Journal of Computational Physics 229 (2010) 1573–1599

Contents lists available at ScienceDirect

Journal of Computational Physics

journal homepage: www.elsevier.com/locate/jcp

# Modeling material responses by arbitrary Lagrangian Eulerian formulation and adaptive mesh refinement method

Ping Wang *

*Lawrence Livermore National Laboratory, P.O. Box 808, L-039, Livermore, CA 94551, USA*

ARTICLE INFO

*Article history:*
Received 13 November 2008
Received in revised form 6 October 2009
Accepted 27 October 2009
Available online 5 November 2009

*Keywords:*
Arbitrary Lagrangian Eulerian
Adaptive mesh refinement
Elastic–plastic flow
Fragmentation
Johnson–Cook failure model

ABSTRACT

In this paper we report an efficient numerical method combining a staggered arbitrary Lagrangian Eulerian (ALE) formulation with the adaptive mesh refinement (AMR) method for materials modeling including elastic–plastic flows, material failure, and fragmentation predictions. Unlike traditional AMR applied on fixed domains, our investigation focuses on the application to moving and deforming meshes resulting from Lagrangian motion. We give details of this numerical method with a capability to simulate elastic–plastic flows and predict material failure and fragmentation, and our main focus of this paper is to create an efficient method which combines ALE and AMR methods to simulate the dynamics of material responses with deformation and failure mechanisms. The interlevel operators and boundary conditions for these problems in AMR meshes have been investigated, and error indicators to locate material deformation and failure regions are studied. The method has been applied on several test problems, and the solutions of the problems obtained with the ALE–AMR method are reported. Parallel performance and software design for the ALE–AMR method are also discussed.



## 1. Introduction

Significant progress has been made in numerical modeling of problems in solid mechanics and material science in the last 20 years. For example, Lagrangian simulations of automobile crashes with millions of zones on modest computer clusters are routinely used by industry to design safer automobiles. Despite these advances, computational solid mechanics (CSM) lags in comparison to computational fluid dynamics (CFD) in many areas. While CFD simulations that are uniformly second-order accurate in both time and space are routine and the AMR method has been successful used in many CFD applications with an Eulerian formulation, Eulerian and Lagrangian CSM simulations are still facing many challenging issues. Reliable simulation technology for material science is still in its infancy for certain areas such as truly predictive models for failure and fragmentation. Fragmentation, for example, occurs at a wide range of temporal and spatial scales from atomistic to cosmological, but the underlying phenomenology is poorly understood, and the simulation technology is primitive. Fragmentation is a phenomenon that is intrinsically multi-scale; small variations in the micro-structure lead to a distribution of localized failure sites that ultimately coalesce to form macroscopic fragments.

In order to simulate solid mechanics with a capability to predict material failure and fragmentation, a numerical approach should be able to solve hydrodynamics problems with a material failure model. For hydrodynamics problems, two mathematical formulations are commonly used: the Eulerian method and the Lagrangian method. In a Lagrangian formulation, the mesh is tied to a material, and the elements in the mesh move and distort with the material. In contrast to the Lagrangian

* Tel.: +1 925 422 1568; fax: +1 925 422 8040.
*E-mail address:* wang32@llnl.gov

0021-9991/$ - see front matter © 2009 Elsevier Inc. All rights reserved.
doi:10.1016/j.jcp.2009.10.045

formulation, the essential characteristic of an Eulerian formulation is that the mesh is fixed and material flows through the mesh as time advances. For fluid problems, Eulerian methods have the advantage of modeling strong shearing and vortical motions. The Lagrangian methods are natural for applications with moving boundaries and multiple materials; material interfaces can be tracked accurately with nodes placed on material boundaries. Both formulations have strengths and weaknesses and neither is suitable for all applications. The ALE methods seek a compromise between the Eulerian and Lagrangian descriptions and can take advantage of both methods in various applications involving all of the aforementioned characteristics: moving boundaries, multiple materials, shear and vortical flow [1]. Therefore, the ALE method is an attractive choice for multi-physics problems that involve both fluids and solids. In this paper, the ALE method is used for hydrodynamics and materials modeling.

AMR is a computational technique for improving the efficiency of numerical simulations of the systems of partial differential equations. The basic idea is to refine, both in space and in time, regions of the computational domain in which a high resolution is needed to resolve developing features while leaving less interesting parts of the domain at lower resolutions [2]. This saves significant computation time compared to using a fine uniform mesh to reach the required accuracy. A simple static AMR mesh can be used as a solution method, where the mesh stays the same during the computation. In general, the static AMR mesh is derived from prior information of an application.

In a Lagrangian formulation, the computational mesh moves with a material, and this can be viewed as an automatic AMR method in some sense. Since each control volume contains the same material regardless of the change of the control volume, the mesh will be adjusted according to the dynamics of a solution. For example, when the material is compressed, the mesh will be automatically refined. So if a predictable motion is identified, a static AMR mesh can be a good approach for some ALE applications. However, some problems are dynamic in both space and time, such as material failure and fragmentation, and a dynamic AMR method will be preferred to resolve all the different physical scales and speed up the computation.

The ALE method for hydrodynamics has been studied for years, and AMR with an Eulerian formulation has also been extensively investigated for decades. Only recently have studies of combining both an ALE method and a dynamic AMR been reported [3]. In that report, a numerical algorithm ALE–AMR for gas dynamics was developed, and it solved several fundamental problems linked with this new approach, such as interpolations and interlevel boundary conditions for AMR on a moving and deforming mesh for gas dynamics. More recent ALE–AMR research work on the solution of the Euler equations has been reported in [4] and [5]. In these papers, a cell by cell anisotropic adaptive mesh technique, in contrast to the block patch approach in [2], is added to an existing staggered mesh Lagrange plus advection finite element ALE code. For solid mechanics and failure and fragmentation models, no research work of ALE–AMR method has been reported.

In many applications of materials modeling, elastic–plastic flows are involved in a non-uniform deformation process, and material failure is a highly localized phenomenon with large deformation near failure zones. Using the ALE–AMR method is a natural choice for these problems. In this paper, we present a numerical method combining ALE and AMR in space and time for solid mechanics and materials modeling. Although some of the approaches in [3] can be applied to our problems, there are still many differences from gas dynamics to solid mechanics and materials modeling, including the mathematical formulation, numerical method, and many other issues associated with AMR for materials modeling. For instance, failure models usually have a large set of complex history variables, such as damage and failure. How to design appropriate interlevel operators and deal with these variables and boundary conditions and how to design error indicators are interesting research areas. In Section 2, a mathematical formulation to model solid mechanics and material responses is given, and a numerical method to solve the mathematical equations is outlined in Section 3. A detailed ALE–AMR method for elastic–plastic flows and a material failure model is reported in Section 4, and implementation and parallel performance are discussed in Section 5. Finally, numerical simulations for several test problems including elastic–plastic flows and material failure and fragmentation are given in Section 6 before conclusions are made.

## 2. Mathematical formulation and numerical method for solid mechanics and material failure model

The fundamental equations for hydrodynamics, derived from the conservation of mass, momentum, and energy, can be written in a Lagrangian formulation as:

$$\frac{D\rho}{Dt} = -\rho\nabla\cdot\vec{U}, \tag{1}$$

$$\frac{D\vec{U}}{Dt} = \frac{1}{\rho}\nabla\cdot\boldsymbol{\sigma}, \tag{2}$$

$$\frac{De}{Dt} = \frac{1}{\rho}V(s_{11}\dot{\epsilon}_{11} + s_{22}\dot{\epsilon}_{22} + s_{33}\dot{\epsilon}_{33} + 2s_{12}\dot{\epsilon}_{12} + 2s_{23}\dot{\epsilon}_{23} + 2s_{31}\dot{\epsilon}_{31}) - P\dot{V}. \tag{3}$$

Here

$$\frac{D}{Dt} = \frac{\partial}{\partial t} + \vec{U}\cdot\nabla \tag{4}$$

is the material derivative or the Lagrange time derivative, and $t$ is the time.

In these equations, $\rho$ is the density, $\vec{U} = (u, v, w)$ is the material velocity, $\boldsymbol{\sigma}$ is the total stress tensor, $e$ is the internal energy, $V$ is the relative volume ($\rho V = \rho_0$ where $\rho_0$ is the reference density), $\dot{V}$ is the rate of change of volume, and $\dot{\boldsymbol{\epsilon}}$ is the strain-rate tensor defined as

$$\dot{\epsilon}_{ij} = \frac{1}{2}\left(\frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i}\right). \tag{5}$$

Here $i, j, k = 1, 2, 3$ which refer to the principal axes.

The deviatoric stresses are

$$s_{ij} = \sigma_{ij} + P\delta_{ij}, \tag{6}$$

where $\delta$ is the Kronecker delta and $P$ is pressure. The pressure is usually determined by an equation of state (EOS) $P = P(\rho, e)$, which returns pressure as a function of density and either energy or temperature. A simple example of an EOS for solid material is a linear EOS where the hydrostatic pressure is computed from the material's bulk modulus $K$ and the ratio of actual density to the reference density.

The stress deviators are usually determined by constitutive relations. A simple example of a constitutive relation is Hooke's Law for isotropic elastic materials, which gives a linear relationship between deviatoric stress and deviatoric strain as

$$s_{ij} = 2\mu\epsilon_{ij}. \tag{7}$$

Here $\mu$ is the shear modulus.

Together with an EOS for a given material, a constitutive relation between stress and strain, and boundary conditions and initial conditions, the set of equations for hydrodynamic behavior with a single material is completed. To model perfectly elastic material behavior, the above equations need to be solved numerically. To model elastic–plastic flows, the von Mises yield condition or other yield conditions can be used. The equivalent plastic strain is used to describe the plastic deformation and the flow stress is calculated from a constitutive equation that describes the material behavior. The von Mises yield condition can be written as:

$$\sqrt{2J} - \sqrt{2/3}Y \leqslant 0, \tag{8}$$

where $Y$ is the plastic flow stress and $2J$ the second invariant of the deviatoric stress tensor.

A power law form of strain hardening is suggested by experimental results and has been proven to be satisfactory for many materials of interest: $Y = a[b + \epsilon^p]^c$. Here, $a$, $b$, and $c$ are material constants, and $\epsilon^p$ is the equivalent plastic strain. We can solve explicitly for $\epsilon^p$ by $\epsilon^p = \tilde{\epsilon}^p + (1/m - 1)Y/(3\mu)$, where $\tilde{\epsilon}^p$ is the equivalent plastic strain value at the previous time step, and $m = \sqrt{2/3}Y/\sqrt{2J}$.

To model material failure, the Johnson–Cook (JC) failure model [6] is used, and damage is based on the material in an element reaching a plastic strain equal to the failure plastic strain. Fracture in the JC failure model is derived from the following cumulative damage law

$$D = \sum \frac{\Delta\epsilon^p}{\epsilon_f}. \tag{9}$$

The failure strain $\epsilon_f$ is given by the expression

$$\epsilon_f = \left(D_1 + D_2 exp(D_3\left(\frac{p}{\bar{\sigma}}\right))\right)\left(1 + D_4 ln\left(\frac{\dot{\epsilon^p}}{\epsilon_0^p}\right)\right)(1 + D_5 T^*), \tag{10}$$

where the effective stress is given by $\bar{\sigma} = \left(\frac{3}{2}\sigma_{ij}\sigma_{ij}\right)$and $T^* = \frac{T - T_{room}}{T_{melt} - T_{room}}$ is the homologous temperature, $T$ is the current temperature, $T_{room}$ is the ambient temperature, and $T_{melt}$ is the melting temperature. $D1$, $D2$, $D3$, $D4$, and $D5$ are material dependent constants. Failure occurs when the damage parameter exceeds unity. Failure is interpreted as fracture and results in the zeroing of all stresses.

$$F = 1 \quad \text{for } D \geqslant 1. \tag{11}$$

Several history variables including damage ($D$), failure ($F$), and relative volume at failure ($F_v$) are computed and tracked at each time step.

The above equations can be solved numerically by finite element, finite difference, or finite volume methods. A description of a finite volume method for the above equations is given in the next section.

## 3. Numerical method

In order to solve the above equations, two major computational phases are considered here: the Lagrange step and an advection step. Using a finite volume method, the differential equations are approximated by discretizing both space and time in a staggered mesh: velocity $\vec{U}$ is node-centered while other quantities are cell-centered. The mesh itself is defined

by the position $\vec{X} = (x, y, z)$ of the nodes. During a single time step, the equations of hydrodynamics are first advanced to a new time using a grid that moves with the material; the velocities of these nodes are accelerated by pressure forces as well as by any other forces present. This updated grid is called the Lagrange grid.

Once the grid motion step is done, an advection step can be used as an option if grid distortion exceeds a criterion. This includes a relaxation method to move nodes in distorted areas from their Lagrange locations toward locations that lower the grid distortions. The resulting grid is called the relaxed grid. In the special case of the grid being returned to the original configuration, the method is effectively Eulerian. Once the relaxed mesh has been constructed, the solution on the Lagrange grid needs to be interpolated or advected to the relaxed grid. In this paper, we give a description of the Lagrange step and a summary of the relaxation and advection scheme, which can also be found elsewhere [7–9]. Extension of the advection scheme of Collela [7] from gas dynamics applications to solid-dynamics applications is discussed as well.

The fundamental equations are organized so that changes of variables associated with a mass point can be interpolated as due to a flux through a surface surrounding the mass point. The difference operators are organized in the same way, i.e., in the spirit of the divergence theorem. A spatial derivative is defined as the summation of the normal components of the flux around an enclosed area (2D) or a volume (3D). The integration path for evaluating the partial derivatives is defined in two ways. For the node-centered velocity, an octahedron constructed from the eight surrounding mesh cells is used for the integration in 3D. In a 2D case, the area for the integration for the node-centered quantities is generated by connecting the midpoints of the original mesh, a diamond-shaped quadrilateral in Fig. 1(a). Fig. 1(b) shows the path for evaluating components of the continuity equation at a cell center. A detailed description of the above numerical approach to Eqs. (1)–(8) can be found in [10].

In our current study, we use a finite volume method for the derivative terms in the original equations, but several other numerical schemes that deal with various issues for the entire system are combined to form a complete, robust and efficient numerical method. A monotonic artificial viscosity, $q$, is used to simulate hydrodynamic shock fronts [9], and a kinematic hourglass filter is added to control hourglass instabilities [11]. A numerical scheme to compute the strain [12] is used for the elastic–plastic flow problems. A time-centered predictor–corrector scheme is applied for time. Using the finite difference formulation for the spatial gradients in the equations, the overall numerical scheme with a predictor-corrector scheme for integrating one time step from $t = n$ to $t = n + 1$ in a Lagrange phase is:

1. Compute the acceleration $\vec{a}^n$ by using the momentum Eq. (2) which includes computing the forces due to pressure [3], artificial viscosity [9], stress deviators [10], and the hourglass filter [11].
2. Integrate the predicted velocity $\vec{U}^p$ by using the velocity $\vec{U}^n$, the time step $\Delta t$, and the acceleration $\vec{a}^n$:

   $$\vec{U}^p = \vec{U}^n + \Delta t \vec{a}^n.$$

3. Integrate the predicted position $\vec{X}^p$ by using the position $\vec{X}^n$, the time step $\Delta t$, and the averaged value of the velocity $\vec{U}^n$ and the predicted velocity $\vec{U}^p$:

   $$\vec{X}^p = \vec{X}^n + 0.5\Delta t(\vec{U}^n + \vec{U}^p).$$

4. Compute the predicted strain rates by using Eq. (5) with the numerical scheme in [12].
5. Compute the predicted volume $V^p$ and the predicted density: $\rho^p = \rho^n V^n / V^p$ by applying the conservation of mass.
6. Compute the predicted internal energy from the energy conservation equation (3) using the predicted volume and the old volume, the old pressure, the predicted strain rate, and the old stress:

   $$e^p = e^n - (P^n + q^n)/(1/\rho^p - 1/\rho^n) + 0.5\Delta t(V^p + V^n)(s_{11}^n \dot{\epsilon}_{11}^p + s_{22}^n \dot{\epsilon}_{22}^p + s_{33}^n \dot{\epsilon}_{33}^p + 2s_{12}^n \dot{\epsilon}_{12}^p + 2s_{23}^n \dot{\epsilon}_{23}^p + 2s_{31}^n \dot{\epsilon}_{31}^p).$$

7. Integrate the predicted stress and the equivalent plastic strain by using a constitutive relation and a yield condition; the present study uses Hooke's Law (7) and von Mises yield condition (8).
8. Apply JC failure model (9) and (11) and set stress to zero if the failure criteria are reached.
9. Compute the predicted pressure $P^p$ by the EOS [10] using the predicted density $\rho^p$ and the predicted energy $e^p$.
10. Compute the acceleration $\vec{a}^{n+1}$ using time $n$ values for $q$, and predicted values for other quantities.
11. Integrate the new velocity $\vec{U}^{n+1}$by using the velocity $\vec{U}^n$, the time step $\Delta t$, and the averaged value of the accelerations $\vec{a}^n$ and $\vec{a}^{n+1}$:

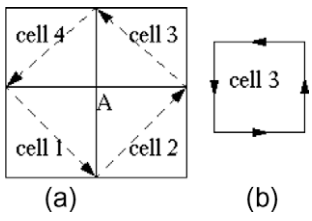


**Fig. 1.** Integration paths for the node and cell components.

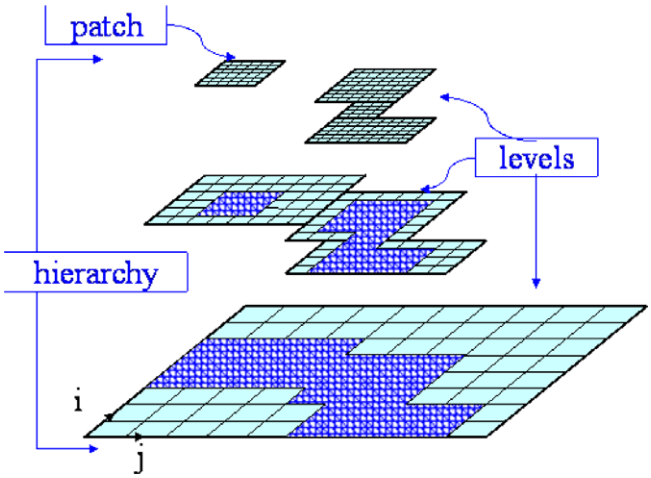


**Fig. 2.** A simple, three-level SAMRAI grid hierarchy.

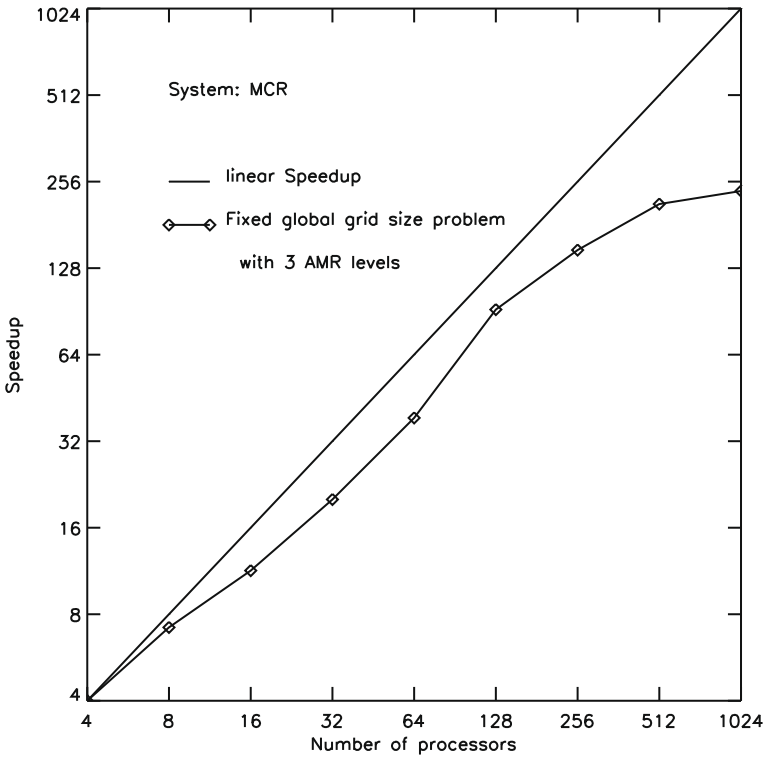


**Fig. 3.** Speedup of a fixed global problem of the ALE–AMR method.

$\vec{U}^{n+1} = \vec{U}^n + 0.5\Delta t(\vec{a}^n + \vec{a}^{n+1})$.

12. Integrate the new position $\vec{X}^{n+1}$ by using the position $\vec{X}^n$, the time step $\Delta t$, and the averaged value of the velocities $\vec{U}^n$ and $\vec{U}^{n+1}$:

$\vec{X}^{n+1} = \vec{X}^n + 0.5\Delta t(\vec{U}^n + \vec{U}^{n+1})$.

13. Compute the new strain rate from the new velocity.
14. Compute the new zone volume $V^{n+1}$ and the new density $\rho^{n+1}$.
15. Compute the new internal energy using the new and old volume, the averaged value of the predicted and old pressures, the new strain rate, the predicted stress:

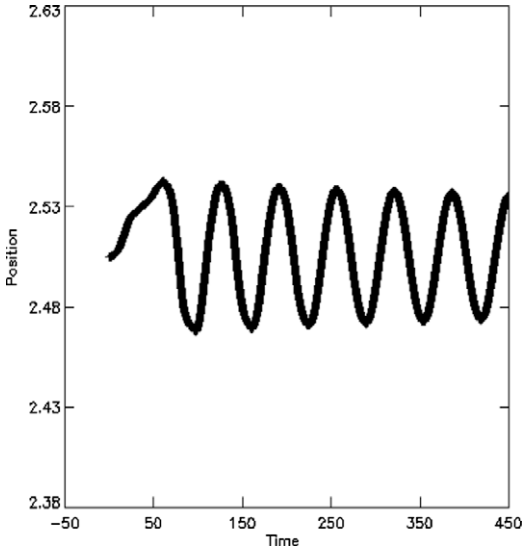


**Fig. 4.** Position vs time of a point on center of a vibrating plate clamped at both ends with mesh $2 \times 12$.

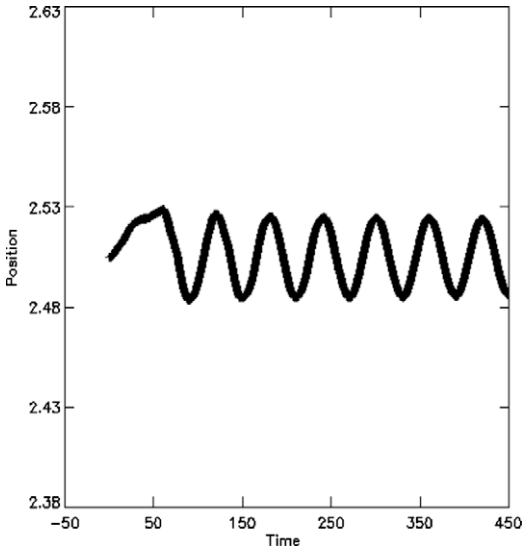


**Fig. 5.** Position vs time of a point on the center of a vibrating plate clamped at both ends with mesh $8 \times 48$.

$$e^{n+1} = e^n - \{0.5(P^n + P^p) + q^n\}/(1/\rho^{n+1} - 1/\rho^n) + 0.5\Delta t(V^{n+1} + V^n)(s_{11}^p \dot{\epsilon}_{11}^{n+1} + s_{22}^p \dot{\epsilon}_{22}^{n+1} + s_{33}^p \dot{\epsilon}_{33}^{n+1} + 2s_{12}^p \dot{\epsilon}_{12}^{n+1} + 2s_{23}^p \dot{\epsilon}_{23}^{n+1} + 2s_{31}^p \dot{\epsilon}_{31}^{n+1}).$$

16. Integrate the new stress and the equivalent plastic strain from the new strain rate.
17. Apply JC failure model (9) and (11) and set stress to zero if the failure criteria are reached.
18. Compute the new pressure using the new density and the new energy.
19. Advection phase (option).

The momentum conservation equation is explicitly used for computing the acceleration, and the internal energy equation is also explicitly used in steps 6 and 15. However, the conservation of mass equation is not used explicitly anywhere. This is because in the Lagrangian formulation of the hydrodynamic equations, conservation of mass is implicitly used because the mass of a zone is constant. This is used in the steps 5 and 14.

Several boundary conditions are considered for our present study to handle various applications including fixed, free-surface and mixed boundaries. Ghost cells are created by a mirror reflection across the boundary. For the fixed-boundary

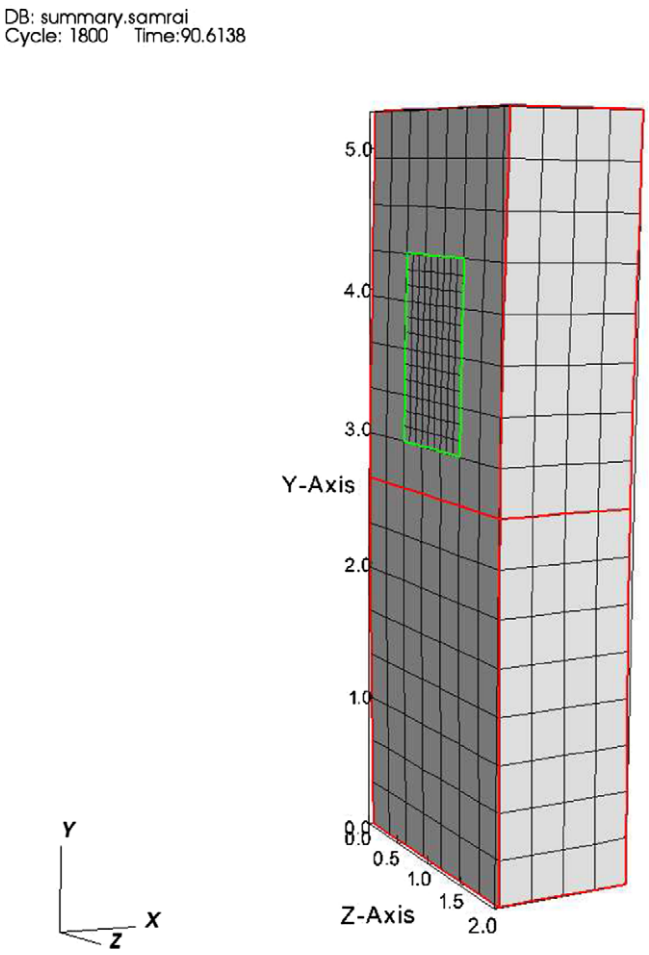


**Fig. 6.** 3D vibrating plate clamped at both ends with a static AMR mesh at time $t = 90$ μs.

calculations, reflected cells are applied for the ghost cells in the computation while all quantities in the ghost cells are taken as zero for the free-surface boundary.

At the end of a Lagrange step, it may happen that the grid becomes distorted or tangles as the grid moves together with the material. At such point, effective smoothing algorithms need to be used to improve mesh quality. A detailed description of equipotential methods can be found in [9]. Once a relaxed mesh is defined, the solution from the Lagrange grid needs to be transferred to the relaxed grid, also known as the advection step. In this step, the mass flux between the Lagrange mesh and the relaxed mesh is calculated, and the advection of mass, momentum, energy, and other quantities is performed. A detailed report of the advection method used for applications in gas dynamics can be found in [3].

For elastic–plastic flow problems, we need to consider how to advect the stress deviator tensor and the equivalent plastic strain, in addition to the usual quantities. In elastic–plastic flows, history variables are often retained since important processes, such as material failure, do not depend solely on the previous time step. The above procedures are not easily generalized for the purpose of evaluating those history variables required for material models since there are no conservation principles on which to base an advection scheme, such as for mass or energy. Plus, the stress fields are not a unique function of the strain fields except in simple cases. Based on those considerations, we decide to advect the stress deviator tensor and plastic strain directly from a Lagrange mesh to a relaxed mesh. The deviatoric stress is constrained to lie on or interior to a yield surface that is defined in terms of these history variables. There is no guarantee that the deviatoric stress satisfies the yield surface constraint after the advection. To resolve the inconsistency between the yield stress and the current stress, recalculation of the second invariant of the stress tensor and the yield condition after the advection is performed. If necessary, the stress deviators are rescaled back to the yield surface as described in Eq. (8) by computing the von Mises yield condition. If $2J - 2/3Y^2 > 0$, each of the stresses is multiplied by $\sqrt{2/3}Y/\sqrt{2J}$.

Here we use a conservative advection scheme to interpolate the solution obtained by a Lagrange step from the grid defined by $X^{old}$ onto the relaxed grid defined by $X^{new}$. We formulate this interpolation as an initial value problem in which we solve

$$\frac{\partial \phi}{\partial \tau} = 0 \tag{12}$$

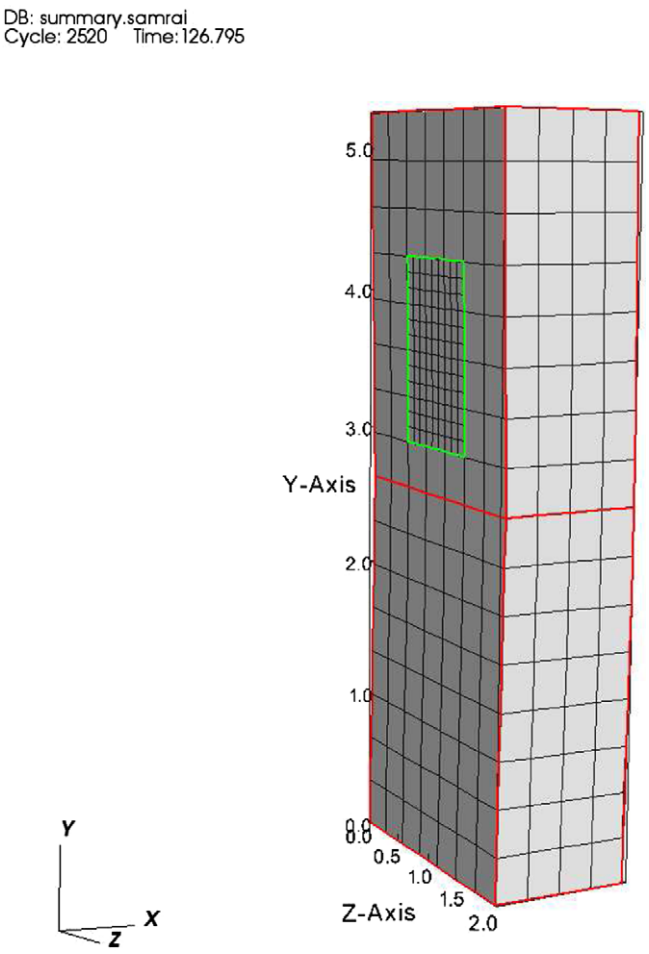


**Fig. 7.** 3D vibrating plate clamped at both ends with a static AMR mesh at time $t = 126$ μs.

on a moving grid. $\phi$ and $\tau$ denote an arbitrary scalar quantity defined on the grid and a pseudo-time, respectively. At $\tau = 0$. $\phi$ is defined by its value at the end of the Lagrange step. We integrate (12) from $\tau = 0$ to $\tau = \tau_1$. Over this time, the grid moves from $X^{old}$ to $X^{new}$ with a grid velocity $\mathbf{s} = (\mathbf{x}^{new} - \mathbf{x}^{old})/\tau_1$. For two dimensional problems, we now transform (12) to space coordinates $\xi, \eta$ and obtain the equation

$$\frac{\partial A\phi}{\partial \tau} - \frac{\partial}{\partial \xi}(\mathbf{n}^{\xi}.\mathbf{s}\phi) - \frac{\partial}{\partial \eta}(\mathbf{n}^{\eta}.\mathbf{s}\phi) = 0, \tag{13}$$

where $A = x_\xi y_\eta - y_\xi x_\eta$, $\mathbf{n}^{\xi} = (y_\xi, -x_\eta)$, and $\mathbf{n}^{\eta} = (-y_\xi, x_\eta)$. We now solve (13) with a conservative difference scheme. We define that transport volumes associated with four edges are the signed quadrilaterals formed by the old and the new edges as $F_L, F_R, F_B, F_T$. Using these definitions, we outline the advection algorithm for the stress and the equivalent plastic strain as the following:

1. Compute transport volumes associated with all edges.
2. Compute the stresses and the equivalent plastic strain at all edges $\phi_L, \phi_R, \phi_B, \phi_T$ using Colella's formulation [7] with van Leer limiters [13].
3. Advect the stresses and the equivalent plastic strain by

$$V^{new}\phi^{new} = V^{old}\phi^{old} + F_L\phi_L - F_R\phi_R + F_B\phi_B - F_T\phi_T, \tag{14}$$

$$V^{new} = V^{old} + F_L - F_R + F_B - F_T. \tag{15}$$

4. Recalculate the second invariant of the stress tensor and the yield condition. If necessary, rescale the stress deviators back to the yield surface as described in the (8) and update the plastic strain.
5. Compute the pressure field on the relaxed mesh after all other variables have been advected.

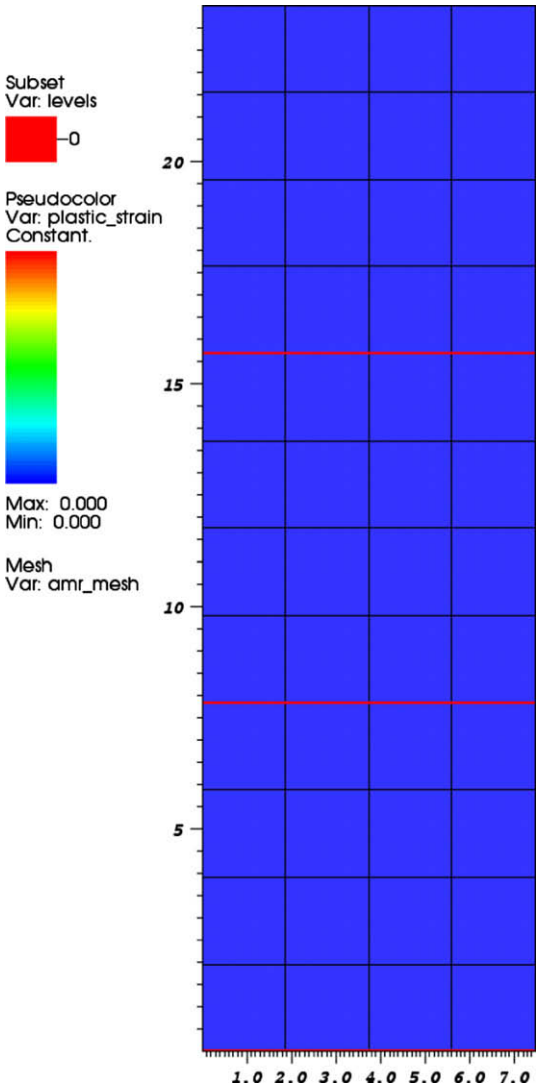


**Fig. 8.** The equivalent plastic strain of the 2D impact problem at time $t = 0$.

This method has been extended to the JC material failure model by using the above advection algorithm. The damage field is advected to the relaxed mesh, then the history variable failure $(F)$ is updated by Eq. (11). An alternative way to deal with the history variables in the JC material failure model is to recalculate damage and failure variables on the relaxed mesh directly using Eqs. (9) and (11) after all other quantities have been advected. Since solving those equations requires less amount of work, this approach is more computationally efficient than using the advection algorithm.

## 4. Arbitrary Lagrangian Eulerian and adaptive mesh refinement

Berger and Oliger [2] implemented their AMR algorithm in one and two dimensions for hyperbolic conservation laws, and the solution is defined on all cells. This method was later extended to higher dimensions in a straightforward way in the Eulerian frame [14].

For applications with an ALE formulation, there are several problems that need to be solved if the Berger–Oliger Method is to be applied. First, the ALE–AMR mesh employs a moving mesh, not a fixed Eulerian mesh like the traditional AMR method. Second, there is an advection step with the ALE method that is not in the traditional AMR method. Third, the ALE–AMR solutions are defined on a staggered grid while traditional AMR works on a cell-centered grid. These uncertainties were first investigated for gas dynamics applications in [3].

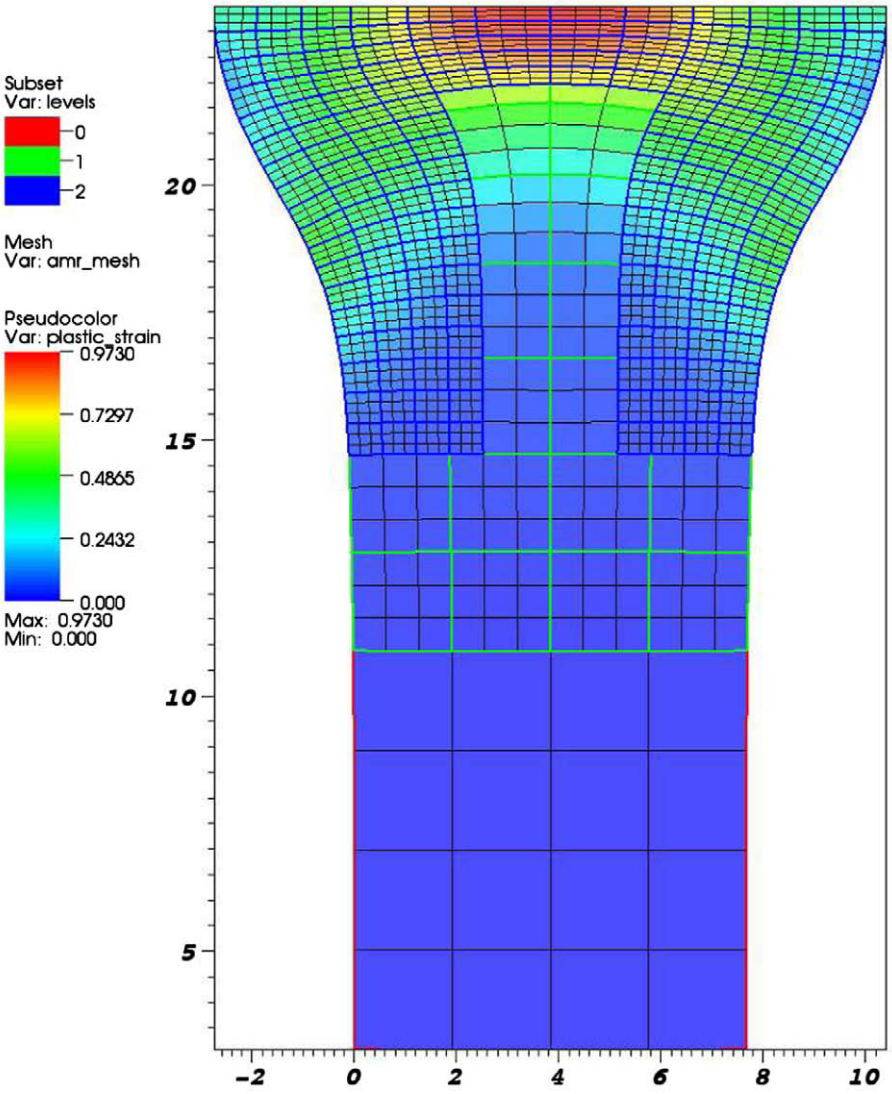


**Fig. 9.** The equivalent plastic strain of the 2D impact problem at time $t = 15$ μs.

In the present paper, we give details of our investigations of ALE–AMR method for solid mechanics with a material failure model. Although some of the approaches in the ALE–AMR method for gas dynamics can be used, there are new challenging issues in computational solid mechanics and materials modeling with an ALE–AMR formulation. New history variables from the solid mechanics and material failure model such as the stress, the equivalent plastic strain, and the damage and failure variables are not defined by conservation laws. Our study focuses on how to interpolate these variables in multiple meshes, how to deal with the interlevel boundary conditions for these new quantities, and how to find effective error indicators for those new applications.

In general, similar to gas dynamics, an ALE–AMR method for materials modeling has several main components: Lagrangian AMR hierarchy integration, solutions on multiple meshes, interlevel interpolations, construction of boundary conditions, relaxation and advection, and criteria for refinement and coarsening. In the following section, we report our approach regarding these issues linked with solid mechanics and materials modeling with the JC failure model.

### *4.1. Lagrangian AMR hierarchy integration*

For a purely Lagrangian method (without the advection option), an explicit time-marching method of a general hierarchy refinement for solving (1)–(11) with an ALE formulation can be expressed as a traditional AMR-hierarchy time step. Details of AMR hierarchy integration can be found in [14].

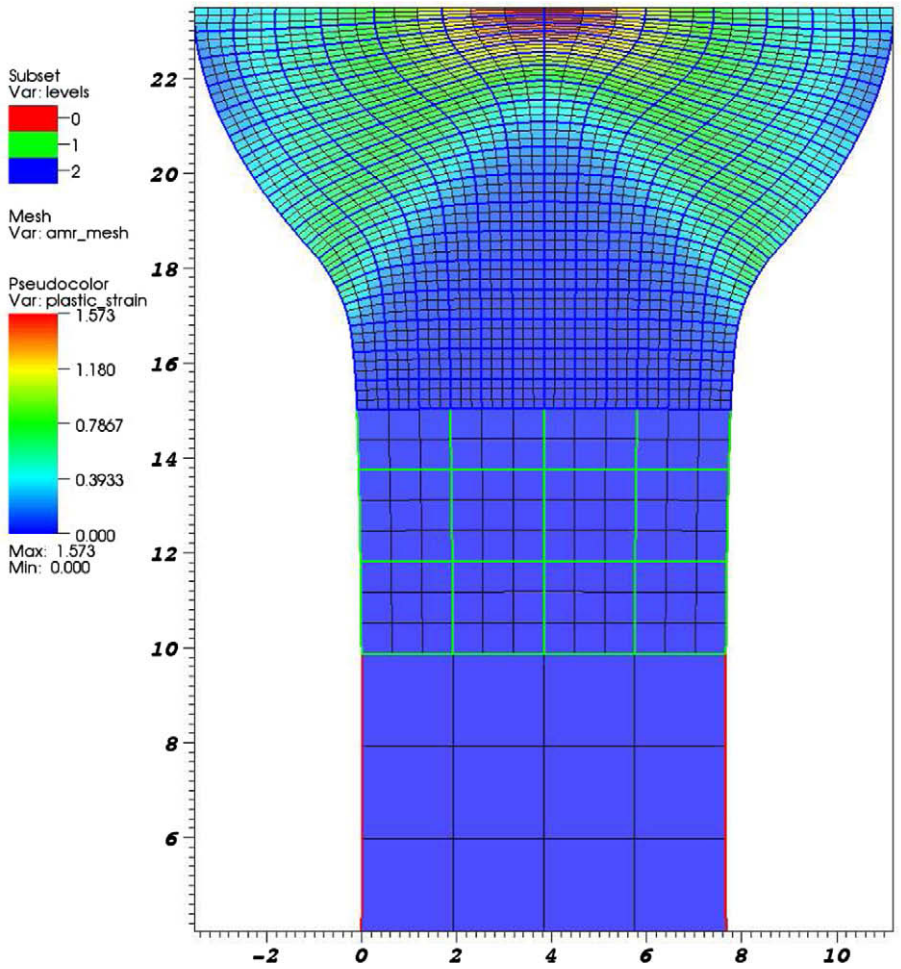


**Fig. 10.** The equivalent plastic strain of the 2D impact problem at time $t = 30$ μs.

### 4.2. Refinement and coarsening

Interlevel-solution transfer operators have two classes: refinement operators for defining fine grid data, and coarsening operators for the reverse. Refinement operators are required when new grids are created while coarsening operators are required for synchronizing coarse and fine data in hierarchy. All variables in the equations have their own operators, which include the nodal positions.

Several principles to design these operators for solid mechanics and materials modeling are considered: (a) constant field preservation, (b) 2nd-order accuracy, (c) monotonicity, (d) local conservation, (e) exact inversion of refinement by coarsening, and (f) positivity preservation. Some of these principles were successfully applied to an ALE–AMR method for gas dynamics simulations in [3]. Here we focus on elastic–plastic flows and materials modeling.

With these design goals in mind, we consider a 3D coarse element (cell or node) that has a value $\phi_0$ and derivatives $\phi'_{x_0}, \phi'_{y_0}, \phi'_{z_0}$ for each coordinate direction. With a constant refinement ratio $r = (r_1, r_2, r_3)$ for each direction, the following interpolation for $\phi$ on a fine grid $I$ from a coarse grid $I-1$ at $(x_0, y_0, z_0)$ is constructed with a second-order accuracy:

$$\phi(x,y,z) = \phi_0 + \phi'_{x_0}(X_1 - 0.5\Delta x_0) + \phi'_{y_0}(X_2 - 0.5\Delta y_0) + \phi'_{z_0}(X_3 - 0.5\Delta z_0). \tag{16}$$

The primitive variables from (1)–(11) and the basis are

$$\phi = (\rho, u, v, w, E, \mathbf{s}, \epsilon^p, D, F_v), \tag{17}$$
$$X = (V, \tilde{m}, \tilde{m}, \tilde{m}, m, V, V, V, V), \tag{18}$$

where $V$ is cell volume, $\tilde{m}$ is nodal mass, $m$ is cell mass, and $E$ is the total energy. This construction for $\phi$ on a fine grid is conservative of $\phi X$ as

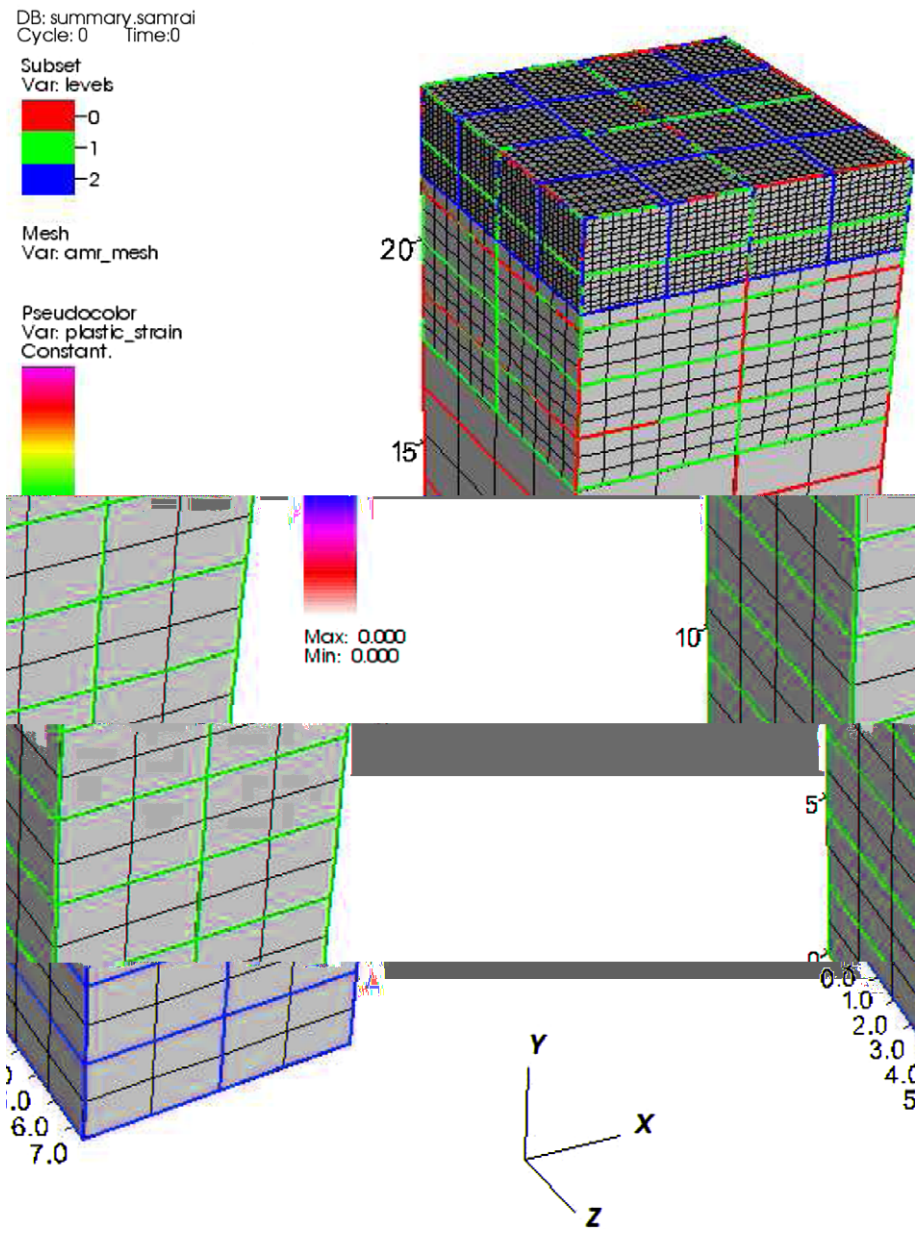


**Fig. 11.** The equivalent plastic strain of the 3D impact problem at time $t = 0$ μs with a three-levels static AMR mesh.

$$\sum_{i,j,k}^{r_1,r_2,r_3} \phi^l X^l = \phi^{l-1} X^{l-1} \tag{19}$$

if the following basis consistency condition holds:

$$\sum_{i,j,k}^{r_1,r_2,r_3} X^l = X^{l-1}. \tag{20}$$

This relationship will be shown to be fundamental to a number of important properties of the interlevel operators. In a constant field, all derivatives of $\phi$ are zero, and constant fields are preserved provided the basis consistency condition (20) holds. In order to address our design principal (c), the monotonicity, we use the well-known van Leer limiter [13] for computing the derivatives. To prevent oscillations in the primitive variables $\phi = (\rho, u, v, w, E, \boldsymbol{s}, \epsilon^p, D, F_v)$, the interpolation basis (18) is re-

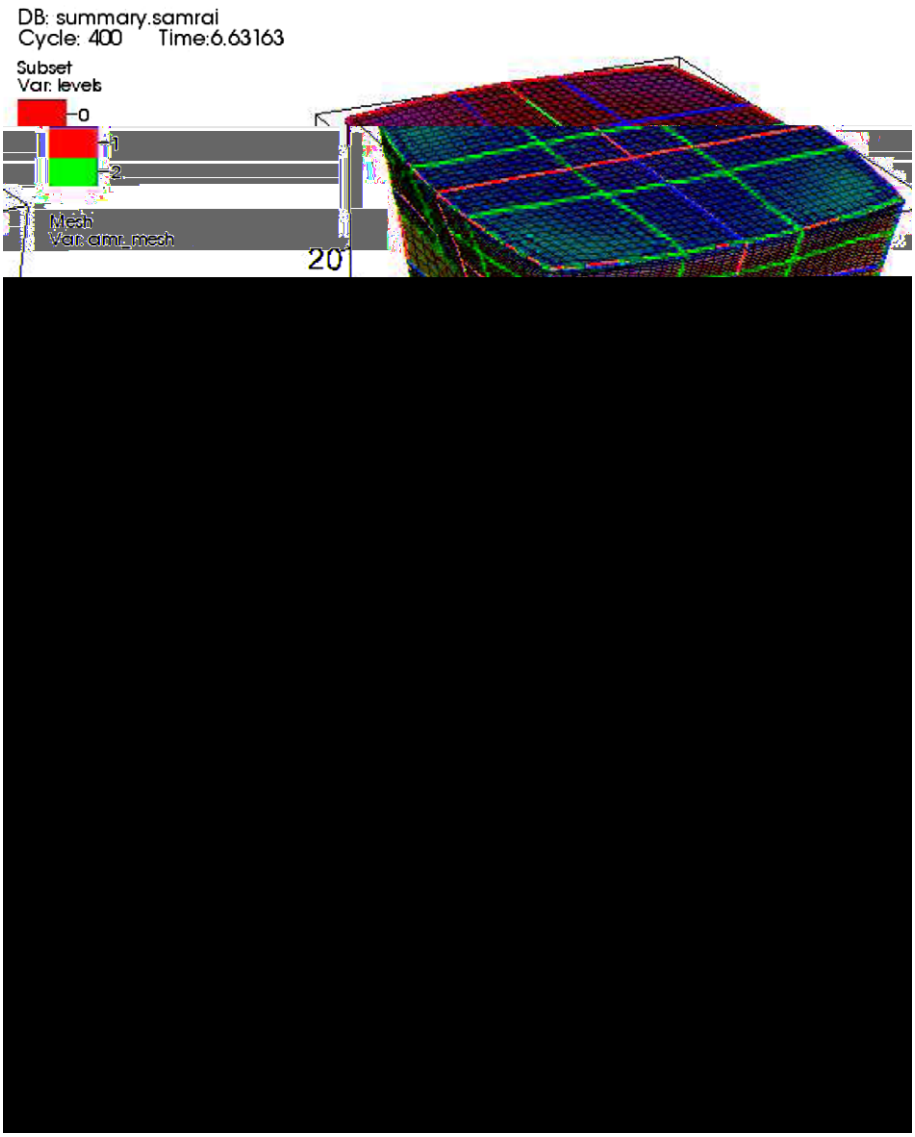


**Fig. 12.** The equivalent plastic strain of the 3D impact problem at time $t = 6.6$ μs.

quired to obtain property (d), local conservation. A simple way to ensure that (e) is easily achieved is to maintain an exclusive $r : 1$ correspondence between fine- and coarse-node data locations, where $r$ is the refinement ratio. By choosing $r$ odd, any conservative distributions of a quantity (cell- and node-centered) from the coarse mesh to its corresponding fine-mesh stencil can be inverted exactly with a simple summation. Internal energy is constructed, as in the advection procedure, from the interpolated total energy and kinetic energy of the interpolated velocity field. For the new variables $\boldsymbol{s}$, $\epsilon^p$, $D$ and $F_v$ required for elastic–plastic flows and material failure model, $V$ is used for the basis to be in line with these design goals, and (f) is enforced by adding an extra constraint to the interpolated field that requires positivity preservation, such as the plastic strain and damage fields. Here, we did not use interlevel operator for the variable failure $F$. Since this variable depends on the damage variable, we just compute it once the damage field is interpolated in a new mesh. This simple calculation will be more accurate (no accuracy loss due to AMR) and improve computational efficiency (one code line as if damage is great than 1, set failure equal to 1) than using an interlevel operator for this particular case.

After the interlevel operations, history variables in nonlinear constitutive models for the strength of materials encounter a similar problem to the one in the advection phase. The deviatoric stress might not satisfy the yield surface constraint after the interpolation since the interpolated stress deviators are not determined by constitutive relations. We apply a simple strategy – let the constitutive model resolve the discrepancy during the next Lagrange step. Through a series of numerical experiments we find our current approach for those history variables gives reasonable numerical results.

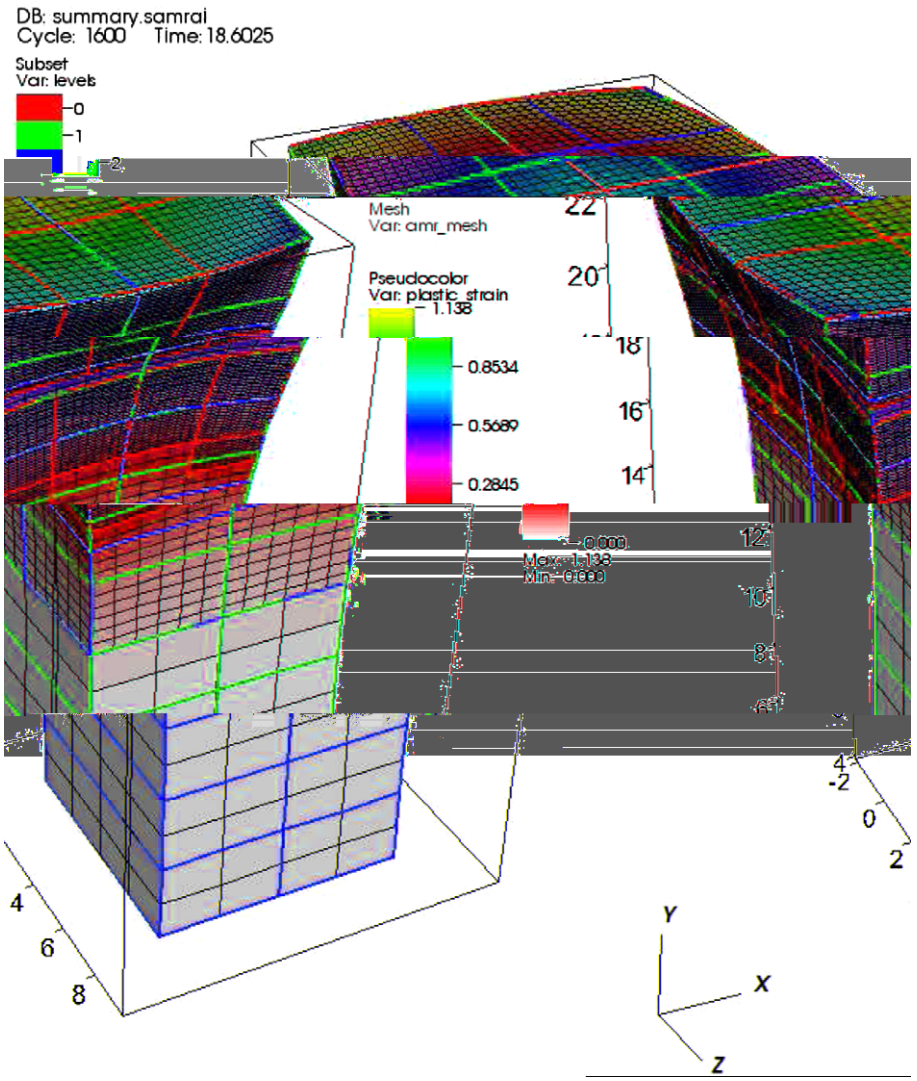


**Fig. 13.** The equivalent plastic strain of the 3D impact problem at time $t = 18.6$ μs.

The design goal of monotonicity is enforced by applying the van Leer limiters. For one-dimensional problems, the monotonicity preserving schemes are positivity preserving [13]. We applied the limiters to higher-dimensional problems and found that positivity is not automatically preserved in the plastic field and small negative values appear at some test cases. In this case, we have to enforce the positivity by inserting another constraint for the plastic field after the interpolation – if the value of the interpolated plastic variable is negative, set the plastic variable equal to zero. This simple constraint works well for our problems. Further study of alternative limiters with a positivity preservation property for higher-dimensional problems need to be investigated.

The coarsening operation begins with injection of nodal position $X^{l-1}_{i,j,k} = X^{l}_{r1,r2,r3}$ for the mesh. Once the basis consistency condition (20) is satisfied, we can choose the following coarsening operator:

$$\phi^{l-1} = \frac{\sum_{i,j,k}\phi^{l}X^{l}}{\sum_{i,j,k}X^{l}} = \frac{\sum_{i,j,k}\phi^{l}X^{l}}{X^{l}}. \tag{21}$$

For history variables, we chose volume $V$ for the basis to be consistent with our approach in the advection phase. During the fine level advance, nodes on the fine level move unconstrained by the coarser levels except on the interlevel boundary. After the initial nodal injection, one or more Lagrangian advances will make the fine and coarse meshes no longer spatially align at intermediate nodes and/or edges. Obviously, the consistency condition in the case of volume, for example, will not generally be satisfied after one or more Lagrange steps, i.e., the sum of the fine volumes corresponding to an underlying coarse cell will

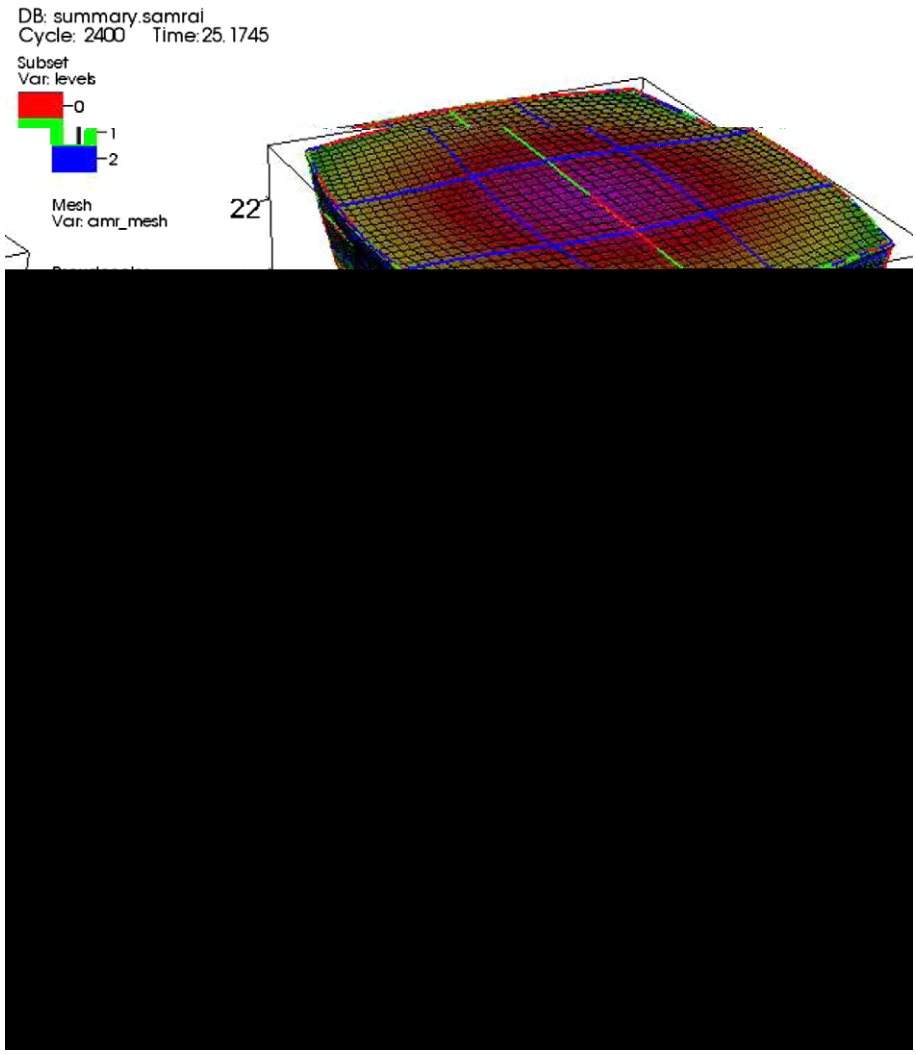


**Fig. 14.** The equivalent plastic strain of the 3D impact problem at time $t = 25.17$ μs.

**Table 1**
Computational efficiency and accuracy comparisons between the results from a single mesh $36 \times 180 \times 36$ simulation and the one from a three-level AMR mesh with an initial coarse mesh $4 \times 20 \times 4$.

| Impact problem | Single mesh | AMR mesh | Differences |
|---|---|---|---|
| Total CPU time (s) | 39,145 | 22,155 | 41% |
| The maximum $\epsilon^p$ | 1.186 | 1.198 | 1% |

not equal to the coarse volume. To insure the coarsening operator to be both constant field preserving and conservative, an intermediate step in the coarsening process is constructed: the solution from the Lagrange fine mesh is advected to a fine mesh that is constructed from linear interpolation from the underlying coarse mesh. The interpolated fine mesh satisfies the volume consistency condition by construction. Since the advection procedure itself is globally conservative and the consistency condition is satisfied, this alternative coarsening procedure is also conservative.

### 4.3. Boundary conditions

In the hierarchy, boundary conditions on a fine grid are constructed from linear interpolation from a coarse level. Once the boundary conditions are obtained, the solution on each level is independently advanced to the next time step based on the numerical scheme. Since the mesh is moving with the material, the coarse- and fine-boundary nodes generally will not stay

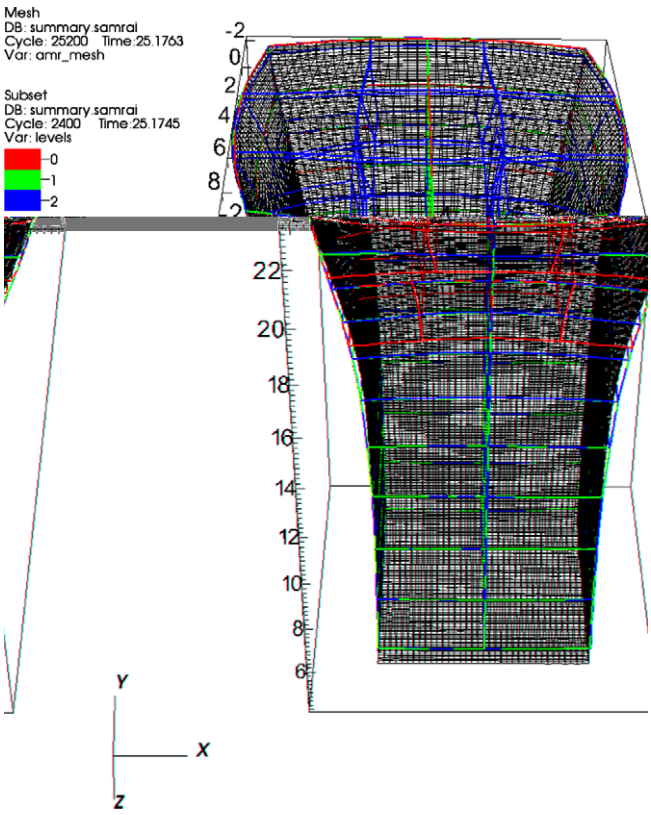


**Fig. 15.** Simulations of the implant plate at time $t = 25.17$ μs with both a single fine mesh (gray) and a three-level AMR mesh (color). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

aligned after the Lagrange step. Sometime, the non-quad or non-hex elements will form on the boundary, which is not allowed with our present numerical algorithm. Therefore, careful treatment of the boundary conditions on a composite mesh is needed to ensure that all elements are quads or hexes. This is enforced by using linear interpolation both in space and time for the positions of boundary nodes from the next coarse level, as well as other quantities in ghost regions that overlap the interlevel boundary area with two ghost cells in a composite mesh.

If an AMR mesh's boundary is a physical boundary, a physical boundary condition will be applied directly to the numerical scheme rather than using the above interpolation method. Appropriate boundary conditions are applied to the acceleration field. A reflection boundary condition is obtained by setting equal to zero the normal component of accelerations of a surface point when it points into the reflection surface. For a free-surface condition, all quantities including the stress field and the plastic strain field associated with the ghost cells neighboring the physical boundary are taken as zero. In our present study, one material is considered, and the material boundaries are in line with the computational domain boundaries so no interface treatment is needed. In our future study of multi-materials flow modeling, we will investigate our ALE–AMR method with interface reconstruction.

As we have discussed earlier, the mesh relaxation for our present ALE–AMR method plays two roles. For a pure ALE method, the mesh relaxation will prevent severe mesh distortion. Another role for the ALE–AMR method is in the coarsening procedure required to satisfy our ALE–AMR design principals. The relaxation is executed on a composite mesh, which has mixed fine and coarse grids after all finer levels have been advanced to the same simulation time with the coarse level. Then an advection procedure is performed to obtain the solution on the relaxed mesh from the original Lagrange mesh.

### 4.4. ALE–AMR criteria

Gradient detectors $C_\alpha$ are used for the selection of refinement and coarsening:

$$C_\alpha = \left\| \frac{\phi_{\alpha+1} - 2\phi + \phi_{\alpha-1}}{max(\phi_0, \epsilon_0)} \right\|_{L_2}, \tag{22}$$

where $\epsilon_0$ is a small constant to prevent blowup, and $\phi_0$ is a global normalization constant. In our present study of elastic flows, we use the internal energy field for the criteria since it has information contributed from multiple fields including

the pressure and the stress terms. Based on several test cases, our current approach has been proven to be a good choice for pure elastic flow problems. For elastic–plastic flow applications, we did more numerical experiments using a couple of different fields in (22)

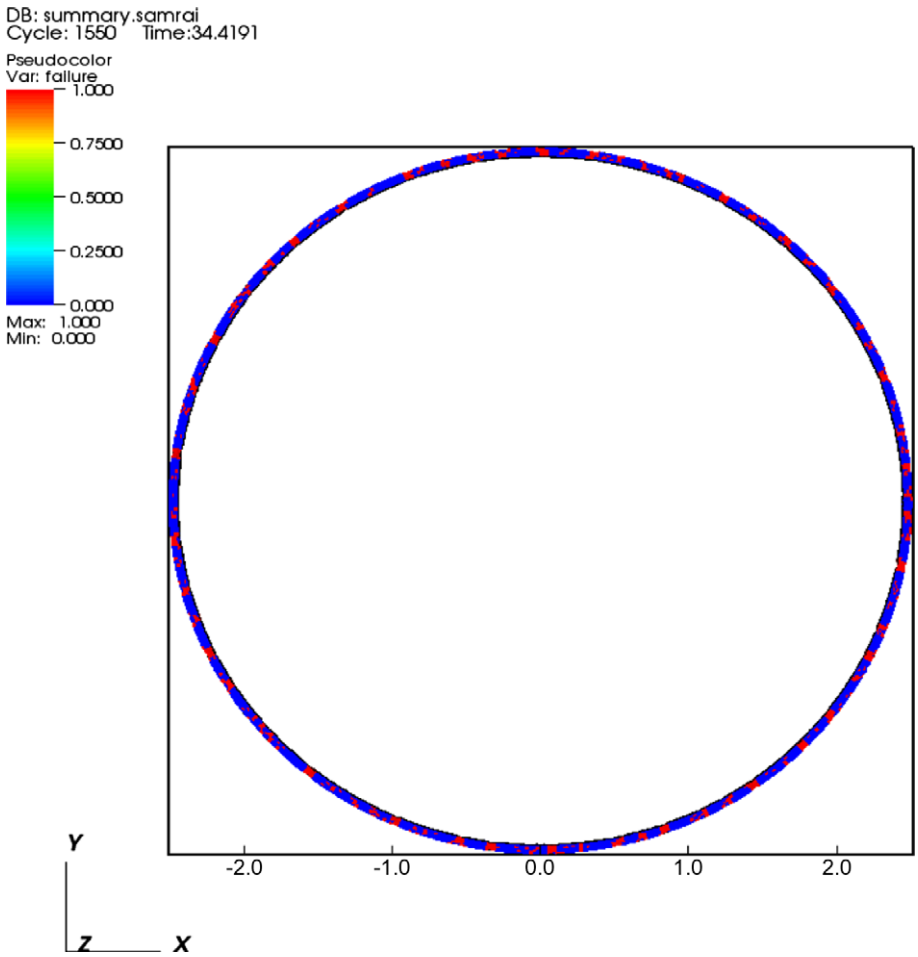


**Fig. 17.** Material failure of the expanding ring at 34 μs by ALE–AMR.

If an Eulerian calculation is desired, multiple cycles of relaxation due to the stability constraint of the advection procedure may be needed to return the mesh to its original configuration.

## 5. Implementation

Implementation of our ALE–AMR method is on parallel systems using C++, FORTRAN 90, and MPI. Structured adaptive mesh refinement application infrastructure (SAMRAI) by C++ [15], an object-oriented framework for the development of the structure-grid, has been used for our present study as a part of our software to handle the data management of the AMR algorithm in hierarchy and parallelization. SAMRAI is a particular approach to adaptive mesh refinement in which the computational grid is implemented as a collection of structured mesh components. The computational mesh consists of a hierarchy of levels of spatial and temporal mesh resolution. Typically, each level in the hierarchy corresponds to a single uniform degree of mesh spacing for a numerical method. Within a SAMRAI hierarchy in Fig. 2, levels are nested; that is, the coarsest level covers the entire computational domain and each successively finer level covers a portion of the interior of the next coarser level. Computational cells on each level are clustered to form a set of logically-rectangular patch regions. Simulation data are stored on these patches in contiguous arrays that map directly to the mesh cells without excessive indirection.

On parallel systems, these patches are distributed to processors to balance computational workload by constructing even work-loaded boxes that cover all patches at each AMR level and sending those boxes evenly to all processors. In general, each processor has multiple patches, and each level is load balanced separately. More detailed studies and discussion of SAMRAI parallel performance and scalability can be found in [16]. The most expensive core computations of the ALE–AMR method are implemented in FORTRAN 90. This dual-language choice takes the advantages of both to maximize the computational efficiency of FORTRAN 90 and to manage the data complexity of the ALE–AMR method by C++ at the same time.

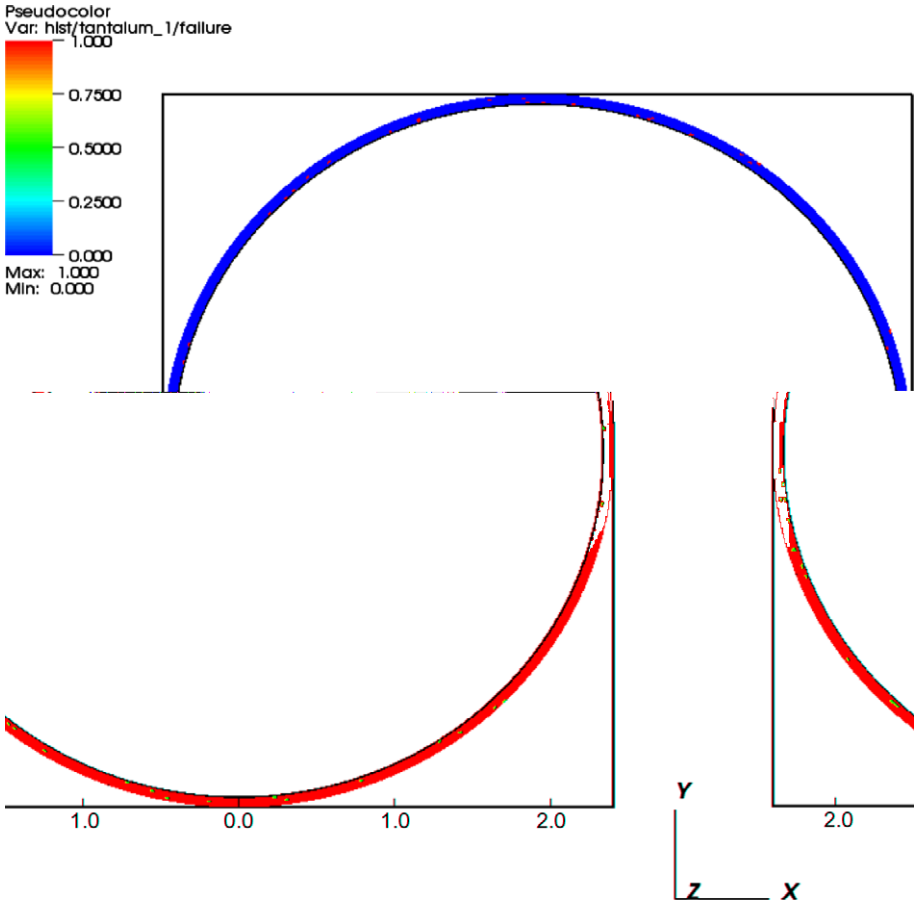


**Fig. 18.** Material failure of the expanding ring starts at 30 μs by ALE3D.

We have carried out a parallel efficiency study for our present implementation. An elastic–plastic flow problem – a metal rectangular box with a constant velocity $V_0$ moving toward a rigid wall – is considered. The impact of the rectangle on a rigid wall is simulated, and plastic deformation is expected from this motion. A fixed global size of the impact problem on a mesh $16 \times 64 \times 16$ with three levels of AMR meshes was designed for the efficiency study. There are about $300k$ total grid points, and they will reduce to 300 grid points for each processor when a 1024-processors system is used. The MCR system at Lawrence Livermore National Laboratory (LLNL), a 1112 dual-node Linux cluster with 2.4 GHz for each node, was used for our current simulations.

Fig. 3 shows the speedup of the parallel ALE–AMR for the fixed-size test problem on the MCR system. It gives an excellent speedup vs. the number of processors, and linear scalability is achieved on 128 processors. Since the AMR method will add extra overhead time to a pure ALE code due to the clustering operations performed during adaptive gridding operations in patch-based structured adaptive mesh refinement applications, we have separate timers to record the time on each major component of our code. For the above application, a total 10–14% AMR time (12% for the mesh refinement process and 2% for the coarsening operation with 128 processors) was noticed when we varied the number of processors. The 4% variation from a small number of processors $(O(10^1))$ to a medium number of processors $(O(10^2))$ indicates that a future scalability study is needed for the ALE–AMR method when a large number of processors is used for large-scale applications. More recent scalability study of the patch-based AMR method for machines of up to $(O(10^4))$ can be found in [17].

## 6. Numerical simulations

Numerical tests for various elastic–plastic applications and material failure problems to verify our ALE–AMR capability are described in this section.

Several perfect elastic problems are studied, and the numerical results are checked with theoretical work. The first test case for purely elastic problems is a 2D calculation of a vibrating elastic plate clamped at the two ends. The plate is described with a simple EOS, $P = 1.88(\rho/\rho_0 - 1)$ Mbar. Other constants are $\rho_0 = 7.72$ g cm$^{-3}$, $\mu = 0.814$ Mbar, and $Y^0 = \infty$. The plate

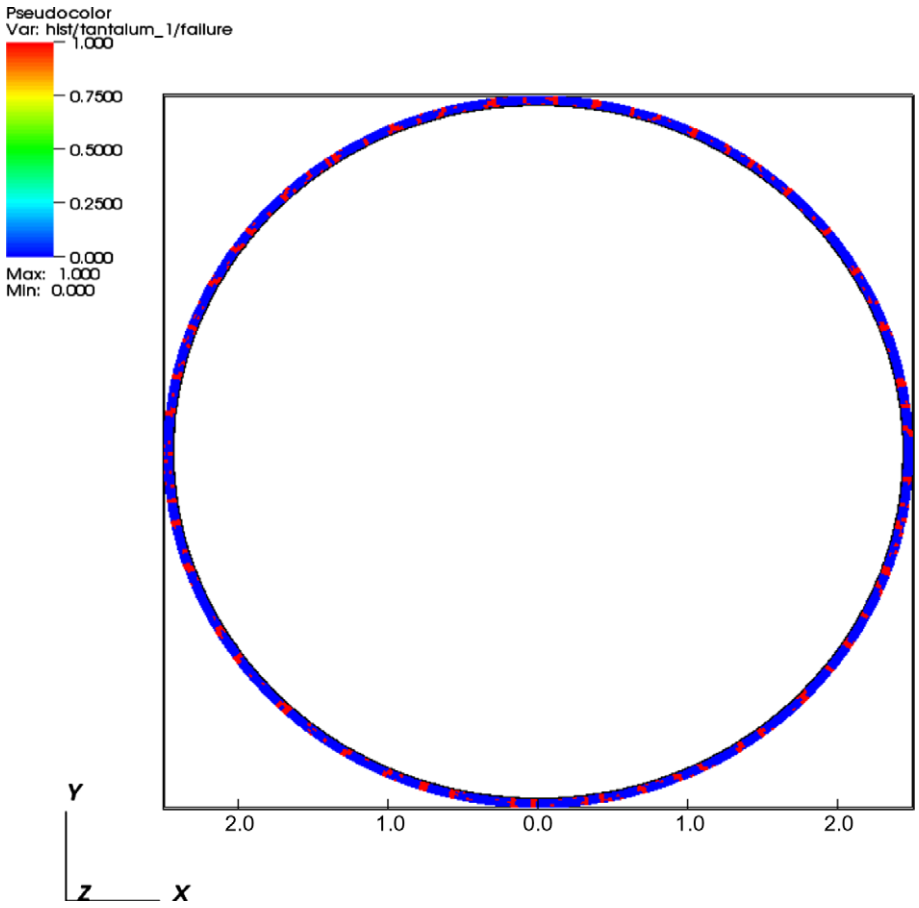


**Fig. 19.** Material failure of the expanding ring at 34 μs by ALE3D.

length is 5.25 cm and the plate thickness is 1 cm. The motion is started by assigning a velocity of $v = 10^{-3}$ cm/μs for a time of $t = 50$ μs to the point at the center of the left side. The position-time plots of this point with different meshes are shown in the Fig. 4 with a single mesh $2 \times 12$ and Fig. 5 with a single mesh $8 \times 48$. It can be readily verified that the vibration period is converging to the correct solution for the fundamental frequency 51 μs with the increase of higher resolutions.

A similar computation for a 3D vibrating elastic plate (1.0 cm × 5.25 cm × 2.0 cm) is carried out with a small stationary AMR patch with a refinement ratio $r = 1 : 3$ added in a coarse mesh $4 \times 16 \times 7$. Figs. 6 and 7 show the displacements at different times. Again, the simulation data are in good agreement with the fundamental frequency, and the static ALE–AMR method works well with a moving mesh. Numerical simulations of a truly dynamic ALE–AMR method for elastic–plastic applications are given below, and an efficiency study of this method with the applications is discussed.

The impact of a metal rectangular plate (for 2D: 7.64 mm × 23.47 mm) and a box (for 3D: 7.64 mm × 23.47 mm× 7.64 mm) on a rigid wall provides an excellent test case for the ALE–AMR method, and it also provides a calculation to test the plasticity aspect of the computer model.

We have carried out both 2D and 3D computations to demonstrate our ALE–AMR method. For this application, we use the constitutive model: EOS: $P = 0.76(\rho/\rho_0 - 1)$ Mbar; Density: $\rho_0 = 2.74$ g cm$^{-3}$; Shear modulus: $\mu = 0.248$ Mbar; and flow stress: $Y = 0.0046(0.008 + \epsilon_p)^{0.1}$ Mbar, where $\epsilon_p$ is the equivalent plastic strain. An initial velocity $V = 0.25$ km/s was applied. We use a coarse mesh $4 \times 12$ at $t = 0$, and use 3-levels dynamic AMR mesh with a gradient detector when $t > 0$. Here, we use a refinement ratio $r = 1 : 3$ for each direction. The bar remains stuck to the wall following impact and there is no friction. The 2D problem was studied by Wilkins [10], and we verify our results with his published solution.

For the 2D case, an initial coarse mesh is set at $4 \times 16$, and three levels of AMR meshes are applied with a gradient detector for the energy field. Figs. 8–10 show the equivalent plastic strain at different times. The deformation starts from the contact area and gradually affects the entire domain. The maximum equivalent plastic strain is in the center of the boundary at the rigid wall. The AMR meshes, indeed, simulate the dynamics of the motion gracefully – the mesh adjusts its resolution with the solution. From the point of view of computational efficiency, it does outperform the method with a uniform mesh since the AMR mesh saves a lot computation time on smooth regions that do not need a high resolution.

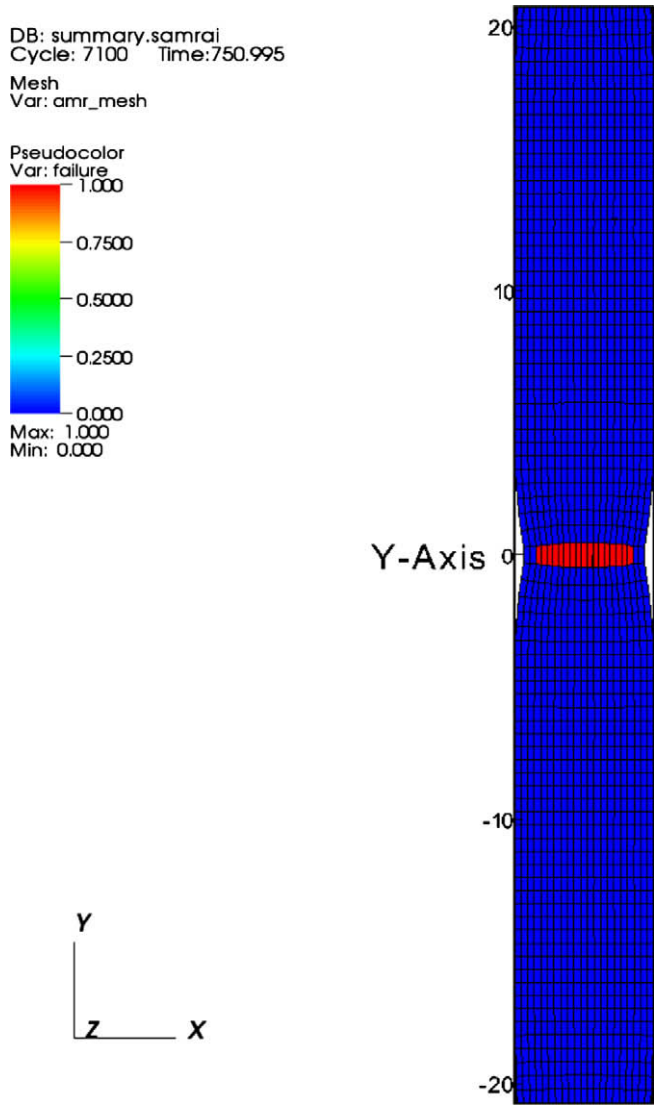


**Fig. 20.** Numerical results of the tensile plate with ALE–AMR at time $t = 750$ μs with a single mesh $20 \times 100 \times 2$.

For the 3D case, an initial coarse mesh is set at $4 \times 20 \times 4$ with a static three-level AMR mesh, and other physical parameters are the same as the 2D case. For this problem, we combine both a static three-level AMR near the wall at $t = 0$ and a dynamics three-level AMR governed by a gradient detector at $t > 0$. Again, we use a refinement ratio $r = 1 : 3$ for each direction for both static and dynamic AMR meshes. Since we know that there is extra overhead time added by the clustering operation of the dynamic AMR method and the most dynamic motion of the 3D impact problem is around the interface of the metal box and the rigid wall, to combine both a static AMR method and a dynamics AMR method for such a problem will take advantages of both approaches. Through some numerical experiments, we found this hybrid strategy is more efficient to attack this kind of applications than a pure dynamic AMR.

Figs. 11–14 show the equivalent plastic strain at different times, and a similar deformation pattern like 2D is obtained. Again, the ALE–AMR method improves computational efficiency. Since the maximum plastic deformation happens at the end of the box, which impacts the wall, a dense mesh is desirable in this region to model the dynamic behavior. Near the other end of the box, the change is relatively smooth so a low-resolution mesh can be used for the small strain region. Through the internal energy gradient detector as the refinement and coarsening criteria, effective AMR meshes are generated to accomplish the rapid change of the solution with time.

In order to study the efficiency of the ALE–AMR method, we perform more numerical experiments using both single-level meshes and AMR meshes for the above impact problem. Because we are interested in how much time we can save using an AMR method compared to a single mesh, we use the numerical data on a single fine mesh $36 \times 180 \times 36$ as our baseline data.

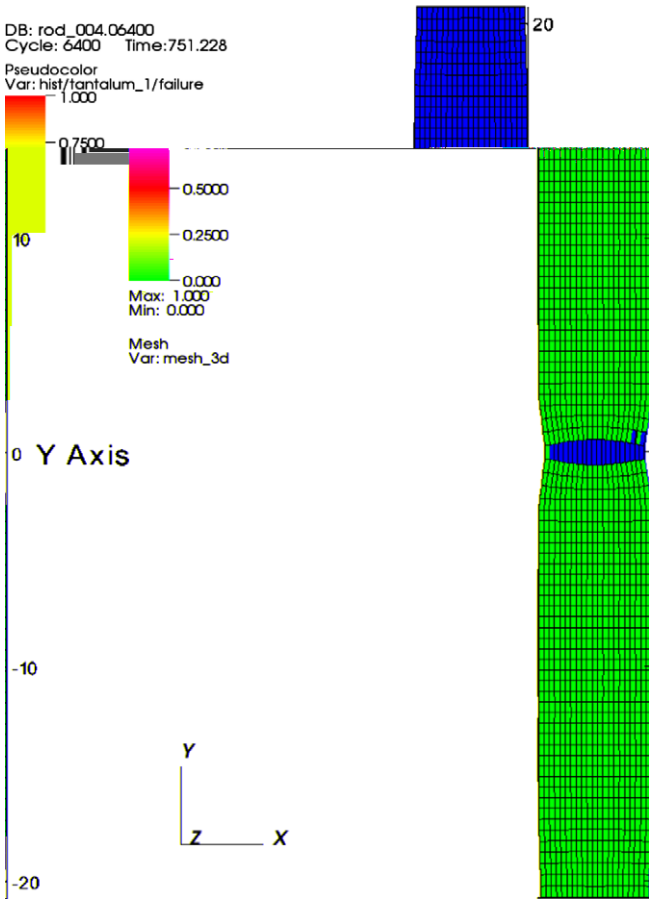


**Fig. 21.** Numerical results of the tensile plate with ALE3D at time $t = 750$ μs.

A static AMR mesh with an initial coarse mesh $4 \times 20 \times 4$ is set at $t = 0$ in Fig. 11, and three levels of AMR meshes are applied with a gradient detector for the internal energy field for $t > 0$.

A refinement ratio 1:3 is used at each refinement level. The numerical experiments are conducted on the MCR system with 16 processors. We run the impact problem to 30 μs and compare the CPU times and accuracies for both cases.

Obviously, we should note that the numerical results from an AMR mesh will never be exactly the same as the results using a single fine mesh, which is equivalent to the finest resolution of the AMR mesh. Now we are interested in how much accuracy and computation efficiency we gain when we use an AMR mesh rather than using a single mesh with a fine resolution. In Table 1 and Fig. 15, detailed comparisons are given. Table 1 lists total CPU times for using a single fine mesh and an AMR mesh and the maximum values of the equivalent plastic strain, and Fig. 15 shows deformations of the plate for both cases. The gray one is from the ALE–AMR simulation and the color one is from a fine single mesh. The discrepancy in Fig. 15 is about 1%, which is similar to the difference in the maximum values of the equivalent plastic strain in Table 1. Here, it is very clear that the AMR mesh takes the advantage of solving the nonlinear regions with higher resolutions and gives superior results compared to those of a single mesh, and at the same time, we save a significant amount of computation time – 41% less than the full-mesh calculation. The strategy of using a combination of a static AMR and a dynamic AMR using a gradient detector is a good approach for the impact problem since we do have a priori knowledge of the solution – most nonlinear phenomenon is near the top surface of the box, and this information can be used to create a static AMR mesh as a good initial mesh. Although those numbers may vary with each application, greater efficiencies are expected if regions of deformation are highly localized.

A couple of numerical simulations for material failure and fragmentation predictions have been carried out to verify our ALE–AMR capability in materials modeling. Our numerical results are compared with published data. A well-established third party code-ALE3D of LLNL [18] is also used to verify our present results.

The first problem to model material failure and fragmentation is a 3D calculation of an expanding ring. This problem has been previously studied by experiments [19,20] and numerical simulations [21]. An experiment used to evaluate material properties and fracture behavior of materials in the strain rate range of $10^3$—$10^4$ $s^{-1}$ is an electromagnetically driven expanding ring. A ring of material is placed concentrically over a mandrel containing a coil of wire. When current is applied to the

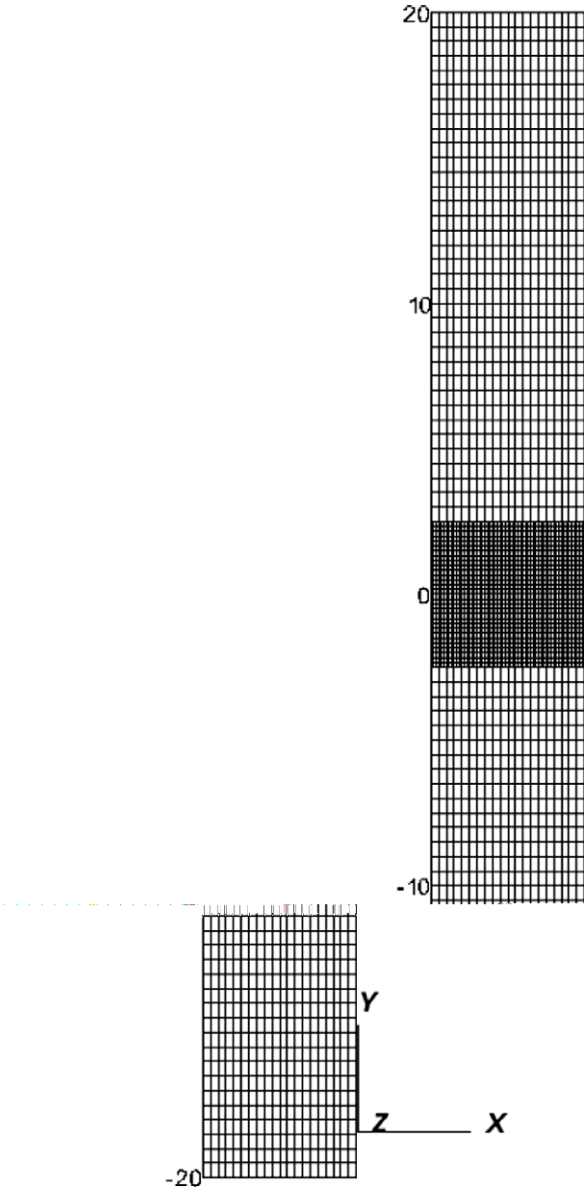


**Fig. 22.** Mesh of the tensile plate with ALE–AMR at time $t = 0$ μs.

wire, the magnetic field induced in the ring interacts with the applied magnetic field creating a uniform radial body force. If the force is large enough, the ring can deform plastically and increase its diameter. With yet higher forces, the ring expands rapidly and fractures dynamically. Numerical simulation for this problem was reported in [21] using a finite element code with explicit dynamic integration of the momentum equation, and Gurson constitutive model with several failure criteria was applied in their calculations.

In both above experiments and numerical simulations, the rings were initially 34.37 mm inside diameter, 35.89 mm outside diameter and 0.76 mm thick. We used the same description of the geometry for our computation by the ALE–AMR method for a full three-dimensional tantalum ring. Fracture is imposed by setting the stress to zero in elements as they reach the failure criteria based on the JC material model. Boundary conditions are initially imposed on the inside of the ring, consistent with the velocity of the driving ring in the experiments. The velocity is specified to match the measured velocity up to the time when the drive ring hits the stop. At that time the boundary conditions are removed and the ring continues to expand driven only by its radial inertia. Here, a velocity $0.03t/12$ cm/μs with $t \leqslant 12.0$ μs is applied on the inner boundary of the ring, and a mesh $5 \times 600 \times 5$ is used for the computation.

Fig. 16 shows the failed elements start to form at 29 μs, and Fig. 17 shows the number of fragments at 34 μs. For a numerical comparison, we use published results in [21] and use their algorithms and software (ALE3D) to compute the

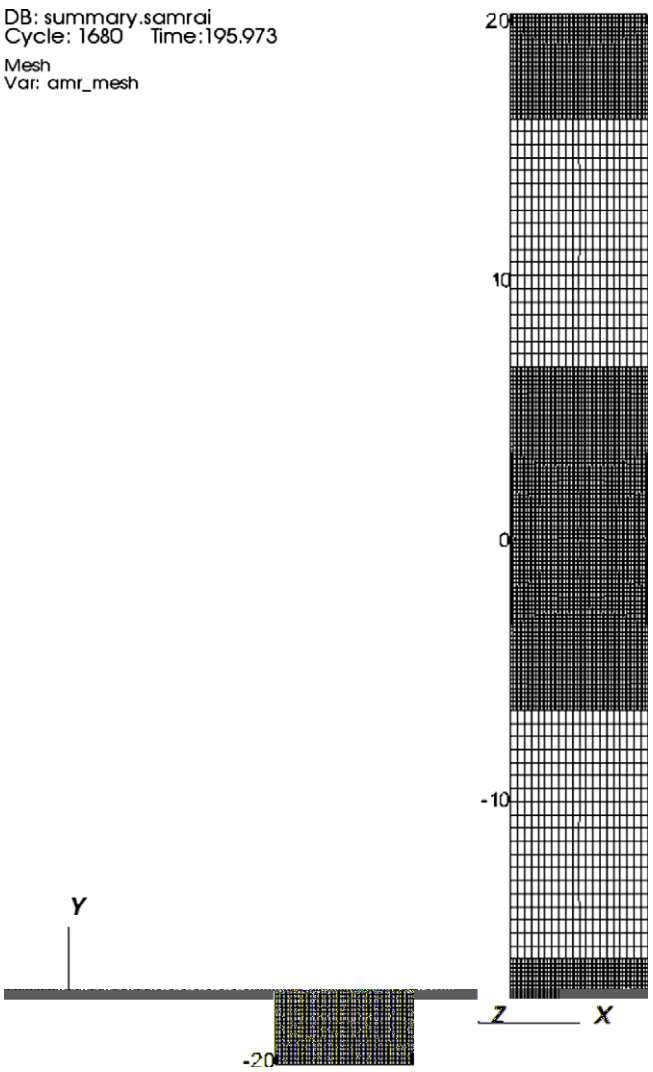


**Fig. 23.** Mesh of the tensile plate with ALE–AMR at time $t = 196$ μs.

same test problem, and the results from their method and code are showed in Figs. 18 and 19. Our numerical results in Figs. 16 and 17 by the ALE–AMR method are in excellent agreement with their numerical data in term of the predicted time to fail for the material and the number of fragments. Our results are also in good agreement with their experimental data – their 4 experiments show that the time to failure of the ring is about 30 μs, and the numbers of fragments are 21,23,26, and 28.

Our next test problem is to simulate a tensile tantalum plate subjected to a time varying uniform axial stress. The standard tensile test is one of the most important engineering procedures used to characterize the mechanical behaviors of materials. Due to the non-uniform stress and strain distribution, a material will start the deformation process. Typically at the beginning, a material behaves elastically, and after the yield strength is reached, the plastic hardening begins. Then diffuse necking will develop and finally a fracture stage will occur. Since the whole process shows complex mechanical behaviors including the failure mechanism at the necking zone, it is an ideal test problem for our current ALE–AMR approach, and the localized phenomenon should be dynamically simulated by our method.

In our numerical experiments, the dimensions of a tantalum plate is $5\,cm \times 20\,cm \times 0.2\,cm$ and a velocity $V_y = v_0 t/14.5\ cm/\mu s$ with $t \leqslant 14.5$ μs is applied at the ends of the plate at $y$ direction with $v_0 = 0.001$. In order to validate our model and numerical methods, we use the ALE3D, a multi-physics code developed by scientists at LLNL and widely used in the hydrodynamics simulation community, to model the same problem. With a uniform mesh, the results from the finite element ALE3D code and our finite volume ALE–AMR code show in Figs. 20 and 21. At $t = 750\ \mu s$, both simulations show the necking process and failure mechanism, and excellent agreement is reached.

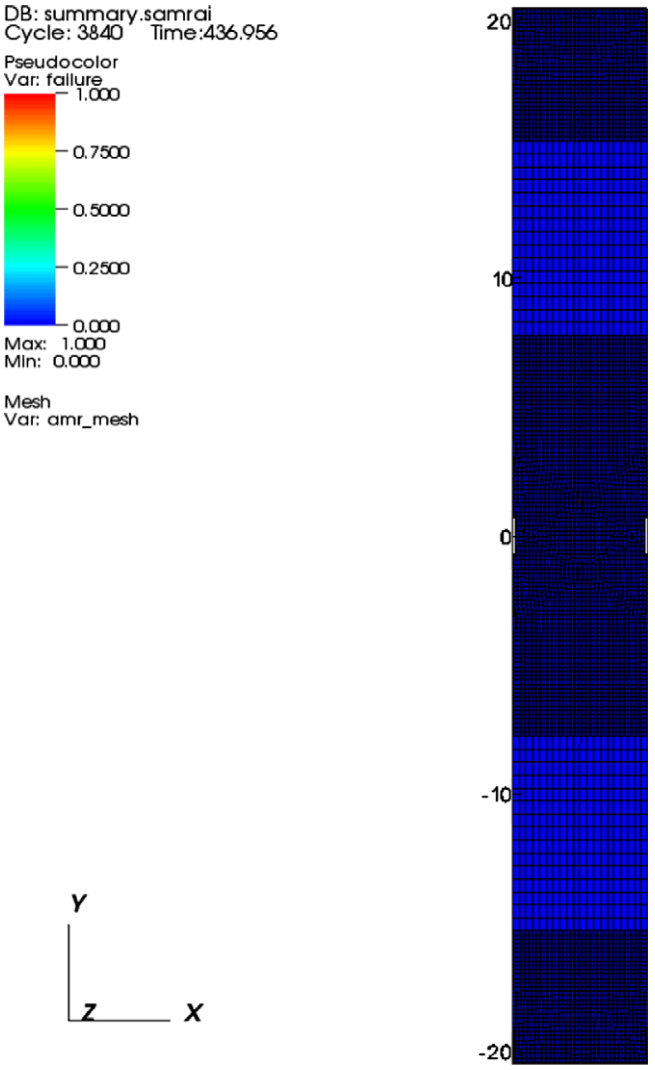


**Fig. 24.** Material failure starts at 437 μs by ALE–AMR.

After this uniform mesh test, we simulate the problem with dynamic ALE–AMR method. Since the middle of the tensile plate will break, we use a static 2-level AMR mesh with a small patch of fine mesh at the middle as our initial mesh showed in Fig. 22. Then the ALE–AMR method is applied to simulate the entire process of the material responses. The numerical results with different physical times are showed in Figs. 23–25. At $t = 196$ μs, the initial patch of the fine mesh expands in the middle and new patches of fine mesh are generated at the ends of the plate where sharp gradients of plastic strain in the middle of the plate and non-uniform stress at the ends are generated. At $t = 437$ μs, the necking zone is formed, and at $t = 557$ μs, the material goes the fracture stage. The dynamics of the section undergoing necking is demonstrated by the AMR mesh, and this highly local phenomenon – material deformation and failure is simulated successfully by our ALE–AMR method.

## 7. Conclusions and discussion

We successfully developed a new numerical method ALE–AMR for solid mechanics and materials modeling. Many important issues of this method are addressed, including Lagrangian procedure, hierarchy integration, refinement and coarsening on a moving and deforming grid, interlevel boundary conditions, and the criteria for AMR to model elastic–plastic flows and material failure. An efficient, flexible, and portable parallel ALE–AMR code is designed using C++, FORTRAN 90, and the MPI programming model. A parallel performance-efficiency study is carried out, and excellent speedup on a large number of processors is achieved.

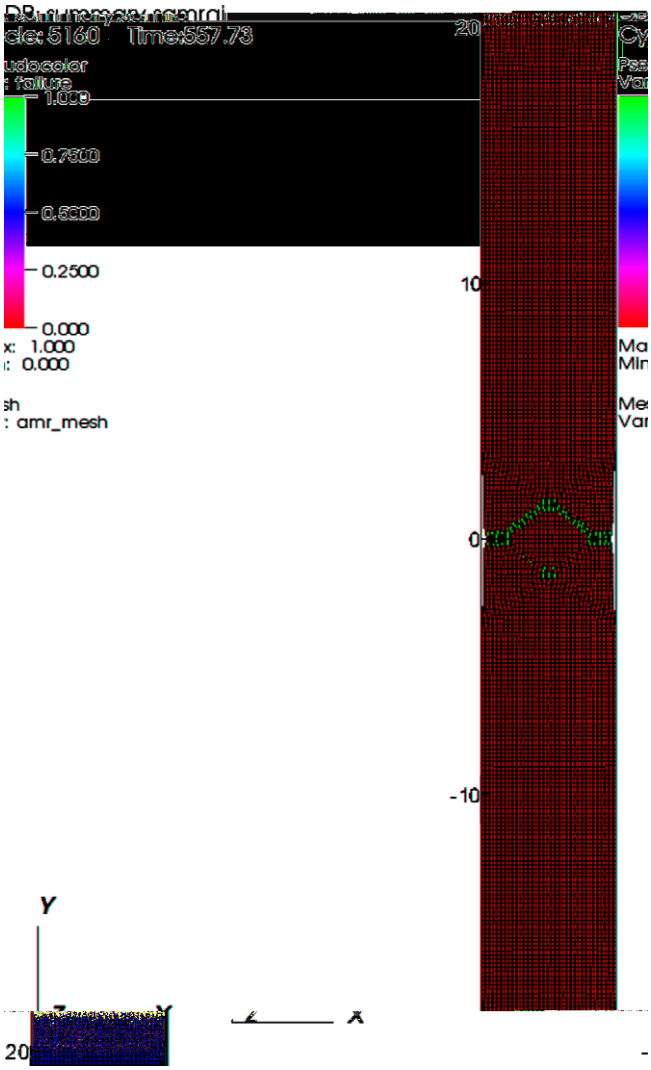


**Fig. 25.** Numerical results of the tensile plate with ALE–AMR at time $t = 557$ μs.

Based on our current numerical method, realistic simulations for elastic–plastic flows, material failure, and fragmentation predictions are carried out in several applications. A detailed numerical study of the impact problem in both 2D and 3D is performed to test both the ALE–AMR method and the plasticity aspect of the computer model. It clearly shows the strengths of this method and simulates the dynamic phenomena of the impact problem in an efficient AMR mesh both in space and time.

The numerical results of the expanding ring are obtained by our finite volume ALE–AMR method, and they are in excellent agreement with published experimental and numerical data from a finite element method. Our model provides a reasonable prediction of the time to failure and the number of fragments. A detailed numerical study of the tensile plate problem is carried out to test the ALE–AMR method with the JC material failure model. The local failure mechanism is well captured by a fine resolution in an AMR mesh, and the results are in excellent agreement with the data from a well-established code-ALE3D, which uses a different numerical approach.

Our numerical simulations show the strengths of ALE–AMR method. The results, illustrated here, clearly demonstrate the great potential for applying this approach to various realistic dynamic applications in materials modeling.

## Acknowledgments

This work is performed under the auspices of the US Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344. The author wishes to acknowledge Dr. Robert Anderson for his support for this work, and the author benefits a lot from his early research work on ALE–AMR for gas dynamics. The author likes to thank Dr. Rich

Becker for suggesting the ring problem and Dr. Alice Koniges for stimulating discussions for possible future applications. The author also likes to acknowledge the SAMRAI group of the Center for Applied Scientific Computing (CASC) , the ALE–AMR group of CASC, and ALE3D group at LLNL for their support for this work.

## References

[1] B.J. Benson, An efficient, accurate, simple ALE method for nonlinear finite element programs, Comput. Methods Appl. Mech. Eng. 72 (1989) 205–350.
[2] M. Berger, J. Oliger, Adaptive mesh refinement for hyperbolic partial differential equations, J. Comput. Phys. 53 (1984) 484–512.
[3] R.W. Anderson, N.S. Elliott, R.B. Pember, An Arbitrary Lagrangian–Eulerian method with adaptive mesh refinement for the solution of the Euler equations, J. Comput. Phys. 199 (2004) 598–617.
[4] J.M. Morrell, P.K. Sweby, A. Barlow, A cell by cell anisotropic adaptive mesh ALE scheme for the numerical solution of the Euler equations, J. Comput. Phys. 226 (2007) 1152–1180.
[5] J.M. Morrell, P.K. Sweby, A. Barlow, A cell by cell anisotropic adaptive mesh ALE method, Int. J. Numer. Methods Fluids 56 (2008) 441–1447.
[6] G.R. Johnson, W.H. Cook, Fracture characteristics of three metals subjected to various strains strain rates, temperatures and pressures, Eng. Fract. Mech. 21 (1985) 31–48.
[7] P. Colella, Multidimensional upwind methods for hyperbolic conservation laws, J. Comput. Phys. 87 (1990) 171–200.
[8] R.B. Pember, R.W. Anderson, Comparison of direct Eulerian Godunov and Lagrange plus remap, artificial viscosity schemes, in: Proc. 15th American Institute of Aeronautics and Astronautics Computational Fluid Dynamics Conference, Anaheim, CA, 2001.
[9] B.J. Benson, Computational methods in Lagrangian and Eulerian hydrocodes, Comput. Methods Appl. Mech. Eng. 99 (1992) 235–394.
[10] M.L. Wilkins, Computer simulation of dynamic phenomena, Springer, 1999.
[11] L. Margolin, J. Pyun. A method for treating hourglass patterns, in: Proc. 4th International Conference on Numerical Methods in Laminar and Turbulent Flows. Montreal, 1987.
[12] D.P. Flanagan, T. Belytschko, A uniform strain hexahedron and quadrilateral with orthogonal hourglass control, Int. J. Numer. Methods Eng. 17 (1981) 679–706.
[13] B. van Leer, Towards the ultimate conservative difference scheme V: a second-order sequel to Godunov's method, J. Comput. Phys. 32 (1979) 101–136.
[14] 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.
[15] R. Hornung, S. Kohn, Managing application complexity in the SAMRAI object-oriented framework, Concurrency: Pract. Exper. 14 (2002) 347–368.
[16] A. Wissink, D. Hysom, R. Hornung, Enhancing scalability of parallel structured AMR calculations, in: Proc. 17th ACM international conference on supercomputing (ICS03), San Francisco, 2003, pp. 336–347.
[17] T.N. Gunney, A. Wissink, D. Hysom, Parallel clustering algorithms for structured AMR, J. Parallel Distributed Comput. 66 (2006) 1419–1430.
[18] R. Sharp and et.al. ALE3D User's Manual. Lawrence Livermore National Laboratory, 2007.
[19] F.I. Niordson, A unit for testing materials at high strain rates, Exper. Mech. 5 (1965) 29–31.
[20] W.H. Gourdin, S.L. Weinland, R.M. Boling, Development of the electromagnetically launched expanding ring as a high-strain-rate test technique, Rev. Sci. Instrum. 60 (1989) 427–432.
[21] R. Becker, Ring fragmentation predictions using the Gurson model with material stability conditions as failure criteria, Int. J. Solids Struct. 39 (2002) 3555–3580.