LAWRENCE
LIVERMORE
NATIONAL
LABORATORY

UCRL-CONF-227042

Code Verification Results of an LLNL ASC Code on Some Tri-Lab Verification Test Suite Problems

S. R. Anderson, B. L. Bihari, K. Salari, C. S. Woodward

January 3, 2007

Nuclear Explosives Code Developers Conference 2006
Los Alamos, NM, United States
October 23, 2006 through October 27, 2006

Program product Verification and Validation Methods

Program element Physics verification studies of ASC codes

UCRL-CONF-227042

Code Verification Results of an LLNL ASC Code on Some Tri-Lab Verification Test Suite Problems (U)

S. R. Anderson
B. L. Bihari
K. Salari
C. S. Woodward

Proceedings of the NECDC 2006

Code Verification Results of an LLNL ASC Code on Some Tri-Lab Verification Test Suite Problems (U)

S. R. Anderson, B. L. Bihari, K. Salari, C. S. Woodward

Lawrence Livermore National Laboratory, Livermore, California 94550

As scientific codes become more complex and involve larger numbers of developers and algorithms, chances for algorithmic implementation mistakes increase. In this environment, code verification becomes essential to building confidence in the code implementation. This paper will present first results of a new code verification effort within LLNL's B Division. In particular, we will show results of code verification of the LLNL ASC ARES code on the test problems: Su Olson non-equilibrium radiation diffusion, Sod shock tube, Sedov point blast modeled with shock hydrodynamics, and Noh implosion. (U)

Introduction

With the continued advancement of high performance computing hardware technologies and scalable algorithms which can efficiently solve mathematical models of physical phenomena, computational codes are more than ever able to deliver solutions to highly complex problems. As a result, simulation is increasingly relied on as a tool for providing scientific insight and discovery. Increasingly, simulation codes are developed by teams of scientists, mathematicians, computer scientists, and other domain specialists working on various modules or packages. In addition, these codes often include new and intricate mathematical algorithms. The inherent complexity in the development process leads to vulnerability for mistakes in code implementation.

Code verification, or "the process by which one verifies the theoretical order-of-accuracy of the numerical algorithms employed by the code to solve its governing equations" *(Knupp and Salari 2003)*, is essential to building confidence in the correctness of the code implementation. Such confidence, along with solid validation, is essential before one can use a simulation code for predictive purposes.

Developers and users of simulation codes have employed a number of techniques for code verification. Many of these are introduced in *(Knupp and Salari 2003)*, and we refer the reader to this book and the references therein for further information. In this paper, we consider order-of-accuracy verification due to its high level of rigor. In this approach to code verification, one employs a number of test problems with known highly accurate solutions that target different segments of the code. Systematic grid refinements

are then applied to extract the observed order-of-accuracy of the implemented discretization schemes.

Over the last several years, astrophysicists and others have conducted accuracy studies using a number of benchmark problems *(Brock et al., 2006, Fryxell et al., 2000, Timmes et al., 2005, Timmes et al., 2006)*. This set of problems tests specialized cases of diffusion, hydrodynamic motion, and heat conduction. This paper reports on a new effort in LLNL's B Division to conduct formal code verification. We have started with problems in the Tri-Lab test suite *(Brock et al., 2006)* as these problems tend to be well understood and provide a solid base for conducting verification.

The rest of the paper is organized as follows. In the next section we overview our approach to verification and order-of-accuracy evaluation. In the following section, we show verification results for the ARES ASC code on the following problems: Su Olson non-equilibrium radiation diffusion, Sod shock tube, Sedov point blast, and Noh implosion modeled with shock hydrodynamics. The last section provides a summary of our results to date.

Methods

For the purposes of computing order-of-accuracy of the implemented schemes, we make the assumption that the error in the computed solution has the following form,

$$\|error\| \le c_1(\Delta x)^{r_1} + c_2(\Delta t)^{r_2}, \qquad [1]$$

where c_1 and c_2 are constants independent of the grid spacing and time step size. Our goal is to estimate the exponents (r_1 or r_2) from the computed solutions. Our assumption that Eq. 1 is an appropriate model of the discretization error relies on the fact that the grid spacing and time step are small enough that we are in the asymptotic regime, i.e., that the errors are monotonically converging to the correct solution and that the coefficients c_1 and c_2 are indeed constant. In general, when starting an analysis of a given code and problem, we will not know when we would expect to be in the asymptotic range, and trial and error is used to establish whether the solutions are converging and whether we are close enough to be in the asymptotic range.

To identify r_1, we choose a small Δt, run the code on a number of successively refined spatial grids, and compute the errors. We then verify that making Δt smaller does not change the computed errors. This step assures us that the spatial error is dominant. We can then estimate r_1 for each pair of runs on successively refined grids through the following,

$$r_1 = log\left(\frac{error_{grid\_a}}{error_{grid\_b}}\right) \Big/ log\left(\frac{\Delta x_a}{\Delta x_b}\right). \qquad [2]$$

One can go through a similar process for r_2, but this exponent tends to be much more difficult to estimate. For hydrodynamic problems solved with explicit schemes, a stability condition relates the time step size to the grid spacing and limits the step size to be fairly small. As a result, we can try to compute r_2 by estimating the spatial error for a fixed grid size and varying time step size below the stability limited values. This is an area of current investigation for our effort.

In this paper, we examine both a diffusion problem with a smooth solution and hydrodynamic problems with shocks, or discontinuities, in the solutions. As a result, we consider three norms of the error over the problem domain. For cell-centered quantities, these are given as

$$L_1 : \|error\|_1 = \sum_i \left| f_{comp,i} - f_{exact,i} \right| \Delta x_i , \qquad [3]$$

$$L_2 : \|error\|_2 = \left(\sum_i \left| f_{comp,i} - f_{exact,i} \right|^2 \Delta x_i \right)^{1/2} , \qquad [4]$$

$$L_\infty : \|error\|_\infty = \max_i \left| f_{comp,i} - f_{exact,i} \right| , \qquad [5]$$

where $f_{comp,i}$ is the computed solution at point x_i, $f_{exact,i}$ is the exact solution at point x_i, $i = 1, \ldots, N$, and N is the number of cells in the domain. For point-centered variables, the sums again go over the number of cells, but the computed and exact values at each endpoint of the cell are arithmetically averaged, with adjustments made for non-uniform grids. The L_∞ norm gives a measure of the maximum error over the computational points of the domain. While this measure can be useful, for problems with shocks solved with methods that do not exactly capture the shock location, this norm will return values close to the size of the shock and may even oscillate with grid spacing depending on cell locations relative to shock position. For these problems, we use the L_1 norm as this norm tempers the error at the shock with the grid spacing. Lastly, for smooth problems, such as diffusion, we apply the L_2 norm. The above norms are used to evaluate observed order-of-accuracy which in turn is compared to the theoretical order-of-accuracy of the implemented numerical schemes. This comparison will establish the degree of confidence in the code verification process.

Results

In this section, we show results of order-of-accuracy studies for the following test problems: Su Olson radiation diffusion, Sod shock tube, Sedov blast, and Noh implosion.

Before beginning, we note that the code cannot be run with a constant time step size. In all runs, the time step is initialized to a small value and allowed to ramp up to a specified maximum value. The final time step size is truncated to enforce the stop time.

Su Olson Problem

The Su Olson problem *(Su and Olson, 1996)* is a 1D, planar, half-space Marshak wave using a Planckian diffusion radiative transfer model coupled to a material temperature model. The diffusion is single group and is driven through a mixed type boundary condition. As the radiation energy increases, energy is transferred to the material. The problem is solved for the time dependent dimensionless radiation energy density and material temperature. The analytic solution is formulated in terms of integrals which contain singularities and which require careful attention for accurate numerical integration. We employed an analytic solution code generated by F. Timmes of LANL *(Timmes, et al., 2005)*. Through comparison against Mathematica calculations of the analytic solution, we determined that the LANL code is accurate to at least 5-6 digits.

Su and Olson's original specification of the problem included results for two coefficients on the incoming flux. We chose to look at the case for the lesser one, $\varepsilon = 0.1$. As specified, the problem has a homogeneous initial condition which we modeled using 10^{-5}. Lastly, the problem is semi-infinite, and we modeled out to *20 cm*. We stopped the simulation at $0.004\ \mu s$. At this time, the wave front is not near the right boundary, so this domain boundary should have no effect on the presented results. The code employs a second order scheme for spatial discretization of diffusion operators and a first order time stepping scheme. As a result, we expect $r_1 = 2$ and $r_2 = 1$.

The time step history for a maximum step of $10^{-4}\ \mu s$ is given in Figure 1a. We see that by making the maximum step size $10^{-5}\ \mu s$, the code will reach that value quickly and maintain a constant step size thereafter.

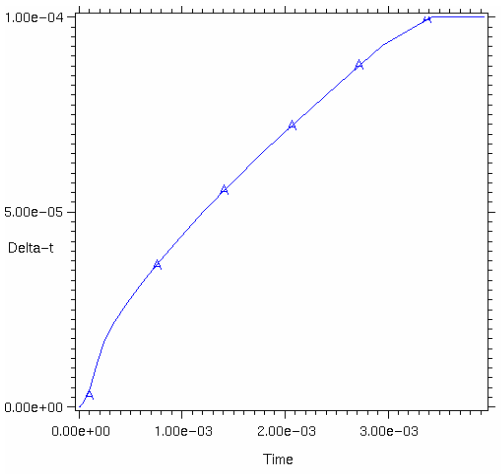

Figure 1a: Time history for Su Olson problem runs with maximum step size set to 1.0e-4.

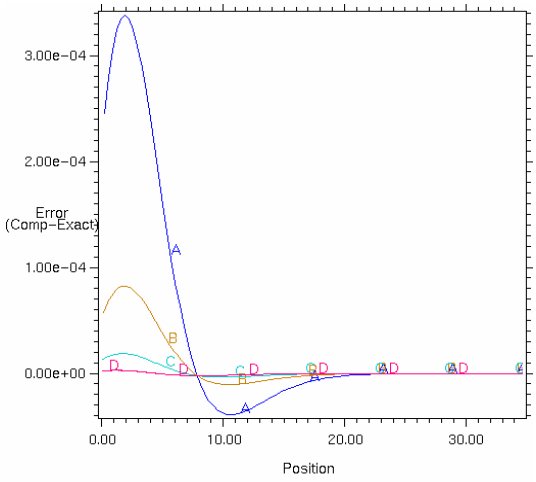

Figure 1b: Differences of computed and exact dimensionless energy densities for Su Olson problem for grids of (A) 100, (B) 200, (C) 400, and (D) 800 cells.

Figure 1b shows the differences between the computed and exact dimensionless energy density for 4 grids. We see the differences decreasing with grid refinement as expected. Although not shown, the dimensionless temperature variable shows the same trend. Table 1 shows computed error norms and rates of spatial convergence for this problem with a maximum time step of 10^{-8} μs. We see that energy density shows a slightly higher than expected convergence rate, which requires further investigation, but the temperature gives results very close to the expected second order value.

Table 1. Spatial discretization results for Su Olson problem

# Zones		100	200	400	800
Energy Density	L_2-norm	5.01e-4	1.22e-4	2.72e-5	5.21e-6
Energy Density	r_1		2.04	2.16	2.38
Temperature	L_2-norm	5.29e-4	1.29e-4	2.88e-5	5.61e-6
Temperature	r_1		2.03	2.10	2.04

Table 2 shows computed error norms and rates of temporal convergence for this problem with 800 spatial zones. Here we see the expected first order convergence once the step size is decreased below 10^{-6} μs. We conclude that the time step size must be this small for the spatial discretization error to be dominant.

Table 2. Temporal discretization results for Su Olson problem

Max. time step (μs)		10^{-4}	10^{-5}	10^{-6}	10^{-7}
Energy Density	L_2-norm	2.23e-3	6.29e-4	9.18e-5	8.15e-6
Energy Density	r_2		0.55	0.84	1.05
Temperature	L_2-norm	2.36e-3	6.56e-4	9.61e-5	8.69e-6
Temperature	r_2		0.56	0.83	1.04

Sod Shock Tube

The Sod problem *(Sod 1978)* is a 1D shock tube where the initial discontinuity includes a pressure ratio of *10* and a density ratio of *8*. The gas is at rest on both sides of the membrane, and the polytropic index is $\gamma = 1.4$ on both sides. This is a purely hydrodynamic problem that tests the code's ability to accurately capture the three important wave transitions: rarefaction waves, contact discontinuities, and shocks. Because the latter two waves are, in principle, true discontinuities and this is a genuinely

nonlinear system of equations, we expect a spatial order of accuracy no higher than first order.

This problem requires the solution of the Euler equations which are a hyperbolic system of conservation laws. There are three real eigenvalues, and the solution is constant along characteristics. Therefore, there exists a similarity solution that depends only on x/t and is constant along these characteristics. For the specific flow conditions outlined above, one can obtain an analytic solution by solving a system of six nonlinear algebraic equations with six unknowns. These equations come from the continuity (invariance) of the Riemann invariants across the three wave transitions – two for each wave. We employed Newton's method in our exact solution code to solve for these invariants. Once the intermediate states that connect the wave transitions are obtained, the exact solution is computed through an evaluation of the similarity solution along the characteristics. We used a low tolerance for stopping Newton's method (*1.0e-15*) and verified that the exact solution results were insensitive to this tolerance at this low value.

Since the physics code cannot run a truly 1D problem, we ran the Sod problem in 2D mode and specified a single cell in the transverse (y) direction. The size of the domain in the y -direction was kept constant (*1.0 cm*) throughout the refinement process in x. The domain was taken to be *2.0 cm* long, and runs were done to a stop time of *0.4 μs*. For all runs we took the time step to be *0.5* times the CFL condition. The solution for this problem consists of three waves moving with speeds u, $u+c$ and $u-c$, where u is the velocity and c is the speed of sound. The rarefaction wave moves to the left with speed $u-c$, while the contact discontinuity and the shock move to the right with speeds u and $u+c$, respectively. The two end boundary conditions are fixed at initial values, and the calculations were stopped before any wave reached the boundary. Except for the expansion (rarefaction) region, the solution contains piecewise constant segments separated by discontinuities.

Table 3 shows results in both the L_∞ and L_1 norms for running the Sod problem in Lagrangian mode. We see a spatially first order accurate result in the L_1 norm for density and pressure but not velocity in the first four grid refinements. The convergence rate further deteriorates for all computed variables on the most refined grids. This deterioration is an area of current investigation and could be due to a number of factors, including interpolation roundoff errors, roundoff stemming from very high aspect ratios, and implementation mistakes. In addition, it is possible that the Lagrangian step is accomplished by uneven movement of grid points along the $y = 0$ and $y = 1$ boundaries when the grid becomes dense. We also see that the L_∞ norm is either close to constant or even slightly oscillatory. This phenomenon arises due to the inability of the numerical scheme to exactly capture the shock position. This behavior is typical of shocked flows where the scheme is not fully conservative.

Table 3. Spatial convergence results for Sod problem in Lagrangian mode

# Zones		360	720	1,440	2,880	5,760	11,520
Density	L_1-norm	6.24e-3	3.07e-3	1.50e-3	7.88e-4	3.99e-4	2.23e-4
	r_1		1.02	1.03	0.93	0.98	0.84
	L_∞-norm	1.72e-1	1.72e-1	1.72e-1	1.72e-1	1.72e-1	1.72e-1
	r_1		0.0	0.0	0.0	0.0	0.0
Pressure	L_1-norm	3.43e-3	1.70e-3	8.51e-4	4.22e-4	2.10e-4	1.09e-4
	r_1		1.01	1.00	1.01	1.01	0.95
	L_∞-norm	6.73e-2	5.43e-2	7.52e-2	6.16e-2	6.14e-2	9.13e-2
	r_1		0.31	-0.47	0.29	0.01	-0.57
Velocity	L_1-norm	7.97e-3	4.16e-3	1.76e-3	1.01e-3	4.34e-4	2.36e-4
	r_1		0.94	1.24	0.79	1.23	0.88
	L_∞-norm	5.86e-1	6.57e-1	3.2e-1	6.16e-1	2.57e-1	4.73e-1
	r_1		-0.17	1.04	-0.94	1.26	-0.88

Figure 2 shows the computed density, pressure, and velocity for this problem on a number of grid resolutions. Pressure and velocity are Riemann invariants across the contact discontinuity. There are no visible undershoots or overshoots at the shock. However, there are small, but visible oscillations in the neighborhood of the contact discontinuity which do not seem to decrease appreciably with resolution. The small oscillations near the corners of the rarefaction wave seem to be responding to grid refinement, however We should note that there were no undershoots at the base of the shock in either density or pressure. Lastly, we note that total energy did not change with time for all grids.

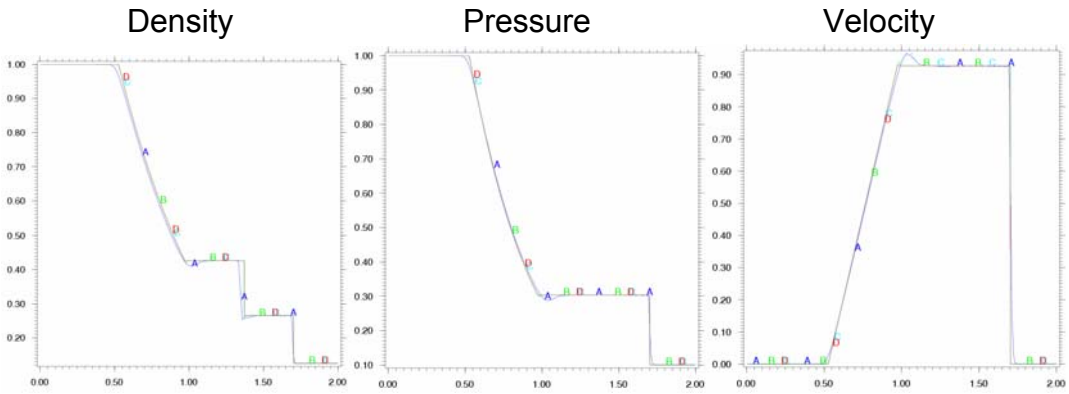

Figure 2: Computed and exact density *(gm/cc)*, pressure *(Mbars)*, and velocity *(cm/μs)* versus position *(cm)* for Sod problem on 180, 2880, and 11520 cell grids.

Sedov Problem

The Sedov problem *(Sedov 1959)* models a 1D spherical point explosion initiated by a set quantity of energy deposited at the origin. The analytical solution is self-similar and yields pressure, density, and fluid velocity. Generating the self-similar solution requires numerical integration of an ODE and a numerical root-find. We employed an analytic solution code generated by F. Timmes of LANL *(Timmes, et al., 2005)* and verified that the generated solution is insensitivie to the tolerances for the root-find and numerical integration at the level of resolution of the physics code. This is a purely hydrodynamic problem that tests the code's ability to accurately capture the position and magnitude of an outward moving shock. Due to the shock present in the solution, we again do not expect greater than first order spatial accuracy.

We conducted verification studies for the Lagrangian formulation using a standard set of parameters for this problem. In particular, we solved the problem on a sphere with radius of *1.2 cm*, an initial energy of *4,935.9e12 ergs*, an ideal gas with $\gamma = 1.4$, a stop time of *0.01 μs*, and a fixed inner radius of *0.01 cm* where the initial energy is deposited for all grid resolutions. For all runs we took the time step to be *0.5* times the CFL condition.

Figure 3 shows the computed and exact density, pressure, and velocity for running the Sedov problem in Lagrangian mode on a number of grid resolutions. We see that for very coarse grids, the code predicts the shock location too far out. However, the location continues to get closer to the exact solution with grid refinement and by the finest grids, we are visually close to the true solution.

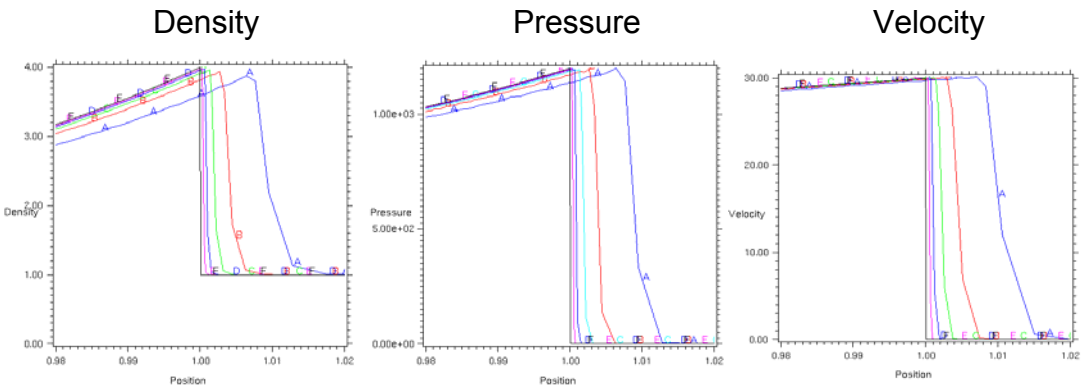

Figure 3: Computed and exact density *(gm/cc)*, pressure *(Mbars)*, and velocity *(cm/μs)* versus position *(cm)* for Sedov problem in Lagrangian mode on 240, 480, 960, 1920, and 3840 cell grids.

Table 4 shows results in both the L_∞ and L_1 norms for running the Sedov problem in Lagrangian mode. For both density and pressure we see that for fairly large grid spacings, the code is in the asymptotic regime, and we see first order spatial convergence. Velocity is a little slower to converge, however, but does show first order convergence by the fourth grid refinement. As with the Sod problem, we also see that the L_∞ norm is close to constant. As can be seen from Figure 3, the norm is just reflecting the size of the shock since the computed solutions are approaching the exact solution from the right.

Table 4. Spatial convergence results for the Sedov problem in Lagrangian mode

# Zones		240	480	960	1,920	3,840
Density	L_1-norm	5.70e-2	2.63e-2	1.31e-2	6.50e-3	3.01e-3
	r_1		1.12	1.01	1.01	1.11
	L_∞-norm	2.88	2.94	2.97	2.98	2.99
Pressure	L_1-norm	18.6	8.44	4.15	2.05	9.10e-1
	r_1		1.14	1.02	1.02	1.17
	L_∞-norm	1.20e3	1.20e3	1.20e3	1.20e3	1.20e3
Velocity	L_1-norm	4.09e-1	1.81e-1	1.13e-1	5.52e-2	2.57e-2
	r_1		1.18	0.67	1.04	1.10
	L_∞-norm	30.1	30.0	30.0	30.0	30.1
% Change in Total Energy		4.37	1.91	0.98	0.46	0.21

We noted above that the behavior of schemes to miss the shock position is typical of shocked flows where the scheme is not fully conservative. Figure 4 shows total energy versus time for the Sedov problem, and from it we can see that the scheme is indeed not

fully conservative. We see for all cases that the total energy increases shortly after the simulation begins. As the grid is refined, however, we also see that the amount of increase in energy decreases and appears to be converging to a small number. We would expect this convergence given the above quantitative results indicating that the solution is significantly improving with grid refinement. In fact, as we saw above, the solution is converging about linearly with grid refinement, and we see this same trend in the percent change of total energy given in Table 4.

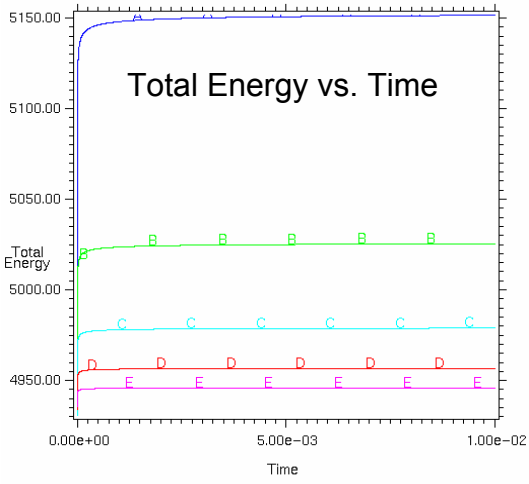

Figure 4: Total energy *(10^{12} erg)* versus time *(μs)* for the Sedov problem in Lagrangian mode on various grids: (A) 240, (B) 480, (C) 960, (D) 1,920, (E) 3,840.

We conducted a second study with the Sedov problem using the Eulerian mode. Here we used different problem specifications corresponding to a set of parameters received from the code group. In particular, we solved the problem on a sphere with radius of *1.2 cm*, an initial energy of *1.0e12 ergs*, an ideal gas with $\gamma = 1.4$, a stop time of *0.12 μs*, and a fixed inner radius of *0.01 cm* where the initial energy is deposited for all grid resolutions. For all runs we took the time step to be *0.5* of that given by the CFL condition.

Figure 5 shows the computed and exact density, pressure, and velocity for running the Sedov problem in Eulerian mode on a number of grid resolutions. We see here that the shock is generally in the correct position, but the scheme in the code diffuses it too much on coarse grids. With grid refinement, however, we see the front get sharper, and by the finest grids, we are visually close to the exact solution.

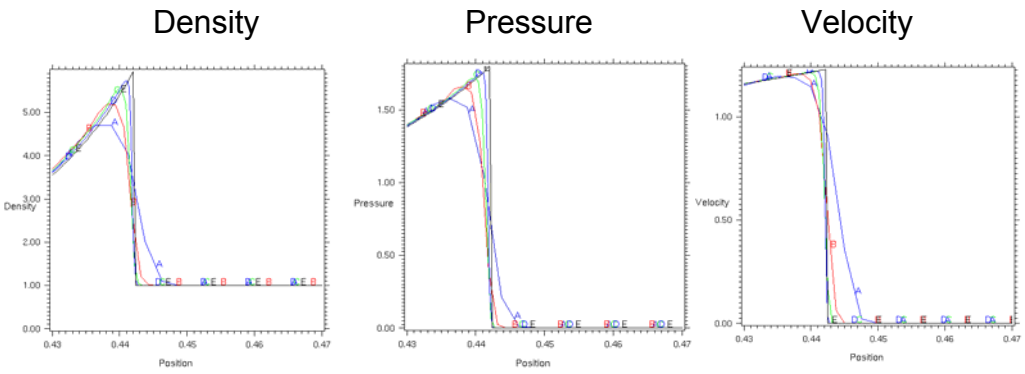

Figure 5: Computed and exact density *(gm/cc)*, pressure *(Mbars)*, and velocity *(cm/μs)* versus position *(cm)* for Sedov in Eulerian mode on 400, 800, 1600, and 3200 cell grids.

Table 5 shows results in both the L_∞ and L_1 norms for running this problem in Eulerian mode. From this data, we can see that we are not yet in the asymptotic regime, as the convergence rates have not fully "settled down" to a steady value. Thus, we cannot say yet whether we see the expected first order convergence in the L_1 norm for density, pressure, and velocity. We do see that both pressure and velocity seem to be approaching the first order convergence, and we would expect to see this rate with further refinement. At this time, we do not have a definite explanation for why the velocity is not showing signs of convergence, and we are actively investigating possible causes.

Table 5. Spatial convergence results for the Sedov problem in Eulerian mode

# Zones		200	400	800	1,600	3,200
Density	L_1-norm	2.25e-2	2.24e-2	2.00e-2	8.42e-3	4.00e-3
	r_1		0.00	0.17	1.25	1.07
	L_∞-norm	1.61	3.42	4.63	4.50	3.76
	r_1		-1.08	-0.44	0.04	0.26
Pressure	L_1-norm	1.21e-2	4.97e-3	4.27e-3	3.46e-3	1.54e-3
	r_1		1.28	0.22	0.30	1.17
	L_∞-norm	1.25	6.03e-1	1.32	1.74	1.53
	r_1		1.06	-1.13	-0.40	0.19
Velocity	L_1-norm	9.51e-3	4.13e-3	1.26e-3	7.01e-4	7.36e-4
	r_1		1.20	1.72	0.84	-0.07
	L_∞-norm	1.02	9.66e-1	5.88e-1	4.80e-1	8.01e-1
	r_1		0.08	0.71	0.29	-0.74
% Change in Total Energy		2.30	-2.45	-1.96	-1.31	-0.71

Unlike the previous two problems, we see that the L_∞ norm is oscillatory for all quantities. These oscillations are due to the fact that the computed solution has the shock very close to the correct position, but, since we do not have nested grids, computed errors near the shock may increase or decrease as cell centers are changed.

Figure 6 shows the total energy versus time for these runs. We see that total energy decreases near the beginning of the run and that this decrease gets smaller with grid refinement below 400 cells. For very coarse grids, the energy actually increases then levels out. This trend can be seen from the percent change in energy in Table 5. For more refined grids, convergence of total energy conservation is still getting significantly better with grid refinement, and we conclude that we are not yet fully in the asymptotic regime where we would expect to be very close to the solution.

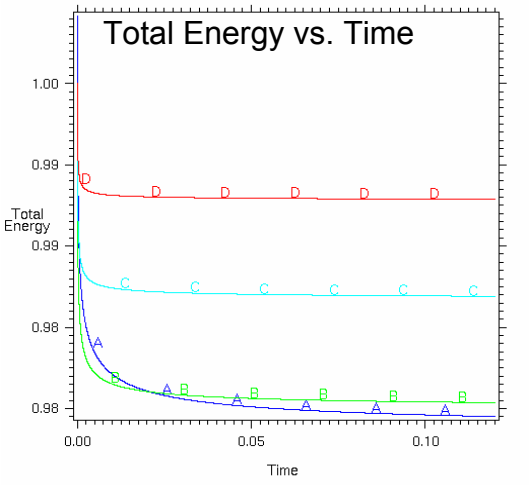

Figure 6: Total energy *(10^{12} erg)* versus time *(μs)* for the Eulerian mode on varying grid resolutions: (A) 400, (B) 800, (C) 1600, and (D) 3200.

Noh Problem

The Noh problem *(Noh, 1987)* models a 1D spherical hydrodynamic shock initiated in an ideal gas with a uniform, radially inward fluid flow. The shock forms at the origin and propagates outward as the flow stagnates. The analytic solutions for density, pressure, and fluid velocity are straightforward. Only algebraic computations are needed and no numerical integrations or root-finds are required. Again, due to the presence of a shock in the solution, we do not expect greater than first order spatial accuracy.

For this problem, we conducted verification studies for the Lagrangian formulation using a standard set of parameters. In particular, we solved the problem on a sphere with radius of *1.0 cm*, an initial velocity of *-1.0 cm/μs*, an ideal gas with $\gamma = 5/3$, a stop time of *0.6 μs*, and a reference density of *1.0 gm/cm^3*. For all runs we took the time step to be *0.5* times the CFL condition.

Table 6 shows results in both the L_∞ and L_1 norms for running the Noh problem in Lagrangian mode. For both velocity and pressure we see that for fairly coarse grids, the code is in the asymptotic regime, and we see the expected first order spatial convergence. Density is converging but at a slower rate. Although not shown here, the L_∞ norm is close to constant for all solution variables, and we again conclude that this just reflects the size of the shock.

Table 6. Spatial convergence results for the Noh problem in Lagrangian mode

# Zones		200	400	800	1,600	3,200
Density	L_1-norm	9.34e-1	5.22e-1	2.88e-1	1.58e-1	8.55e-2
	r_1		0.84	0.86	0.87	0.88
Pressure	L_1-norm	9.65e-2	4.87e-2	2.44e-2	1.24e-2	6.11e-3
	r_1		0.99	1.00	0.98	1.02
Velocity	L_1-norm	4.98e-3	2.50e-3	1.25e-3	6.26e-4	3.15e-4
	r_1		0.99	1.00	1.00	0.99
% Change in Total Energy		-0.172	-0.085	-0.042	-0.021	-0.011

Figure 7a shows computed and exact densities for varying grid sizes. As expected from the above data, the computed solutions are indeed converging to the exact solution. We also see that the most significant deviations from the exact solution are nearest the "wall" or left boundary, as would be consistent with wall heating. Figure 7b shows total energy as a function of time for the 4 most refined grids. As with the other test problems, we see that total energy is not conserved. The difference here is that the scheme slowly loses energy over time as opposed to suffering a loss at the start and conserving from that point forward. We also see that as the grid is refined, energy conservation improves significantly. We note on this problem that the code developers choose a discretization scheme that more accurately handles compression as a trade-off to more precisely conserving energy.

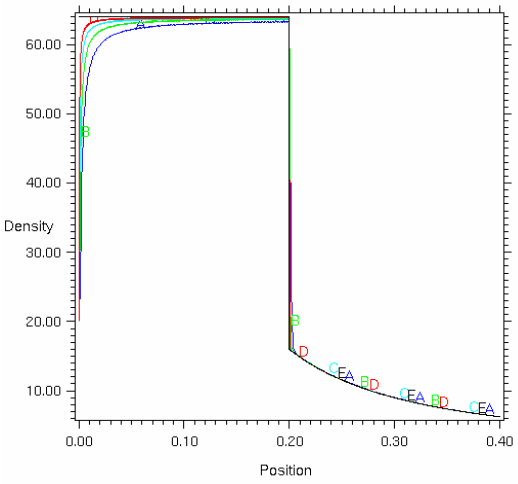

Figure 7a: Computed and exact density as a function of radius for the Noh problem for varying grids: (A) 400, (B) 800, (C) 1600, (D) 3200, and (E) exact.

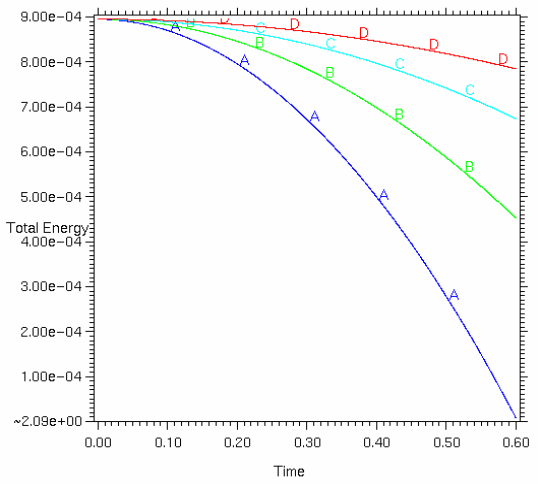

Figure 7b: Total energy versus time for the Noh problem for varying grids: (A) 400, (B) 800, (C) 1600, and (D) 3200.

Summary

We have shown initial results of code verification studies of an LLNL ASC code on some Tri-Lab Test Suite problems. Results on the Su Olson diffusion test problem show that the code is giving at least the expected rates of convergence for both the spatial and temporal discretizations. The Sod shock tube problem revealed first order spatial convergence in the L_1-norm as expected with minor degradation for very refined meshes. We saw good first order spatial convergence in the L_1-norm for the Sedov problem with Lagrangian mode, but not as good results in Eulerian mode. Lastly, we also saw first order spatial convergence for pressure and velocity for the Noh problem, but a rate of only about *0.88* for the density. These results for the Noh problem potentially highlight a modification of the scheme that still is correct but with lower order-of-accuracy for the density or a possible coding mistake. The L_∞-norm for these problems showed no convergence, but this was expected since algorithms implemented in the code reflect a tradeoff of energy conservation for other properties. If a scheme does not precisely conserve energy, the shock position will not be computed perfectly and the L_∞-norm will reflect the shock size. In light of this last issue, it should be noted that the algorithms in the code should be considered in choosing the norms for order-of-accuracy assessments.

Acknowledgements

The authors wish to thank Brian Pudliner, David Miller, and Scott Brandon for numerous helpful discussions of code behavior and verification process. This work was performed under the auspices of the U.S. Department of Energy by the University of California Lawrence Livermore National Laboratory under contract W-7405-Eng-48. The release number for this document is UCRL-CONF-227042.

References

Brock, J.S., Kamm, J.R., Rider, W.J., Brandon, S., Woodward, C.S., Knupp, P., and Trucano, T.G., *Verification Test Suite for Physics Simulation Codes*, Los Alamos National Laboratory, Los Alamos, NM, LA-UR-06-8421 (2006).

Fryxell, B., Olson, K., Ricker, P., Timmes, F.X., Zingale, M., Lamb, D.Q., MacNeice, P., Rosner, R., Truran, J.W., and Tufo, H., "FLASH: An Adaptive Mesh Hydrodynamics Code for Modeling Astrophysical Thermonuclear Flashes," *The Astrophysical Journal Supplement Series*, **131**, 273-334 (2000).

Knupp, P., and Salari, K., *Verification of Computer Codes in Computational Science and Engineering*, (Chapman and Hall/CRC, Boca Raton, FL, 2003).

Noh, W.F., "Errors for Calculations of Strong Shocks Using and Artificial Viscosity and an Artificial Heat Flux," *J. Comput. Phys.*, **72**, 78-120, (1987).

Roache, P.J., *Verification and Validation in Computational Science and Engineering*, (Hermosa, Albuquerque, 1998).

Sedov, L.I., *Similarity and Dimensional Methods in Mechanics*, (Academic Press, New York, 1959).

Sod, G.A., "A Survey of Several Finite Difference Methods for Systems of Nonlinear Hyperbolic Conservation Laws," *J. Comput. Phys.*, **27**, 1-31, (1978).

Su, B. and Olson, G.L., "Benchmark Results for the Non-Equilibrium Marshak Diffusion Problem," *J. Quant. Spectro. Radiat. Transfer*, **56**, 337-351, (1996).

Timmes, F.X., Gisler, G., and Hrbek, G.M., *Automated Analyses of the Tri-Lab Verification Test Suite on Uniform and Adaptive Grids for Code Project A*, Los Alamos National Laboratory, Los Alamos, NM, LA-UR-05-6865 (2005). Solution codes: LA-CC-05-101, available at www.cococubed.com/code_pages/vv.shtml.

Timmes, F.X., Gisler, G., and Hrbek, G.M., *Analytic Solution Codes for Tri-Lab Test Suite Problems*, Los Alamos National Laboratory, Los Alamos, NM, LA-CC-05-101 (2005). Available at http://www.cococubed.com/code_pages/vv.shtml.

Timmes, F.X., Fryxell, B., and Hrbek, G.M., *Spatial-temporal convergence properties of the Tri-Lab Verification Test suite in 1D for code project A*, Los Alamos National Laboratory, Los Alamos, NM, LA-UR-06-6444 (2006).