A Compatible Lagrangian Hydrodynamics Algorithm for Unstructured Grids

J.C. Campbell[†] and M.J. Shashkov[‡]

[†] *T-7 and T-CNLS, Los Alamos National Laboratory, Los Alamos, New Mexico, 87545*
[‡] *T-7, Los Alamos National Laboratory, Los Alamos, New Mexico, 87545*

E-mail: jamesc@lanl.gov

LA-UR-00-3231

We have extended a compatible Lagrangian hydrodynamics algorithm to unstructured grids, where each zone is a polygon with an arbitrary number of sides. An unstructured grid offers more flexibility than a logical grid in the construction of the initial grid, especially for domains with complex geometry. The compatible hydrodynamics algorithm is designed to conserve momentum and total energy exactly in discrete form. It achieves this by deriving the discrete energy equation from the discrete momentum equation using the conservation of total energy. The compatible hydrodynamics algorithm includes subzonal pressures, which are used to control spurious grid motions, and an edge-centered artificial viscosity. The edge-centered viscosity uses a limiter function to ensure that the viscosity switches off under uniform compression and rotation. On a logical grid this limiter function uses the local coordinate system to determine neighbor edges. On an unstructured grid a local coordinate system at a node can not be used, requiring instead alternative methods for selecting neighbor edges. Computational results show that the algorithm functions well on an unstructured grid.

---

1. INTRODUCTION

The compatible Lagrangian hydrodynamics algorithm described in [3–5] is a staggered grid, finite-difference, algorithm for solving the equations of fluid dynamics in Lagrangian form. The algorithm is designed so that it exactly conserves momentum and internal energy in the discrete case. The algorithm is written in terms of corner forces, a corner force being the contribution of one computational zone to the total force at a point. The use of corner forces allows the algorithm to be generalized to include any force that can be written in this manner, specifically subzonal pressure forces [4] and artificial viscosity forces [5].

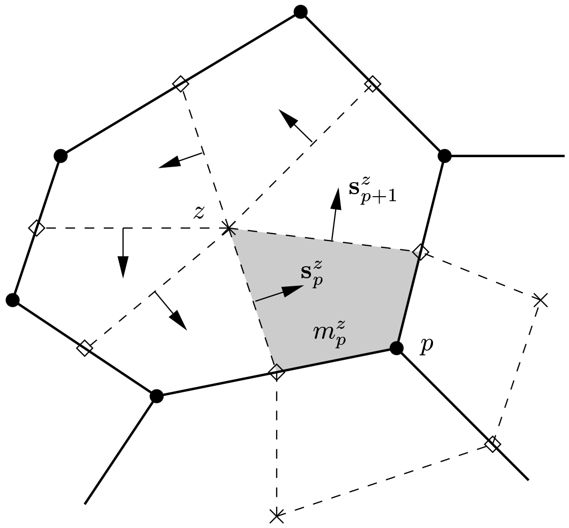


FIG. 1. Fragment of an unstructured staggered grid, including zone z and point p. The solid lines define the grid, and the dashed lines show the median mesh. The median mesh is formed by connecting the zone centers, $\times$, to the mid-side points, $\diamond$. The $\mathbf{s}$ vectors are normal to the median mesh. The shaded area shows a subzonal Lagrangian volume.

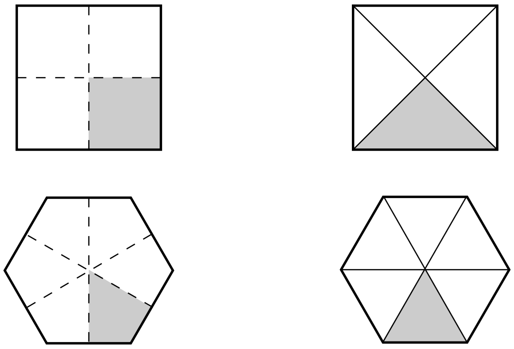
This paper describes the implementation of a compatible hydrodynamics algorithm, including subzonal pressures and artificial viscosity, on an unstructured grid. We assume that the unstructured grid is made from arbitrary polygons, so we can make no assumption as to the maximum number of sides a zone may have or the number of zones that meet at a node. An unstructured grid offers more flexibility than a logical grid in the construction of the initial grid, especially for domains with complex geometries. Using an unstructured grid also allows one to change the grid connectivity and refine the grid during calculations if required, but this is not considered in this paper.

In section 2, we describe the compatible hydrodynamics algorithm for unstructured grids. In section 3 we describe a good data structure for implementing an unstructured grid in a real code. Then in section 4 we show numerical results for several problems. Finally the conclusions are given, in section 5.

2. A COMPATIBLE HYDRODYNAMICS ALGORITHM

A compatible Lagrangian hydrodynamics algorithm for 2D is described in a set of papers by Caramana et al. [3–5]. This section will present the algorithm in a form suitable for unstructured grids. As the derivation of the algorithm is described in depth in the papers be Caramana et al., we present only the essentials of the method. The algorithm is based on the discretizations of the continuum equations [3], the subzonal pressure method for control of hourglass type motions [4], and an artificial viscosity calculated on zone edges [5].

The algorithm uses a spatially staggered grid. The position and velocity are defined at the grid points, while the density, pressure and internal energy are defined in the zone centers. Fig. 1 shows a section of a staggered grid; the solid lines denote the grid and the dashed lines the median mesh. The median mesh is constructed



a b

FIG. 2. Subzonal volumes are the same regardless of the number of sides of the zone. (a) Corner volumes remain quadrilaterals. (b) Side volumes remain triangles.

by connecting the zone centers with the mid-side points. A core assumption for a Lagrangian method is that each zone of the grid represents a discrete volume element that may deform, but may not gain or loose mass. If the coordinate of the zone center is calculated in a Lagrangian manner then the same assumption can be made about the median mesh. This leads to the definition of a subzonal corner volume, shown in Fig. 1 as the shaded area, which is also a Lagrangian volume. We denote the mass of this corner volume as m_z^p where the indices denote the zone and point with which it is associated. We define $m_z^p = m_p^z$, but follow the convention of always summing with respect to the lower index. The corner mass can then be used to define both the zone mass, M_z, and point mass, M_p.

$$M_z = \sum_{p \in S(z)} m_p^z \;\; , \;\; M_p = \sum_{z \in S(p)} m_z^p \; , \tag{1}$$

where the sums are over the stencil associated with the zone or point. The stencil for a zone is each point that is a vertex of the zone, so the sum includes every subzone that is associated with the zone. For a point the stencil is every adjacent zone, so the sum includes every subzone that is associated with the point. As the mass of a volume does not change, the density is easily found.

A useful property of subzonal volumes on an unstructured mesh is that they have a constant number of sides regardless of the number of sides of the zone, see Fig. 2. Thus writing the equations using the subzonal volumes means that the form of the equations is the same, regardless of the grid used.

The governing equations in a hydrodynamics code are the conservation equations. In the Lagrangian framework the equations of conservation of mass (2), momentum (3) and energy (4) are written

$$\frac{1}{\rho}\frac{d\rho}{dt} = -\nabla \cdot \mathbf{v} \; , \tag{2}$$

$$\rho \frac{d\mathbf{v}}{dt} = -\nabla P \; , \tag{3}$$

$$\rho \frac{de}{dt} = -P \; \nabla \cdot \mathbf{v} \; , \tag{4}$$

where ρ is density, P is pressure, e is specific internal energy and $\mathbf{v}$ is the velocity vector.

The discrete momentum equation at point p is written

$$M_p \frac{d\mathbf{v}_p}{dt} = \mathbf{F}_p = \sum_{z \in S(p)} \mathbf{f}_z^p . \tag{5}$$

This equation introduces the corner force $\mathbf{f}_z^p$. This is the force from zone z that acts on point p, so summing the contributions from all zones adjacent to p gives the total force at that point, $\mathbf{F}_p$. The corner pressure force is

$$\mathbf{f}_z^p = P_z(\mathbf{s}_{p+1}^z - \mathbf{s}_p^z), \tag{6}$$

where the area vectors $\mathbf{s}_{p+1}^z$ and $\mathbf{s}_p^z$ are defined in Fig. 1.

The change in internal energy of a zone due to the corner forces can be found by using the conservation of total energy. The discrete total energy of the system can be written as the sum of the internal energy of each zone and the kinetic energy of each point

$$\sum_z E_z = \sum_z M_z e_z + \sum_p \frac{1}{2} M_p {\mathbf{v}_p}^2 \; . \tag{7}$$

These sums are over all zones and points. e_z is the specific internal energy, and E_z the total energy, of a zone. Differentiating with respect to time gives an expression for the change in total energy, which is equal to the external work done on the system. Assuming no external work has been done gives

$$\begin{aligned} \frac{d}{dt}\left(\sum_z E_z\right) &= \sum_z M_z \frac{de_z}{dt} + \sum_p M_p \mathbf{v}_p \frac{d\mathbf{v}_p}{dt} \\ &= \sum_z \left(M_z \frac{de_z}{dt} + \sum_{p \in S(z)} \mathbf{f}_p^z \mathbf{v}_p \right) \\ &= 0. \end{aligned} \tag{8}$$

If the sum over zones is set to zero for each zone this gives an expression for the change in internal energy for a zone in terms of the corner forces from that zone

$$M_z \frac{de_z}{dt} = - \sum_{p \in S(z)} \mathbf{f}_p^z \mathbf{v}_p \; . \tag{9}$$

The corner forces in equations (5) and (9) are arbitrary and can result from any source.

The set of discrete operators used in the compatible hydrodynamics algorithm obey the same relationships that occur in the continuous equations. Using the conservation of total energy to derive an expression for the internal energy ensures that total energy is conserved to numerical roundoff in calculations.

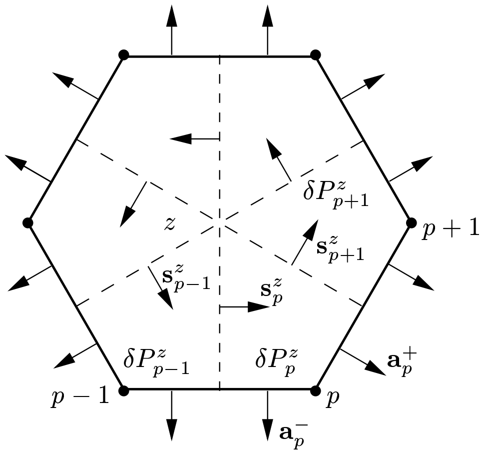


FIG. 3. A zone, dashed lines denote the median mesh. Vectors **a** are outward half-edge normals to the zone.

The subzonal pressure method [4] was developed to control artificial grid distortions, such as the hourglass modes. As the corner volumes are Lagrangian they have a fixed mass; from this mass and the volume a subzonal density can be calculated. In the case of a logically rectangular grid, a quadrilateral zone has eight degrees of freedom, two translations, two extensions, two shear and two hourglass. All but the hourglass modes are physical, but only for the hourglass modes does the density of a corner volume differ from the zone to which it belongs. The subzonal pressure method uses this to calculate corner forces that are proportional to the difference between the subzonal and zonal densities, and oppose the hourglass motion.

For each sub-zone the pressure difference between the zone and subzone, δP^z_p, is calculated using the equation of state

$$\delta P^z_p = (\rho^z_p - \rho_z)\frac{c_z^2}{\gamma}, \tag{10}$$

where c_z is the speed of sound in the zone, and γ is a parameter of the equation of state. This expression assumes that the specific internal energy of a subzone is equal to the specific internal energy of the zone. The corner force due to this pressure difference is then calculated by integrating around the boundary of the subzonal volume. The corner force at point p is

$$\mathbf{f}^z_p = \delta P^z_p(\mathbf{a}^+_p + \mathbf{a}^-_p) + \frac{1}{2}\left[(\delta P^z_p - \delta P^z_{p+1})\,\mathbf{s}^z_{p+1} + (\delta P^z_{p-1} - \delta P^z_p)\,\mathbf{s}^z_p\right] \tag{11}$$

where the **a** vectors are outward half-edge normals to the zone; see Fig. 3. In practice this corner force is then multiplied by a dimensionless merit factor which is user defined at the start of the calculation. Typically the value of the merit factor is between 0.0 and 1.0.

The final element of the algorithm is the artificial viscosity [5]. This is an edge-centered viscosity, meaning that the viscosity is calculated at zone edges. The viscosity is designed to satisfy the following set of conditions: that it only acts to decrease kinetic energy, that it vanishes smoothly as the velocity field vanishes and

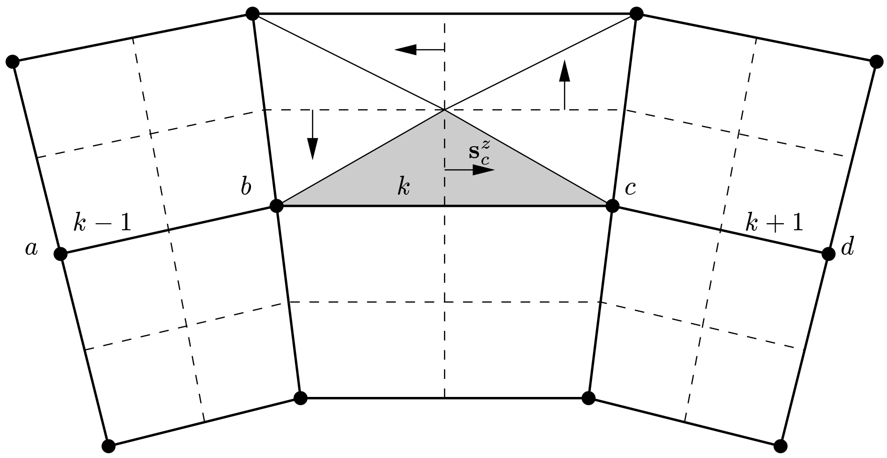


FIG. 4. Section of a logical grid. The solid interior lines delineate triangular subzones, and dashed lines the median mesh.

as compression stops, that it be zero in expansion, that it has no effect along a wave front of constant phase and vanishes for uniform contraction and rigid rotation. To effect the final two conditions the viscosity includes a limiter function, which is a multidimensional generalization of a TVD advection limiter.

For simplicity we discuss the main concept of the artificial viscosity on a logically rectangular grid. Fig. 4 shows a section of a logical grid, edge k connects the two points b and c. The viscous force between the two points due to zone z is $\mathbf{f}_k$.

$$\mathbf{f}_k = \begin{cases} -(1-\psi_k)\, q_{Kur,k}(\widehat{\Delta\mathbf{v}}_k \cdot \mathbf{s}_c^z)\widehat{\Delta\mathbf{v}}_k & \text{if } (\Delta\mathbf{v}_k \cdot \mathbf{s}_c^z) < 0 \\ 0 & \text{if } (\Delta\mathbf{v}_k \cdot \mathbf{s}_c^z) \geq 0\ . \end{cases} \tag{12}$$

$\mathbf{f}_k$ contributes to the corner forces at points b and c, it is subtracted from $\mathbf{f}_c^z$ and added to $\mathbf{f}_b^z$. The velocity difference for the edge is $\Delta\mathbf{v}_k = \mathbf{v}_b - \mathbf{v}_c$, and $\widehat{\Delta\mathbf{v}}_k$ is the unit vector in the direction of this velocity difference. $q_{Kur,k}$ is the Kurapatenko form of the basic scalar artificial viscosity

$$q_{Kur,k} = \rho_k \left\{ c_2 \frac{(\gamma-1)}{4}|\Delta\mathbf{v}_k| + \sqrt{c_2^2\left(\frac{\gamma-1}{4}\right)^2(\Delta\mathbf{v}_k)^2 + c_1^2 c_{s,k}^2} \right\} |\Delta\mathbf{v}_k| \tag{13}$$

c_1 and c_2 are dimensionless constants that multiply the linear and nonlinear viscosity terms, generally set to unity. The edge density, ρ_k, and sound speed, $c_{s,k}$ are

$$\rho_k = \frac{2\rho_b\rho_c}{\rho_b+\rho_c}\ \ ,\ \ c_{s,k} = \min(c_{s,b}, c_{s,c}) \tag{14}$$

where the density and sound speed at point b and c are area-weighted averages of the values for the surrounding zones.

The limiter function, ψ_k, is

$$\psi_k = \max[0, \min(0.5(r_{l,k}+r_{r,k})\,,\, 2r_{l,k}\,,\, 2r_{r,k}\,,\, 1)], \tag{15}$$

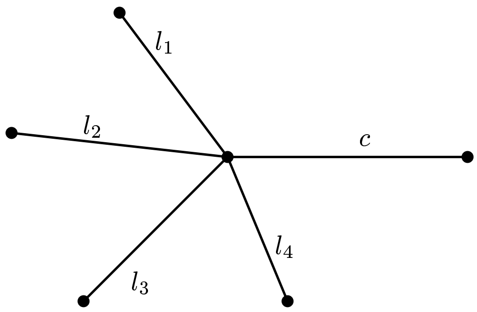


FIG. 5. On an unstructured mesh there is no clear choice of neighbor edge for the viscosity limiter function.

where

$$r_{r,k} = \frac{\Delta \mathbf{v}_{k+1} \cdot \widehat{\Delta \mathbf{v}}_k}{\Delta \mathbf{x}_{k+1} \cdot \widehat{\Delta \mathbf{x}}_k} \Big/ \frac{|\Delta \mathbf{v}_k|}{|\Delta \mathbf{x}_k|} \quad , \quad r_{l,k} = \frac{\Delta \mathbf{v}_{k-1} \cdot \widehat{\Delta \mathbf{v}}_k}{\Delta \mathbf{x}_{k-1} \cdot \widehat{\Delta \mathbf{x}}_k} \Big/ \frac{|\Delta \mathbf{v}_k|}{|\Delta \mathbf{x}_k|} \; . \tag{16}$$

r_l and r_r are the left and right velocity derivative ratios, on a logical grid the definition of left and right edges is clear from the logical indexing. For uniform compression or rigid rotation the ratios are both 1.0, and so $\psi_k = 1.0$ and the viscosity is switched off. Near a shock ψ_k is small or zero. At a boundary, where the left or right edge does not exist, the ratio for that edge is set to 1.0.

On an unstructured mesh the choice of left and right edges for the limiter function is not clearly defined. Fig. 5 shows an example where there are four possible left edges, l_1 to l_4, for the center edge, c. Logical indexing can not be used to chose an edge. For this case Caramana et al. [5] suggest calculating the r_l ratio for each of the possible left edges, and taking the maximum value, ignoring any edge with an angle of less than 90° to the center edge. For the example shown in Fig. 5 r_l is calculated for edges l_1, l_2 and l_3, edge l_4 is ignored because of its angle with the center edge.

A second option for calculating the ratio r_l is to find the edge that forms the largest angle with the center edge and use that as the left edge. So in the example in Fig. 5, the l_2 edge would be used as the left edge. With this approach there is the further choice of fixing the left and right neighbors at the start of the calculation, which is closest to the logical mesh implementation, or of choosing a new neighbor every time step. The disadvantage of fixing the neighbors at the start is that on a unstructured mesh it is possible to have two edges with the same angle to the center edge, in this case the actual choice of neighbor edge is determined by how the search is implemented in the code. This is not uncommon in grids containing points with only three neighbor zones. Fixing the neighbor edges for this type of grid can cause asymmetric results in symmetric problems. These asymmetries do not occur when a new neighbor is chosen every time step.

For the calculations shown in section 4 we use the maximum angle choice, unless stated otherwise.

3. DATA STRUCTURES

The coding for unstructured meshes can be made simple and efficient through the use of well designed connectivity structures. For the implementation of the compatible hydrodynamics algorithm we have used a data structure designed by Burton [2]. He suggests the use of a number of different objects for which data may be stored. Of the objects that are suggested there are four that are useful for implementing the compatible hydrodynamics algorithm:

Point. At a point information such as position and velocity are known.

Zone. In a zone information such as volume and pressure are known.

Corner. The corner is related to a single zone and point. It is a quadrilateral formed by a point, zone center and the center of the two adjacent edges. Information such as corner volume and corner force are known for a corner.

Edge. An edge connects two adjacent nodes. Information such as the coordinate of the point at the center of the edge and value of the limiter function is known.

Using these objects it is convenient to hold the connectivity information for the corners. To implement the momentum and energy equations is necessary to know the point and zone associated with each corner, and the two adjacent edges. The edges are required as the coordinate of the center of the edge is needed to calculate the $\mathbf{s}$ vectors. The two nodes associated with each edge must also be known.

To implement the subzonal pressure method requires additional connectivity information: each corner needs to know the two neighbour corners that share the same zone, these neighbour corners are associated with the $p+1$ and $p-1$ points in equation (11). The $\mathbf{a}$ vectors are calculated from the point and edge coordinates.

The edge viscosity requires the neighbour corners that are not associated with the same zone, and for an edge the four adjacent corner volumes. The extra information for a corner is used to identify all edges that meet at a point, and to calculate the nodal values for density and sound speed used by the viscosity. The neighbor corner volumes for an edge are required to both calculate and distribute the viscosity force.

It is possible to implement the algorithm using only zones and points but this becomes inefficient when zones in a single grid have a varying number of sides.

4. RESULTS

In this section we show a series of numerical examples to illustrate the behavior of the compatible hydrodynamics algorithm on an unstructured grid. We only present test problems in two-dimensional plane geometry. The test code uses a fixed connectivity Lagrangian grid. The subzonal pressure force merit factor used for each problem is given. Unless stated otherwise the maximum angle choice for limiter neighbors is used.

4.1. Piston problem.

This problem models the propagation of a one-dimensional shock wave driven by a piston. A box of width 1.0 and height 0.1 is filled with a perfect gas with $\gamma = 5/3$. A piston moves from left to right with a velocity of 1.0. All other boundaries are fixed. This is the same as the Saltzman piston problem [7], but the Saltzman problem also uses a specific logically rectangular grid.

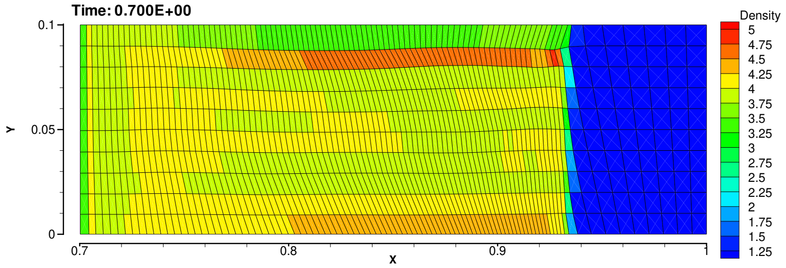


FIG. 6. Piston results for Saltzman grid. These results are the same as results from a logical grid code. Merit factor 0.5.

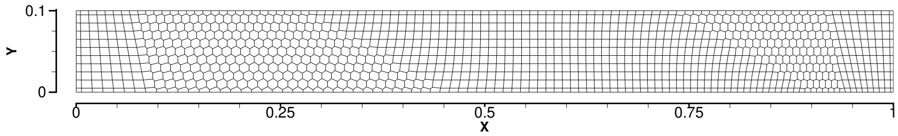


FIG. 7. Unstructured grid for piston problem.

The compatible hydrodynamics algorithm can run the Saltzman piston problem without large mesh distortion [4] on the Saltzman grid. To test the unstructured code, the piston problem was first run on the Saltzman mesh, using the unstructured mesh code to solve a structured mesh problem. The results from structured and unstructured codes should be the same, except for minor differences due to implementation such as the limiter function. Fig. 6 shows the unstructured code results at time 0.7, when the exact solution gives the shock wave x-coordinate as 0.933, and the density behind the shock wave as 4.0. The results show no major mesh distortion behind the shock.

To generate an unstructured grid for this problem we took the nodal positions of the Saltzman grid, and generated a Voronoi tessellation [1] using these points. The resulting grid was improved by removing very short zone edges. The resulting grid is shown in Fig. 7. This grid contains quadrilaterals, pentagons and hexagons. There are three regions of quadrilateral zones and two of hexagons, the pentagonal zones forming the interfaces between these regions. The quadrilateral zone regions result from the removal of very short edges from the original Voronoi tessellation, which produces few zones that are not hexagons for this set of points.

Fig. 8 shows results for the unstructured grid at time 0.7. These results show no grid distortion, and the shock wave remains planar. The results do show that the density is high (≈ 5.0) where the grid changes from quadrilateral to hexagonal, and low (≈ 3.5) where the grid changes back.

At time 0.7 there is minimal difference between the results using the two different methods for finding the limiter ratios. However a clear difference between the two methods can be seen when the piston problem is run to time 0.925. By this time the shock wave has reflected off both the fixed and moving ends of the tube. The exact solution has a shock moving from left to right with an x coordinate of 0.95; the density in front of the shock is 10.0, and behind the shock is 20.0. Fig. 9

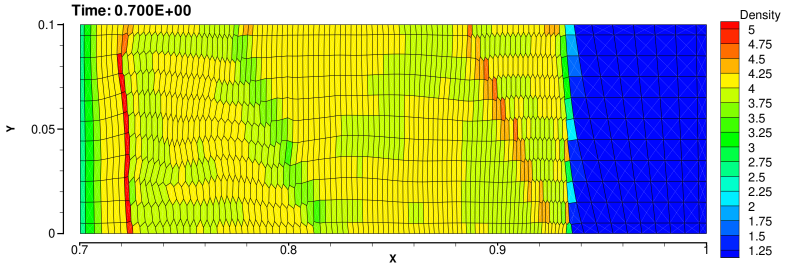


FIG. 8. Piston results for unstructured grid, merit factor 0.5.

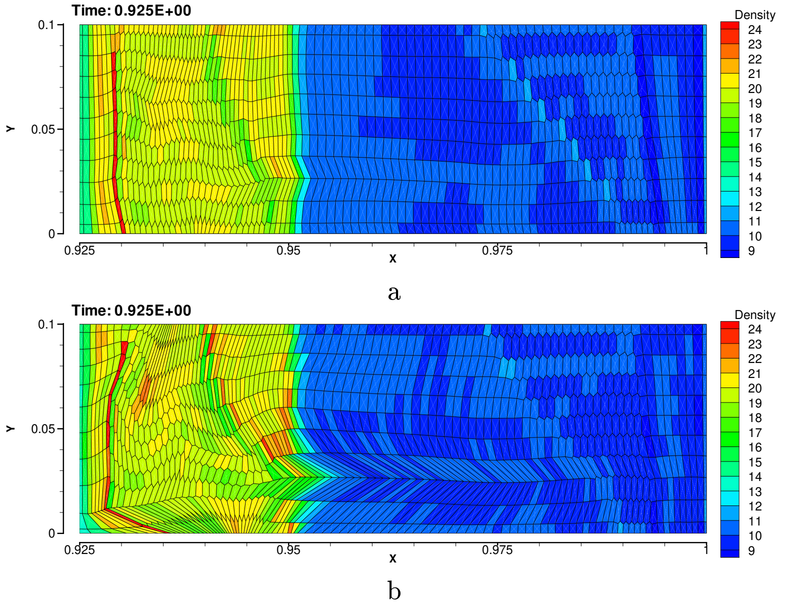


FIG. 9. Piston results for time 0.925, merit factor = 0.5. For clarity the x scale has been stretched. (a) Maximum angle neighbors. (b) Maximum ratio from candidate neighbors.

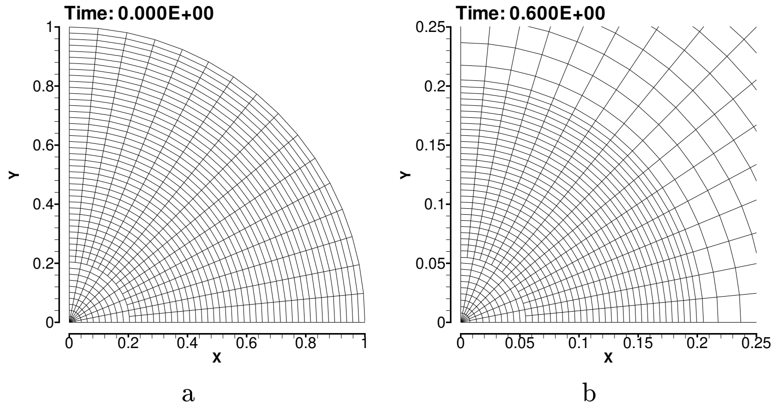


FIG. 10. Grid for Noh Problem. (a) Initial. (b) Final.

shows results for the two methods. Although both methods give good results the maximum angle neighbor choice clearly gives a smoother grid and more planar shock wave.

The sensitivity of computational results to the choice of limiter function neighbors varies between problems. Where a difference between the two methods is seen, the maximum angle choice gives a smoother mesh.

4.2. Noh problem

The Noh problem [8] is a well known test problem. A perfect gas with $\gamma = 5/3$ is given an initial inward radial velocity of magnitude 1.0. A circular shock wave is generated which at time 0.6 has a radius of 0.2. Fig. 10 shows the initial and final grid used for this problem. Using an unstructured grid has allowed us to coarsen the grid near the center. Halving the angular resolution in this area allows a larger time step that results in a total CPU time 43% of the time required to solve the problem with constant angular zoning equal to the outer area. The density profile shows that the change in resolution does affect the final solution, Fig. 11. The change in grid resolution has introduced a single oscillation in the density profile; outside of this area the density profile agrees well with the results with no grid coarsening. Fig. 11 also shows that changing the merit factor has no significant effect on the final solution.

The difference due to choice of the merit factor can be seen is in the behavior of the grid immediately after the shock wave has reached the interface between the coarse and fine zoning. At this time the layer of pentagonal interface zones become non-convex, Fig. 12a, with the merit factor having no effect. The zones become non-convex because of the difference in artificial viscosity forces at the interface points. The merit factor does affect the speed at which the grid is restored, with a lower merit factor, Fig. 12b, taking longer than a higher merit factor, Fig. 12c. Modifying the subzonal pressures so that they are higher in the region of a shock wave may mitigate this problem. As subzonal pressures do not resist volumetric zone deformation, this should not affect the final results.

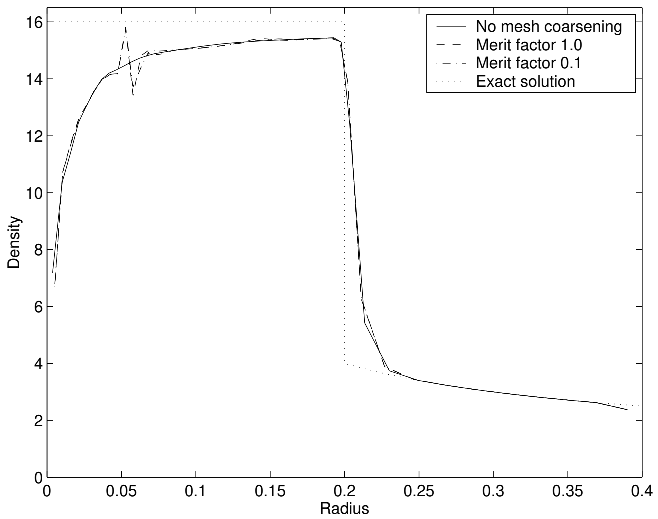


FIG. 11. Comparison of Noh problem results.

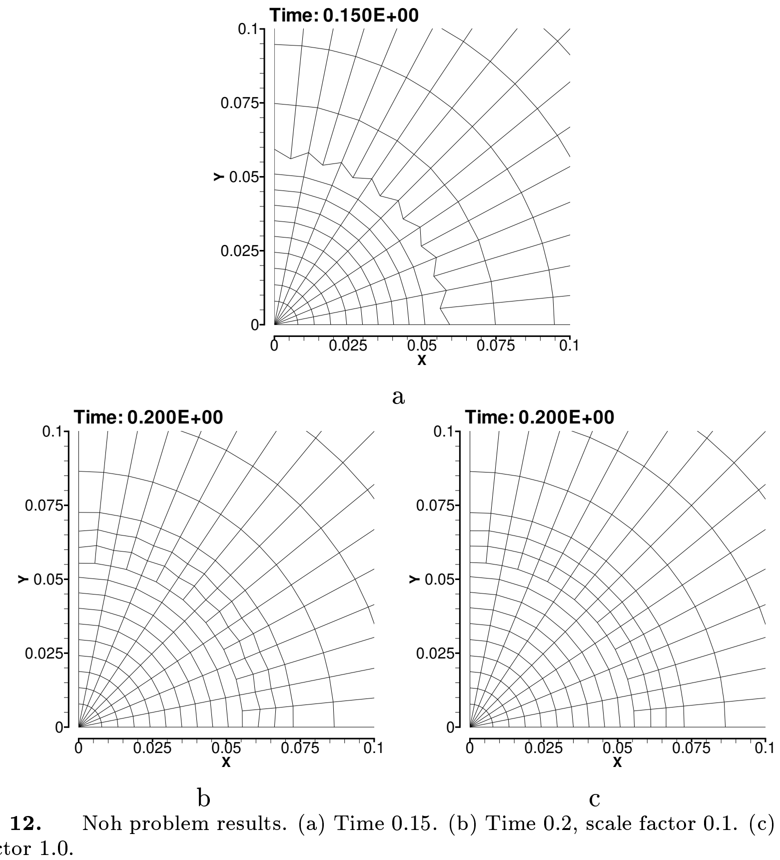


FIG. 12. Noh problem results. (a) Time 0.15. (b) Time 0.2, scale factor 0.1. (c) Time 0.2, scale factor 1.0.

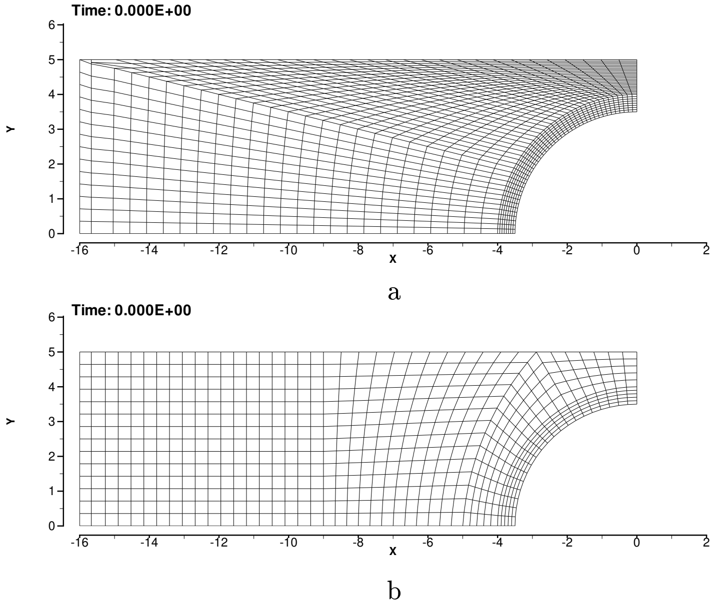


FIG. 13. Initial grids for shaped charge problem. (a) Logically rectangular grid. (b) Unstructured quadrilateral grid.

4.3. Shaped charge grid

This problem shows the benefit of an unstructured grid over a structured grid when the geometry is not well suited for a structured grid. We have taken the geometry from a hemispherical shaped charge problem described in a paper by Lee [6], who used a logically rectangular grid. As the test code only contains a perfect gas equation of state, we changed the initial conditions of the problem. Fig. 13 shows two initial grids, the first a logically rectangular grid that is equivalent to that used by Lee, the second an unstructured quadrilateral grid. The unstructured grid code is employed for both grids. Every zone uses a perfect gas equation of state with $\gamma = 1.4$; the section equivalent to the hemispherical liner has an initial density of 10.0, while the rest has an initial density of 1.0. To generate a shock wave the left most column of zones are given a specific internal energy of 60.0, all other zones starting cold. The top, bottom and left boundaries are considered to be walls. These initial conditions generate a planar shock wave that travels from left to right, and then interacts with the circular area of high density. This produces deformations in this area similar to those seen in the shaped charge simulation by Lee.

Fig. 14 shows results for both grids at time 0.475, shortly before the planar shock wave reaches the high density area. The unstructured grid shows a planar shock, with no grid problems. The logical grid shows grid problems in the upper half of the grid, where it is initially distorted. By time 0.75, see Fig. 15, the logical grid shows severe grid distortion, and shortly after this time the calculation is halted due to inverted zones. The unstructured grid is still smooth and it is possible to continue the calculation until the material distortions become too large for a fixed connectivity Lagrangian mesh.

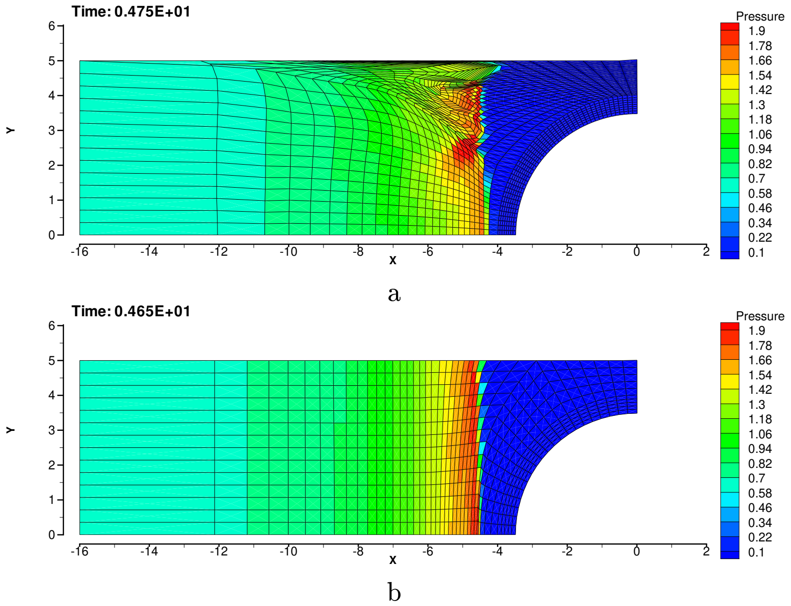


FIG. 14. Results at time 0.475, merit factor 0.5. (a) Logical grid. (b) Unstructured grid.

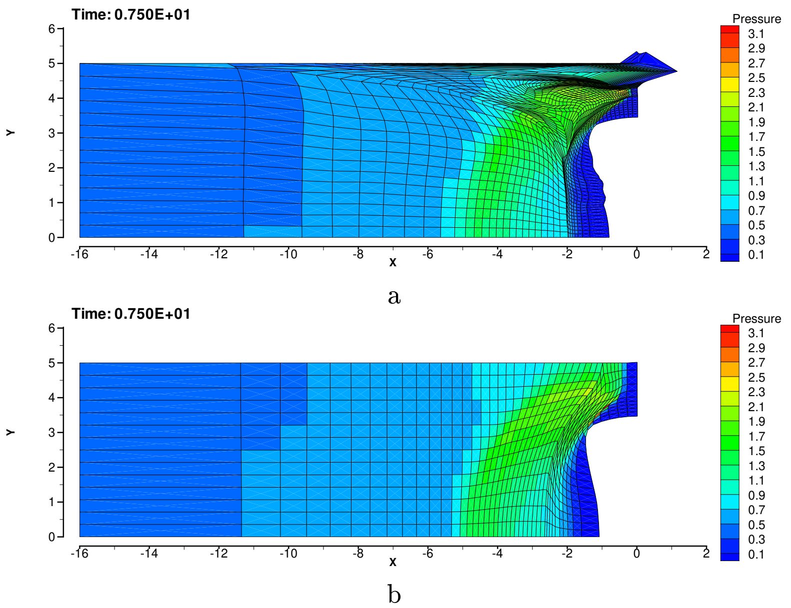


FIG. 15. Results at time 0.75, merit factor 0.5. (a) Logical grid. (b) Unstructured grid.

5. CONCLUSIONS

A compatible Lagrangian hydrodynamics algorithm has been implemented on an unstructured grid, where the zones consist of arbitrary polygons. Implementing the artificial viscosity requires a different method for choosing the neighbor edges for the limiter function than that used on a logical mesh. On an unstructured mesh there is no unique choice for neighbor edges. Two approaches were investigated, of the two the best was to choose the edge with the maximum angle to the center edge. Numerical results showed that results can be sensitive to this choice so investigating the limiter function further could further improve our results.

The subzonal pressure approach is effective at controlling mesh distortion, but does not prevent zones becoming concave while a shock wave is passing through the zone. This can affect the final value of solution variables in the concave and surrounding cells. So a method for preventing the cells becoming concave would improve results. Modifying the subzonal pressure force so that it is larger in zones in the vicinity of a shock wave is a possible strategy.

Numerical results show that the compatible hydrodynamics algorithm is well suited for unstructured mesh codes. An unstructured grid allows the solution of a wider class of problems than a structured grid. For future work an Arbitrary Lagrangian-Eulerian (ALE) algorithm will be added to the unstructured code, also grid reconnection will be added. These will give the code the ability to smooth and change the grid during a calculation.

ACKNOWLEDGMENT

We thank R. Bos and C. Rosculp for their advice on data structures for unstructured grids, and E. Caramana and L. Margolin for useful discussions and comments. This research was supported by the Department of Energy, under contract W-7405-ENG-36.

REFERENCES

1. F. Aurenhammer, Voronoi diagrams: A survey of a fundamental geometric data structure, *Computing Surveys* **23**, 345 (1991).
2. D.E. Burton, *Connectivity structures and differencing techniques for staggered-grid free-Lagrange hydrodynamics*, Tech. Rep. UCRL-JC-110555, Lawrence Livermore National Laboratory (1992).
3. E.J. Caramana, D.E. Burton, M.J. Shashkov, and P.P. Whalen, The construction of compatible hydrodynamics algorithms utilizing conservation of total energy, *J. Comput. Phys.* **146**, 227 (1998).
4. E.J. Caramana and M.J. Shashkov, Elimination of artificial grid distortion and hour-glass type motions by means of Lagrangian subzonal masses and pressures, *J. Comput. Phys.* **142**, 521 (1998).
5. E.J. Caramana, M.J. Shashkov, and P.P. Whalen, Formulations of artificial viscosity for multi-dimensional shock wave computations, *J. Comput. Phys.* **144**, 70 (1998).
6. W.H. Lee, Two-dimensional Lagrangian method for elastic-plastic flow, *Comput. Methods Appl. Mech. Engrg.* **156**, 149 (1998).
7. L.G. Margolin, *A centered artificial viscosity for cells with large aspect ratios*, Tech. Rep. UCRL-53882, Lawrence Livermore National Laboratory (1988).
8. W.F. Noh, Errors for calculations of strong shocks using an artificial viscosity and an artificial heat flux, *J. Comput. Phys.* **72**, 78 (1987).