



An arbitrary Lagrangian–Eulerian method with adaptive mesh refinement for the solution of the Euler equations

R.W. Anderson ^{*}, N.S. Elliott, R.B. Pember

Center for Applied Scientific Computing, Lawrence Livermore National Laboratory, Livermore, CA 94550, USA

Received 25 March 2003; received in revised form 4 November 2003; accepted 6 February 2004
Available online 7 May 2004

Abstract

A new algorithm that combines staggered grid arbitrary Lagrangian–Eulerian (ALE) techniques with structured local adaptive mesh refinement (AMR) has been developed for the solution of the Euler equations. The novel components of the method are driven by the need to reconcile traditional AMR techniques with the staggered variables and moving, deforming meshes associated with Lagrange based ALE schemes. Interlevel coupling is achieved with refinement and coarsening operators, as well as mesh motion boundary conditions. Elliptic mesh relaxation schemes are extended for use within the context of an adaptive mesh hierarchy. Numerical examples are used to highlight the utility of the method over single level ALE solution methods, facilitating substantial efficiency improvements and enabling the efficient solution of a highly resolved three-dimensional Richtmyer–Meshkov instability problem.

© 2004 Published by Elsevier Inc.

1. Introduction

The numerical simulation of unsteady compressible flows involving broad ranges of spatial and temporal scales is a computational challenge in many important application areas including astrophysics, inertial confinement fusion (ICF), and plasma physics. Resolution of localized small scale features such as shocks, mixing layers, and reaction zones requires a large number of computational cells in these regions. When solving problems including time-dependent boundaries, Lagrangian and ALE techniques have often been favored in the above application areas [1]. In addition to providing a natural mechanism for handling moving boundaries, an attractive feature of Lagrange based ALE methods is that they are, in a limited sense, inherently adaptive, i.e., cells cluster into high density regions behind shocks, and material discontinuities can be tracked intrinsically.

However, this form of adaptivity through mesh motion present in ALE methods, while an advantage over pure Eulerian codes in some applications, is less general and robust than a dynamically adaptive method in which the number of cells may change with time.

^{*} Corresponding author. Tel.: +925-424-2858; fax: +925-423-8704.
E-mail address: anderson110@llnl.gov (R.W. Anderson).

It is common in a wide variety of applications to be interested in the evolution of a simple initial condition, which can be described by relatively few degrees of freedom, into a system of increasing complexity with time, requiring increasing degrees of freedom in the discrete solution vector to maintain a desired level of accuracy. This is the fundamental motivation for the introduction of a dynamic spatio-temporal description of the solution into the class of ALE methods described herein. An approach which has proven effective in providing the kind of dynamic solution description that we desire is structured grid local adaptive mesh refinement (AMR) [2–5].

Various forms of adaptive mesh refinement on moving grids have previously been investigated. Bell et al. [6] have developed AMR for moving quadrilateral grids for a gasdynamics method starting from a collocated Eulerian formulation. Extensive work has also been done on in the finite element formulation, typically on unstructured meshes [7]. The main distinguishing element of this work is the use of a staggered variable, structured grid Lagrangian method as the underlying approach to the physics of the problem.

Extension of the classical AMR ideas to Lagrangian and ALE solution techniques is nontrivial for several reasons. The first is that typical AMR solution techniques are based on cell-centered discretizations, in contrast with the class of staggered mesh discretizations considered here that use a nodal description of mass element position and velocity, but cell-centered descriptions of thermodynamic quantities. The second source of additional complexity is that AMR methods typically utilize (nested) Cartesian meshes, while our Lagrange based method must consider moving, deforming meshes. Finally, our approach to ALE mesh motion will be of the “grid relaxation” variety, in which some smoothing operator will be applied to prevent mesh tangling which would otherwise occur as the mesh undergoes large deformation. Typical smoothing operators are based on elliptic equations, and the application of these smoothing operators must be extended to the case of a multi-level adaptive mesh hierarchy. The hierarchy advance scheme must successfully combine the requirements of both the hyperbolicity of the physics of the solution as well as the globally coupled elliptic component of the mesh evolution.

We will first review the single level ALE methods that form the basis for the adaptive method. Then, after reviewing some foundational AMR work, we extend those ideas into the context of staggered grid Lagrangian AMR. We then discuss the extensions required for integrating an elliptic mesh relaxation procedure into the method to arrive at the ALE-AMR method. We will then characterize the ALE-AMR method’s ability to generate solutions of essentially equivalent accuracy to single level ALE methods at a greatly reduced computational cost. Finally, we will demonstrate the utility of the method in generating solutions that would not be practical with a single level method for fixed computational resources.

2. Single level Lagrangian and ALE methods

The governing equations of inviscid gasdynamics for a single fluid over a Lagrangian volume V bounded by a moving surface S which, by definition, has a local velocity identical to the flow velocity are

$$\frac{d}{dt} \int_V \rho \, dV = 0, \quad (1a)$$

$$\frac{d}{dt} \int_V \rho \mathbf{u} \, dV = - \int_S p \mathbf{n} \, dS, \quad (1b)$$

$$\int_V \frac{de}{dt} \, dV = - \int_V p \frac{dv}{dt} \, dV, \quad (1c)$$

where ρ , e , p , \mathbf{u} , and $v = 1/\rho$ are the fluid density, internal energy, pressure, velocity, and specific volume, respectively, and t is time. An equation of state $p = p(\rho, e)$ closes the equation set. In this work we use $p = (\gamma - 1)\rho e$. Extensions to more general equations of state generate no fundamentally new considerations with respect to the adaptive method.

The ALE method employed for integration of the system (1) is of the explicit, time-marching, Lagrange plus remap type. Schemes of this type involve two distinct phases. In the first phase, a Lagrange step advances the flowfield through a physical time step. The optional second phase involves a modification of the grid and a remapping (interpolation) of the solution to the new grid. If the new grid is identical to the original grid prior to the Lagrange step, the net result is a fully Eulerian method. Typically, however, the new grid will be a “relaxed” grid that has been smoothed in some manner. This grid smoothing procedure is designed to alleviate the mesh tangling problem inherent in the Lagrangian methods for flows with regions of shear. The mesh relaxation algorithm employed here is essentially a Laplace iteration, and has its origins in the work of Winslow [8]. The subsequent solution interpolation procedure is formulated as an apparent advection problem, and is discussed in Section 2.1.

2.1. Summary of ALE methodology

A structured mesh composed of quadrilaterals in two dimensions and hexahedra in three dimensions is used to discretize (1). As an initial basis for constructing an ALE-AMR scheme, we will employ for the Lagrange step the general approach taken by Tipton [9], etc. We use a predictor–corrector discretization in time, and a staggered spatial discretization [10–12]. The scheme employs a monotonic artificial viscosity, q , due to Christensen [13], and a kinematic hourglass filter [14]. The two-dimensional scheme has been described extensively previously; algorithmic details as well as comparisons with more widely known Eulerian methods can be found in a recent work by Pember and Anderson [15]. Here we present those features of the algorithm which have particular relevance for the extension to AMR.

In particular, the staggered spatial discretization plays an important role in the design of the interlevel operators described in Section 3.2. Although some Lagrange schemes use a staggered discretization in time as well as space, we have chosen to employ a time-centered predictor–corrector scheme. Note that this dramatically simplifies the hierarchy advance algorithm.

The Lagrange predictor step begins with a calculation of the acceleration of each node, which is computed from a discretized (1b) by constructing a control volume around each node, which we will refer to as a “dual cell”. These dual cells are constructed from a “median mesh”, as shown in Fig. 1. The median mesh is generated by connecting the mid-points of the primary mesh, as shown. In 3D, this generalizes to tri-linear interpolation of the primary mesh.

The nodal acceleration is computed from $\vec{F} = \tilde{m}\vec{a}$, where \tilde{m} is a nodal mass and \vec{F} is a sum of net force contributions at the node. These force contributions come from the physical pressure p and the artificial viscosity q acting on the faces of the dual control volumes, as well as from the hourglass filter. A kinematic advance of nodal positions and velocities is then followed by an update of the energy in each zone corresponding to the associated $p dv$ work. The corrector then repeats these steps using time-centered values.

The nodal mass \tilde{m} is defined to be the average of the neighboring cell masses. This detail of the Lagrange method is of particular consequence for the adaptive method. This definition has important implications for the momentum conservation properties of the interlevel transfer operators described in Section 3.2.

A second detail of the Lagrange step which has consequences for the property of the adaptive method arises in 3D. As will be shown in Section 3.2, the definition of the hexahedral volume is central to the mass conservation properties of the AMR scheme. The best choice for consistency in the adaptive method is an exact integration of the volume of a tri-linear solid. An efficient formula for this calculation is given by Dukowicz [16]. Additional details regarding the properties of the tri-linear mapping and the computation of its volume may be found in [17].

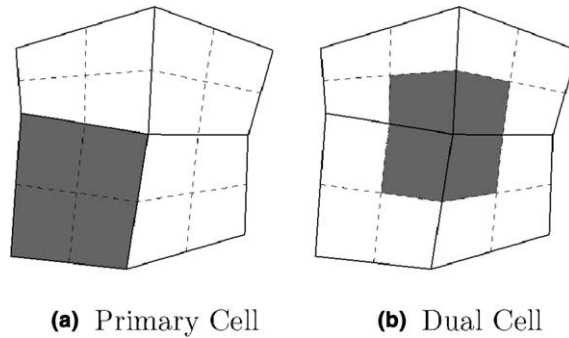


Fig. 1. Primary (a) and dual cells (b) are composed of 2^d corners as shown for $d = 2$.

Our Lagrange scheme conserves mass and momentum in all cases, and energy in 1D. However, in multi-dimensions, it does not conserve energy exactly. An example of a similar but energy conserving Lagrange step is given by Caramana et al. [18]. An advantage of our Lagrange scheme is that it lends itself to a relatively straightforward advection based remap. Moreover, our remap couples well with our adaptive method because it is both conservative and constant field preserving, and is used as one component of the interlevel operations.

At the end of a Lagrange step, it is often desirable to smooth the grid to prevent excessive mesh distortion which can lead to inaccuracy or even failure of the Lagrangian algorithm. An effective smoothing algorithm can be based upon a Laplace iteration for the transformed coordinates with respect to the Cartesian coordinates of each node. This type of equipotential method [19,20] is briefly discussed here to introduce the relevance for the resulting AMR scheme.

Given a transformation function for the node locations from a logical space

$$\mathbf{x} = f(\xi) \tag{2}$$

we can construct an iteration from the equation

$$\frac{\partial^2 f}{\partial \mathbf{x}^2} = 0. \tag{3}$$

While there are a multitude of potential motivations for the choice of (3), one perhaps instructive observation is that in 2D, a five-point stencil for the Laplace operator is an expression of the condition that every point be an average of its nearest neighbors, and therefore a solution of (3) is, in the accordant sense, optimally “smooth”.

While evaluation of an equation such as

$$\frac{\partial^2 \hat{f}}{\partial \xi^2} = 0, \tag{4}$$

where $\xi = \hat{f}(\mathbf{x})$, is more straightforward, in practice it is clear that the results are not as well behaved as what is obtained from (3), and it is also not unexpected, as we wish to smooth with respect to the \mathbf{x} reference frame, not the ξ generalized coordinates [20].

In order to construct an explicit iteration for the positions of nodes, we must invert (3) by expanding derivatives in a chain rule. After some lengthy computation, we can derive an expression for the central point in terms of its nearest neighbors along with some metric terms which act as weights on the stencil [19,20]. This expression is then iterated to smooth the mesh, either by a user-controlled number of iterations, or until some geometric criteria of the mesh are satisfied.

While the method can be generalized in a number of ways, e.g., the addition of weights, such generalizations should introduce no additional difficulties over the basic method when employed in the adaptive context. The key consideration is that we will be introducing an elliptic operator into the AMR hierarchy advance, which is traditionally designed for hyperbolic systems exhibiting local physics. In the case of ALE-AMR, we must accommodate both the local physics of the Lagrange step as well as the globally coupled mesh relaxation procedure in our advance algorithm.

Once the relaxed mesh has been defined, it remains to remap, or interpolate, the solution from the original Lagrange grid to the relaxed grid. (In this paper, we use the term remap for an interpolation process of this kind between meshes of logically equivalent resolution, and use the terms refinement and coarsening for interpolation between meshes of differing logical resolution, i.e., between different levels in the hierarchy.) A convenient way to formulate this remapping problem is in terms of an apparent advection equation, which puts at our disposal a wide variety of well-proven solution methods, from which we choose to implement a variant of the Corner Transport Upwind (CTU) method [21] which extends to 3D [22], as applied to the advection equation [6] on a staggered grid [15]. The staggered grid formulation is detailed in [15], and those same ideas extend to 3D straightforwardly. We typically remap ρ , \mathbf{u} , and E , where E is a total energy formed from the cell-centered internal energy e and an average of the kinetic energies of the surrounding nodes. When E is remapped, e is constructed from the remapped E and \mathbf{u} fields. Alternatively, e can be remapped directly if one wishes to prioritize monotonicity or positivity of internal energy, particularly in high-speed flows when the kinetic energy may dominate the total energy.

3. ALE-AMR algorithm

3.1. AMR overview

The conceptual starting point for the AMR methodology development is the pioneering work of Berger et al. [3,4]. In this approach, a hierarchical grid structure is employed which changes dynamically in time, and is composed of logically rectangular, uniform grid “patches” of varying resolution. In the original work, the grid hierarchy is constructed so that a coarse grid cell is covered precisely by r^d fine grid cells, where r is called the refinement ratio. The solution is defined on all cells, including coarse cells which underlay cells of finer resolution. The collection of grid patches at a given resolution is referred to as a level, notated as L_θ , where θ is a level number counting from 0 at the coarsest level.

An explicit time-marching method of a general hierarchy of θ_{\max} levels of refinement can be expressed as a recursive procedure beginning with the coarsest level $\theta = 0$:

```

repeat
  construct necessary  $L_\theta$  bc's by interpolating  $L_{\theta-1}$ 
  advance  $L_\theta$  to  $\min(t_\theta + \Delta t_\theta, t_{\theta-1})$ 
  if  $\theta < \theta_{\max}$  then
    recurse with  $\theta = \theta + 1$ 
    synchronize  $L_{\theta+1}$  and  $L_\theta$ 
    optionally regrid  $L_{\theta+1}$  through  $L_{\theta_{\max}}$ 
  end if
until  $t_\theta = t_{\theta-1}$ 

```

where Δt_θ is a stable time step for L_θ , and t_{-1} is the desired end simulation time. Typically $\Delta t_\theta \approx r \Delta t_{\theta+1}$, resulting in a nesting or “subcycling” of time steps for the levels in the hierarchy. See [2–4] and the review in [5] for more details. Note that in our “synchronize” step we replace coarse grid data by coarsened fine grid wherever finer data exist in the hierarchy.

We will construct the necessary interlevel solution transfer operators first for a purely Lagrangian method for which the above procedure can remain essentially unchanged, and then extend the ideas into the ALE-AMR method, which requires the introduction of some fundamental modifications to the above algorithm to accommodate mesh relaxation. We describe the algorithm in 3D, the restrictions to 2D and 1D are straightforward.

3.2. Lagrangian AMR

Interlevel solution transfer operators fall into two classes, refinement operators for defining fine grid data, and coarsening operators for the reverse. Refinement operators are required when new grids are created and for the generation of boundary conditions on finer levels in the hierarchy. Coarsening operators are required for synchronizing coarse and fine data in the hierarchy. Note that data here include the nodal positions themselves.

The refinement and coarsening operators presented here are designed with the following properties in mind:

- (P1) Constant field preservation.
- (P2) Second order accuracy (in smooth regions).
- (P3) Monotonicity.
- (P4) Local conservation.
- (P5) Exact inversion of refinement by coarsening.

A simple way to ensure that P5 is easily achieved is to maintain an exclusive $r:1$ correspondence between fine and coarse node data locations. In this case, any conservative distribution of a quantity from the coarse mesh to its corresponding fine mesh stencil may be inverted exactly with a simple summation. This is only achieved for *both* cell- and node-centered quantities by choosing r odd, as shown in 1D in Fig. 2.

3.2.1. Refinement

Solution refinement begins with linear mesh interpolation in each coordinate direction. The solution scalars are then interpolated onto the finer mesh using a series of linear interpolations utilizing either volume or mass coordinates in each coordinate direction. We first describe the general interpolation form, and then the particular sequence of interpolations appropriate for our particular discretization.

Consider a 3D coarse element (cell or node) which has a mean value ϕ^\vee , with corresponding limited differences $\delta^\alpha \phi^\vee$ in each logical direction. From here on we use superscript \vee and \wedge to indicate coarse and fine mesh quantities, respectively. Here and in the following discussion α is an arbitrary coordinate direction i, j , or k . We will notate using a constant refinement ratio r , but all constructions trivially extend to a vector of ratios to allow for different ratios in each coordinate direction.

We can construct a discrete interpolant of r^3 values $\phi_{l,m,n}^\wedge$, where l, m, n are local indices varying from 1 to r , onto the finer mesh using a local coordinate basis X as

$$\phi_{l,m,n}^\wedge = \phi^\vee + \delta^i \phi^\vee \left(\frac{\bar{X}_l^{i,\wedge}}{X^\vee} - \frac{1}{2} \right) + \delta^j \phi^\vee \left(\frac{\bar{X}_m^{j,\wedge}}{X^\vee} - \frac{1}{2} \right) + \delta^k \phi^\vee \left(\frac{\bar{X}_n^{k,\wedge}}{X^\vee} - \frac{1}{2} \right),$$

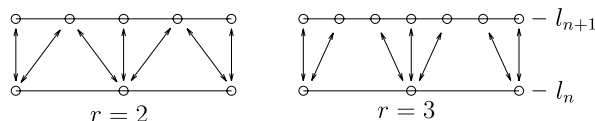


Fig. 2. Odd refinement ratios are required to maintain an $r:1$ correspondence between fine nodes and their corresponding coarse nodes in any given nodal stencil.

where the $\bar{X}_\beta^{\alpha,\wedge}$ is a fine mesh discrete mean coordinate in direction α , and β is the corresponding l, m , or n index corresponding to i, j , or k , respectively. This fine mesh coordinate is constructed from the increments $X_\beta^{\alpha,\wedge}$ as

$$\bar{X}_\beta^{\alpha,\wedge} = \sum_{\beta'=1}^{\beta-1} X_{\beta'}^{\alpha,\wedge} + \frac{1}{2} X_\beta^{\alpha,\wedge}.$$

The $X_\beta^{\alpha,\wedge}$ are accumulated from the fine basis $X_{l,m,n}^\wedge$ as

$$X_l^{i,\wedge} = \sum_{m=1}^r \sum_{n=1}^r X_{l,m,n}^\wedge$$

with cyclic expressions for the other coordinate directions.

This construction for ϕ^\wedge is conservative of ϕX in the sense that

$$\sum_{i,j,k} \phi^\wedge X^\wedge = \phi^\vee X^\vee$$

provided that the *basis consistency condition*

$$\sum_{i,j,k} X^\wedge = X^\vee \tag{5}$$

holds. This relationship will be shown to be fundamental to a number of important properties of the interlevel operators.

In a constant field, all differences $\delta_x \phi^\vee$ are zero, and constant fields are preserved provided (5) holds. In order to address P3, we employ the well-known van Leer limiter for computing the differences. If we desire to prevent oscillations in the primitive variables $\phi = (\rho, u, v, w, E)$, where E is the total energy and u, v, w are the components of velocity, the required interpolation basis to obtain property P4, local conservation, is $x = (V, \tilde{m}, \tilde{m}, \tilde{m}, m)$, where V is cell volume, \tilde{m} is nodal mass, and m is cell mass. Internal energy is constructed, as in the remap procedure, from the interpolated total energy and kinetic energy of the interpolated velocity field. It is also possible to interpolate the internal energy e directly if monotonicity of internal energy is of greater concern than exact conservation of total energy.

The sequence of interpolations then is to first interpolate density using cell volumes, compute the resulting fine mesh cell and nodal masses, and then use those bases to compute interpolated energies and velocities.

We now face a significant problem in reconciling the design goal P4 in the case of momentum, with the relationship between cell and nodal mass. Nodal mass, defined as an average of neighboring cells, does not meet the basis consistency condition (5) on the fine mesh after a cell-centered interpolation of density.

In 1D, intuitively it can be seen that such a nodal mass definition is equivalent to a definition which prescribes a constant density profile in each cell, and integrates those profiles over the control volume bounded by the median mesh surrounding each node. However, to satisfy P2, we interpolate cell densities using non-zero slopes, which is a fundamentally incompatible formulation. One can readily see that non-zero slopes in a cell serve to distribute mass unequally to the fine nodal stencils corresponding to the neighboring nodes of the coarse cell; whereas on the coarse mesh, that distribution is assumed to be equal by definition. In multi-dimensions, even a zero slopes integration of density over corner volumes is inconsistent with the averaged mass definition, and will suffer the same incompatibility problem.

One way to reconcile this inconsistency is by interpolating an adjusted coarse velocity field

$$u' \sum \tilde{m}^\wedge = u^\vee \tilde{m}^\vee + \sum u^* \delta m, \tag{6}$$

where, borrowing from remap terminology, u^* is an “edge state” and δm is a “transport mass” on the coarse grid. If the transport masses are constructed such that

$$\sum \tilde{m}^\wedge = \tilde{m}^\vee + \sum \delta m,$$

and edge states are constructed such that all $u^* = u_0$ in a constant coarse grid velocity field, then this adjusted field both conserves momentum and preserves a constant field upon interpolation to the fine mesh. This adjustment procedure has a natural interpretation as a nodal flux of momentum due to the implied mass transport across the median mesh by the imposition of cell density slopes in the mass interpolation step. We find that an arithmetic average for the edge state is sufficient for a variety of test cases, although one can apply upwinding based on the sign of the mass flux as an alternative. A minor concession with this procedure is condition P5, a loss of precise invertibility.

3.2.2. Coarsening

During the fine level advance, nodes on the fine level move unconstrained by the coarser levels except on the interlevel boundary. Therefore, coarsening begins with injection of nodal position, i.e.,

$$\mathbf{x}_{i,j,k}^\vee = \mathbf{x}_{r_i,r_j,r_k}^\wedge.$$

For the solution scalars, by choosing r odd, we have enabled particularly simple coarsening operators which are local weighted sums of conserved quantities, to satisfy both P4 and P5. The basis consistency condition (5) is again a central issue. Unless the condition is satisfied, there are two choices for a coarsening operator

$$\phi^\vee = \sum_{i,j,k} \phi^\wedge X^\wedge / \sum_{i,j,k} X^\wedge \tag{7}$$

or

$$\phi^\vee = \sum_{i,j,k} \phi^\wedge X^\wedge / X^\vee. \tag{8}$$

One can readily see that (7) is the constant field preserving choice, and (8) is the conservative choice, but not vice versa. We coarsen the primitive variables using the same bases used for interpolation. We also coarsen the same energy (total or internal) that we have chosen to refine.

After one or more Lagrangian advances and the initial nodal injection, the fine and coarse meshes will no longer be spatially aligned at intermediate nodes and/or edges, as shown in Fig. 3. Indeed, if by some artificial constraint this were not the case, the AMR method would not be providing the additional degrees of freedom in the solution we desire on finer levels. One can see that the consistency condition in the case of volume, for example, will not in general be satisfied after one or more Lagrange steps, i.e., the sum of the fine volumes corresponding to an underlying coarse cell will not equal the coarse volume. In order that the coarsening operator be both constant field preserving and conservative, we construct, as an intermediate step in the coarsening process, a remapped solution from the Lagrangian fine mesh to a fine mesh which is constructed from linear interpolation from the underlying coarse mesh. The interpolated fine mesh satisfies the volume consistency condition by construction. Since the remap procedure is itself (globally) conservative, and the consistency condition is satisfied, this alternative coarsening procedure is also conservative. It is important to note that this remapped solution is a temporary construction for use in the coarsening procedure, and does not replace the original fine solution for the next time step on the fine grid. This procedure concedes strict locality of conservation due to the fluxes from the remap procedure.

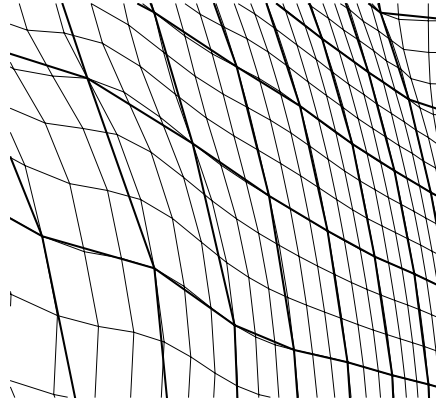


Fig. 3. Lagrange grid fine-coarse misalignment under deformation. Heavy lines are coarse mesh, thin lines are fine mesh.

3.2.3. Interlevel boundary conditions

Boundary conditions on finer grids in the hierarchy require careful treatment. The principle consideration is the spatial synchronization of the coarse and fine boundary nodes which will in general not stay aligned without special treatment. We have chosen to linearly interpolate, first in time, if necessary, and then in space, the positions of boundary nodes from the next coarser level, and employ the refinement operators developed in the previous section for all other quantities in ghost regions. This is motivated by a desire to always have quad or hex elements on a composite mesh. If boundary node positions were integrated according to the numerical scheme rather than imposed, then in general there would form non-quad or non-hex elements, as shown in 2D in Fig. 4.

This procedure is no compromise as long as the adaption criteria are sufficiently well designed that the coarse mesh solution alone is of acceptable accuracy in the vicinity of coarse-fine mesh boundaries.

3.3. ALE-AMR

The introduction of a mesh relaxation operator of the type described in Section 2.1 necessitates a fundamental change to the AMR hierarchy integration procedure as outlined in Section 3.2 to accommodate the intrinsic global coupling of the solution for the relaxed mesh. The basic requirement is that coarse meshes cannot be relaxed independently of finer meshes in the hierarchy. The key idea is that *we must defer the relaxation of a level's grids until all finer levels have been advanced to the same simulation time*. While it may seem natural to simply define a new operator that combines the Lagrange and remap steps into a composite step, it turns out that such an algorithm fails, as it decouples the coarse mesh relaxation from finer meshes, leading to ill-behaved mesh motion on the composite grid.

We define a “level grid” g_θ as the set of all nodes on L_θ , and a “composite grid” g_θ^c as

$$g_\theta^c = g_\theta \setminus I(g_{\theta+1}) \cup g_{\theta+1}^c. \quad (9)$$

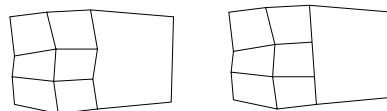


Fig. 4. On the left, “free” fine boundary nodes introduce non-quad elements. On the right, interlevel boundary conditions interpolate positions to preserve quad elements.

where I is a restriction operator of nodal injection (Section 3.2.2), and the definition $g_{\theta_{\max}}^c = g_{\theta_{\max}}$ closes the recursion.

An ALE-AMR algorithm is then

```

repeat
  construct necessary  $L_\theta$  bc's by interpolating  $L_{\theta-1}$ 
  advance  $L_\theta$  Lagrangian to  $t = \min(t + \Delta t, t_{\theta-1})$ 
  if  $\theta < \theta_{\max}$  then
    recurse with  $\theta = \theta + 1$ 
  repeat
    relax  $g_\theta^c$ 
  until  $g_\theta^c$  is sufficiently smooth
  remap levels  $\theta$  to  $\theta_{\max}$ 
  synchronize levels  $\theta$  through  $\theta_{\max}$ 
  optionally regrid levels  $\theta + 1$  to  $\theta_{\max}$ 
  end if
until  $t = t_{\theta-1}$ 
  
```

The logical diagram in Fig. 5 visualizes the process for a 3-level hierarchy.

If a strictly Eulerian calculation is desired, the stability constraint of the remap procedure can require multiple cycles of relaxation to return the mesh to its original configuration. An alternative which is typically more efficient is to forgo time refinement, and advance all levels by a global minimum time step. In this case subcycling of the remap on finer grids will not be required to remap to the original grid, if that time step is appropriately limited. Since time refinement adds a single factor of $1/r$ to the work ratio on coarser grids, spatial refinement tends to dominate the work savings associated with a multi-dimensional AMR scheme, and little efficiency advantage is lost relative to the fully refined calculation.

3.3.1. Multi-level mesh relaxation

The remaining key issue is to construct a relaxation operator for a composite grid g_θ^c . Define

$$g_\theta^* = g_\theta \setminus I(g_{\theta+1}), \tag{10}$$

the union of which for all θ is the composite grid.

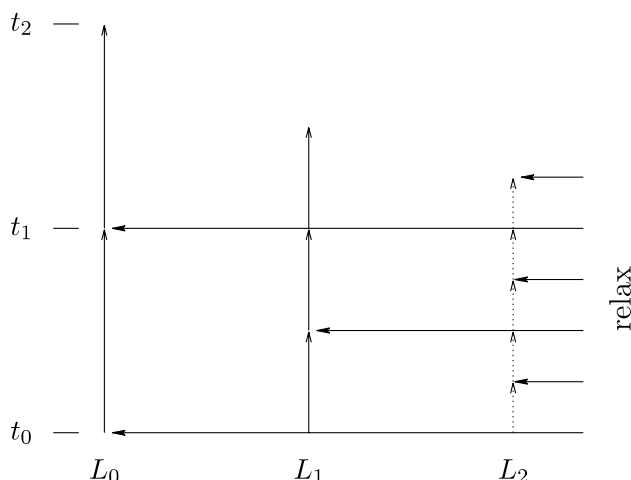


Fig. 5. ALE-AMR hierarchy integration logical diagram, shown for a 3-level hierarchy. Vertical lines are Lagrange steps, horizontal lines are relaxation operations.

A design principle of the composite mesh operator R is

$$\text{if } R(g_\theta^*) = g_\theta \quad \forall \theta, \quad \text{then } R(g_0^c) = g_0^c. \quad (11)$$

Intuitively, a uniform mesh at each level should remain uniform, as opposed to letting the finer meshes expand, which would occur if we applied some type of unstructured or semi-structured discretization of (3) to the composite mesh directly. In order to get the desired behavior (11), the discretization must respect the per-level index space of each node in the composite mesh, with the interlevel coupling occurring through injection and interpolation of boundary conditions between the g_θ^* . Each iteration of $R(g_\theta^c)$ consists of

```

for  $m = \theta_{\max}$  to  $\theta$  do
    constructed interpolated boundary conditions around  $g_m$  from  $g_{m-1}$ 
    relax  $g_m^*$ 
    inject  $g_m$  to  $g_{m-1}$ 
end for

```

3.4. Implementation

The implementation utilizes SAMRAI [23], an object-oriented framework for the development of structured grid adaptive mesh refinement applications. The framework has been extended to accommodate many of the novel or unusual AMR features developed in the current work. The SAMRAI framework is a C++ library, and the application code was developed using both C++ and FORTRAN 90, with FORTRAN 90 being reserved for performance of critical inner loop constructs. We have found this dual language choice to be an effective paradigm for scientific calculation when the algorithms and data structures are of sufficient complexity to warrant the abstraction mechanisms provided by the C++ language.

4. Numerical results

A successful AMR extension of a given method is one which provides solutions as accurate as the single-grid formulation at a reduced computational cost. It is important to separate the issues associated with this property of the AMR extensions in particular, and larger issues relating to the characteristics of the single level method. With this in mind we seek to characterize the performance of the ALE-AMR method, as well as demonstrate the potential for improving the state of the art in high resolution 3D ICF calculations.

The criteria used for the selection of refined regions are constructed from normalized second differences

$$C_x = \left\| \frac{\phi_{x+1} - 2\phi + \phi_{x-1}}{\max(\phi_0, \epsilon)} \right\|_{L^2}, \quad (12)$$

where ϵ is a small constant to prevent blowup, and ϕ_0 is a global normalization constant. In practice, we define two criteria separated by a tolerance of a few percent. The smaller of the two values is required to untag a refined cell, and the larger to tag for refinement an untagged cell. This helps prevent “chatter” of refining and de-refining when the criterion is near the threshold. The scalar ϕ is usually evaluated both from the pressure and density with an “or” operation on the result of the evaluation. Pressure criteria alone can miss contact surfaces, and density differences alone often miss critical regions of large acceleration. While additional work into refinement criteria is warranted, we find that this simple heuristic criterion seems to work quite well in practice. For all of the following calculations, the refinement ratio is chosen to be 3.

4.1. Relative accuracy and efficiency characterization

In order to evaluate the adaptive Lagrangian-only method, we begin with a one-dimensional interacting blast wave problem due to Woodward and Colella [24]. In one dimension we can solve non-trivial problems in a Lagrangian manner without mesh tangling difficulties. The initial conditions are

$$p = \begin{cases} 1000, & 0 < x < 1/10, \\ 0.01, & 1/10 < x < 9/10, \\ 100, & 9/10 < x < 1 \end{cases} \tag{13}$$

with everywhere unity density and zero velocity, and $\gamma = 1.4$. The boundary conditions are reflecting walls at both ends.

In Fig. 6, the results at $t = 0.038$ are compared between three solutions. The two dashed line solutions are 300 and 900 cell single level Lagrangian calculations. The coarse solution is clearly underresolved. The solid lines represent the adaptive solution, and the presence of the symbols shows the final location of the refined grid regions. The fine and adaptive solutions are essentially indistinguishable.

To quantify the trade-off between relative accuracy and efficiency, we can compute normalized L1 norms

$$\frac{\|\rho - \rho_{\text{ref}}\|}{\|\rho_{\text{ref}}\|} = \frac{\int |\rho - \rho_{\text{ref}}| \, dV}{\int \rho_{\text{ref}} \, dV} \tag{14}$$

of difference vectors between the adaptive solutions ρ and fully resolved calculations ρ_{ref} , as well as measure CPU execution times on a 1.5 GHz Intel Xeon workstation. By relative accuracy we mean the accuracy with which the adaptive solution replicates a solution using fine grid everywhere, as quantified by the magnitude of the mass error (14).

Since we are computing the difference of solutions on two grids which can differ both from mesh motion and with respect to resolution of AMR levels, we employ the following method for computing the L1 norm

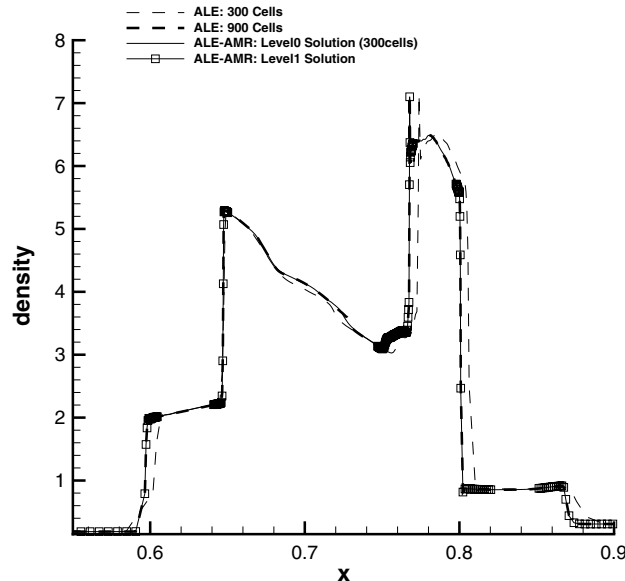


Fig. 6. 1D Colella–Woodward blast wave solutions at $t = 0.038$ s. The adaptive solution has a coarsest resolution of 300 cells and two refinement levels, and is essentially indistinguishable from the fully resolved solution with 2700 cells.

of the difference solution (14). The solution on both grids is reconstructed from discrete data using piecewise constant functions per cell. Since all differences are computed with respect to a solution which is entirely at the finest level, we use this grid as the reference grid, and perform numerical quadrature using its cells as quadrature regions. For each cell on the reference grid, its centroid is used to locate a corresponding cell on the other adaptive grid to provide a difference value. Since this is not in general an exact integration of the difference of the reconstructed functions due to cell overlap, we repeat the procedure for a refined reference grid using the same piecewise constant reference solution, iterating the refinement until the norm is stationary in three significant digits.

Fig. 7 lists norms which represent relative mass error for a matrix of calculations with varying numbers of adaptive grid levels and effective resolutions. By effective resolution, we mean the resolution at the finest level of the calculation. Consider the 2-level solution at an effective resolution of 8100 cells. The difference vector norm shown is always relative to the single level calculation at 8100 cells, and for this case is of order 10^{-3} . To provide a scale for the significance of that relative error, we can compare to a calculation at the next coarser resolution on a single level, which produces a relative error of the order 10^{-2} , an order of magnitude greater. This result should be weighed against the CPU cost of each solution, as listed in Fig. 8. Note that the computational savings associated with advancing coarsened cells in the adaptive algorithm is proportional to r^{d+1} ; for a given proportion of finest grid, the efficiency benefits tend to increase with increasing spatial dimension.

A similar exercise, this time utilizing relaxation and remap, for the 2D Taylor–Sedov blast wave produces the results shown in Figs. 9 and 10. The initial conditions for this problem consist of a no flow, uniform density field. The energy is everywhere zero except at the origin. We compute one quadrant of the symmetric solution, using an initial energy of 8, $\gamma = 5/3$, and $\rho = 1$. The AMR speedup increases to about 16 for this problem, even though we are covering more of the domain with fine grid, due to the improved efficiency associated with the work scaling factor r^{d+1} . An example 3-level solution is shown in Fig. 11.

An alternative way to view the efficiency of the adaptive method is to compare the actual error (relative to exact solution) reduction rate as a function of increasing CPU cost of additional resolution, for both single level ALE and ALE-AMR methods. The resulting trendlines for mass, momentum, and energy errors for a matrix of 2D Taylor–Sedov blast wave calculations are shown in Fig. 12. The momentum and energy errors are computed analogously to (14), using the momentum magnitude and total energy in place of density. Each calculation was performed on 8 processors beginning with an initial resolution of 81^2 cells,

$(\ \rho_N - \rho_{8100}\) / \ \rho_{8100}\ $				
	Effective Resolution			
Levels	8100	2700	900	300
1	0	.010546	.030476	.1035780
2	.001605	.011424	.039504	-
3	.004807	.020680	-	-
4	.015228	-	-	-

Fig. 7. $\|\rho_N - \rho_{8100}\| / \|\rho_{8100}\|$ for Colella–Woodward blast wave problem for a matrix of gridding strategies. “Effective resolution” is the finest resolution occurring in the calculation. ρ_{8100} is the solution for the maximum resolution single level calculation, and ρ_N is the solution for the N -level adaptive calculation.

Speedup (t_{8100}/t_n)				
	Effective Resolution			
Levels	8100	2700	900	300
1	1	10.1	89.5	760
2	4.6	25	72	-
3	7.0	18	-	-
4	5.3	-	-	-

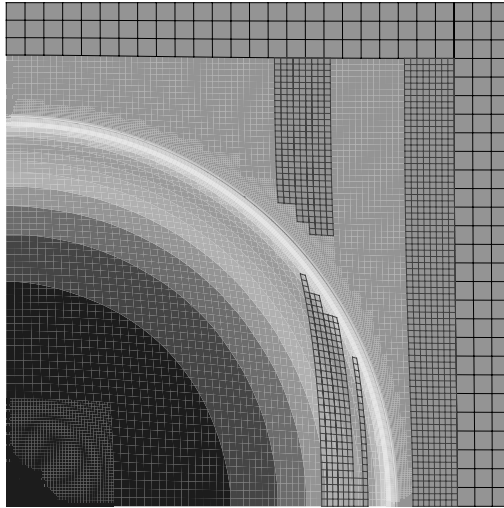
Fig. 8. Speedup for Colella–Woodward blast wave problem.

$(\ \rho_N - \rho_{729}\) / \ \rho_{729}\ $				
	Effective Resolution			
Levels	729^2	243^2	81^2	27^2
1	0	.00610	.0254	.0711
2	.00184	.00570	.0246	-
3	.00290	.00617	-	-
4	.00362	-	-	-

Fig. 9. $\|\rho_N - \rho_{729}\| / \|\rho_{729}\|$ for the Taylor–Sedov blast wave problem for a matrix of gridding strategies. “Effective resolution” is the finest resolution occurring in the calculation. ρ_{729} is the solution for the maximum resolution (729×729) single level calculation, and ρ_N is the solution for the N -level adaptive calculation.

Speedup (t_{729}/t_n)				
	Effective Resolution			
Levels	729^2	243^2	81^2	27^2
1	1	32.3	858	17900
2	11.3	159	2250	-
3	16.4	180	-	-
4	12.1	-	-	-

Fig. 10. Speedup for 2D Taylor–Sedov blast wave problem.



and twice increasing the resolution: by factors of 3^2 in the single level case, and adding a level with $r = 3$ in the ALE-AMR case. All calculations used the same parallel implementation. From this result it can be seen that the momentum and energy errors due to AMR are at least as well-behaved as the density errors tabulated in the other demonstrations.

The Eulerian capability of the algorithm is demonstrated on the double Mach reflection problem [21]. In this problem a Mach 10 planar shock in a $\gamma = 1.4$ gas impinges on a ramp at 60° , as shown in Fig. 13. The

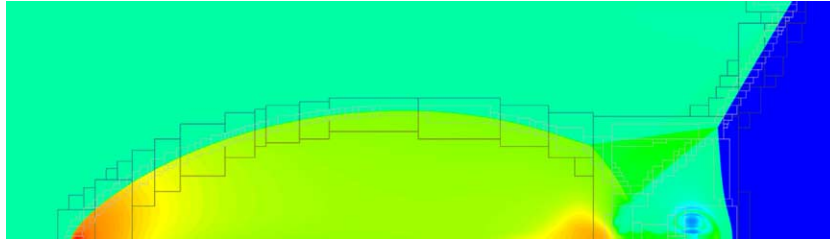


Fig. 13. Double Mach reflection of a strong shock. Dark outlines are regions of first level of refinement, lighter outlines are regions of second level of refinement.

preshock density and pressure are 1.4 and 1, respectively. The domain is 3.5 units long and 1 unit high. We choose a base resolution of 224×76 , and use two levels of refinement. The flow is self-similar and produces smaller-scale instability phenomena in the “wall jet” region, as shown in Fig. 14, at the time $t = 0.21$. It is for this kind of problem that is characterized by multiple, well-separated scales, for which AMR techniques excel.

A demonstration of a 3D Lagrangian capability is shown in Fig. 15 with one octant of the solution of a spherical Taylor–Sedov blast wave problem using a base resolution of $32 \times 32 \times 32$, and one level of adaptive refinement.

4.2. 3D Richtmyer–Meshkov instability in ICF configuration

To demonstrate the potential of the ALE-AMR algorithm in more demanding applications, we have run a proof-of-principle calculation of a Richtmyer–Meshkov instability in an idealized ICF implosion. In this

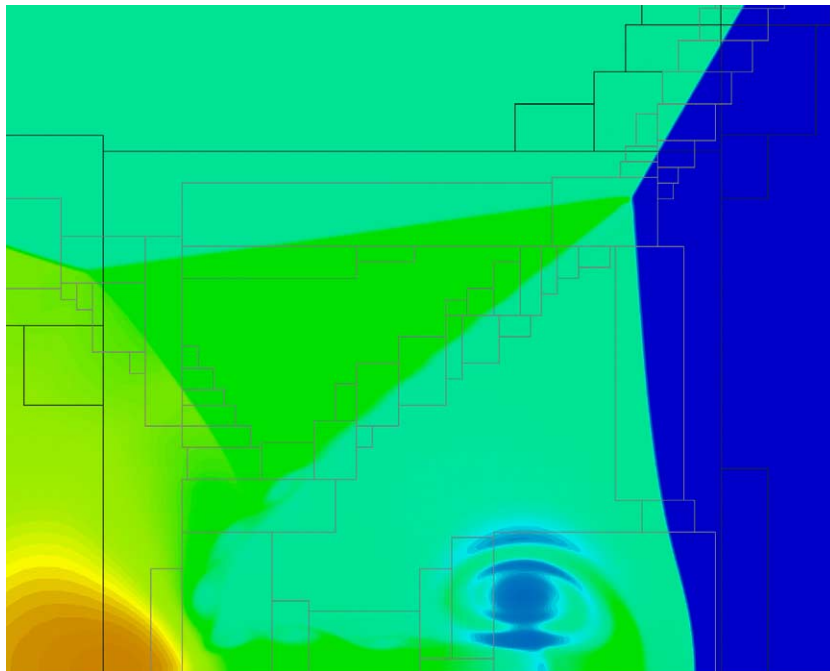


Fig. 14. Detail of “wall jet” region of double Mach reflection of a strong shock. Dark outlines are regions of first level of refinement, lighter outlines are regions of second level of refinement.

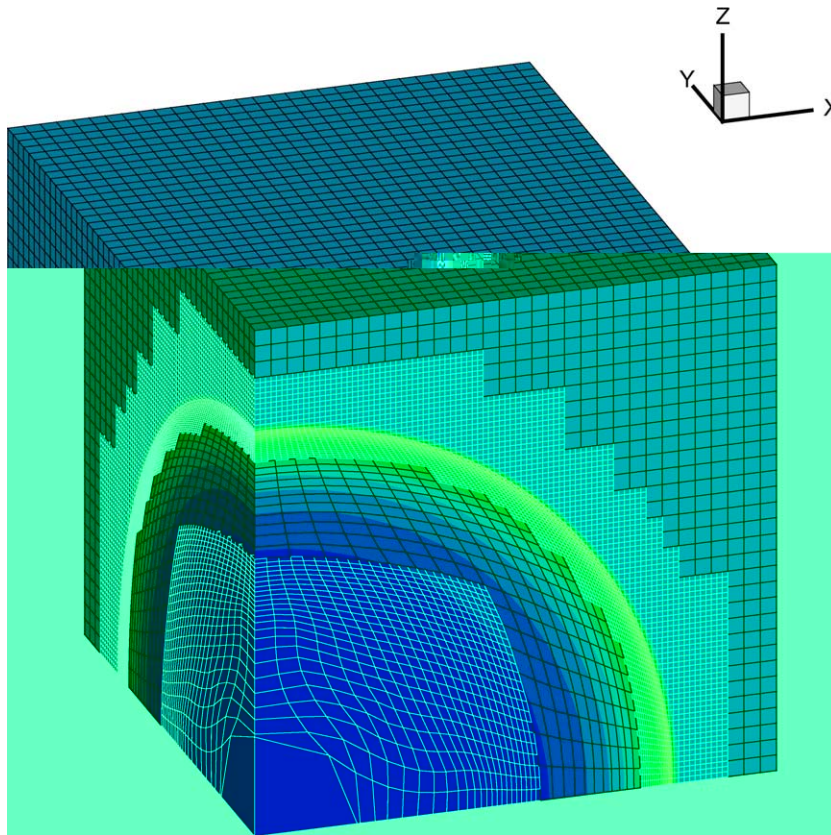


Fig. 15. 3D Taylor–Sedov blast wave solution with one level of refined grids.

model problem, we follow the unstable evolution of a single mode perturbation on the outer surface of the inner shell of a double-shell ICF target. The initial condition is modeled after the 250 eV NIF 670 kJ design described by Amendt et al. [25], and consists of an inner core of DT fuel with radius 343 μm , surrounded by an inner shell of Au with an outer radius of 397 μm , surrounded by foam with an outer radius of 1276 μm , surrounded by an outer shell of Be with an outer radius of 1601 μm . We model each of these materials with the initial densities from the original work, and a uniform $\gamma = 5/3$.

The initial conditions contain a high pressure region in the outer surface of the outer shell to model the effects of ablation by rapid laser energy deposition. Previous calculations using other models [25] have shown that approximately 0.67 MJ of energy is transferred to the hohlraum from 2.5 MJ of laser energy. We assume complete thermalization, and choose a pressure at constant density to match this energy deposition at $t = 0$.

Our computational domain is a “wedge” configuration bounded by four planes with a divergence angle sufficient to capture three wavelengths of the initial perturbation, which was taken to be 10% of the inner shell thickness in amplitude, with a wavenumber of approximately 80. Spherical symmetry boundary conditions are assumed on these planes. The calculation used four adaptive meshing levels with a base resolution of $800 \times 10 \times 10$, resulting in a finest resolution of $21600 \times 270 \times 270$, or 1.5 billion zones, if fine grid covered the entire domain. The finest mesh size is initially 1/3 μm , although with the ALE mesh motion it becomes significantly smaller in regions of high compression. Since the solution to this problem is

symmetric until the incoming shock impinges on the perturbation at approximately 5.52 ns, we ran a 1D calculation to generate initial conditions for the 3D simulation. After initializing the 3D flowfield using the 1D calculation dataset, the inner shell perturbation was overlaid to generate the full 3D initial conditions. Fig. 16 illustrates the range of scales under consideration in the calculation at the initial condition, from the largest associated with the outgoing shock to the smallest associated with the details of the perturbation of the inner shell.

This calculation was then run on 404 667 MHz EV67 processors of the LLNL TC2K Compaq cluster for 48 wall clock hours over 15,900 coarse time steps. The fluid-mechanical instability was well resolved, and the solution at the end of the calculation is shown in Fig. 17 in an isosurface of density and a cutting plane showing contours of density at 7.65 ns. The black boxes represent refinement block boundaries.

The addition of AMR to ALE based solution methods in the ICF application area appears to be essential as the state of the art moves forward toward 3D simulations that capture, rather than model, instabilities at decreasing scales. While it is clear that direct numerical simulation to the Kolmogorov scale is not attainable, it is an open problem to determine the range of scales which are sufficient to adequately resolve yield degradation due to mixing. AMR techniques allow researchers to continue to push further into highly resolved solution spaces by significantly flattening the growth curve of the required computational resources.

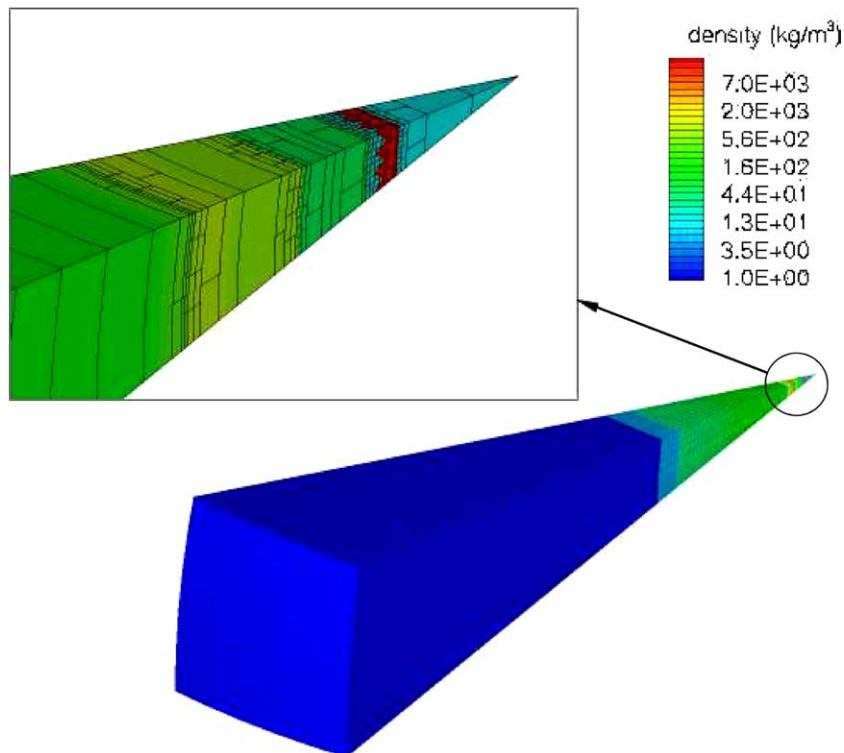


Fig. 16. Initial condition for 3D phase of calculation at $t = 5.52$ ns. Inset showing refinement blocks on initial perturbation of inner shell.

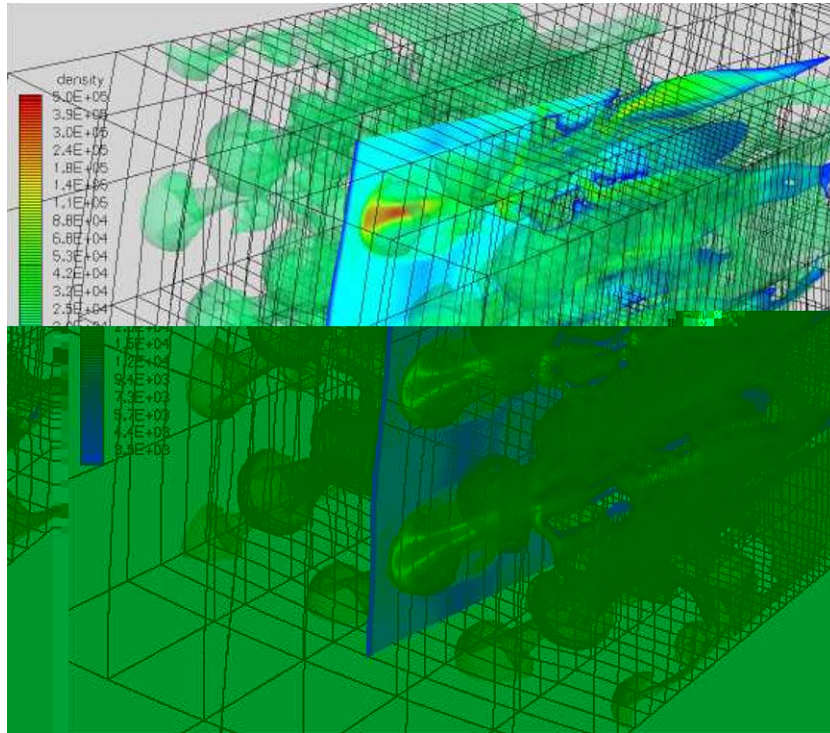


Fig. 17. Richtmyer–Meshkov instability growth at 7.65 ns. Isosurfaces of density. Black boxes represent refinement block boundaries. Cutting plane with colormap represents density. Density units on scale are kg/m^3 .

5. Conclusions

We have successfully combined the ALE and AMR methods to create a new, more powerful adaptive method for solution of the equations of inviscid gasdynamics. We have presented both quantitative and qualitative evidence that the method displays promise for improving computational efficiency without undermining the robustness or accuracy of the underlying ALE method. The adaptive components of the algorithm are applicable to a class of methods that employ staggered variables of the type described, and are not limited to the particular details of the hydrodynamical or mesh relaxation methods we have employed and presented in this work.

Significant efficiency increases have been demonstrated on simple test problems in 1D and 2D, and a highly resolved proof-of-principle ICF model problem was demonstrated in 3D.

In future work, we will pursue improvements in the base ALE-AMR methodology, in particular, in the areas of strategies to account for interlevel flux mismatches, discrete momentum and energy conservation, and refinement criteria. In addition, in order to further explore the potential of ALE-AMR based methods, we are currently adding capabilities relevant to the ICF application, including a radiation diffusion capability to model, rather than impose, the radiant energy deposition. Such additional physics capabilities open up new opportunities to exploit the advantages of AMR to enable the solution of problems at higher resolution, or to eliminate ad hoc or semi-empirical models and replace them with methods based on first principles.

Acknowledgements

This work was performed under the auspices of the US Department of Energy by University of California Lawrence Livermore National Laboratory under contract No. W-7405-Eng-48.

References

- [1] B.J. Benson, An efficient, accurate, simple ALE method for nonlinear finite element programs, *Comput. Meth. Appl. Mech. Eng.* 72 (1989) 205–350.
- [2] J. Bell, M. Berger, J. Saltzman, M. Welcome, Three dimensional adaptive mesh refinement for hyperbolic conservation laws, *SIAM J. Sci. Comput.* 15 (1994) 127–138.
- [3] M. Berger, J. Olinger, Adaptive mesh refinement for hyperbolic partial differential equations, *J. Comput. Phys.* 53 (1984) 484–512.
- [4] M. Berger, P. Colella, Local adaptive mesh refinement for shock hydrodynamics, *J. Comput. Phys.* 82 (1989) 64–84.
- [5] R. Pember, J. Bell, P. Colella, W. Crutchfield, M.L. Welcome, An adaptive Cartesian grid method for unsteady compressible flow in complex geometries, *J. Comput. Phys.* 120 (1995) 278–304.
- [6] J. Bell, P. Colella, J. Trangenstein, M. Welcome, Adaptive mesh refinement on moving quadrilateral grids, in: *Proceedings, AIAA 9th Computational Fluid Dynamics Conference*, Buffalo, New York, June 14–16, 1989, pp. 471–579.
- [7] R. Lohner, *Applied Computational Fluid Dynamics Techniques: An Introduction Based on Finite Element Methods*, Wiley, New York, 2001.
- [8] A. Winslow, Numerical solution of the quasilinear Poisson equation in a nonuniform triangle mesh, *J. Comput. Phys.* 1 (1967) 149–172.
- [9] R. Tipton, Unpublished report, Lawrence Livermore National Laboratory, 1990.
- [10] M.L. Wilkins, Calculation of elastic–plastic flow, *Meth. Comput. Phys.* 3 (1964) 211–263.
- [11] R. Sharp, HEMP advection model, Technical Report UCID-17809, Lawrence Livermore National Laboratory, 1978.
- [12] M.L. Wilkins, *Computer Simulation of Dynamic Phenomena*, Springer, Berlin, 1999.
- [13] R. Christensen, Godunov methods on a staggered mesh – an improved artificial viscosity, Technical Report UCRL-JC-105269, Lawrence Livermore National Laboratory, 1990.
- [14] L. Margolin, J. Pyun, A method for treating hourglass patterns, Technical Report LA-UR-87-439, Los Alamos National Laboratory, 1987.
- [15] R. Pember, R. Anderson, Comparison of direct Eulerian Godunov and Lagrange plus remap artificial viscosity schemes for compressible flow, Technical Report AIAA Paper 2001-2644, 2001.
- [16] J. Dukowicz, Efficient volume computation for three-dimensional hexahedral cells, *J. Comput. Phys.* 74 (2) (1988) 493–496.
- [17] O.V. Ushakova, Conditions of non-degeneracy of three-dimensional cells. A formula of a volume of cells, *SIAM J. Sci. Comput.* 23 (4) (2001) 1274–1290.
- [18] E.J. Caramana, D.E. Burton, M.J. Shashkov, P.P. Whalen, The construction of compatible hydrodynamics algorithms utilizing conservation of total energy, *J. Comput. Phys.* 146 (1) (1998) 227–262.
- [19] A. Winslow, Equipotential zoning of two dimensional meshes, Technical Report UCRL-7312, Lawrence Livermore National Laboratory, 1963.
- [20] R. Tipton, Grid optimization by equipotential relaxation, Lawrence Livermore National Laboratory, 1992.
- [21] P. Colella, Multidimensional upwind methods for hyperbolic conservation laws, *J. Comput. Phys.* 87 (1990) 171–200.
- [22] J. Saltzman, An unsplit 3-D upwind method for hyperbolic conservation laws, *J. Comput. Phys.* 115 (1994) 153–168.
- [23] R. Hornung, S. Kohn, Managing application complexity in the samrai object-oriented framework, *Concurrency: Practice and Experience* 14 (2002) 347–368.
- [24] P. Woodward, P. Colella, The numerical simulation of two-dimensional fluid flow with strong shocks, *J. Comput. Phys.* 54 (1984) 115–173.
- [25] P. Amendt, J. Colvin, R. Tipton, D. Hinkel, M. Edwards, O. Landen, J. Ramshaw, L. Suter, W. Varnum, R. Watt, Indirect-drive noncryogenic double-shell ignition targets for the national ignition facility: design and analysis, *Phys. Plasmas* 9 (5) (2002) 2221–2232.