LA-UR- 11-00062

Approved for public release;
distribution is unlimited.

Title:	A General Higher-Order Remap Algorithm for ALE Calculations
Author(s):	Vincent Chiravalle
Intended for:	NECDC 2010 Proceedings

Los Alamos
NATIONAL LABORATORY
EST. 1943

Form 836 (7/06)

A General Higher-Order Remap Algorithm for ALE Calculations

Vincent Chiravalle (XTD-2)

Los Alamos National Laboratory, Los Alamos, New Mexico 87545

A numerical technique for solving the equations of fluid dynamics with arbitrary mesh motion is presented. The three phases of the Arbitrary Lagrangian Eulerian (ALE) methodology are outlined: the Lagrangian phase, grid relaxation phase and remap phase. The Lagrangian phase follows a well known approach from the HEMP code; in addition the strain rate and flow divergence are calculated in a consistent manner according to Margolin. A donor cell method from the SALE code forms the basis of the remap step, but unlike SALE a higher order correction based on monotone gradients is also added to the remap. Four test problems were explored to evaluate the fidelity of these numerical techniques, as implemented in a simple test code, written in the C programming language, called Cercion. Novel cell-centered data structures are used in Cercion to reduce the complexity of the programming and maximize the efficiency of memory usage. The locations of the shock and contact discontinuity in the Riemann shock tube problem are well captured. Cercion demonstrates a high degree of symmetry when calculating the Sedov blast wave solution, with a peak density at the shock front that is similar to the value determined by the RAGE code. For a flyer plate test problem both Cercion and FLAG give virtually the same velocity temporal profile at the target-vacuum interface. When calculating a cylindrical implosion of a steel shell, Cercion and FLAG agree well and the Cercion results are insensitive to the use of ALE.

Introduction

The field of Arbitrary Lagrangian Eulerian (ALE) hydrodynamics has enjoyed a long and fruitful development. ALE is an indispensible tool for researchers studying energetic materials, plasmas, and turbulence among many other applications. There are two parts to ALE: the relaxation of the computational mesh and the remapping of mass, energy and momentum onto the new mesh. Various approaches for both mesh relaxation and remapping have been explored in the literature. An excellent summary of the various methods is given by Benson[1].

One of the first ALE codes, SALE[2], was developed in 1981. SALE solves the hydrodynamic equations in three sequential phases: a Lagrangian phase, a mesh relaxation phase and a remap phase. The momentum equations and the internal energy equations are updated during the Lagrangian phase in a manner similar to other purely Lagrangian codes such as HEMP[3]. In HEMP the material strength treatment first involves finding the flow strains from the velocity field, updating the von Mises flow stress using an appropriate model such as the PTW model[4] or the Steinberg model[5], and then finally using the updated flow stress together with a yield surface criterion to find the stress deviator components. Unlike HEMP, SALE does not utilize material strength. Subsequent to SALE's development, a similar code was formulated, SHALE[6], which not only incorporated ALE but also had an improved strength treatment relative to HEMP. Both HEMP and SHALE solve the fluid equations in a two dimensional geometry with cylindrical symmetry and there are four components of the stress deviator tensor with three of these being independent. The stress deviator components enter into the momentum equation and the stress related work term enters into the energy equation. During

the Lagrangian phase in SALE the momentum equations are solved on a grid that is staggered spatially with respect to the corresponding grid for the cell-centered quantities such as mass, energy and the stress deviator components. This staggered grid arrangement is also used in HEMP. In addition to spatial staggering, the vertex-center velocities are also temporally staggered in HEMP.

Having obtained new values for velocity after the Lagrangian phase, SALE then updates the positions of the vertices on the mesh. At this point mesh relaxation is considered and SALE has three options: maintain the Lagrangian positions, remap to the initial mesh positions (Eulerian method) or remap to positions corresponding to a user specified weight between these two extremes. As research in ALE methods continued and as ALE codes began to be more widely applied, sophisticated techniques were developed for mesh relaxation. Contemporary methods involve the optimization of a global integral relating to smoothness, orthogonal nature of the grid lines, and other properties of the mesh, as described by Brackbill and Saltzman[7]. Minimizing the integral involves a few iterations on the vertex positions each cycle using the conjugate gradient technique. An exact solution for the optimum vertex positions is not obtained at any given cycle but with time the mesh converges to the optimum solution. A simpler alternative to a global optimization method is the regular use of a finite difference vertex relaxer such as the one proposed by Winslow[8] which adjusts the vertex points locally using only the positions of neighboring vertices to achieve an equipotential mesh in an iterative fashion after many cycles.

The task of remapping the physical quantities from the old mesh to the new mesh is performed during the remap phase. SALE utilizes a first order donor-cell method for the mass and energy remap. The donor cell method involves determining flux volumes for each side of the cell associated with the movement of material from the old to the new mesh, and assigning a material density and energy density to these volumes. It is well known that first-order methods do not preserve sharp gradients as discussed by Laney[9] and therefore these methods are rarely used in ALE codes. Higher-order remap methods have been developed and two good examples are the second order sign-preserving method of Margolin and Shaskov[10] and the Barth-Jespersen method[11]. These higher order methods add corrections to the fluxes computed by the donor-cell method. In particular the Barth-Jespersen method constructs a density slope at each cell center, suitably limited to preserve monotonicity. For each remapped quantity the appropriate density slope is used to construct the flux, be it a mass or energy.

Both SALE and SHALE do not allow remapping across material interfaces; it is assumed that material interfaces always evolve in a Lagrangian fashion. This assumption can be problematic in many situations where turbulence arises. One way to circumvent this difficulty is to use an interface reconstruction technique such as the one developed by Youngs[12]. Youngs' volume of fluid (VOF) method represents a material interface in a piecewise linear fashion. In each cell there are three parameters describing the local straight line representation of the interface. The slope of the line is determined by the local volume fraction gradient and its position relative to the cell center is determined to match the volume of material in the cell. Additional computational details of the VOF method and a description of other alternate methods for interface reconstruction are given by Morgan[13].

A central issue for any code that solves the equations of hydrodynamics, be it Lagrangian, Eulerian or ALE, is the type of artificial viscosity employed to capture non-linear discontinuities such as shock waves. Since its introduction in 1950 by von Neumann and Richtmyer, the scalar artificial viscosity, with a quadratic dependence on velocity difference has been ubiquitous in the CFD literature. The corresponding artificial stress term is added to the pressure in the momentum equation to suppress numerical oscillations in regions with strong shocks. Bowers and Wilson[14] describe how to implement the scalar artificial viscosity in a two dimensional, cylindrically symmetric Eulerian code. In this case the radial and axial components are treated separately but analogously and the two components of artificial viscosity each have both a quadratic term and a linear term. Wilkins[15] gives an excellent discussion of a tensor artificial viscosity in a two dimensional cylindrical geometry and Christiansen[16] describes a higher order flux-limited artificial viscosity.

The aim of this work is to numerically implement the ALE method in a new research code called Cercion, using the proven Lagrangian scheme from HEMP together with a simplified variation of the Barth-Jespersen remapping technique incorporating a VOF approach for material interfaces in mixed cells. The details of the numerical implementation are discussed in Section 2, including the novel cell-centered data structures that are used in Cercion to reduce the complexity of the programming and maximize the efficiency of memory usage. In Section 3, a series of three test problems are considered to validate Cercion; these problems include the Riemann shock tube, the Sedov blast wave and the aluminum flyer plate. Comparisons are made with analytic solutions and calculations using other fluid dynamics codes. In Section 4, a cylindrical implosion of a steel shell is studied, both with and without material strength, and the results are compared with FLAG calculations. I finish with a summary and conclusions in Section 5.

2.0 Numerical Implementation

Cercion follows a numerical implementation of the ALE technique that is similar to SALE but there are some key differences and additional features. The hydrodynamics is solved on a block-structured mesh where each mesh block contains only quadrilateral cells arranged in a regular fashion. A given mesh block has four faces and four corner points. The user specifies in the input deck how each of the mesh blocks is connected to the other blocks in terms of the boundary faces and corners. Velocities are stored at the vertices of the cell, whereas density and pressure are stored at the cell center. The indexing convention for a cell is shown in Figure 1. Cercion was written using the ANSI standard C programming language, with a Fortran style of indexing for the array data structures.

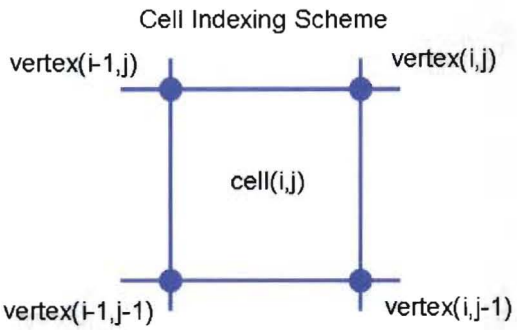

Figure 1. Cell and vertex indexing scheme.

There are three essential elements to the numerical implementation in Cercion: the cell-centered data structures, the Lagrangian phase velocity update, and the higher order remap method.

The data structures in Cercion are cell-oriented. Each cell is represented by a composite data type, cell_t, including a dynamic material list as well as storage for all of the cell-centered quantities, such as mass, pressure and specific energy, and storage for the top-right vertex positions and velocities. With this data type, information related to the cell can be obtained without searching various disparate and unconnected arrays.

An example of a dynamic linked list of materials is given in Figure 2.

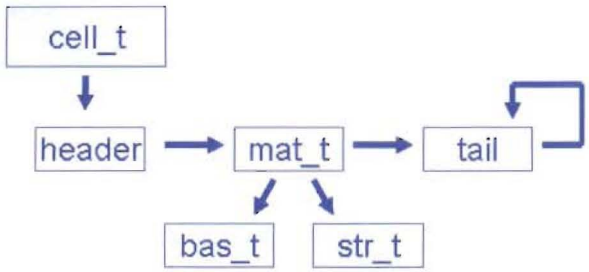

Figure 2. Linked list for a cell with 1 material.

Within the linked list there is a single element for each material in the cell; a composite data type, mat_t, is used to store all the information for a given material. The elements in the dynamic material list are ordered according to the onion skin method. Each element points to two other data types, bas_t and str_t. Basic material properties including volume fraction, mass, energy, density, density derivatives, and VOF interface parameters are stored in a bas_t data structure. For materials that have strength, strength properties are stored in a str_t data structure. Strength properties encompass stress deviator components, equivalent plastic strain, and the derivatives associated with these quantities.

In addition to a dynamic material list, the cell_t data structure also contains pointers to two other linked lists, one associated with the top boundary of the cell and another associated with the right boundary. These material flux linked lists each contain information about the materials crossing a cell boundary. An example of a material flux linked list with one material in the list is given in Figure 3.

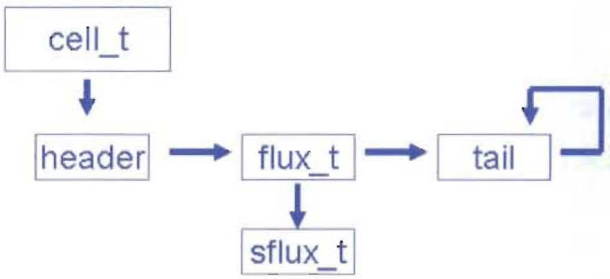

Figure 3. Material flux linked list with 1 material crossing the cell boundary.

A material flux linked list is comprised of a single data structure, flux_t, for each material crossing the boundary. The flux_t data structure stores the volume, mass and energy fluxes for the material. Furthermore, for those materials having strength, the flux_t data structure points to a sflux_t data structure holding the stress energy fluxes and strain fluxes. With these basic data structures the three phase solution to the hydrodynamic equations is implemented.

The first part of the hydrodynamic solution involves solving for the updated velocity components using the Lagrangian algorithm from HEMP which constructs a control volume around vertex (i,j) that resembles a diamond with the faces of the diamond passing through the centers of cells A, B, C, and D as shown in Figure 4. The corners of the diamond are the four neighboring vertices.

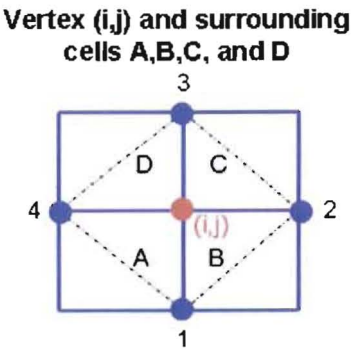

Figure 4. HEMP control volume for the velocity at vertex (i,j).

The finite difference equation for the axial velocity component, u, is given as Equation 1 and the radial component, v, as Equation 2. The values of the Cauchy stress tensor components, stored at the cell centers, are used to update the velocity components. The Cauchy stress tensor, Σ, includes the pressure, the stress deviators and the artificial viscous stress for numerical stability. In a cylindrical geometry, there are additional contributions to the velocity equations from the stress deviator tensor proportional to 1/r; these are denoted as β and α in Equation 3. An area-weighted scheme is used in HEMP for calculating these terms as illustrated below. The same method is applied in Cercion but the true mass of a cell over the complete solid angle is used, whereas HEMP does not use the true mass, hence the $\pi/2$ factors in Equation 3. The corresponding difference equations in HEMP have factors of 1/4 instead.

$$u_{i,j}^{n+1} = u_{i,j}^{n} - \frac{\Delta t}{2\phi_{i,j}}\left[\Sigma_{xx}^{A}(y_4 - y_1) + \Sigma_{xx}^{B}(y_1 - y_2) + \Sigma_{xx}^{C}(y_2 - y_3) + \Sigma_{xx}^{D}(y_3 - y_4)\right]$$
$$+ \frac{\Delta t}{2\phi_{i,j}}\left[\Sigma_{xy}^{A}(x_4 - x_1) + \Sigma_{xy}^{B}(x_1 - x_2) + \Sigma_{xy}^{C}(x_2 - x_3) + \Sigma_{xy}^{D}(x_3 - x_4)\right] + \Delta t\alpha_{i,j} \quad (1.)$$

$$v_{i,j}^{n+1} = v_{i,j}^{n} + \frac{\Delta t}{2\phi_{i,j}}\left[\Sigma_{yy}^{A}(x_4 - x_1) + \Sigma_{yy}^{B}(x_1 - x_2) + \Sigma_{yy}^{C}(x_2 - x_3) + \Sigma_{yy}^{D}(x_3 - x_4)\right]$$
$$- \frac{\Delta t}{2\phi_{i,j}}\left[\Sigma_{xy}^{A}(y_4 - y_1) + \Sigma_{xy}^{B}(y_1 - y_2) + \Sigma_{xy}^{C}(y_2 - y_3) + \Sigma_{xy}^{D}(y_3 - y_4)\right] + \Delta t\beta_{i,j} \quad (2.)$$

$$\phi_{i,j} = \frac{1}{4}\left[(\rho A)_A + (\rho A)_B + (\rho A)_C + (\rho A)_D\right]$$

$$\alpha_{i,j} = \frac{\pi}{2}\left[\left(\frac{\Sigma_{xy}A}{M}\right)_A + \left(\frac{\Sigma_{xy}A}{M}\right)_B + \left(\frac{\Sigma_{xy}A}{M}\right)_C + \left(\frac{\Sigma_{xy}A}{M}\right)_D\right] \quad (3.)$$

$$\beta_{i,j} = \frac{\pi}{2}\left[\left(\frac{\Sigma_{yy} - \Sigma_{\theta\theta}}{M}A\right)_A + \left(\frac{\Sigma_{yy} - \Sigma_{\theta\theta}}{M}A\right)_B + \left(\frac{\Sigma_{yy} - \Sigma_{\theta\theta}}{M}A\right)_C + \left(\frac{\Sigma_{yy} - \Sigma_{\theta\theta}}{M}A\right)_D\right]$$

Prior to solving the velocity equations, the stress deviator components are updated from the previous cycle using the strain rate components. Cercion does not use the HEMP methodology for calculating the strain rate components but rather the method proposed by Margolin[17] is adopted, whereby the flow divergence is calculated in a manner that is consistent with the strain rate. The difference scheme for the strain rate components is given in Equation 4 and Figure 5 identifies the vertices that are used in the calculation of the strain rate components.

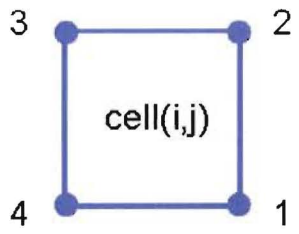

Figure 5. Vertices used to calculate the strain rate for cell (i,j).

The calculation of the strain rate components relies on area and radius terms denoted A_{ijk} and r_{ijk} in Equation 5, where the subscripts i,j,k span the four vertices surrounding the cell in Figure 5. The Margolin approach employs a volume averaged approximation to the spatial velocity derivatives whereas the HEMP approach uses an area averaged approximation. Margolin demonstrates that in certain situations the area averaged method is not a good approximation to the true strain rate and the volume averaged technique is required for physical fidelity.

Besides being more versatile than other methods, the volume averaged technique of Margolin also has the desirable property that the flow divergence can be calculated from the strain rate components exactly as shown in Equation 6, without introducing additional errors into the finite difference equations. The flow divergence is required to calculate the change in internal energy during the Lagrangian phase and it is also used to determine the artificial viscous stress as given in Equation 7. The internal energy update performed in Cercion is the two-step process borrowed from HEMP, where the equation of state is evaluated twice for pressure during the update.

$$\dot{\varepsilon}_{xx} = \frac{1}{6V}\left[(u_3 r_{234} - u_1 r_{412})(y_2 - y_4) + (u_4 r_{341} - u_2 r_{123})(y_3 - y_1)\right]$$

$$\dot{\varepsilon}_{yy} = \frac{1}{6V}\left[(v_1 r_{412} - v_3 r_{234})(x_2 - x_4) + (v_2 r_{123} - v_4 r_{341})(x_3 - x_1) - (v_1 A_{412} + v_2 A_{123} + v_3 A_{234} + v_4 A_{341})\right]$$

$$\dot{\varepsilon}_{\theta\theta} = \frac{1}{2V}\left[v_1 A_{412} + v_2 A_{123} + v_3 A_{234} + v_4 A_{341}\right]$$

$$\dot{\varepsilon}_{xy} = \frac{1}{6V}\left[(v_3 r_{234} - v_1 r_{412})(y_2 - y_4) + (v_4 r_{341} - v_2 r_{123})(y_3 - y_1)\right] \qquad (4.)$$

$$+\frac{1}{6V}\left[(u_1 r_{412} - u_3 r_{234})(x_2 - x_4) + (u_2 r_{123} - u_4 r_{341})(x_3 - x_1) - (u_1 A_{412} + u_2 A_{123} + u_3 A_{234} + u_4 A_{341})\right]$$

$$V = \frac{1}{6}\left[r_{123} A_{123} + r_{234} A_{234} + r_{341} A_{341} + r_{412} A_{412}\right]$$

$$A_{ijk} = \frac{1}{2}\left[(y_k - y_j)(x_i - x_j) - (y_i - y_j)(x_k - x_j)\right]$$

$$r_{ijk} = y_i + y_j + y_k \qquad (5.)$$

$$\nabla \bullet U = \varepsilon_{xx} + \varepsilon_{yy} + \varepsilon_{\theta\theta} \qquad (6.)$$

There are two components in Equation 7 for the artificial viscous stress, one a linear term in flow divergence and the other a quadratic term. This form of artificial viscous stress is scalar in nature in that the artificial viscous stress only appears in two Cauchy stress tensor components, Σ_{xx} and Σ_{yy}, as opposed to

affecting all components. The two component formulation from Equation 7, where a is the sound speed, is a straightforward way to generalize the von Neumann viscous stress intended for one dimensional problems into two dimensions; this linear-quadratic formulation was used in later versions of HEMP as described by Wilkins[15]. All the calculations in this paper have coefficients of C_1=2.0 and C_2=0.1 for the quadratic and linear terms respectively.

$$q = \rho\left[C_1 A(\nabla \bullet U)^2 - C_2\sqrt{A}a(\nabla \bullet U)\right] \tag{7.}$$

Cercion uses the simple finite difference mesh relaxer proposed by Winslow with a nine point stencil. The user can specify in the input deck whether any particular set of vertices are relaxed and the time interval for the relaxation. During the remap phase material fluxes are used to redistribute materials among cells whose vertices have been relaxed. Figure 6 illustrates a cell containing two materials where three of the cell vertices have been relaxed.

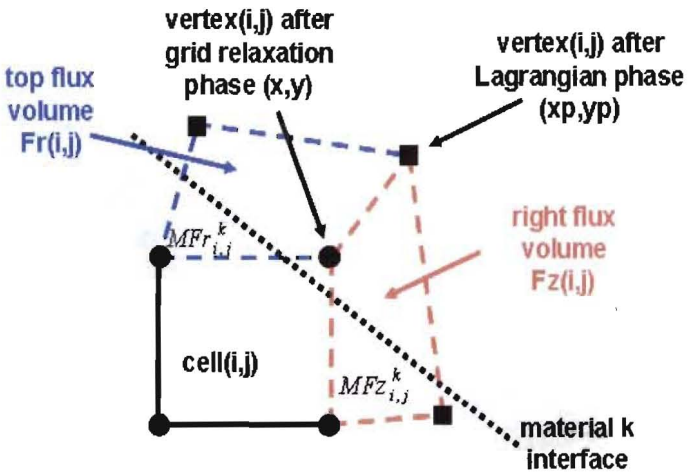

Figure 6. Flux volumes for cell (i,j).

The vertex positions after the Lagrangian phase (xp,yp) are denoted with squares and the relaxed positions (x,y) with circles. The flux volume through the top face, Fr, and the flux volume through the right face, Fz, both of which are positive, are calculated using the vertex positions as shown in Figure 6. These flux volumes are partitioned to represent the individual material fluxes using the appropriate piece-wise linear interface; the mass of material k contained in the flux volume Fz is denoted MFz^k in Figure 6. MFz^k is determined using the donor cell technique as presented in Equation 8.

$$MFz^k_{i,j} = \rho^k_{upwind} Fz^k_{i,j} \tag{8.}$$

The donor cell technique uses the upwind density for material k, such that if the material flows out of cell (i,j) then the density from cell (i,j) is the upwind density otherwise the density from cell(i+1,j) is the upwind density. The update of material k mass from the Lagrangian value, M^k, to the value after the remap step, M^{k*}, is performed as shown in Equation 9. Mass fluxes from four neighboring cells are used to update the material k mass. An analogous equation is solved for the material k internal energy and other cell-centered quantities. This donor cell method was used in SALE for the remap phase; however, unlike Cercion, SALE did not allow for cells with multiple materials.

$$M^{k*}_{i,j} = M^k_{i,j} - \left[MFz^k_{i,j} - MFz^k_{i-1,j} + MFr^k_{i,j} - MFr^k_{i,j-1}\right] \tag{9.}$$

In Cercion, a second order correction is made to the donor cell method for those cells with a single material. The method for calculating the higher order correction is much simpler than the Barth-Jespersen remap method[11] and is displayed in Figure

7 for the material mass flux crossing the top boundary of cell (i,j). The volume of the cell after the Lagrangian phase is colored in red and the volume after mesh relaxation in black. The top flux volume is shown in blue. The geometric centers of the top flux volume and the Lagrangian volume of the cell are highlighted in Figure 7 as triangles.

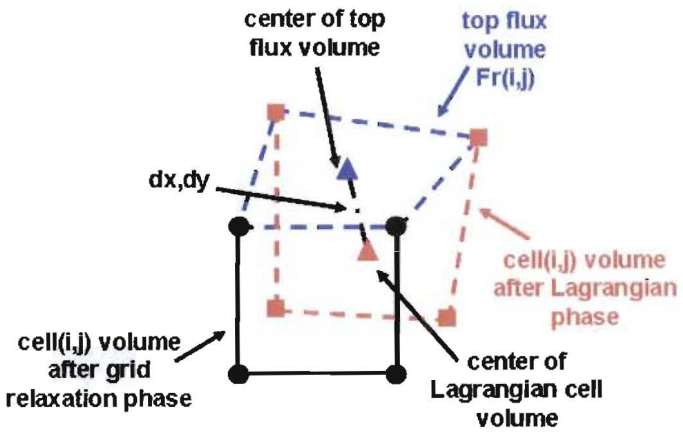

Figure 7. Geometry for the higher order flux calculation.

The distance between the centers (dx,dy) is used together with the upwind density derivatives in a straightforward way to achieve a higher order mass flux given by Equation 10.

Although the remap fluxes themselves are simpler, the density derivatives are calculated using the Barth-Jespersen approach[11] involving a control volume formed from all eight neighboring cells. The Barth-Jespersen method also imposes a limit on the value of each derivative to ensure a monotone flow. The limiting function proposed by Barth and Jespersen is used in Cercion to the same effect.

The mass fluxes calculated during the remap of the cell-centered quantities are retained for use during the momentum remap; this was inspired by the approach of Bowers and Wilson[14]. In addition to the cell-center quantities a higher order remap of the form of Equation 10 is also used for momentum but the momentum control volume is vertex centered, and the momentum flux volumes are entirely different, owing to the spatially staggered nature of the Lagrangian method used in Cercion.

$$MFr_{i,j} = \rho_{upwind} Fr_{i,j} + Fr_{i,j}\left[dx \frac{\partial \rho}{\partial x}_{upwind} + dy \frac{\partial \rho}{\partial y}_{upwind} \right] \qquad (10.)$$

3.0 Test Problems

Having described the fundamental algorithms, in this section Cercion is tested with a series of simple problems in both one and two dimensions. A Riemann shock tube was used to assess the fidelity in capturing shocks and contact discontinuities. The degree of symmetry in the code solution was tested with a Sedov blast wave problem and the ability to calculate elastic-plastic response in an aluminum flyer plate problem was also evaluated.

The Riemann shock tube is common in the CFD literature and since there is an analytic solution it is a good test of the basic conservation properties of any numerical fluid dynamics algorithm[9]. An initial discontinuity between two ideal gas regions (γ=1.4) exists at time zero. The first region has a density of 1 kg/m^3 and a pressure of 10^5 N/m^2. The second region has a density of 0.01 kg/m^3 and a pressure of 10^3 N/m^2. There is a contact discontinuity at the interface between the two regions, which moves as the problem progresses. An Eulerian mesh consisting

of a uniform box 25 m by 1 m having 500 axial zones and 2 radial zones was used for the Cercion simulation. The numerical solution at 0.01s for the density and pressure is given in Figure 8, together with the analytic solution. The location of both the shock at about 9m and the contact discontinuity at about 6m are well captured by the simulation. An expansion fan is also visible in the numerical solution for both pressure and density, agreeing nicely with the analytic solution. Cercion does a reasonable job representing the spatial profiles of velocity and sound speed as shown in Figure 9. The shock is spread out over about 4 zones as indicated in Figure 9 (a). In Figure 9 (b) a slight overshoot in the sound speed appears at the contact discontinuity and the calculated sound speed in front of the contact discontinuity is a few percent less than the analytic value. Overall Cercion does a good job in capturing the important details of the Riemann shock tube.

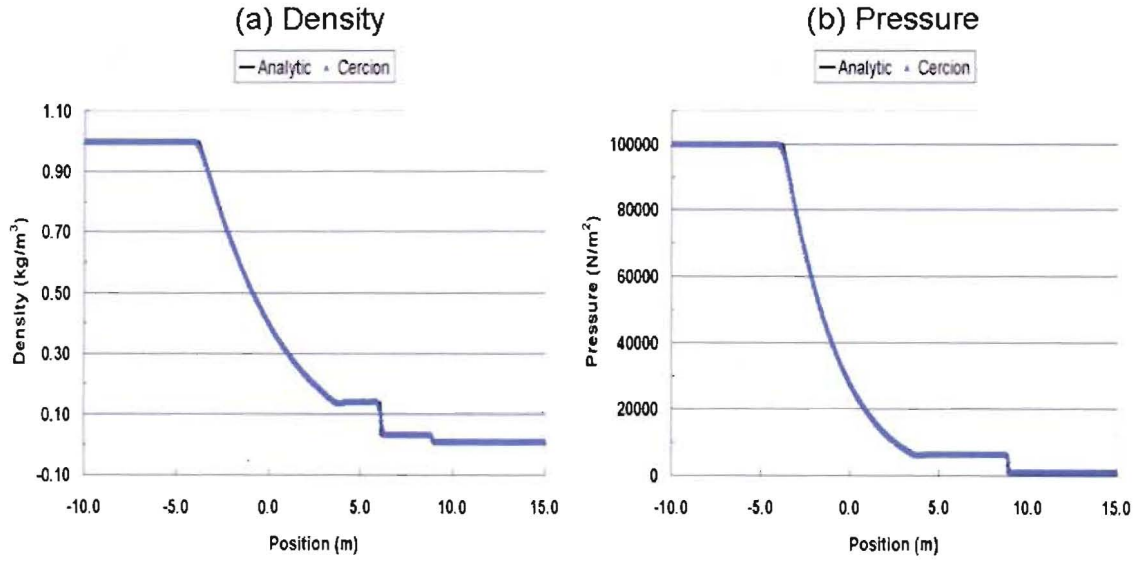

Figure 8. Pressure and density for the Riemann shock tube at t=0.01 s.

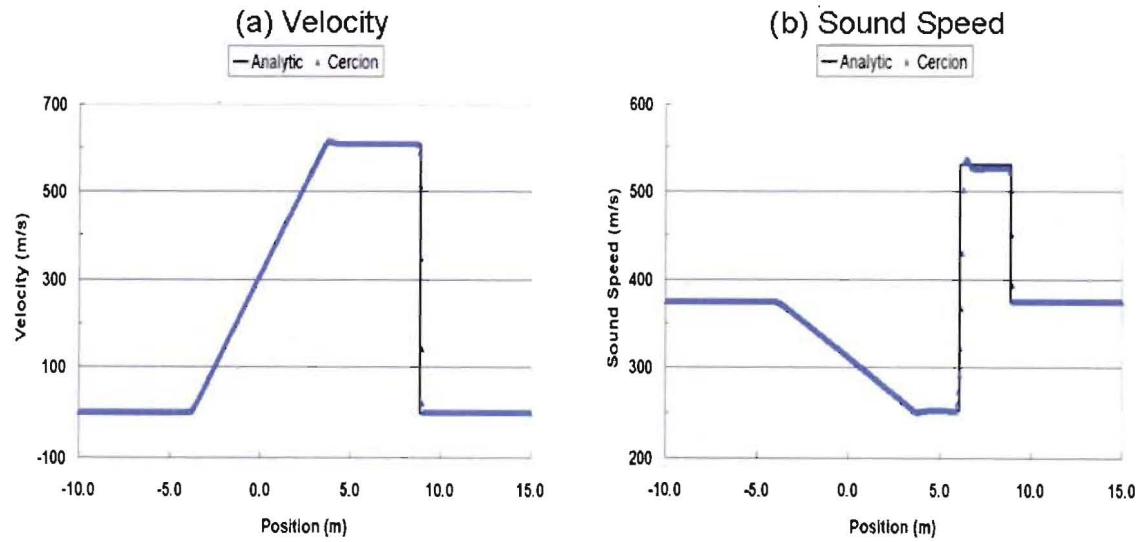

Figure 9. Velocity and sound speed for the Riemann shock tube at t=0.01 s.

The Sedov blast wave problem involves the propagation of a strong disturbance arising from a localized energy source of 85 kJ at the origin at time zero. As time progresses a self-similar solution is obtained that is spherically symmetric[18]. In a two dimensional cylindrical geometry, the Sedov blast wave is a good way to evaluate the symmetry preserving properties of the numerical algorithm. A uniform Eulerian mesh was employed with 300 axial zones and 150 radial zones; the resolution in both directions was 200 μm. The energy source at the origin spanned 4 zones, two radial and two axial.

At 10 μs the solution from Cercion is compared with the corresponding solution from the Eulerian code RAGE in Figure 10. RAGE[19] has an entirely different numerical algorithm than Cercion for solving the fluid equations. Cercion (red) calculates a pressure behind the blast wave of about 2.6 kbar and RAGE (purple) calculates a higher pressure of 3.0 kbar. The self-similar solution (green) gives a pressure of 2.4 kbar which is lower than the result from either code. Although there is disagreement in minimum pressure behind the shock, both codes predict a peak density of about 4.6 g/cc behind the shock which is less than the value of 6.0 g/cc from the self-similar solution. The one dimensional profiles of density and pressure measured from the origin overlap each other at 90 degrees and 45 degrees in the Cercion solution, indicating that symmetry is well preserved. This is generally true for the RAGE solution also and the pressure profiles overlap well; however the peak density behind the shock at 45 degrees is noticeably higher than at 90 degrees. Although both codes smear the shock over several zones, it is clear that the Cercion simulation does a good job predicting the location of the shock at about 2.5 cm from the origin. RAGE predicts a shock position that is slightly ahead of the Cercion value.

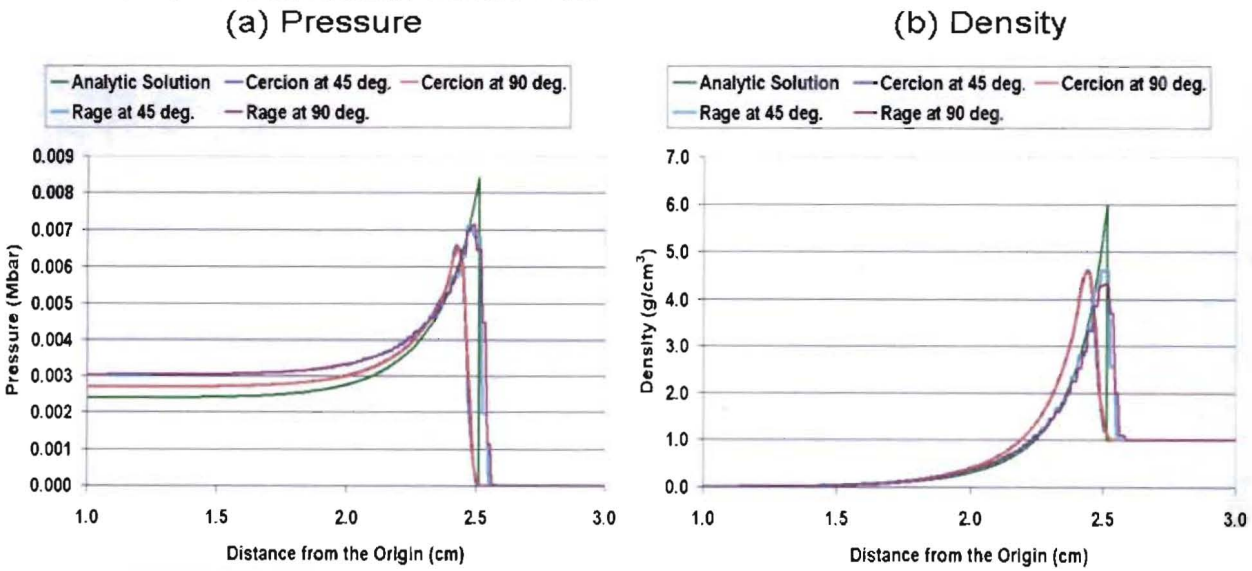

Figure 10. Pressure and density for the Sedov blast wave at 10 μs.

Part of the reason why neither code captures the correct minimum pressure or peak density behind the shock is that self-similar conditions have not been fully achieved in the calculations. Even at 10 μs, strong velocity perturbations still exist owing to the two dimensional nature of the cylindrical energy source at the origin.

This is clearly evident in Figure 11 which shows the one dimensional profile of velocity magnitude as a function of distance from the origin at 10 μs. The solution from both codes departs markedly from the green line representing the self-similar solution as one approaches the origin.

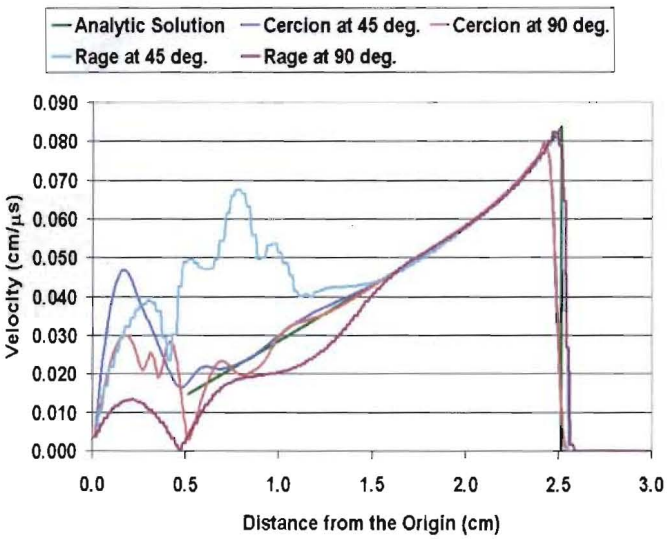

Figure 11. Velocity magnitude for the Sedov blast wave at 10 μs.

A flyer plate test problem was formulated to evaluate the material strength treatment in Cercion. In one dimensional geometry, an aluminum projectile which is 2mm in length (40 cells) and has a velocity of 0.07 cm/μs impacts an aluminum target at rest that is 1 cm in length (200 cells). There is no mesh relaxation in the problem and the fluid equations are solved in a Langrangian fashion. A Mie-Gruneisen EOS is used to represent aluminum with the parameters taken from Steinberg[5]. A simple elastic-plastic material strength model is assumed for aluminum with a constant shear modulus of 270 kbar and a constant yield stress of 0.4 kbar.

The velocity temporal profile at the target-vacuum interface is presented in Figure 12. A companion calculation was performed with another Lagrangian code, FLAG. The FLAG method[20] is distinct from the HEMP technique in that the discrete equations are formulated to preserve total energy exactly; furthermore it has been argued in the literature that the FLAG approach better represents the partition between kinetic and internal energies[16]. Both Cercion and FLAG show an elastic precursor wave in the velocity profile starting just before 1.5 μs and lasting for 0.2 μs. There is agreement in peak velocity as well with both codes giving a value of 0.07cm/μs, which conforms to the theoretical value for this kind of impact. The duration of the velocity pulse at the target-vacuum interface is the same and in general there is good agreement between the two codes for the release wave.

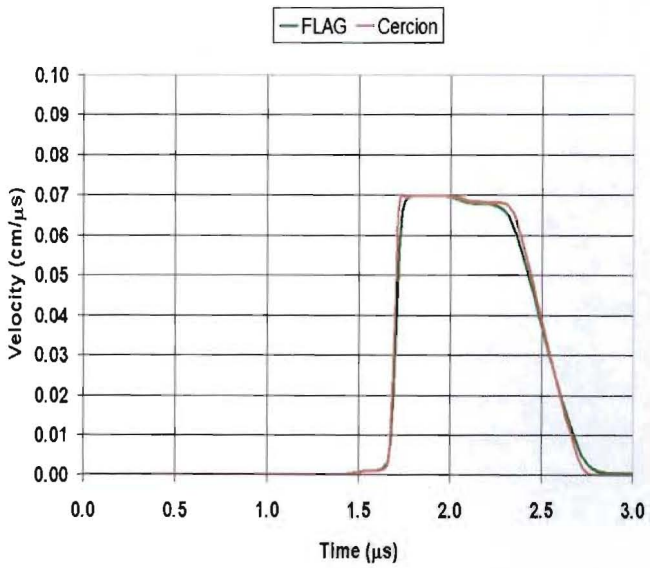

Figure 12. Cercion and FLAG velocity profiles for the flyer plate problem.

The test problems in this section have helped establish confidence in the basic equations for the Lagrangian velocity update, Equations 1 and 2, as well as the calculation of the strain rate given by

Equation 4 and the higher order remap algorithm from Equation 10.

4.0 Steel Shell Implosion

An imploding steel shell is a good test of energy conservation in converging cylindrical geometry. A simple test problem involving two steel shells, separated by a high pressure gas region was constructed in a two dimensional cylindrical geometry as shown in Figure 13. Each of the two steel shells has 10 radial zones with the outer shell being 0.25 cm thick and the inner shell being 0.5 cm thick. There are 180 axial zones in the problem and the cylindrical shells are 5 cm long. A Mie-Gruneisen EOS was used for steel with parameters for stainless steel 304 taken from Steinberg[6]. Cercion calculations were performed both with and without material strength. A simple material strength model was adopted with constant yield strength of 0.05 Mbar and constant shear modulus of 0.895 Mbar.

The high pressure gas region in Figure 13 contains 80 radial zones. Within the inner steel shell there is a low pressure gas region, consisting of 20 radial zones.

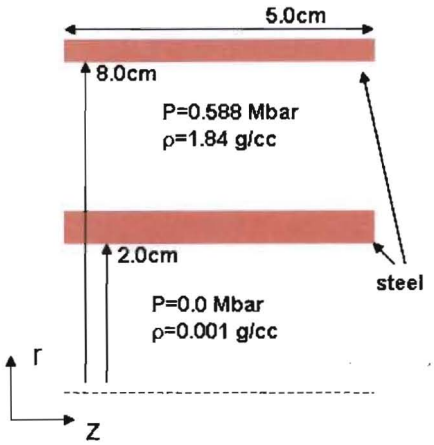

Figure 13. Initial geometry of the steel shell cylindrical implosion problem.

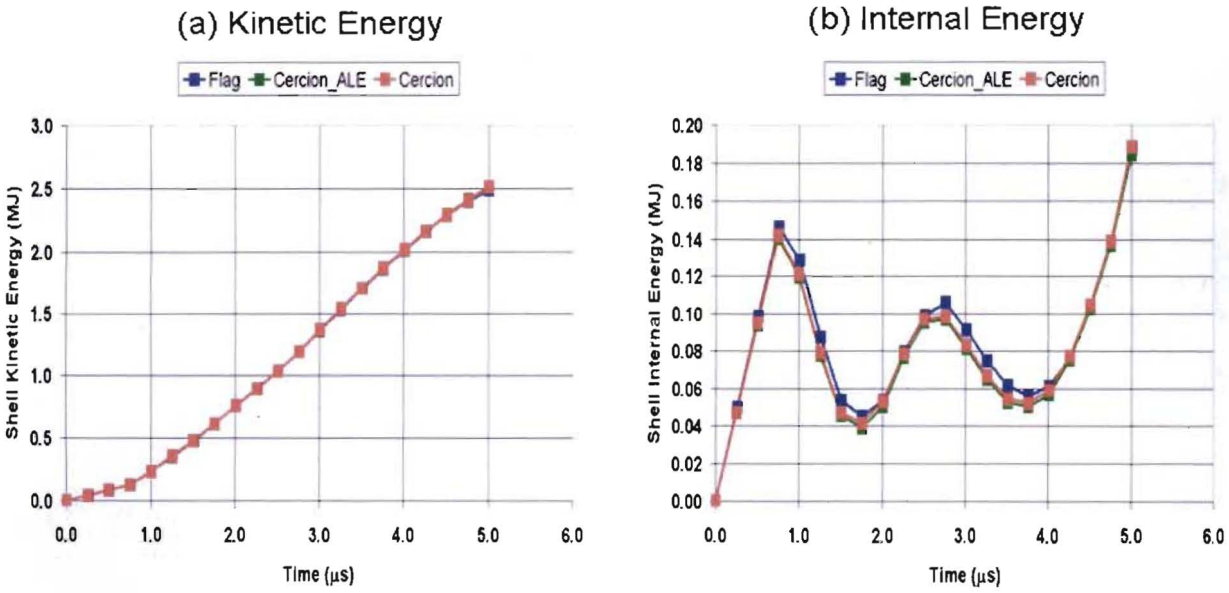

Figure 14. Inner steel shell kinetic and internal energies without material strength.

Both gas regions were represented as ideal gases in the calculations and initialized according to the conditions specified in Figure 13. The simulations lasted 5 μs and comparisons were made with FLAG calculations having the same mesh, using a purely Lagrangian treatment of the hydrodynamics.

For the case without material strength FLAG and Cercion give the same kinetic energy of about 2.5 MJ for the inner steel shell at 5 μs as shown in Figure 14 (a). In general the calculated values of kinetic energy are the same for the two codes during the implosion interval; maximum compression of the shell occurs shortly after 5 μs. The Cercion calculation is presented as the red curve in Figure 14 with the FLAG result as the blue curve. An additional Cercion simulation was performed using ALE, such that mesh relaxation occurred everywhere at every cycle except inside the low pressure gas region and at material boundaries. The Cercion result with ALE is shown as the green curve in Figure 14. It is evident that the use of ALE during the implosion has virtually no effect on the Cercion solution, both for kinetic energy and internal energy. Although the agreement for the two codes is excellent for kinetic energy and generally good for internal energy there are noticeable differences between Cercion and FLAG. The FLAG internal energy is consistently higher than Cercion between 1 and 4 μs. The internal energy difference between FLAG and Cercion is approximately 10% at 3 μs. Both codes give nearly the same total energy for the inner shell as a function of time, with a maximum total energy of about 2.7 MJ at 5 μs.

FLAG is somewhat unique in its use of the temporary triangular subzonal (TTS) method to dissipate unphysical hourglass instabilities[21]. The TTS method is an edge-based technique whereas the prevalent approach in the literature for controlling hourglass instabilities is a volume-based method. In certain situations, the TTS method can lead to relatively large amounts of mesh stiffening. The Cercion calculations were performed without any anti-hourglass dissipation and perhaps the internal energy difference can be explained by the use of the TTS method in FLAG.

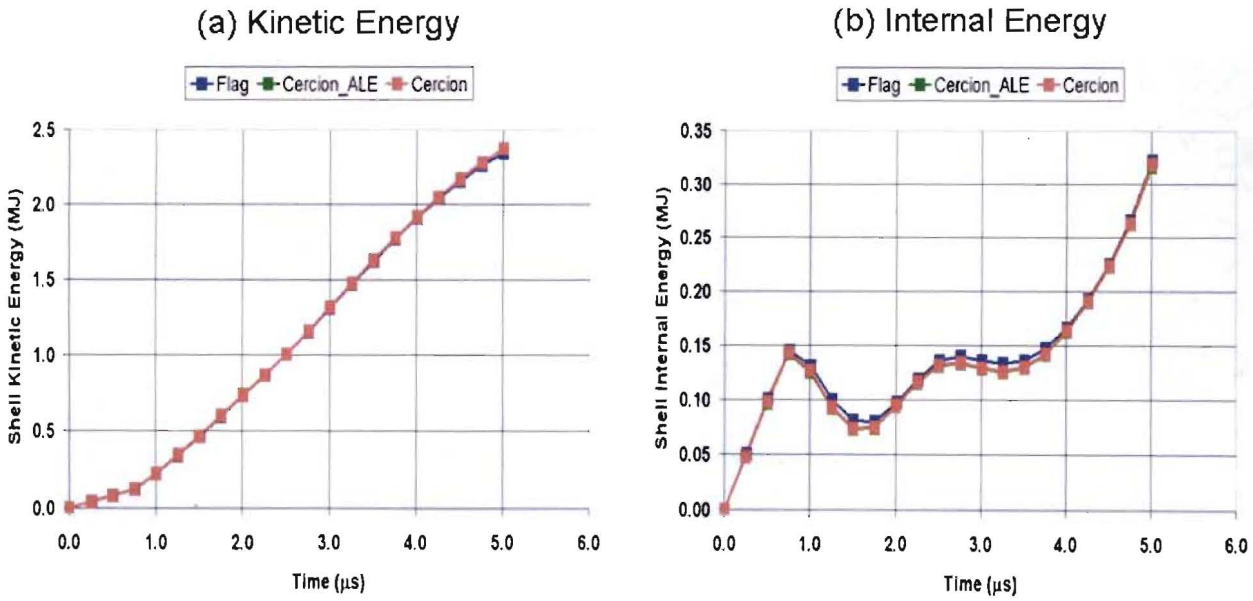

Figure 15. Inner steel shell kinetic and internal energies with material strength.

When material strength is included in the calculations it is also true that FLAG and Cercion agree well for the kinetic energy of the inner shell as shown in Figure 15 (a). The peak kinetic energy in this case is about 2.4 MJ. As before, the purely Lagrange Cercion calculation in red and the Cercion calculation with ALE in green are indistinguishable. The total energy of the inner shell as a function of time remains unchanged from the previous set of calculations without strength. There is excellent agreement to within 3% between Cercion and FLAG for total energy of the shell. The addition of material strength only changes the partition of energy between internal and kinetic modes in the calculations. With regards to internal energy, both codes show the same general temporal behavior during the implosion; however FLAG gives more internal energy than Cercion during most of the implosion time interval, as illustrated in Figure 15 (b). The internal energy of the inner shell near the end of the implosion at 5 μs is about 0.32 MJ and this is nearly the same for both codes. At 3 μs FLAG gives an internal energy that is about 5% higher than Cercion. For the case without strength, the internal energy at 5 μs is lower at about 0.19 MJ and both codes give a similar value.

5.0 Conclusions

Cercion uses proven numerical methods for the solution of the Langrangian equations of motion and the calculation of material strength properties. Cell-centered data structures simplify the programming of the code and allow for efficient memory allocation for multi-material problems. A second-order accurate remap method is used for density, energy and momentum, enabling ALE calculations to be performed. Results from this paper indicate that the remap method is robust and has been implemented correctly in Cercion.

Cercion shows excellent agreement with the analytic solution for the Riemann shock tube problem. There is good symmetry in the Cercion solution for the Sedov blast wave problem but both Cercion and RAGE have a peak density behind the shock that is lower than the self-similar solution. In addition to the inherent smearing of the shock due to the discrete solution method, the fact that the calculations have not fully reached a self-similar state is also a likely contributor to the discrepancy. In pure Langrangian mode, both Cercion and FLAG give similar velocity profiles at the target-vacuum interface for the flyer plate test problem. This establishes confidence in the numerical implementation of the Margolin strain rate method.

For the steel shell cylindrical implosion problem Cercion and FLAG calculate total energies for the inner steel shell that agree to within 3% during the implosion time interval. For cases with and without material strength, FLAG gives a higher internal energy than Cercion, but the kinetic energy of the inner shell is in agreement between the two codes. The Cercion results are insensitive to the use of ALE during the calculation.

Perhaps the internal energy difference between Cercion and FLAG is due to the TTS anti-hourglass treatment in FLAG. It would be informative to perform the cylindrical implosion calculation in FLAG without the TTS model, if possible, in order to test this hypothesis in future work. Besides the HEMP based Lagrange method in Cercion and the total energy preserving method of FLAG there are a number of distinct Lagrangian hydrodynamic algorithms in the literature. A comparison with some of these other

numerical methods would help illustrate how different techniques partition energy between internal and kinetic modes.

References

1. Benson, D.J., "Computational Methods in Lagrangian and Eulerian Hydrocodes," Computer Methods in Applied Mechanics and Engineering, 99, (1992).

2. Amsden, A.A., Ruppel, H.M., and Hirt, C.W., "SALE: A Simplified ALE Computer Program for Fluid Flow at All Speeds," Los Alamos National Laboratory, Los Alamos, NM, LA-8095 (1980).

3. Wilkins, M., "Calculation of Elastic-Plastic Flow," Lawrence Livermore National Laboratory, Livermore, CA, UCRL-7322 (1963).

4. Preston, D.L., Tonks, D.L., and Wallace, D.C., "Model of Plastic Deformation for Extreme Loading Conditions," Journal of Applied Physics, 93, (2003).

5. Steinberg, D.J., "Equation of State and Strength Properties of Selected Materials," Lawrence Livermore National Laboratory, Livermore, CA, UCRL-MA-106439 (1996).

6. Demuth, R.B., Margolin, L.G., Nichols, B.D., Adams, T.F., and Smith, B.W., "SHALE: A Computer Program for Solid Dynamics," Los Alamos National Laboratory, Los Alamos, NM, LA-10236 (1985).

7. Brackbill, J.U. and Saltzman, J.S., "Adaptive Zoning for Singular Problems in Two Dimensions," Journal of Computational Physics, 46, (1982).

8. Winslow, A.M., "Equipotential Zoning of Two-Dimensional Meshes," Lawrence Livermore National Laboratory, Livermore, CA, UCRL-7312 (1963).

9. Laney, C., Computational Gasdynamics, (Cambridge University Press, New York, 1998).

10. Margolin, L.G., and Dhashkov, M., "Second-Order-Sign-Preserving Conservative Interpolation (Remapping) on General Grids," Journal of Computational Physics, 184, (2003).

11. Barth, T.J., "Numerical Methods for Gasdynamics Systems on Unstructured Grids," Proceedings of the International School on Theory and Numerics for Conservation Laws, Freiburg/Littenweiler, Germany, October 20-24, 1997 (1999).

12. Youngs, D.L., "An Interface Tracking Method for a 3D Eulerian Hydrodynamics Code," Atomic Weapons Establishment, Aldermaston, UK, Technical Report 44/92/35 (1984).

13. Morgan N., "A New Liquid-Vapor Phase Transition Technique for the Level Set Method," Georgia Institute of Technology, PhD Dissertation in Mechanical Engineering (2005).

14. Bowers, R. and Wilson, J., Numerical Modeling in Applied Physics and Astrophysics, (Jones and Bartlett Publishers, Boston, 1991).

15. Wilkins, M., "Use of Artificial Viscosity in Multidimensional Fluid Dynamic Calculations," Journal of Computational Physics, 36, 281-303 (1980).

16. Christiansen, R.B., "Godunov Methods on a Staggered Mesh: An Improved Artificial Viscosity," Lawrence Livermore National Laboratory, Livermore, CA, UCRL-JC105269 (1991).

17. Margolin, L.G., and Adams, T.F., "Spatial Differencing for Finite Difference Codes," Los Alamos National Laboratory, Los Alamos, NM, LA-10249 (1985).

18. Barenblatt, G.I., Scaling, Self-Similarity, and Intermediate Asymptotics, (Cambridge University Press, New York, 1996).

19. Clover, M., "The New Hydrodynamic Method in RAGE," Los Alamos National Laboratory, Los Alamos, NM, LA-UR-02-3802 (2002).

20. Caramana, E.J., Burton, D.E., Shashkov, M.J., and Whalen, P.P., "The Construction of Compatible Hydrodynamics Algorithms Utilizing Conservation of Total Energy," Journal of Computational Physics, 146, 227-262 (1998).

21. Caramana, E.J., and Shashkov, M.J., "Elimination of Artificial Grid Distortion and Hourglass-Type Motions by Means of Lagrangian Subzonal Masses and Pressures," Journal of Computational Physics, 142, 521-561 (1998).