

An Arbitrary Lagrangian-Eulerian Method with Local Structured Adaptive Mesh Refinement for Modeling Shock Hydrodynamics

R.W. Anderson, R.B. Pember, N.S. Elliott

This article was submitted to
40th American Institute for Aeronautics and Astronautics Aerospace
Science Meeting and Exhibit, Reno, NV, January 14-17, 2002

U.S. Department of Energy

Lawrence
Livermore
National
Laboratory

October 22, 2001

DISCLAIMER

This document was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor the University of California nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or the University of California. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or the University of California, and shall not be used for advertising or product endorsement purposes.

This is a preprint of a paper intended for publication in a journal or proceedings. Since changes may be made before publication, this preprint is made available with the understanding that it will not be cited or reproduced without the permission of the author.

This report has been reproduced directly from the best available copy.

Available electronically at <http://www.doc.gov/bridge>

Available for a processing fee to U.S. Department of Energy
And its contractors in paper from
U.S. Department of Energy
Office of Scientific and Technical Information
P.O. Box 62
Oak Ridge, TN 37831-0062
Telephone: (865) 576-8401
Facsimile: (865) 576-5728
E-mail: reports@adonis.osti.gov

Available for the sale to the public from
U.S. Department of Commerce
National Technical Information Service
5285 Port Royal Road
Springfield, VA 22161
Telephone: (800) 553-6847
Facsimile: (703) 605-6900
E-mail: orders@ntis.fedworld.gov
Online ordering: <http://www.ntis.gov/ordering.htm>

OR

Lawrence Livermore National Laboratory
Technical Information Department's Digital Library
<http://www.llnl.gov/tid/Library.html>

An Arbitrary Lagrangian-Eulerian Method with Local Structured Adaptive Mesh Refinement for Modeling Shock Hydrodynamics

R. W. Anderson*, R. B. Pember†, N.S. Elliott‡

Center for Applied Scientific Computing
Lawrence Livermore National Laboratory
P.O.Box 808
Livermore, CA, 94551

A new method that combines staggered grid Arbitrary Lagrangian-Eulerian (ALE) techniques with structured local adaptive mesh refinement (AMR) has been developed for solution of the Euler equations. This method facilitates the solution of problems currently at and beyond the boundary of soluble problems by traditional ALE methods by focusing computational resources where they are required through dynamic adaption. Many of the core issues involved in the development of the combined ALE-AMR method hinge upon the integration of AMR with a staggered grid Lagrangian integration method. The novel components of the method are mainly driven by the need to reconcile traditional AMR techniques, which are typically employed on stationary meshes with cell-centered quantities, with the staggered grids and grid motion employed by Lagrangian methods. Numerical examples are presented which demonstrate the accuracy and efficiency of the method.

1 Introduction

The numerical simulation of compressible flows with shocks and discontinuities is a computational challenge in many important application areas including aeroelasticity, inertial confinement fusion (ICF), astrophysics, and plasma physics. Resolution of small scale flowfield features such as shocks, material interfaces, and regions of instability requires a large number of computational cells in these regions. Lagrangian and ALE techniques have often been favored in the above application areas,¹ in part due to the self-adapting nature of Lagrangian grid motion, e.g., contact discontinuities are tracked automatically,

*Mail Stop: L-551. Phone: (925) 424-2858. E-mail: anderson110@llnl.gov

†Mail stop: L-560. Phone: (925) 422-4549. E-mail: pember@llnl.gov

‡Mail stop: L-560. Phone: (925) 422-4004. E-mail: eliott22@llnl.gov

Copyright © 2002 by the American Institute of Aeronautics and Astronautics, Inc. All rights reserved.

and cells are clustered into high density regions behind shocks. However, this inherent form of adaption present in Lagrangian and 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. When the number of cells in a structured grid is not variable, such as in traditional ALE methods, an increase in resolution in one region necessitates a decrease in resolution in some other region. Furthermore, deformation of structured grids suffers from limitations in the geometrical complexity that can be described, much in the same way that structured grids have limitations for complex geometries. Finally, efficiency is in general suboptimal with a statically adaptive grid, since resolution requirements may vary throughout the simulation time.

An approach which has proven effective in addressing these problems is structured grid local adaptive mesh refinement (AMR).²⁻⁵ AMR involves the addition and removal of mesh cells as required to maintain a specified level of accuracy, as opposed to maintaining a fixed cost. As illustrated in Fig. 1, this technique involves the successive refinement of regions of a structured mesh where additional resolution is required. Extension of the AMR idea to Lagrangian and in turn ALE solution techniques is nontrivial due to several factors. The first is that AMR was originally developed in the context of algorithms that employ cell-centered variables on Cartesian meshes. Lagrangian methods, by contrast, are often developed by utilizing a staggered grid, where thermodynamic quantities such as density and energy are located at cell centers, while the kinematic quantities of position and velocity are located at the mesh intersections, or nodes. This change, along with the non-Cartesian nature of a Lagrangian grid, necessitates new solution transfer operators between coarse and fine meshes. Furthermore, the methods responsible for synchronization of fine grid and coarse grid solutions must be modified to suit a Lagrangian context on a staggered grid. Finally, grid motion is a new component with respect to AMR methods, and the coupling between fine and coarse grid motion in the AMR hierarchy must be estab-

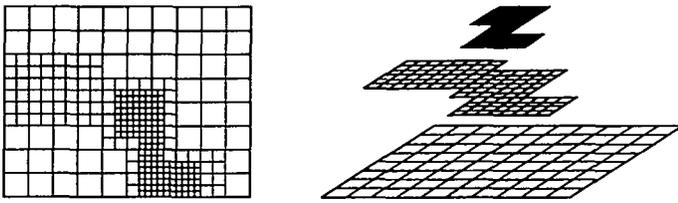


Fig. 1 An example of adaptive mesh refinement showing a properly nested hierarchy of grids.

lished. Although some work has been done with non-Cartesian structured grids in an Eulerian context,⁶ the methodology presented here is a fundamentally different approach.

2 Equations of Motion and the ALE Method

The governing equations of inviscid gasdynamics are discretized from the Lagrangian form:

$$\frac{D\rho}{Dt} + \rho \vec{\nabla} \cdot \vec{V} = 0 \quad (1)$$

$$\rho \frac{D\vec{V}}{Dt} + \vec{\nabla} p = 0 \quad (2)$$

$$\rho \frac{De}{Dt} + p \vec{\nabla} \cdot \vec{V} = 0. \quad (3)$$

where ρ , e , p , and \vec{V} are the fluid density, internal energy, pressure, and velocity respectively, and t is time.

The ALE method employed for integration of the system (1)-(2)-(3) 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 timestep. The second phase involves a modification of the grid and a remapping of the solution to the new grid. The modification may be to the original Eulerian grid, or it may be to a "relaxed" grid that has been smoothed in some manner. The grid remapping procedure alleviates the mesh tangling problem inherent in the Lagrangian methods for flows with vorticity. The mesh relaxation algorithm employed here is essentially a Laplace iteration, and has its origins in the work of Winslow.⁷ The solution interpolation procedure is formulated as an apparent advection problem, and is discussed in Section 2.3.

2.1 Lagrange Step

The Lagrange step follows the general approach taken by Tipton.⁸ It employs a predictor-corrector discretization in time, and the HEMP spatial discretization.^{9,10} The scheme employs a monotonic artificial viscosity due to Christensen,¹¹ and a kinematic hourglass filter.¹² 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, et al.¹³

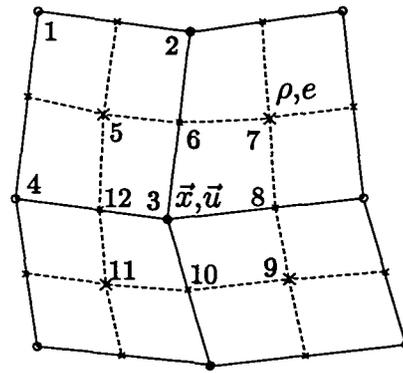


Fig. 2 Staggered Mesh and Control Volumes. Mass and energy control volumes 1 - 2 - 3 - 4 - 1 surround each cell, while momentum control volumes 5 - 6 - 7 - 8 - 9 - 10 - 11 - 12 - 5 surround each node.

The key feature of the Lagrange step with respect to the development of the AMR algorithm is its staggered mesh arrangement with density and energy stored per cell, while position and velocity are stored per node, as shown in Fig. 2. This choice leads to two control volumes, the first formed from the mesh quadrilaterals such as 1 - 2 - 3 - 4 - 1, and the second formed from the "dual mesh" octagons, such as 5 - 6 - 7 - 8 - 9 - 10 - 11 - 12 - 5. The dual mesh is constructed from segments with endpoints located at the midpoint of edges and cell centers. Following Tipton, Pember¹³ describes an optimization which constructs equivalent dual control volumes. However, in the interest of consistency with forthcoming three-dimensional extensions of the method, we do not exploit that optimization in this work.

The Lagrange step summarized in its most physically intuitive form consists of the computation of $\vec{F} = m\vec{a}$ to obtain accelerations at every node, followed by a kinematic integration of node positions and velocities, and concludes with an energy update for the resulting $p dV$ work on each cell over the time step. The forces involved in the computation of nodal accelerations include a pressure which is assumed constant in each cell, an artificial viscosity which suppresses oscillations around shocks, and anti-hourglassing forces which prevent the growth of nonphysical hourglassing or checkerboard modes arising from a nullspace of the evolution operator. In order to compute nodal accelerations, a nodal mass must be defined, and the HEMP discretization defines the nodal mass to be the average of the surrounding cell masses. This definition has important consequences for the interlevel transfer operators described in Section 3.2.

2.2 Mesh Relaxation

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 al-

gorithm can be based upon a Laplace iteration for the transformed coordinates with respect to the Cartesian coordinates of each node. We employ a Laplace equation, expressed in two dimensions as

$$\frac{\partial^2 \xi}{\partial x^2} + \frac{\partial^2 \xi}{\partial y^2} = 0 \quad (4)$$

where $\xi = i, j$ are the transformed coordinates, and the x, y are the Cartesian coordinates of a node, to construct an iteration that drives mesh lines towards an equipotential solution. Note that this is distinct from a simple averaging method, which represents an iteration derived from a Laplace equation for the Cartesian coordinates with respect to the transformed coordinates. Simple averaging methods are known to suffer from robustness problems and can tend to produce "folded" grids. Upon inspection this is not too surprising, since such an iteration drives the nodes towards an equipotential solution with respect to the transformed space, which is not necessarily well-behaved.

In order to construct an explicit iteration for the positions of nodes, we invert equation 4 to obtain

$$g_{22} \frac{\partial^2 f}{\partial i^2} - 2g_{12} \frac{\partial^2 f}{\partial i \partial j} + g_{11} \frac{\partial^2 f}{\partial j^2} = 0$$

where f is the x or y coordinate of a node, and the local metric terms g_{mn} are given by

$$\begin{aligned} g_{11} &= \left(\frac{\partial x}{\partial i} \right)^2 + \left(\frac{\partial y}{\partial i} \right)^2 \\ g_{12} &= \frac{\partial x}{\partial i} \frac{\partial x}{\partial j} + \frac{\partial y}{\partial i} \frac{\partial y}{\partial j} \\ g_{22} &= \left(\frac{\partial x}{\partial j} \right)^2 + \left(\frac{\partial y}{\partial j} \right)^2 \end{aligned}$$

Discretizing with central differences produces the mesh relaxation iteration

$$f = \frac{1}{2(g_{11} + g_{22})} \left(g_{22} (f_{i-} + f_{i+}) + g_{11} (f_{j-} + f_{j+}) + \frac{1}{2} g_{12} (f_{i+j-} - f_{i+j+} + f_{i-j+} - f_{i-j-}) \right)$$

where the notation f_{i+} denotes $f_{i+1,j}$, etc.

2.3 Solution Remap

Once the relaxed mesh has been defined, it remains to interpolate the solution from the old Lagrange grid to the relaxed grid. We cast this interpolation in terms of an apparent advection equation

$$\frac{\partial Jw}{\partial \tau} - \frac{\partial}{\partial \xi} (\mathbf{n}^\xi \cdot s\mathbf{w}) - \frac{\partial}{\partial \eta} (\mathbf{n}^\eta \cdot s\mathbf{w}) = 0. \quad (5)$$

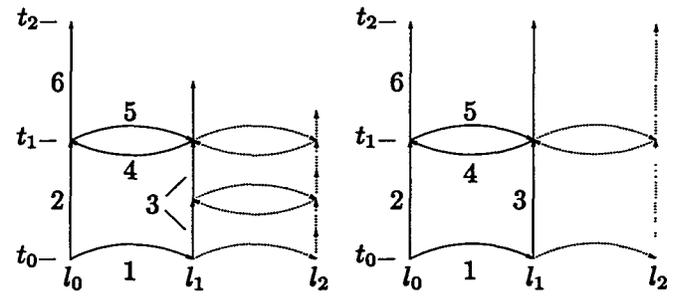


Fig. 3 Hierarchy time stepping - logical diagrams. The sequence on the left represents a time refined calculation, and the sequence on the right represents a calculation with no time refinement. Numbered sequence is for two level hierarchy. Dotted lines indicated operations with third level.

w and τ denote an arbitrary scalar quantity defined on the grid and a pseudo-time, respectively, and

$$\begin{aligned} J &= x_\xi y_\eta - y_\xi x_\eta \\ \mathbf{n}^\xi &= (y_\eta, -x_\xi) \\ \mathbf{n}^\eta &= (-y_\xi, x_\eta) \\ \mathbf{s} &= (\mathbf{x}^{new} - \mathbf{x}^{old}) / \Delta\tau \end{aligned}$$

are the jacobian, normal vectors, and apparent grid velocity.

This advection equation is solved using a variant of the Corner Transport Upwind (CTU) scheme⁶ for use on a staggered grid. The algorithmic details of the scheme have been discussed in detail in Pember,¹³ et al.

3 ALE-AMR Algorithm

3.1 AMR Overview

The conceptual starting point for the AMR methodology development is the pioneering work of Berger, Olinger, and Colella.^{3,14} In this approach, a hierarchical grid structure is employed which changes dynamically in time, and is composed of logically rectangular, uniform grids 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 a user specified integer called the refinement ratio, and d is the spatial dimension of the simulation. The solution is defined on all cells, including coarse cells which underlay cells of finer resolution. The collection of grids at a given resolution is referred to as a level. The hierarchy can have an arbitrary number of levels and any number of grids at a given level. A time step on the hierarchy of levels is a recursive procedure when time refinement is used, which we describe here for a two level calculation.

The first step is to regrid the current solution, shown in Fig. 3 as process 1 at time t_0 from level l_0 to l_1 , by refining regions of the flow requiring greater resolution. Cells requiring refinement are tagged and then grouped with untagged cells to create relatively large blocks of the coarse grid that will be refined. These

blocks are then subdivided to create the new fine grids. When new fine grid cells are generated in the process, the solution is defined by interpolation of the coarse grid data. Interpolation operators employed in the Lagrangian context are discussed in the following section.

The next step is a coarse grid advance, in which the coarse grid data is integrated to some new time t_1 , shown as step 2 in the diagram. In a time refined calculation, the fine grid is then advanced through multiple time steps to the same time as the coarse grid, as indicated by step 3. The principal consideration in the fine grid advance is the boundary conditions employed on the fine mesh, in particular at coarse-fine mesh boundaries. Typically this is achieved through time and space interpolation of coarse grid data to provide Dirichlet boundary conditions. It turns out that the split nature of the ALE-AMR scheme complicates the boundary conditions for subcycled time steps to the point where we have foregone time refinement in the ALE context. For a strictly Lagrangian calculation, there is no difficulty and time refinement may be employed.

The final step in a traditional hierarchy advance is synchronization of the coarse grid and fine grid data, shown as step 4 in the diagram. This is often accomplished in two steps. First, a coarsening operator replaces any coarse grid data for which fine grid data is available, on the presumption that the solution available from the fine grid is more accurate than that on the coarse grid. Second, the coarse cells which are adjacent to a fine grid boundary are updated to reflect the fluxes which were provided by the fine grid at those boundaries over the succession of fine grid advances. This flux correction procedure ensures conservation in a global sense on a composite mesh consisting of only the finest available data, since the coarse grid and fine grid fluxes will not in general be the same at grid boundaries. For the staggered mesh method employed in the Lagrange step, a quadrature rule to define global conservation, especially with respect to momentum, is less clear and is certainly not unique. As a consequence, we have opted at this stage not to employ any type of flux correction. While we would prefer to have a flux correction that maintains global conservation with respect to some quadrature rule, it is found that in practice there is no appreciable loss of accuracy in not doing so. At present, we defer this development to future refinement of the algorithm.

3.2 Interlevel Transfer Operators

Interlevel transfer operators are required when new grids are created, for the generation of pseudo boundary conditions on finer levels in the hierarchy, for synchronizing coarse and fine data in the hierarchy, and upon the removal of refined grids.

The operators development here are designed with the following properties in mind:

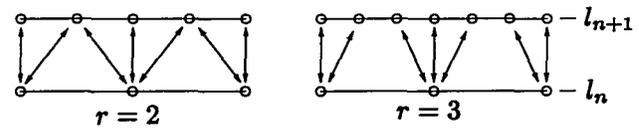


Fig. 4 Odd refinement ratios are required to maintain a one-to-one correspondence between fine nodes and their corresponding coarse nodes in the locally conserved stencil.

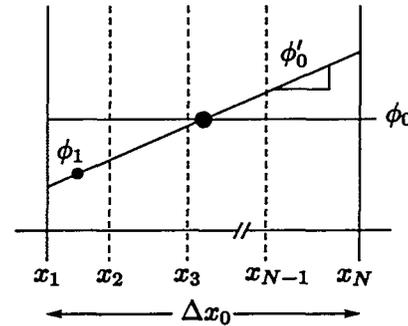


Fig. 5 A coarse cell refined into $N-1$ finer cells.

- i) Freestream preservation
- ii) 2nd order accuracy (in smooth regions)
- iii) Monotonicity
- iv) Local conservation
- v) Coarsening identically inverts refinement

Property v supports the notion that the interlevel operators should not result in nonphysical evolution of the flowfield. A simple way to ensure that properties iv and v are simultaneously achieved is to maintain a one-to-one correspondence from fine nodes to coarse nodes in the interpolation stencil, such that the locally conserved regions on the fine mesh do not overlap. In this case inverting a locally conservative interpolation is simply a matter of summing the fine values of the conserved quantity in the stencil. This leads to a choice of odd refinement ratios, as illustrated in Fig. 4. For a cell-centered scheme, the choice of refinement ratio is arbitrary, but for a nodal or staggered scheme, only odd ratios preserve the one-to-one correspondence.

Consider a one-dimensional interpolation of some scalar density function ϕ with a known slope ϕ'_0 and average value ϕ_0 over some interval Δx_0 , into $N-1$ arbitrary subintervals $\Delta x_k = x_{k+1} - x_k$, as shown in Fig. 5.

An interpolation in which values are taken from the centers of the subintervals

$$\phi_k = \phi_0 + \phi'_0 \left(\bar{x}_k - \frac{1}{2} \Delta x_0 \right)$$

where $\bar{x}_k = 1/2(x_k + x_{k+1})$, is locally conservative of $\phi \Delta x$ in the sense that

$$\sum_{k=1}^N \phi_k \Delta x_k = \phi_0 \Delta x_0,$$

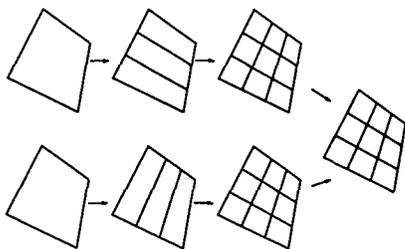


Fig. 6 Results of the directionally split refinement operators may be averaged to obtain symmetry with respect to i and j coordinate directions.

since

$$\sum_{k=1}^N \phi_k \Delta x_k = \phi_0 \Delta x_0 + \phi'_0 \left(\sum_{k=1}^N \bar{x}_k \Delta x_k - \frac{1}{2} \Delta x_0^2 \right). \quad (6)$$

The final term in brackets is identically zero, in discrete analogy with $\int x dx = \frac{1}{2} x^2$. In a free stream, all slopes ϕ'_0 are zero, and constant fields are preserved independently of the mesh. We now have a general one-dimensional expression for interpolation that satisfies i, ii, and iv. In order to address iii, we employ the Van Leer limited slope defined by

$$\Delta = \frac{1}{2} |\text{sgn}(\Delta^L) + \text{sgn}(\Delta^R)| \min(|\Delta^C|, 2|\Delta^L|, 2|\Delta^R|)$$

where $\Delta^L \phi = \phi_i - \phi_{i-1}$, $\Delta^R \phi = \phi_{i+1} - \phi_i$, and $\Delta^C \phi = \frac{1}{2} (\phi_{i+1} - \phi_{i-1})$, in determining the slopes ϕ'_0 from the coarse grid data.

If we desire to prevent oscillations in the primitive variables $\phi = (\rho, u, v, E)$, where E is the total energy, the required interpolation basis to obtain local conservation is $x = (V, \bar{m}, \bar{m}, m)$, where V is volume, \bar{m} is nodal mass, and m is cell mass, i.e., we choose "volume coordinates" for density, and "mass coordinates" for components of velocity and total energy.

A difficulty arises with respect to conservation of momentum upon refinement. The demonstration of conservation (6) only holds provided the interpolation basis itself is locally conserved, i.e.,

$$\sum_{k=1}^N \Delta x_k = \Delta x_0.$$

In the case of density refinement, cell based volume coordinates do satisfy this condition since subvolumes are generated by a subdivision of coarse volumes. Since cell volumes and densities are conserved, so are cell masses, and hence energy interpolation is also conservative. However, in general it will not be true that the sum of the fine nodal masses will correspond to the coarse nodal mass. It is not difficult to construct a conservative interpolation if one is willing to dispense with freestream preservation, but we consider freestream preservation to be more fundamental than conservation of momentum. It may also be possible,

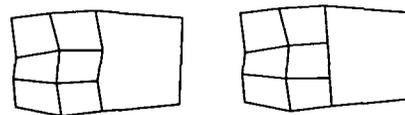


Fig. 7 On the left, "free" fine boundary nodes introduce non-quad elements. On the right, inter-level boundary conditions interpolate positions to preserve quad elements.

for some values of r , to construct a mesh subdivision such that the requisite local nodal mass conservation is preserved, but at this stage of the development of the ALE-AMR algorithm we prefer to choose the mesh refinement arbitrarily at the expense of strict momentum conservation.

The one-dimensional refinement operators may be extended to multi-dimensions by a directional splitting and successive application of the one-dimensional method. There are two choices for the ordering in two dimensions, and in general the operators will not commute. If ij symmetry is desired, the results of both the ij and ji refinement operators may be averaged, as illustrated in Fig. 6. Note that the limited slopes are computed once from the coarse data and are not recomputed after the first refinement pass.

The coarsening operators are then simply weighted sums of the conserved quantities on the fine mesh, i.e.,

$$\begin{aligned} \rho_0 &= \frac{\sum \rho_i V_i}{\sum V_i} \\ u_0 &= \frac{\sum u_i \bar{m}_i}{\sum \bar{m}_i} \\ E_0 &= \frac{\sum E_i m_i}{\sum m_i} \end{aligned}$$

where i varies over the refinement stencil corresponding to each coarse node. The coarse mesh is formed by selection of every r 'th mesh point.

3.3 Interlevel Boundary Conditions

Pseudo boundary conditions at refinement boundaries 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, 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 shaped 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-quadrilateral elements, as shown in Fig. 7.

Except in the special case of a strictly Lagrangian calculation, interlevel boundary condition considerations lead us not to employ refinement in time for the

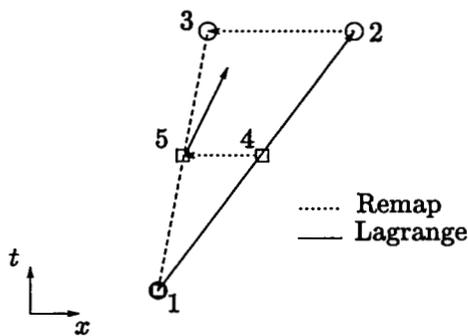


Fig. 8 □ - fine mesh boundary node. ○ - coarse mesh node at coarse-fine boundary.

following reason. Consider a Lagrange plus remap hierarchy advance in a two-level calculation at a coarse-fine interface in one dimension, as shown in Fig. 8. First the coarse level is advanced in time from position 1 to 2 in a Lagrange step and then remapped to its final location 3. The coarse Lagrange step data is used to enforce a boundary condition on the fine grid advance from 1 to 4. Any other choice for boundary condition on the Lagrange step would not be freestream preserving. After a remap on the fine grid to, say, position 5, there is no coarse grid Lagrangian solution through point 5 in $x-t$ to provide a boundary condition for further Lagrange subcycles. One alternative would be to fix the boundary node during the relaxation step, and continue using the Lagrange path 1-2 as a boundary condition until the fine grid was advanced to the same time as the coarse grid. In order to then synchronize the coarse and fine grids, the fine boundary node would need to be relaxed to position 3. In general this would require an iterative relaxing and remapping procedure as the stability criterion on remap requires the overlap of the original and final grids. Therefore a bound on the distance from 2-3 is the coarse mesh width, which is roughly r times the fine mesh width. It is reasonable then to expect that on the order of r relax/remap iterations would be required to advance the fine mesh boundary node from position 2 to 3. Rather than implement this additional complexity into the algorithm, we have chosen not to use time refinement, except in the case of a pure Lagrangian calculation.

A generalization of the analysis done by Quirk¹⁵ to N -dimensional calculations provides some additional insight into the impact on the efficiency of the method without time refinement.

If level 0 is composed of m cells, and each finer level l covers a constant fraction α of the level $l-1$ below it, then the total work of a time-refined calculation through one coarse grid timestep is

$$W_{tr} = m \sum_{l=0}^{l_{max}} \alpha^l r^{l(d+1)}$$

where d is the spatial dimension of the calculation. Since it will require $r^{l_{max}}$ time steps to advance all

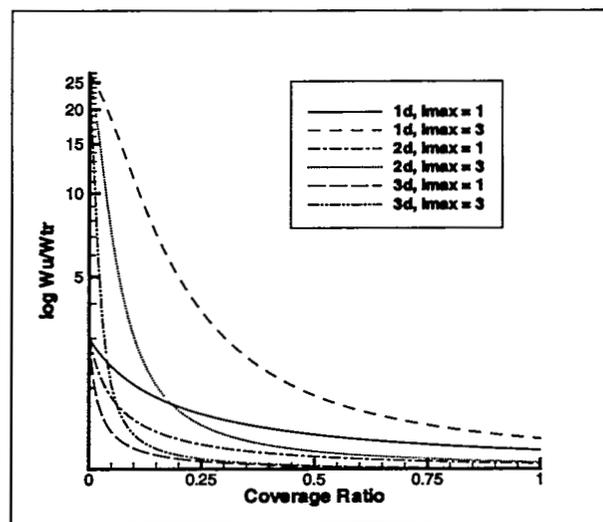


Fig. 9 Theoretical work ratio of spatial and time refinement over spatial refinement only, as a function of the coverage ratio of finer grids with respect to the next coarser grid.

levels in a calculation without time refinement to the same time, the total work in that case is

$$W_u = r^{l_{max}} m \sum_{l=0}^{l_{max}} \alpha^l r^{ld}$$

Taking $r = 3$ as representative for our purposes, we plot the log of the speedup of a calculation employing time refinement in Fig. 9 for l_{max} of 1 and 3. l_{max} is the number of refined levels, so that the total number of levels in the calculation is $l_{max} + 1$.

It is evident that for calculations that employ only very small fractions of refinement of the total domain, time refinement provides a great benefit. However, especially in three dimensions, for which efficiency becomes paramount, the curves fall quite rapidly with increasing α , and indeed for a calculation with ratios of more than about 15%, the benefits of time refinement become increasingly marginal.

3.4 Implementation

The implementation utilizes SAMRAI,¹⁶ 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.

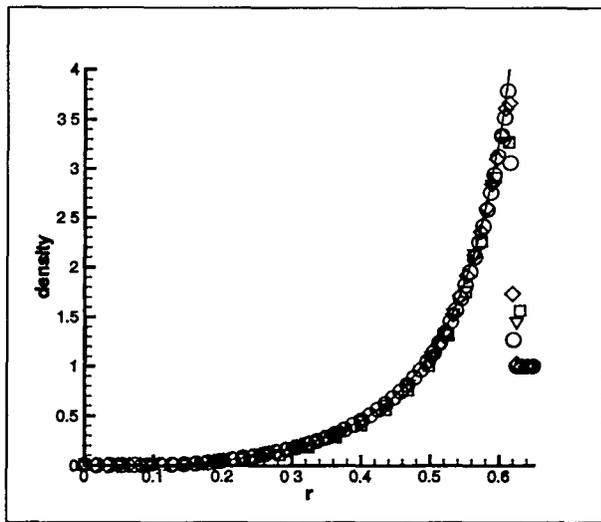


Fig. 10 Adaptive Sedov solution with one level of refinement. Base grid 64×64 , $r = 3$. \square - 32×32 , No adaption. \circ - 96×96 , no adaption. ∇ - 32×32 level 0; \diamond - 32×32 level 1. - - exact solution

4 Numerical Results

For the following examples we employ a simple heuristic error indicator based on $f'' \Delta x$. We in fact set two thresholds for the relative error $f'' \Delta x / f$, one which is used when a region is already refined, and one which is used when a region is unrefined. This is useful to prevent excessive refinement and de-refinement in regions which are hovering near the indicator's threshold.

4.1 Cylindrical Taylor-Sedov Blast Wave

The Taylor-Sedov blast wave describes the self similar motion following an intense explosion. It consists of the sudden release of an amount of energy E concentrated at a point, and has an exact solution which can be derived entirely on dimensional grounds.¹⁷ We model this with initial conditions of a no-flow, uniform density field, and an initial energy which is everywhere zero except for one cell at the origin. We compute one quadrant of the symmetrical flowfield with an initial energy of 8, $\gamma = 5/3$, and $\rho_0 = 1$.

Two non-adaptive calculations are compared with an ALE-AMR calculation utilizing one level of refinement. The single level resolutions are 32×32 and 96×96 cells. The ALE-AMR calculation has a base resolution of 32×32 cells, and one level of refinement at $r = 3$, such that the "effective" resolution is 96×96 cells.

Fig. 10 compares data from all three solutions with the exact similarity solution at $t = 0.1$ from the origin to the shock front. Fig. 11 details the shock region. The adaptive calculation is nearly indistinguishable from the calculation at full resolution everywhere.

4.2 Double Mach Reflection of a Strong Shock

It is for more ambitious calculations with a wider disparity of length scales that the adaptive capability

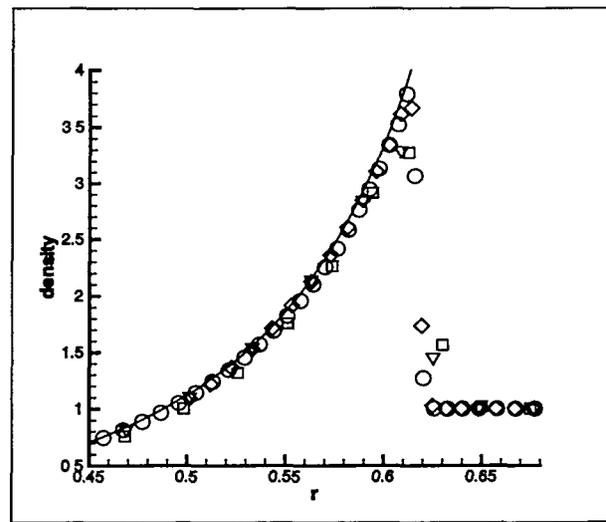


Fig. 11 Detail of Fig. 10. \square - 32×32 , No adaption. \circ - 96×96 , no adaption. ∇ - 32×32 level 0; \diamond - 32×32 level 1. - - exact solution.

really begins to pay significant dividends with respect to efficiency. In the double mach reflection problem, first presented by Woodward and Colella,¹⁸ a Mach 10 planar shock in a perfect gas with $\gamma = 1.4$ reflects from a ramp at an incident angle of 60° . The domain is 3.5 units long and 1 unit high, and the ramp begins .3 units from the left end of the domain. The preshock density and pressure are 1.4 and 1, respectively. At the top face of the domain, inflow boundary conditions representing the exact evolution of the propagating shock are imposed. For this problem, we choose to demonstrate the Eulerian capability of the algorithm, in part due to the simplification afforded in imposing the propagating shock conditions at the top of the domain on a stationary, non-deforming grid.

An $r = 3, l_{max} = 3$ calculation with a base resolution of 224×76 is shown in Fig. 13. Finer level grid boundaries are outlined, and line contours of density are plotted. The closeup of Fig. 14 displays the details of the "wall jet" region, including the well-captured vortex rollup. Comparing with the results of Pember,¹³ we can find no qualitative indications that the AMR components of the scheme are adversely affecting the solution, and indeed find solutions to be qualitatively indistinguishable when computed either adaptively or fully at a given "effective resolution," provided that the adaption criteria are sufficiently capturing the salient features of the flowfield.

The efficiency of the adaptive capability is indicated by comparing the wall clock time for the $l_{max} = 3$ calculation with a fully resolved calculation. On a single 1.5 GHz Pentium Xeon processor, the adaptive calculation required 49,765 s of wall clock time, and the equivalent, fully resolved calculation required 180,367 s, for a speedup ratio of 3.6. Note that we have made no particular efforts at this juncture to optimize our implementation with respect to the AMR overhead,

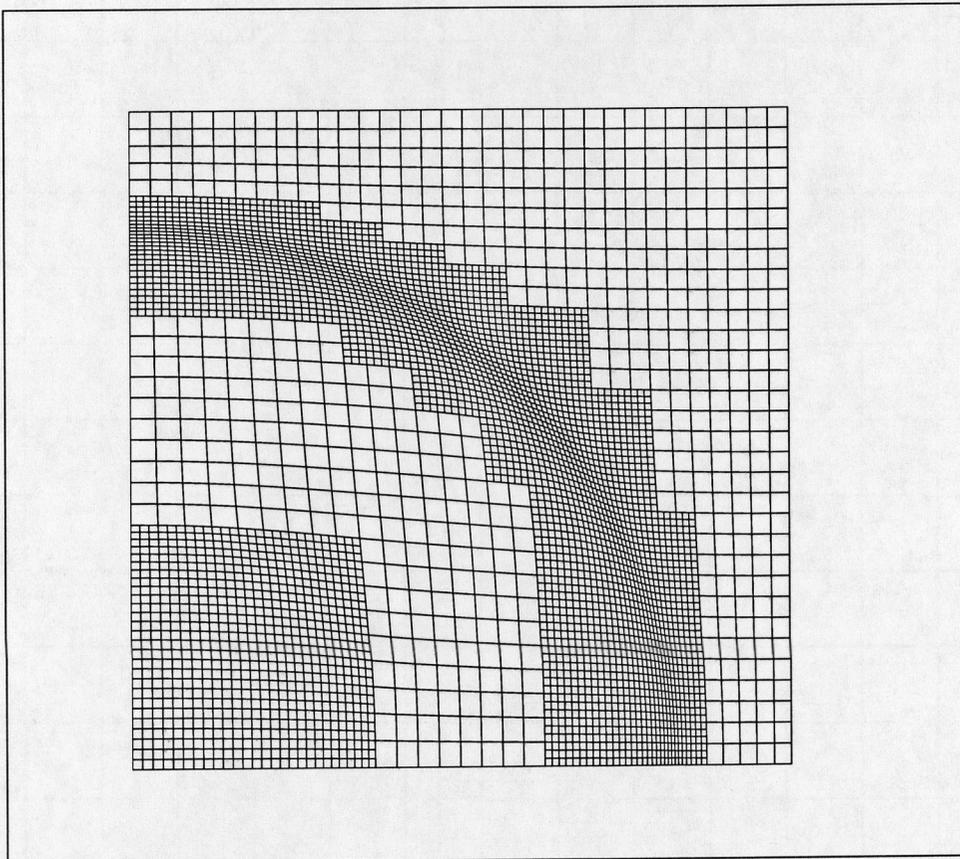


Fig. 12 Taylor-Sedov blast wave with one level of refinement from a 32x32 base resolution, $r = 3$. Adaptive grid structure.

and indeed we have recognized several opportunities for such optimization that we expect will improve the overall efficiency of the adaptive capability. However, we still see significant efficiency improvements over the base algorithm, even for relatively modestly sized problems.

5 Conclusion and Future Work

We have successfully combined the ALE and AMR methods to create a new, more powerful adaptive method for solution of the equations of inviscid gas-dynamics. 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 method we have employed and presented in this work.

Further refinements of the algorithm which motivate future work include a resolution of the staggered mesh momentum conservation difficulty described in Section 3.2, as well as a flux correction scheme and a suitable quadrature rule to define and maintain global hierarchy conservation.

The algorithms presented here have been extended

to three dimensions, and we plan to present those developments, along with further refinement and characterization of the ALE-AMR method, in a subsequent work.

6 Acknowledgment

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

References

- ¹Benson, B. J., "An Efficient, Accurate, Simple ALE Method for Nonlinear Finite Element Programs," *Comp. Meth. Appl. Mech. Eng.*, Vol. 72, 1989, pp. 205-350.
- ²Bell, J., Berger, M., Saltzman, J., and Welcome, M., "Three Dimensional Adaptive Mesh Refinement for Hyperbolic Conservation Laws," *SIAM J. Sci. Comp.*, Vol. 15, 1994, pp. 127-138.
- ³Berger, M. and Olinger, J., "Adaptive Mesh Refinement for Hyperbolic Partial Differential Equations," *J. Comput. Phys.*, Vol. 53, 1984, pp. 484-512.
- ⁴Berger, M. and Colella, P., "Local Adaptive Mesh Refinement for Shock Hydrodynamics," *J. Comput. Phys.*, Vol. 82, 1989, pp. 64-84.
- ⁵Pember, R., Bell, J., Colella, P., Crutchfield, W., and Welcome, M. L., "An Adaptive Cartesian Grid Method for Unsteady Compressible Flow in Complex Geometries," *J. Comput. Phys.*, Vol. 120, 1995, pp. 278-304.
- ⁶Bell, J., Colella, P., Trangenstein, J., and Welcome, M., "Adaptive Mesh Refinement on Moving Quadrilateral Grids,"

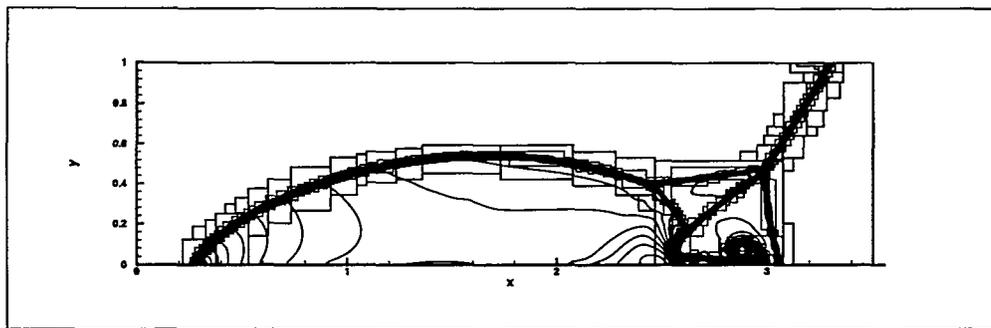


Fig. 13 Double Mach Reflection of a Mach 10 Shock. ALE-AMR algorithm in Eulerian mode, i.e., remapping to the original grid. $l_{max} = 2$. l_1 grid boundaries are in heavy black outline, l_2 grid boundaries are in light black outline.

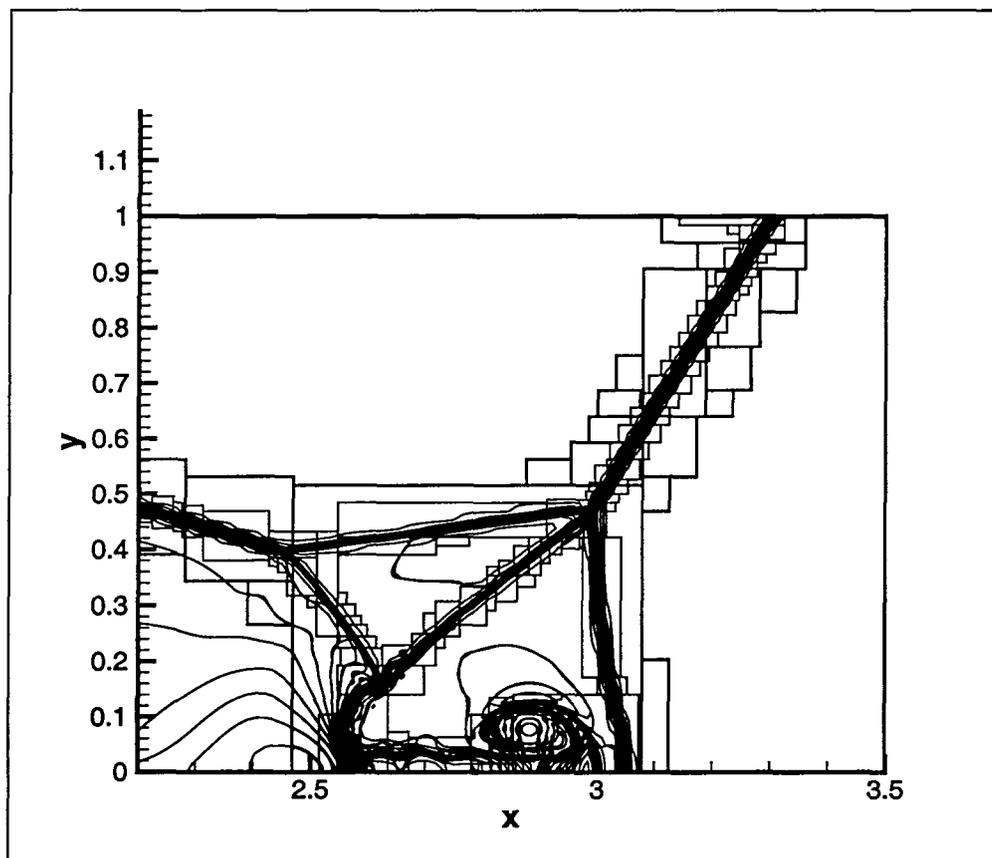


Fig. 14 Detail of Fig. 13 showing the wall jet rollup.

Proceedings, AIAA 9th Computational Fluid Dynamics Conference, Buffalo, New York, June 14-16, 1989, p.471-579.

⁷Winslow, A., "Numerical Solution of the Quasilinear Poisson Equation in a Nonuniform Triangle Mesh," *J. Comput. Phys.*, Vol. 1, 1967, pp. 149-172.

⁸Tipton, R., Unpublished report, Lawrence Livermore National Laboratory, 1990.

⁹Wilkins, M. L., "Calculation of Elastic-Plastic Flow," *Meth. Comp. Phys.*, Vol. 3, 1964, pp. 211-263.

¹⁰Sharp, R., "HEMP Advection Model," Tech. Rep. UCID-17809, Lawrence Livermore National Laboratory, 1978.

¹¹Christensen, R., "Godunov Methods on a Staggered Mesh — An Improved Artificial Viscosity," Tech. Rep. UCRL-JC-105269, Lawrence Livermore National Laboratory, 1990.

¹²Margolin, L. and Pyun, J., "A Method for Treating Hourglass Patterns," Tech. Rep. LA-UR-87-439, Los Alamos National Laboratory, 1987.

¹³Pember, R. and Anderson, R., "Comparison of Direct Eulerian Godunov and Lagrange Plus Remap Artificial Viscosity Schemes for Compressible Flow," Tech. Rep. AIAA Paper 2001-2644, 2001.

¹⁴Berger, M. and Colella, P., "Local Adaptive Mesh Refinement for Shock Hydrodynamics," *J. Comput. Phys.*, Vol. 82, 1989, pp. 64-84.

¹⁵Quirk, J., *An Adaptive Grid Algorithm for Computational Shock Hydrodynamics*, Ph.D. thesis, Cranfield Institute of Technology, U.K., 1991.

¹⁶Hornung, R. and Kohn, S., "Future Directions for Adaptive Mesh Refinement in ASCI and other LLNL Simulation Projects," Tech. Rep. UCRL-ID-12833, Lawrence Livermore National Laboratory, September, 1997.

¹⁷Whitham, G., *Linear and Non-Linear Waves*, Wiley-Interscience, New York, 1974.

¹⁸Colella, P. and Woodward, P., "The Piecewise Parabolic

Method (PPM) for Gas-Dynamical Simulations," *J. Comput. Phys.*, Vol. 54, 1984, pp. 174-201.