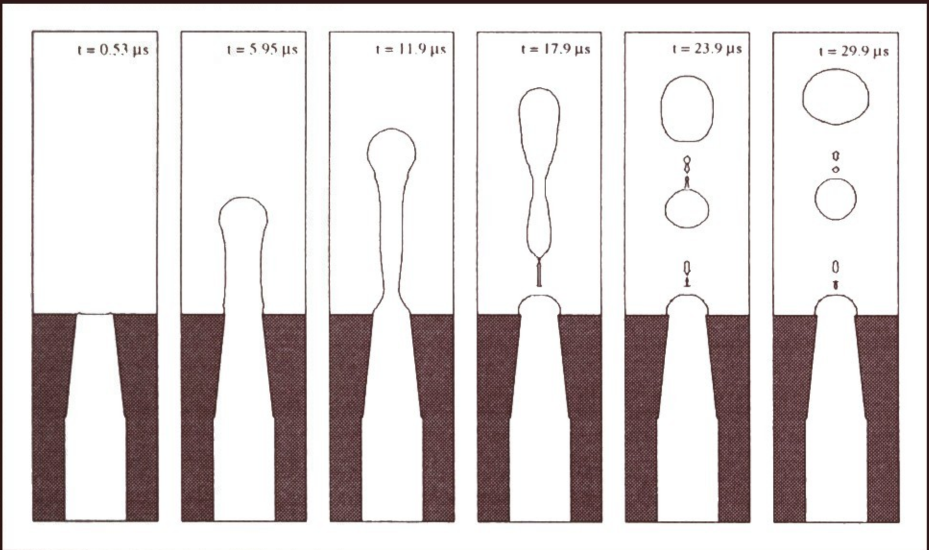
ICASE/LaRC Interdisciplinary Series in Science and Engineering

Barriers and Challenges in Computational Fluid Dynamics



Edited by
V. Venkatakrishnan, Manuel D. Salas
and Sukumar R. Chakravarthy

Springer-Science+Business Media, B.V.

Barriers and Challenges in Computational Fluid Dynamics

edited by

V. Venkatakrishnan

Boeing Commercial Airplane Group,
Seattle, Washington, U.S.A.

Manuel D. Salas

ICASE,
Hampton, Virginia, U.S.A.

and

Sukumar R. Chakravarthy

Metacomp Technologies Inc.,
Westlake Village, California, USA

SPRINGER-SCIENCE+BUSINESS MEDIA, B.V.

A C.I.P. Catalogue record for this book is available from the Library of Congress.

ISBN 978-94-010-6173-5   ISBN 978-94-011-5169-6 (eBook)
DOI 10.1007/978-94-011-5169-6

The cover illustration shows the ejection of water into air from a nozzle driven by a 10 microsecond impulse in the shape of 1/2 of a cosine wave. The domain is 60 x 240 microns and has been covered by a uniform square grid of 32 x 128 cells. This computation was conducted by Igor D. Aleinov and E.G. Puckett in the Department of Mathematics at the University of California, Davis using a high-order accurate projection method with a volume-of-fluid interface tracking algorithm, a Cartesian grid model of the geometry and a generalized continuum surface force model of surface tension.

Printed on acid-free paper

ACCURATE AND ROBUST METHODS FOR VARIABLE DENSITY INCOMPRESSIBLE FLOWS WITH DISCONTINUITIES

W. J. RIDER AND D. B. KOTHE
Los Alamos National Laboratory
Los Alamos, NM 87545

AND

E. G. PUCKETT AND I. D. ALEINOV
University of California, Davis
Mathematics Department
Davis, CA 95616

Abstract.

We are interested in the solution of incompressible flows possessing densities that vary both discontinuously and smoothly. Smooth density variations might be caused by temperature effects, whereas abrupt variations are present at immiscible fluid interfaces. If interfaces are present, we wish to model their gross topological changes. The design of an incompressible flow algorithm that maintains solution accuracy and simulation robustness in the presence of density variations presents challenges, especially if the variations are discontinuous and topologically complex.

We discuss the construction of robust, high-fidelity fractional-step projection methods for computing such flows. We focus on algorithms for the projection, the numerical linear algebra, and interfacial physics such as surface tension. Also discussed are algorithms for multi-dimensional advection and volume tracking. Numerical examples are presented to illustrate our current capabilities.

1. Introduction

The flow of incompressible fluids with discontinuous density variations (interfaces) occurs in widespread applications. Water/air free surface flow is

V. Venkatakrishnan et al (eds.), Barriers and Challenges in Computational Fluid Dynamics, 213-230.

a classical example, e.g., a water drop falling into a pool of water. Another important example is the filling of a cast metal mold with a molten metal alloy. Yet another is the production and transport of micron-sized ink drops during inkjet printer operation. Reliable simulation of these types of flows demands a numerical model with accuracy, fidelity, and robustness.

Accuracy is defined as the quality of deviating slightly from fact. For our purposes, this definition is refined as the measured error for a given solution. There is also a distinction between order of accuracy and numerical accuracy. For reasonable grid resolution, methods with a higher order of accuracy can be accompanied by significantly larger numerical error than the lower order method. This naturally leads to our next definition.

Fidelity is defined as exact correspondence with fact. A solution that possesses fidelity is one that is physically meaningful. A method is considered to be high-fidelity when it produces solutions that are accurate relative to the computational resources (the mesh size) applied to them. For example, interface tracking mechanisms can increase solution fidelity by maintaining interface discontinuities as the interface is advected and/or undergoing topological change.

Robustness is the property of being powerfully built or sturdy. A robust method will not fail in a catastrophic manner, but rather "degrade gracefully." Robustness implies that the algorithm can be used with confidence on a difficult problem. The degree to which the degradation is graceful is subject to interpretation. A robust method should produce physically reasonable results beyond the point where accuracy is expected or achieved.

In the next several sections we will focus on the key elements in our incompressible flow solvers. In Section 2 we introduce our projection algorithms for discretizing the incompressibility constraint in a robust manner. Next, in Section 3, we discuss issues related to solving the pressure equation effectively. The methods for computing interface and flow kinematics (advection) are discussed in Section 4. Section 5 follows with an introduction and discussion of our surface tension model. Next we present a set of sample applications to amplify our arguments. Finally, in Section 7, we conclude with a summary of various outstanding numerical issues.

2. Projection Methods

Here we introduce the principal aspects of a projection method. Our basic goal with projection methods is to advance a velocity field, $\mathbf{u} = (u, v)^T$ without regard for the solenoidal nature of $\mathbf{u}$, then recover the proper solenoidal velocity field, $\mathbf{u}^d$ ($\nabla \cdot \mathbf{u}^d = 0$). The means to this end is a projection operator, $\mathcal{P}$, which projects $\mathbf{u}^d$ out of $\mathbf{u}$:

$$\mathbf{u}^d = \mathcal{P}(\mathbf{u}) .$$

The velocity $\mathbf{u}$ can be decomposed into a solenoidal vector, $\mathbf{u}^d$, and a curl-free vector, expressed as the gradient of a potential, $\boldsymbol{\nabla}\varphi$. This decomposition is written

$$\rho\mathbf{u} = \rho\mathbf{u}^d + \boldsymbol{\nabla}\varphi, \tag{1}$$

or

$$\mathbf{u} = \mathbf{u}^d + \sigma\boldsymbol{\nabla}\varphi, \tag{2}$$

where $\sigma = 1/\rho$. Taking the divergence of (2) yields an elliptic equation for φ:

$$\nabla \cdot \mathbf{u} = \nabla \cdot \sigma\boldsymbol{\nabla}\varphi. \tag{3}$$

Given the solution φ to equation (3), $\mathbf{u}^d$ results from the correction,

$$\mathbf{u}^d = \mathbf{u} - \sigma\boldsymbol{\nabla}\varphi.$$

2.1. PROJECTIONS: THE BASIC IDEA

Our fractional-step algorithm consists of a predictor step, in which the solenoidal nature of the velocity field is ignored, and a corrector step, in which a projection recovers the solenoidal velocity field. In the predictor step, the time n velocity in cell (i,j), $\mathbf{u}_{i,j}^n$, is advanced with the convection-diffusion equation:

$$\begin{aligned}\mathbf{u}_{i,j}^{*,n+1} &= \mathbf{u}_{i,j}^n - \Delta t \Bigg[(\mathbf{u}\cdot\boldsymbol{\nabla}\mathbf{u})_{i,j}^{n+\frac{1}{2}} + \sigma_{i,j}^{n+\frac{1}{2}}\mathbf{G}_{i,j}\phi^{n-\frac{1}{2}} \\ &- \frac{\nu\sigma_{i,j}^{n+\frac{1}{2}}}{2} L_{i,j}\left(\mathbf{u}^n + \mathbf{u}^{*,n+1}\right) - \mathbf{f}_{i,j}^{n+\frac{1}{2}} \Bigg], \end{aligned} \tag{4}$$

where $\mathbf{G}_{i,j}$ is the discrete gradient and $L_{i,j}$ is the discrete Laplacian. This provides a nominally second-order discretization. The advection term is discretised with an unsplit high-order Godunov method (Bell *et al.*, 1989; Colella, 1990). For variable density flows, this method is described in (Bell & Marcus, 1992; Puckett *et al.*, 1997).

Several variations of the projection implied by (4) are possible. By removing the gradient of pressure from (4), φ in equation (3) is actually a pressure rather than an increment in pressure. The form of $\mathbf{u}$ in the discrete divergence on the LHS of (3) can be chosen several ways, e.g., the advanced-time predictor velocity, $\mathbf{u}^{*,n+1}$, or the predicted change, $(\mathbf{u}^{*,n+1} - \mathbf{u}^n)$. One might assume these differences to be higher order effects, but experience has shown otherwise for both exact and approximate projections, as discussed later.

An exact projection results when the Laplacian operator (L) on the RHS of equation (3) is derived from the discrete discrete divergence (D) and gradient ($\mathbf{G}$) operators: $L = D\sigma\mathbf{G}$. The discrete velocity divergence in an exact projection is zero to within the convergence tolerance of the solution to equation (3).

The exact discrete projections given above provide a good foundation in the numerical implementation of projection methods, but have some practical difficulties. These problems are discussed in (Almgren *et al.*, 1996; Lai *et al.*, 1993). The pressure/velocity decoupling can interact poorly with localized source terms (e.g., chemical reactions), leading to instabilities. Additionally, the local decoupling renders multigrid techniques cumbersome (Howell, 1993), and hampers the implementation of adaptive grid techniques (Almgren *et al.*, 1993; Howell, 1993).

To address these problems, new types of projection algorithms have been developed. In "approximate" projections, introduced in (Almgren *et al.*, 1996), the operator L is derived from a discretization of the continuous projection operator. The discrete velocity divergence in an approximate projection is not zero, but is rather a function of the truncation error. The operators D and $\mathbf{G}$ have the same form as the exact projection, but the Laplacian is modified.

2.2. ROBUST PROJECTION METHODS

In approximate projections, the velocity divergence is not constrained to be zero (to some small tolerance), hence robust algorithms can be elusive. We will demonstrate this with a single test problem, then describe improvements.

The principal problem with approximate projections is the presence and growth of null spaces in the discrete operators, which is manifested as high-frequency noise. This noise can be controlled by identifying and filtering unphysical velocity modes (Lai, 1993; Rider; 1994) or by carefully formulating the form of the approximate projection (Rider, 1994). Without these steps, approximate projection methods are prone to failure on more difficult problems.

We currently damp these spurious modes in two ways: the explicit addition of a high-order viscosity, or the use of an iterated projection derived from a discrete stencil that differs from the approximate projection stencil. These methods are most effective when used in concert.

The formulation of the projection directly affects the time evolution of the discrete divergence. If the divergence on the LHS of (3) is the difference between the predicted and old time velocity,

$$\nabla \cdot \left(\mathbf{u}^{*,n+1} - \mathbf{u}^{n}\right), \tag{5}$$

then the discrete divergence errors *accumulate* in time. On the other hand, if a predictor velocity is used,

$$\nabla \cdot \mathbf{u}^{*,n+1}, \tag{6}$$

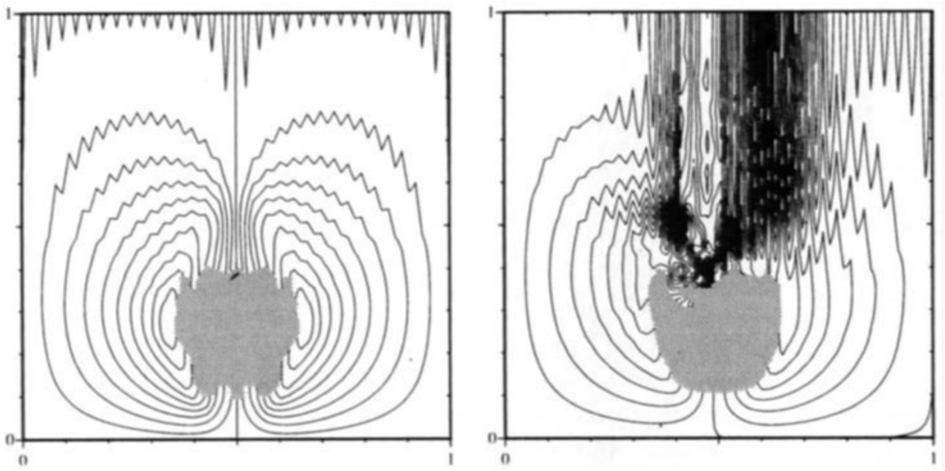
than the discrete divergence errors *do not accumulate* in time, which is preferable. It should be noted that equation (5) is the form standard in the literature.

Consider the following test problem, which will illuminate these subtle issues. A circular drop with radius 0.15 is placed at $(0.5, 0.75)$ in a unit square computational domain that is partitioned with a 64×64 grid. Gravity is unity (downward) and all boundaries are frictionless (free-slip). The drop fluid is 1000 times more dense than the background fluid (having unit density). The flow is integrated forward in time to $t = 1$ using the Euler equations. A high-order Godunov method and an unsplit piecewise linear volume tracking algorithm (discussed in Section 4) is used to advect the flow. The CFL number is $\frac{1}{2}$ unless otherwise stated. The unsteady flow is computed with variations of both the exact and approximate projection methods. Each method demonstrates second-order convergence (in space and time) on sufficiently smooth problems.

Solutions obtained with the standard exact and approximate projection methods (i.e., without filters) are shown in Figure 1[1]. Both solutions exhibit spurious features in the velocity field in the flow above the drop. The exact projection solution (Figure 1a) displays some velocity field decoupling and slight asymmetries. Despite the use of a smaller time step in integrating the flow (CFL=1/4), the approximate projection solution in Figure 1b is unacceptable. As discussed later, this solution is compromised in part because projection in equation (3) is the projecting equation (5) and solving for a pressure increment rather than a total pressure.

When the predictor velocity ($\mathbf{u}^{*,n+1}$) is projected (rather than the velocity difference) and the total pressure (rather than the pressure increment) is solved for in equation (3), the drop solutions improve significantly, as shown in Figure 2. The exact projection solution in Figure 2a now exhibits symmetry and a coupled velocity field. The approximate projection solution (Figure 2b) additionally requires velocity filters before its solution quality matches and surpasses that of the exact projection. The decoupled velocity field and asymmetries evident in Figure 1b are effectively suppressed.

[1] The standard formulation solves for an increment in pressure and projects $\nabla \cdot (\mathbf{u}^{*,n+1} - \mathbf{u})$



(a) Exact pressure increment projection of the velocity difference.

(b) Approximate pressure increment projection of the velocity difference.

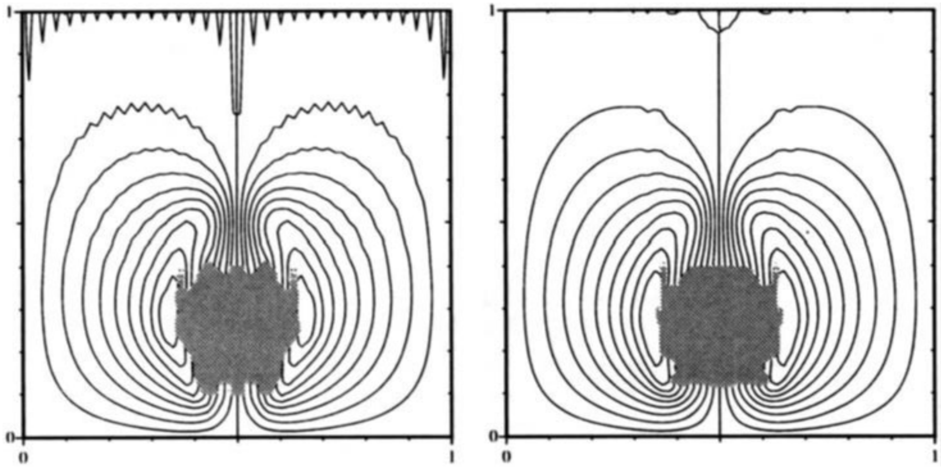
Figure 1. Drop solutions for our standard exact and approximate projections. Both solutions use a grid where all variables are cell-centered. The droplet outline and the velocity streamlines are shown. In both cases the projection equation LHS is given by equation (5).

3. Linear Algebra

The cost of incompressible flow solutions based on projection methods is typically dominated by the effort required to find solutions to the elliptic pressure equation (3). Designing an efficient and scalable method for solving this system of linear equations is therefore of paramount importance. We have used three methods in our study: a preconditioned conjugate gradient (CG) method, a multigrid (MG) method, and a multigrid preconditioned conjugate gradient (MGCG) method. Our results indicate that the MGCG method is the most effective.

3.1. CONJUGATE GRADIENT METHODS

Because the exact and approximate projections produce symmetric semi-positive-definite and positive-definite systems of linear equations, we can use CG methods (Golub & Van Loan, 1989). We can also employ preconditioning to improve convergence, which is especially important when density ratios across interfaces are large. Typically we use an incomplete Cholesky



(a) Exact pressure projection of the predictor velocity.

(b) Filtered approximate pressure projection of the predictor velocity.

Figure 2. The end product of the changes in the projection formulation and the filtering of approximate projections. The projection equation LHS is given by equation (6).

method or some other iterative procedure (SSOR, Jacobi) in finding approximate solutions to the preconditioning equation.

Several properties of preconditioned CG method's ability to solve equation (3) are worth noting. First, it is extremely robust (having never failed in our experience), and is also general, being applicable to any of our exact or approximate projections. On the other hand, the amount of work required by the preconditioned CG method grows as $N^{3/2}$ (N is the total number of unknowns). Therefore the CG method demands an ever-increasing proportion of the CPU time as the grid is refined.

3.2. MULTIGRID METHODS

Solutions to the linear systems arising from approximate projections can also be obtained with a MG algorithm (Briggs, 1987). Use of a MG algorithm is desirable because of its attractive scaling. The operation count for classical direct linear algebra solution techniques (e.g., Cholesy) scale like N^3. This scaling improves to N^2 for banded solvers that take advantage of the structure of the linear system. MG scales linearly with N. Thus, MG (where it works) will eventually provide the fastest route to a solution as the grid is refined.

One of the most important tasks in formulating a MG algorithm is the approximation of the coarse-grid linear equations. One approach is using intergrid transfer functions to define variational or Galerkin coarse grid operators. The complication and expense of this task, however, has motivated a simpler approach based on suggestions in (Liu *et al.*, 1992; Lai, 1993). The operator remains the same, as do the boundary conditions on the course grids, but σ must be defined at each level. The basic idea is to construct coarse grid approximations that produce an average cell value for $\sigma\nabla\varphi$ that is identical to the fine-level value. In using the cell-centered MG framework, this control volume derivation of the equations comes quite naturally.

We find that our MG algorithm converges quickly to a solution in most cases, but fails on occasion with flows having interfaces possessing large density variations and complex topology (e.g., a drop splashing into a pool).

3.3. MULTIGRID PRECONDITIONED CONJUGATE GRADIENT

Our current solution to the MG robustness problem is to employ the symmetric MG algorithm to precondition a standard CG method. The idea was first proposed and demonstrated by Tatebe (Tatebe, 1993). Experimentation has proven the utility of combining these two methods. This combined MGCG method usually scales like a MG algorithm, but occasionally CG-like scaling is exhibited. Nevertheless, we have found it to be robust. Ultimately we wish to design a robust method that consistently exhibits MG-like scaling.

4. Advection and Interface Tracking

4.1. HIGH-ORDER GODUNOV METHODS

We now discuss the basic principles of our advection method, which is based on the framework established in (van Leer, 1984; Bell *et al.*, 1988; Colella, 1990). Our method has many similarities with Colella's formulation, as modified for incompressible flow algorithms based on projection methods (Bell *et al.*, 1989). These methods are "unsplit", i.e., a full multi-dimensional solution is updated in a single time step. Single-step, multi-dimensional integration is important because the incompressibility constraint is inherently multi-dimensional. The flow solver should therefore reflect the intrinsic coupling of the velocity field.

Multi-dimensional advection algorithms are constructed via the time-centered approximation of the dependent variables at cell-edges. Time-centering is accomplished with the variable's full PDE (with all terms included). While the details of the methods are given in the above-stated references, important contributions in the use of these methods have recently been made. Brown and Minion discuss the nature of these solutions when resolution is not adequate (Brown & Minion, 1995), and Minion has suggested stability-enhancing improvements (Minion, 1996).

4.2. VOLUME TRACKING OF INTERFACES

The essential features of volume tracking methods are as follows. First, an initial (known) fluid interface geometry is used to generate initial fluid volume fractions in each computational cell. This task requires computing the volume truncated by the fluid interface in each cell containing an interface. Exact interface information is then discarded in favor of the discrete volume fraction data.

Interfaces are subsequently "tracked" by evolving fluid volumes in time with the solution of a standard advection equation. At any time in the solution, an exact interface location is not known, i.e., a given distribution of volume fraction data does not guarantee a unique interface topology. Interface geometry is instead inferred (based on assumptions of the particular algorithm) and its location is "reconstructed" from local volume fraction data. Interface locations are then used to compute the volume fluxes necessary for the advective term in the volume evolution equation. Volume fluxes are therefore approximated geometrically rather than algebraically. Typical implementations of these algorithms are one-dimensional, with multidimensionality built up through operator splitting. The assumed interface geometry, interface reconstruction, and volume flux calculation typically comprise the unique features of a given volume tracking method.

Our piecewise linear volume tracking algorithm, as implemented, is straightforward, simple, and extensible. This is accomplished by drawing upon the extensive literature available in the field of of computational geometry (O'Rourke, 1993). The algorithm is robust, second-order accurate in time and space, and is constructed from a set of simple geometric functions. A detailed account of our volume tracking algorithm, including comparison with other methods, is given in (Rider & Kothe, 1996). Pilliod and Puckett (Pilliod & Puckett, 1997) also review and analyze volume tracking methods, as well as introducing their own unsplit time integration scheme.

5. Surface Tension

Our current models for interfacial surface tension begin with methodology established in the continuum surface force (CSF) method (Brackbill *et al.*, 1992). The basic premise of the CSF method to model physical processes specific to and localized at fluid interfaces (e.g, surface tension) by applying the process to fluid elements everywhere within interface transition regions. Surface processes are replaced with volume processes whose integral effect properly reproduces the desired interface physics. This approach falls under the general class of immersed interface methods (Leveque & Li, 1994) whose origin dates back to the pioneering work of Peskin (Peskin, 1977). The CSF method lifts all topological restrictions without sacrificing accuracy, robustness, or reliability. It has been verified extensively in 2-D flows through its implementation in a classical algorithm for free surface flows (Kothe *et al.*, 1991; Kothe & Mjolsness, 1992), where complex interface phenomena such as breakup and coalescence have been modeled.

In the CSF model, surface tension is reformulated as a volumetric force given by

$$\mathbf{F_s} = \mathbf{f_s}\delta_s\,. \tag{7}$$

Here δ_s is a surface delta function and $\mathbf{f_s}$ is the surface tension force per unit interfacial area (Brackbill *et al.*, 1992):

$$\mathbf{f_s} = \tau\kappa\hat{\mathbf{n}} + \nabla_s\tau\,, \tag{8}$$

where τ is the surface tension coefficient, ∇_s is the surface gradient, $\hat{\mathbf{n}}$ is the interface unit normal, and κ is the mean interfacial curvature:

$$\kappa = -(\nabla\cdot\hat{\mathbf{n}})\,. \tag{9}$$

The first term in (8) is a force acting normal to the interface, proportional to the curvature κ. The second term is a force acting along the interface (tangentially) toward regions with higher surface tension coefficient values. The normal force tends to smooth and propagate regions of high curvature, whereas the tangential force tends to force fluid along the interface toward regions of higher τ.

The surface delta function was proposed in the original CSF model to be (Brackbill *et al.*, 1992)

$$\delta_s = \frac{|\nabla c|}{[c]} = |\mathbf{n}| \tag{10}$$

where c is the characteristic (color) function uniquely identifying each fluid in the problem and $[c]$ is the jump in the color function across the interface in question, which is unity since volume fractions serve as the color function in this work. If a wide stencil is used for $\hat{\mathbf{n}}$ in (10), then δ_s will be nonzero in cells that are in close proximity to the interface. We currently force δ_s to be zero in these cells, which causes the CSF to be non-zero only within the interface transition region. A proper δ_s insures that the CSF is normalized to recover the conventional description of surface tension as the local product $\kappa h \to 0$.

Despite the success of the CSF model and related immersed interface methods, outstanding issues remain (Kothe *et al.*, 1996). If these issues can be resolved adequately, a wider range of surface tension-driven flows will be modeled reliably. For example, improved forms for δ_s, displaying better convergence and smoothness properties, are needed. Our current numerical results are very sensitive to the form used for δ_s, indicating that the quality of CSF model relies heavily on the quality of the form used for δ_s (Kothe *et al.*, 1996). Recent results by Aleinov and Puckett (Aleinov & Puckett, 1995) motivate the use of other kernels, such as the Peskin kernel (Peskin, 1977) or higher-order Nordmark (Nordmark, 1991) kernel.

Perhaps the most stringent test for a surface tension model is a test of the ability to maintain an equilibrium (minimal energy) configuration. A 2-D or 3-D static drop is such an example (Kothe *et al.*, 1996). Here a perfectly spherical drop is placed in a lighter-density background fluid, and all forces are ignored except the drop interfacial surface tension. The drop should remain stationary, as the net surface tension force is zero. An incompressible flow solution for this system, however, generates false flow dynamics (dubbed "parasitic currents") that can grow with time (sometimes unbounded) (Kothe *et al.*, 1996). The source of these currents originates in part with the surface tension model, as the computed pressure gradient at the drop interface does not exactly cancel the surface tension force. New developments in surface tension models must address this inability to maintain an equilibrium configuration.

6. Applications

6.1. A MOLD-FILLING PROBLEM

As an example of our current 3-D capabilities, consider the following sample "mold-filling" problem. A rectangular box, spanning $0 \leq x, y \leq 24$ and $0 \leq z \leq 30$, is partitioned with $24 \times 24 \times 30$ unit cubical cells. The box is initially filled with a quiescent background fluid having a density ten times less than the filling fluid. At time zero, filling fluid is injected with velocity $(0.0, 0.0, -88.6)$ through a hole in the box at $0 \leq x, y \leq 5$ and $z = 30$ (the top corner). The background fluid is allowed to escape through a vent at $19 \leq x, y \leq 24$ and $z = 30$ (the opposite top corner). Gravity is $(0.0, 0.0, -980.6)$, and both the background and filling fluid are assumed to be incompressible and inviscid. Surface tension at interfaces between the background and filling fluid is neglected. The filling dynamics (as inferred from the interface topology) are followed up to a time of 3.0.

As is evident from Figure 3, which depicts the filling fluid interface topology at four different times, this idealized calculation presents an energetic and rigorous test of our ability to model complex topology free surface flows encountered in the mold-filling process. Solutions here are obtained with a cell-centered approximate pressure projection method without filters. Equation (6) is used for the LHS of the projection. Interfaces are tracked with our piecewise-planar volume tracking method (Kothe *et al.*, 1996). Additional results of this simulation (including animations) can be found in (Kothe *et al.*, 1995).

Next we consider another difficult interfacial flow application, namely the production and transport of ink drops in an inkjet printing process.

6.2. INKJET PRINTER NOZZLE

Surface Tension The presence of the surface delta function in the expression for the continuum surface tension force is often a source for instabilities and poor convergence of projection methods. If we suppose that the surface tension coefficient τ is constant, then the singularity can be eliminated. Upon substitution of (8) and (10) into equation (7) for the surface tension force, and using the relation $\mathbf{n}|\nabla c| = \nabla c$, we can write

$$\mathbf{F}_s = \tau\kappa\nabla c = \nabla(\tau\kappa c) - \tau c\nabla\kappa \,. \tag{11}$$

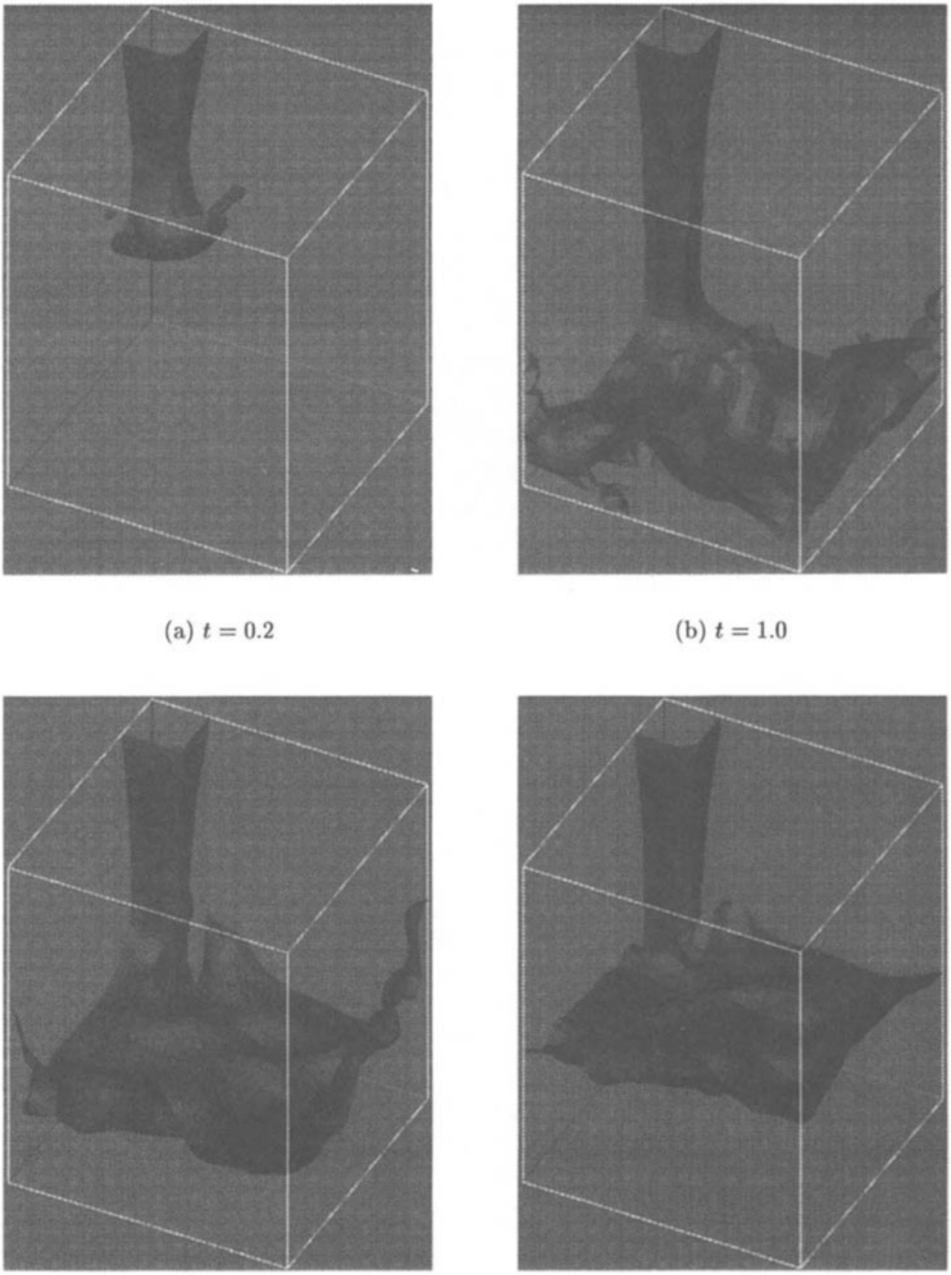
Of course, this equation is valid only if κ is defined in the entire domain and is smooth. This is not the case, since κ is defined only on the interface, but if the interface is smooth then κ can be spread smoothly over the entire domain by some simple averaging procedure. In this case the first term in the RHS of (11), which contains the main singularity, can be included into a definition of the pressure and the remaining surface tension force expression will be less singular (it will contain a Heaviside function instead of a delta function). In other words, we can introduce new definitions for the pressure and surface tension force

$$\tilde{p} = p - \sigma\kappa c \tag{12}$$

$$\tilde{\mathbf{F}}_s = -\sigma c\nabla\kappa \,. \tag{13}$$

for which the Navier-Stokes equations will have the same form, except the volumetric force will not have a delta function. This method was used by Sussman (Sussman *et al.*, 1996) with a level set method. We have found this method to be more robust than the standard CSF method. It enabled us to perform computations for which the standard CSF method failed (e.g., the computation presented here).

Geometry If the computational domain has a complex geometry, special techniques are required to impose the boundary conditions. We use a Cartesian grid, in which a regular rectangular grid is cut by external boundaries. Such a grid remains rectangular away from the boundary but has irregular "incomplete" cells on the fluid-body interface. The algorithms described above need to be reformulated for the irregular boundary cells.



(a) $t = 0.2$

(b) $t = 1.0$

(c) $t = 2.0$

(d) $t = 3.0$

Figure 3. Three-dimensional simulation of a box being filled with a heavier fluid from the top corner. Shown is the filling fluid interface topology (the 0.5 volume fraction isosurface) at four different times.

The problem that arises in the velocity advection algorithm is due to the CFL constraint that requires a time step size to be proportional to the cell size. Since we want to model an arbitrary geometry, we may have very small cells on the fluid-body boundary. This places a severe restriction on the time step size, slowing down the computations and causing other difficulties. We solve this problem using the flux redistribution algorithm, described in (Almgren *et al.*, 1997), which allows us to use a time step size computed according to the regular cell CFL constraint. The general idea is to first advect velocities using a conservative transport algorithm while ignoring the fluid-body boundary. Next, in each boundary cell we apply a correction computed according to the stability requirement. Extra fluxes resulting from this correction are then distributed over the neighboring cells. Though stability of this algorithm has not been proven, its behavior is robust in our experience.

A general approach to elliptic solvers is to use a standard stencil for elliptic operator away from the boundary, but modify it for irregular cells. We use a finite element approach to the approximate projection in which pressure is located at the vertices of a regular grid. This algorithm is transferred to a Cartesian grid by changing the domain of integration in the weak form of the evolution equations. Integrals over the entire cells are replaced by integrals over those parts of the cells which are inside the fluid. The viscous solver can also be reformulated for a Cartesian grid, though it requires more work. At the moment we use a "stair-step" approximation for the viscous solver in which boundary cells are treated as if they are entirely in the fluid and no-slip boundary conditions are imposed at edges entirely in the wall.

Volume tracking methods are more complicated on Cartesian grids, since they need an interface to be reconstructed and transported within non-rectangular cells. Currently we employ a "stair-step" approximation for interface reconstruction in those cells. Another issue in the presence of multiple fluids and boundary cells are contact angles between fluid-fluid interfaces and the fluid-body boundary. At the moment we allow only two contact angles for wettable and non-wettable surfaces.

Computations Figures 4 and 5 depict computations made for ejection of water into air from the nozzle of an ink jet printer. The surface inside the nozzle is assumed to be wettable and the surface outside the nozzle to be non-wettable. Although our numerical method is purely two-dimensional, the surface tension force is computed using cylindrical symmetry. This is necessary, because curvature in the third dimension is significant in such problems, therefore neglecting it gives unrealistic results. Uniform Dirichlet boundary conditions on velocity are used on the lower boundary. Other boundary conditions for velocity are free flow on the upper boundary and

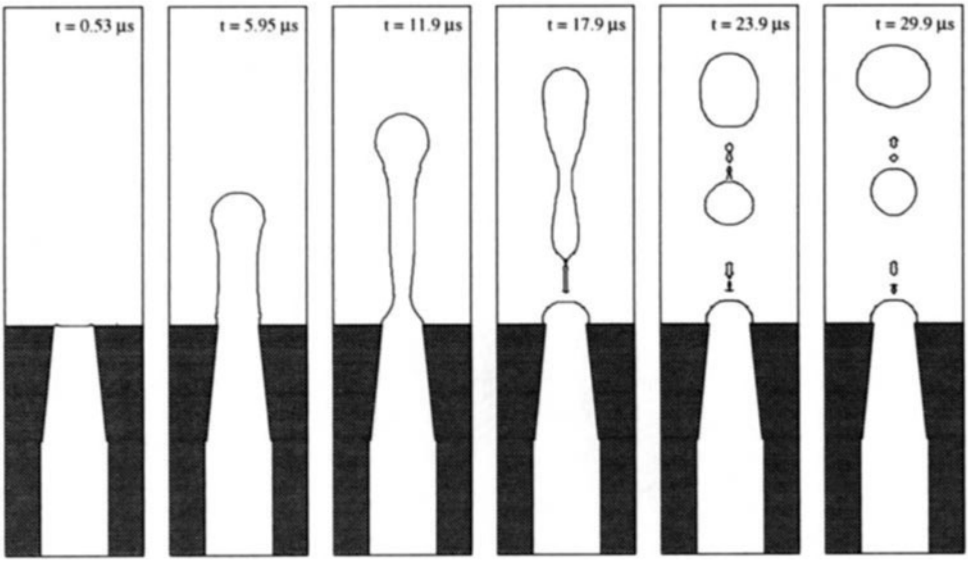


Figure 4. Ejection of water into air from a nozzle: 1/2 cosine wave impulse. The dimensions of the domain are 60 x 240 microns, with the duration of the impulse being 10 μs. The grid is 32 $\times$ 128 cells. The fluid densities are 1000 kg/m^3 and 1 kg/m^3 with viscosities of 0.001137 kg/m s and 0.00001776 kg/m s.

no-slip on the left and right boundaries. The inflow velocity at the bottom boundary is given by an impulse model using 1/2 and 1/4 of a cosine wave to $\approx$ 30 μs, followed by no further inflow. In both cases the duration of the impulse and the amount of the injected fluid are the same, except the impulse shapes are slightly different. We are interested in following the formation of the droplet and its separation from the nozzle. In both cases we observe the formation of a satellite, with its size depending on the shape of the impulse. The slightly unphysical meniscus shape at 0.53 microseconds is due to our initial conditions (flat surface) being incompatible with the specified contact angle.

7. Next Steps

We have made significant progress in achieving the goal of constructing accurate, robust solvers for variable density incompressible flows with discontinuities. Despite significant progress, much work remains. Application of our methodology to a wider variety of flows is focusing our attention on current weaknesses. Chief among these is the extension of our method to nonorthogonal, three-dimensional grids, which is currently being pursued in a new casting simulation tool (Kothe *et al.*, 1995). Additional research effort is targeting improved surface tension models, as their robustness and

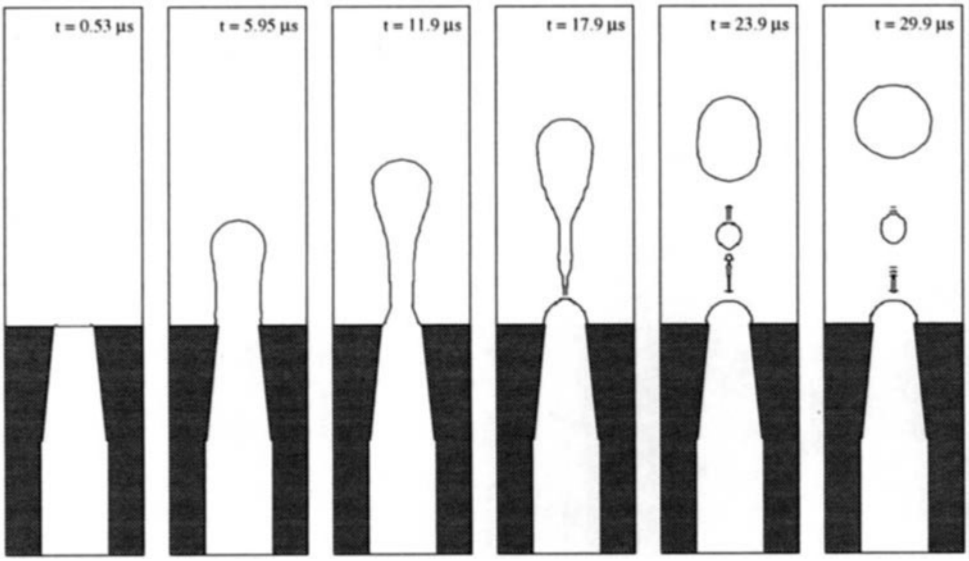


Figure 5. Ejection of water into air from a nozzle: 1/4 cosine wave impulse.

accuracy is currently less than satisfactory. Efficient linear algebra on 3-D complex grids is needed, hence will also be the focus of future efforts.

References

Aleinov, I. and Puckett, E. G. Puckett, "Computing Surface Tension with High-Order Kernels," *6th International Symposium on Computational Fluid Dyanamics*, 1995.

Almgren, A. S., Bell, J. B., Colella, P., and Howell, L. H., "An adaptive projection method for the incompressible Euler equations," *Proceedings of the AIAA Eleventh Computational Fluid Dynamics Conference*, J.L. Thomas, ed., AIAA Paper 93-3345, pp. 530-539, 1993.

Almgren, A. S., Bell, J. B., Colella, P., and Marthaler, T., "A Cartesian grid projection method for the incompressible Euler equations in complex geometries," *SIAM J. Sci. Comput.*, Vol. 18, No. 1, September 1997.

Almgren, A. S., Bell, J. B., and Szymczak, W. G., "A numerical method for the incompressible Navier-Stokes equations based on an approximate projection," *SIAM Journal of Scientific Computing*, Vol. 17, 1996, pp. 358-369.

Bell, J. B., Colella, P., and Glaz, H. M., "A second-order projection method of the incompressible Navier-Stokes equations," *Journal of Computational Physics*, Vol. 85, 1989, pp. 257-283.

Bell, J. B., Dawson, C. N., and Shubin, G. R., "An unsplit, higher order Godunov method for scalar conservation laws in multiple dimensions," *Journal of Computational Physics*, Vol. 74, 1988, pp. 1-24

Bell, J. B. and Marcus, D. L., "A second-order projection methòd variable-density flows," *Journal of Computational Physics*, Vol. 101, 1992, pp. 334-348.

Brackbill, J. U., Kothe, D. B., and Zemach, C., "A continuum method for modeling surface tension," *Journal of Computational Physics*, Vol. 100, 1992, pp. 335-354.

Briggs, W. L., "A Multigrid Tutorial," *SIAM*, 1987.

Brown, D. L. and Minion, M. L., "Perfromance of underresolved two-dimensional incompressible flow simulations," *Journal of Computational Physics*, Vol. 122, 1995, pp. 165-183.

Colella, P., "Multidimensional upwind methods for hyperbolic conservation laws," *Journal of Computational Physics*, Vol. 87, 1990, pp. 171-200.

Golub, G. H. and Van Loan, C. F., *Matrix Computations*, Johns Hopkins University Press, 1989.

Howell, L. H., "A multilevel adaptive projection method for unsteady incompressible flow," *Proceedings of the Sixth Copper Mountain Conference on Multigrid Methods*, N. D. Melson, T. A. Manteuffel, and S. F. McCormick, eds., 1993, pp. 243-257.

Kothe, D. B. et al., "Computer simulation of metal casting processes: A new approach," Technical Report LALP-95-197, Los Alamos National Laboratory, 1995. Available on World Wide Web at `http://gnarly.lanl.gov/Telluride/Telluride.html`.

Kothe, D. B. and Mjolsness, R. C., "Ripple: A new model for incompressible flows with free surfaces," *AIAA Journal*, Vol. 30, 1992, pp. 2694-2700.

Kothe, D. B., Mjolsness, R. C., and Torrey, M. D., "Ripple: A computer program for incompressible flows with free surfaces," Technical Report LA-12007-MS, Los Alamos National Laboratory, 1991.

Kothe, D. B., Rider, W. J., Mosso, S. J., Brock, J. S., and Hochstein, J. I., "Volume tracking of interfaces having surface tension in two and three dimensions," Technical Report AIAA 96-0859, AIAA, 1996. Presented at the 34rd Aerospace Sciences Meeting and Exhibit.

Lai, M. F., "A Projection Method for Reacting Flow in the Zero Mach Number Limit," Ph.D. thesis, University of California at Berkeley, 1993.

Lai, M., Bell, J. B., and Colella, P., "A projection method for combustion in the zero Mach number limit," *Proceedings of the AIAA Eleventh Computational Fluid Dynamics Conference*, J.L. Thomas, ed., 1993, pp. 776-783; AIAA Paper 93-3369.

Leveque, R. J. and Li, Z., "The immersed interface method for elliptic equations with discontinuous coefficients and singular sources," *SIAM Journal on Numerical Analysis*, Vol. 31, 1994, pp. 1019-1044.

Liu, C., Liu, Z., and McCormick, S., "An efficient multigrid scheme for elliptic equations with discontinuous coefficients," *Communications in Applied Numerical Methods*, Vol. 8, 1992, pp. 621-631.

Minion, M. L., "On the stability of Godunov projection methods for incompressible flow," *Journal of Computational Physics*, Vol. 123, 1996, pp. 435-449.

Nordmark, H. O., "Rezoning for higher order vortex methods," *Journal of Computational Physics*, Vol. 97, 1991, pp. 366-397.

O'Rourke, J., *Computational Geometry in C*, Cambridge, 1993.

Peskin, C. S., "Numerical analysis of blood flow in the heart," *Journal of Computational Physics*, Vol. 25, 1977, pp. 220-252.

Pilliod, J. E., Jr. and Puckett, E. G., "Second-Order Volume-of-Fluid Algorithms for Tracking Material Interfaces," in preparation.

Puckett, E. G., Almgren, A. S., Bell, J. B., Marcus, D. L., and Rider, W. J., "A High-Order Projection Method for Tracking Fluid Interfaces in Variable Density Incompressible Flows," *Journal of Computational Physics*, Vol. 130, 1997, pp. 269-282.

Rider, W. J., "Filtering nonsolenoidal modes in numerical solutions of incompressible flows," Technical Report LA-UR-94-3014, Los Alamos National Laboratory, 1994. Available on World Wide Web at `http://www-xdiv.lanl.gov/XHM/personnel/wjr/Web_papers/pubs.html`.

Rider, W. J., "The robust formulation of approximate projection methods for incompressible flows," Technical Report LA-UR-94-3015, Los Alamos National Laboratory, 1994. Available on World Wide Web at `http://www-xdiv.lanl.gov/XHM/personnel/wjr/Web_papers/pubs.html`.

Rider, W. J. and Kothe, D. B., "Reconstructing volume tracking," Technical Report LA-UR-96-2375, Los Alamos National Laboratory, 1996. Available on

World Wide Web at `http://www-xdiv.lanl.gov/XHM/personnel/wjr/Web_papers-/pubs.html`, also submitted to the *Journal of Computational Physics.*

Sussman, M., Almgren, A., Bell, J., Colella, P., Howell, L., and Welcome, M., "An adaptive level set approach for incompressible two-phase flows," *Proceedings of the ASME Fluids Engineering Summer Meeting*, July 1996.

Tatebe, O., "The multigrid preconditioned conjugate gradient method," *Proceedings of the Sixth Copper Mountain Conference on Multigrid Methods*, N. D. Melson, T. A. Manteuffel, and S. F. McCormick, eds., 1993, pp. 621-634.

van Leer, B., "Multidimensional explicit difference schemes for hyperbolic conservation laws," *Computing Methods in Applied Sciences and Engineering VI*, In R. Glowinski and J.-L. Lions, eds., 1984, pp. 493-497.

A VARIATIONAL APPROACH TO DERIVING SMEARED-INTERFACE SURFACE TENSION MODELS

DAVID JACQMIN
NASA Lewis Research Center
Cleveland, Ohio

Abstract.

Smeared-interface surface tension models can be derived variationally. The advantage of the variational approach is that the resulting models, both continuum and discrete, conserve total energy. This paper explains the variational approach and shows how to apply it to derive energetically consistent CSF (Brackbill et al., 1992) and distributed force (Unverdi & Tryggvason, 1992) surface tension models.

1. Introduction

Smeared interface models have recently become popular for conducting numerical simulations of multi-phase flow. These models use fixed grids and model the interface as having finite width that is resolvable on the grid. The fractional presence of a phase can then be represented by a field variable that varies continuously through the interface. Fluid properties are keyed to this variable and surface tension forces are distributed through the width of the model interface. These models have slow convergence but they allow the approximate solution of problems that would otherwise be intractable.

This paper gives a brief introduction to the use of a variational method for generating such models. The method starts with a model of the thermodynamics, or, if the system is isothermal, just the free energy density, of the model two-phase fluid. The surface tension forces are derived from the fact that advection can change fluid energy, by changing shape, but doesn't change total energy. An advective process that changes fluid energy must therefore be subject to forces that change kinetic energy so as to keep the total energy constant. Calculus of variations can be used to investigate arbitrary advective perturbations to the system and to deduce the required

V. Venkatakrishnan et al (eds.), Barriers and Challenges in Computational Fluid Dynamics, 231-240.

offsetting forces. The method is really the same as the method of virtual work commonly used in elasticity and to analyze mechanical systems. The variational approach is applicable to both discrete and continuum energy models. A great advantage of the method is that its resulting surface tension models are always energy conserving.

The two main smeared surface tension techniques are the distributed force method of Unverdi & Tryggvason (1992) and the continuum surface tension method of Brackbill, Kothe & Zemach (1992). A newer technique is the phase-field method (Anderson & McFadden, 1996; Antonovskii, 1995; Chella & Viñals, 1996; Jacqmin, 1995b, 1996; Jasnow & Viñals, 1996; Nadiga & Zaleski, 1996). This method is the source of the ideas to be discussed and applied here.

The method of Unverdi & Tryggvason tracks interfaces by following the advection of control points. These points mark the smeared interface's center. The interfaces are further defined by connecting the control points by curves or line segments (in 2-D) or triangular surfaces (in 3-D). Surface tension forces are calculated from the control point positions and distributed to the fixed grids. Changes in fluid properties across the interface are smoothed so as to occur over several grid cells.

In the continuum-surface-tension method of Brackbill et al. the surface tension forcing is set equal to the interface gradient times its curvature. The total forcing on the fluid through an interface is thus proportional to the interface's gradient-averaged curvature. The method has been applied using volume-of-fluid (Lafaurie et al. 1994; Rider et al., 1995; Kothe et al., 1996), TVD (Jacqmin,1995a), and level-set (Sussman et al., 1994; Sussman et al., 1996) methods. VOF methods actually use what amounts to two functions to define an interface. The transition from one phase to another is represented by a "color function" which varies from 0 to 1. VOF advection operating on this color function generates a very steep interface, less than two cells wide. This is too steep to allow smooth curvature calculations so surface tension forcing and material property variations are computed using a locally averaged (mollified) form of the color function. The level-set method makes use of a smooth "approximate-distance" function. The zero value of this function marks the smeared-interface center. Convection methods appropriate for smooth functions can be used. The function is initialized as a distance function and thereafter maintained against excessive advective distortion by relaxation, downgradient to the (evolving) distance function, and/or by occasional reinitializations. Surface tension forces are found by calculating the curvature of the function in the vicinity of its zero level.

The above methods are based on models of surface tension *forces.* Phase field methods are based on models of fluid *energy.* The simplest model of energy density that gives two phase flow is

$$e = \frac{1}{2}\alpha|\vec{\nabla}C|^2 + \beta\Psi(C) \tag{1}$$

a formulation that goes back to van der Waals (1893). The first term is gradient energy, the second bulk energy. Two phases are possible if Ψ has two minima. The phase field method uses smooth advection routines with interface profiles maintained against distortion by energy-downgradient antidiffusion.

This paper will show how the variational approach works and how the distributed force method and the VOF version of the CSF method can be derived variationally . A motive for this work has been to find out how the mollified color function can be made consistent with energy conservation. I show how this can be done and also briefly indicate some possibly useful additional smoothing techniques.

2. Variational Derivation of Surface Tension Forces for an Incompressible, Isothermal Fluid System

The surface energy of an isothermal fluid system is equal to its surface area times its surface tension σ. To keep the analysis simple, consider a 2D system with surface height h a single valued function of x. Then the energy of the system is

$$\mathcal{E} = \sigma \int \sqrt{1 + \left(\frac{\partial h}{\partial x}\right)^2} dx + \int \int \frac{1}{2}\rho(u^2 + v^2)dxdy = \mathcal{F} + \mathcal{K} \tag{2}$$

$\mathcal{F}$ is the free energy of the system, $\mathcal{K}$ the kinetic energy. The variation of the free energy with time is

$$\frac{d\mathcal{F}}{dt} = \sigma \int \frac{\partial h/\partial x}{\sqrt{1 + (\partial h/\partial x)^2}} \frac{\partial^2 h}{\partial t \partial x} dx \tag{3}$$

Upon integrating by parts this becomes

$$\frac{d\mathcal{F}}{dt} = -\sigma \int \frac{\partial}{\partial x}\left(\frac{\partial h/\partial x}{\sqrt{1 + (\partial h/\partial x)^2}}\right)\frac{\partial h}{\partial t}dx = -\sigma \int \kappa(x)\frac{\partial h}{\partial t}dx \tag{4}$$

where $\kappa(x)$ is the interface curvature. The integration by parts requires boundary conditions for h. Either h fixed or contact angle fixed can be used.

The evolution of h is given by the kinematic equation

$$\frac{\partial h}{\partial t} = v - u\frac{\partial h}{\partial x} \tag{5}$$

and the evolution of $\mathcal{K}$ is given by

$$\frac{d\mathcal{K}}{dt} = \int\int (F_x u + F_y v) dx dy \tag{6}$$

The effective part of $\vec{F}$, since the pressure in an incompressible fluid doesn't change total kinetic energy, is due only to surface tension forcing. The total energy $\mathcal{E}$ remains constant. This gives

$$\int\int \left(\left(F_x + \sigma\delta(y-h)\kappa(x)\frac{\partial h}{\partial x}\right)u + \left(F_y - \sigma\delta(y-h)\kappa(x)\right)v\right)dxdy = 0 \tag{7}$$

This is true for arbitrary incompressible $\vec{u}$ and therefore $\vec{F}$ must, to within the gradient of an arbitrary scalar field, be

$$F_x = -\sigma\delta(y-h)\kappa(x)\frac{\partial h}{\partial x} \quad ; \quad F_y = \sigma\delta(y-h)\kappa(x) \tag{8}$$

$\vec{F}$ is a vector delta function with magnitude equal to $\sigma\sqrt{1+(\partial h/\partial x)^2}\,\kappa$, the surface tension times the curvature times interface length per dx.

In terms of u and v the evolution of free energy is given by

$$\frac{d\mathcal{F}}{dt} = \sigma \int\int \delta(y-h)\kappa(x)\left(\frac{\partial h}{\partial x}u - v\right)dxdy \tag{9}$$

Comparing to (8), we see that

$$F_x = -\frac{\delta}{\delta u}\frac{d\mathcal{F}}{dt} \quad ; \quad F_y = -\frac{\delta}{\delta v}\frac{d\mathcal{F}}{dt} \tag{10}$$

This relationship is generally applicable to smeared interface models, both continuum and discretized.

3. Variational Derivation of Surface Tension Forces for a Discretized Interface

The simplest case is given by an interface with straight line segments connecting the control points. For this the free energy is

$$\mathcal{F} = \sigma \sum_i \sqrt{(x_{i+1}-x_i)^2 + (y_{i+1}-y_i)^2} \tag{11}$$

The evolution of free energy is given by

$$\frac{d\mathcal{F}}{dt} = \sum_i \frac{\partial\mathcal{F}}{\partial x_i}\frac{dx_i}{dt} + \sum_i \frac{\partial\mathcal{F}}{\partial y_i}\frac{dy_i}{dt} \tag{12}$$

where

$$\frac{\partial \mathcal{F}}{\partial x_i} = \sigma\left(\frac{x_i - x_{i-1}}{S_{i-1/2}} + \frac{x_i - x_{i+1}}{S_{i+1/2}}\right) \tag{13}$$

and similarly for $\partial\mathcal{F}/\partial y_i$. $S_{m+1/2}$ denotes $\sqrt{(x_{m+1} - x_m)^2 + (y_{m+1} - y_m)^2}$.

If the discretized interface is modeled as being in a continuum fluid then $d\vec{x}_i/dt = \vec{u}(x_i, y_i)$. Application of (10) yields

$$\vec{F} = -\delta(x - x_i, y - y_i)\sigma\left(\frac{\vec{x}_i - \vec{x}_{i-1}}{S_{i-1/2}} + \frac{\vec{x}_i - \vec{x}_{i+1}}{S_{i+1/2}}\right) \tag{14}$$

delta-function forces located at the control points. If the fluid velocity field is discretized on a fixed grid then movement of a control point must be determined by an interpolation from nearby velocity grid points. In general, $\dfrac{dx_i}{dt} = \sum_j a_{ij}u_j$, $\dfrac{dy_i}{dt} = \sum_j b_{ij}v_j$. Then

$$\frac{d\mathcal{F}}{dt} = \sum_j \left(\sum_i a_{ij}\frac{\partial \mathcal{F}}{\partial x_i}\right)u_j + \sum_j \left(\sum_i b_{ij}\frac{\partial \mathcal{F}}{\partial y_i}\right)v_j \tag{15}$$

giving

$$F_{x_j} = -\sum_i a_{ij}\frac{\partial \mathcal{F}}{\partial x_i} \quad ; \quad F_{y_j} = -\sum_i b_{ij}\frac{\partial \mathcal{F}}{\partial y_i} \tag{16}$$

We see that the interpolation stencils for calculating grid velocities set the distribution of surface tension forces to the grid. Interpolation of grid velocities to the control points and distribution of forces to the grid cannot be done independently while maintaining energy conservation. If, for smoothness, a wide distribution of surface tension forces to the grid is desired, then a wide stencil should be used for the interpolation of the velocities.

Another device could be used to help ensure smoothness, the use of diffusion or relaxation to prevent small scale jaggedness on the interface. The simplest relaxation process that is downgradient in free energy, and therefore guaranteed smoothing because it moves in the direction of reducing surface area, is

$$\frac{d\vec{x}_i}{dt} = -D\frac{\partial \mathcal{F}}{\partial \vec{x}_i} \tag{17}$$

This, however, works equally on all wavelengths. A process that works mainly on small scales is

$$\frac{d\vec{x}_i}{dt} = D\left(\frac{\partial \mathcal{F}}{\partial \vec{x}_{i-1}} - 2\frac{\partial \mathcal{F}}{\partial \vec{x}_i} + \frac{\partial \mathcal{F}}{\partial \vec{x}_{i+1}}\right) \tag{18}$$

This can be further modified, while remaining proveably downgradient, to meet the constraint of no net change of phase volume.

All the above can be done in three dimensions, though some of the geometric calculations are then much lengthier. The simplest general case is that of control points defining a triangulated surface. Rates of change of triangular areas must be calculated in order to find $d\mathcal{F}/dt$ and the forces.

4. Variational Derivation of Surface Tension Forces for the Continuum Surface Tension Model

The CSF model of Brackbill et al. assumes smeared interfaces. The fluid phases are represented by a field variable that will be taken here to be the color function C. This is assumed to change in a continuous fashion through an interface from 0 to 1. The forces in the fluid are modeled as proportional to the curvature of the color function field times its gradient:

$$\vec{F} = -\sigma\left(\vec{\nabla} \cdot \left(\vec{\nabla}C/|\vec{\nabla}C|\right)\right)\vec{\nabla}C = -\sigma\kappa\vec{\nabla}C \tag{19}$$

The free energy density corresponding to this is $\sigma|\vec{\nabla}C|$. This can be shown using the same approach as in the previous sections. We have

$$\mathcal{F} = \int \sigma|\vec{\nabla}C|dV \tag{20}$$

from which

$$\frac{d\mathcal{F}}{dt} = \sigma \int \frac{\vec{\nabla}C}{|\vec{\nabla}C|} \cdot \vec{\nabla}\left(\frac{\partial C}{\partial t}\right) dV \tag{21}$$

After integration by parts (21) becomes

$$\frac{d\mathcal{F}}{dt} = -\sigma \int \kappa \frac{\partial C}{\partial t} dV \tag{22}$$

For an incompressible fluid

$$\frac{\partial C}{\partial t} = -u\frac{\partial C}{\partial x} - v\frac{\partial C}{\partial y} = -\frac{\partial uC}{\partial x} - \frac{\partial vC}{\partial y} \tag{23}$$

The middle of (23) inserted into (22) immediately leads to the CSF forcing given by (19). Insertion of the right hand side followed by an integration by parts leads to

$$\vec{F} = \sigma C\vec{\nabla}\kappa \tag{24}$$

(24) differs from (19) by the gradient of $\sigma\kappa C$ but this difference is negated by the fluid's incompressibility. For (19) the scalar potential that maintains incompressibility is the true pressure, for (24) it is the pressure plus $\sigma\kappa C$.

There are many possible discrete versions of (19). Since VOF CSF uses a mollified C for computing curvatures I consider here a free energy expressed in terms of M_i, the discrete mollified color function. M_i is formed by a local averaging of C_i; $M_i = \sum_j a_{ij} C_j$, where $\sum_j a_{ij} = 1$. The discrete free energy densities formed from the M_i have the general form

$$f_i = \sigma \sqrt{\sum_j \sum_k \gamma_{ijk} (M_k - M_j)^2} \tag{25}$$

A simple discretization on a square grid is

$$f_{k,l} = \frac{\sigma}{\sqrt{2}h} \sqrt{\sum_{r=\pm 1} (M_{k,l} - M_{k+r,l})^2 + \sum_{s=\pm 1} (M_{k,l} - M_{k,l+s})^2} \tag{26}$$

where $f_{k,l}$ is the free energy density at grid point (x_k, y_l) and h is the grid spacing.

The free energy functional is the sum of the free energy densities times area factors, $\mathcal{F} = \sum_i A_i f_i$. Its variation is

$$\frac{d\mathcal{F}}{dt} = \sum_i A_i \frac{df_i}{dt} = \sum_j \left(\sum_i A_i \frac{\partial f_i}{\partial M_j} \right) \frac{dM_j}{dt} = \sigma \sum_j A_j \kappa_j \frac{dM_j}{dt} \tag{27}$$

$\sum_i A_i \frac{\partial f_i}{\partial M_j} \big/ \sigma A_j$ is the discrete curvature κ_j. As an example, the curvature corresponding to discretization (26) is

$$\kappa_{k,l} = \frac{1}{\sqrt{2}\,h} \sum_{r=\pm 1} \left(\frac{1}{f_{k+r,l}} + \frac{1}{f_{k,l}} \right) \left(M_{k,l} - M_{k+r,l} \right)$$
$$+ \frac{1}{\sqrt{2}\,h} \sum_{s=\pm 1} \left(\frac{1}{f_{k,l+s}} + \frac{1}{f_{k,l}} \right) \left(M_{k,l} - M_{k,l+s} \right) \tag{28}$$

From the relationship between M and C

$$dM_j/dt = \sum_k a_{jk} dC_k/dt \tag{29}$$

The evolution of C_k is

$$\frac{dC_k}{dt} = \sum_m \left(\alpha_{km} \overline{C}_m u_m + \beta_{km} \overline{C}_m v_m \right) \tag{30}$$

(30) is applicable to a general unstructured grid in which the u_m and v_m are located at cell interfaces and the C_k are located at cell centers. The $\overline{C}_m$

are located at the cell interfaces. In the VOF method the $\overline{C}_m$ are calculated via nonlinear interpolations between the C in nearby cells. The summation in m includes just those interfaces m that border the cell k. The α_{km} and β_{km} are zero except for the two ks, say $k_-(m)$ and $k_+(m)$, that are divided by the interface m. The values of $\alpha_{k_+,m}$ and $\beta_{k_-,m}$ depend on interface orientation, interface length, and cell area. Also, if the cells k_- and k_+ have the same area, $\{\alpha_{k_-,m}, \beta_{k_-,m}\} = -\{\alpha_{k_+,m}, \beta_{k_+,m}\}$.

Substituting (30) into (29) and then the result into (27), we obtain

$$\frac{d\mathcal{F}}{dt} = \sigma \sum_m \overline{C}_m \left(\sum_{k=k_-,k_+} \alpha_{km} \sum_j a_{jk} A_j \kappa_j \right) u_m + \sigma \sum_m \overline{C}_m \left(\sum_{k=k_-,k_+} \beta_{km} \sum_j a_{jk} A_j \kappa_j \right) v_m \tag{31}$$

from which

$$F_x = -\sigma \overline{C}_m \sum_{k=k_-,k_+} \alpha_{km} \sum_j a_{jk} A_j \kappa_j = -\sigma \overline{C}_m \sum_{k=k_-,k_+} \alpha_{km} A_k \overline{\kappa}_k \tag{32}$$

$$F_y = -\sigma \overline{C}_m \sum_{k=k_-,k_+} \beta_{km} \sum_j a_{jk} A_j \kappa_j = -\sigma \overline{C}_m \sum_{k=k_-,k_+} \beta_{km} A_k \overline{\kappa}_k \tag{33}$$

at (x_m, y_m). The $\overline{\kappa}_k$ are locally averaged curvatures, $\overline{\kappa}_k = \sum_j a_{jk} A_j \kappa_j \big/ A_k$. (32-33) shows that mollifying C results in a secondary smoothing: u and v are then forced by mollified curvatures. *If a mollified C is used to compute curvature then this curvature must be further mollified in order to conserve energy. This mollified curvature field is what is used to force $\vec{u}$.*

An additional smoothing technique is available, the use of smeared or averaged velocities to convect the C. Instead of (30) one can use

$$\frac{dC_k}{dt} = \sum_m \left(\alpha_{km} \overline{C}_m \overline{u}_m + \beta_{km} \overline{C}_m \overline{v}_m \right) \tag{34}$$

where $\{\overline{u}_m, \overline{v}_m\} = \{\sum_n b_{mn} u_n, \sum_n b_{mn} v_n\}$, $\sum_n b_{mn} = 1$. $\overline{u}$ and $\overline{v}$ must satisfy the discretized continuity equation. A diffusive process that maintains continuity constraints can be used to derive and define the smoothed velocities. The surface tension forcing that results is

$$F_x = -\sigma \sum_m b_{mn} \left(\overline{C}_m \sum_{k=k_-,k_+} \alpha_{km} A_k \overline{\kappa}_k \right) \tag{35}$$

$$F_y = -\sigma \sum_m b_{mn} \left(\overline{C}_m \sum_{k=k_-,k_+} \beta_{km} A_k \overline{\kappa}_k \right) \tag{36}$$

at (x_n, y_n). This constitutes a local averaging of the forces given in (32-33). As with the distributed force model, this averaging mirrors the averaging given the velocities.

Before closing, one minor difficulty with the CSF method needs discussing, the indefiniteness of the curvature in locations where gradients are zero. Where this occurs curvature calculations must be made using some limiting procedure. In discretized systems a zero gradient means the discretized free energy is zero. This results in divide-by-zero problems when calculating curvatures. Numerators are also zero so the curvatures are undefined.

In the continuum case the problem is resolved by using the momentum forcing given by equation (19). (19) gives either a zero or a smoothly varying non-zero forcing when the gradient is zero. In the latter case, which holds at maxima and minima with non-zero second derivatives, the local forcing can be calculated by interpolation from nearby points.

The form of the surface tension forcing (32-33) for the VOF CSF model is equivalent to that of (24). It is difficult to transform it to the form of (19) because of the nonlinear calculation of the $\overline{C}_m$. There are, however, at least two alternative ways to resolve the curvature issue.

One way is to use free energy densities for the curvature calculation that are averaged over each time step. An averaged free energy density will be zero only if it is zero initially and remains unchanged throughout the time step. In this case, however, it can be dropped from the calculation - to be specific, from the first summation in (27) - because it makes no contribution to free energy change during the time step and can thus make none to changes in momentum.

Another, perhaps easier, way to resolve the problem is to use a modified free energy density. One possibility is

$$f = \sigma\left(\sqrt{|\vec{\nabla} M|^2 + \epsilon^2} - \epsilon\right) \tag{37}$$

The modified curvature corresponding to this is zero whenever the gradient is zero for any positive ϵ. A non-zero ϵ also gives smoother surface tension calculations. It introduces an $O(\epsilon)$ error into the calculation of the surface energy but this can be made small compared to the error already introduced by interface smearing.

References

Anderson, D. M., and McFadden, G. B., "A Diffuse-Interface Description of Fluid Systems, National Institute of Standards and Technology," Gaithersburg, MD, Report No. NISTER 5887, 1996.

Antanovskii, L. K., "A Phase Field Model of Capillarity," *Phys. Fluids*, Vol. 7, 1995, pp. 747-753.

Brackbill, J., Kothe, D. B., and Zemach, C., "A Continuum Method for Modeling Surface Tension," *J. Comp. Phys.*, Vol. 100, 1992, pp. 747-753.

Chella, R. and Viñals, J., "Mixing of a Two-Phase Fluid by Cavity Flow," *Phys. Rev. E.*, Vol. 53, 1996, pp. 3832-3840.

Jacqmin, D., "Three-Dimensional Computations of Droplet Collisions, Coalescence, and Droplet/Wall Interactions using a Continuum Surface Tension Method," AIAA 95-0833, presented at the 33rd Aerospace Sciences Meeting, Reno, NV, 1995.

Jacqmin, D., "An Energy Approach to the Continuum Surface Tension Method: Applications to Droplet Coalescences and Droplet/Wall Interactions," Proceeedings of the 1995 ASME IMECE, San Francisco, CA, 1995.

Jacqmin, D., "An Energy Approach to the Continuum Surface Tension Method," AIAA 96-0858, presented at the 34th Aerospace Sciences Meeting, Reno, NV, 1996.

Jasnow, D. and Viñals, J., "Coarse-Grained Description of Thermo-Capillary Flow," *Phys. Fluids*, Vol. 8, 1996, pp. 660-669.

Kothe, D. B., Rider, W. J., Mosso, S. J., and Brock, J. S., "Volume Tracking of Interfaces Having Surface Tension in Two and Three Dimensions," AIAA 96-0859, presented at the 34th Aerospace Sciences Meeting, Reno, NV, 1996.

Nadiga, B. T. and Zaleski, S., "Investigations of a Two-Phase Fluid Model," to appear in *Eur. J. Mech., B/Fluids*, Vol. 15, 1996.

Lafaurie, B., Nardone, C., Scardovelli, R., Zaleski, S., and Zanetti, G., "Modeling Merging and Fragmentation in Multiphase Flows with SURFER," *J. Comp. Phys.*, Vol. 113, 1994, pp. 134-147.

Rider, W. J., Kothe, D. B., Moddo, S. J., and Cerutti, J. H., "Accurate Solution Algorithms for Incompressible Multiphase Flows," AIAA 95-0699, presented at the 33rd Aerospace Sciences Meeting, Reno, NV, 1995.

Sussman, M., Fatemi, E., Smeraka, P., and Osher, S. J., "An Improved Level Set Method for Incompressible Two-Phase Flow," to appear in *J. Comp. and Fluids*, 1996.

Sussman, M., Smeraka, P., and Osher, S. J., "A Level Set Approach for Computing Solutions to Incompressible Two-Phase Flow," *J. Comp. Phys.*, Vol. 114, 1994, pp. 146-159.

Unverdi, S. 0. and Tryggvason, G., "A Front-Tracking Method for Viscous, Incompressible, Multi-Fluid Flows," *J. Comp. Phys.*, Vol. 100, 1992, pp. 25-37.

van der Waals, J. D., "The Thermodynamic Theory of Capillarity Flow under the Hypothesis of a Continuous Variation of Density," *Verhandel/Konink. Akad. Weten.*, Vol. 1, 1893.