Title:

ACCURATE AND ROBUST METHODS FOR VARIABLE DENSITY
INCOMPRESSIBLE FLOWS WITH DISCONTINUITIES

Author(s):

W. J. Rider, D. B. Kothe, E. G. Puckett

Submitted to:

NASA ICASE Workshop on Challenges & Barrier
in Computational Fluids Dynamics
Langley, VA, Aug. 5-7, 1996

MASTER

Los Alamos
NATIONAL LABORATORY

Los Alamos National Laboratory, an affirmative action/equal opportunity employer, is operated by the University of California for the U.S. Department of Energy under contract W-7405-ENG-36. By acceptance of this article, the publisher recognizes that the U.S. Government retains a nonexclusive, royalty-free license to publish or reproduce the published form of this contribution, or to allow others to do so, for U.S. Government purposes. The Los Alamos National Laboratory requests that the publisher identify this article as work performed under the auspices of the U.S. Department of Energy.

DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED

DISCLAIMER

**Portions of this document may be illegible
in electronic image products. Images are
produced from the best available original
document.**

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

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
University of California, Davis
Mathematics Department
Davis, CA 95616

We are interested in the solution of incompressible flows which are characterized by large density variations, interfacial physics, arbitrary material topologies and strong vortical content. The issues present in constant density incompressible flow are exacerbated by the presence of density discontinuities. A much greater premium requirement is placed the positivity of computed quantities. The mechanism of baroclinic vorticity generation exists ($\nabla \rho \times \nabla p$) to further complicate the physics.

1. Introduction

We present the key components of algorithms for computing accurate and robust solutions to incompressible flows with discontinuities between immiscible fluids. A classical example of this is water and air. For instance, the problem of the interaction of a drop with a pool of water can be modeled using our methods. Another example is the process of filling a mold with a molten metal alloy. Yet another is the operation of an inkjet printer head.

Accuracy is defined as the quality of only deviating slightly from fact. For our purposes, the definition is refined as the measured error estimates for a given solution. There is a distinction between order of accuracy and numerical accuracy. For reasonable grid resolution methods with a higher order of accuracy can have a significantly larger numerical error. This is often the case for advectively dominated physics when comparing high-

order centered methods with upstream-centered methods. This naturally leads to our next definition.

Fidelity is defined as exact correspondence with fact. Here this does not necessarily mean accuracy, but rather that the solutions being physically meaningful. A method is said to have high fidelity when it produces solutions that are accurate relative to the computational resources applied to them. The use of various slope or flux limiters result in high fidelity solutions through removing unphysical features from the flow solution. Another example of high fidelity methods are interface tracking mechanisms which remove the smearing of interfaces commonly associated with shock capturing based approaches.

Robustness is the property of being powerfully built or sturdy. A robust method will not fail in a catastrophic manner, but rather “degrade gracefully.” The degree to which the degradation is graceful is subject to interpretation. A method should continue to produce physically reasonable results beyond the point where accuracy is expected or achieved. Robustness also plays a role with the solution of linear systems arising from the discrete pressure equation. The economical solution of this equation is a form of robustness. This is to say that a solver which is too costly for the given level of resolution has failed.

In the next several sections we will focus on the key elements in our incompressible flow solvers. Section 2 will introduce the projection algorithms used to discretize 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 the advective derivative are discussed in Section 4. Section 5 follows with an introduction and discussion of the continuum surface force method (CSF). We then give a set of sample applications to amplify our arguments. Finally, we will conclude with various outstanding issues in the solution of variable density incompressible flows in Section 7.

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, $\mathbf{u} = (u, v)^T$ by some convenient means disregarding 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, $\mathcal{P}$, which has the effect

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

The curl-free portion will be denoted by the gradient of a potential, $\nabla\varphi$. This decomposition can be written

$$\mathbf{u} = \mathbf{u}^d + \nabla\varphi, \quad (1)$$

which is the key for computing solutions to incompressible flows.

The velocity decomposition for variable density flows is written

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

or

$$\mathbf{u} = \mathbf{u}^d + \sigma\nabla\varphi, \quad (2)$$

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

$$\nabla \cdot \mathbf{u} = \nabla \cdot \sigma\nabla\varphi. \quad (3)$$

Once this equation is solved, the correction equation to the velocity is

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

2.1. FRACTIONAL STEP METHODS AND PROJECTIONS

Our algorithm will consist of two steps: a predictor step where the solenoidal nature of the velocity field is ignored and a corrector step using the projection to force the velocity to be solenoidal. This is often referred to as a "fractional step method." In the first step, the velocity is computed solving the motion equations as convection-diffusion equations

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

where $\sigma_{i,j}^{n+\frac{1}{2}} = (\sigma_{i,j}^n + \sigma_{i,j}^{n+1})/2$, $\mathbf{f}_{i,j}^{n+\frac{1}{2}} = (\mathbf{f}_{i,j}^n + \mathbf{f}_{i,j}^{n+1})/2$, or $\mathbf{f}_{i,j}^{n+\frac{1}{2}} = \mathbf{f}(\mathbf{x}_{i,j}, t^{n+\frac{1}{2}})$ and $L_{i,j}$ is the discrete Laplacian. This provides a nominally second-order discretization. The advection can be discretised with an unsplit high-order Godunov method (BCG89; Col90).

Several alternative formulations of the projection are possible. By removing the gradient of pressure from (5), the projection solves for the pressure rather than an increment in pressure. The form of the discrete

divergence can be chosen several ways: to project the new time velocity, $\mathbf{u}^{*,n+1}$ or the difference $\mathbf{u}^{*,n+1} - \mathbf{u}^n$. Nominally these differences should make no difference, but experience has shown otherwise for both exact and approximate projections, as we will discuss below.

In order for this discussion above to be useful for computing flows, we must define a discrete analog to the operators above. These are standard centered differences on a regular grid. The substitution of this difference approximation into the projection framework defines the operators for an "exact" projection. With exact projections the discrete divergence is solved to be zero (within some small tolerance for the solution of the resulting linear equations).

The "approximate projections" introduced in (ABS93) have somewhat different properties in that they do not have the properties of a projection at a discrete level, but rather are discretizations of the continuous projection operator. The discrete divergence is not exactly zero, but is rather a function of truncation error.

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 commented in (ABS93; LBC93). The decoupling of the pressure fields interacts poorly with chemical source terms leading to instabilities. Additionally, the local decoupling makes multigrid techniques cumbersome (How93) as well as hampering the implementation of adaptive grid techniques (ABCH93; How93).

We define discrete methods based on continuous projections, rather than demanding that the discrete system algebraically match the conditions for being a projection. Thus, a straightforward means to discretize each operator ($\nabla \cdot$, ∇ and $\nabla \cdot \sigma \nabla$) will be chosen.

2.2. ROBUST PROJECTIONS

In approximate projections, the velocity divergence is not constrained to be zero to some small tolerance, robust computations can be difficult. We will demonstrate this on a single test problem and show how to improve matters. This will also demonstrate that aside from the efficiency issues related to linear algebra, the exact projections are superior. Additionally, the exact projections can benefit from the experience gained with approximate projections.

The primary problem with approximate projections is that components of the discrete operators with non-trivial nullspaces pollute the solution. This noise can be controlled in two manners, the filtering of the modes (Rid94b) and the proper formulation of the approximate projec-

tion (Rid94a). Without these steps the approximate projections can fail spectacularly.

There are two forms of filters used: the explicit addition of a high order viscosity to the spurious modes, or forming an iterated projection on a different discrete stencil to damp the oscillations. These filters are most effective when used together.

The issue of formulation of the projection can have a large impact on the evolution of the discrete divergence in time. If the elliptic equation's right hand side is

$$\nabla \cdot (\mathbf{u}^{*,n+1} - \mathbf{u}^2),$$

the discrete divergence errors sum in time. On the other hand, if the right hand side is

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

the discrete divergence errors are local in time. Having the errors local in time is certainly preferable.

The following test problem that will illuminate these subtle issues. ^t The domain is $[0, 1] \times [0, 1]$ and no flow conditions at all boundaries, a circle of heavy fluid is placed at $(0.5, 0.75)$ with a radius of 0.15, and unit gravity pointing downward. The heavy fluid is 1000 times more dense than the background fluid. The flow is evolved until $t = 1$ using the Euler equations (inviscid). We use the high-order Godunov methods and the unsplit PLIC algorithm discussed later for advective transport. We should also note that each of the methods used below will demonstrate second-order convergence on problems with sufficient regularity.

We start by establishing a the baseline performance of the methods without filters and using the standard formulation¹. Figure 1 gives the solution found using the standard exact and approximate projection algorithms. Both methods produce a velocity field that has significant spurious structures located above the position of the drop. The exact projection solution also shows some spurious droplet breakup and a lack of symmetry in the solution. As we will see this is largely the product of the formulation of the projection (solving for an increment of pressure and projecting the difference in velocity).

Applying the change in formulation of the projection improves the solutions significantly so that they are qualitatively believable. Figure 2a shows both symmetry, but also a physically realistic velocity field. Figure 2b shows the effect of using both the change in formulation and filters with an ap-

Figure 1. Our base calculation using the exact and approximate projections. The droplet outline and the velocity vectors are shown.

Figure 2. The end product of the changes in the projection formulation and the filtering of approximate projections.

proximate projection. The effects of the decoupling evident in Figure 1 is effectively suppressed.

3. Linear Algebra

The cost of a projection method is typically dominated by the solution of the elliptic pressure equation. Designing an efficient and scalable method for solving this system of linear equations is of paramount importance. We have used three methods in our study: a preconditioned conjugate gradient method, a multigrid method and a multigrid preconditioned conjugate gradient method. Based on our results, the multigrid preconditioning coupled with conjugate gradient is the most effective method.

3.1. CONJUGATE GRADIENT METHODS

Because the projection and approximate projection produce symmetric semi-positive-definite and positive-definite systems of linear equations, we can use conjugate gradient methods (GL89). We will also employ preconditioning to improve convergence. Typically we can use an incomplete Cholesky method or some iterative procedure.

For the pressure equation we have noticed several properties of the solution. The preconditioned conjugate gradient method appears to be extremely robust having never failed in our experience. The method is also general, applying to any of our exact or approximate projections. On the other hand, the amount of work needed to solve the system of equations grows as $N^{3/2}$, where N is the total number of grid points. Therefore the method takes an ever increasing proportion of the CPU time as the grid is refined.

3.2. MULTIGRID METHODS

The linear systems arising with approximate projections can also be solved via a multigrid algorithm. A good basic reference on the multigrid method is Briggs' book (Bri87). The reason for choosing multigrid is the optimality of the scaling of this algorithm.

The operation count on classical linear algebra solvers like LU decomposition scale with the number of equations, N^3 . This can be improved to N^2 for banded solvers that take advantage of the structure of the linear system. As we have stated conjugate gradient methods scale like $N^{3/2}$,

¹The standard formulation solves for an increment in pressure and uses $\nabla \cdot (\mathbf{u}^{*,n+1} - \mathbf{u})$ on the right hand side of the elliptic equation

whereas multigrid scales linearly with N . Thus, as N grows larger, eventually multigrid (where it works) will be the fastest route to a solution.

One of the most important tasks in setting up the multigrid algorithm is the process of approximating the linear equations on the coarse grid. One approach would be to use intergrid transfer functions to define a variational or Galerkin coarse grid operator. Because of the complication and expense of this step, we have implemented a simpler, less expensive approach based in part on suggestions in (LLM92). The operator remains the same, as do the boundary conditions on the coarse grids, but σ must be defined at each level. The basic idea is to construct coarse grid approximations that give the same average value in a cell of the quantity $\sigma \nabla \varphi$ as on the finer levels. In using the cell-centered multigrid framework, the control volume derivation of the equations comes quite naturally.

We find that this procedure produces a multigrid algorithm that works in most cases, but fails on occasion with flows with large density variations and a severe topology change. This type of flow is exemplified by a drop colliding with the surface of a pool.

3.3. MULTIGRID PRECONDITIONED CONJUGATE GRADIENT

Our current approach to the problems with robustness was to use a symmetric multigrid algorithm to compute the preconditioner for the standard conjugate gradient algorithm. The method was first proposed by Tatebe (Tat93). Subsequent experimentation has proved the utility of combining these two methods. Generally the combined method scales like a multigrid algorithm, but occasionally the method scales like a conjugate gradient method. Nevertheless, we have found it to be robust. Ultimately, one would like to have a method that is both robust and uniformly exhibits multigrid scaling.

4. Advection and Interface Tracking

4.1. HIGH-ORDER GODUNOV METHODS

Here we discuss the basic principles of our advection method. These methods were introduced in a series of papers (vL84; BDS88; Col90). Our methods are based most closely on Colella's formulation as modified for incompressible flow solvers based on a projection method (BCG89). These methods are all unsplit, thus the full multidimensional solution is updated in a single step. This is important because the incompressibility constraint is multidimensional and the solver should reflect the intrinsic coupling of the velocity field.

This is equivalent to approximating the dependent variables at cell-edges in a time-centered manner. It is important to include all of the solution physics in time-centering (including a projection). While the details of the methods are given in the above stated reference, several important contributions have been made recently to the use of such methods. Brown and Minion discuss the nature of solutions when physical phenomena is under-resolved (BM95), and Minion has suggested some improvements to the standard formulation of the method to enhance stability (Min96).

4.2. VOLUME-OF-FLUID METHODS

The essential features of the VOF interface tracking method are as follows. First, the initial (known) fluid interface geometry is used to compute 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. This step is motivated by the desire to conserve volume with the algorithm. If the exact interface information is retained there are an insufficient number of degrees of freedom (without using a beyond linear interface) to also conserve mass.

Interfaces are subsequently "tracked" by evolving fluid volumes in time with the solution of a standard convection equation. Volume fractions result from normalization of fluid volumes relative to the cell control volume. 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 convective term in the volume evolution equation. Typical implementations of these algorithms are one-dimensional, with multi-dimensionality obtained through operator splitting. The assumed interface geometry, interface reconstruction, and volume flux calculation typically comprise the unique features of a given VOF method.

The approach we use is straightforward and provides a algorithmic framework that is simple and extensible. Furthermore, it allows us to tap into a significant amount of work done in the area of computational geometry (O'R93). A full accounting of this approach to the design of VOF methods is given in (RK94). The VOF method and its PLIC variant in particular can be broken up into a set of geometric primitives that lead to a concise statement of the method. Pilliod and Puckett (PP) give an excellent review, analysis and discussion of the standard methods for interface reconstruction.

5. Surface Forces Calculations

5.1. THE CONTINUUM SURFACE FORCE METHOD

The basic premise of the CSF (continuum surface force) method (BKZ92) is to model physical processes specific to and localized at fluid interfaces (e.g, surface tension, phase change) by applying the process to fluid elements everywhere within the 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 (LL94) whose origin dates back to the pioneering work of Peskin (Pes77). 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 (KMT91; KM92), where complex interface phenomena such as breakup and coalescence have been modeled.

For surface tension, the relevant surface physics is a force per unit area arising from local interface curvature and local (tangential) variations in the surface tension coefficient. In the following, we briefly review the CSF model and discuss important outstanding issues that are the focus of our current efforts.

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

$$\mathbf{F}_s = \mathbf{f}_s \delta_s. \quad (5)$$

Here δ_s is a surface delta function and $\mathbf{f}_s$ is the surface tension force per unit interfacial area (BKZ92):

$$\mathbf{f}_s = \sigma \kappa \hat{\mathbf{n}} + \nabla_s \sigma, \quad (6)$$

where σ 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 [Wea]:

$$\kappa = -(\nabla \cdot \hat{\mathbf{n}}). \quad (7)$$

The first term in (6) 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 σ .

Since discontinuous interfaces are represented as finite thickness transition regions within which fluid volume fractions vary smoothly from zero

to one, the surface delta function should be nonzero only within these transition regions. The surface delta function was proposed in the original CSF model to be (BKZ92)

$$\delta_s = \frac{|\nabla c|}{[c]} = |\mathbf{n}| \quad (8)$$

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 fluid volume fractions serve as the color function in this work.

If a wide stencil is used for $\hat{\mathbf{n}}$ in (8), then δ_s will be nonzero in noninterface cells that are in close proximity to the interface. We are currently forcing δ_s to be zero in these cells, which causes the CSF to be 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 $\kappa h \rightarrow 0$. The integral effect of the CSF should be correct, in that its line integral directed normally through the interface transition region should be approximately equal to the pressure jump (BKZ92).

A density-scaling of the CSF, given by,

$$\tilde{\mathbf{F}}_s = \frac{\rho}{\langle \rho \rangle} \mathbf{F}_s, \quad (9)$$

where ρ is the local density and $\langle \rho \rangle$ is an average interface density, was found to improve the CSF method's ability to model surface tension on fine meshes at high density ratio (≥ 10) interfaces (KMT91).

Despite the success of the CSF model and other related immersed interface methods, we are currently addressed several outstanding issues. If these issues can be resolved adequately, a wider range of surface tension-driven flows will be reliably modeled by the CSF method.

We have found, as have others (AP), that improved forms for δ_s present one of the key outstanding issues for the CSF model. Desired forms for should display attractive convergence and smoothness properties. Our current numerical results are very sensitive to the form used for δ_s , indicating that the quality of CSF model (or any immersed interface model) relies heavily on the quality of the form used for δ_s . Recent results by Aleinov and Puckett (AP) motivate the use of higher order kernels such as the Peskin (Pes77) or Nordmark (Nor91) kernel. We will pursue these (and other) kernels, as their impact on the CSF model is likely to be significant.

As suggested in (BKZ92), smoother variations in κ generally result if a mollified volume fraction is used to compute the face normals in (??). A variety of smoothing operators (e.g., citebkz- spline or point-Jacobi) have been

found to give the desired results, which is the mitigation of high wavenumber contributions to κ . Aleinov and Puckett use the Nordmark kernel for both a smoothing operator and δ_s , which is attractive. It is clear that a smoothing operator (kernel) is desirable, but the type and extent of the operator may very well depend upon the interfacial flow being modeled.

Smoothing should be used with caution because it is paradoxical: too little smoothing allows noise in the curvature field, while too much allows high frequency capillary waves to persist unphysically. Nevertheless, smoothing of the volume fraction field used to estimate was the preferred choice in recent numerical studies. Although some examples of the effects of smoothing can be found in (BKZ92), it still warrants further investigation, especially as it relates to convergence and consistency of the CSF model.

The representation of surface tension in the CSF model as an explicit force is linearly stable only for time steps smaller than the maximum allowable value time step necessary to resolve the propagation of capillary waves (BKZ92). This constraint can be restrictive, especially when fluid interfaces undergo topology changes such as pinch-off. Since $\delta t_s \propto h^{3/2}/\sigma$, an explicit CSF can be restrictive when h is small or σ is high. An implicit treatment would alleviate this constraint, and preliminary work (Wil93) indicates this is possible within the CSF methodology. Implicit curvature algorithms have also been recently introduced for boundary integral schemes (HLS94). An implicit extension of the CSF model (currently under investigation) will expand its applicability.

6. Applications

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 $24 \times 24 \times 30$ cubical cells (having unit width). The box is initially filled with a quiescent background fluid having a density ten times less than a filling fluid. At time zero, filling fluid (e.g., the molten metal) is injected with a velocity $\mathbf{v}$ of $(0.0, 0.0, -88.6)$ through a hole in the box located at $0 \leq x, y \leq 5$ and $z = 30$ (the top corner). The background fluid is allowed to escape through another hole located at $19 \leq x, y \leq 24$ and $z = 30$ (the opposite top corner). Gravity $\mathbf{g}$ is taken to be $(0.0, 0.0, -980.6)$, 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. We wish to follow the filling dynamics (as inferred from the interface topology) up to a time of 3.0, which is about halfway through the total filling time of 7.8.

As is evident from Figure 3a-d which depict the background/filling fluid interface topology at times of 0.2, 1.0, 2.0, and 3.0, this idealized calcula-

tion presents an energetic and rigorous test of our ability to model complex topology free surface flows encountered in the mold-filling process. For this simulation, solutions to the incompressible Euler equations are obtained with a cell-centered, variable-density approximate projection method. Interfaces are tracked with our piecewise-planar VOF method.

Applying the change in formulation of the projection

7. Barriers and Challenges

References

- A. S. Almgren, J. B. Bell, P. Colella, and L. H. Howell. An adaptive projection method for the incompressible Euler equations. In J. L. Thomas, editor, *Proceedings of the AIAA Eleventh Computational Fluid Dynamics Conference*, pages 530–539, 1993. AIAA Paper 93–3345.
- A. S. Almgren, J. B. Bell, and W. G. Szymczak. A numerical method for the incompressible Navier-Stokes equations based on an approximate projection. *SIAM Journal of Scientific Computing*, 17:358–369, 1993.
- I. Aleinov and E. G. Puckett. “Computing Surface Tension with High-Order Kernels,” presented at the 6th International Symposium on Computational Fluid Dynamics, 1995.
- J. B. Bell, P. Colella, and H. M. Glaz. A second-order projection method of the incompressible Navier-Stokes equations. *Journal of Computational Physics*, 85:257–283, 1989.
- J. B. Bell, C. N. Dawson, and G. R. Shubin. An unsplit, higher order Godunov method for scalar conservation laws in multiple dimensions. *Journal of Computational Physics*, 74:1–24, 1988.
- J. U. Brackbill, D. B. Kothe, and C. Zemach. A continuum method for modeling surface tension. *Journal of Computational Physics*, 100:335–354, 1992.
- D. L. Brown and M. L. Minion. Performance of underresolved two-dimensional incompressible flow simulations. *Journal of Computational Physics*, 122:165–183, 1995.
- W. L. Briggs. *A Multigrid Tutorial*. SIAM, 1987.
- P. Colella. Multidimensional upwind methods for hyperbolic conservation laws. *Journal of Computational Physics*, 87:171–200, 1990.
- G. H. Golub and C. F. Van Loan. *Matrix Computations*. Johns Hopkins University Press, 1989.
- T.Y. Hou, J.S. Lowengrub, and M.J. Shelley. Removing the stiffness from interfacial flows with surface tension. *Journal of Computational Physics*, 114:312–338, 1994.
- L. H. Howell. A multilevel adaptive projection method for unsteady incompressible flow. In N. D. Melson, T. A. Manteuffel, and S. F. McCormick, editors, *Proceedings of the Sixth Copper Mountain Conference on Multigrid Methods*, pages 243–257, 1993.
- D. B. Kothe and R. C. Mjolsness. Ripple: A new model for incompressible flows with free surfaces. *AIAA Journal*, 30:2694–2700, 1992.
- D. B. Kothe, R. C. Mjolsness, and M. D. Torrey. Ripple: A computer program for

(a) $t = 0.2$ (b) $t = 1.0$ (c) $t = 2.0$ (d) $t = 3.0$

Figure 3. Box filling simulation using Telluride,

- incompressible flows with free surfaces. Technical Report LA-12007-MS, Los Alamos National Laboratory, 1991.
- M. Lai, J. B. Bell, and P. Colella. A projection method for combustion in the zero Mach number limit. In J. L. Thomas, editor, *Proceedings of the AIAA Eleventh Computational Fluid Dynamics Conference*, pages 776-783, 1993. AIAA Paper 93-3369.
- R. J. Leveque and Z. Li. The immersed interface method for elliptic equations with discontinuous coefficients and singular sources. *SIAM Journal on Numerical Analysis*, 31:1019-1044, 1994.
- C. Liu, Z. Liu, and S. McCormick. An efficient multigrid scheme for elliptic equations with discontinuous coefficients. *Communications in Applied Numerical Methods*, 8:621-631, 1992.
- M. L. Minion. On the stability of Godunov projection methods for incompressible flow. *Journal of Computational Physics*, 123:435-449, 1996.
- H. O. Nordmark. Rezoning for higher order vortex methods. *Journal of Computational Physics*, 97:366-397, 1991.
- J. O'Rourke. *Computational Geometry in C*. Cambridge, 1993.
- C. S. Peskin. Numerical analysis of blood flow in the heart. *Journal of Computational Physics*, 25:220-252, 1977.
- J. E. Pilliod, Jr. and E. G. Puckett. Second-Order Volume-of-Fluid Algorithms for Tracking Material Interfaces, In preparation.
- W. J. Rider. 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.c3.lanl.gov/~wjr/pubs.html>.
- W. J. Rider. 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.c3.lanl.gov/~wjr/pubs.html>.
- W. J. Rider and D. B. Kothe. A systematic development of second-order volume-of-fluid methods in two dimensions. Technical Report LA-UR-96-2375, Los Alamos National Laboratory, 1994. Available on World Wide Web at <http://www.c3.lanl.gov/~wjr/pubs.html>.
- O. Tatebe. The multigrid preconditioned conjugate gradient method. In N. D. Melson, T. A. Manteuffel, and S. F. McCormick, editors, *Proceedings of the Sixth Copper Mountain Conference on Multigrid Methods*, pages 621-634, 1993.
- B. van Leer. Multidimensional explicit difference schemes for hyperbolic conservation laws. In R. Glowinski and J.-L. Lions, editors, *Computing Methods in Applied Sciences and Engineering VI*, pages 493-497, 1984.
- T. L. Williams. An implicit surface tension model. Master's thesis, Memphis State University, 1993.