.04 ms) that spreads from the impact point to the edge. This is due to the correlated action of the strong inertia arising from the dynamics of impact and the motionless liquid near the boundaries. The coupling between an LWT interface-tracking method and a local mesh refinement method describes the merging of the drop with the liquid film and the ejection of small liquid drops. In spite of the diffusion appearing in the shearing zone of the interface (Fig. 21) at the tip of the lamella, a second-order convergence was measured in this problem (Table IV). A comparison between the solution proposed in Fig. 20 and the results computed with an equivalent single grid shows differences more or less equal to the fine grid cell. Indeed, after 1.04 ms, $E_{\mathrm{S,M}}$ is equal to 3.4310^{-5} m. The geometrical improvement ratio is

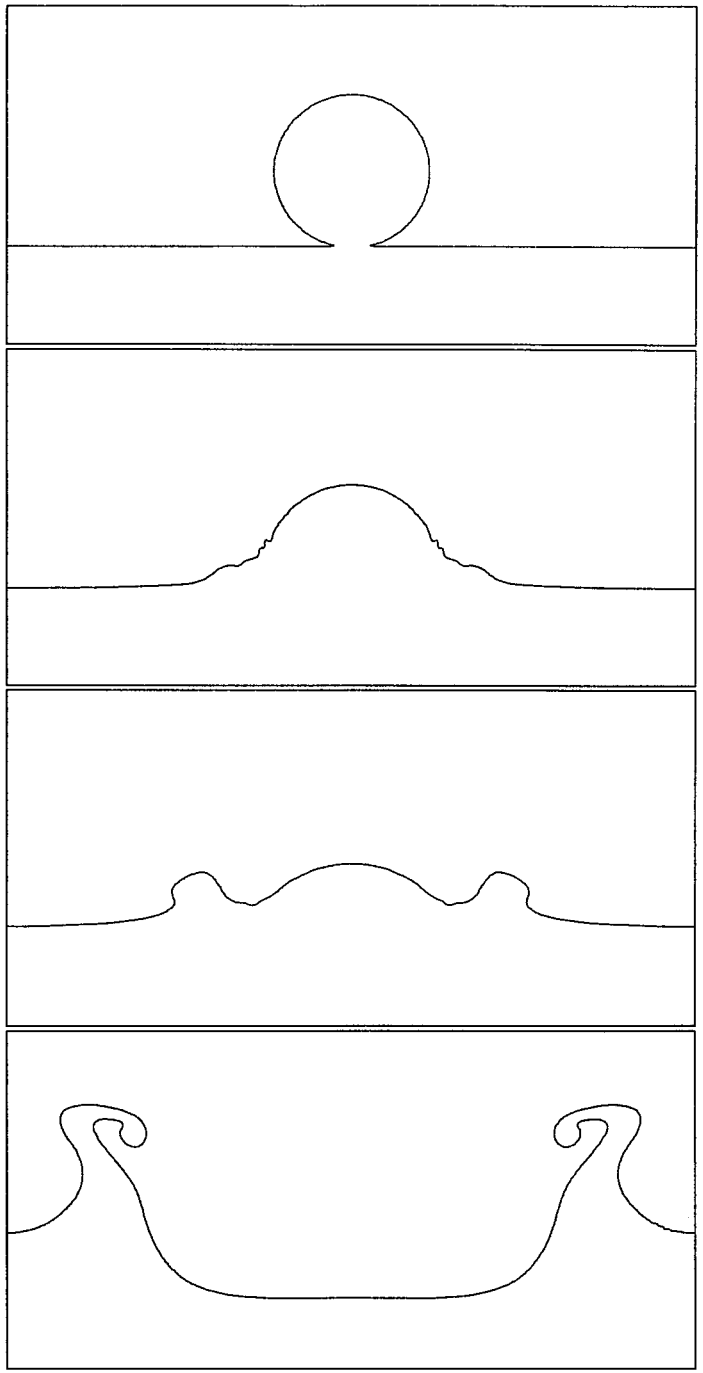

FIG. 20. Numerical simulation of droplet impact on a liquid film. Two grid levels are computed with a 100×50 coarse grid. The viscosity and density ratios are equal to 10^3. The results are presented at times 0.0, 0.16, 0.32, and 1.04 ms.

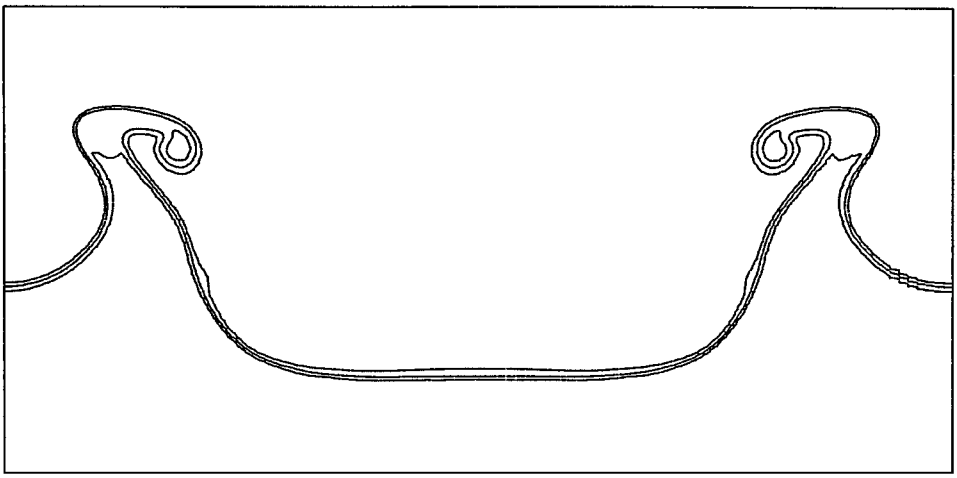

FIG. 21. Diffusion of the phase function 1.04 ms after impact. Isolines 0.05, 0.5, and 0.95 are presented. Artificial diffusion is generated at the tip of the spreading lamella, where the shearing is important.

not presented for the drop impact problem because the same bahaviour as in the previous sections was observed. A maximum value of 0.12 was measured for R_{GI}.

A comparison with the results of Yarin and Weiss [51] on ethanol drops was carried out. Let C_c be the nondimensional crown radius, R_c the nondimensional rim elevation of the lamella, and H_c the nondimensional bump radius. These characteristic variables of the flow are made nondimensional thanks to the initial radius of the drop. Correlation appears between the observations of Yarin and Weiss and our results: the crown radius C_c evolves as a square root function of time, the rim elevation R_c is linearly dependent on the radius C_c of the lamella, and the difference $(H_c - H)$ evolves linearly with time.

5. CONCLUSIONS

An original one-cell local multigrid (OCLM) method for computing two-dimensional unsteady and incompressible multiphase flows has been presented. An implicit augmented Lagrangian method has been used to solve the coupling between pressure and velocity as well as the incompressibility constraint. The motion of the interface has been modelled by solving a conservation law for a phase function by means of a Lax-Wendroff TVD scheme. The local character of the adaptative mesh refinement method has been used to focus the computations on cells surrounding the interface. The Navier–Stokes equations have been successfully solved using the OCLM method (see Fig. 22) thanks to the development

TABLE IV
Convergence Study for Drop Impact on a Liquid Film

Grid	Surface (cm^2)	E_S (cm^2)	Order
G_0	0.2027	0.0123	N/A
G_1	0.21364	0.00136	2.0
G_2	0.21483	0.00017	2.0

Note. The volume conservation E_S is studied for a three-level multigrid solution.

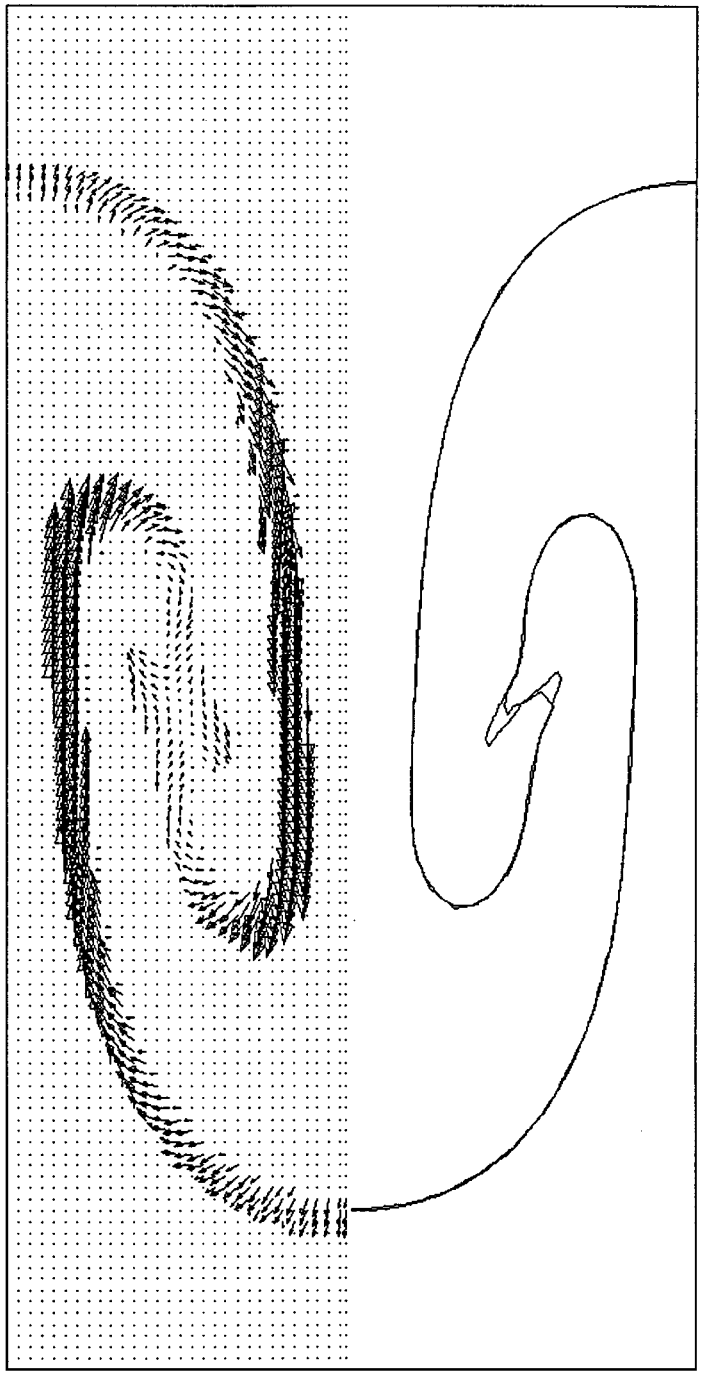

FIG. 22. Numerical simulation of a Rayleigh–Taylor instability using the OCLM method with $l_{\max} = 1$. The characteristics of the problem are the following: We $= +\infty$, $A = 0.05$, and Re $= 10$. G_0 is a 40×80 grid. The left side of the picture presents the velocity field on G_1 and the right side the comparison between the solutions on G_0 and G_1. The fine-grid solution is the most stretched one.

of new composite boundary conditions (CBC). The multigrid method has been validated against several two-dimensional simulations including the vortex test, bubble oscillations, Rayleigh–Taylor instability, and the droplet impact on liquid film. Convergence, memory, and time performance have been estimated in all the problems solved.

Several objectives have been reached:

- The local mesh refinement method can track a free surface on several grid levels maintaining the refined control volumes over a maximum of four or five cells around the interface.
- The ratio between the number of multigrid and equivalent single-grid calculation points is always less than 25%. Memory costs have been decreased by 50 to 80% in real two-phase flow simulations.
- The OCLM solution requires less computational time than the comparable single-grid solution as soon as the problem becomes complex in terms of mesh size and interface deformations.
- The difference between the multigrid and the single-grid solutions and the conservation of the fluid volumes converge with second-order rates.

Future research and further developments are planned on the OCLM method:

- The methodology presented in this article is currently undergoing extension to three-dimensions. Problems requiring large three-dimensional grids, such as the droplet impact on liquid film or the viscous liquid jet flattening under three-dimensional instability [49], are being solved with the OCLM method. On modern computers, it is expected that grids as large as 500^3 or 1000^3 can be reached. Implementing dynamic memory allocation is necessary to maximize the benefits from the OCLM method. In the present work, a FORTRAN 90 program was developed.
- Work still has to be carried out to improve the coupling between the fine and the coarse grids. The development of the CBC proposed in this article could certainly be completed by taking into account the surface tension and the two-phase character of the flow in the viscous stress tensor. Moreover, assuming the OCLM method to be a discretisation on a globally unstructured mesh (locally orthogonal Cartesian grid), we are currently generalising this method to a fully implicit solver. In this way, all the grids are solved at the same time by linking unknown variables in a unique linear system.
- The OCLM method is a universal multigrid algorithm, applicable to single fluid as well as multiphase flow problems. The scale changes can lead to solving different models according to the scales of the phenomena, or to computing different numerical solvers. One can imagine solving the problem on the coarse grid with a preconditioned iterative solver, whereas a direct resolution method such as the LU one would be implemented on the fine grids, where the size of the linear system is reduced. In the same way, without any particular work, a Marker technique, a PLIC VOF method, or a level-set approach could be used in the OCLM method to track the interface, whereas a projection method could be implemented to solve the motion equation.
- The linear systems are solved independently on the different multigrid calculation domains, as the coupling between each fine grid is provided by the interpolation and the restriction operators. A parallelisation of the OCLM method will be carried out in the near future. We can expect the calculation time to be significantly reduced. Moreover, thanks to the reasonable memory and calculation costs of the multigrid method, bigger coarse grids could be chosen in three dimensions while a suitable calculation time was maintained.

APPENDIX A

Q1 Interpolation Operator

To initialise a scalar or a vectorial unknown variable on a subgrid $G_{l,s}$, an interpolation operator $P_{l-1,l}$ is carried out between a coarse grid G_{l-1} and a fine grid G_l. If $\mathrm{Cr}_{\mathrm{phys}} > 0$ at point $(i\Delta x, j\Delta y) \in G_{l-1}$, a Q1 polynomial interpolation, associated to the extension of a scalar field ϕ on the control volume cell (i, j) (see Fig. 23), is developed according to the coarse grid values of ϕ as

$$\begin{aligned}\phi_{i_l,i_l} &= P_{l-1,l}(\phi^{l-1}, I, J)\\ &= \left[1 - \frac{\mathrm{Inc}_x}{R_{\mathrm{ef}}} - \frac{\mathrm{Inc}_y}{R_{\mathrm{ef}}} + \frac{\mathrm{Inc}_x \cdot \mathrm{Inc}_y}{R_{\mathrm{ef}}^2}\right]\phi_{\mathrm{I,J}}^{l-1} + \left[\frac{\mathrm{Inc}_x}{R_{\mathrm{ef}}} - \frac{\mathrm{Inc}_x \cdot \mathrm{Inc}_y}{R_{\mathrm{ef}}^2}\right]\phi_{I+1,J}^{i-1}\\ &\quad + \left[\frac{\mathrm{Inc}_y}{R_{\mathrm{ef}}} - \frac{\mathrm{Inc}_x \cdot \mathrm{Inc}_y}{R_{\mathrm{ef}}^2}\right]\phi_{I,J+1}^{l-1} + \left[\frac{\mathrm{Inc}_x \cdot \mathrm{Inc}_y}{R_{\mathrm{ef}}^2}\right]\phi_{I+1,J+1}^{l-1}, \end{aligned} \tag{49}$$

where i_l and j_l are indices on the fine grid G_l, I and J are indices on the coarse grid G_{l-1}, ϕ^{l-1} is the scalar variable on G_{l-1}, Inc_x and Inc_y are position indices in the refined cell, R_{ef} is the odd refinement rate, and ϕ_{i_l,j_l} is the interpolated function on G_l. In the present article, R_{ef} is chosen to be 3.

The refinement criterion $\mathrm{Cr}_{\mathrm{phys}}$ defined in (16) is only estimated at the physical nodes. Consequently, as a staggered MAC grid is implemented to discretize the equation system, the statement of the position indices Inc_x and Inc_y is different according to whether we consider a scalar or a vector unknown variable (Fig. 23). The general expression for these indices is

$$\begin{cases}\mathrm{Inc}_x = i_l - i\,R_{\mathrm{ef}} + S_x,\\ \mathrm{Inc}_y = j_l - j\,R_{\mathrm{ef}} + S_y,\end{cases} \tag{50}$$

where S_x and S_y are gap indices due to the MAC grid. For example, to interpolate C or p with (50), $S_x = 0$ and $S_y = 0$. However, if $\mathbf{u}$ is expended from G_{l-1} to G_l, the indices

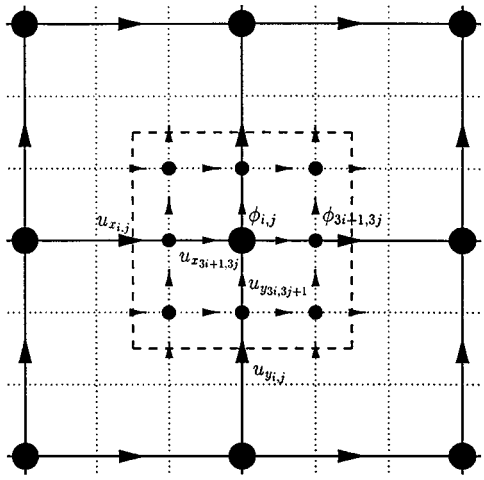

FIG. 23. Physical variable position and indices on a refined cell.

TABLE V
Interpolation Indices in Each Space Direction and Grid Definition for a Scalar and a Vectorial Variable

Refinement of G_{l-1} at point (i, j)	
Scalar variable	
Interpolation on a 3×3 grid	
x-direction	Nodes $3i-1$ to $3i+1$
y-direction	Nodes $3j-1$ to $3j+1$
Vectorial variable	
Interpolation of the first component on a 4×3 grid	
x-direction	Nodes $3i$ to $3i+4$
y-direction	Nodes $3j$ to $3j+3$
Interpolation of the second component on a 3×4 grid	
x-direction	Nodes $3i$ to $3i+3$
y-direction	Nodes $3j$ to $3j+4$

corresponding to its first component u_x are $S_x = 1$ and $S_y = 0$, whereas the ones dedicated to the second component u_y are $S_x = 0$ and $S_y = 1$. When the point (i, j) is detected to be refined on G_{l-1}, a scalar field and a vector field are prolongated on the same refinement cell but they are not interpolated on the same grid, as described on Fig. 23.

Table V provides the extension indices in each direction on the fine grid. According to the type of variable considered, the resulting grid dependency of $P_{l-1,l}$ on the coarse values is different. Nine coarse points are needed to interpolate a scalar field ϕ,

$$\phi_{I,J} \begin{cases} I = i-1, i+1, \\ J = j-1, j+1. \end{cases}$$

For a vector field, six points are expected for each component. For example, the interpolation operator on the first component of $\mathbf{u}$ needs the coarse terms

$$u_{x_{I,J}} \begin{cases} I = i, i+1, \\ J = j-1, j+1, \end{cases}$$

whereas the second component requires the use of

$$u_{y_{I,J}} \begin{cases} I = i-1, i+1, \\ J = j, j+1. \end{cases}$$

A generic interpolation algorithm is deduced from the previous expressions and comments as detailed in (51). The parameters S_x and S_y allow us to apply the same extension procedure to any detected point and to any variable. They ensure the gap of the loop indices in the interpolation routine

$$\begin{bmatrix} \text{For } \mathrm{Inc}_x = (-1+S_x) \text{ to } (1+2S_x) \\ \begin{bmatrix} \text{For } \mathrm{Inc}_y = (-1+S_y) \text{ to } (1+2S_y) \\ P_{l-1,l}\big(\phi^{l-1}, \mathrm{INT}\big(\frac{R_{\mathrm{ef}} \cdot i_{l-1} + \mathrm{Inc}_x}{R_{\mathrm{ef}}}\big), \mathrm{INT}\big(\frac{R_{\mathrm{ef}} \cdot j_{l-1} + \mathrm{Inc}_y}{R_{\mathrm{ef}}}\big)\big), \end{bmatrix} \end{bmatrix} \tag{51}$$

where i_{l-1} and j_{l-1} correspond respectively to the coordinates i and j of a coarse point to be refined and INT is the integer part of a real number.

APPENDIX B

Divergence of the Interpolated Velocity Field

We consider in this section a pressure control volume $V_c^{l-1,s}$ of G_{l-1} which is detected by the refinement criterion $\mathrm{Cr}_{\mathrm{phys}}$. It is supposed that the solution $(\mathbf{u}^{l-1,n+1}, p^{l-1,n+1})$ on the multigrid level $l-1$ has been calculated by the solver $\mathrm{AL}^{n+1} - \mathrm{IC}^{n+1}$. The local mesh refinement procedure generates a 3×3 grid $G_{l,s'}$ corresponding to the cutting of $V_c^{l-1,s}$. $(\mathbf{u}^{l-1,n+1}, p^{l-1,n+1})$ is interpolated on $G_{l,s'}$ with the Q1 interpolation operator presented in appendix A. We propose to calculate the sum of the divergence of the interpolated velocity field $\mathbf{u}^{l,n+1}$ on $G_{l,s'}$. A discrete coordinate system is introduced on $G_{l,s'}$ to locate each component of the velocity field on the 3×3 MAC grid: u_x is characterised by the indices i_x and j_x whereas i_y and j_y describe u_y. The staggered grid induces the following variation gap of indexes (see Fig. 24):

$$\begin{aligned} &1 \le i_x \le 4 \text{ and } 1 \le j_x \le 3, \\ &1 \le i_y \le 3 \text{ and } 1 \le j_y \le 4. \end{aligned} \tag{52}$$

$\mathbf{u}^{l-1,n+1}$ satisfies the divergence-free property with a nearly computer error. In this way, $\nabla \cdot \mathbf{u}^{l-1,n+1}$ is equal to 0 on $V_c^{l-1,s}$. In discretized form one obtains

$$\frac{u^{l-1}_{x_{i+1,j}} - u^{l-1}_{x_{i,j}}}{h_{l-1}} + \frac{u^{l-1}_{y_{i,j+1}} - u^{l-1}_{y_{i,j}}}{h_{l-1}} = 0, \tag{53}$$

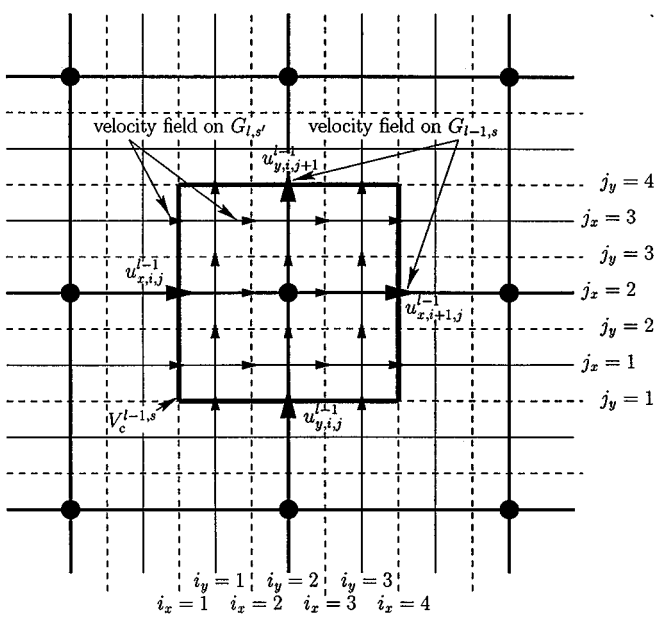

FIG. 24. Description of the discrete distribution of the velocity field on a coarse control volume $V_c^{l-1,s}$ detected to be refined and on the corresponding fine grids $G_{l,s'}$. Indices i_x, i_y, j_x, and j_y are introduced to discretize $\mathbf{u}^{l,n+1}$.

where $u^{l-1}_{x_{i+1,j}}$, $u^{l-1}_{x_{i,j}}$, $u^{l-1}_{y_{i,j+1}}$, and $u^{l-1}_{y_{i,j}}$ are respectively the horizontal and vertical components of $\mathbf{u}^{l-1,n+1}$ on $V_c^{l-1,s}$ (see Fig. 24). Let $S^{l,s'}_{\text{div}}$ be the sum of the discrete divergence on $G_{l,s'}$. One then obtains the expression

$$S^{l,s'}_{\text{div}} = \sum_{i_x=1}^{3}\sum_{j_x=1}^{3} \frac{u_{x_{i_x+1,j_x}} - u_{x_{i_x,j_x}}}{h_l} + \sum_{i_y=1}^{3}\sum_{j_y=1}^{3} \frac{u_{y_{i_y,j_y+1}} - u_{y_{i_y,j_y}}}{h_l}. \tag{54}$$

Substituting the values of the velocities deduced from the interpolation procedure, one can write expression (54) as

$$S^{l,s'}_{\text{div}} = \frac{7}{3}\left(\frac{u^{l-1}_{x_{i+1,j}} - u^{l-1}_{x_{i,j}}}{h_l} + \frac{u^{l-1}_{y_{i,j+1}} - u^{l-1}_{y_{i,j}}}{h_l}\right) + \left(\frac{u^{l-1}_{x_{i+1,j-1}} - u^{l-1}_{x_{i,j-1}} + u^{l-1}_{x_{i+1,j+1}} - u^{l-1}_{x_{i,j+1}}}{3h_l} + \frac{u^{l-1}_{y_{i-1,j+1}} - u^{l-1}_{x_{i-1,j}} + u^{l-1}_{y_{i+1,j+1}} - u^{l-1}_{y_{i+1,j}}}{3h_l}\right). \tag{55}$$

Taking into account the incompressibility of u^{l-1} on $V_c^{l-1,s}$ through expression (53) in (55), we show that $S^{l,s'}_{\text{div}} \neq 0$. Indeed, we have

$$\begin{aligned} A &= \frac{u^{l-1}_{x_{i+1,j-1}} - u^{l-1}_{x_{i,j-1}} + u^{l-1}_{x_{i+1,j+1}} - u^{l-1}_{x_{i,j+1}}}{h_{l-1}} \neq 0, \\ B &= \frac{u^{l-1}_{y_{i-1,j+1}} - u^{l-1}_{x_{i-1,j}} + u^{l-1}_{y_{i+1,j+1}} - u^{l-1}_{y_{i+1,j}}}{h_{l-1}} \neq 0. \end{aligned} \tag{56}$$

We thus have demonstrated that the interpolated velocity field does not verify the divergence-free property on $G_{l,s'}$. The error induced by the interpolation procedure on the incompressibility constraint is directly obtained from (56) as the sum of A and B.

REFERENCES

1. P. Angot, J.-P. Caltagirone, and K. Khadra, Une méthode adaptative de raffinement local: La correction du flux à l'interface, *C. R. Acad. Sci. Ser. II b*, 315:739–745, 1992.
2. P. Angot, *Contribution à l'étude des transferts thermiques dans des systèmes complexes: Application aux composants électroniques* (Ph.D. thesis, University of Bordeaux I, 1989).
3. M. J. Berger and P. Collela, Local adaptative mesh refinement for hyperbolic partial differential equations, *J. Comput. Phys.* **82**, 64 (1989).
4. J. U. Brackbill, B. D. Kothe, and C. Zemach, A continuum method for modeling surface tension, *J. Comput. Phys.* **100**, 335 (1992).
5. J.-P. Caltagirone, K. Khadra, and P. Angot, On a local multigrid mesh refinement method for solving Navier–Stokes equations, *C. R. Acad. Sci. Ser. IIb* **320**, 295 (1995).
6. S. Chandrasekhar, *Hydrodynamic and Hydromagnetic Stability*, Oxford Univ. Press, Oxford, 1961.
7. A. Chapman, Y. Saad, and L. Wigton, High order ILU preconditioners for CFD problems, AMSI Technical Report, 1996.
8. B. J. Daly, Numerical study of two-fluid Rayleigh–Taylor instability, *Phys. Fluids* **10**, 297 (1967).
9. B. J. Daly, Numerical study of the effect of surface tension on interface instability, *Phys. Fluids* **12**, 1340 (1969).
10. A. Elgowainy and N. Ashgriz, The Rayleigh–Taylor instability of viscous fluid layers, *Phys. Fluids* **9**(6), 1635 (1997).

11. M. Fortin and R. Glowinsky, *Méthodes de Lagrangien Augmenté. Application à la Résolution Numérique de Problèmes aux Limites* (Dunod, Paris 1982).
12. D. Gueyffier and S. Zaleski, Formation de digitations lors de l'impact d'une goutte sur un film liquide, *C. R. Acad. Sci. Ser. IIb* **326**, 839 (1998).
13. D. Gueyffier, *Etude de l'impact de gouttes sur un film liquide mince. Développement de la corolle et formation de projections* (Ph.D. thesis, University of Paris VI, 2000).
14. I. Gustafsson, *On First and Second Order Symmetric Factorization Methods for the Solution of Elliptic Difference Equations* (Chalmers University of Technology, 1978).
15. W. Hackbush, Multi-Grid Methods and Applications, in Series in Computational Mathematics (Springer-Verlag, Berlin, 1985), Vol. 4.
16. F. H. Harlow and J. E. Welsh, Numerical calculation of time dependent viscous incompressible flow, *Phys. Fluids* **8**, 21 (1965).
17. C. Hirsch, in *Numerical Computation of Internal and External Flows*, Wiley–Interscience, New York, 1990, Vol. 2, p. 493.
18. C. W. Hirt and B. D. Nichols, Volume of fluid (vof) methods for the dynamics of free boundaries, *J. Comput. Phys.* **39**, 201 (1981).
19. K. Khadra, P. Angot, J.-P. Caltagirone, and P. Morel, Concept de zoom adaptatif en architecture multigrille locale: Étude comparative des méthodes L.D.C., F.A.C. et F.I.C., *Math. Model. Numer. Anal.* **30**, 39 (1996).
20. B. Lafaurie, C. Nardone, R. Scardovelli, S. Zaleski, and G. Zanetti, Modelling merging and fragmentation in multiphase flows with SURFER, *J. Comput. Phys.* **113**, 134 (1994).
21. M. Laugier, P. Angot, and L. Mortier, Nested grid methods for ocean model: a comparative study, *Int. J. Numer. Methods Fluids* **23**, 1163 (1996).
22. B. P. Leonard, A stable and accurate convective modelling procedure based on quadratic upstream interpolation, *Comp. Methods Appl. Mech. Eng.* **19**, 59 (1979).
23. R. J. LeVeque, Numerical Methods for Conservation Laws, O. E. Landford, editor, in Lectures in Mathematics, edited by Lanford (Birkhauser, Zurich, 1992), p. 173.
24. J. Li, Piecewise linear interface calculation, *C. R. Acad. Sci. Ser. IIb* **320**, 391 (1995).
25. N. Lock, M. Jeager, M. Medale, and R. Occelli, Local mesh adaptation technique for front tracking problem, *Int. J. Numer. Methods Fluids* **28**, 719 (1998).
26. T. S. Lundgren and N. N. Mansour, Oscillations of drops in zero gravity with weak viscous effects, *J. Fluid Mech.* **194**, 479 (1988).
27. J. Magnaudet, M. Rivero, and J. Favre, Accelerated flows around a rigid sphere or a spherical bubble, *J. Fluid Mech.* **284**, 97 (1995).
28. X. Nicolas, P. Traore, A. Mojtabi, and J.-P. Caltagirone, Augmented Lagrangian method and open boundary conditions in 2D simulation of Poiseuille–Bénard channel flow, *Int. J. Numer. Methods Fluids* **25**, 265 (1997).
29. R. Peyret and T. D. Taylor, eds., *Computational Methods for Fluid Flow*, Springer Series in Computational Physics (Springer-Verlag, New York, 1993).
30. S. Popinet and S. Zaleski, A front-tracking algorithm for accurate representation of surface tension, *Int. J. Numer. Methods Fluids* **30**, 775 (1999).
31. W. J. Rider and D. B. Kothe, Stretching and tearing interface tracking methods, AIAA Paper 95–1717, 1995.
32. M. Rudman, Volume-tracking methods for interfacial flow calculation, *Int. J. Numer. Methods Fluids* **24**, 671 (1997).
33. R. Scardovelli and S. Zaleski, Direct numerical simulation of free-surface and interfacial flow, *Annu. Rev. Fluid. Mech.* **31**, 567 (1999).
34. J. Strain, Tree methods for moving interfaces, *J. Comput. Phys.* **151**, 616 (1999).
35. M. Sussman and P. Smereka, Axisymmetric free boundary problems, *J. Fluid Mech.* **341**, 269 (1997).
36. M. Sussman, A. S. Almgren, J. B. Bell, P. Collela, L. H. Howell, and M. Welcome, An adaptative level set approach for incompressible two-phase flows, *J. Comput. Phys.* **148**, 81 (1999).
37. P. K. Sweby, High resolution schemes using flux limiters for hyperbolic conservation laws, *SIAM J. Numer. Anal.* **21**, 995 1984.

38. R. Temam, ed., Navier–Stokes Equations (North-Holland, Amsterdam, 1984).
39. C. P. Thompson and P. Lezeau, Application of the full approximation storage method to the numerical simulation of two-dimensional steady incompressible viscous multiphase flows, *Int. J. Numer. Methods Fluids* **28**, 1217 (1998).
40. G. Tryggvason, Numerical simulations of the Rayleigh–Taylor instability, *J. Comput. Phys.* **75**, 253 (1988).
41. S. O. Unverdi and G. Tryggvason, Computations of multi-fluid flows, *Physica D* **60**, 70 (1992).
42. H. Uzawa, in *Iterative Method for Concave Programming*, edited by Arrow, Hurwicz, and Uzawa, Studies in Linear and Nonlinear Programming (Stanford Univ. Press, Stanford, CA, 1958).
43. S. Vincent and J.-P. Caltagirone, Efficient solving method for unsteady incompressible interfacial flow problems, *Int. J. Numer. Methods Fluids* **30**, 795 (1999).
44. S. Vincent, J.-P. Caltagirone, and E. Arquis, Numerical simulation of liquid metal particules impacting onto solid substrate: Description of hydrodynamic processes and heat transfers, *J. High Temperatures Material Processes*, in press.
45. S. Vincent, J.-P. Caltagirone, and E. Arquis, Direct numerical simulation of liquid droplet flattering on both solid substrate and liquid film, in *Proceedings, Second International Symposium on Two-Phase Flow Modelling and Experimentation, Pisa, May 23–25, 1999*, Vol. 2, p. 995.
46. S. Vincent and J.-P. Caltagirone, Implicit finite volume approximation of incompressible multiphase flows using an original one cell local multigrid method, in *Proceedings, Finite Volumes for Complex Applications, Hermes, Duisburg, July 19–22, 1999*, p. 367.
47. S. Vincent and J.-P. Caltagirone, Numerical solving of incompressible Navier–Strokes equations using an original local multigrid refinement method, *C. R. Acad. Sci. Ser. IIb* **328**, 73 (2000).
48. S. Vincent and J.-P. Caltagirone, Solving two-phase flows with a coupled TVD interface capturing/local mesh refinement method in *Godunov Methods: Theory and Applications*, to appear.
49. S. Vincent, *Modelling Incompressible Flows of Non-miscible Fluids* (Ph.D. thesis, Univ. of Bordeaux I, 1999).
50. H. A. Van Der Vorst, Bi-CGSTAB: A fast and smoothly converging variant of Bi-CG for the solution of non-symmetric linear systems, *SIAM J. Sci. Stat. Comput.* **13**, 631 (1992).
51. A. L. Yarin and D. A. Weiss, Impact of drops on solid surfaces: Self-similar capillary waves, and splashing as a new type of kinematic discontinuity, *J. Fluid Mech.* **283**, 141 (1995).
52. D. L. Youngs, Time-dependent multi-material flow with large fluid distortion, in *Numerical Methods for Fluid Dynamics*, edited by Morton and Baines, Institute for Mathematics and Its Applications Conference Series (Academic Press, New York, 1982), Vol. 4, p. 27.