NOTE:
A Second-Order Method for Interface Reconstruction in Orthogonal Coordinate Systems

P. Colella,*,[1] D. T. Graves *,[1] and J. A. Greenough†,[1]

E-mail: DTGraves@lbl.gov

Applied Numerical Algorithms Group, Lawrence Berkeley National Laboratory, Berkeley, CA 94720; †A-Division, Lawrence Livermore National Laboratory, Livermore, CA 94550

We present a second-order algorithm for reconstructing an interface from a distribution of volume fractions in a general orthogonal coordinate system with derivatives approximated using finite differences. The method approximates the interface curve by a piecewise-linear profile. An integral formulation is used that accounts for the orthogonal coordinate system in a natural way. We present results obtained using this method for tracking a material interface between two compressible media in spherical coordinates

Key Words: Volume of Fluid Methods, Compressible Flow, Orthogonal Coordinate Systems, Interface Reconstruction

1. INTRODUCTION

The volume-of-fluid approach provides an implicit representation of a sharp front by specifying the volume occupied on each side of the front in each grid cell. This representation was originally motivated by the specific case of the interface between two materials, in which the volumes contain parcels of two different fluids, each with its own mass, energy and equation of state. The volume-of-fluid approach to tracking interfaces has been used by many including [5], [3] and [4]. In a more general front tracking setting, the volume-of-fluid approach gives a convenient representation, particularly in settings where discrete conservation is important. In that case, the volumes on either side of the front form the basis for performing a conservative, finite-volume discretization of conservation laws.

[1]This research was supported at the Lawrence Berkeley National Laboratory by the U.S. Department of Energy: Director, Office of Science, Office of Advanced Scientific Computing, Mathematical, Information, and Computing Sciences Division under Contract DE-AC03-76SF00098.

Work performed under the auspices of the U.S. Department of Energy by the Lawrence Livermore National Laboratory under contract No. W-7405-ENG-48.

In this note, we present a method for computing a volume-of-fluid representation of a front suitable for use in any orthogonal coordinate system. It is based on the relationship (first noted in [2]) between a volume-of-fluid description of a front and the local representation of that front as the graph of a function.

In the case of Cartesian coordinates, the method in [2] provides a second-order accurate reconstruction of the surface geometry from the volume fractions and is exact for flat surfaces. There are a number of algorithms that meet these requirements for Cartesian coordinates in two and three dimensions. See, for example [1] and [4]. However the extension of these algorithms to other orthogonal coordinate systems has been unwieldy. The principal contribution of the algorithm described in this note is that it is second-order accurate and straightforward to implement for the most commonly used orthogonal coordinate systems.

2. VOLUME OF FLUIDS OVERVIEW

We are given a velocity field $\vec{s}$ which extends smoothly to the entire domain and the location of the interface $\vec{x}$. The motion of the interface is given by

$$\frac{d\vec{x}}{dt} = (\vec{s} \cdot \hat{n})\hat{n} \tag{1}$$

where $\hat{n}$ is the interface normal. We can describe the interface as a piecewise-constant function f^α which is zero outside fluid α and unity inside. If we do so, equation 1 is formally equivalent to

$$\frac{\partial f^\alpha}{\partial t} + \vec{s} \cdot \nabla f^\alpha = 0. \tag{2}$$

By definition, f^α is the volume fraction of fluid α. We use the conservative form of equation 2:

$$\frac{\partial f^\alpha}{\partial t} + \nabla \cdot (f^\alpha \vec{s}) = f^\alpha \nabla \cdot \vec{s}. \tag{3}$$

We now describe the Cartesian algorithm and the algorithm for general orthogonal coordinates.

2.1. Cartesian Algorithm Specification

Define our Cartesian coordinates to be (x, y) We discretize the coordinate space with equally-spaced grid points with separation $\Delta x = \Delta y = h$. A cell in real space is $\Delta_{i,j} = [ih, (i+1)h] \times [jh, (j+1)h]$. In two-dimensional Cartesian coordiantes, equation 3 becomes

$$\frac{\partial f^\alpha}{\partial t} + \frac{\partial (s^x f^\alpha)}{\partial x} + \frac{\partial (s^y f^\alpha)}{\partial y} = f^\alpha \frac{\partial s^x}{\partial x} + f^\alpha \frac{\partial s^y}{\partial y}. \tag{4}$$

The volume-of-fluid method can be divided into three steps which are repeated for each dimension in an operator-split fashion:

- We use the volume fractions of each fluid to reconstruct as a local graph the connectivity and geometric configuration of each fluid.

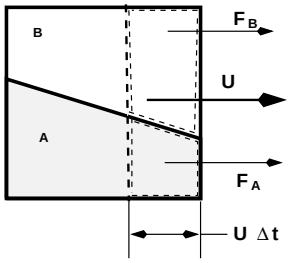

FIG. 1. Illustration of advection of interface. The volume fluxed out of the cell during a time step Δt is shown by the dashed rectangle. The quadrilateral volume of fluid 1 and fluid 2 with this rectangle are the volume fluxes ($F_{V\alpha}$) of fluid 1 and fluid 2 during the time step.

- We use this local graph to geometrically construct the volume flux of each fluid.
- We update the volume fractions with an operator- split discretization of equation 4.

We locally reconstruct the surface as a graph of a function g of the form $(x, g(x))$ or $(g(y), y)$. See Miller and Puckett [4] for details of how this choice is made. If we set $x^2_{Low} = (j - s)\Delta x^1$ for some integer s which is sufficiently large, then the column height C^α of the fluid is given by

$$C^\alpha_{i,j} = \sum_{j'=j-s}^{j} f^\alpha_{i,j'} h^2 \tag{5}$$

The function g is a linear function whose slope m is computed using finite differences of the column heights of the fluid

$$m = \frac{C^\alpha_{i+1,j} - C^\alpha_{i-1,j}}{2h} \tag{6}$$

For each sweep, we geometrically construct the volume flux and then update the cell. Consider figure 1 and suppose that we are doing a sweep in the x direction. The distance that a particle travels to get to the face of the cell is $d_x = s^x k$ where k is the time step. Define $F_{\alpha,i+\frac{1}{2}}$ to be the volume of fluid α between the the $x = h(i + \frac{1}{2}) - s^x k$ line and the $x = h(i + \frac{1}{2})$ line. This is the flux through the face of fluid α. The discrete divergence of F_α is a consitent discretization of $\nabla \cdot (\vec{s} f^\alpha)$. The update of the volume fraction for the x sweep takes the form

$$\frac{1}{k}\left(f^{\alpha,n+1}_{i,j} - f^{\alpha,n}_{i,j}\right) = -\frac{1}{h}\left(F^{n+\frac{1}{2}}_{\alpha,i+1/2,j} - F^{n+\frac{1}{2}}_{\alpha,i-1/2,j}\right) + f^{\alpha,n}\left(\frac{s^x_{i+\frac{1}{2}} - s^x_{i-\frac{1}{2}}}{h}\right) \tag{7}$$

When the $\nabla \cdot \vec{s}$ in the right-hand side of equation 3 is the divergence of the velocity field, often the right-hand side is not directionally operator-split, but included fractionally in each update.

3. RESULTS AND CONCLUSIONS

The present reconstruction algorithm is used in the context of multi-fluid gas dynamics as done in Miller and Puckett [4] and Colella, Glaz and Ferguson [1].

These are higher-order Godunov methods in which the fluid velocity is single- valued in multi-fluid cells. Appropriate single-fluid average states are computed in multi-fluid cells. The state is extrapolated from cell centers to edges and there a Riemann problem is solved using a single-fluid algorithm. The interface is then advected using this edge-centered Godunov velocity. The only substantial way in which the current algorithm differs from Miller and Puckett is the way in which the construction is done.

We present results of the reconstruction method for a hydrodynamics problem in spherical geometry. The problem is that of a dense gas spherical shell embedded in a light gas background. The outer edge of the dense shell is pressurized (heated) so that a strong shock is driven into the shell towards the origin thereby forcing an implosion. The outer edge of the shell is driven outward where it eventually moves out of the computational domain. We run the calculation using spherical $(r - \phi)$ coordinates. Figure (2) shows the evolution of the volume fraction in $r - \phi$ coordinates. As the solution proceeds, the grid resolution remains adequate and the interface remains smooth.

We have presented a volume-of-fluid reconstruction algorithm for representing sharp fronts in orthogonal coordinates in two dimensions. The ideas here can be routinely extended to three dimensional orthogonal coordinates. An intriguing possibility is that of extending this approach to general mapped grids.

4. ACKNOWLEDGEMENTS

The authors would like to thank John Bell for providing key obervations and guidance during the formulation of this algorithm. We would also like to thank Ann Almgren for producing the version of the graphics tool that allowed visualization of the spherical data.

REFERENCES

1. P. Colella, H. M. Glaz, and R. E. Ferguson. Mutifluid algorithms for Eulerian finite difference methods. unpublished.
2. J. Helmsen, P. Colella, and E. G. Puckett. Non-convex profile evolution in two dimensions using volume of fluids. Technical Report LBNL-40693, Lawrence Berkeley National Lab, June 1997.
3. C. W. Hirt and B. D. Nichols. Volume of fluid (vof) method for the dynamics of free boundaries. *J. Comput. Phys.*, 39, 1981.
4. G. H. Miller and E. G. Puckett. A high-order Godunov method for multiple condensed phases. *J. Comput. Phys.*, 128, 1996.
5. W. F. Noh and P. Woodward. Simple line interface calculation. In *Fifth International Conference on Numerical Methods in Fluid Dynamics*, 1976.

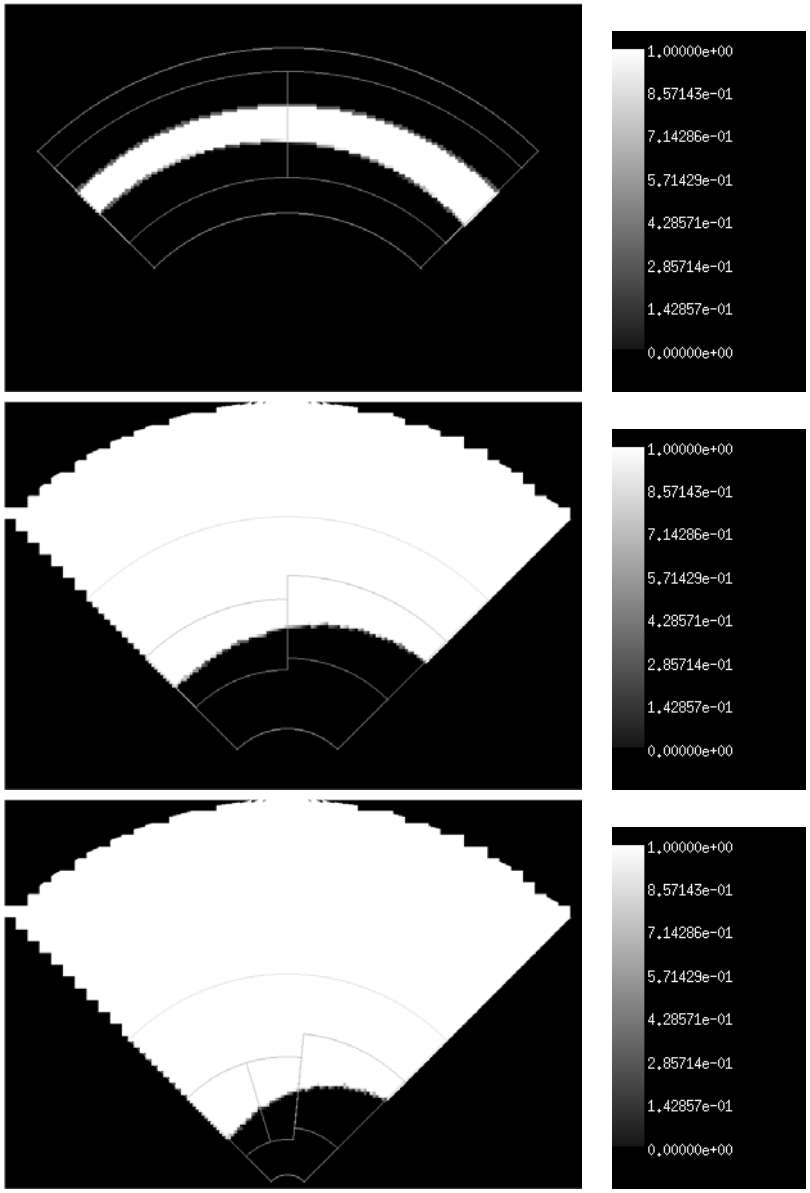

FIG. 2. The evolution of the volume fraction in $r - \phi$ coordinates.