LA-UR-09-01324

Approved for public release;
distribution is unlimited.

Title: A comparative study of interface reconstruction methods for multi-material ALE simulations

Author(s): Milan Kucharik, Rao V. Garimella, Samuel P. Schofield, Mikhail J. Shashkov

Intended for: Journal of Computational Physics

A comparative study of interface reconstruction methods for multi-material ALE simulations

Milan Kucharik

Applied Math and Plasma Physics (T-5), Los Alamos National Laboratory

Rao V. Garimella

Applied Math and Plasma Physics (T-5), Los Alamos National Laboratory

Samuel P. Schofield

Applied Math and Plasma Physics (T-5), Los Alamos National Laboratory

Mikhail J. Shashkov

Applied Math and Plasma Physics (T-5), Los Alamos National Laboratory

Abstract

In this paper we compare the performance of different methods for reconstructing interfaces in multi-material compressible flow simulations. The methods compared are a material-order-dependent Volume-of-Fluid (VOF) method, a material-order-independent VOF method based on power diagram partitioning of cells and the Moment-of-Fluid method (MOF). We demonstrate that the MOF method provides the most accurate tracking of interfaces, followed by the VOF method with the right material ordering. The material-order-independent VOF method performs somewhat worse than the above two while the solutions with VOF using the wrong material order are considerably worse.

Key words:
Interface Reconstruction, Moment-of-fluid method, compressible flow

Email addresses: `kucharik@lanl.gov`, `kucharik@karkulka.fjfi.cvut.cz` (Milan Kucharik), `rao@lanl.gov` (Rao V. Garimella), `sams@lanl.gov` (Samuel P. Schofield), `shashkov@lanl.gov` (Mikhail J. Shashkov).

1 Introduction

Accurate simulation of multi-material and multi-phase flows, requires effective tracking and management of material interfaces. Due to their ability to strictly conserve the mass of different materials, volume-of-fluid (VOF) methods using interface reconstruction are widely used in such simulations [1–4]. Originally developed by Hirt and Nichols [5], VOF methods do not explicitly track interfaces but rather track the volume of each material. The interface between materials is first reconstructed in cells based on the material volume fractions. Then the volume fluxes of each material between cells are estimated from the geometric reconstruction and finally, the fluxes are used to compute new volume fractions in each cell, in preparation for the next time step.

More recently, an interface tracking method has been devised based on tracking both the volume (zeroth moment) and centroid (ratio of first and zeroth moment) of the materials in mesh cells. This new method, called the Moment-of-Fluid (MOF) method [6], reconstructs interfaces more accurately than VOF methods and is able to resolve interfacial features on the order of the local mesh size whereas VOF methods do poorly in resolving features smaller than 3-4 times the local mesh size.

In this paper, we present a comparative study of different VOF methods and the MOF method for a complex compressible flow simulations involving more than two materials.

2 VOF Methods with Nested Dissection (VOF-PLIC)

Early VOF methods used a straight line aligned with a coordinate axis to partition the cell according to the material volume fractions. This is often referred to as the simple line interface calculation (SLIC) originally due to Noh and Woodward [7]. Youngs [8,9] extended the method to permit the material interface to have an arbitrary orientation within the cell (called PLIC or Piecewise Linear Interface Calculation by Rider and Kothe [3]). In Youngs' method, the outward normal of the interface separating a material from the rest of the cell is taken to be the negative gradient of the "volume fraction function". The "volume fraction function" is treated as a smooth function whose cell-centered values are given by the cell-wise material volume fractions. The interface is then defined by locating a line with the prescribed normal that cuts off the correct volume of material from the computational cell.

Gradient based methods are in general first order accurate although they may exhibit near second order accuracy on regular Cartesian grids. However,

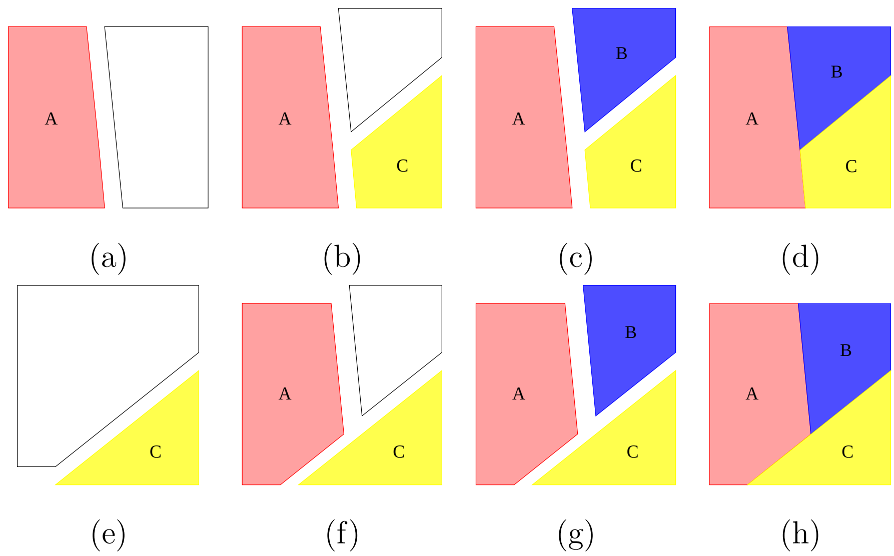


Fig. 1. Nested dissection interface reconstruction for three materials in the order ACB: (a) the first (A) material is removed leaving a smaller available polygon, (b) the second (C) material is removed from the available polygon, (c) the remaining available polygon is assigned to material B, (d) the resulting partitioning of the computational cell. (e)-(g) show the same procedure but the materials are processed in a different (CAB) order leading to a different reconstruction (h).

there are extensions that make the reconstruction second-order accurate for general grids. The LVIRA technique by Pilliod and Puckett [10] tries to find an extended straight line interface that cuts off the exact volume fraction in the cell of interest and minimizes the error in matching the volume fractions in the surrounding cells. LVIRA uses a minimization procedure with a gradient-based normal as the initial guess. An alternative is the interface smoothing procedure based on Swartz's quadratically convergent procedure [11] for finding a straight line that cuts off the right volume fractions from two arbitrary planar shapes [1]. Mosso et.al [14] and Garimella et.al. [15] have used this procedure in slightly different ways to devise interface smoothing procedures. For a given mixed cell, Garimella et.al. compute a straight line cutting off the right volume fractions from the cell and each of its mixed cell neighbors by the Swartz method. The normals of these different straight lines are then averaged to give a smoothed interface normal for the cell.

VOF-PLIC techniques have been successfully used to accurately simulate two-phase (or two-material) flows and free-surface flows in two and three dimensions. However, their application to flows involving three or more materials that come closer than the mesh spacing and even form junctions has been mostly *ad hoc*. Examples of such phenomena are flows of immiscible fluids (*e.g.* oil-water-gas), inertial confinement fusion, armor-antiarmor penetration and powder metallurgical simulation of multiple materials.

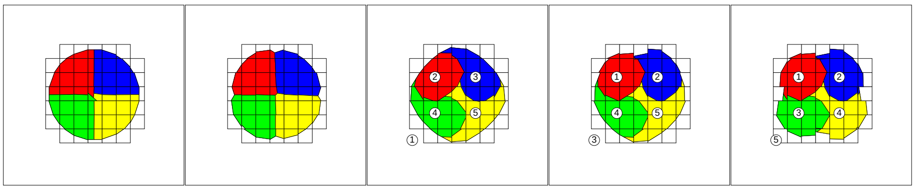
[1] This is commonly known as the "ham-sandwich" or Steinhaus problem[12,13]

The most common extensions of PLIC to cells with more than two materials (multi-material cells) [2], is to process materials one by one leading to a reconstruction that is strongly dependent on the order in which the materials are processed. Of the different ways to sequentially partition a cell, one of the most general and accurate ways is called the "nested dissection" method [6], where each material is separated from the others in a specified order. In the method, a pure polygon (or polyhedron) representing the first material is marked out from the cell, leaving a mixed polygon for the remaining materials. Then, a polygon representing the second material is marked out from the mixed polygon and the process continues until the last material is processed. This method is illustrated in Figure 1 and described in detail in [6,16,17]. Clearly, such an order dependent method can easily place materials in wrong locations in the cells if the chosen order of processing is incorrect. Even if the order of the materials is right, the computation of the interface normals in multi-material cells is ambiguous. In computing the normal as the negative gradient of the volume fraction function of a material, it is unclear whether one should use the volume fractions with respect to original cells or the part of the cells remaining after the earlier materials have been removed. It is also not clear where these function values should be centered – at the center of the original cell or the center of the unprocessed part of the cell.

The most significant adverse effect of these incorrect reconstructions, however, is in material advection in flow simulations. An improper material ordering may result in materials being advected prematurely (or belatedly) into neighboring cells. This can further lead to small pieces of the material getting separated and drifting away from the bulk of the material (sometimes known as "flotsam and jetsam"). The effect of material ordering is illustrated clearly in an example from [18] in which a four-material disk (with each material occupying one quadrant of the disk) is advected diagonally for 30 time steps. The results in Figure 2 show dramatically different results with different material orderings and a complete loss of the cross-shaped interface.

The most common and trivial way to deal with the material order dependency is to select the "correct" global ordering for a problem. However, this is obviously problematic if the same materials must be processed differently in different parts of the mesh or if the material configurations change as the problem advances in time. Also, some interface configurations may not be reproducible by any particular order, such as the four material example referred

[2] In a strict sense, any cell with more than one material is a multi-material cell. However, we choose to distinguish two material cells from cells with more than two materials by calling the latter multi-material cells. The reason for this distinction is that interface reconstruction for one material is (in the case of VOF methods) complementary to the second in a cell with two-materials while it is not for more than two materials



MOF Powerdiagram Youngs^1 Youngs^2 Youngs^3

Fig. 2. Four material disk at time $T = 0.5$ translated from the initial position $(0.2, 0.2)$ with 30 time steps at a velocity of $(1, 1)$ on 32×32 mesh of the $[0, 1]^2$ domain. Material reconstruction done by several methods – MOF, VOF with power diagrams, and Youngs' VOF. The material orderings for Youngs' are indicated in the figure.

to above. While there has been some work on automatically deriving material order, most of these attempts assume a layered structure for the interface [14,19] and cannot handle multiple materials coming together at a point very well.

3 VOF Methods with Power Diagram Reconstruction (VOF-PD)

Recently, Schofield et. al. [18] developed a new VOF-based reconstruction method that is completely material order independent. This method, called the Power Diagram method for Interface Reconstruction, does not sequentially carve off materials from a cell using straight lines. Rather it first locates materials approximately in multi-material cells and then partitions the cell simultaneously into multiple material regions using a weighted Voronoi decomposition thereby avoiding the order dependence problem. We describe this procedure below referring to it as the VOF-PD method.

In the first step of the VOF-PD method, approximate locations or "centroids" of the materials in a cell are determined using the volume fractions of the materials in the cell and its neighbors. This is accomplished by treating the volume fractions of each material in the cell and its neighbors as pointwise values of a pseudo-density function. The pointwise values of this pseudo-density function are then used to obtain a linear reconstruction of the function along with application of a limiter restricting the minimum and maximum values to 0 and 1 respectively. Then the linear approximation of this pseudo-density function is used to derive an approximate centroid for the material in the cell. While this method does not locate the material centroids very accurately in an absolute sense, it does locate the materials quite well relative to each other.

In the second step of the procedure, the approximate centroids of the materials are used as generators for a weighted Voronoi or Power Diagram subdivision [20,21] of the cell. The weights of the different generators are chosen iteratively such that the volume fractions of the different Voronoi polygons truncated by the cell boundary match the specified material volume fractions exactly.

The authors have shown that this procedure is in general first-order accurate and for two materials, exactly reproduces a gradient-based subdivision of the cell. They have also presented a smoothing procedure for the power diagram-based subdivision which results in a second-order accurate reconstruction but slows the procedure down considerably unless applied only to cells with more than two materials.

4 Moment-of-Fluid (MOF) Method

While VOF methods track only volume fractions of the individual materials in mesh cells, the recently developed Moment-of-Fluid (MOF) method [6] tracks both the volume (zeroth moment) and centroid (ratio of first and zeroth moment) of the materials in the cells. By tracking both moments the MOF method reconstructs the material interface with higher accuracy than VOF methods and is able to resolve interfacial details on the order of the local mesh size. In contrast, VOF methods can only resolve details on the order of 3-4 times the local mesh size. Also, since a line can be determined by only two parameters (an intercept and a slope), the linear interface in a cell is actually over-determined by specifying the volume fraction and centroid. This implies that MOF can perform an exact reconstruction of a linear interface and a second-order reconstruction of a smoothly curved interface in a cell without the need for information from neighboring cells.

Given the volume fraction and centroid of a material in a cell, the MOF reconstruction method computes a linear interface such that the volume fraction of the material is exactly matched and the discrepancy between the specified centroid and the centroid of the polygon or polyhedron behind interface is minimized. This is done by an optimization process with the slope of the linear interface (or its angle with respect to the x-direction) as the primary variable. For any given slope, the intercept of the line is determined uniquely by matching the specified material volume fraction.

The MOF reconstruction is also typically implemented as a nested dissection method where materials are carved off from a cell sequentially thereby making it an order-dependent problem. However, it is possible to combinatorially determine the correct sequence of material reconstructions in MOF by reconstructing with all possible sequences and choosing the sequence which leads to the least discrepancy between the reconstruction and specified centroids. More complex configurations such as 4 materials coming together at a point can be reconstructed by recursively reconstructing the interface between groups of materials first and then resolving the interfaces between materials in each group. Since the number of materials in a cell is typically small, this does not impose a significant computational penalty. Such a technique has proved very effective in accurately reconstructing multi-material interfaces.

Further details of the MOF technique of interface reconstruction are given in [6,17].

5 Compressible flow simulation with VOF and MOF reconstructions

Here we briefly describe an arbitrary-Eulerian-Lagrangian (ALE) compressible flow simulation algorithm used to compare the effects of the VOF and MOF reconstruction techniques. Since the purpose of this paper is to compare the different interface reconstruction methods, we deliberately do not provide many details of the ALE code to avoid overwhelming the discussion. We believe the general conclusions of this comparative study will hold regardless of the ALE code used.

Our 2D research multi-material ALE code (RMALE) has a standard structure shown in Figure 3. It consists of three main components – multi-material

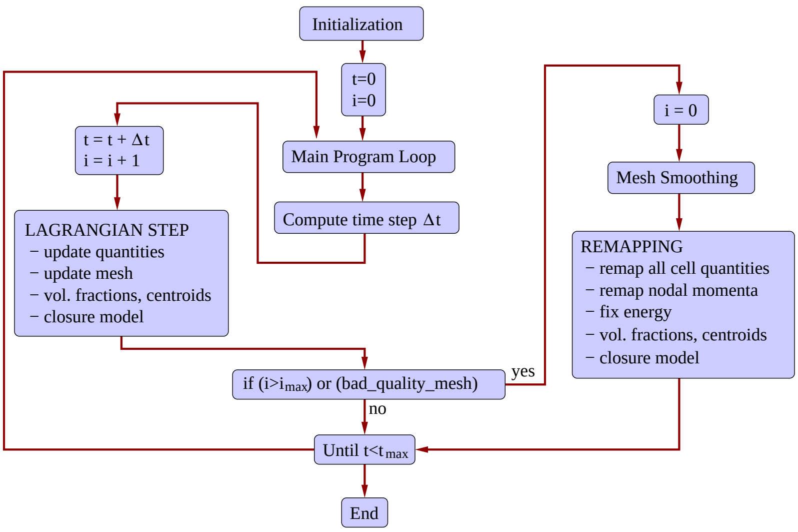


Fig. 3. Flowchart of our research multi-material code. Material reconstruction is hidden in the update of material centroids at the end of the Lagrangian step.

Lagrangian solver, mesh untangling and smoothing method, and a flux-based multi-material remapper. The Lagrangian step is repeated, until the mesh smoothing condition is fulfilled (for example, poor mesh quality, or rezoning counter reaches given number of hydro steps). When mesh smoothing is applied to improving the mesh quality it is followed by a remapping step conservatively interpolating all quantities on the new mesh. Then, a new Lagrangian cycle can begin. The entire code employs a staggered Mimetic Finite Difference discretization [22], where scalar fluid quantities (density, mass, pressure, internal energy) are located inside mesh cells, and vector quantities (positions, velocities) on mesh nodes. The multi-material ALE framework allows more than one material inside one computational cell, where the amount of each material is defined by its volume and mass fractions, and if we use MOF, the relative location of each material is defined by the material centroid. In each multi-material cell, scalar quantities are defined separately for every material, but the variables in the primary equations are the average cell quantities. Contrary to a single-material approach, our multi-material Lagrangian step and remapper must update not only all fluid quantities, but also material volume and mass fractions, and material centroids.

The Lagrangian solver solves the following set of hydrodynamic equations

$$\frac{1}{\rho}\,\frac{d\,\rho}{d\,t} = -\nabla\cdot\mathbf{w}\,, \quad \rho\,\frac{d\,\mathbf{w}}{d\,t} = -\nabla\cdot p\,, \quad \rho\,\frac{d\,\varepsilon}{d\,t} = -p\,\nabla\cdot\mathbf{w} \tag{1}$$

representing conservation of mass, momenta in both directions, and total energy, completed by the ideal gas equation of state $p = (\gamma - 1)\,\rho\,\varepsilon$. Here, ρ is the fluid density, $\mathbf{w}$ is the vector of velocities, p is the fluid pressure, ε is the specific internal energy, and γ is the ratio of specific heats. The solver is based on evaluation of several types of forces affecting each mesh node [22] – zonal pressure force representing forces due to the pressure in all neighboring zones, artificial viscosity force (edge viscosity [23] is used in the examples), and anti-hourglass stabilization force introduced in [24], suppressing some unphysical modes in the mesh motion. For volume fraction update and common pressure construction, a multi-material closure model is applied [25], which adjusts the material volume fractions such that material pressures equilibrate to a common pressure value. The last part of the Lagrangian step is a method for updating the material centroids. In the first step, we advect them by keeping their parametric coordinates constant. Appendix A shows that this method reproduces the Lagrangian motion of the centroid for compressible flows with second-order accuracy. These centroids are then used (together with updated volume fractions) as reference centroids for the next material reconstruction step. The final material centroids are then set to the centroids of the reconstructed polygons.

Our code incorporates several mesh-untangling and mesh-smoothing methods. All ALE examples in this paper use classical Winslow mesh smoothing algorithm [26].

The last essential part of the ALE code is a remapping technique interpolating all fluid and material quantities between Lagrangian and smoothed computational meshes. Our flux-based remapper uses the multi-material extension of the technique described in [27] – it constructs inward and outward fluxes of integrals of 1, x, y, and some higher order polynomials using overlays (intersections) of Lagrangian cells (or pure material polygons in the case of mixed cells) with their neighbors in the smoothed mesh, and vice versa. Note that these integrals of polynomials over polygons can be computed analytically. These integrals are then used for construction of fluxes for all cell- and material-centered quantities. They are also used for advancing material volumes (and consequently volume fractions) and centroids in a flux form. For remapping nodal mass, we need to construct inter-nodal mass fluxes, which we interpolate from inter-cell mass fluxes as described in [28], extended by split side fluxes for adjacent cells and corner fluxes. All nodal quantities are then remapped by attaching them to these inter-nodal mass fluxes (for example, the momentum fluxes are obtained by multiplication of the mass fluxes by an interpolated flux velocity). This approach allows us to construct two kinetic energies at each node – conservative kinetic energy obtained by its remap, and non-conservative kinetic energy obtained from remapped velocities. This kinetic energy discrepancy is resolved by a standard energy fix [1], it is redistributed into the remapped internal energy of adjacent materials, and thus global energy conservation is guaranteed.

6 Problem description

We demonstrate the properties of the described material reconstruction methods in the context of multi-material ALE hydrocode for a triple point problem suggested by Maire [29]. The initial data for this problem is shown in Figure 4.

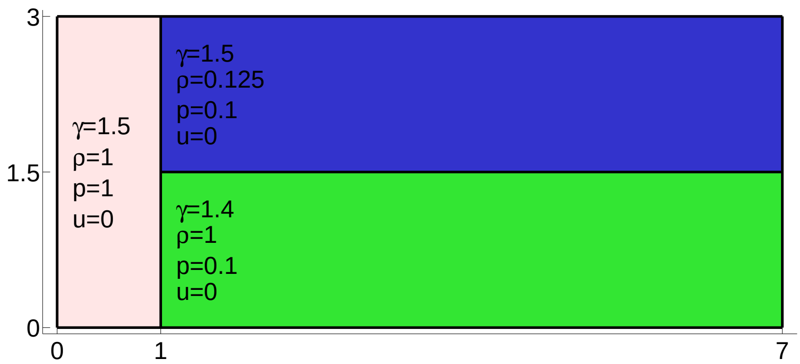


Fig. 4. Initial conditions for static triple point problem. Materials are shown in different colors, and values of ratio of specific heats γ, density ρ, pressure p, and velocity u are listed.

The computational domain has a rectangular shape with 7×3 edge ratio. In all simulations, we use an equidistant orthogonal initial computational mesh with 140×60 cells. It includes three materials at rest, initially forming a T-junction. The high-pressure material (in light red or white) creates a shock wave moving to to the right, through the low pressure blue (or darkest gray) and green (medium gray) materials. Due to different material properties, it moves faster in the blue or dark gray (lower density) material, and therefore a vortex evolves around the triple point. In the later stages of the simulation (final time $T = 5$), we can observe thin filaments of materials rotating around the vortex. In our comparison, we focus especially on the material topology (relative position of the materials) and on how well the thin tip of green or light gray material filament is resolved.

It is to be noted that no mixed cells are present at the beginning of the simulations, however, they appear during the first remap.

7 Results

Here, we compare a traditional gradient-based VOF method with different orderings, the MOF method, and a VOF method based on power diagrams (VOF-PD). We perform the comparison for two types of simulations: Eulerian and full ALE. In the Eulerian approach, the solution is remapped back to the orthogonal initial mesh after each Lagrangian step, while in the ALE approach, Winslow mesh smoothing and consecutive remapping is performed after every 20 Lagrangian steps.

In Figure 5, we can see the first snapshot of the Eulerian simulation, corresponding to time $T = 0.1$. In this early moment, the white-blue interface is shifted more to the right than the white-green one. As we can see, smooth interfaces are preserved when using VOF starting with white material, which is the correct local material ordering for this particular problem, and when using the MOF method. The VOF with Power diagrams still provide acceptable results, while VOF methods using wrong orderings created very distorted interfaces leading to problems in later stages of the simulation.

A snapshot in the middle of the simulation ($T = 2.5$) is shown in Figure 6. A thin filament of green material is starting to develop, which is reasonably resolved using MOF and VOF with the correct ordering. VOF with power diagrams keeps the correct topology of materials, but starts to have problems with resolving the thin filament. VOF with the wrong material orderings provides the worst results – the filament starts to separate from the heavy blue material, and there are small pieces of white material between green and blue that are not easily visible at this scale.

In Figure 7, we can see the final snapshot of the Eulerian simulation corre-

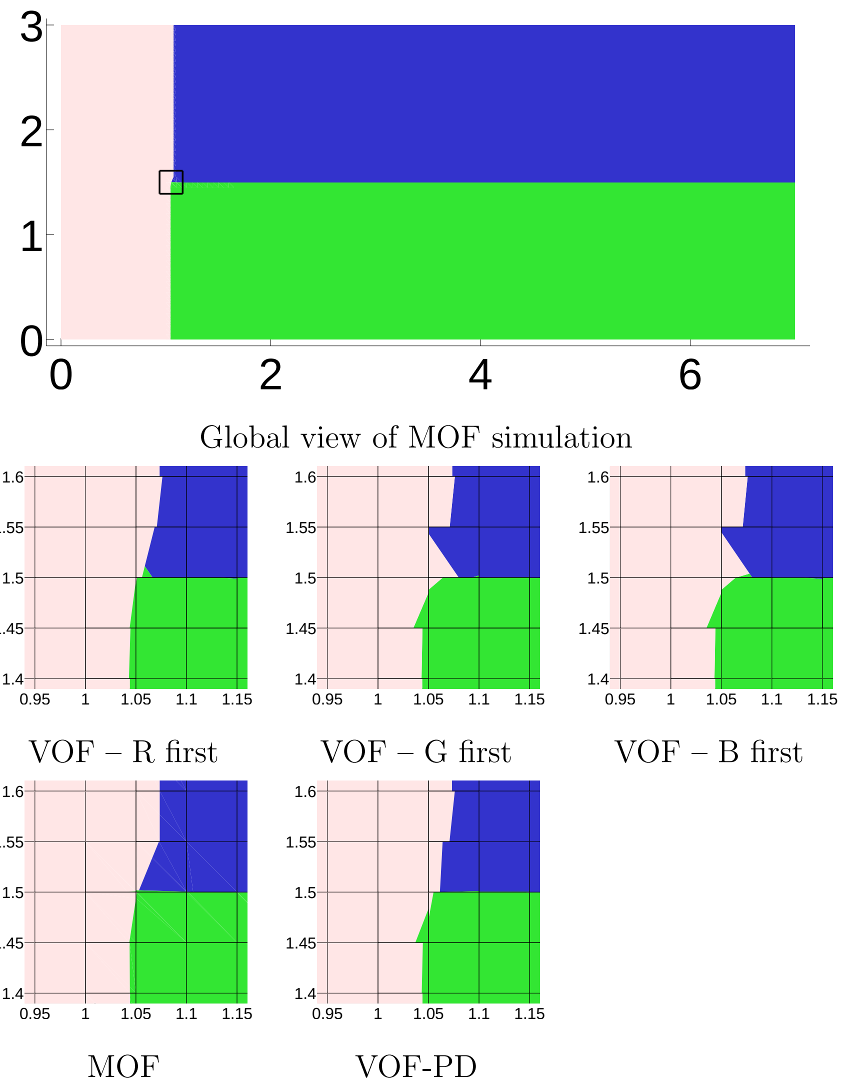


Fig. 5. Materials of triple point problem simulation, time $T = 0.1$. Eulerian runs (as Lagrangian step and remap to the initial orthogonal mesh) using different methods for material reconstruction are shown: global view on the entire computational domain for MOF method, and zooms to the three material junction for Young's VOF method (with different material orderings), MOF, and Power Diagram based methods are shown.

sponding to time $T = 5$. Again, MOF and VOF in the correct ordering resolve the thin part of the green filament reasonably well. VOF with the wrong material orderings give us unacceptable results – filament transforms into a drip separating from the blue material, and there are many tiny droplets of white material between the blue and green materials VOF with power diagrams also do not succeed in resolving the thin part of the filament, but the result is qualitatively better: the material topology is correct, no droplets appear, and green material stays attached to the blue one.

In the next set of figures, the results of the same problem obtained by ALE

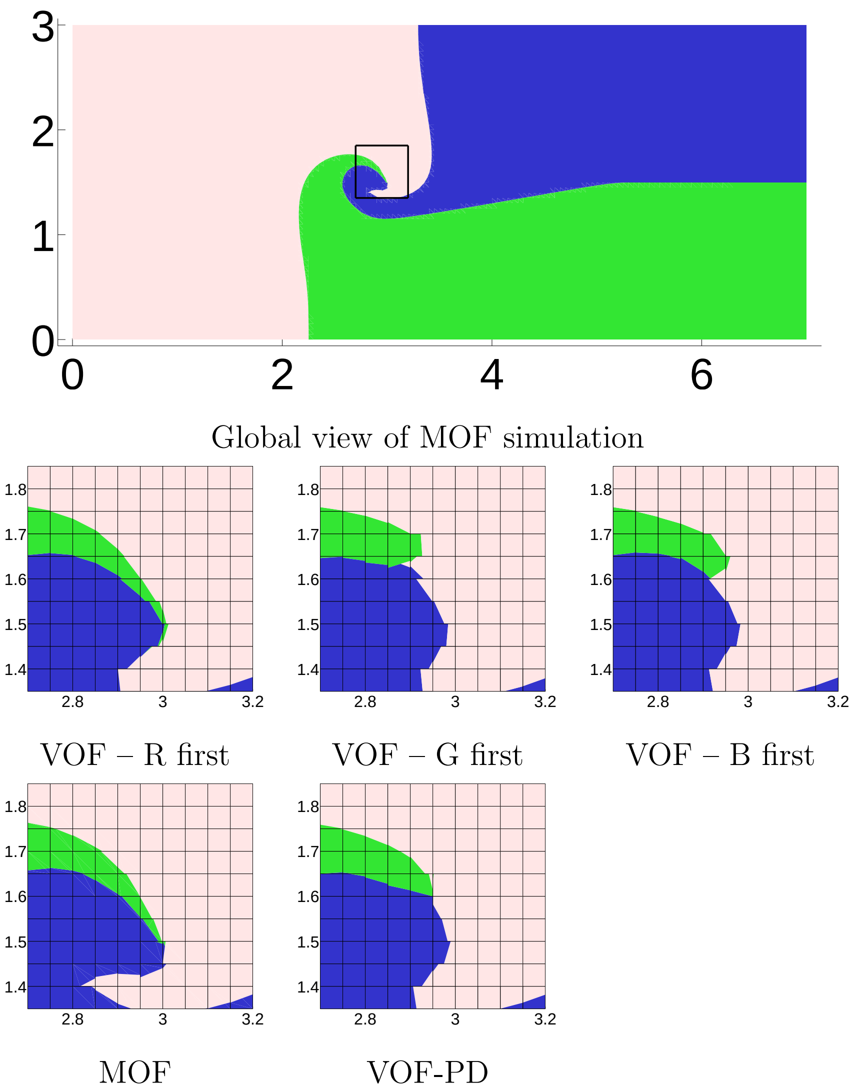


Fig. 6. Materials of triple point problem simulation, time $T = 2.5$. Eulerian runs (as Lagrangian step and remap to the initial orthogonal mesh) using different methods for material reconstruction are shown: global view on the entire computational domain for MOF method, and zooms to the three material junction for Young's VOF method (with different material orderings), MOF, and Power Diagram based methods are shown.

approach are presented. Generally, the results are worse than for the Eulerian simulations due to the distorted computational mesh.

In Figure 8, the early stages of an ALE simulation at time $T = 0.1$ are presented for the same example. As we can see, the MOF results are best of all methods being compared, the multi-material interface smoothly transitions from the white-blue to the white-green interface and no major jumps appear. The results of VOF in correct ordering are comparable to the results of VOF with power diagrams at this early stage. We can observe minor material jumps and smoothness of the interface is violated. The worst results are clearly ob-

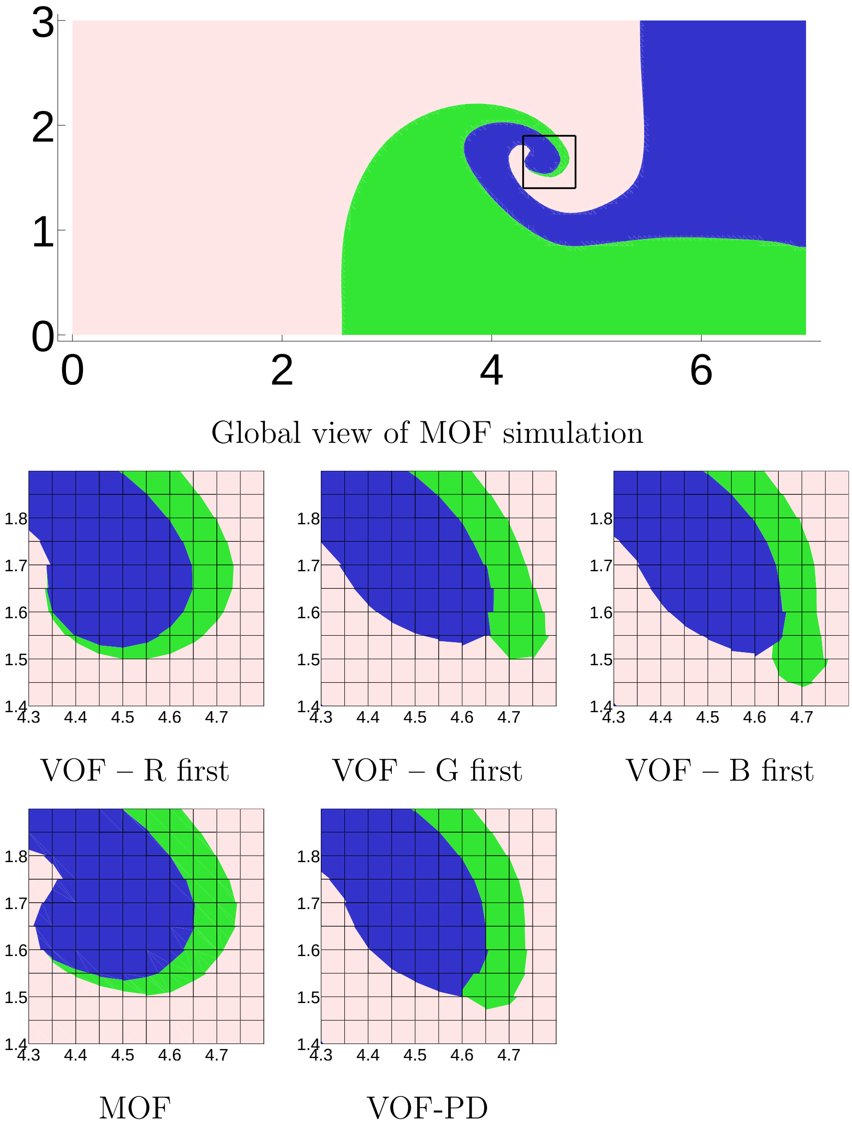


Fig. 7. Materials of triple point problem simulation, time $T = 5.0$. Eulerian runs (as Lagrangian step and remap to the initial orthogonal mesh) using different methods for material reconstruction are shown: global view on the entire computational domain for MOF method, and zooms to the three material junction for Young's VOF method (with different material orderings), MOF, and Power Diagram based methods are shown.

tained by VOF methods using the wrong material orderings. The T-shape of the interface is completely violated and an unphysical wedge of white material starts to separate blue and green materials, leading to more severe problems in later stages of the simulations.

Figure 9 presents results in the middle of the simulation ($T = 2.5$). In this time moment, the (initially orthogonal) computational mesh is already relatively distorted. As we can see, VOF in correct ordering resolves the longest green filament. Filament resolved by MOF is shorter, compact, with a relatively

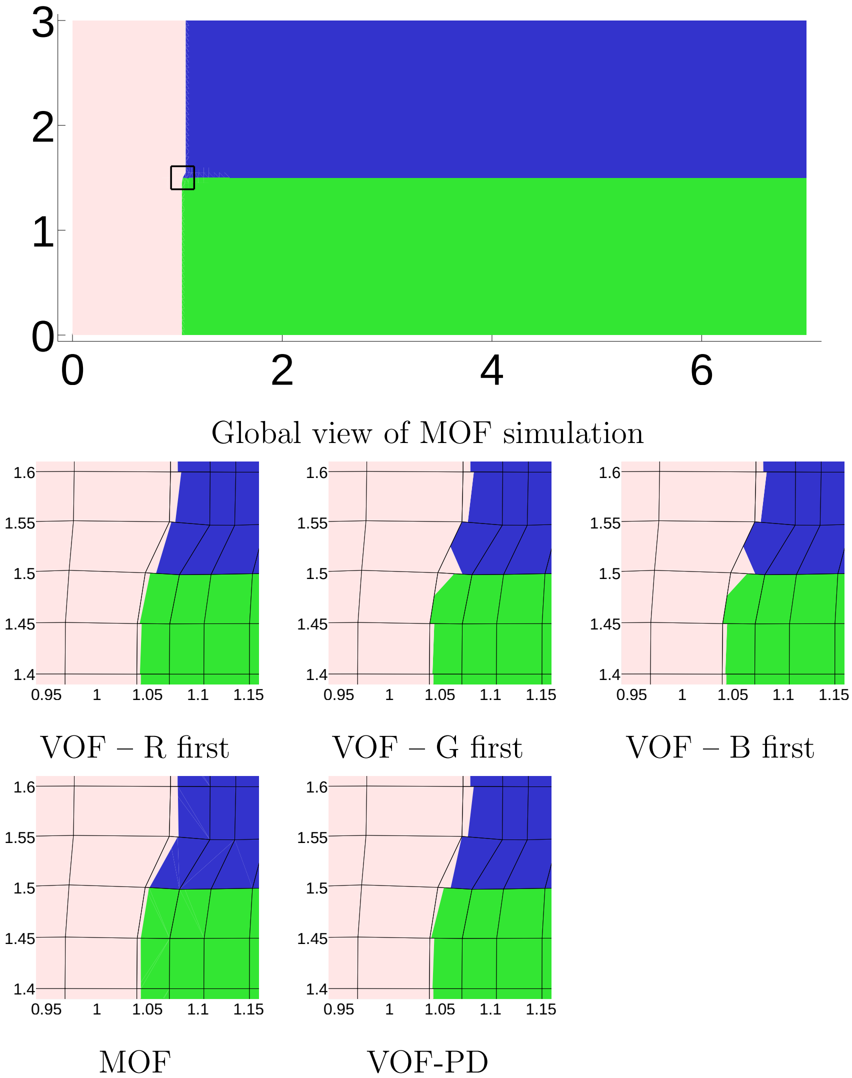


Fig. 8. Materials of triple point problem simulation, time $T = 0.1$. ALE runs (as Lagrangian step and remap to the Winslow smoothed mesh after every 20 Lagrangian steps) using different methods for material reconstruction are shown: global view on the entire computational domain for MOF method, and zooms to the three material junction for Young's VOF method (with different material orderings), MOF, and Power Diagram based methods are shown.

smooth interface. Power diagrams and VOF with wrong material orderings do not resolve the filament very well, but power diagrams surpass VOF in material topology – no fragment of white and blue material appear on the other side of the green filament.

In Figure 10, we can see the last moment ($T = 5$) of the ALE simulation. MOF provides best result again – the filament is compact, relatively smooth, no separated tiny droplets are present. We can observe such small pieces for all VOF methods, even for correct ordering, where a tiny thin fiber of green material separates white-blue interface upto the picture boundary. As for power dia-

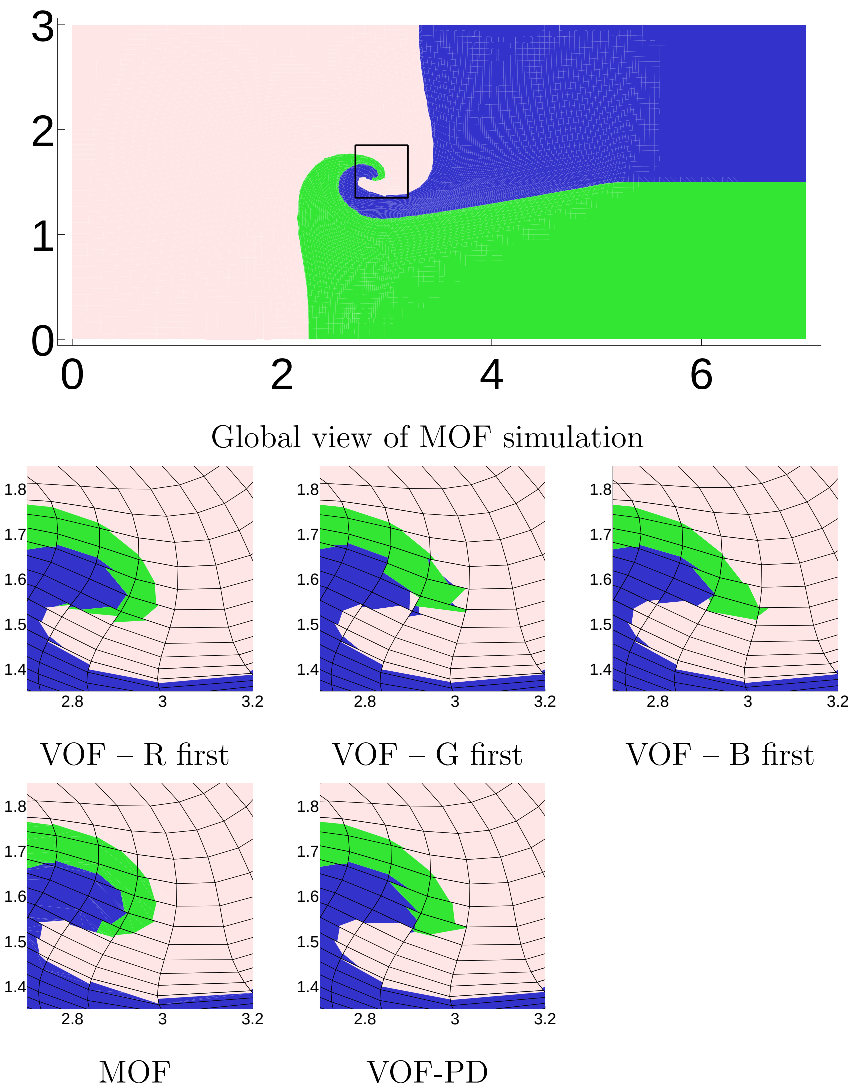


Fig. 9. Materials of triple point problem simulation, time $T = 2.5$. ALE runs (as Lagrangian step and remap to the Winslow smoothed mesh after every 20 Lagrangian steps) using different methods for material reconstruction are shown: global view on the entire computational domain for MOF method, and zooms to the three material junction for Young's VOF method (with different material orderings), MOF, and Power Diagram based methods are shown.

grams, no droplets appear, but we can see that the green filament has broken into two parts.

8 Conclusions

We have presented a comparison of a material-order-dependent VOF method, a material-order-independent VOF method and a material-order-independent

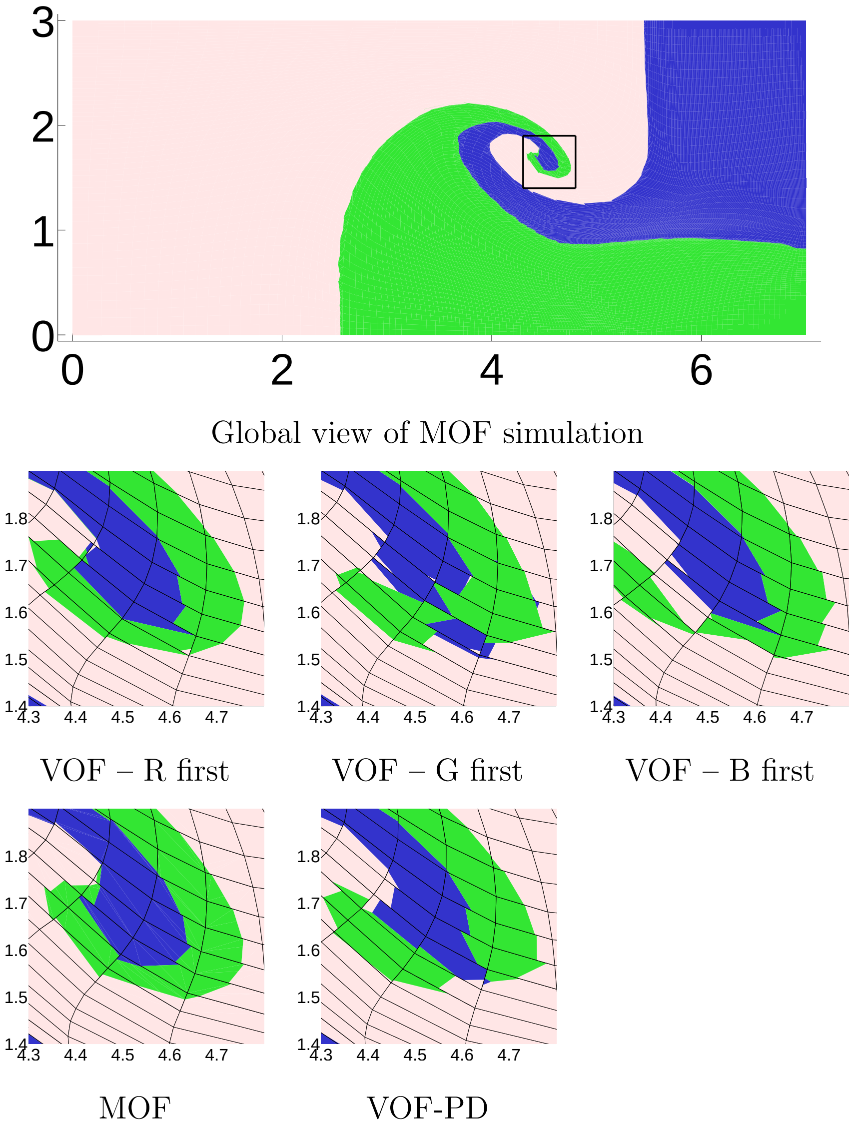


Fig. 10. Materials of triple point problem simulation, time $T = 5.0$. ALE runs (as Lagrangian step and remap to the Winslow smoothed mesh after every 20 Lagrangian steps) using different methods for material reconstruction are shown: global view on the entire computational domain for MOF method, and zooms to the three material junction for Young's VOF method (with different material orderings), MOF, and Power Diagram based methods are shown.

MOF method for a complex compressible flow involving more than two materials. The VOF methods track volumes of fluids and the MOF method tracks both volumes and centroids of fluids. The first VOF method uses a nested dissection or sequential removal of materials from the cell to reconstruct the multi-material interface, making its results dependent on the material ordering. The second VOF method partitions the cells simultaneously into multiple material regions using a power diagram and is therefore, independent of any material order specification. The MOF method performs sequential subdivision of the cell but considers all possible material orders and chooses the one that minimizes the discrepancy between the specified and reconstructed moments.

From the simulations that we have run, we conclude that:

- MOF performs the most accurate reconstructions, generally capturing filaments accurately and getting the material topology correct. Since MOF is quite recent it generally does not exist in many codes. Therefore, this method is the best choice when developing new flow codes or when revamping the interface tracking machinery. It is not advisable to introduce MOF reconstruction into a flow code without ensuring that the advection (or remapping) of centroids is done accurately through overlays.
- VOF with the correct material order performs remarkably well although the resolution of filaments and other small features is poorer than MOF. Since VOF commonly exists in flow codes that perform this type of interface tracking, it is a natural choice when the flow is simple and the material order can be predicted quite easily. It is also a good choice when the flow has only two materials and no filamentary or other structures smaller than 3-4 times the grid resolution are expected.
- VOF with power diagrams performs more poorly than MOF or VOF with the right material order but usually gets the interface topology right. This method is a good choice when the advection machinery cannot be revamped to perform overlays but the interface reconstruction can be rewritten simply to partition cells using the power diagram.
- VOF with the wrong order performs poorly even for simple flows and is not advised. If the ordering cannot be predicted or enforced strictly, it is better to use VOF with the power diagram reconstruction.

A Appendix

A.1 Lagrangian update of material centroids

The Lagrangian step may be viewed as the implicit creation of a family of maps, $\phi^n(\mathbf{x}) : \mathbb{R}^d \mapsto \mathbb{R}^d$, such that $\mathbf{x}^{n+1} = \phi^{n+1}(\mathbf{x}^n)$. Any material region, $\Omega^t \subset \mathbb{R}^d$, evolves over a time step as

$$\Omega^{n+1} = \phi^{n+1}(\Omega^n) \tag{A.1}$$

The map ϕ^{n+1} is illustrated in Figure A.1.

If the map is an affine transformation, that is

$$\phi^{n+1}(\mathbf{x}) = A\mathbf{x} + \mathbf{b} \tag{A.2}$$

where $A \in \mathbb{R}^{d\times d}$ is invertible and $\mathbf{b} \in \mathbb{R}^d$, then if $\mathbf{x}_c(\Omega^n)$ is the centroid of the region and $\Omega^{n+1} = \phi^{n+1}(\Omega^n)$, then $\mathbf{x}_c(\Omega^{n+1}) = A\mathbf{x}_c(\Omega^n) + b$. That is, the transformed centroid is the centroid of the transformed region.

To demonstrate this,

$$\begin{aligned}
\|\Omega^{n+1}\|\mathbf{x}_c(\Omega^{n+1}) &= \int_{\Omega^{n+1}} \mathbf{x}\, dx \\
&= \int_{\Omega^n} (A\mathbf{y} + b) \det A\, dy \\
&= (\det A)\|\Omega^n\|A\mathbf{x}_c(\Omega^n) + \mathbf{b}(\det A)\|\Omega^n\|
\end{aligned}$$

Noting that

$$\|\Omega^{n+1}\| = \int_{\Omega^{n+1}} dx = \int_{\Omega^n} \det A\, dx = \|\Omega^n\| \det A$$

we obtain,

$$\mathbf{x}_c(\Omega^{n+1}) = A\mathbf{x}_c(\Omega^n) + b$$

The actual Lagrangian evolution of the region is given by the pointwise equation

$$\frac{d\mathbf{x}}{dt} = \mathbf{u}(\mathbf{x}, t) \quad \forall \mathbf{x} \in \Omega^t \tag{A.3}$$

assuming the velocity field is known. The transformation, ϕ^{n+1}, is then the solution to Equation A.3 over the time interval $[t^n, t^{n+1}]$.

With sufficient regularity, the velocity field can be expanded as

$$u_j(\mathbf{x}, t) = u_j(\mathbf{x}_0, t^n) + (t - t^n)\frac{\partial u_j(\mathbf{x}_0, t^n)}{\partial t} + (x_i - x_i^0)\frac{\partial u_j(\mathbf{x}_0, t^n)}{\partial x_i} + \mathcal{O}(\Delta x^2) + \mathcal{O}(\Delta t^2) + \mathcal{O}(\Delta x \Delta t). \tag{A.4}$$

Substituting this into Equation A.3 and integrating, we find that

$$\phi^{n+1}(\mathbf{x}) = \mathbf{x} + \mathbf{u}(x_0, t^n)\Delta t + \mathcal{O}(\Delta t^2) + \mathcal{O}(\Delta t \Delta x). \tag{A.5}$$

Assuming $\Delta t \approx \Delta x$, then the transformation defining the Lagrangian evolution over a time step may be approximated as an affine transformation with second order accuracy.

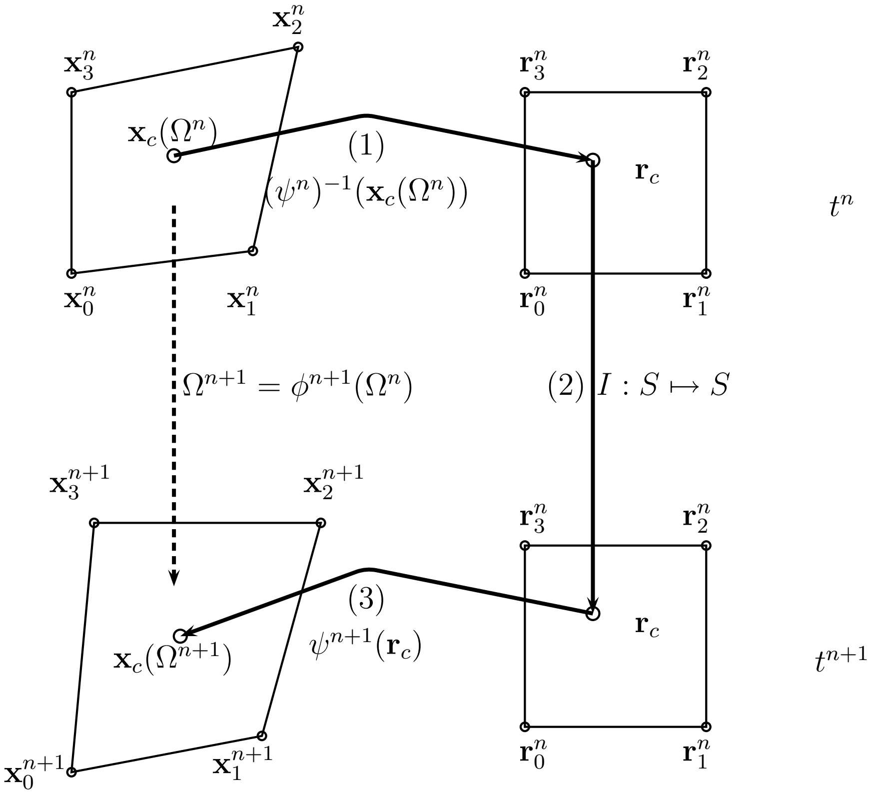


Fig. A.1. Steps in the constant parametric coordinate method. (1) The logical coordinates of the centroid at time t^n are calculated. (2) It is assumed the centroid has the same logical coordinates at time t^{n+1}. (3) The logical coordinates are mapped to physical coordinates to give the location. This gives a second order accurate approximation to the centroid of the evolved region $\Omega^{n+1} = \phi^{n+1}(\Omega^n)$.

A.2 Constant parametric coordinate method

A method for updating material centroids during a Lagrangian step can exploit this implicit evolution operator described above. The method described [2,19] is based on the existence of a mapping of the computational cell to and from a logical space. It is assumed that the centroid of the material region has the same logical coordinates, before and after the Lagrangian motion of the cell. To obtain the centroid after the Lagrangian motion, the logical coordinates of the centroid at the previous step are given to the logical to physical mapping corresponding to the cell after the motion. This process is illustrated in Figure A.1. It is important to note that the logical to physical space mapping is different for each time step and the cells evolve in time.

The accuracy of the method relies on the properties of the logical to physical coordinate transformations used.

Assume each cell has local coordinates, $\mathbf{r} \in S$, with an invertible map into physical coordinates, $\psi^n : S \mapsto \Omega^n$.

We define a family of local parameterizations, $\{\psi^n\}$ to be **linearity preserving**, if points from the parametric space, S, are mapped such that if

$$\mathbf{x}^{n+1} = A\mathbf{x}^n + b, \tag{A.6}$$

then if $\mathbf{x}^n = \psi^n(\mathbf{r})$,

$$\mathbf{x}^{n+1} = \psi^{n+1}(\mathbf{r}) = A\psi^n(\mathbf{r}) + b = A\mathbf{x}^n + \mathbf{b} \tag{A.7}$$

Equivalently,

$$\psi^{n+1} = A\psi^n + \mathbf{b} \tag{A.8}$$

The bilinear parameterization of quads satisfies this property: the two orthogonal coordinates, $(r, s) \in [0, 1]^2$ linearly interpolate the vertices (see Figure A.1 for node numbering)

$$\psi^n(r, s) = (1 - r)\left[(1 - s)\mathbf{x}_0^n + s\mathbf{x}_3^n\right] + r\left[(1 - s)\mathbf{x}_1^n + s\mathbf{x}_2^n\right] \tag{A.9}$$

Clearly, $\psi^{n+1} = A\psi^n + \mathbf{b}$ as $\mathbf{x_j}^{n+1} = A\mathbf{x_j}^n + \mathbf{b}$ for $j = 0, \ldots, 3$.

The barycentric coordinates of polygon with vertices $\{\mathbf{v}_i\}$ also satisfies the linearity preserving property. To demonstrate this, barycentric coordinates satisfy the properties [30],

$$\mathbf{x} = \sum_i \lambda_i \mathbf{v}_i = \psi^n(\lambda), \tag{A.10}$$

$$\sum_i \lambda_i = 1, \tag{A.11}$$

$$\lambda_i \geq 0. \tag{A.12}$$

If $\mathbf{x}$ has barycentric coordinates λ, then if $\mathbf{x}^n = \psi^n(\lambda)$,

$$A\mathbf{x} = \sum_i \lambda_i A\mathbf{v}_i \tag{A.13}$$

$$A\psi^n(\lambda) + \mathbf{b} = \sum_i \lambda_i A\mathbf{v}_i + \mathbf{b}\sum_i \lambda_i \tag{A.14}$$

$$A\psi^n(\lambda) + \mathbf{b} = \sum_i \lambda_i (A\mathbf{v}_i + \mathbf{b}) = \psi^{n+1}(\lambda) \tag{A.15}$$

where $\sum_i \lambda_i = 1$ was utilized in the second step.

If the family of transformations satisfy the linearity preserving property, then we may analyze the accuracy of the constant parametric coordinate method. If the parameterization family, $\{\psi^n\}$, is linearity preserving, then updating the location of a material centroid by assuming its parametric coordinates are unchanged is exact for linear motions, since for an arbitrary subdomain mapped with an affine transformation,

$$\mathbf{x}_c(\Omega^{n+1}) = A\mathbf{x}_c(\Omega^n) + b \tag{A.16}$$

If the transformation is linearity preserving, then

$$\mathbf{x}_c(\Omega^{n+1}) = \varphi^{n+1}(\mathbf{r}) = A\varphi^n(\mathbf{r}) + b = A\mathbf{x}_c(\Omega^n) + b \tag{A.17}$$

In general, the Lagrangian motion will not be linear. However, as was shown in the previous section, for sufficient regularity in time an affine approximation to the Lagrangian motion is second order accurate.

Acknowledgments

This work was performed under the auspices of the National Nuclear Security Administration of the US Department of Energy at Los Alamos National Laboratory under Contract No. DE-AC52-06NA25396 and supported by the DOE Advanced Simulation and Computing (ASC) program. The authors acknowledge the partial support of the DOE Office of Science ASCR Program.

References

[1] Benson DJ. Computational methods in Lagrangian and Eulerian hydrocodes. *Computer Methods in Applied Mechanics and Engineering* 1992; **99**: 235-394.

[2] Benson DJ. Volume of fluid interface reconstruction methods for multi-material problems. *Applied Mechanics Review* 2002; **55**(2): 151-165.

[3] Rider WJ, Kothe DB. Reconstructing volume tracking. *Journal of Computational Physics* 1998; **141**: 112-152.

[4] Scardovelli R, Zaleski S. Direct numerical simulation of free-surface and interfacial flow. *Annual Review of Fluid Mechanics* 1999; **31**: 567-603.

[5] Hirt CW, Nichols BD. Volume of fluid (VOF) method for the dynamics of free boundaries. *Journal of Computational Physics* 1981; **39**: 201-225.

[6] Dyadechko V, Shashkov MJ. Reconstruction of multi-material interfaces from moment data. *Journal of Computational Physics*, 2008; **11**: 5361-5384.

[7] Noh WF, Woodward P. SLIC (simple line interface calculation). In van der Vooren AI and Zandbergen PJ, editors, *5th International Conference on Numerical Methods in Fluid Dynamics*, pages 330-340. Springer-Verlag, 1976.

[8] Youngs DL. Time dependent multi-material flow with large fluid distortion. In K. W. Morton and M. J. Baines, editors, *Numerical Methods for Fluid Dynamics*, pages 273-285. Academic Press, 1982.

[9] Youngs DL. An interface tracking method for a 3D Eulerian hydrodynamics code. Technical Report AWE/44/92/35, AWRE Design and Math Division, 1984.

[10] Pilliod JE Jr., Puckett EG. Second-order accurate volume-of-fluid algorithms for tracking material interfaces. *Journal of Computational Physics* 2004; **199**: 465-502.

[11] Swartz BK. The second-order sharpening of blurred smooth borders. *Mathematics of Computation* 1989; **52**(186): 675-714.

[12] Steinhaus H. A note on the ham sandwich theorem *Mathesis Polska*, 1938; **9**: 26-28.

[13] Stone AH, Tukey JW. Generalized “sandwich” theorems *Duke Mathematical Journal*, 1942; **9**: 356-359.

[14] Mosso S, Clancy S. A geometrically derived priority system for Young’s interface reconstruction. Technical Report LA-CP-95-0081, Los Alamos National Laboratory, Los Alamos, NM, 1995.

[15] Garimella RV, Dyadechko V, Swartz BK, Shaskov MJ. Interface reconstruction in multi-fluid, multi-phase flow simulations. In *Proceedings of the 14th International Meshing Roundtable*, pages 19-32, San Diego, CA, September 2005. Springer.

[16] Schofield SP, Garimella RV, Francois MM, Loubere R. Material order-independent interface reconstruction using power diagrams *International Journal for Numerical Methods in Fluids*, 2007; **56**(6): 643-659.

[17] Ahn HT, Shashkov M. Multi-material interface reconstruction on generalized polyhedral meshes. *Journal of Computational Physics* 2007; **226**: 2096-2132.

[18] Schofield SP, Garimella RV, Francois MM, Loubere R. A second-order accurate material-order-independent interface reconstruction technique for multi-material flow simulations *Journal of Computational Physics*, 2009; **228**(3): 731-745.

[19] Benson DJ. Eulerian finite element methods for the micromechanics of heterogeneous materials: Dynamic prioritization of material interfaces. *Computer Methods in Applied Mechanics and Engineering*, 1998, **151**: 343-360.

[20] Aurenhammer F. Power diagrams: properties, algorithms and applications. *SIAM J. Computing* 1987; **16**(1): 78-96.

[21] Imai H, Iri M, Murota K. Voronoi diagram in the Laguerre geometry and its applications. *SIAM J. Computing* 1985; **14**(1): 93-105.

[22] Caramana EJ, Burton DE, Shashkov MJ, Whalen PP. The construction of compatible hydrodynamics algorithms utilizing conservation of total energy. *Journal of Computational Physics* 1998; **146**: 227-262.

[23] Caramana EJ, Shashkov MJ, Whalen PP. Formulations of artificial viscosity for muti-dimensional shock wave computations. *Journal of Computational Physics* 1998; **144**: 70-97.

[24] Scovazzi G, Love E, Shashkov M. Multi-scale Lagrangian shock hydrodynamics on Q1/P0 finite elements: Theoretical framework and two-dimensional computations. *Comput. Methods Appl. Mech. Engrg.* 2008; **197**: 1056-1079.

[25] Shashkov M. Closure models for multidimensional cells in arbitrary Lagrangian-Eulerian hydrocodes. *Int. J. Numer. Meth. Fluids* 2008; **56**: 1497-1504.

[26] Winslow AM. Equipotential zoning of two-dimensional meshes. Technical Report UCRL-7312, Lawrence Livermore National Laboratory, 1963.

[27] Margolin LG, Shashkov M. Second-order sign-preserving conservative interpolation (remapping) on general grids. *Journal of Computational Physics* 2003; **184**: 266-298.

[28] Pember RB, Anderson RW. A comparison of staggered-mesh Lagrange plus remap and cell-centered direct Eulerian Godunov schemes for Eulerian shock hydrodynamics. Technical Report UCRL-JC-139820, Lawrence Livermore National Laboratory, 2000.

[29] Maire P-H. Personal Communication. Los Alamos National Laboratory, 2007.

[30] J. Warren, S. Schaefer, A. N. Hirani, and M. Desbrun. Barycentric coordinates for convex sets. *Advances in Computational Mathematics*, 2007; **27**(3): 319-338.