LA-UR-95- 1145

Conf-950634--1

TITLE: A COMPARISON OF INTERFACE TRACKING METHODS

AUTHOR(S): Douglas B. Kothe, T-3
William J. Rider, CIC-3

SUBMITTED TO: *12th AIAA CFD Conference, June 19–22, 1995, San Diego, California*

DISCLAIMER

A Comparison of Interface Tracking Methods. *

Douglas B. Kothe† and William J. Rider‡

March 27, 1995

§

1 Introduction

In this paper we provide a direct comparison of several important algorithms designed to track fluid interfaces. In the process we propose improved criteria by which these methods are to be judged. We compare and contrast the behavior of the following interface tracking methods: high order monotone capturing schemes, level set methods, volume-of-fluid (VOF) methods, and particle-based (particle-in-cell, or PIC) methods. We compare these methods by first applying a set of standard test problems, then by applying a new set of enhanced problems designed to expose the limitations and weaknesses of each method. We find that the properties of these methods are not adequately assessed until they are tested with flows having spatial and temporal vorticity gradients.

Our results indicate that the particle-based methods are easily the most accurate of those tested. Their practical use, however, is often hampered by their memory and CPU requirements. Particle-based methods employing particles only along interfaces also have difficulty dealing with gross topology changes. Full PIC methods, on the other hand, do not in general have topology restrictions. Following the particle-based methods are VOF volume tracking methods, which are reasonably accurate, physically based, robust, low in cost, and relatively easy to implement. Recent enhancements to the VOF methods using multidimensional interface reconstruction and improved advection provide excellent results on a wide range of test problems.

*This work performed under the auspices of the U.S. Department of Energy by Los Alamos National Laboratory under Contract W-7405-ENG-36.

§As a rough draft, this paper is not to be redistributed or referenced without permission of the authors.

†Fluid Dynamics Group, Los Alamos National Laboratory, Mail Stop B216, Los Alamos, New Mexico, 87545, USA, E-mail:dbk@lanl.gov

‡Computer Research and Applications Group, Los Alamos National Laboratory, Mail Stop B265, Los Alamos, New Mexico, 87545, USA, E-mail:wjr@lanl.gov

The performance of level set methods methods fall short of VOF methods, but is slightly better than conventional capturing methods. We find a number of outstanding problems with level set methods that could, without resolution, prohibit their practical use. Foremost among these is a lack of mass conservation, a desirable property that is not currently a constraint on the algorithm as published in [1, 2, 3]. In addition, the level set methods degrade considerably when interfaces possess high curvature relative to the mesh spacing (e.g., sharp corners). While topology changes (tearing or merging of interfaces) are treated naturally by level set methods, the reinitialization scheme needed for maintaining solution quality as a result becomes prohibitively expensive. Capturing methods do not have the same difficulties, but instead suffer from excessive (and frequently unacceptable) smearing of interfaces.

Just as shocks dominate high-speed flow, many practical problems in low-speed flow are dominated by interfaces between fluids. Although the prototypical interface is between immiscible fluids, interfaces can in general represent any abrupt change in fluid quantities. They can be created and propagate internal to a fluid, as in solidification, condensation, fracture, or porosity grow, or can disappear, as in bubble collapse. Frequently special physics occur along interfaces. Typical examples are phase change, surface tension, wall adhesion, and surfactant diffusion.

2 Interface Tracking Methods

Here we briefly introduce each of the methods we test. All of the methods in this paper solve the equation

$$\frac{\partial f}{\partial t} + \mathbf{u} \cdot \boldsymbol{\nabla} f = 0, \tag{1a}$$

where f is some scalar carrying interface or "color" information. An equivalent equation for incompressible flows is

$$\frac{\partial f}{\partial t} + \nabla \cdot (\mathbf{u} f) = 0, \tag{1b}$$

since $\nabla \cdot \mathbf{u} = 0$.

Our particle method is perhaps the most straightforward, drawing upon recent advances in PIC algorithms [4]. Particles are assigned a mass according to the density of the fluid in which they reside and a volume (hence size) according to the interpolation function chosen to interpolate quantities to and from the computational grid. While this method provides a superior multidimensional grid-independent advection scheme, there are as a result some practical difficulties, namely the cost and accuracy associated with interpolating the particle information to an Eulerian grid [5, 6]

Simply discretising (1) with a high resolution finite difference scheme is quite appealing. An advection algorithm is typically an integral part of a flow solver. This is done in the methods presented in [7, 8]. From the advances in high speed flows in the last decade, there are a number of methods that minimize numerical dissipation. An example of this are high-order Godunov methods in particular PPM [9, 10].

Problems with the numerical dissipation (leading to a thickening interface) led researchers to propose an ingenious compromise. The level set methods could be implemented with the same difference techniques already well developed for advection, but without allowing the interface to smear. The interface is defined as the zero level set of a distance function, ϕ, from that interface. Instead of (1), the following equation is solved

$$\frac{\partial \phi}{\partial t} + \mathbf{u} \cdot \boldsymbol{\nabla} \phi = 0.$$

We also study improvements suggested by Sussman, Smereka and Osher [3].

VOF methods have been used for several decades starting at the national laboratories (Livermore [11] and Los Alamos [12] and later Sandia [13]). The earlier work is typified by the SLIC [14] algorithm and the original method with the moniker VOF [15]. In each of these methods the interface is designated as a straight line in a cell defined by the volume of a given fluid in that cell. Youngs [16] improved the general method greatly by allowing the reconstruction of the interface to be multidimensional and linear in nature. Youngs further extended his method to three dimension in [17]. Recently, Pilliod and Puckett have improved the accuracy [18, 19, 20]. Here we refer to this method as the piecewise linear interface calculation (PLIC).

3 Results

In this section we will first describe the implementation of the methods we use, followed by results obtained using these methods.

3.1 Implementation

For interface capturing, we will use two methods to characterize this approach spanning from the simplest to the most complex. Both approaches will be implemented with Strang splitting [21] to give second-order results in time (where spatial differencing allows this) and remove (most) problems with directional bias. Our simple approach is to use first-order upwind differencing. We expect this to provide poor results because of the excessive numerical diffusion present in this method. Our complex approach will use the PPM method with discontinuity sharpening. This method provides the sharpest resolution of linear discontinuities available with capturing methods [22], capturing discontinuities within two cells.

Next, we discuss our implementation of level set methods. We use the unsplit Godunov method of Colella [23] with his low phase error limiters (fourth-order limited differences). We also implement the reinitialization scheme (a second-order extension) as described by Sussman, Smereka and Osher (SSO) [3]. We reconstruct the volume fractions in same manner as SSO reconstruct densities from ϕ, by means of a smoothed interface of width $3/2h$, where h is the mesh spacing.

Our third set of methods are the VOF approaches. We use two methods: a simple implementation of SLIC and PLIC employing a modified version of Pilliod and Puckett's least squares reconstruction. The SLIC method is first-order while the PLIC method we use is second-order. In contrast to the above methods, VOF algorithms provide interfaces that lie within one cell.

Our final method uses particles. It is primarily included for demonstration purposes. Given an analytic velocity field, particles provide accurate depictions of the exact solutions. We can also use the method in a more practical fashion for interface tracking where the velocity field in not known as an analytical function. Both use a second-order predictor-corrector method to push the particles (a predictor to find the time-centered location of the particles, followed by a full timestep integration). For the practical method, the velocities are recovered from point values via bilinear interpolation. By assigning each particle a mass, the methods can be linked simply to an underlying Eulerian calculation.

3.2 Test Problems

We will use three test problems in this paper. All three will use the same initial condition: a circle of radius 0.15 with a volume fraction of 1 centered at $(0.50, 0.75)$, the remainder of the domain has a volume fraction of 0. The domain is 1×1 and all boundaries are periodic.

The first problem is fairly standard. The velocity field is set to a constant vorticity centered at $(0.50, 0.50)$. This will cause objects to rotate about this center. Our problem is set up so that in π time units one rotation will be accomplished. Objects will not be deformed by this velocity field.

The next two problems are more challenging than the standard test bed for interface tracking algorithms.

Our second problem uses a single vortex. The velocity field is defined by the streamfunction

$$\Psi = \frac{1}{\pi} \sin(\pi x) \sin(\pi y) .$$

This velocity field will deform objects. The results for our initial conditions at $t = 3$ are shown using the analytic particle methods in Figure 1a. The object has deformed into a spiral that has wrapped about the center approximately two and a half times.

Table 1: L_1 error norms for various methods for the circular advection problem.

Method	32^2	64^2	128^2
1st Order Upwind	8.25×10^{-2}	6.38×10^{-2}	4.75×10^{-2}
PPM	3.79×10^{-3}	1.88×10^{-3}	1.14×10^{-3}
Level Sets	1.01×10^{-2}	4.14×10^{-3}	2.35×10^{-3}
Level Sets (reinit)	9.34×10^{-3}	4.14×10^{-3}	2.35×10^{-3}
SLIC	6.47×10^{-3}	3.46×10^{-3}	2.72×10^{-3}
PLIC	1.24×10^{-3}	2.61×10^{-4}	8.79×10^{-5}

Our third problem uses a more complex velocity field given by the stream function

$$\Psi = \frac{1}{4\pi} \sin\left(4\pi\left(x + \frac{1}{2}\right)\right) \cos\left(4\pi\left(y + \frac{1}{2}\right)\right).$$

The velocities are multiplied by $\cos(\pi t/T)$ to cause the flow to return to its initial conditions at $t = T$ [24]. We choose $T = 2$. Our analytic particle method gives the time evolution as shown in Figures 2a and 3a. The circle deforms and forms filaments that revolve around the other vortices in the problem away from the two vorticies that the body is initially placed near, then reverts to its inital condition.

3.3 Test Results

For the first test problem, the solutions all look quite similar and accurate. Differences can be seen through an error measure and convergence rate. These results are shown in Table 1. At every resolution PLIC is better than the other methods, and because of its second-order convergence (first-order at best for all the others) its advantage grows as the grid is refined.

Next, we give the single domain-centered vortex results. As Figure 1 shows, the PPM and PLIC methods are faithful to the solution, but level set methods begins to lose integrity at the tail of the flow. This loss can be attributed to numerical diffusion in the solver for the distance function and the concomitant loss of mass. The PLIC method produces the best results although the tail of the flow has begun to form discrete blobs rather than the filaments exhibited by the true solution. This is caused by phase error in the advection technique that manifests itself as numerical surface tension. Here, we do not use level set methods with reinitialization, because it required many iterations (that grew with time). This caused a large increase in system mass destroying any solution quality.

Finally, we show the deformation field results. Figure 2 shows, the results when the largest deformation of the initial body has taken place. Again, the level set solution has lost much of its integrity and mass (see Figure 4c). This mass loss shows itself as the flattening of the circle top and bottom in Figure 3c when the initial data should be recovered. The PPM solution is passable, but numerical diffusion in the method destroys most of the fine scale features. Finally the PLIC solution preserves the large scale features of the flow and makes a reasonable approximation to the fine scale features. The return to the initial data at $t = 2$ is not high quality, but the general shape of the body can be seen and the errors are not gross, especially when compared to other methods.

Somewhat troubling is the mass loss experienced by the level set approach even under sedate circumstances. For the rotation problems, the mass loss is given in Figure 4a. This shows that this mass loss subsides with higher mesh resolution, but is quite large on coarse meshes. For the two problems with nonconstant vorticity, the mass loss in the level set formulation becomes unbearably large. The PLIC method also losses mass, but not in a nature that endangers the ability of the method to compute an accurate solution.

4 Conclusions

As the results in the previous section indicate, the PLIC methodology is superior to other Eulerian interface tracking options. With lower error, sharp interfaces and second-order convergence, PLIC offers great returns on the computational investment. Particles offer unparalleled accuracy, but are difficult to link efficiently to Eulerian methods (shared by front-tracking methods). Level set methods provide acceptable accuracy at fine enough resolution, but their mass loss properties and inability to maintain accuracy during topology changes limit applicability.

Finally, capturing although simple is rendered quite ineffective when compared to other available technologies. With respect to cost, the clear winner is the PLIC method. The most expensive is the particle method, followed by PPM and level set methods.

References

[1] W. Mulder, S. Osher, and J. A. Sethian. Computing interface motion in compressible gas dynamics. *Journal of Computational Physics*, 100:209–228, 1992.

[2] J. Zhu and J. Sethian. Projection methods coupled to level set interface techniques. *Journal of Computational Physics*, 102:128–138, 1992.

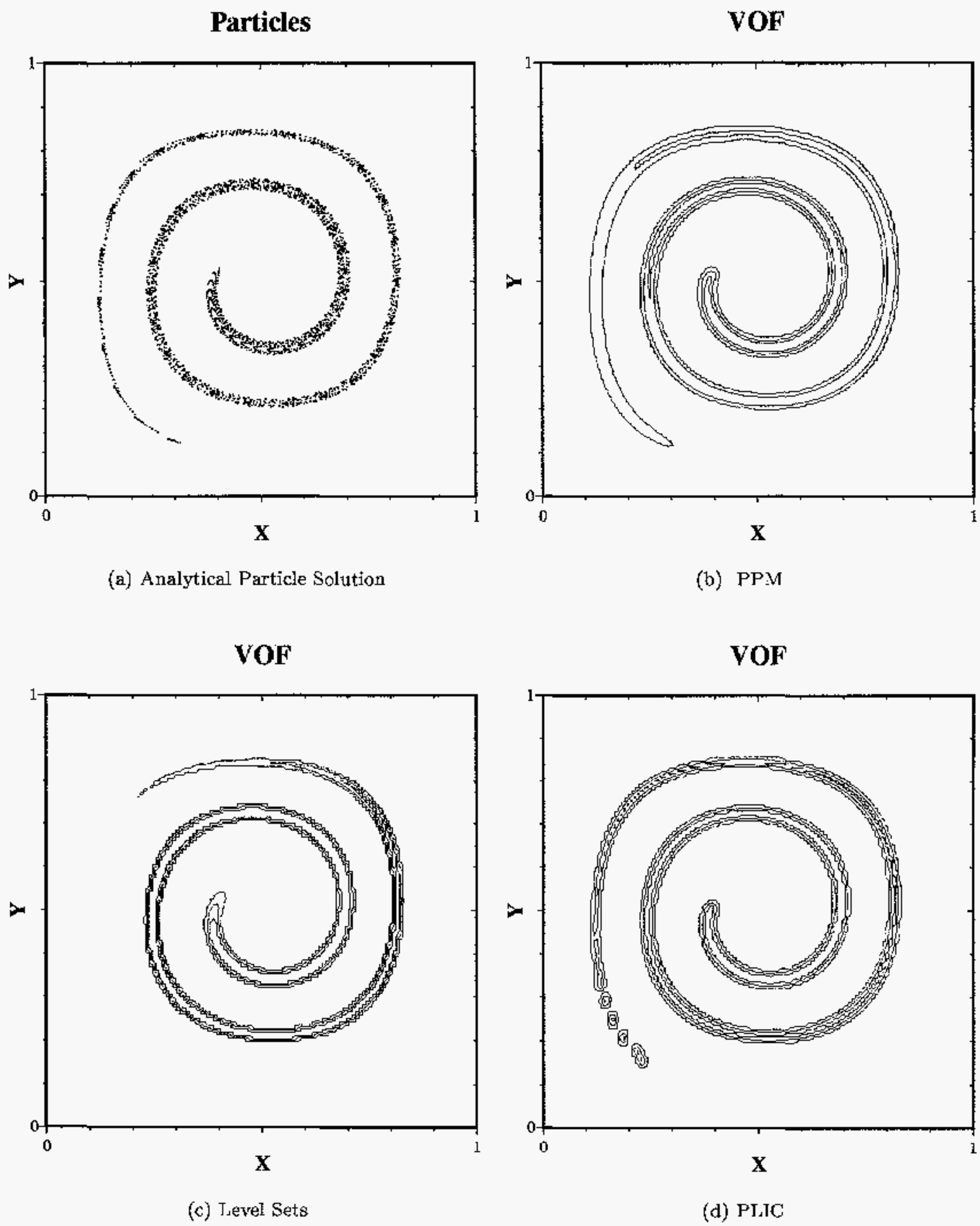
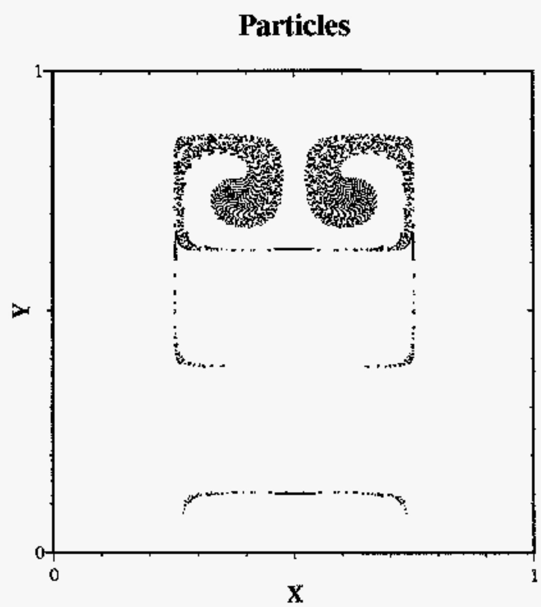
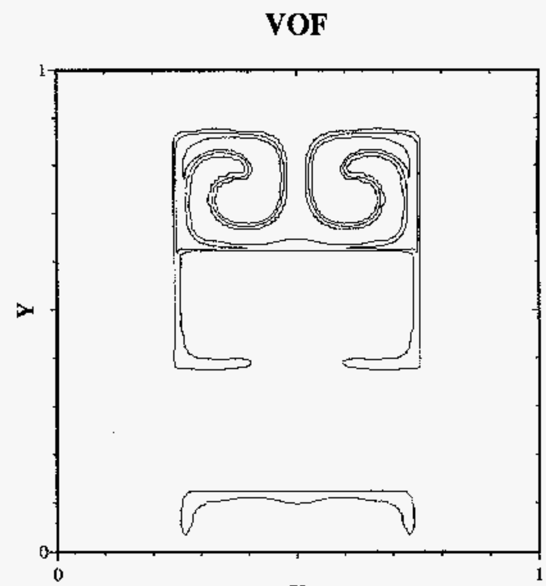


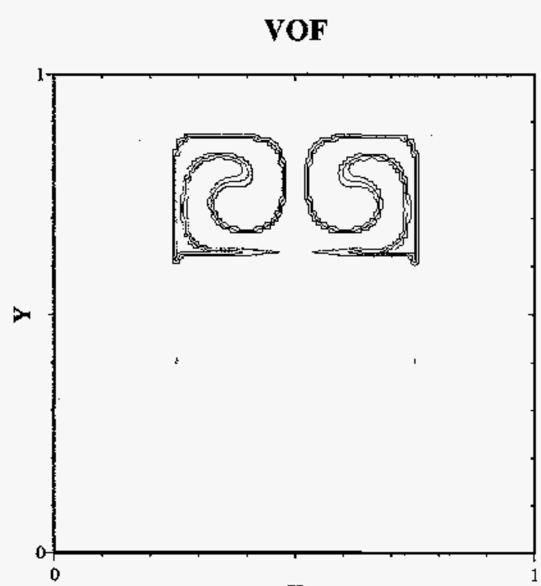
Figure 1: Results for the single vortex problem on a 128^2 grid. Contours of 0.05, 0.50, and 0.95 are plotted.



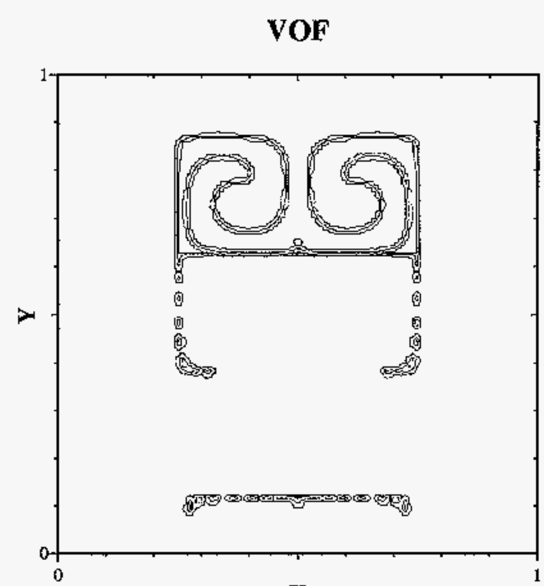
(a) Analytical Particle Solution



(b) PPM



(c) Level Sets



(d) PLIC

Figure 2: Results for the deformation field problem on a 128^2 grid at time = 1. Contours of 0.05, 0.50, and 0.95 are plotted.

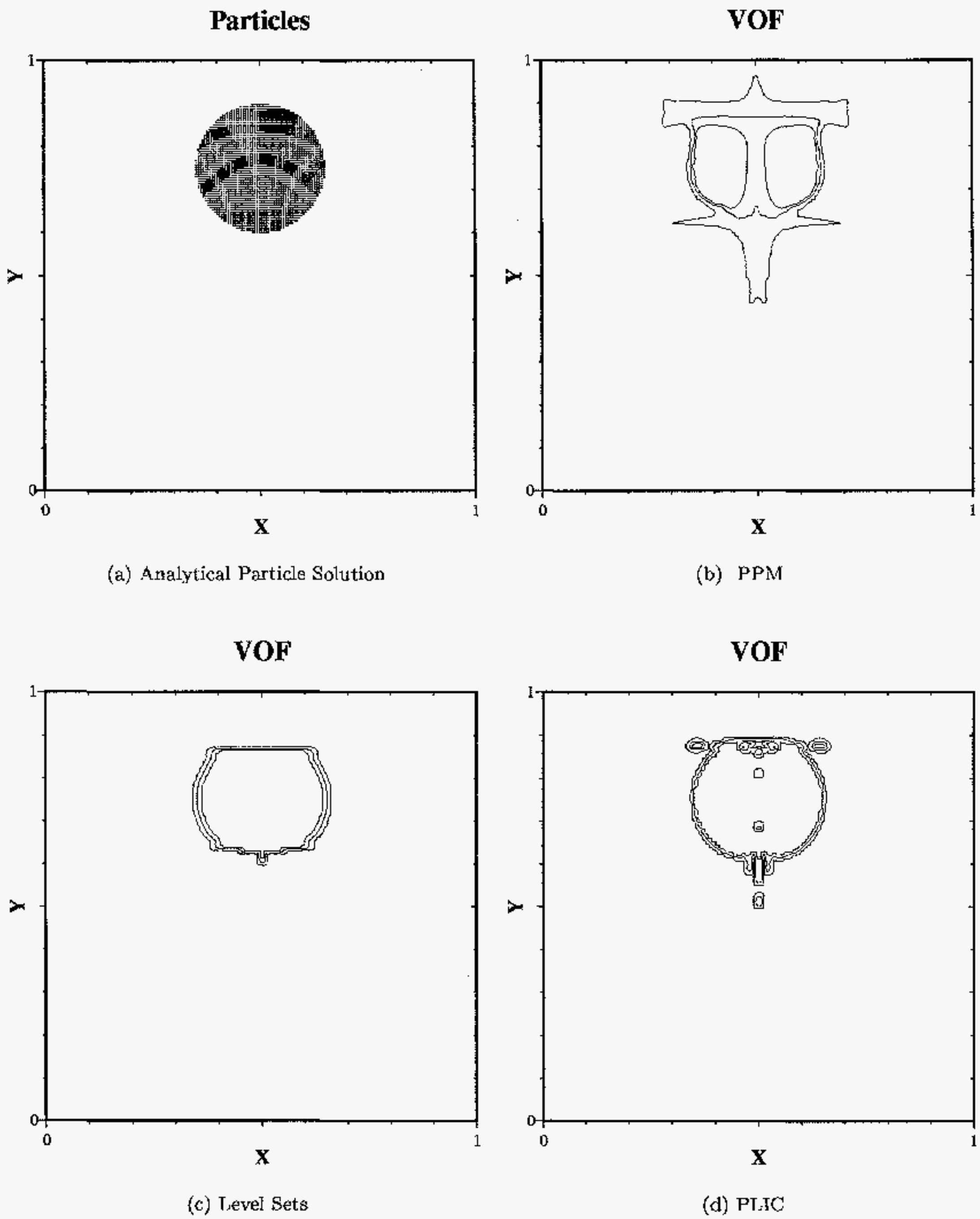
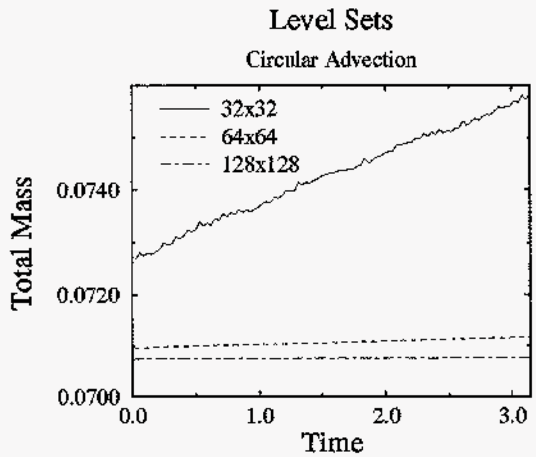
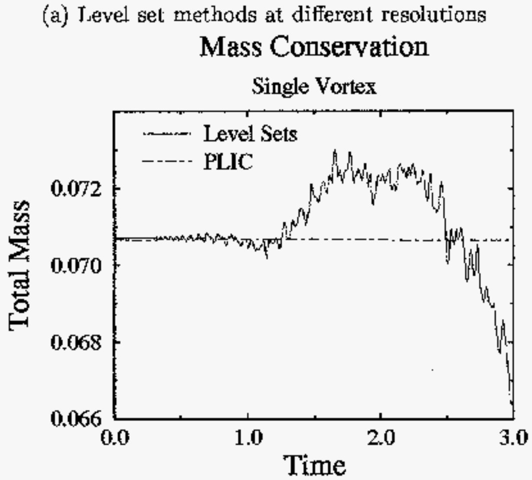


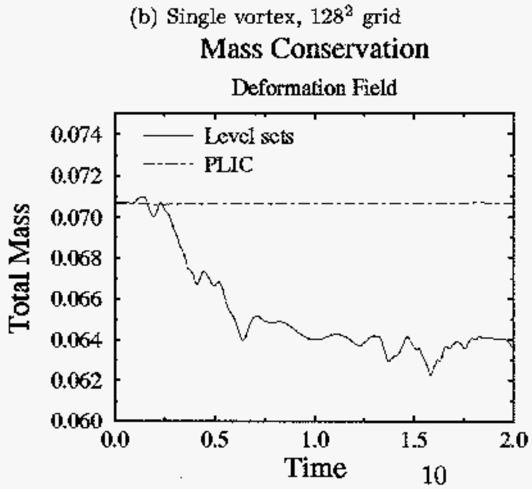
Figure 3: Results for the deformation field problem on a 128^2 grid at time = 2. Contours of 0.05, 0.50, and 0.95 are plotted.



(a) Level set methods at different resolutions



(b) Single vortex, 128^2 grid



(c) Deformation field, 128^2 grid

Figure 4: The mass conservation of level set methods and PLIC. On the circular advection problem reinitialization is used, but not on the two harder problems (see the text).

[3] M. Sussman, P. Smereka, and S. Osher. A level set approach for computing solutions to incompressible two-phase flow. *Journal of Computational Physics*, 114:146–159, 1994.

[4] J. U. Brackbill, D. B. Kothe, and H. M. Ruppel. FLIP: A low-dissipation, particle-in-cell method for fluid flow. *Computer Physics Communications*, 48:25–38, 1988.

[5] S. O. Unverdi and G. Tryggvason. A front-tracking method for viscous, incompressible, multi-fluid flows. *Journal of Computational Physics*, 100:25–37, 1992.

[6] S. O. Unverdi and G. Tryggvason. Computations of multi-fluid flows. *Physica D*, 60:70–83, 1992.

[7] J. B. Bell and D. L. Marcus. A second-order projection method variable-density flows. *Journal of Computational Physics*, 101:334–348, 1992.

[8] W. G. Szymczak, J. C. W. Rogers, J. M. Solomon, and A. E. Berger. A numerical algorithm for hydrodynamic free boundary problems. *Journal of Computational Physics*, 106:319–336, 1993.

[9] P. Colella and P. Woodward. The piecewise parabolic method (PPM) for gas-dynamical simulations. *Journal of Computational Physics*, 54:174–201, 1984.

[10] P. Woodward and P. Colella. The numerical simulation of two-dimensional fluid flow with strong shocks. *Journal of Computational Physics*, 54:115–173, 1984.

[11] R. DeBar. Fundamentals of the KRAKEN code. Technical Report UCIR-760, LLNL, 1974.

[12] K. S. Holian, S. J. Mosso, D. A. Mandell, and R. Henninger. MESA: A 3-D computer code for armor/anti-armor applications. Technical Report LA–UR–91-569, Los Alamos National Laboratory, 1991.

[13] J. S. Perry, K. G. Budge, M. K. W. Wong, and T. G. Trucano. Rhale: A 3-D MMALE code for unstructured grids. In ASME, editor, *Advanced Computational Methods for Material Modeling, AMD-Vol. 180/PVP-Vol. 268*, pages 159–174, 1993.

[14] W. F. Noh and P. R. Woodward. SLIC (simple line interface method). In A. I. van de Vooren and P. J. Zandbergen, editors, *Lecture Notes in Physics 59*, pages 330–340, 1976.

[15] C. W. Hirt and B. D. Nichols. Volume of fluid (VOF) method for the dynamics of free boundaries. *Journal of Computational Physics*, 39:201–225, 1981.

[16] D. L. Youngs. Time-dependent multi-material flow with large fluid distortion. In K. W. Morton and M. J. Baines, editors, *Numerical Methods for Fluid Dynamics*, pages 273–285, 1982.

[17] D. L. Youngs. An interface tracking method for a 3D Eulerian hydrodynamics code. Technical Report 44/92/35, AWRE, 1984.

[18] J. E. Pilliod, Jr. and E. G. Puckett. Second-Order Volume-of-Fluid Algorithms for Tracking Material Interfaces, In preperation.

[19] E. G. Puckett. A volume of fluid interface tracking algorithm with applications to computing shock wave rarefraction. In *Proceedings of the 4th International Symposium on Computational Fluid Dynamics*, pages 933–938, 1991.

[20] E. G. Puckett and J. S. Saltzman. A 3D adaptive mesh refinement algorithm for multimaterial gas dynamics. *Physica D*, 60:84–93, 1992.

[21] G. Strang. On the construction and comparison of difference schemes. *SIAM Journal on Numerical Analysis*, 5:506–517, 1968.

[22] B. Fryxell, E. Müller, and D. Arnett. Instabilities and clumping in SN 1987a. I. Early evolution in two dimensions. *Astrophysical Journal*, 367:619–634, 1991.

[23] P. Colella. Multidimensional upwind methods for hyperbolic conservation laws. *Journal of Computational Physics*, 87:171–200, 1990.

[24] R. J. LeVeque. High-Resolution Conservative Algorithms for Advection in Incompressible Flow, submitted to the SIAM Journal of Numerical Analysis.