LLNL-TR-732040

Lawrence Livermore National Laboratory

ALE3D: An Arbitrary Lagrangian-Eulerian Multi-Physics Code

Charles Noble, Andrew Anderson, Nathan Barton, Jamie Bramwell, Arlie Capps, Michael Chang, Jin Chou, David Dawson, Emily Diana, Timothy Dunn, Douglas Faux, Aaron Fisher, Patrick Greene, Ines Heinz, Yuliya Kanarska, Saad Khairallah, Benjamin Liu, Jon Margraf, Albert Nichols, Robert Nourgaliev, Michael Puso, James Reus, Peter Robinson, Alek Shestakov, Jerome Solberg, Daniel Taller, Paul Tsuji, Christopher White, Jeremy White

May 23, 2017

Disclaimer

Introduction

ALE3D is a multi-physics numerical simulation software tool utilizing arbitrary-Lagrangian-Eulerian (ALE) techniques. The code is written to address both two-dimensional (2D plane and axisymmetric) and three-dimensional (3D) physics and engineering problems using a hybrid finite element and finite volume formulation to model fluid and elastic-plastic response of materials on an unstructured grid. As shown in Figure 1, ALE3D is a single code that integrates many physical phenomena.

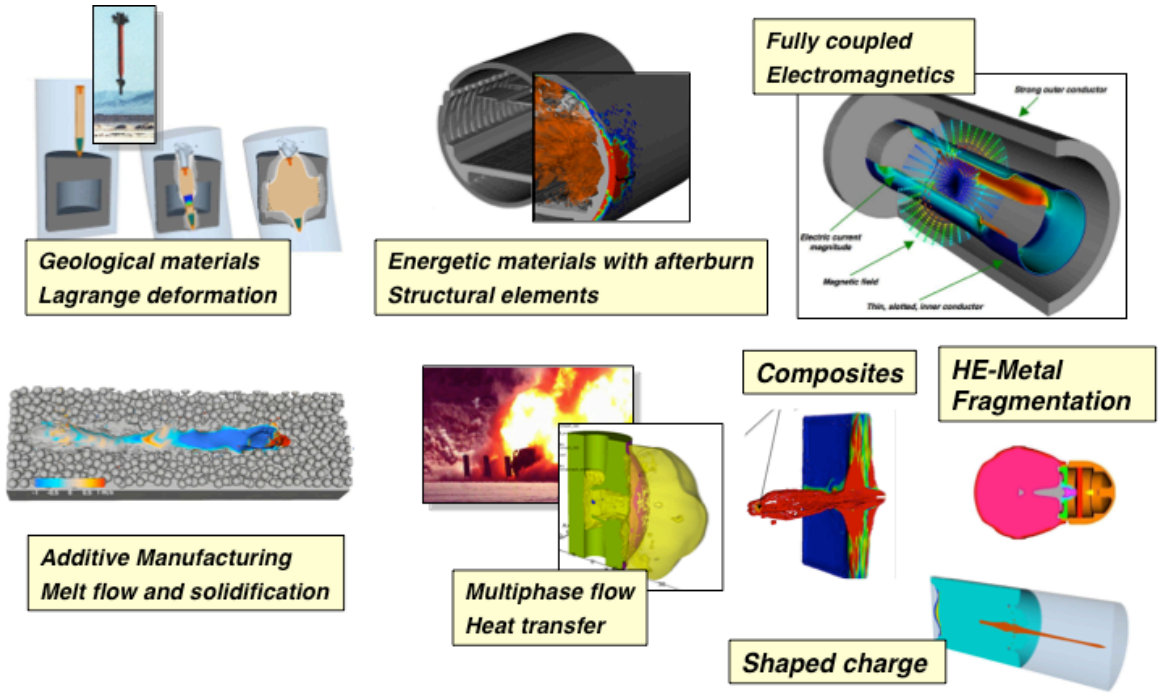

Figure 1. ALE3D supports a wide range of applications.

ALE3D supports a wide range of simulation needs. The ALE and mesh relaxation capabilities broaden the scope of application in comparison to tools restricted to Lagrangian-only or Eulerian-only approaches, while maintaining accuracy and efficiency for large, multi-physics and complex geometry simulations. For some applications ALE can deliver accuracy similar to Eulerian techniques using as few as 1/10th the number of mesh elements and a reduction in memory requirements.

Figure 2 provides a chart showing that ALE3D has an integrated flexible and extendable architecture. Beyond its foundation as a hydrodynamics and structures code, ALE3D has multi-physics capabilities that integrate various packages through an operator splitting approach. Additional ALE3D features include heat conduction, chemical kinetics, species diffusion, incompressible flow, a wide range of material models, chemistry models, multi-phase flow, and magnetohydrodynamics, which can be used in numerous combinations for long (implicit) to short (explicit) time-scale applications.

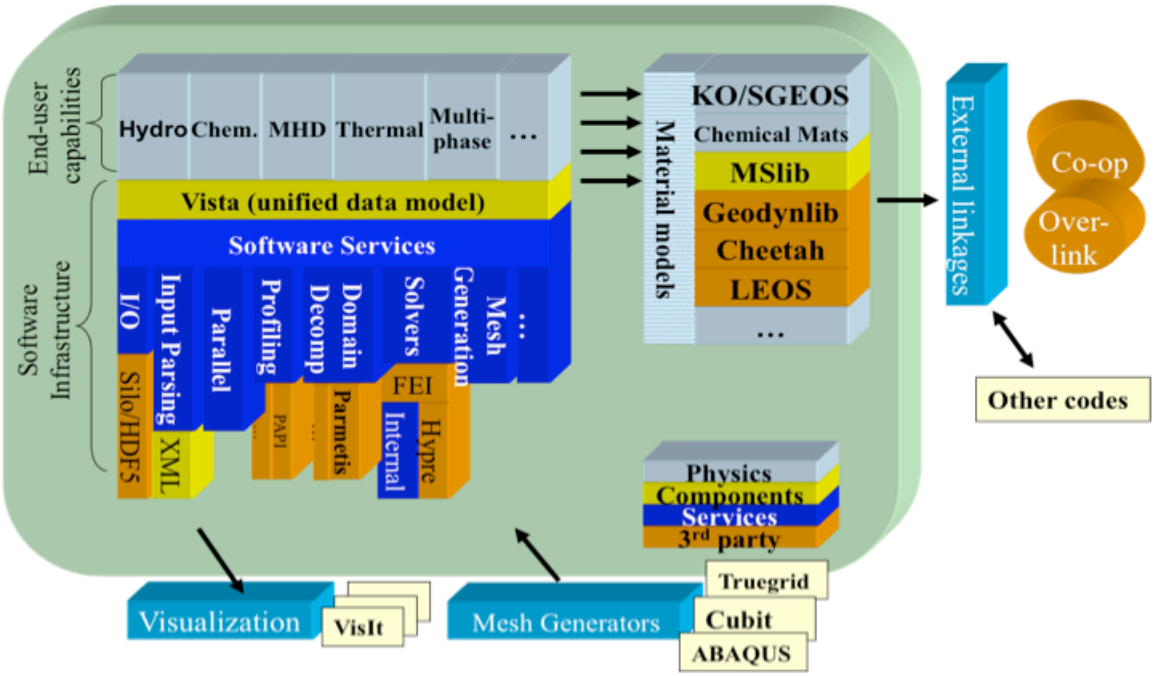

Figure 2. ALE3D has an integrated flexible and extendable architecture supporting a variety of mesh generators, and material models can be pluggable components.

The discretized domain or mesh may consist of arbitrarily connected hexahedral, shell and beam elements. In 2D simulations, the mesh can comprise of arbitrarily connected quadrilaterals. The mesh can be constructed from disjointed blocks of elements that interact at the boundaries via slide surfaces or other types of boundary conditions.

Large mesh distortions can be addressed via mesh relaxation and/or the use of ALE techniques where material is advected through the mesh. All components of the code participate in advection and all the mature capabilities operate with slide surfaces. Advection is the process whereby the mesh is modified to alleviate tangling or to preserve an Eulerian grid. Advection is implemented in a Lagrange plus remap fashion. For each cycle, after a Lagrangian motion, the state variables flow from the original mesh to the modified mesh. The mesh is allowed to cross material boundaries and create multi-material elements. The ALE and mesh relaxation capability broadens the scope of applications in comparison to tools restricted to Lagrangian or Eulerian (advection) only approaches, while maintaining accuracy and efficiency for large, multi-physics and complex geometry simulations.

Slide surfaces are boundaries between disjoint sections of the grid that may or may not be in contact. They represent either physical contact discontinuities or a discontinuity in the zoning. The unstructured grid is composed of 3D hexagonal elements that can be arbitrarily connected. Triangular or prismatic elements are not allowed, but shell and beam structural elements can couple to 3D elements.

The 2D capability includes explicit and implicit hydrodynamics, thermal diffusion, chemistry, deflagration, shape generation, advection, and most explicit slide surface features.

Thermal and chemistry packages are tightly coupled and have been successfully used for long (implicit) to short (explicit) time-scale applications. The incompressible flow module has been used in simulations of turbulent thermal convection. A multi-phase flow package (a continuum representation of solid particulate transport in solids, liquids and gases and impulse load at impact) is available for simulation of multi-phase fluid, structure interaction. Another package that solves the magnetohydrodynamics equations (MHD in 3D only) can simulate a variety of high magnetic field, high velocity, thermal applications.

ALE3D operates on a wide variety of platforms, ranging from laptops to the world's largest supercomputers. ALE3D has native implementations for Windows™ and Mac workstations for smaller scale problem sets, and it is portable to virtually any Unix-based machine with C++/C and Fortran compilers available. The code will also run in parallel on multi-processor Windows™ and Mac machines. While most users will be interested in Linux-based versions of the code, it has also been ported to several other lightweight kernel operating systems.

Operation on massively parallel machines has always been a core requirement for the code. Parallelization is implemented by decomposing the mesh into computational subdomains. Message passing is the primary mode for communication between domains. The emphasis on message passing means that it is possible to run parallel computations on a network of workstations, as well as utilizing multi-core processors available on most workstations.

Explicit Hydrodynamics

ALE3D's hydrodynamics capability captures the behavior of solids and fluids and has been successfully used for long (implicit) to short (explicit) time-scale applications. The explicit hydrodynamics module was developed to model the behavior of objects undergoing deformation due to the application of shocks in the kilobar regime. For such problems, the natural time step is consistent with the Courant time scale which governs the stability of the equations.

The code explicitly conserves mass and momentum. Following DYNA3D (Hallquist, 1982), the stress gradients and strain rates for the Lagrange step are evaluated by a lowest-order finite-element method. A diagonal mass matrix is used. For second-order accuracy a staggered space and time grid is also used. The stress gradient calculation has been modified so that it is represented by an integral of the shape function over the surface of an element rather than an integral of the gradient of the shape function over the volume. (The stress in general is discontinuous at the surface so that the stress gradient is a delta function.) This modification corrects a problem in which distorted elements generated forces even though the stress field was constant. Likewise, a more accurate volume calculation is done following J. Dukowicz (JCP) (1984). Hour-glass modes are damped using the method of Flanagan and Belytschko (1981).

The hydrodynamics is energy based rather than temperature based. Energy conservation is not explicitly enforced but depends on the accuracy of the time integration. This averts the problem of converting lost kinetic energy into internal energy and overheating materials. The pressure, viscosity and strain work evaluations are fully time centered. To integrate –PdV for nonlinear equations-of-state, a third-order Runge-Kutta method is used. Although the explicit hydrodynamics module is isentropic except at shocks, a temperature variable exists for each region, and the temperature Equation of State (EOS) can be evaluated either as part of a constitutive model or from EOS tables.

Shocks are treated with a scalar artificial viscosity. The default version simply uses $\delta v/v$ for compressing elements. Both linear and quadratic terms are available. R. Christensen's monotonic artificial viscosity (Christensen, 1990) has also been implemented. This scalar viscosity is constructed using velocity gradients from which the linear component of the gradient has been subtracted. The form can be derived from the linearized Riemann problem.

The most commonly used equations-of-state and constitutive models are analytic models using the Mie-Grüneisen and Steinberg-Guinan forms, respectively (Steinberg, 1980 and 1996). Several engineering constitutive models are available to facilitate modeling of structures. Tabular equation-of-state models can be accessed through the Livermore equation-of-state (LEOS) tables. The majority of the available material models are for isotropic materials and a von Mises yield condition is applied. The deviatoric part of the stress tensor is integrated through time, and the Jaumann stress rate is used to satisfy objectivity. A number of micro-structurally based models have also been added to provide a more accurate representation of damage evolution mechanisms leading to fracture and to capture anisotropy at both the single crystal and polycrystal size scales. A few of these latter models do not have time-centered energy evaluations or use high-order integration for the EOS. For all material models, a Navier-Stokes viscosity is available to augment the stress tensor.

For most problems, high explosive (HE) detonation is simulated by using a programmed burn model with a beta burn override. Several options are available for computing HE lighting times. Reactive flow models for HE detonation are also available. These models use pressure and volume dependent rate laws to describe the detonation process. The reactive flow models are zoning dependent, need high resolution, and may have limited usefulness in 3D.

Implicit Mechanics

The implicit mechanics module was developed to model problems that evolve at time scales that are orders of magnitude greater than the Courant time scale that determines the stability of the explicit equations. The implicit hydrodynamics is a finite element displacement formulation with single- or eight-point integration. Single-point integration requires the addition of hourglass stabilization forces. The formulation solves the non-linear equilibrium equations using a Newton-Raphson iteration surrounding a linearization of the equations. This linearization is based on estimates for material properties, etc., at the end of the time step. Convergence of the non-linear iterations is achieved when corrections to the displacements and the nodal forces are sufficiently small. Slide surface constraints are supported. The implicit time integration can be run using a quasi-static approximation, or the inertial terms can be included via a Hilber-Hughes-Taylor stabilization of the standard Newmark time integration. The code can convert automatically from implicit to explicit hydrodynamics when the time step is sufficiently small that the explicit integration is more computationally efficient. The reverse transfer from explicit to implicit hydrodynamics is also possible.

Slide Surfaces

The interaction between multiple material surfaces is captured with a numerical technique called slide surfaces. As mentioned above, slide surfaces model contact discontinuities or mesh discontinuities. Slide surfaces may be either two-sided (master / slave) or single-sided. In two-

sided surfaces, the nodes of each side interact with the faces of the other side to conserve momentum and enforce impenetrability. With single-sided surfaces, any node may interact with any face, barring nearest neighbors. Single-sided slides are useful for surfaces that fold upon themselves and surfaces consisting of many discrete segments.

There are several options for contact enforcement: the point-on-plane method for explicit or implicit hydrodynamics, the common-plane method (available only for single-sided surfaces) for explicit hydrodynamics, and a mortar method for implicit hydrodynamics. In addition, ALE3D provides an automatic slide surface capability with either the point-on-plane or common-plane approach.

For the point-on-plane method, slide surface velocities are computed with a momentum conserving algorithm. Each side of a slide surface is independently accelerated using interpolated physics quantities from the opposing side. First a node is accelerated as if it were on a free surface or a surface with a pressure boundary condition. The boundary pressure is the average of the normal stresses on each side of the surface. The acceleration of the node is decomposed into normal and tangential components. The normal component for each side is combined with the interpolated normal component from the opposite side to form a center-of-mass acceleration. This acceleration is used to integrate the velocity. Unless friction is called for, the tangential acceleration remains that of the free surface. A distinction is then made between master and slave surfaces for the final application of continuity boundary conditions. This last step corrects for any lack of perfect continuity in the normal direction due to truncation errors.

Techniques for ordering nodes on one side of the slide surface with respect to nodes on the other side and for making projections normal to the slide surface are borrowed from DYNA3D. These techniques have been modified, however, to make them more robust. The ordering algorithm has been improved so that nodes are not prone to penetrating the slide surface and getting lost. The normal projection algorithm uses normals that vary across a slide surface element and capture the effects of curvature of the surface.

The opposing sides of a slide surface may be separated and come into contact during the course of a calculation. This is called a void. When voids close, the momenta from the opposing sides are combined to form the center-of-mass momentum. The collision is inelastic for the first layer of nodes on each side of the slide surface. Void closing is always calculated from the perspective of the master side. This is done to avoid miscounting momentum transferred from one side to the other if nodes on each side of the surface close in different cycles.

By default, the tangential velocities on each side of a slide surface are decoupled. However, coulomb friction can be applied. In this case, a tangential force is applied that opposes any relative velocity. This force is proportional to the normal force that is inferred from the acceleration required to change the normal velocity from its free surface value to its center-of-mass value. For so-called tied sliding both the normal and tangential accelerations are combined into center-of-mass accelerations. This enables one to use slide surfaces to affect a zoning change in the middle of a region.

Special provision is made for intersecting or overlapping slide surfaces. Intersections are assumed to be orthogonal and errors grow to the extent that this condition is violated.

The common-plane method of contact enforcement checks for contact between each pair of faces, defined as penetration of a plane constructed midway between the two faces. If both faces

penetrate the plane, then the intersection of their penetrations projected onto the common plane is the area of contact, and an opposing force (penalty method) is applied to resist and reverse penetration. This force is distributed to the nodes of the faces in contact.

The mortar method of contact is a face-on-face type of method and is implemented for use with implicit hydrodynamics. This method considers the faces on one side of a surface that are connected to a node, and the faces on the other side that overlap these faces, and enforces that a weighted sum of the gap volumes of the overlap be zero.

At most one slide surface may be designated as an autocontact surface. The faces of the autocontact surface are not explicitly specified, but typically consist of all external faces (faces with a zone on only one side) which are not included in any other slide surface.

FEusion Embedded Mesh Coupling

The FEusion library provides an embedded mesh method that allows the coupling of two separate, spatially overlapping meshes. The embedded mesh approach can greatly simplify the pre-processing requirements while potentially avoiding many run-time issues related to tangling of an ALE mesh conforming around Lagrange objects.

The approach used here, which was developed by Puso et al., uses Lagrange multipliers from a piecewise constant space on the cut background elements to constrain the jump between background and foreground velocities. A stabilization scheme penalizes the difference in face-adjacent Lagrange multipliers tractions. These multipliers are solved for implicitly using a conjugate gradient (CG) iterative method on the subset of the mesh where they are active, i.e., only on the cut background cells.

Advection

ALE3D uses an arbitrary Lagrangian-Eulerian (ALE) algorithm. The algorithm consists of two distinct steps: (i) a Lagrangian step that updates nodal positions, nodal velocities, and zonal quantities, and (ii) a remap/advection step that remaps the results of the Lagrangian step onto a mesh determined by the relaxation method specified by the user.

In the ALE3D implementation of the ALE algorithm, the Lagrangian step is executed, a new representation of the mesh is created, and then material variables such as mass, momentum, and energy are advected from the old mesh to the new mesh. The advection of the material variables is done by computing fluxes of the state variables between the Lagrangian mesh and the new mesh. The pressure is recalculated after the remap with a call to the EOS routines. If the materials have strength, the stress deviators, plastic strain, and other constitutive model variables are also advected.

The advection step comprises a relaxation phase and an advection phase. An “ideal” grid is first created using an equipotential or condition number grid relaxation algorithm. (A pure Eulerian option is also available.) The state variables are then remapped onto the new grid by constructing fluxes between the old grid and the new grid. The fluxes for extensive variables (mass, internal energy, and momentum) are conservative. For pure zones, a second-order, monotonic algorithm is used to calculate the fluxes. This technique was pioneered by van Leer (1977). Velocities are updated by applying the results of momentum conservation. This leads to a loss of kinetic energy

(which is quadratic rather than linear in the velocity), and high-order advection is required to limit the loss. As a rule, lost kinetic energy is not added to internal energy to explicitly force total energy conservation. However, kinetic energy lost in the vicinity of a shock can be put back into the internal energy field. This is useful for the purpose of propagating shocks over long distances without degradation.

For problems with several material regions, the process of advection can create mixed elements in which more than one material region resides in an element. While material region interfaces are currently not tracked explicitly during the advection step, they are inferred from the volume fractions of each region in neighboring zones. This is done to determine the order in which regions are moved from one element to another. The ordering process preserves the integrity of region objects as they move through the mesh. When volume fractions provide no guidance, regions are moved from one element to another in the order in which they are numbered, so it can be very important to order regions in such a way that "more important" (i.e., non-background) materials have lower region numbers. Separate values of the thermodynamic state variables are maintained for each region in a mixed element, and advection is done using a first-order upwind method. Regions in mixed elements are allowed to relax towards pressure equilibrium. The relaxation algorithm is based on a linearized solution to the Riemann problem.

Free surfaces are not relaxed unless tangential relaxation is explicitly set up for a free surface nodeset. Advection can take place up to, but not across, a slide surface. Ordinarily only the slave nodes of a two-sided slide surface are relaxed, but master-side relaxation is available (similar to tangential relaxation). If necessary, and if slide surface nodes are forced to line up across the slide surface, the slide surface can be deleted during the course of the calculation and thereby allow for advection after that time.

Thermal Diffusion

Thermal diffusion is the conduction of heat from a hot to a cold temperature location in a solid or fluid. The thermal diffusion module was originally incorporated to model manufacturing processes such as casting, forging, rolling, and extrusion. It is implemented in a manner that allows for application to most engineering heat transfer problems. The thermal diffusion module has also been coupled to a chemical kinetics module for the purpose of modeling thermal ignition of high explosives. The existing capabilities include conduction with orthotropic and temperature dependent properties, phase changes, enclosure thermal radiation, thermal contact resistance across interfaces, and temperature, flux, thermal radiation, and convection boundary conditions.

Two modes of coupling between the hydrodynamics and the heat transfer are supported. The first solves the hydrodynamics and heat transfer consecutively at each step by operator splitting. The heat transfer step changes energy without changing volume and the hydrodynamics step changes volume without transferring heat. Cell-centered energy is made consistent with nodal temperatures by including an adiabatic expansion source term, $\gamma\ (\delta V/V)$, and strain heating. Chemical reactions also contribute a source term. The second mode solves the hydrodynamics and heat transfer iteratively using the intermediate results of the package until they both converge.

The heat conduction equations are solved using a finite element approach. The solution includes implicit time integration, direct or iterative matrix solution, hexahedral elements with second-order Gaussian quadrature integration, nonlinear convergence by successive substitution or Newton-Raphson methods, and variable time step control with sub-cycling of the thermal diffusion and hydrodynamics.

Heat transfer can occur across a slide surface by perfect thermal contact or by heat conduction using 'virtual elements'. Virtual elements are conduction and thermal radiation resistive elements that are inserted between the slide surfaces. The virtual elements can have zero thickness or their resistance can be scaled with gap distance. The perfect thermal contact algorithm is formulated as a penalty method. It maintains symmetry at boundary edges through geometrical and mass weighting, and scales the penalty function with the magnitude of the diagonal term of the coefficient matrix.

Chemistry and Chemical Diffusion

ALE3D's chemistry model is capable of capturing the composition, structure and properties of chemically reacting materials. The chemical kinetics module was developed to model decomposition of high explosive materials in thermal environments. A chemical material is a set of materials that can transform among each other. These materials (or species) define all the properties of the chemical material. Several models for combining the properties of the individual species are provided. In principle, any material model that supplies a temperature and pressure can be used as a species in a chemical material. It is even possible to combine several models by including a chemical material as a species in another chemical material.

The composition of a chemical material can be modified through a set of user defined chemical reactions. ALE3D organizes the reactions into groups that act on the same set of species. Only reactions that involve all of the species present in a chemical material are included in that material's full reaction scheme. The change in composition is calculated implicitly with a self-correcting Newton-Raphson technique. The temperature derivative of the change is also calculated for use by the thermal module.

The chemical kinetics package is normally run tightly coupled to the thermal module. However, it can also be run during the hydrodynamics phase. When run during the hydrodynamics phase, the chemical kinetics package can be used to emulate a variety of reactive flow detonation propagation models. This emulation is accomplished by including the appropriate set of reaction mechanisms with the appropriate material models for the reacting species. Both the standard Lee-Tarver (Lee and Tarver, 1980) and PERMS (Propellant Energetic Response to Mechanical Stimuli) (Maienschein et al., 1997) models can be implemented this way.

A chemical diffusion model is also available in ALE3D. It allows for diffusion within a chemical material (not between chemical materials). The species diffusion equations are solved using a finite-volume approach. The operator is explicit in time and computes fluxes of chemical species across faces between elements based on the locally determined chemical potential gradient. Both tracer, where the diffusing species is assumed to be only a small fraction of the total mass of the system, and non-tracer models are available.

Incompressible Flow

The incompressible flow package solves the incompressible Navier-Stokes equations and is useful for simulations involving a fluid that can be approximated with a constant density, such as low-speed aerodynamics or internal fluid mechanics. As a rule of thumb, flows with a Mach number less than 0.3 are candidates to be considered incompressible. An obvious exception to this rule is any flow where the density variations drive the flow, no matter how fast it is going. The package can be coupled to the thermal package for simulations of natural convection of a fluid with temperature gradients. The incompressible flow model may also be run simultaneously with the standard ALE hydro package for coupled fluid-structure problems. The fluid and solid components are coupled through their boundary conditions. The incompressible flow module includes explicit, semi-implicit (either the advection or viscous terms implicit) and fully implicit time integration options. An advection-diffusion solver is also available to model species concentrations within an incompressible material.

Multiphase Flow

The ALE3D multiphase flow model was developed for simulating mixed materials with separate velocity fields. This model reuses the species framework developed for the chemical materials. In principle, any ALE3D material model that supplies a temperature and pressure can be used as a species in the chemistry and multiphase packages. Often, it is desirable to create a hierarchical chemical material for use in the multiphase model. An example of this is a multiphase flow of particles in a fluid flow, each composed of multiple constituents that can react amongst themselves. This hierarchy of material models enables chemical reactions to be modeled in a unified manner with multiphase flow.

There are several models for drag and compaction in the multiphase flow package. These include semi-analytic models where the drag terms can be integrated analytically to those which are evaluated using backward Euler. For cases where the compaction viscosity is unknown, the pressure relaxation module defaults to an infinite relaxation rate or pressure equilibration.

Magnetohydrodynamics

ALE3D's magnetohydrodynamics model is capable of capturing the dynamics of electrically conducting solids and fluids. The magnetohydrodynamics (MHD) module was developed primarily for the modeling of coupled electro-thermal-mechanical (ETM) systems that are inherently 3D in nature. Example applications for this capability include explosively driven magnetic flux compression generators, induction heating / metal forming and electromagnetic rail gun systems. The ALE3D MHD module solves the resistive magnetic induction equation given a collection of specified current and voltage sources. The equation is solved in the Lagrangian frame using a mixed finite element method employing *H(Curl)* and *H(Div)* finite element basis functions which preserves the solenoidal nature of the magnetic field to machine precision. Electromagnetic force and resistive Joule heating terms are coupled to the equations of motion and thermal diffusion in an operator split manner. For problems that require mesh relaxation, magnetic advection is performed using the method of algebraic constrained transport that is valid for unstructured hexahedral grids with arbitrary mesh velocities. The advection method maintains the divergence free nature of the magnetic field and is second-order accurate

in regions where the solution is sufficiently smooth. For regions in which the magnetic field is discontinuous (e.g., MHD shocks), the advection step is limited using the method of algebraic flux correction, which is local extremum diminishing and divergence preserving.

Parallelization and Scaling

ALE3D is parallelized across the problem space using domain decomposition to associate separate pieces of physical space to individual processors. The implementation uses MPI to communicate across processor boundaries. Several of the key characteristics of the ALE3D software infrastructure contribute to the code's scalability. Field data are contained in simple C arrays, giving the compiler canonical loop iteration patterns to optimize. Ghost boundary data is always stored contiguously, removing the need for gather / scatter operations during MPI communication. Nearly all MPI communication is point-to-point, with only two global reductions per time step: the duration of the next time step, and a global error status check.

The amount of point-to-point communication required is dependent on the type of calculation being run. Problems running explicit hydrodynamics require a single communication to collect the sum of the forces at nodes along domain boundaries, although some optional algorithms (e.g. monotonic Q) require more communication. Problems running advection require a 20x – 30x greater amount of communication, both in terms of the number of communication points, and the amount of data typically sent. A rough breakdown of the steps performed in the advection are: nodal relaxation; calculation of volume fluxes; identification of mixed elements and interface reconstruction; advection of element centered variables; and momentum (node centered) advection. If there are slide surfaces, additional communications are required. The approach taken in ALE3D is to use a separate decomposition for slide surfaces. Nodes on one side of a surface (the "master" side) are assigned statically to the various processors in a load-balanced manner, and then nodes on the other ("slave") side of the surface that are currently "close" to the masters on a processor are assigned to the same processor.

The daunting task of fully characterizing the performance of ALE3D's many packages has never been done in a systematic way, but we present an overview of the scaling of the Lagrangian hydrodynamics package that forms the core of ALE3D. Figure 3 shows the weak scaling behavior of a 5695 Element / process Sedov problem from 256 to 96,000 processes.

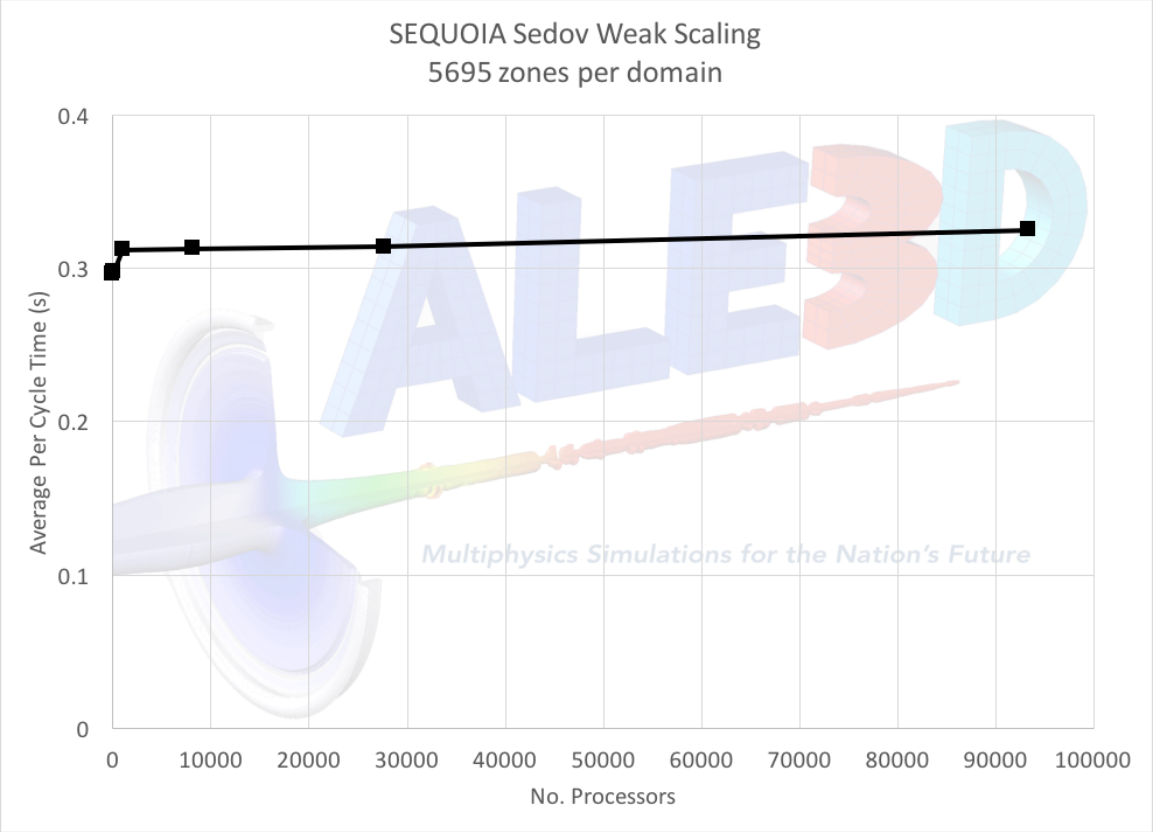

Figure 3. Weak scaling results for ALE3D's Lagrangian mechanics package.

Conclusion

The ALE3D code continues be under active development at LLNL. In addition to new physics and methodology improvements, the underlying computer science framework is being modified to ensure excellent performance on the latest generation of High Performance Computing machines. This Export Controlled and Official Use Only code is available to analysts in the Department of Defense and associated contractors for work related to national defense. Contact the authors for additional information.

References

Christensen, Randy B., "Godunov Methods on a Staggered Mesh – An Improved Artificial Viscosity," Lawrence Livermore National Laboratory, Livermore, CA, UCRL-JC-105269 (1990).

Dukowicz, J. K., "Conservative Rezoning (Remapping) for General Quadrilateral Meshes," *J. Comp. Phys.* **54**, 411-424 (1984).

Flanagan, D.P. and Belytschko, T., "A uniform strain hexahedron and quadrilateral with orthogonal hourglass control," *Int. J. Numer. Methods Eng.* **17**, 679-706 (1981).

Hallquist, J.O., “Theoretical Manual for DYNA3D,” Lawrence Livermore National Laboratory, Livermore, CA, UCID-19401 (June 1982).

Steinberg, D.J., Cochran, S.G., and Guinan, M.W., *J. Appl. Phys.* **51**, 1498 (1980).

Steinberg, D.J., “Equation of State and Strength Properties of Selected Materials,” Lawrence Livermore National Laboratory, Report UCRL-MA-106439 (1996).

Lee, EL and Tarver, C.M., "Phenomenological Model of Shock Initiation in Heterogenious Explosives," Physics of Fluids, Vol 23, Issue 12, pp. 2362-2372.

Maienschein, J.L., Lee, E.L., Reaugh, J.E., Merrill, C.I., and Lambert, R.R., “Modeling the Impact Response of Booster Propellants,” Proc. JANNAF Propulsion Systems Hazards Subcommittee Meeting, 27-30 October 1997, CPIA Publication 657 Volume 2, pp 163-177 (1997).

Puso, M., Sanders, J., Settgast, R. and Liu, B., “An embedded mesh method in a multiple material ALE,” Computer Methods in Applied Mechanics and Engineering, 245, 273-289 (2012).

Van Leer, B, “Monotonicity and conservation combined in a second order scheme,” *J. Comp. Phys.* **14**, 361-70 (1977).

Wilkins, M., “Calculation of Elastic-Plastic Flow,” *Methods in Computational Physics, Vol. 3, Fundamental Methods in Hydrodynamics*, Academic Press, pp. 211-263 (1964).