Conservative Remapping and ALE Methods for Plasma Physics

Milan Kuchařík, Richard Liska and Mikhail Shashkov

Abstract. Arbitrary Lagrangian-Eulerian (ALE) method overcomes troubles with Lagrangian moving mesh distortion by mesh smoothing and conservative remapping from the old mesh to the smoother one. Conservative, bound-preserving remapping of conserved quantities by swept area method is presented. 2D ALE code on quadrilateral, logically orthogonal meshes has been developed for laser plasma physics applications. A particular example of high velocity disc impacting solid target cannot be simulated by pure Lagrangian method due to mesh distortion while the ALE code produces reasonable results.

1. Introduction

Many problems in compressible fluid dynamics modeled by Euler equations involve moving boundaries. Treatment of moving boundaries on static Eulerian computational mesh is rather complicated and in some cases as modeling of large compression or expansion almost impossible. For such problems one usually use Lagrangian coordinates moving with the fluid which naturally treat moving boundaries and are able to deal with large scale changes of computational domain as large compression or expansion. Typical examples of problems with moving boundaries come from laser plasma physics. The interaction of very high intensity laser beams with matter produces laser plasma which can be treated as compressible fluid and modeled by Euler equations. Simulated problems involve target compression and corona expansion where the volume of computational domain changes multiple times during the computation and simulations are typically performed in Lagrangian coordinates.

In the Eulerian methods using static coordinate frame, the fluid flows through the static computational mesh. On the other hand, in the Lagrangian methods, the mesh moves with the fluid, there is no mass flux between computational cells, the advective parts of fluxes are included through the movement of the mesh, so that e.g. in momentum equation the pressure gradient remains as the only flux. Lagrangian methods are suitable for problems involving large changes of volume.

They treat naturally moving boundaries. Due to no mass flux between computational cells they treat very well multi-material problems and contact discontinuities appearing not only on material interfaces. For some problems, as modeling shear or physical instabilities, however Lagrangian methods suffer from deformations (see Fig. 1(b) for example of such distorted mesh) of the moving mesh following the fluid movement. The mesh might distort and degenerate so much that some computational cells become inverted when the node of the cell crosses an opposite edge of the cell. At such point, when the computational grid tangles and loses its regularity, the Lagrangian computation cannot continue as the assumptions of the method (regularity of the grid) are violated and the method fails. A way, how to deal with such failures, is to use the Arbitrary Lagrangian-Eulerian (ALE) method.

The ALE method consists of several Lagrangian computational time steps followed by a mesh rezoning and a conservative quantities remapping. The mesh rezoning step smooths the Lagrangian computational mesh and avoids its distortion. During the remapping step the conservative quantities are conservatively remapped from the old Lagrangian mesh to the new smoothed one. After remapping the Lagrangian computation continues until the next rezone/remap steps which introduce the Eulerian flavor into the method allowing mass flux between computational cells. The rezone/remap steps keep the quality of the moving mesh good enough during the whole computation and are performed either regularly after fixed amount of Lagrangian time steps or when mesh quality deteriorates under some threshold. Although the ALE method was first proposed in 1974 [1] and became popular for transient, large deformation problems in solid mechanics [2], its usage in compressible fluid simulations attracted attention only after Margolin [3] introduced the reprint of [1] in the 30 years anniversary issue of the *Journal of Computational Physics* in 1997. Recently the ALE method is becoming more and more popular even for compressible fluid dynamics [4], [5].

The ALE method consists of repeating three parts: 1. Lagrangian computation, 2. rezone mesh smoothing, 3. conservative remapping. We briefly outline the methods we use in the first two parts, present the third one in more detail, describe the developed 2D ALE code and one its particular application to laser plasma physics.

2. Lagrangian Solver

The Lagrangian solver is the essential part of the ALE algorithm. It solves the fluid equations in the Lagrangian form

$$\begin{aligned} \frac{d\,\rho}{d\,t} + \rho \ \mathrm{div}\ \vec{u} &= 0 \\ \rho \frac{d\,\vec{u}}{d\,t} + \mathrm{grad}\ p &= 0 \\ \rho \frac{d\,\epsilon}{d\,t} + p \ \mathrm{div}\ \vec{u} &= 0, \end{aligned} \tag{1}$$

where $\vec{u} = d\,\vec{x}/d\,t$ is the velocity of the fluid particles, ρ is the fluid density, ϵ is the specific internal energy, which gives us the fluid pressure p through the equation of state $p = p(\rho, \epsilon)$. Note that the differential operator d/dt in (1) applied to quantities depending on space and time is the operator of total derivative, e.g. $d\rho/dt = \partial\rho/\partial t + \vec{u}\cdot\mathrm{grad}\,\rho$, the advective term $\vec{u}\cdot\mathrm{grad}\,\rho$ is included in the Lagrangian moving coordinates.

For the numerical treatment of (1) we employ the compatible Lagrangian hydrodynamics algorithm proposed in [6] and [7]. The scalar quantities, (fluid density, pressure and internal energy) are defined inside the grid cells. The vector quantities (nodal positions and velocities) are defined at the grid nodes. The node movement results from solving the ordinary differential equation $d\vec{x}/dt = \vec{u}$ at each node. Density is obtained from mesh movement and basic Lagrangian assumption of constant mass of each cell which trivially gives mass conservation. The compatible algorithm is based on the computation of the forces at each node from the surrounding cells. The velocity and internal energy update is controlled by these forces in a way which guarantees the conservation of momentum and total energy.

The nodal forces consist of three main parts affecting the nodal movement. The zonal pressure force represents the force from the surrounding cells to the node due to the pressure inside them. The subzonal pressure force protects grid from the unphysical, hourglass-type motion, and it depends on the difference of the pressure in the zone (cell) and the pressure in the subzone – the quadrilateral region between the cell center, the appropriate node and two midpoints of two appropriate cell edges. The artificial viscosity force adds viscosity pressure in the areas of fluid compression. Other forces as gravitation can be included into the method.

3. Mesh Rezoning

The mesh rezoning step of the ALE algorithm performs untangling and smoothing techniques to regularize and improve computational Lagrangian grids. To decrease the numerical error of the following remapping step the grid should be modified as little as possible.

Most often we apply the Winslow's smoothing [8] technique, which is an improved version of weighted averaging. One iteration of the node positions mesh smoothing is given by

$$\vec{x}_{i,j}^{\,k+1} = \frac{1}{2\,(\alpha^k + \gamma^k)} \Big(\alpha^k\,(\vec{x}_{i,j+1}^{\,k} + \vec{x}_{i,j-1}^{\,k}) + \gamma^k\,(\vec{x}_{i+1,j}^{\,k} + \vec{x}_{i-1,j}^{\,k}) - \frac{1}{2}\,\beta^k\,(\vec{x}_{i+1,j+1}^{\,k} - \vec{x}_{i-1,j+1}^{\,k} + \vec{x}_{i-1,j-1}^{\,k} - \vec{x}_{i+1,j-1}^{\,k}) \Big), \tag{2}$$

where the coefficients $\alpha^k = x_\xi^2 + y_\xi^2$, $\beta^k = x_\xi\,x_\eta + y_\xi\,y_\eta$, $\gamma^k = x_\eta^2 + y_\eta^2$, and where (ξ, η) are logical coordinates $\xi_i = i/M$, $\eta_j = j/N$ for $i = 0, \ldots, M$ and $j = 0, \ldots, N$. Derivatives with respect to ξ and η are approximated by the central differences $(x_\xi)_{i,j} \approx (x_{i+1,j} - x_{i-1,j})/2\,\Delta\xi, (x_\eta)_{i,j} \approx (x_{i,j+1} - x_{i,j-1})/2\,\Delta\eta$ and analogically for y.

Other available rezoning methods include reference Jacobian optimization-based strategies [4] or untangling by the combination of the feasible set approach and the numerical optimization [9].

4. Conservative Remapping

The last part of the ALE algorithm is the remapping method for conservative interpolation of the conservative quantities to the new, smoothed grid. The remapping of all conservative variables on the staggered grid is quite a complex problem discussed in [10]. We focus here just to remapping of one quantity, suppose the fluid density.

We know the density values ρ_C inside the grid cells C, which represent the mean values of some (unknown) underlying function g. The output of the remapping process must be the mean values $g_{\tilde{C}}$ of the function g in the cells $\tilde{C}$ of the new grid corresponding to the new densities. To preserve second order of accuracy of the complete ALE algorithm we want our remapping process to be linearity-preserving. Also, we do not want to create new local extremes, the algorithm should protect local bounds. Total mass must be the same, our method has to be conservative.

The most natural way to satisfy these conditions is based on the piecewise-linear reconstruction of the underlying function followed by its integration over the overlapped regions of the original and the new cells, which gives us the new mean values. The problem is, that for determining the overlapped regions we have to compute all the intersections of the original and the new cells, which is very expensive to compute, and thus rather slow. Our method [11] is based on the approximate integration and does not require finding these intersections. On the other hand, the local-bound preservation can be violated, so one more stage is needed – the repair, which corrects the values to the original local bounds.

The first stage – piecewise-linear reconstruction – remains the same for both methods. It computes slopes of the underlying function in each cell from the surrounding mean values. We use the method of integrating the function derivatives over each cell, which corresponds to computing the average slopes in the cells. From the Green theorem, we reduce the integrals of the function derivatives over cells to the boundary integrals of the original function, which can be computed exactly. After dividing by the cell volume, we get the unlimited average slopes $(\partial g/\partial x)_C^{\mathrm{unlim}}$. Now, we apply the Barth-Jasperson limiter [12] ϕ_C to limit the values of the slopes $(\partial g/\partial x)_C^{\mathrm{BJ}} = \phi_C\,(\partial g/\partial x)_C^{\mathrm{unlim}}$, such that in each cell C with center (x_C, y_C), the underlying function

$$g_C^{\mathrm{reconstr}} = g_C + (\partial g/\partial x)_C^{\mathrm{BJ}}\,(x - x_C) + (\partial g/\partial y)_C^{\mathrm{BJ}}\,(y - y_C) \tag{3}$$

has all values within its local extremes in the neighboring cells.

The second stage – the integration – is different. In the natural "exact integration" algorithm, the new mean values are obtained by direct evaluation of the integrals of the underlying function over the overlapped areas of the original and the new grid. Our approximate integration is based on computation of the masses

of the swept regions. Every quadrilateral cell C has four swept regions F_C^i defined by the smooth movement of each edge to its new position. By integration of the reconstructed underlying function over these swept regions we compute the swept masses $\mathcal{F}_C^i$ (in signed sense – swept mass is positive, if the the edge moves outward from the cell, and it is negative, if it moves inwards)

$$\mathcal{F}_C^i = \int_{F_C^i} g^{\text{reconstr}}(x, y)\, dV\,, \tag{4}$$

and the reconstruction is taken from the cell, in which more of the swept region lies. The new mass of the cell is than defined as the original mass plus all swept masses

$$m_{\tilde{C}} = m_C + \sum_i \mathcal{F}_C^i \tag{5}$$

and after dividing by the cell volume we get the new densities $g_{\tilde{C}}$.

The approximate integration method is much faster than the exact integration, but it may produce new mean values out of the local bounds. Thus, we developed the third stage – repair – to correct this problem. The repair is a conservative redistribution of the conservative quantity. In each cell we check, whether the mean value $g_{\tilde{C}}$ lies within the local bounds or not. If so, we do not do anything, but otherwise we perform the repair process. Suppose, that the local minimum is violated $g_{\tilde{C}} < g_C^{\min}$. At first, we compute the mass which is needed to be added to the cell to bring the value back to the minimum. Then, we search for the mass, which can safely be taken from the neighboring cells without violating their local minimum. If we found enough mass, we perform the repair – we bring the "wrong" value to the minimum and take the mass from the neighboring cells proportionally to the mass available in each cell. If there is still not enough mass available, we extend the neighborhood and repeat the whole process. We have proved [11], that the whole repair process will successfully finish in a finite number of steps. The repair process with more natural bounds in density, velocity and internal energy (not in conserved variables as here) is more complicated [13].

The complete reconstruct-integrate-repair process satisfies all our mentioned conditions for the remapping method. Moreover, it is very efficient when compared to the natural remapping algorithm, and it is applicable to remapping of all conservative quantities between general grids both in 2D and 3D [14].

5. ALE Code for Plasma Simulation

We have developed a 2D ALE code on logically orthogonal quadrilateral meshes for laser plasma simulations. Laser plasma can be approximated as a compressible fluid however its equation of state describing relation between pressure, density, internal energy and temperature is far from the ideal gas one. The quotidian equation of state (QEOS) is used to approximate the plasma equation of state.

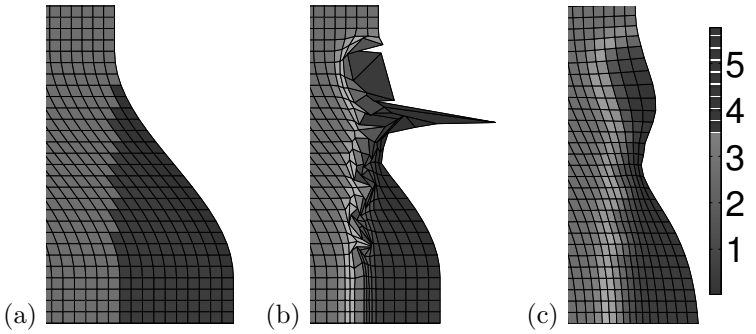

FIGURE 1. Density colormap [in g/cm^3] for the disc impact problem: initial conditions (a); pure Lagrangian (b) and complete ALE (c) results at time 0.2 ns.

Heat conductivity plays an important role in many problems from laser plasma physics and often cannot be neglected. It is modeled by a parabolic term in the energy equation. By splitting, we solve the parabolic part separately by the support operator method with mimetic discretization [15]. The implicit treatment of parabolic equation allows to use the same time step as in the hyperbolic part treated by ALE. Mimetic discretization results in a linear system with positive definite matrix which is solved by the conjugate gradient algorithm.

As an example, we present here the numerical simulation of the impact of the laser accelerated thin Aluminum disc into a massive Aluminum target. The parameters are taken from a part of one experiment performed at the PALS (Prague Laser Asterix System) facility [16], [17]. The thin (11 μm) Aluminum disc of diameter 300 μm is irradiated by the 390 J laser beam operating on the third harmonic. The disc is ablatively accelerated to a very high speed (150 km/s) against the massive target. Presented ALE simulation starts at the moment of the disc impact into the target with initial conditions shown in Fig. 1(a). The disc is on the right, the target is on the left and the disc is flying to the left towards the target. The disc is already heated by laser in plasma/gas phase state with lower density than solid state Aluminum density. Very early after the impact the computational mesh of the pure Lagrangian simulation degenerates considerably as shown in Fig. 1(b) at time 0.2 ns. The mesh distortion leads to failure of pure Lagrangian computation soon after this time. On the other hand, the complete ALE simulation continues without any problem, see Fig. 1(c) for ALE result at time 0.2 ns, the grid remains reasonable smooth during the whole simulation. This is the main reason for using and developing ALE methods – for some simulations, the ALE method can reach results, which cannot be obtained by pure Lagrangian methods. Pure Eulerian method would have in this case troubles how to treat corona expanding to the right at later stages.

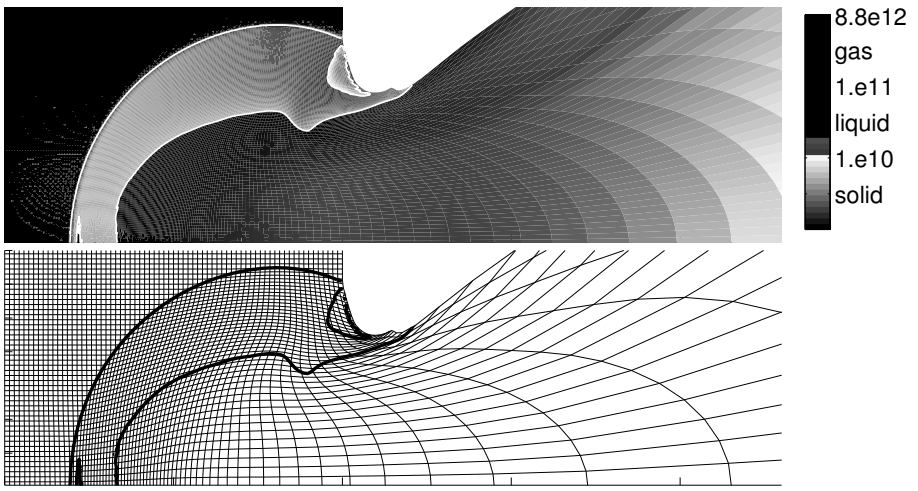

FIGURE 2. Internal energy increase [in erg] and coarsened computational mesh at time 40 ns with material phase interfaces.

After the impact the huge kinetic energy of the flying disc is transformed into heat energy which melts and evaporates Aluminum in the target. The disc starts to sink into the target and is reflected in the plasma/gas phase together with some target material creating a crater in the target. The reflected very low density corona expands multiple times to the right, a part of corona appears on the right of Fig. 2. Circular shape shock wave is formed in the target which coincides with the solid-liquid phase interface presented in Fig. 2. The shock wave penetrates into the target melting its material. The second phase interface shown in Fig. 2, the liquid-gas interface, moves to the left into the target until the time 40 ns at which the Fig. 2 is taken. Fig. 2 presents the internal energy increase colormap and coarsened mesh including isolines corresponding to energy needed to: (1) heat and melt for solid-liquid phase interface; (2) heat, melt and evaporate for the liquid-gas phase interface. Due to limited available resolution the displayed computational mesh is coarsened three times so that one plotted cell includes nine (3×3 patch) computational cells. Plotted area is 2.2 mm wide and 0.7 mm high while the corona expanding right and up is almost 6 mm wide and 2 mm high and expands fastfurther. More physics related details can be found in [18],

Acknowledgment: Partial support of the project No. 202/03/H162 of the Grant Agency of the Czech Republic, No. FRVS 2087/2004 of the Ministry of Education of the Czech Republic, No. CTU 0410914 of the Czech Technical University in Prague is acknowledged. The authors thank R. Garimella, P. Havlík, R. Loubère,

P. Váchal and B. Wendroff for fruitful discussions and constructive comments. Presentation at this conference was possible thanks to the Young Researchers' Grant of the HYP 2004 conference.

References

[1] C. W. Hirt, A. A. Amsden, and J. L. Cook. An arbitrary Lagrangian-Eulerian computing method for all flow speeds. *Journal of Computational Physics*, 14:227–253, 1974.

[2] D.J. Benson. Computational methods in Lagrangian and Eulerian hydrocodes. *Computer Methods in Applied Mechanics and Engineering*, 99:235–394, 1992.

[3] L. Margolin. Introduction to "an arbitrary Lagrangian-Eulerian computing method for all flow speeds". *J. Comp. Phys.*, 135:198–202, 1997.

[4] P. M. Knupp, L. G. Margolin, and M. J. Shashkov. Reference Jacobian optimization-based rezone strategies for arbitrary Lagrangian Eulerian methods. *Journal of Computational Physics*, 176(1):93–128, 2002.

[5] R. W. Anderson, N. S. Elliott, and R. B. Pember. An arbitrary Lagrangian-Eulerian method with adaptive mesh refinement for the solution of the Euler equations. *Journal of Computational Physics*, 199(2):598–617, 2004.

[6] E. J. Caramana, D. E. Burton, M. J. Shashkov, and P. P. Whalen. The construction of compatible hydrodynamics algorithms utilizing conservation of total energy. *Journal of Computational Physics*, 146(1):227–262, 1998.

[7] J. Campbell and M. Shashkov. A compatible Lagrangian hydrodynamics algorithm for unstructured grids. *Selcuk Journal of Applied Mathematics*, 4:53-70, 2003.

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

[9] P. Váchal, R. Garimella, and M. Shashkov. Untangling of 2D meshes in ALE simulations. *Journal of Computational Physics*, 196(2):627–644, 2004.

[10] R. Loubere and M. Shashkov. A subcell remapping method on staggered polygonal grids for arbitrary-Lagrangian-Eulerian methods. Technical Report LA-UR-04-6692, Los Alamos National Laboratory, 2004.

[11] M. Kuchařík, M. Shashkov, and B. Wendroff. An efficient linearity-and-bound-preserving remapping method. *Journal of Computational Physics*, 188(2):462–471, 2003.

[12] T. J. Barth. Numerical methods for gasdynamic systems on unstructured meshes. In C. Rohde D. Kroner, M. Ohlberger, editor, *An introduction to Recent Developments in Theory and Numerics for Conservation Laws*, pages 195–284, Berlin, 1997. Springer.

[13] Mikhail Shashkov and Burton Wendroff. The repair paradigm and application to conservation laws. *J. Comp. Phys.*, 198(1):265–277, 2004.

[14] R. Garimella, M. Kuchařík, and M. Shashkov. Efficient algorithm for local-bound-preserving remapping in ALE methods. In M. Feistauer et.al., editors, *Numerical Mathematics and Advanced Applications*, pages 358–367. Springer-Verlag Berlin Heidelberg New York, 2004.

[15] M. Shashkov and S. Steinberg. Solving diffusion equation with rough coefficients in rough grids. *J. Comp. Phys.*, 129:383–405, 1996.

[16] S. Borodziuk, et.al. Application of laser simulation method for the analysis of crater formation experiment on PALS laser. *Cz. J. Phys.*, 53(9):799–810, 2003.

[17] M. Kalal, et.al. High power laser interaction with single and double layer targets. In *Proceedings of XXVIII ECLIM*, 2004. To appear.

[18] M. Kuchařík, J. Limpouch, R. Liska, and P. Havlík. ALE simulations of laser interactions with flyer targets. In *Proceedings of XXVIII ECLIM*, 2004. To appear.

Czech Technical University in Prague, Faculty of Nuclear Sciences and Physical Engineering, Břehová 7, Praha 1, 115 19, Czech Republic
E-mail address: kucharik@karkulka.fjfi.cvut.cz,liska@siduri.fjfi.cvut.cz

Los Alamos National Laboratory, Group T-7, Los Alamos, NM 87544, USA
E-mail address: shashkov@lanl.gov