JOURNAL OF COMPUTATIONAL PHYSICS **135,** 198–202 (1997)
ARTICLE NO. CP975727

# Introduction to "An Arbitrary Lagrangian–Eulerian Computing Method for All Flow Speeds"

## 1. INTRODUCTION

The initial description of the ALE methodology by Hirt *et al.* [1] is a remarkable paper in the sense that it describes a generalized framework for fluid simulation consisting of many diverse techniques, even though many of the component pieces were not yet available. The ALE methodology combines the best features of Lagrangian and Eulerian representations, resulting in a flexible and robust solution algorithm. As a result, the ALE method has gained popularity for transient, high speed, large deformation problems of solid mechanics. For a review, see [2]. Nevertheless, ALE has not achieved comparable popularity in other fields. One explanation for this might be that the combination of so many diverse techniques requires a broad range of expertise not available to most researchers. In this brief introduction, I will summarize these disparate elements/issues as assembled by Hirt, Amsden, and Cook, and cite more recent references that provide effective solutions. I will finish by describing some as yet unresolved issues.

## 2. THE ALE METHODOLOGY

Prior to the 1974 paper of Hirt *et al.,* the computational fluid dynamicist was limited to Lagrangian or Eulerian methods. In Lagrangian simulations, the mesh moves with the local fluid velocity. Lagrangian methods maintain good resolution during large scale compressions/expansions and are well-suited to maintaining multimaterial interfaces. A particular advantage of the Lagrangian representation is that it is the only frame in which the advective terms vanish identically. On the other hand, multidimensional Lagrangian meshes tend to tangle and in general cannot represent large deformations due to shear and vorticity. In Eulerian simulations, the mesh is fixed in space and the fluid moves from one cell to another. Eulerian meshes are not subject to tangling, but the solutions are diffusive and it is difficult to maintain sharp material interfaces.

The Lagrangian and Eulerian representations are only two special cases of mesh motion; the essence of the ALE idea is that the mesh motion can be chosen arbitrarily, providing additional flexibility and accuracy. The three phases of an ALE simulation, as described by Hirt *et al.,* are an explicit Lagrangian update, an implicit iteration of the momentum equation with the equation of state, and a rezone/map phase. The Lagrangian phase provides an explicit update of the equations of motion. When the material velocities are much smaller than the fluid sound speed, the optional implicit phase allows sound waves to move many cells per cycle, thereby significantly improving computational efficiency. Finally, a rezone algorithm may prescribe mesh velocities relative to the fluid, thus necessitating a remap phase in which the solution from the end of phase two is mapped onto the new mesh. A particularly important property of ALE is that it provides a means of minimizing advection errors; this was emphasized by Brackbill and Pracht [3] in their construction of an almost-Lagrangian algorithm.

Hirt, Amsden, and Cook identify and discuss important aspects of each of the phases, some for the first time. For example, in the Lagrange phase the importance of conservation is emphasized, and a surface integral formulation of the equations is presented—an early example of what are now known as finite volume techniques. A short discussion of artificial viscosity alludes to the advantages of a full viscous stress tensor. The issue of null space modes is presented in the context of hourglassing patterns and an alternate node coupler is proposed.

The pressure iteration is described heuristically in terms of sound wave propagation. A simple point Newton–Raphson iteration with overrelaxation is proposed. This iteration is inefficient when compared with preconditioned conjugate gradient techniques that became widely known a few years later by Kershaw [4]. However, the reader should realize that Newton–Raphson is a matrix-free iterative technique and that computers of the early 1970s had relatively small memories. The remapping algorithms are similarly simple, being based on interpolated donor cell. Again, the reader should realize that the first nonoscillatory advection schemes were just being developed in the early 1970s [5].

One issue that is surprisingly discussed only briefly is that of strategies for rezoning. I believe the authors felt that the most effective rezoners were specific to the particular problem at hand. In general, the strategy will be to maintain the Lgarangian nature of calculation as much as possible. However, as the mesh moves, the truncation error associated with irregular spacing may begin to dominate the



0021-9991/97 $25.00

diffusive error of remapping. I will describe two approaches to quantifying these ideas in the next section.

In addition to the description of algorithms, Hirt *et al.* provide valuable detailed discussions of several practical matters. These include the use of cylindrical coordinates and an efficient programming method for using the same coding for both *X-Y* and *R-Z* simulations; the implementation of several different boundary conditions; and the necessary time step controls to ensure computational stability.

## 3. RECENT ALGORITHMIC ADVANCES

In this section I refer the reader to newer ideas for many of the component algorithms of an ALE code.

### a. *Phase* 1

Much effort has gone into improving the quality of the Lagrangian update, from the points of view of accuracy, preserving flow symmetries, and energy conservation. The fluid equations can be written in terms of a few spatial operators, such as gradient, divergence, and curl. It has been realized that the discrete representation of these operators should be chosen to preserve certain properties of analytic operators and to preserve relationships between the operators.

A particular example concerns the conservation of energy. Although Hirt *et al.* suggest using the total energy equation, it is preferable to use the internal energy equation for high speed (i.e., kinetic energy dominated) flows. The reason is that small errors in the kinetic energy, which is ultimately derived from the momentum equation, lead to large errors in the internal energy and, hence, to large errors in the pressure derived from the equation of state. By choosing the discrete divergence operator to be adjoint to the gradient operator, it is possible to use the internal energy equation and still ensure total energy conservation to machine accuracy. This fact was made use of by Sulsky and Brackbill [6]. A systematic approach to deriving discrete operators, termed support operator theory, has been developed by Shashkov and is presented in his book [7]. Of course, the discrete operators can only preserve a subset of the properties of the analytic operators. Whalen [8] provides an interesting discussion of the possibilities and trade-offs of different discretizations in cylindrical coordinate systems. For example, Whalen demonstrates the difficulty of simultaneously conserving total energy, preserving entropy, and maintaining symmetry in a spherical implosion.

The problems of null space modes has also received attention over the past 20 years, and many effective solutions have been put forth. Null space modes refer to patterns of velocity that distort the shape or size of a cell, but which do not produce restoring forces. Hourglassing and chevroning are two examples of such patterns. One method for controlling null space modes is filtering. Margolin and Pyun [9] construct an approximate local projection of the global hourglass mode and use it to filter the velocity field. A second approach, due to Dukowicz and Meltz [10] shows that at least part of the unphysical mesh distortion is due to spurious vorticity introduced by discretization errors. Dukowicz and Meltz propose correcting the vorticity and then reconstructing the velocity field. This algorithm requires solving a single Poisson equation. Still at third approach, termed temporary triangular subzoning or TTS [11], subdivides the computational cell into four triangles each with its own pressure. This subgrid structure allows the cell to sense null space patterns and to respond to them. Although the original TTS formulations were found to overly stiffen the mesh, there is now renewed interest in this approach.

Artificial viscosity has been the subject of continuous research since its introduction by von Neumann and Richtmyer in 1950. Wilkins [12] gives an excellent review of the formulation of a tensor viscosity, with special attention paid to the choice of length scale in the viscous coefficient. Noh [13] discusses more subtle problems associated with lack of steadiness of the shock front on a nonuniform mesh. Noh also discusses wall-heating effects and introduces the idea of artificial heat conduction. Several of the test problems introduced by Noh have become standards of the community. Dukowicz [14] offers an insightful discussion of the relation of artificial viscosity in finite difference/finite volume schemes to the Riemann solution in low order Godunov schemes. Christiansen [15] has extended these ideas to derive a ''flux-limited'' viscosity from consideration of higher order Godunov schemes. Flux-limited viscosities enable finite difference/finite volume schemes to capture shocks in only two computational cells. At present, these schemes are implemented in multidimensional codes by direction splitting.

### b. *Phase* 2

The computational efficiency of iterative solvers for matrix equations has increased dramatically over the past 20 years, mainly due the use of Krylov space methods combined with powerful preconditioners. An early example of these technique implemented on an irregular Lagrangian mesh (albeit for diffusion rather than hydrodynamics) is given by Kershaw [4]. A physics-based derivation of a Krylov space method, with applications to incompressible flows, is found in Smolarkiewicz and Margolin [16]. Here the iterative solution is found as the steady state of a second-order (or higher) wave equation (see also Ramshaw and Mesina [17]). An interesting application of discretization based on support operator theory to the formulation of the elliptic operator is given by Shashkov and Steinberg [18]. An advantage of this formulation

is that discrete approximations to self-adjoint positive-definite operators are symmetric and positive definite. This means that conjugate gradient methods can be applied, yielding efficient iterative solutions.

c. *Phase* 3

Although effective strategies can be found for particular problems, the key to a successful ALE code is a general rezoning algorithm that does not require user intervention. The landmark paper by Brackbill and Saltzman [19] approaches this issue in terms of the computational mesh. Brackbill and Saltzman formulate a rezoner based on a variational principle to optimize the smoothness, orthogonality, and other desirable properties of the mesh. Each property is quantified as a global sum of quantities determined by the local mesh geometry; the user can determine the relative importance of these properties by prescribing weights for each of the sums. The variational principle leads to an elliptic problem; usually only a few iterations of this problem are required on each computational cycle—that is, the rezone velocities need only relax toward the optimal mesh.

An alternate approach by Dukowicz [20] is derived from the technique of moving finite elements (MFE) [21]. MFE is an extension of the usual finite element approach in which the mesh coordinates themselves are considered as independent variables whose values can be varied to optimize the accuracy of the solution. Dukowicz derives an approximate splitting of the total MFE operator (which advances both the fluid solution and the mesh) and isolates the part of the operator that moves the mesh. This approach optimizes the rezoning in the metric of the equations, i.e., by minimizing the solution error.

When the rezone velocities are sufficiently small, the remapping is local in the sense that the cell mass is gained or lost only from neighboring cells. The remapping is then described by an advection equation. The production of nonoscillatory schemes for one-dimensional advection has exploded over the past 20 years, beginning with the FCT schemes of Boris and Book [5] and the early work of van Leer [22]. Some recent algorithms of interest include the FCT formulation of Zalesak [23], and the general class of total variation diminishing (TVD) schemes (see Sweby [24], for a review).

One issue with these techniques is that they are inherently one-dimensional and are applied in multidimensional codes by direction splitting (see also Leveque [25]). This splitting can lead to noticeable errors on a regular Eulerian mesh and may be even more serious on an irregular Lagrangian mesh. An additional issue is that of corner coupling. When the rezone velocity of a cell vertex points into a diagonal neighbor cell (that is, a cell that shares a vertex but not a full side), a second-order error results, reducing the overall accuracy of the advection to the first order.

An alternate approach to multidimensional advection is the MPDATA (multidimensional positive definite advection transport algorithm) of Smolarkiewicz [26]. MPDATA is not based on flux-limiting, but rather on the properties of upwind differencing. MPDATA limits the advective velocities as opposed to the flux components, and so it is fully multidimensional. MPDATA is also a multipass algorithm and accurately represents corner coupling. There is no requirement that the rezone/remap phase must be part of every computational cycle, and indeed, large improvements in computational efficiency can be realized when phase 3 is invoked less frequently. In this case, there is no guarantee that the rezone velocities are small, in which case a nonlocal remapper is required. The general rezoning algorithm of Dukowicz and Kodis [27] provides an efficient mechanism for finding the overlay of two quadrilateral, but otherwise arbitrary, meshes. This technique has been extended to hexahedral meshes in three dimensions [28].

## 4. UNRESOLVED ISSUES

a. *ALE vs SALE*

A major limitation of the algorithm described by Hirt, Amsden, and Cook is that there can be no rezoning across material interfaces. A powerful extension to these simple ALE schemes (SALE) is to incorporate a volume of fluid capability (VOF; see Hirt and Nichols [29]) into the Lagrangian cells. VOF technques were originally developed to introduce a Lagrangian tracking capability for material interfaces into Eulerian codes. In fact, VOF methods do not track interfaces, but rather, they reconstruct them on every cycle based on local values of the VOF variable. VOF methods require a generalization of the data structures of code—some cells will have more than one energy, density, and perhaps other thermodynamic variables. Furthermore, the number of materials in a cell will vary over the course of a calculation.

The chief difficulty in implementing VOF in an ALE code does not center on the irregularity of the Lagrangian mesh, but rather in the compressibility of the fluid. VOF techniques have rigorous basis only for incompressible flows. In particular, during the Lagrangian phase update of a mixed cell, how should one apportion the total change of volume and the total work between the two materials? The simple assumption that the VOF variable does not change is certainly inadequate when the two materials have much different compressibilities—consider the case of a mixed cell containing a solid and a gas. Assumptions of pressure and/or temperature equilibrium are similarly inadequate during the passage of a shock wave. The evolution of the VOF variable during the Lagrange phase of a compressible flow must still be considered an unsolved problem.

b. *Remapping on the Staggered Mesh*

Another open issue concerns the remapping of momentum on the staggered mesh. Hirt *et al.* chose to remap momentum on the dual mesh (that is, the mesh formed of momentum control volumes). In addition to the extra computation required by this process, there are inconsistencies associated with the fact that vertex masses are not remapped, but rather, they are reconstructed from the cell masses. A consistent treatment of the vertex masses is essential, since it is momentum that is advected.

Various schemes for momentum advection on a staggered mesh are reviewed and compared by [30]. Some of these are based on a cell-centered advection, in which vertex quantities are averaged to the cell centers, advected, and the net change then is redistributed to the vertices. No one scheme is to be preferred overall, either in terms of accuracy or computational efficiency.

c. *Eulerian vs Lagrange plus Remap*

One option of an ALE calculation is a total rezone of the grid; that is, the rezone velocities are chosen as the negative of the fluid velocities. This leads to an Eulerian simulation that is in principle equivalent to a fully Eulerian code. In practice, however, many researchers have found that the Lagrange plus remap simulation is more accurate than the Eulerian simulation. A hint as to why the Lagrange plus remap solution is better may be found in a paper by Smolarkiewicz and Margolin [31], where the use of a Lax–Wendroff compensation for temporal truncation error is shown to contain a term proportional to the divergence of the forcing terms that appear on the right-hand side of the equations. In the momentum equation, this forcing term would include the pressure gradient. In most Eulerian codes, this extra term is ignored, leading to a second-order error. This term does not appear in phase 3 of an ALE code, since the splitting between the Lagrange and the remap phases is exact and there are no forcing terms in the advection equations.

## 5. SUMMARY

The ALE methodology as originally formulated by Hirt, Amsden, and Cook leads to a powerful and flexible solution procedure. The accuracy and efficiency of an ALE code depends on assembling a diverse set of numerical algorithms. The original paper is a remarkable effort given the breadth of the numerical methods required and the lack of prior research in several key areas. At the present time, there are still unresolved issues to be addressed.

## REFERENCES


1. C. W. Hirt, A. A. Amsden, and J. L. Cook, An arbitrary Lagrangian–Eulerian computing method for all flow speeds, *J. Comput. Phys.* **14,** 227 (1974).
2. D. J. Benson, Computational methods in Lagrangian and Eulerian hydrocodes, *Comput. Methods Appl. Mech. Eng.* **99,** 235 (1992).
3. J. U. Brackbill and W. E. Pracht, An implicit, almost-Lagrangian algorithm for magnetohydrodynamics, *J. Comput. Phys.* **13,** 455 (1973).
4. D. S. Kershaw, The incomplete Cholesky–conjugate gradient method for the iterative solution of systems of linear equations, *J. Comput. Phys.* **26,** 43 (1978).
5. J. P. Boris and D. L. Book, Flux-corrected transport. I. SHASTA, A fluid transport algorithm that works, *J. Comput. Phys.* **11,** 38 (1973).
6. D. Sulsky and J. U. Brackbill, A numerical method for suspension flow, *J. Comput. Phys.* **96,** 339 (1991).
7. M. Shashkov, ''Conservative Finite-Difference Methods on General Grids,'' CRC Press, New York, 1996.
8. P. P. Whalen, Algebraic limitations on two-dimensional hydrodynamics simulations, *J. Comput. Phys.* **124,** 46 (1995).
9. L. G. Margolin and J. J. Pyun, A method for treating hourglass patterns, *in Proceedings, Numerical Methods in Laminar and Turbulent Flows* edited by C. Taylor and M. M. Hafey (Pineridege, Swansea, 1987), p. 149.
10. J. K. Dukowicz and B. J. A. Meltz, Vorticity errors in multidimensional Lagrangian codes, *J. Comput. Phys.* **99,** 115 (1992).
11. K. B. Wallick, *Rezone: A Method for Automatic Rezoning in Two-Dimensional Lagrangian Hydrodynamic Problems,* Los Alamos National Laboratory Report LA-10829-MS, 1987.
12. M. L. Wilkins, Use of artificial viscosity in multidimensional fluid dynamic calculations, *J. Comput. Phys.* **36,** 281 (1980).
13. W. F. Noh, Errors for calculations of strong shocks using an artificial viscosity and an artificial heat flux, *J. Comput. Phys.* **72,** 78 (1987).
14. J. K. Dukowicz, A general non-iterative Riemann solver for Godunov's method, *J. Comput. Phys.* **61,** 119 (1985).
15. R. B. Christiansen, *Godunov Methods on a Staggered Mesh—An Improved Artificial Viscosity,* UCRL—JC105269, 1991.
16. P. K. Smolarkiewicz and L. G. Margolin, Variational solver for elliptic problems in atmospheric flows, *Appl. Math Comput. Sci.* **4,** 527 (1994).
17. J. D. Ramshaw and G. L. Mesina, A hybrid pseudocompressibility method for transient incompressible fluid flow, *Comput. & Fluids* **20,** 165 (1991).
18. M. Shashkov and S. Steinberg, Support-operator finite-difference algorithms for general elliptic problems, *J. Comput. Phys.* **118,** 131 (1995).
19. J. U. Brackbill and J. S. Saltzman, Adaptive zoning for singular problems in two dimensions, *J. Comput. Phys.* **46,** 342 (1982).
20. J. K. Dukowicz, A simplified adaptive mesh technique derived from the moving finite element method, *J. Comput. Phys.* **56,** 324 (1984).
21. R. J. Gelinas, S. K. Doss, and K. Miller, The moving finite element method: Applications to partial differential equations with multiple large gradients, *J. Comput. Phys.* **40,** 202 (1981).
22. B. van Leer, Towards the ultimate conservative difference scheme. IV. A new approach to numerical convection, *J. Comput. Phys.* **23,** 276 (1977).
23. S. J. Zalesak, Fully multidimensional flux-corrected transport algorithms for fluids, *J. Comput. Phys.* **31,** 335 (1979).
24. P. K. Sweby, High-resolution schemes using flux limiters for hyperbolic conservation law, *SIAM J. Numer. Anal.* **21,** 995 (1984).
25. R. Leveque, High-resolution conservative algorithms for advection in incompressible flow, *SIAM J. Numer. Anal.* **33,** 627 (1996).
26. P. K. Smolarkiewicz, A fully multidimensional positive-definite ad-

vection scheme with small implicit diffusion, *J. Comput. Phys.* **54,** 325 (1984).

27. J. K. Dukowicz and J. W. Kodis, Accurate conservative remapping (rezoning) for arbitrary Lagrangian–Eulerian computations, *SIAM J. Sci. Stat. Comput.* **8,** 305 (1987).
28. J. K. Dukowicz and N. T. Padial, *REMAP3D: A Conservative Three-Dimensional Remapping Code,* Los Alamos National Laboratory Report LA-12136-MS, August 1991.
29. C. W. Hirt and B. D. Nichols, Volume of fluid (VOF) method for the dynamics of free boundaries, *J. Comput. Phys.* **39,** 201 (1981).
30. D. J. Benson, Momentum advection on a staggered mesh, *J. Comput. Phys.* **100,** 143 (1992).
31. P. K. Smolarkiewicz and L. G. Margolin, On forward-in-time differencing for fluids, *Mon. Wea. Rev.* **121,** 1847 (1993).

L. G. Margolin

*Los Alamos National Laboratory*
*Los Alamos, New Mexico 87545*