An overview of mesoscale material modeling with Eulerian hydrocodes

K Olney, D J Benson and V Nesterenko
University of California, San Diego, 9500 Gilman Drive, La Jolla, CA, 92093-0085
E-mail: dbenson@ucsd.edu

Abstract. Eulerian hydrocodes were originally developed for simulating strong shocks in solids and fluids, but their ability to handle arbitrarily large deformations and the formation of new free surfaces makes them attractive for simulating the deformation and failure of materials at the mesoscopic scale. A summary of several numerical techniques that have been developed to address issues that commonly arise for this class of problems is presented.

1. Introduction
Multi-material Eulerian codes are attractive for modeling materials at the mesoscale for several reasons. First, they permit arbitrarily large strains which avoids the problem of small time steps and inverted zones that frequently occur with Lagrangian codes. Second, the interface reconstruction permits the natural evolution of new free surfaces associated with material failure. And finally, chemical reactions are modeled in a straightforward way by evolving the volume fractions of the different materials. Despite these advantages, modeling the evolution of the microstructure presents computational challenges. Some of the methods that have been found useful are summarized here.

Generating a realistic model of the material microstructure is the first challenge. For example, numerically packing a powder to the correct initial density while satisfying the statistical size and shape distributions is a difficult problem. However, since Eulerian codes permit multiple materials within each element, or cell, the material interfaces are not required to follow the mesh lines. The use of digital image processing to map the pixels of micrographs to the Eulerian mesh has proven to be a popular and useful means of creating accurate models of complex microstructures from powders to biological materials. Micro CT scans have been used to extend this approach to three dimensions for several classes of materials.

Modeling the interaction between individual constituents of the microstructure is another major challenge. Continuing with the powder example, the individual particles may melt and form jets during shock compression, and the voids between them collapse. Dynamic material ordering has become a necessity for the accurate reconstruction of the material interfaces, and many codes now have a suite of options for handling multi-material mechanics. True contact algorithms are now replacing multi-material approximations in some cases.

Even for very small representative volume elements, it is not possible to model every detail of the material. Subscale phenomena may be crudely approximated by randomly varying the spatial distribution of the material properties. Using a large number of material species to represent the variations is usually unattractive. Directly specifying the properties point-wise as history...
variables has not proven successful because the limiters in the transport algorithms quickly smooth out the variations, but circumventing the limiter problem is shown to be relatively simple.

2. Generating realistic microstructures
A representative volume element (RVE), roughly speaking, is the smallest volume of a material that behaves in a manner representative of the overall response of the bulk material. For a periodic material, a single cell is sufficient. Back when even the largest computers had only a single processor, many materials were modeled as periodic simply because of the limited computer power of the time. For example, porous metals were modeled as a uniform cube of metal with a single spherical void at its center. In many cases, including porous metals, the results were surprisingly good and provided substantial insights into the dominant deformation mechanisms in the material.

Voronoi tessellation is a popular approach to generating RVEs of polycrystalline metals. Like many methods for generating synthetic microstructures, it is often difficult to guarantee that the synthetic microstructure matches the statistical properties (e.g., grain size) of the real microstructure.

Ideally, the RVE should be obtained directly from the actual material. Digitized micrographs may be directly imported into an Eulerian mesh in a straightforward manner [1]. Each rectangular pixel in an image is regarded as consisting of a single constituent. The particular constituent is determined from the color or gray scale value of the pixel. The area of each pixel is added to the total area of the appropriate constituent in an element it overlaps, and the final totals are divided by the element area to determine the volume fractions. It is not necessary for the finite difference zones or finite elements to be an exact multiple of a pixel's: the algorithms for efficiently calculating the overlapping area of two arbitrary polygons are well known in computer graphics [2]. This approach works well for materials composed of distinctive constituents. In practice, the image must often be cleaned up, and choosing an image that is

Figure 1. The sequence of operations for making an image suitable for generating a microstructure. The initial micrograph is on the left, the middle image is an intermediate step, and the final model is shown on the right.
“representative” may be difficult (especially since researchers are most drawn to microstructures that are interesting).

Figure 1 illustrates the process. The first image on the left is the original micrograph of a HMX powder (note that it includes an arrow). The middle image shows the micrograph after thresholding has isolated the HMX particles and the arrow has been deleted. The final image on the right shows the final image for the model. Note that the particles have been assigned different gray scale values, a different gray level for each material. Particles that are adjacent to each other must have different gray scales so that the interface reconstruction algorithm can resolve the material interface between them. If the particles are subjected to strong shocks, the number of materials must be increases such that the jetting of the particles does not cause two initially remote particles from bleeding into each other.

A common problem with transport algorithms is the production of spurious entropy profiles within the material due to shock reflection from boundaries and collision between shocks. These spurious profiles are often associated with significant increases in local heating. Because Eulerian techniques are particularly appealing for the strong shock loading of heterogeneous energetic materials that can result in complex interactions between shocks at the meso-scale, it is reasonable to expect that these interactions may lead to erroneous hot-spot predictions which are important for predicting shock induced ignition and initiation. At this moment, there is not a perfect solution for this problem. In our own work, we have used two different approaches, and it is not clear which is superior. The first is to use transmitting boundary conditions, which eliminates the problem of spurious reflections, however it underestimates hot spot formation because it throws away the shocks, and their energy, as they exit the RVE. The second uses periodic boundary conditions, which may be theoretically better because it doesn’t discard the shocks, but it comes with the expense of violating our assumptions that the RVEs are uncoupled and perfectly random. At the moment, the second approach is favored. However, we usually run both cases and regard them as rough approximations to the lower and upper bounds on the hot spot formation.

3. Interactions between material interfaces
Contact algorithms that permit slip and separation are well developed for Lagrangian formulations [3]. For Eulerian calculations, the situation has been different, with mixture theories [4, 5] acting as the de facto contact algorithms in elements containing multiple materials. The underlying limitation of mixture theories is the single continuous velocity field that spans all the materials in the calculation, which, for example, makes it impossible to represent the discontinuous velocity field associated with pure slip.

The limited ability of a mixture theory to model slip is often not a problem in mesoscale calculations. For example, a mesoscale simulation involving shear bands usually resolves the bands. Similarly, ductile failure results in fully damaged materials that can’t sustain a shear stress, permitting relative slip. There are situations, however, such as granular flow, where an accurate model of contact is important. Perhaps the most useful application is the modeling of local debonding between material interfaces [6] where the failure criteria is based on the traction at the interface.

Recent work [7–9] locally introduces extra degrees of freedom for elements containing multiple materials such that each material has its own independent velocity field. Conceptually, each material can be thought of as having its own mesh. See figure 2 for a comparison between the results for Lagrangian contact, Eulerian mixture theory, and the new Eulerian approach. Memory requirements are reduced by eliminating the empty elements and the nodes surrounded by empty elements for each mesh.

After each material has been given its own, independent mesh, the problem then becomes preventing material boundaries defined by the interface reconstruction algorithm from
Figure 2. Planar Taylor anvil solutions demonstrating contact methods: a) LS-DYNA Lagrangian solution using the penalty method, b) Eulerian solution using a mixture theory, and c) the Eulerian solution using contact.

interpenetrating just as in the contact algorithms of Lagrangian codes, with the same methods being applicable. At the start of a Lagrangian time step, the meshes are coincident. The motion of a mesh containing a particular material moves according to the stress in the material, the contact forces between materials, and the traction boundary conditions. At the end of the Lagrangian time step, the meshes are, in general, not coincident. The transport volumes between elements is determined from the Lagrangian coordinates of each mesh, and the Eulerian coordinates, which are common across the meshes.

Since the material interfaces are calculated by the same interface reconstruction methods as before, parallelization doesn’t require any additional information to be exchanged between processors. There is a minor memory and processor burden associated with the transport of the independent velocity fields.

4. Modeling nonuniform materials

There are two common situations where modeling nonuniform materials are important. The first occurs at the mesoscale when the material is genuinely nonuniform in its properties. The second, which is even more common, is to avoid having large volumes of material fail simultaneously. For example, in a tension test, the stress state in the material is designed to be uniform. A failure criterion based on stress or equivalent plastic strain will result in a large volume of material to fail instead of generating a nice fracture surface. If the material properties vary spatially, the stress state will not be uniform, and the desired fracture surface will be generated [10–12].

Varying the material properties in the mesh is straightforward in Lagrangian formulations: each element is simply randomly assigned material parameters by appropriately sampling the desired distribution. This approach does not work for Eulerian codes because the transport algorithm quickly smears out the spatial variations as shown in figure 3.

To circumvent this problem, the initial coordinates are transported instead, and their values are used to interpolate the appropriate random material variables from a fixed, reference mesh [12]. To make this simple idea concrete, consider two meshes. The first is the standard mesh used for the analysis, and the current coordinates of a material point are $x$. Associated with this mesh are all the solution variables, e.g., the velocity, the stress, the density, and etc. In this mesh, the transported solution variables are augmented by the initial coordinates $X$ of the stress. The second mesh is the reference mesh, with coordinates $X$. Note that the coordinates of the nodes for the reference mesh are constant for all time, as are the values of the spatially varying material properties associated with the nodes. Furthermore, just to make
Figure 3. The diffusion of a random field due to transport with the second order MUSCL algorithm. On the left are the initial MUSCL reconstruction with a width 20 elements, and on the right is the final MUSCL reconstruction after transporting across 100 elements.

Figure 4. Failure surfaces as a function of the normalized random variations $\delta$ in the plastic strain for failure. The colors show the normalized variation. Green is 1, with blue for $1-\delta$ and red for $1+\delta$.

The calculation of the material properties as simple as possible, the reference mesh is assumed to have a uniform element size of $\triangle X_i$ in the $i$-th direction. At any material point and time in the analysis mesh, the logical address for the reference element containing the material point is $(X_1/\triangle X_1 + 1, X_2/\triangle X_2 + 1, X_3/\triangle X_3 + 1)$, and the value of the material parameters is linearly interpolated from its eight nodes. This strategy permits the initial profile in figure 3 to be perfectly maintained during the transport process.

It is important to recognize that the spatial variations are material properties that should be measured experimentally if at all possible. At the macroscale, spall may be idealized as occurring on a single smooth plane. If the material fails once it exceeds a specified value of the equivalent plastic strain in the calculation, the calculation will predict the idealized plane. The region experiencing spall, however, will broaden and generate an increasingly rough failure surface as the amplitude of the variation of the failure strain increases (see figure 4).

5. Future directions
The success of current mesoscale calculations does not imply that there is not substantial room for improving them or that there are many topics that still need to be addressed. A few
future directions of research are suggested here; most will also be beneficial to macroscopic scale calculations.

Many material models incorporate damage. Typically there is a damage variable that has a range of $[0, 1]$, where a material that has completely failed has a value of 1. Any monotonic transport algorithm will necessarily diffuse the damage, leading to the familiar problem of a failed material “healing.” Special algorithms that are applied after transport need to be developed.

Transport algorithms are designed for transporting the solutions variables independently. After transporting the stress tensor, there is no guarantee that the stress remains on the yield surface defined by the equivalent plastic strain, and returning the stress to the yield surface reduces the elastic energy of the system. Depending on the material model, there may be additional constraints between the state variables, e.g., the stress tensor may have constraints on its invariants. Transport methods that can satisfy general algebraic constraints need to be developed.

Not all material models work equally well with an Eulerian formulation. Some models, especially those with complicated yield stresses, may give nonphysical results. To date, there has not been any systematic investigation of theses problems. One possibility is the choice of the state variables for the model. While the state variables after transport are guaranteed to be monotonic, their differences are not. Simple operations on the state variable may therefore generate nonphysical results.

References
[1] Benson D J 1999 Modelling and Simulation in Materials Science and Engineering 7 333–354
[2] Angell I O and Griffith G 1987 High-Resolution Computer Graphics Using FORTRAN 77 (MacMillan Education Ltd.)
[3] Benson D 1992 Comput. Method. Appl. M. 99 235–394
[4] Benson D 1997 Comput. Method. Appl. M. 140 59–86
[5] Benson D 2002 Appl. Mech. Rev. 55 151–165
[6] Vitali E, Wei C T, Benson D J and Meyers M A 2011 Acta Materialia 59 5869–5880
[7] Vitali E and Benson D 2006 Int. J. Numer. Meth. Eng. 67 1420–1444
[8] Vitali E and Benson D 2008 Int. J. Numer. Meth. Eng. 76 893–921
[9] Vitali E and Benson D 2009 Comput. Mech. 43 847–857
[10] Becker R 2002 Int. J. Solids Struct. 39 3555–3580
[11] Brannon R, Wells J and Strack O 2007 Metall. Mater. Trans. A 38 A 2861–2878
[12] Vitali E and Benson D 2012 Comput. Mech. 49 197–212