

# A Fast Algorithm for Moving Interface Problems

S. Dutta<sup>1</sup>, J. Glimm<sup>1,2</sup>, J. W. Grove<sup>3</sup>, D. H. Sharp<sup>4</sup>, and Y. Zhang<sup>1</sup>

<sup>1</sup> Department of Applied Mathematics and Statistics,  
University at Stony Brook, Stony Brook, NY 11794-3600,  
yzhang@ams.sunysb.edu, glimm@ams.sunysb.edu, kuttush@ams.sunysb.edu

<sup>2</sup> Center for Data Intensive Computing,  
Brookhaven National Laboratory, Upton NY 11793-6000

<sup>3</sup> Methods for Advanced Scientific Simulations Group,  
Computer and Computational Science Division,  
Los Alamos National Laboratory, Los Alamos, NM 87545,  
jgrove@lanl.gov

<sup>4</sup> Complex Systems Group, Theoretical Division,  
Los Alamos National Laboratory, Los Alamos, NM 87545,  
dhs@t13.lanl.gov

**Abstract.** Numerical simulations of a spherical shock refraction have been successfully conducted by a front tracking method. We demonstrate the efficiency of the front tracking algorithm by comparing the  $L_1$ -error of spherical simulations by tracked and untracked methods. We find that the tracked algorithm is about 64 (256) times faster than the corresponding method without tracking the interface for a 2d (3d) simulation.

## 1 Introduction

The numerical method we use for computing axisymmetric flow in the paper is a front tracking method. Front tracking is an adaptive computational method in which a lower dimensional moving grid is fit to and follows distinguished waves in a flow. Tracked waves explicitly include jumps in the flow state across the waves and keep discontinuities sharp. A key feature is the avoidance of finite differencing across discontinuity fronts and thus the elimination of interfacial numerical diffusion including mass and vorticity diffusion. In addition, nonlinear instability and post-shock oscillations are reduced at the tracked fronts. Front tracking as implemented in the code *FronTier* includes the ability to handle multidimensional wave interactions in both two [7, 13, 18] and three [6, 5] space dimensions and is based on a composite algorithm that combines shock capturing on a spatial grid with a specialized treatment of the flow near the tracked fronts. Applications have included the Rayleigh-Taylor (RT) [3, 12, 11] and Richtmyer-Meshkov (RM) [8, 17, 19, 20, 10] instabilities. The Rayleigh-Taylor instability occurs when a fluid interface is accelerated in a direction opposite to the density gradient across the interface, while Richtmyer-Meshkov instability is induced by the refraction of shock waves through a fluid interface. *FronTier* is implemented for distributed memory parallel computers and some of the fundamental algorithms used in this code are described in [1, 14, 15, 4, 9, 16].

In this paper, we give a brief summary of the front tracking algorithm for axisymmetric flow. We report simulations of spherical shock refraction by applying this algorithm. Spherical shock refraction arises in the evolution of supernova and in inertial confinement fusion, and thus of fundamental importance to science and technology. We show that the front tracking is a fast algorithm through an error comparison study for both tracked and untracked spherical simulations.

## 2 Formulation

The three dimensional Euler equations in rectangular coordinates  $(x, y, z)$  can be written as

$$\rho_t + \nabla \cdot (\rho \mathbf{v}) = 0, \quad (1)$$

$$(\rho \mathbf{v})_t + \nabla \cdot (\rho \mathbf{v} \otimes \mathbf{v}) + \nabla p = \rho \mathbf{g}, \quad (2)$$

$$(\rho E)_t + \nabla \cdot (\rho E \mathbf{v} + p \mathbf{v}) = \rho \mathbf{v} \cdot \mathbf{g}, \quad (3)$$

where  $\rho$  is the mass density of the fluid,  $\mathbf{v}$  is the fluid velocity,  $E = e + \frac{1}{2} \mathbf{v} \cdot \mathbf{v}$  is the total specific energy with the specific internal energy  $e$ ,  $p$  is the pressure and  $\mathbf{g}$  is the body force, which we will take as pointing in the  $\mathbf{e}_3$  directing, i.e. vertically upwards. The equations (1), (2) (3) describe the conservation laws of mass, momentum, and total energy for a non-reacting compressible fluid.

We introduce cylindrical coordinates  $(r, \theta, z)$  by the transformation:

$$\begin{aligned} x &= r \cos \theta, \\ y &= r \sin \theta, \\ z &= z. \end{aligned}$$

Let  $\mathbf{e}_1 = (1, 0, 0)$ ,  $\mathbf{e}_2 = (0, 1, 0)$ ,  $\mathbf{e}_3 = (0, 0, 1)$  be the unit vector basis for the rectangular coordinate system. Let  $(\mathbf{r}, \boldsymbol{\theta}, z)$  be the unit vector basis for the rotational coordinate system defined by

$$\begin{aligned} \mathbf{r} &= \mathbf{e}_1 \cos \theta + \mathbf{e}_2 \sin \theta, \\ \boldsymbol{\theta} &= -\mathbf{e}_1 \sin \theta + \mathbf{e}_2 \cos \theta, \\ z &= \mathbf{e}_3. \end{aligned}$$

Let  $\mathbf{v} = v_0 \mathbf{r} + v_1 z + v_\theta \boldsymbol{\theta}$  and  $\mathbf{g} = g_0 \mathbf{r} + g_1 z + g_\theta \boldsymbol{\theta}$ . Assuming rotational symmetry of the solution  $\rho, \rho \mathbf{v}, \rho E$  of (1)-(3),  $v_\theta = 0, g_\theta = 0$  and the system is independent of  $\theta$ . Under rotational symmetry and using this rotational coordinate system the equations (1), (2), (3) can be transformed to:

$$\rho_t + (\rho v_0)_r + (\rho v_1)_z = -\frac{1}{r} \rho v_0, \quad (4)$$

$$(\rho v_0)_t + (\rho v_0^2)_r + (\rho v_0 v_1)_z + p_r = -\frac{1}{r} \rho v_0^2 + \rho g_0, \quad (5)$$

$$(\rho v_1)_t + (\rho v_1 v_0)_r + (\rho v_1^2)_z + p_z = -\frac{1}{r} \rho v_1 v_0 + \rho g_1, \quad (6)$$

$$\begin{aligned}
(\rho E)_t + (\rho E v_0)_r + (\rho E v_1)_z + (p v_0)_r + (p v_1)_z = & -\frac{1}{r} \rho E v_0 - \frac{1}{r} p v_0 \\
& + \rho (g_0 v_0 + g_1 v_1). \quad (7)
\end{aligned}$$

The system is closed via a thermodynamic equation of state relates density, pressure, and energy, most commonly through a functional relation  $p = p(\rho, e)$ .

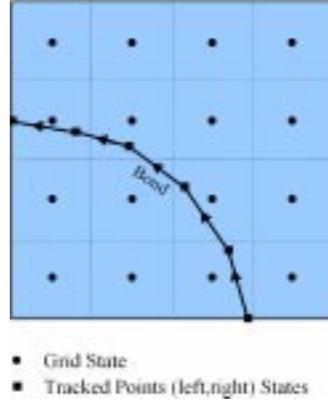
### 3 Axisymmetric Front Tracking Algorithm

In this section, we present a brief summary of the front tracking algorithm for axisymmetric flow. For a detailed description of the front tracking algorithm, we refer to Glimm, Grove and Zhang [9]. The validation was carried out by comparison with experiment, see Drake *et al.* [2].

Front tracking is a numerical method in which selected waves are explicitly represented in the discrete form of the solution. Examples include shock waves, contact discontinuities, and material interfaces. Other waves, such as leading and trailing edges of rarefaction waves, have continuous states but jumps in their first derivatives. Tracked waves are propagated using the appropriate equations of motion for the given model. For example, if the system of equations consists of a set of hyperbolic conservation laws,  $\mathbf{u}_t + \nabla \cdot \mathbf{f} = \mathbf{h}$ , then the instantaneous velocity  $s$  of a discontinuity surface satisfies the Rankine-Hugoniot equations,  $s[\mathbf{u}] = [\mathbf{f}] \cdot \mathbf{n}$ . Here  $\mathbf{n}$  is the unit normal to the discontinuity surface. During a time step propagation, the type of a wave, and the flow field in a neighborhood of the wave determine a local time integrated velocity for each point on the wave in the direction normal to the wave front. Wave propagation consists of moving each point a distance  $s\Delta t$  in the normal direction as well as computing the time updated states at the new position. Tracking preserves the mathematical structure of the discontinuous waves by maintaining the discrete jump at the wave front, thus eliminating numerical diffusion. It also allows for the direct inclusion of the appropriate flow equations for the wave front in the numerical solution.

The discrete representation of the flow is based on a composite grid that consists of a spatial grid representing the flow field in the bulk fluid, together with a co-dimension one grid that represents the fronts. Fig. 1 shows a two dimensional schematic of a time step snapshot of such a grid. The front is represented by a piecewise linear curve, the sections of which are called bonds. In contrast to the spatial grid, which is fixed in time (*i.e.*, Eulerian), the fronts move according to the dynamics of the wave fronts that they represent. A single time step is divided into two processes, the propagation of the fronts, and the updating of the solution on the spatial grid.

Stored with each point are two states to represent the discontinuity across the wave. An orientation is given along the curve so that we may speak of the left and right states at a point. Therefore, propagation of the front can be defined as updating the position of each point on the front and updating the corresponding left and right states associated to it at a new time  $t + \Delta t$ . Operator splitting, in the rotated coordinate system  $(\mathbf{T}, \mathbf{N})$ , allows separate propagation steps in



**Fig. 1.** A representation of the grid for a front tracking computation. The solution is represented on the union of a spatial finite difference grid and a dynamic grid that follows the fronts.

directions normal to and tangent to the front, where  $\mathbf{N} = N_0\mathbf{r} + N_1z$ ,  $\mathbf{T} = T_0\mathbf{r} + T_1z$  are the normal and tangential unit vectors at some point on the front. Then the velocity  $\mathbf{v}$  can be rewritten as  $\mathbf{v} = v_N\mathbf{N} + v_T\mathbf{T}$ ,  $v_0 = v_N N_0 + v_T T_0$  and  $\mathbf{g} = g_N\mathbf{N} + g_T\mathbf{T}$ . The equations in the normal propagation step are formed by the projection of the equations (4)-(7) onto the normal direction as the following

$$\rho_t + \frac{\partial}{\partial N}(\rho v_N) = -\frac{1}{r}\rho v_N N_0, \quad (8)$$

$$(\rho v_N)_t + \frac{\partial}{\partial N}(\rho v_N^2 + p) = -\frac{1}{r}\rho v_N^2 N_0 + \rho g_N, \quad (9)$$

$$(\rho v_T)_t + \frac{\partial}{\partial N}(\rho v_N v_T) = -\frac{1}{r}\rho v_T v_N N_0, \quad (10)$$

$$(\rho E)_t + \frac{\partial}{\partial N}(\rho E v_N) + \frac{\partial}{\partial N}(p v_N) = -\frac{1}{r}\rho E v_N N_0 - \frac{1}{r}p v_N N_0 + \rho g_N v_N. \quad (11)$$

The tangential propagation is performed on each side of the front followed by the normal propagation. Since the solution is smooth during the tangential propagation, a convenient finite difference scheme, such as the second order MUSCL scheme, can be used to update the states at each point on the front by solving the follow equations which are formed by the projection of the equations (4)-(7) onto the tangential direction

$$\rho_t + \frac{\partial}{\partial T}(\rho v_T) = -\frac{1}{r}\rho v_T T_0, \quad (12)$$

$$(\rho v_T)_t + \frac{\partial}{\partial T}(\rho v_T^2 + p) = -\frac{1}{r}\rho v_T^2 T_0 + \rho g_T, \quad (13)$$

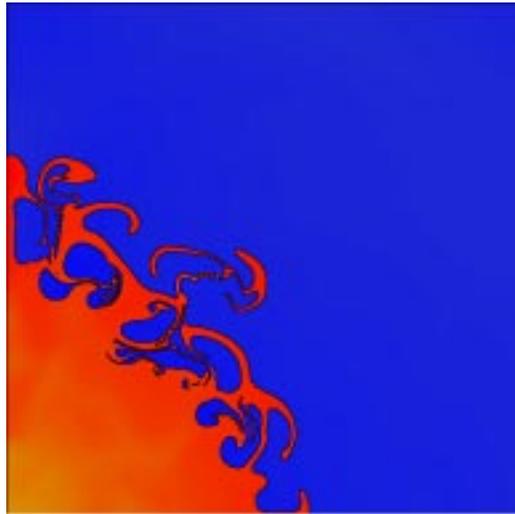
$$(\rho v_N)_t + \frac{\partial}{\partial T}(\rho v_T v_N) = -\frac{1}{r}\rho v_N v_T T_0, \quad (14)$$

$$(\rho E)_t + \frac{\partial}{\partial T}(\rho E v_T) + \frac{\partial}{\partial T}(p v_T) = -\frac{1}{r}\rho E v_T T_0 - \frac{1}{r}p v_T T_0 + \rho g_T v_T. \quad (15)$$

Notice that tangential propagation of points on the front is equivalent to remeshing of the front, in the limit  $\Delta t \rightarrow 0$ , so it is not essential to move these points during the tangential update. Finally we update the states in the interior smooth region by solving the system of equations (4)-(7) using the MUSCL algorithm with the front data as a boundary condition. Thus we never perform finite differencing across the front and keep all discontinuities perfectly sharp.

## 4 Spherical Shock Refraction

In this section, we report the simulation of a spherical shock refraction problem by applying the algorithm described in the previous section. An application of axisymmetric front tracking algorithm in chaotic fluid mixing is shown in Fig. 2, that presents a cross-sectional view of the mixing layer generated by RM instability in a randomly perturbed axisymmetric  $SF_6$  sphere driven by an imploding shock wave from the air outside the sphere. See Glimm *et al.* [10], where a series of numerical validation issues are addressed.



**Fig. 2.** *Cross-sectional view of the growth of instability in a randomly perturbed axisymmetric  $SF_6$  sphere driven by an imploding shock wave in the air.*

Now we show that our front tracking is a fast algorithm by comparing the  $L_1$ -error in the tracked simulations to the error of untracked method. In the untracked computation we apply the second MUSCL scheme to the entire domain.

Our comparison study is carried out for an unperturbed interface since this case admits an easily understood exact solution which can be obtained by solving a one dimensional spherical problem on a very fine mesh. The  $(r, z)$  computational domain is  $[0, r_1] \times [0, z_1]$ , with  $r_1 = z_1 > 0$ . The origin is denoted by  $P_0 = (0, 0)$ . Let  $R$  denote the distance from any point in the computational domain to  $P_0$ . The contact surface is located at the circle  $R = R_0$ . In our experiment, we place the incident shock wave in the outer light fluid at a sphere  $R = R_1 > R_0$ , moving toward to the origin. Due to the rotational symmetry, we are considering a spherical imploding problem. The initial configuration of the system contains three regions: the region behind the incident shock, the region between the incident shock and the fluid interface, and the region enclosed within the interface. The states ahead of the shock are initialized by the prescription of the densities inside and outside of the contact surface, the pressure and the velocities of two fluids. The state behind the shock is determined by a prescription of the the Mach number of the shock. A reflecting boundary condition is used at the left side, i.e. the  $r = 0$  axis. Flow-through boundary conditions are applied at the circular boundary  $R = R_{max}$  outside of which is a constant flow region where no computation is needed. The physical parameters for our simulations are: the density ratio 20:1, the shock Mach number  $M = 10$ . We use a gamma gas law with  $\gamma = 1.66$  for our calculations. The density plots for spherical shock refraction is shown in Fig. 3 for both tracked and untracked simulations from which we see that mass diffusion is evident in the untracked density plots.

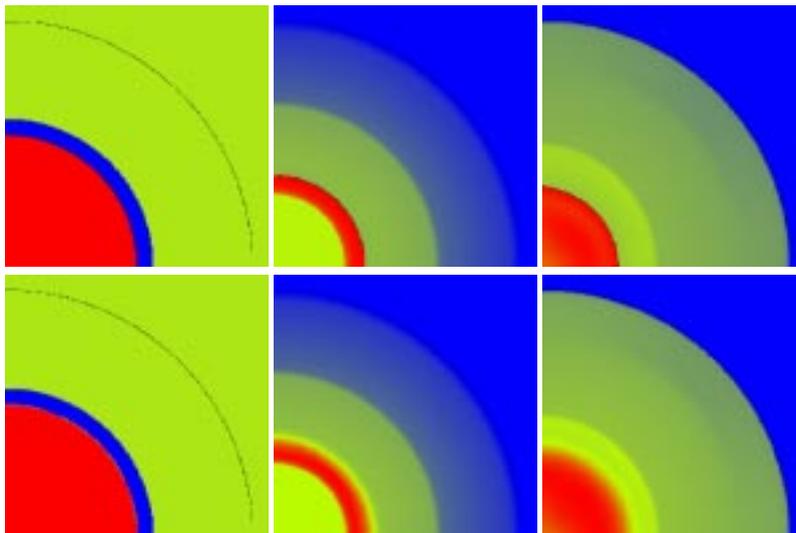
In order to compute the errors, we denote  $\rho_h(r, z, t)$  the discrete density from a 2D simulation. Let  $\rho(r, z, t)$  be the exact density which is obtained by mapping the fine grid 1D spherical density to the 2D domain. We define the total time integrated  $L_1$ -error on the computational domain  $\Omega = \{(r, z) : \sqrt{r^2 + z^2} \leq R_{max}\}$  as following:

$$\|\rho_h - \rho\|_{L_1(L_1)} = \int_0^T \int_{\Omega} |\rho_h(r, z, t) - \rho(r, z, t)| r dr dz dt.$$

**Table 1.** Comparison of the total  $L_1$  error for tracked and untracked simulations for the  $200 \times 200$ ,  $400 \times 400$ ,  $800 \times 800$  grid levels.

Method	$200 \times 200$	$400 \times 400$	$800 \times 800$
Tracked	$2.3 \times 10^{-5}$	$1.4 \times 10^{-5}$	$0.8 \times 10^{-5}$
Untracked	$4.2 \times 10^{-5}$	$3.1 \times 10^{-5}$	$2.2 \times 10^{-5}$

We conduct the simulations for the grid size of  $200 \times 200$ ,  $400 \times 400$ ,  $800 \times 800$  using tracked and untracked methods. The errors for various cases are listed in Table 1, from which we find that the tracked error for a  $200 \times 200$  grid is comparable to the untracked error for a  $800 \times 800$  grid. Therefore, tracking can reduce the required mesh refinement by a factor of 4 in each space time



**Fig. 3.** Density plots for a spherical implosion simulation with an unperturbed interface. The three images on the top show the evolution of a tracked contact at  $t = 0$ , after the transmission of the shock and after the reshock respectively. The three images on the bottom show the evolution of the untracked contact at the same three times. The grid size is  $200 \times 200$ .

**Table 2.** Comparison of the CPU time (minutes) for tracked and untracked simulations for the  $200 \times 200$ ,  $400 \times 400$ ,  $800 \times 800$  grid levels.

Method	$200 \times 200$	$400 \times 400$	$800 \times 800$
Tracked	28	230	1747
Untracked	25	202	1687

dimension. The corresponding CPU times are listed in Table 2. We observe that for the same grid level the tracked method only spends 3%–15% more time than the untracked one. We also notice that the CPU time of the untracked simulation for  $800 \times 800$  grid is about 60 times of the tracked run for  $200 \times 200$  grid. (Note 64 times is a theoretical prediction). Therefore the tracked algorithm is about 64 times (256 times) faster than the corresponding method without tracking the interface for 2d (3d) simulations.

## 5 Conclusions

We have presented a spherical shock refraction simulation by the front tracking method. We have shown that the tracking algorithm can reduce the level of

mesh refinement by a factor of four compared to the required mesh refinement in the untracked method for a given error tolerance. Therefore the front tracking method can reduce the computational time enormously (a factor of 256 in space time zones for 3D simulations).

## References

1. I-L. Chern, J. Glimm, O. McBryan, B. Plohr, and S. Yaniv. Front tracking for gas dynamics. *J. Comp. Phys.*, 62:83–110, 1986.
2. R. P. Drake, H. F. Robey, O. A. Hurricane, B. A. Remington, J. Knauer, J. Glimm, Y. Zhang, D. Arnett, D. D. Ryutov, J. O. Kane, K. S. Budil, and J. W. Grove. Experiments to produce a hydrodynamically unstable spherically diverging system of relevance to instabilities in supernovae. *Astrophysical Journal*, 564:896–906, 2002.
3. C. L. Gardner, J. Glimm, O. McBryan, R. Menikoff, D. Sharp, and Q. Zhang. The dynamics of bubble growth for Rayleigh-Taylor unstable interfaces. *Phys. Fluids*, 31:447–465, 1988.
4. J. Glimm, J. W. Grove, X.-L. Li, K.-M. Shyue, Q. Zhang, and Y. Zeng. Three dimensional front tracking. *SIAM J. Sci. Comp.*, 19:703–727, 1998.
5. J. Glimm, J. W. Grove, X.-L. Li, and D. C. Tan. Robust computational algorithms for dynamic interface tracking in three dimensions. *SIAM J. Sci. Comp.*, 21:2240–2256, 2000.
6. J. Glimm, J. W. Grove, X.-L. Li, and N. Zhao. Simple front tracking. In G.-Q. Chen and E. DiBenedetto, editors, *Contemporary Mathematics*, volume 238, pages 133–149. Amer. Math. Soc., Providence, RI, 1999.
7. J. Glimm, J. W. Grove, W. B. Lindquist, O. McBryan, and G. Tryggvason. The bifurcation of tracked scalar waves. *SIAM Journal on Computing*, 9:61–79, 1988.
8. J. Glimm, J. W. Grove, and Y. Zhang. Three dimensional axisymmetric simulations of fluid instabilities in curved geometry. In M. Rahman and C. A. Brebbia, editors, *Advances in Fluid Mechanics III*, pages 643–652. WIT Press, Southampton, UK, 2000.
9. J. Glimm, J. W. Grove, and Y. Zhang. Interface tracking for axisymmetric flows. *SIAM J. SciComp*, 24:208–236, 2002. LANL report No. LA-UR-01-448.
10. J. Glimm, J. W. Grove, Y. Zhang, and S. Dutta. Numerical study of axisymmetric richtmyer-meshkov instability and azimuthal effect on spherical mixing. *J. Stat. Physics*, 107:241–260, 2002.
11. J. Glimm, X.-L. Li, R. Menikoff, D. H. Sharp, and Q. Zhang. A numerical study of bubble interactions in Rayleigh-Taylor instability for compressible fluids. *Phys. Fluids A*, 2(11):2046–2054, 1990.
12. J. Glimm and X.-Li. On the validation of the Sharp-Wheeler bubble merger model from experimental and computational data. *Phys. Fluids*, 31:2077–2085, 1988.
13. J. W. Grove. Anomalous waves in shock wave – fluid interface collisions. In B. Lindquist, editor, *Current Progress in Hyperbolic Systems: Riemann Problems and Computations*, volume 100 of *Contemporary Mathematics*, pages 77–90. American Mathematics Society, Providence, RI, 1989.
14. J. W. Grove. The interaction of shock waves with fluid interfaces. *Adv. Appl. Math.*, 10:201–227, 1989.
15. J. W. Grove. Applications of front tracking to the simulation of shock refractions and unstable mixing. *J. Appl. Num. Math.*, 14:213–237, 1994.

16. J. W. Grove. FronTier: A compressible hydrodynamics front tracking code; A short course in front tracking. LANL Report LA-UR 99-3985, Los Alamos National Laboratory, 1999.
17. J. W. Grove, R. Holmes, D. H. Sharp, Y. Yang, and Q. Zhang. Quantitative theory of Richtmyer-Meshkov instability. *Phys. Rev. Lett.*, 71(21):3473–3476, 1993.
18. J. W. Grove and R. Menikoff. The anomalous reflection of a shock wave at a material interface. *J. Fluid Mech.*, 219:313–336, 1990.
19. R. L. Holmes, J. W. Grove, and D. H. Sharp. Numerical investigation of Richtmyer-Meshkov instability using front tracking. *J. Fluid Mech.*, 301:51–64, 1995.
20. Q. Zhang and M. J. Graham. A numerical study of Richtmyer-Meshkov instability driven by cylindrical shocks. *Phys. Fluids*, 10:974–992, 1998.