Journal of Computational Physics **166,** 173–185 (2001)
doi:10.1006/jcph.2001.6648, available online at http://www.idealibrary.com on IDEAL®

# NOTE

# Timestep Relaxation with Symmetry Preservation on High Aspect-Ratio Angular or Tapered Grids[1]

E. J. Caramana

*Applied Theoretical and Computational Physics Division, Los Alamos National Laboratory, Los Alamos, New Mexico 87545*
E-mail: ejc@lanl.gov

Received April 13, 2000; revised August 29, 2000

*Key Words:* Lagrangian hydrodynamics; spatial grid stiffness; fluid dynamics; shock wave simulation; grid generation.

## 1. INTRODUCTION

The purpose of this note is to provide a simple and effective solution to the problem of spatial grid stiffness and the consequent timestep restriction that occurs when grids are constructed with angular coordinates, or in any grid construction where the grid lines become more closely spaced in one direction. Logically connected grids commonly generated with polar or spherical coordinates in 2D or 3D have this difficulty. The north and south poles of meridian grids are also prime examples.

Consider the single zone shown in Fig. 1a that appears as a rectangle that is much longer in the $x$ direction than in the $y$ direction. Such a zone is said to have a high aspect-ratio since its length divided by its width $\Delta x/\Delta y$ is much greater than one. For a velocity flow $\mathbf{v}$ that is oriented mostly along the elongated $x$ direction two difficulties occur. The first and principal problem is that for an explicit time advance the timestep is restricted by the time for a sound wave to transit the small length scale $\Delta y$, when, assuming a roughly isotropic gradient, the actual gradient scale lengths are at the much larger distance $\Delta x$. The second difficulty is that forces oriented mainly along the $x$ direction, if slightly skewed into the $y$ direction

[1] The U.S. Government's right to retain a nonexclusive royalty-free license in and to the copyright covering this paper, for governmental purposes is acknowledged.

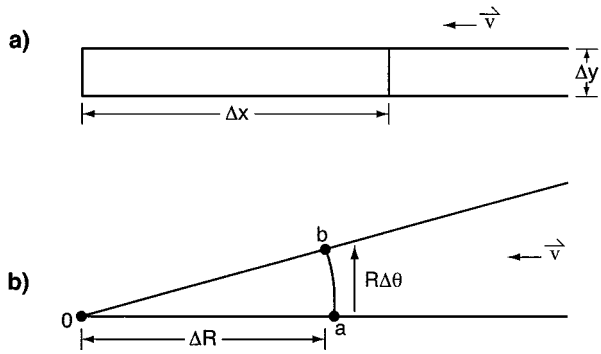


**FIG. 1.** (a) Rectangular high aspect–ratio zone with velocity **v** in the $x$ direction; (b) Angular high-aspect-ratio zone with velocity **v** in the radial direction.

due to truncation error, apply a large effective torque to this zone that can lead to spurious hourglass-like distortions that destroy the entire calculation. To avoid these difficulties one should not have high aspect–ratio zones in any calculation where the principal flow field and gradient scales are oriented roughly parallel to the elongated zone direction. However, it is not always possible to avoid this situation, as can be seen from considering the angular wedge shown in Fig. 1b. Here the zone inscribed by points $ab0$ has a length $\Delta R$, where $R$ is the major radius, and a width of order $R\Delta\theta = \Delta R\Delta\theta$. It thus has an aspect ratio of $\Delta R/(\Delta R\Delta\theta) = 1/\Delta\theta$, which for an angular spacing of $1^\circ$ results in an aspect ratio and spatial grid stiffness factor of $\approx 60$. Since the major radius $R$ increases as one moves to the right of point 0, eventually $R\Delta\theta \geq \Delta R$, so that for a constant value of $\Delta R$ the zones become about equal in both dimensions, giving roughly unit aspect ratio, and then reversing with zones longer in the angular direction than radially.

An effective solution to both of the above noted problems is to replace the simple logically connected grid with an unstructured one with zones of arbitrary, or properly restricted, topology so that all zones always have an aspect ratio near unity [1]. However, other difficulties arise in this case so that one may wish to preserve the simple logical grid structure. First, the simple angular grid may have the important property that it supports the limit of perfectly one-dimensional cylindrical or spherical symmetry in terms of the radial coordinate [2, 3], a property that cannot be achieved with an arbitrary unstructured grid [1]. In fact, it is true in general that changes in grid topology introduce numerical perturbations into the solution, much like that seen with waves moving through a medium with sharp changes in the index of refraction. Also, a simple grid structure (for instance, all quads or hexes) facilitates the inclusion of physics such as the transport of radiation that is more difficult to discretize using arbitrary zone topology [4]. Thus we wish to maintain the simple grid structure obtained with a logical grid, while correcting the timestep and hourglass difficulties that arise from the regions of these grids that have high aspect–ratio zones.

To put this in a global perspective, consider the polar grid shown in Fig. 2 with a center of convergence at point $O$; we refer to the radial lines as "$k$-lines" and to the angular lines as "$l$-lines." The distance between adjacent points along an $l$-line is $R\Delta\theta$, at some major radius $R$, whereas the distance between adjacent points along a $k$-line is $\Delta R$; both types of lines may be unequally spaced. Points shown as open circles can be removed from the grid in

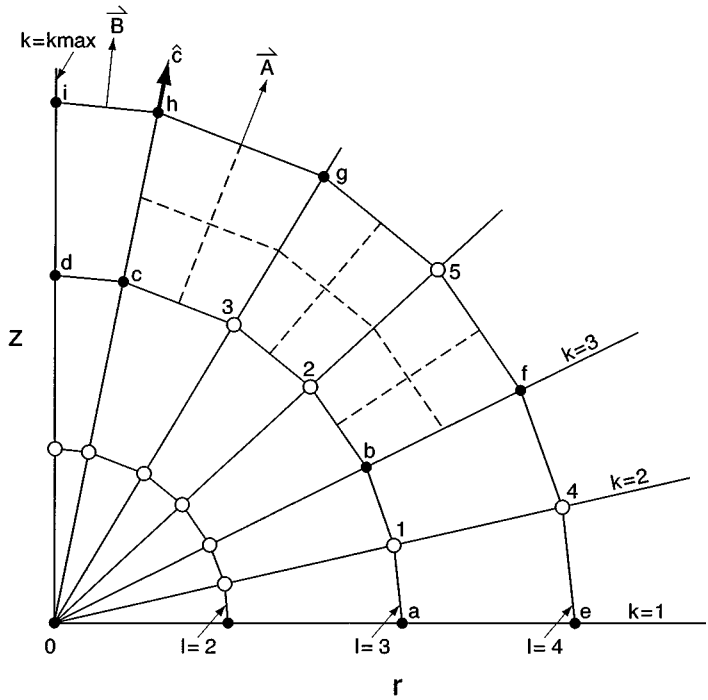


**FIG. 2.** Angular grid with center of convergence at point. "$O$." Solid dots are dynamical points; circles denote nondynamical points; a "generalized zone" is delineated by points $bf5ghc32b$; dashed lines are median mesh in this region. Vector **A** is normal to line $gh$ and has as its magnitude the distance $gh$; analogously for **B**. $R\Delta\theta \ll \Delta R$, so figure is not to scale.

one of two ways. First, they can just be eliminated directly, wherein the $k$-lines that contain them terminate before reaching the center of convergence. This is a direct way to create an unstructured grid from a logical one and has been used effectively in modeling flows on the surface of a sphere, where it is referred to as a "reduced grid" model [5, 6]. It is important to note that these researchers conclude that the reduced grid method "reduces the accuracy less than either filtering or using implicit techniques" [5]. Implicit time differencing introduces a "stabilizing denominator" that acts to compress all characteristic speeds so that they are less than the local grid speed $\Delta x/\Delta t$ [7]. This results in unwanted phase error in high-speed flow calculations where the CFL timestep condition is both an accuracy and a stability criterion. That is, implicit time differencing smooths dependent variables in a somewhat uncontrolled manner. The solution presented herein is different from the reduced grid method in that it retains the points shown as open circles in Fig. 2 and thus retains the underlying grid topology and its previously noted favorable properties; however, our solution is like the reduced grid method in that these points are removed from the dynamics of the calculation, allowing the timestep restriction to be relaxed.

Our solution is illustrated by the following example. Suppose a purely 1D flow calculation is performed on the 2D grid of Fig. 2, and we want the timestep to be set only by the CFL number in the radial direction. First, the masses and forces associated with all the points in Fig. 2 are computed using compatible finite volume differencing [8]. Then we add the masses and forces associated with all points on an $l$-line to those of the $r$-axis, the $k = 1$ line of Fig. 2. However, we rotate the force by the angle between the $r$-axis and the $k$-line

point where it is initially defined so that all forces are along the $r$ direction. The points on the $r$-axis become the "dynamical" points, because their velocities are advanced in time by solving the momentum equation. The other points of an $l$-line are the "nondynamical," or enslaved, points. The velocities of these points are set equal to those of points on the $r$-axis at the same value of major radius after an opposite rotation through the previously mentioned angle. The work done by all forces is now calculated compatibility [8] with respect to all zones; a staggered grid placement of dependent variables is assumed: position and velocity at nodes; density, internal energy, and others in zones. The force addition step is what ensures numerical stability based only on the scale length $\Delta R$. Last, the coordinates of all points are advanced in time for a Lagrangian calculation; if Eulerian, all variables are fluxed with respect to the fixed grid, a Lagrangian step plus remap. The 2D grid is thus just a somewhat clumsy device on which to perform a purely 1D computation.

The above idea is generalized in the remainder of this paper to achieve a grid that effectively has nearly unit aspect ratio when viewed from only the locations of the dynamical points. The underlying structure of the grid is maintained; spatial grid stiffness is eliminated by making all of the points on certain $k$-lines nondynamical inside an initially determined value of major radius, or distance from a tapered end. The masses and forces of these nondynamical points are then "donated" to nearby dynamical points, and their velocities enslaved by appropriate interpolations. One chooses how much spatial grid stiffness, if any, is to be allowed in a calculation as part of the initialization. Thus at the initial time the spatial scale lengths that the grid is capable of resolving are determined by the choice of dynamical and nondynamical points associated with the underlying spatially stiff grid. The underlying grid is logical in this paper but could be an unstructured grid; for example, a soccer-ball tesellation of a sphere in 3D that retains the spatial grid stiffness difficulty in the radial direction. Our exposition is in 2D; all changes needed for 3D are given. The basic discretization method is compatible, control volume differencing of the fluid equations in Lagrangian form as detailed in its various aspects in previous work [2, 3, 8–10].

## 2. FORM OF EQUATIONS AND GRID POINT DESIGNATION

In the form of control volume differencing that we utilize on a staggered spatial grid the momentum equation is written as

$$\Delta \mathbf{v}_p = \mathbf{F}_p \Delta t / M_p \equiv \sum_z \mathbf{f}_z^p \Delta t / M_p, \tag{1}$$

where $\Delta \mathbf{v}_p$ is the change in the velocity of a point $p$ in time $\Delta t$, $\mathbf{F}_p$is the total force on this point, and $M_p$ is its mass. The total force on a point is composed of the corner forces $\mathbf{f}_z^p$ that act from the zone $z$, and onto the point $p$, where the indices $z$ and $p$ range over all zones and points of the underlying grid. The exact construction of these forces (mean zone pressure, subzone pressure, artificial viscosity, and material strength) utilizing the coordinate–line and median meshes (solid lines and dashed lines, respectively, in Fig. 2) is detailed in [8–10]. The important result that we refer to as "compatibility" of force and work is given by the equation for the change in specific internal energy $e_z$ of a zone in terms of the work done by these same corner forces as

$$\Delta e_z = -\sum_p \mathbf{f}_p^z \cdot \mathbf{v}_p^{n+1/2} \Delta t / M_z, \tag{2}$$

where $M_z$ is the mass of a zone, and $\mathbf{f}_p^z = \mathbf{f}_z^p$ except that sums are performed over the lower index with the upper one fixed. Also, $\mathbf{v}_p^{n+1/2}$ is the average of the grid point velocity at the old and new time level [8].

The grid shown in Fig. 2 is divided into sets of dynamical points (solid dots) and nondynamical points (open circles) by the following procedure. We create a template that lists the points that are tagged dynamical on every $l$-line. First, all points on the line $k = 1$ are tagged dynamical, then we proceed from the line $l = 1$ and decide which additional points on an $l$-line are also dynamical. The next $k$-line to become dynamical after $k = 1$ is $k = k_{\max}$, which occurs when $R\theta \geq \alpha \Delta R$ along an $l$-line, where $(0.2 \leq \alpha \leq 1)$ is the spatial grid stiffness parameter set by the user. Once a point on a $k$-line is dynamical, all points at larger major radii (larger $l$ index) are also. Next, by comparing the broken line distances from a point on an $l$-line to its two adjacent dynamical points, versus the local $\alpha \Delta R$ between $l$-lines, one decides if that point is also to be declared dynamical. At some outer $l$-line all points are dynamical and the procedure is terminated. One now has two lists of points: dynamical points with logical coordinates $(k, l)$ and the value of the integer stride index to their nearest upper and lower neighbor $l$-line dynamical points (if either is missing, the stride index is zero), and nondynamical points with logical coordinates $(k, l)$ and the value of the integer stride index to their nearest neighbor upper and lower $l$-line dynamical points. In addition, at the initial time, interpolation coefficients that enable one to donate force and mass to the dynamical points from the nondynamical points, and velocity the other way, are set. The exact form and use of these coefficients is discussed next.

## 3. MASS, FORCE, WORK, AND VELOCITY INTERPOLATION

First, all forces are calculated with respect to the underlying grid. It is assumed that these forces obey a simple composition rule such that if the nondynamical points are erased the primitive force, as calculated on the resulting larger grid stencil, is equal to that obtained by adding the force computed on the nondynamical points to the dynamical ones (exceptions to this rule are dealt with last). Although these forces must in general be rotated, our discussion first omits this point, which is included later.

Both forces and masses are donated to their nearest neighbor dynamical points along an $l$-line with coefficients of either one or one-half. For instance, in Fig. 2 all of the force and mass of point *3* is donated to point *c*, and likewise, point *2* donates all force and mass to point *b*, while point *5* contributes one-half of its force and mass to each of points *f* and *g*. The force and mass of all points along the line $l = 2$ in Fig. 2 are donated to the single dynamical point on the $k = 1$ line, and inside this radius the calculation becomes one-dimensional. The velocity is advanced on all dynamical points by means of Eq. (1). The work done must be calculated using Eq. (2), which is written in terms of the undonated corner forces $\mathbf{f}_p^z$. The most straightforward way to do this is to set $\mathbf{v}_p^{n+1/2}$ at the nondynamical points by interpolating the known $\mathbf{v}_p^{n+1/2}$ from the dynamical points with the same coefficients, and opposite angle of rotation, used to donate forces. Then Eq. (2) can be applied directly to all zones of the underlying grid to yield a value of $\Delta e_z$ for each zone. With area–weight differencing [2, 8], the above procedure is necessary since it is the area–inertia and not the nodal mass that is directly donated. Next, the velocity $\mathbf{v}_p^{n+1/2}$ is discarded from the nondynamical points; $\mathbf{v}_p^{n+1}$ is then set at these points by interpolation from the neighboring dynamical points, after a rotation, using coefficients that are ratios of the distance along the

broken $l$-line segment at the initial time to give a second-order-accurate interpolation. Last, coordinates of all points are advanced in time.

Although all work has now been computed, our calculation of both density and specific internal energy for zones containing nondynamical points is not complete. Indeed, when looked at from only the dynamical points the quadrilateral zones of the underlying grid of Fig. (2) become of quite general topology. Galilean invariance of Eq. (2) requires that $\sum_p \mathbf{f}_p^z = 0$. That is, the corner forces must sum to zero inside a zone, and thus from Eq. (2), the "change" in specific internal energy must be uniform in this "generalized zone"; an example is the volume delineated by points $bf5ghc32b$ in Fig. (2). Thus we average $\Delta e_z$ with respect to zone mass throughout all zones above and adjacent to any segment of an l-line with dynamical endpoints. Although $\Delta e_z$ is uniform within the generalized zones on a timestep, $e_z$ can still vary with respect to the zones of the underlying grid due to other physical processes, such as the transport of radiation, that are performed on the underlying grid structure.

The zone density is also averaged in the same manner as $\Delta e_z$ to define a mean density for the generalized topology zones. This is done because we must define new "subzonal" densities in the generalized topology zones in order to calculate subzonal pressure forces that resist grid tangling [9]. These subzonal densities are set by averaging the density, using the Lagrangian subzonal corner mass $m_z^p$, so that two subzone densities are defined along the top and along the bottom of each $l$-line segment containing dynamical endpoints; subzonal pressure forces are then calculated with respect to the underlying grid as before [9].

Last, the assumption previously made of a "composition rule" regarding the addition of forces is not always valid for our edge-centered artificial viscosity. This is because the artificial viscosity is velocity-dependent [10]. If an edge is formed using two nondynamical points whose velocities are interpolated from the same dynamical point velocities, then this edge is not used to define an artificial viscosity force contribution. This situation occurs along the $l$-line of Fig. 2. Thus, along $l$-lines the artificial viscosity is defined separately and then added into the respective corner forces of each two dynamical point pairs along the given $l$-line. This edge viscosity force is computed as before [10], but using only dynamical grid point data and a single median mesh vector that is the arithmetic average of those median mesh vectors lying between each two $l$-line dynamical point pairs. However, the artificial viscosity force along all $k$-line edges is still valid since it is computed from velocity data derived from different sets of dynamical points.

### 3.1. *Rotation of Force and Velocity*

It is apparent from the example given in the Introduction and from the curved grid of Fig. (2) that the force and velocity vectors should be rotated before the interpolation factors are applied in order to obtain exactly the limit of one-dimensional cylindrical or spherical flow. The argument for employing a rotation when the flow field is not totally one-dimensional, but where the gradient scale lengths of the problem are still approximately isotropic based on the spacing of the dynamical points, relies on the fact that pressure always acts normal to the grid lines and thus produces a force in approximately the local outward normal $\hat{c}_p$ direction. Thus, if the grid curvature is locally, approximately constant about a point (as it is for well-resolved spatial gradients), and the pressure force about one point is to be donated to a nearby point, then points should be made nondynamical on either side of a dynamical point in a symmetrical manner so that changes in the net donated force due

to grid curvature will be close to zero. However, this means removing points by simple factors of 2 or 4 and thus creating specially preferred numbers of $k$-lines for performing calculations. This we wish to avoid. The net effect of a symmeterical removal of points is instead mimicked by rotating the donated force through the angle between the local outward normal directions of a dynamical point and its nondynamical neighbors. As noted, this gives the correct symmetry limit when pressure is constant between adjacent $l$-lines.

The angle of rotation must be suitably defined and allowed to vary with time. The outward normal to the $l$-line surfaces of the dynamical points of Fig. 2, denoted as $\hat{c}_p$, at point $p$ is defined by fitting circles (2D), or spheres (3D), through a given dynamical point and its nearest neighbor $l$-line dynamical points by the prescriptions given as Eq. (8) of [2] and Eq. (15) of [3]; the former is repeated here as

$$\hat{c} = (B^2\mathbf{A} + A^2\mathbf{B})/|B^2\mathbf{A} + A^2\mathbf{B}|, \tag{3}$$

where the vectors $\mathbf{A}$ and $\mathbf{B}$ are shown in Fig. (2) to yield a value of $\hat{c}$ at point $h$.

It is imperative that all of the interpolations performed utilize time-independent coefficients; else when stressed, such calculations will exhibit negative feedback from the nondynamical points that will result in numerical instability. To avoid this problem we employ the following procedure to compute $\hat{c}_p$ at the nondynamical points at a later time, given that we always know $\hat{c}_p$ at all dynamical points from Eq. (3). At the initial time the ratio of the angle between every nondynamical point and each of its two neighbor dynamical points, divided by the angle between these neighbor dynamical points, is stored as an interpolation fraction; they sum to unity. Then the value of $\hat{c}$ at a nondynamical point is obtained in time from those of its two neighbor dynamical points by the following steps: Find the angle between the two dynamical point neighbors (arccos $(\hat{c}_1 \cdot \hat{c}_2)$, for points 1 and 2) of a nondynamical point; rotate both $\hat{c}_1$ and $\hat{c}_2$ through the respective angles found by multiplying arccos $(\hat{c}_1 \cdot \hat{c}_2)$ by the respective angular fraction set initially; multiply these vectors by the interpolation factors used to set the advanced velocity $\mathbf{v}_p^{n+1}$ at the nondynamical point; add the result and renormalize the magnitude to unity. The $\hat{c}$'s along an $l$-line that contains only one dynamical point (for example, the $l = 2$ line of Fig. 2) are fixed at their initial value. Now, given the $\hat{c}_p$ vectors at all points, the cosine of the angle between any two points is well defined and all rotations in 2D are trivial.

A rotation in 2D of a vector $\mathbf{a}$ with respect to an angle $\cos\theta \equiv \hat{c}_1 \cdot \hat{c}_2$, and with a sense of direction $\hat{\omega}$ defined by $\hat{\omega}\sin\theta \equiv \hat{c}_1 \times \hat{c}_2$, resulting in a vector $\mathbf{a}'$ can be written as

$$\mathbf{a}' = \mathbf{a}\cos\theta + (\hat{\omega} \times \mathbf{a})\sin\theta. \tag{4}$$

This expression enables us to derive the equivalent one that is needed to perform rotations in 3D where a vector $\mathbf{F}$ may not lie in the plane of $\hat{c}_1$ and $\hat{c}_2$. Separating $\mathbf{F}$ into $\mathbf{F}_\parallel \equiv (\mathbf{F} \cdot \hat{\omega})\hat{\omega}$ and $\mathbf{F}_\perp = \mathbf{F} - \mathbf{F}_\parallel$, substituting $\mathbf{F}_\perp$ into Eq. (4) as $\mathbf{a}$, and then adding the component $\mathbf{F}_\parallel$ that is unchanged by the rotation to the rhs, one arrives at the result

$$\mathbf{F}' = \mathbf{F}\cos\theta + (\hat{\omega} \times \mathbf{F})\sin\theta + (1 - \cos\theta)(\mathbf{F} \cdot \hat{\omega})\,\hat{\omega}. \tag{5}$$

This is the only change necessary to carry our procedure over into three dimensions.

Rotations only give accurate answers when performed with respect to points with nearly constant curvature. Large errors can result if rotations are performed when short spatial scale

hourglass-like motions are present, since vector direction is very sensitive to these short scale distortions. Thus, rotations are not performed if the curvature, as measured between the dynamical points of an $l$-line, varies substantially about a point. The amount of rotation performed is automated by letting $\cos\theta \to \chi_p \cos\theta + (1 - \chi_p)$, where $\chi_p$ lies between zero and one and is determined by the RMS curvature deviation about point $p$ as defined by Eq. (34) in [3]. (If $\chi_p = 0$, then $\cos\theta = 1$.) Although $\chi_p$ is defined directly only with respect to the dynamical points, it is interpolated to the nondynamical points by means of the coefficients used to set $\mathbf{v}_p^{n+1}$ at these points.

As a simple illustration of our interpolation procedure, consider the example given in the Introduction where the 2D grid of Fig. 2 becomes 1D by making all points that are not on the $k = 1$ line nondynamical. For this case all interpolation coefficients are unity; all $\hat{c}_p$ direction vectors are fixed in time at their initial values so that all rotation angles are fixed in time. The specific internal energy and mean zone density are uniform between adjacent $l$-lines; the subzone corner density has only two distinct values between adjacent $l$-lines. The artificial viscosity acts only between adjacent $l$-lines. This is because each $l$-line contains only one dynamical point, and therefore an edge cannot be defined along these $l$-lines.

## 4. TIMESTEP DETERMINATION

The force addition procedure enables us to determine the overall timestep using only the distances between adjacent dynamical points. An effective wave speed $c^*$ must be defined between these points. This is done in conjuction with the artificial viscosity as follows. The maximum value of the zone sound speed squared $c_s^2$ of the zones adjacent to an edge is added to the edge artificial viscosity sound speed squared (the scalar part of the edge artificial viscosity is $\rho C_q^2$ [10]) to obtain $c^* \equiv \sqrt{c_s^2 + C_q^2}$. The new timestep $\Delta t^{n+1}$ is then determined on the predictor step as

$$\Delta t^{n+1} = \min\left( f_1 (\Delta x)_{i,j}^n / c^*, 0.8\, f_1 / (\nabla \cdot \mathbf{v})_z^n \right), \tag{6}$$

where $(0.25 \le f_1 \le 0.5)$ and $(\Delta x)_{i,j}^n$ is the straight line distance between adjacent dynamical points $i$ and $j$ at time level $n$. The first entry on the rhs of Eq. (6) is taken over all of the edges connected by two adjacent dynamical points, while the second entry is defined with respect to all zones of the underlying grid and requires that these zones not develop negative volumes. Since the second entry is the fractional volume change of a zone in a timestep, it does not reflect spatial grid stiffness. The predictor step as given by Eqs. (1) and (2) with forces $\mathbf{f}_z^p$ computed at the $n$ time level is numerically unstable with respect to pressure forces. Stability and second-order accuracy in time are achieved by applying a single corrector step where all forces $\mathbf{f}_z^p$ are centered at the $n + 1/2$ time level, after which all variables are advanced in time.

## 5. NUMERICAL EXAMPLES AND CONCLUSIONS

As the first numerical example of the effectiveness of our procedure we choose Noh's problem. Here a shock wave propagates outward from the origin at the initial time; the detailed setup and result are given in [3, 10, 11]. It is run on an angular grid with $101 \times$

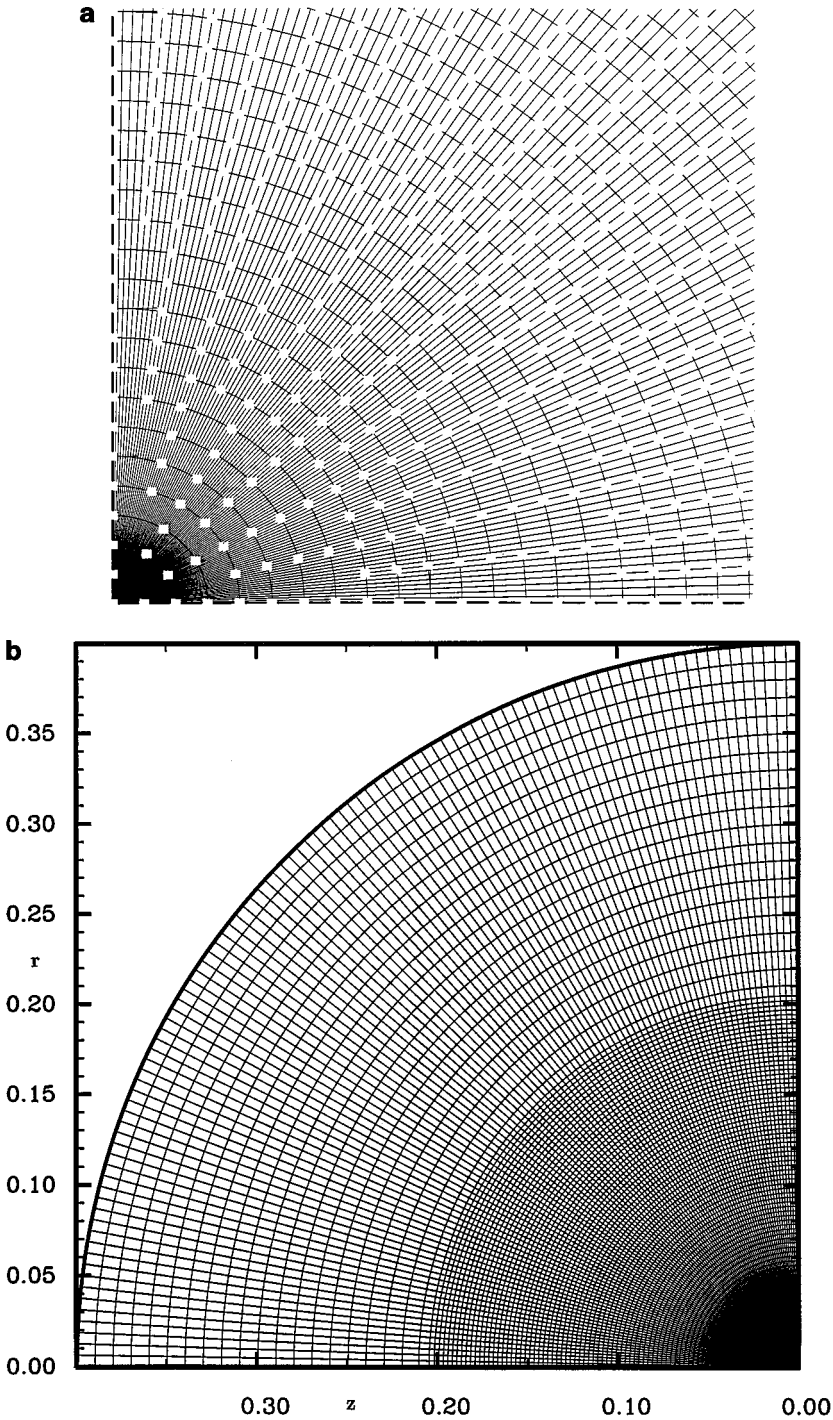


**FIG. 3.** (a) Noh's problem: inner portion of initial grid. White squares center on dynamical points. (b) Noh's problem: full grid $101 \times 101$ at final time $t = 0.6$. (c) Noh's problem: contours of density at time $t = 0.6$.

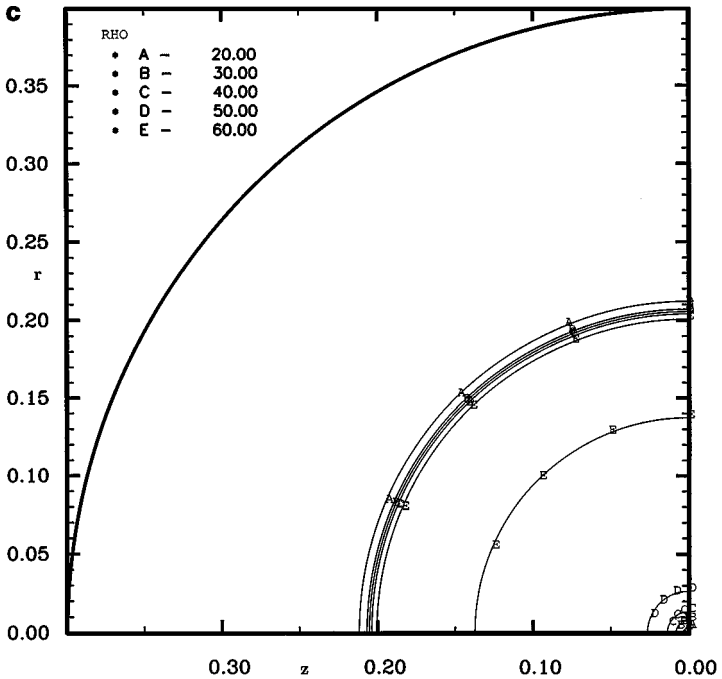


**FIG. 3**—*Continued*

$101(k, l)$ lines forming a 90° quadrant. Thus, the initial spatial grid stiffness near the origin is about 70 : 1. This spherical problem is run in 2D cylindrical geometry using control volume differencing, and with symmetry corrections [2]. The grid stiffness parameter $\alpha$ in the initialization is set to unity ($R\Delta\theta \approx \Delta R$). In Fig. 3a the inner portion of the grid is shown at the initial time with the dynamical points labeled as white squares. On the $l = 2$ line only the two points $k = 1, 101$ are dynamical, on the $l = 3$ line the points $k = 1, 33, 65, 101$ are dynamical, etc. For $l \geq 65$ all points are dynamical. The entire grid is shown in Fig. 3b at the final time of 0.6; a contour plot of the density at this time is given in Fig. 3c. This result agrees with previous large-angle calculations [10]. The calculation required about 900 timesteps, the same as needed for a 1D calculation in spherical coordinates [$f_1 = 0.25$ in Eq. (6)]; all spatial grid stiffness has been eliminated. The computational overhead associated with our interpolation procedure is found to be a negligible cost compared to that of the overall timestep.

The next example is a velocity-driven, spherical implosion that has a self-similar solution. The detailed setup and results are given in [2, 3]. This problem is run in 2D cylindrical geometry using an initial angular grid with 51 $k$-lines and 101 radial $l$-lines filling a wedge of 10° with respect to the $r$-axis (0.2° zoning). The initial grid stiffness parameter $\alpha$ is 0.2 ($R\Delta\theta \approx 0.2\Delta R$) or 5 : 1 instead of the 300 : 1 stiffness value that would result near the origin. The inner portion of the initial grid is shown in Fig. 4a with the dynamical points labeled as white squares. On the $l = 2$ line only the single point $k = 1$ is dynamical; on the $l = 3$ line points $k = 1, 51$ are dynamical; on the $l = 10$ line the points $k = 1, 11, 21, 28, 36, 43, 51$ are dynamical. For $l \geq 59$ all points are dynamical. Figure 4b shows the inner portion of the grid at a time of 0.8, after the shock wave has reflected from the origin and is moving outward; a contour plot of density at this time is given in

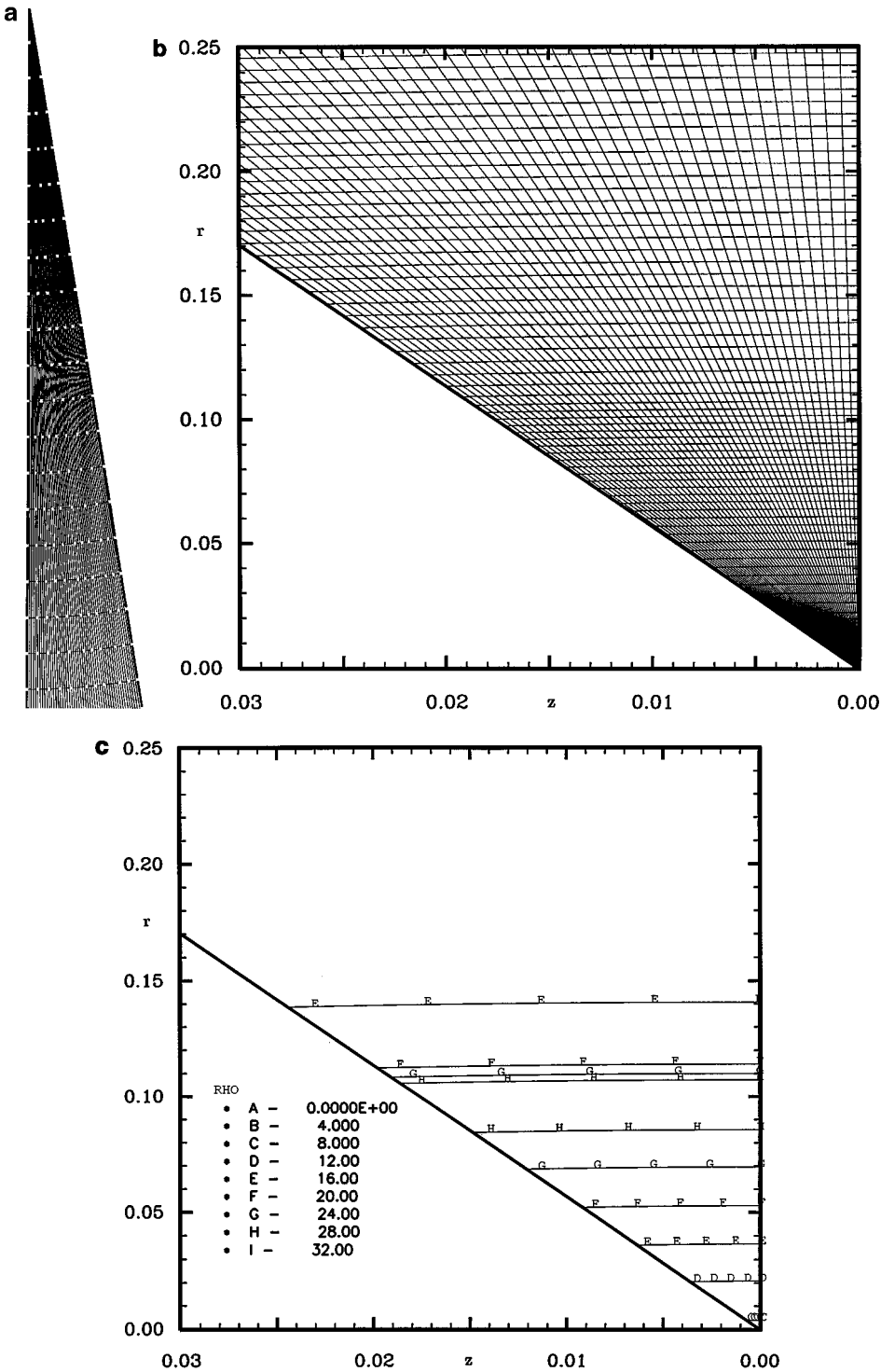


**FIG. 4.** (a) Self-similar implosion: inner portion of initial grid, 10° wedge. White squares center on dynamical points. (b) Self-similar implosion: inner portion of grid at final time $t = 0.8$. Shock wave is moving out as seen by compression wave at $R \approx 0.11$. (c) Self-similar implosion: contours of density at time $t = 0.6$. (d) Self-similar implosion: all points are dynamical. Grid is shown at time $t \approx 0.7$ when calculation terminates due to hourglass-type instability.

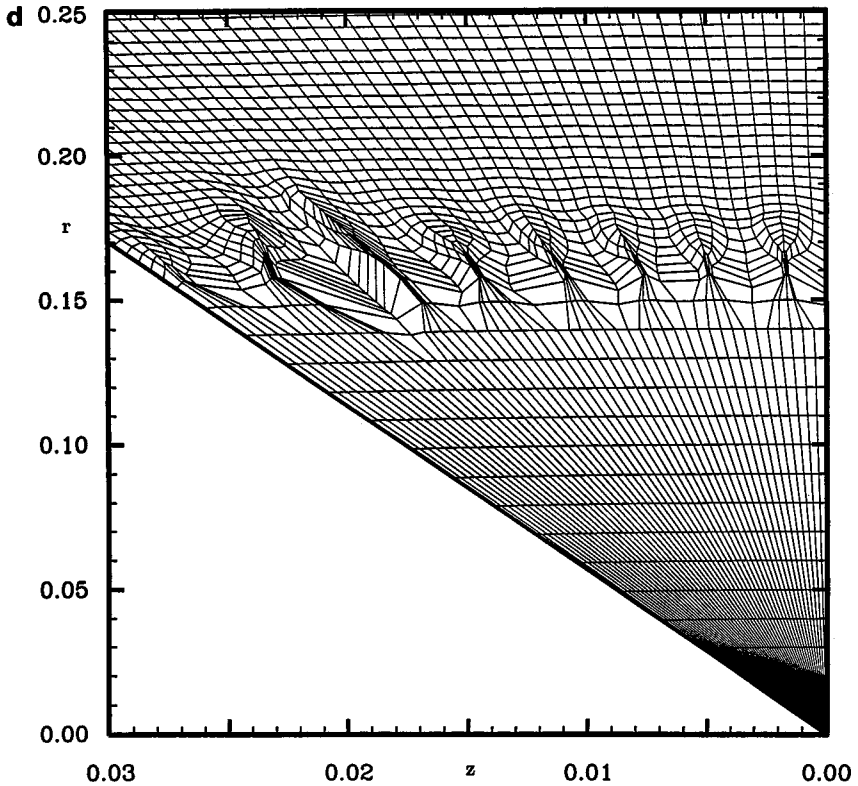


**FIG. 4**—*Continued*

Fig. 4c. These results are essentially the same as those given previously using large-angle zoning [2, 3]. It took just less than 2000 timesteps to run to the final time. If this calculation is performed without spatial grid stiffness removal the timestep decreases rapidly as the shock wave propagates into the region of increasingly high aspect ratio. It terminates due to a virulent hourglass-like distortion at about $t = 0.7$. The final grid in this case is shown in Fig. 4d. Grid distortion occurs at $r \approx 0.2$, where the zone aspect ratio is about 15 : 1 initially. Subzonal pressure forces do not stabilize this instability, and in fact make it appear somewhat sooner than with these forces absent. The interpolation procedure employed here removes this sensitivity to hourglass instability because the grid effectively has the aspect ratio of the generalized zones that are formed only by the dynamical grid points.

In conclusion, it is found that our procedure relaxes the timestep restriction associated with a grid that contains high aspect-ratio zones and prevents the development of an hourglass-type instability that can prevent spatially stiff calculations from being run to completion, while preserving the favorable symmetry and discretization properties of the simple underlying grid. Although only 1D examples have been shown, other "endurance-type" calculations have been performed to test the robustness of the method. It is found that the robustness of the calculations is not adversely affected by the presence of many nondynamical points if the complete interpolation procedure as detailed herein is implemented. The spatial grid resolution is made more isotropic as is desired, and the regularity conditions as a center of convergence is approached are automatically satisfied.

## ACKNOWLEDGMENTS

Discussions with the late S. Robert Orr motivated the ideas in this article. This work was supported by the Advanced Strategic Computing Initiative (ASCI) of the U.S. Department of Energy.

## REFERENCES

1. J. C. Campbell and M. J. Shashkov, *A Compatible Lagrangian Hydrodynamics Algorithm for Unstructured Grids*, Report LAUR-00-3231 (Los Alamos National Laboratory, 2000).
2. E. J. Caramana and P. P. Whalen, Numerical preservation of symmetry properties of continuum problems, *J. Comput. Phys.* **141**, 174 (1998).
3. E. J. Caramana, C. L. Rousculp, and D. E. Burton, A compatible, energy and symmetry preserving Lagrangian hydrodynamics algorithm in three-dimensional Cartesian geometry, *J. Comput. Phys*. **157**, 89 (2000).
4. J. E. Morel, personal communication.
5. J. M. Reisner, L. G. Margolin, and P. K. Smolarkiewicz, A reduced grid model for shallow flows on the sphere, in *Parallel Computational Fluid Dynamics: Implementations and Results Using Parallel Computers*, edited by A. Ecer, J. Periaux, N. Satofuka, and S. Taylor (Elsevier, Amsterdam/New York, 1995) p. 537.
6. P. J. Rasch, Conservative shape-preserving two-dimensional transport on a spherical reduced grid, *Mon. Weather Rev.* **122**, 1337.
7. E. J. Caramana, Derivation of implicit difference schemes by the method of differential approximation, *J. Comput. Phys.* **96**, 484 (1991).
8. 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).
9. E. J. Caramana and M. J. Shashkov, Elimination of artificial grid distortion and hourglass-type motions by means of lagrangian subzonal masses and pressures, *J. Comput. Phys.* **142**, 521 (1998).
10. 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).
11. W. F. Noh, Errors for calculations of strong shocks using an artificial viscosity and an artificial heat flux, *J. Comput. Phys.* **72**, 78 (1987).