

Accepted Manuscript

Direct arbitrary-Lagrangian–Eulerian ADER-MOOD finite volume schemes for multidimensional hyperbolic conservation laws

Walter Boscheri, Raphaël Loubère, Michael Dumbser

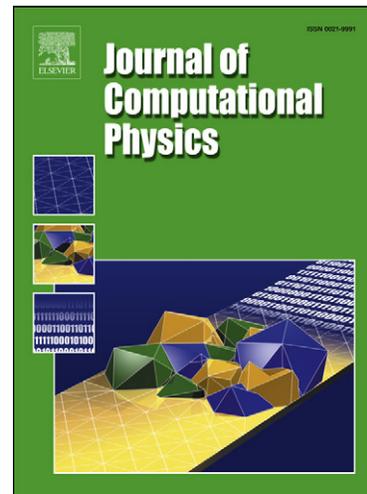
PII: S0021-9991(15)00151-5
DOI: <http://dx.doi.org/10.1016/j.jcp.2015.03.015>
Reference: YJCPH 5771

To appear in: *Journal of Computational Physics*

Received date: 4 November 2014
Revised date: 5 March 2015
Accepted date: 6 March 2015

Please cite this article in press as: W. Boscheri et al., Direct arbitrary-Lagrangian–Eulerian ADER-MOOD finite volume schemes for multidimensional hyperbolic conservation laws, *J. Comput. Phys.* (2015), <http://dx.doi.org/10.1016/j.jcp.2015.03.015>

This is a PDF file of an unedited manuscript that has been accepted for publication. As a service to our customers we are providing this early version of the manuscript. The manuscript will undergo copyediting, typesetting, and review of the resulting proof before it is published in its final form. Please note that during the production process errors may be discovered which could affect the content, and all legal disclaimers that apply to the journal pertain.



Direct Arbitrary-Lagrangian-Eulerian ADER-MOOD Finite Volume Schemes for Multidimensional Hyperbolic Conservation Laws

Walter Boscheri^a, Raphaël Loubère^b, Michael Dumbser^{*a}

^aLaboratory of Applied Mathematics, Department of Civil, Environmental and Mechanical Engineering, University of Trento, Via Mesiano, 77 - 38123 Trento, Italy.

^bCNRS and Institut de Mathématiques de Toulouse (IMT) Université Paul-Sabatier, Toulouse, France

Abstract

In this paper we present a new family of efficient high order accurate direct Arbitrary-Lagrangian-Eulerian (ALE) one-step ADER-MOOD finite volume schemes for the solution of nonlinear hyperbolic systems of conservation laws for moving unstructured triangular and tetrahedral meshes. This family is the next generation of the ALE ADER-WENO schemes presented in [16, 20]. Here, we use again an element-local space-time Galerkin finite element predictor method to achieve a high order accurate one-step time discretization, while the somewhat expensive WENO approach on moving meshes, used to obtain high order of accuracy in space, is replaced by an *a posteriori* MOOD loop which is shown to be less expensive but still as accurate. This *a posteriori* MOOD loop ensures the numerical solution in each cell at any discrete time level to fulfill a set of user-defined detection criteria. If a cell average does not satisfy the detection criteria, then the solution is locally re-computed by progressively decrementing the order of the polynomial reconstruction, following a so-called *cascade* of predefined schemes with decreasing approximation order. A so-called parachute scheme, typically a very robust first order Godunov-type finite volume method, is employed as a last resort for highly problematic cells. The cascade of schemes defines how the decrementing process is carried out, i.e. how many schemes are tried and which orders are adopted for the polynomial reconstructions. The cascade and the parachute scheme are choices of the user or the code developer. Consequently the iterative MOOD loop allows the numerical solution to maintain some interesting properties such as positivity, mesh validity, *etc.*, which are otherwise difficult to ensure. We have applied our new high order unstructured direct ALE ADER-MOOD schemes to the multi-dimensional Euler equations of compressible gas dynamics. A large set of test problems has been simulated and analyzed to assess the validity of our approach in terms of both accuracy and efficiency (CPU time and memory consumption).

Keywords: Arbitrary-Lagrangian-Eulerian, a posteriori limiter, MOOD paradigm, ADER schemes, moving unstructured triangular and tetrahedral meshes, high order of accuracy in space and time, high performance computing (HPC), hyperbolic conservation laws

1. Introduction

In the last decades a lot of research has been focused on the development of Lagrangian numerical schemes for solving nonlinear systems of hyperbolic conservation laws [94, 12, 93, 23, 112, 90, 26], because of the excellent properties in the resolution of material interfaces and contact waves typically achieved by Lagrangian algorithms. In the Lagrangian approach the nonlinear convective terms of the governing

*Corresponding author

Email addresses: walter.boscheri@unitn.it (Walter Boscheri), raphael.loubere@math.univ-toulouse.fr (Raphaël Loubère), michael.dumbser@unitn.it (Michael Dumbser*)

equations disappear, hence such schemes exhibit virtually no numerical dissipation across contact discontinuities.

Furthermore Lagrangian algorithms require a *moving mesh* framework, with the physical variables either located at the cell barycenter [33, 107, 90, 91, 89] or defined at different positions within each control volume [84, 85]. The first approach leads to *cell-centered* Lagrangian schemes, while the latter is usually addressed as the *staggered mesh* approach, where the velocity is defined at the cell vertices or interfaces and the other variables at the cell center. In [93] Munz proposed cell-centered Godunov-type schemes of the Roe and HLL type for the equations of Lagrangian hydrodynamics, while in [26, 36, 37] within a cell-centered Lagrangian context the evolution equations of the geometry are coupled with the equations of the flow field yielding a weakly hyperbolic system of conservation laws. Multi-dimensional unstructured meshes have also been considered by Maire [87, 92, 89, 88], who presented first and second order accurate cell-centered Lagrangian schemes. A node-centered solver is adopted to evaluate the time derivatives of the fluxes and it can be seen as a multi-dimensional extension of the Generalized Riemann Problem (GRP) methodology used, for example, in the ADER schemes of Titarev and Toro [119, 116, 117]. Curved meshes have been used in [33], where a cell-centered Lagrangian method which is translation invariant is proposed. Mesh motion usually leads to large element deformation or distortion, hence requiring a mesh quality optimization process during the simulation. This is achieved by employing remeshing or rezoning strategies. Within the framework of indirect cell-centered Arbitrary-Lagrangian-Eulerian (ALE) algorithms [107, 14, 78, 80, 77, 13], the mesh quality and the overall scheme robustness are improved by performing a purely Lagrangian phase with subsequent remeshing/rezoning and remapping. The ALE approach allows the mesh velocity to be chosen independently from the local fluid velocity, therefore the grid nodes can be arbitrarily moved. Such an approach is also suitable for solving multi-phase and multi-material flow problems [58, 127, 21, 108, 67, 98, 112]. All the Lagrangian schemes listed so far achieve at most second order of accuracy in space and time.

Remapping is no longer needed as a separate step in the so-called direct ALE schemes. Indeed the final mesh velocity is already taken into account in the numerical flux formulation. Such algorithms are widely used in the context of fluid-structure interaction problems (FSI), see for example the high order ALE discontinuous Galerkin (DG) schemes presented in [43, 57, 56, 27]. In both approaches, direct or indirect, the mesh velocity must be deduced taking into account competing desires: the gain in accuracy brought by maintaining the mesh as Lagrangian as possible, and, on the other hand, the development of instabilities due to a highly distorted mesh that results from a purely Lagrangian mesh motion. Consequently, the mesh velocity must be designed in such a way that it maintains a geometrically valid mesh having a good geometrical quality, but, conjointly, it should follow the main structures of the flow. Usually some sort of 'node solver' is designed to determine such an appropriate mesh velocity.

Discontinuous Galerkin methods have also been applied to Lagrangian schemes in [61, 59, 60, 79], while higher order finite element methods have been presented in [97, 109, 41, 40, 42] for solving the equations of Lagrangian hydrodynamics. The first high order Godunov-type Lagrangian finite volume schemes were proposed in [28, 81], where a third order accurate essentially non-oscillatory (ENO) reconstruction operator has been employed on moving curved structured meshes.

In [54] Dumbser et al. have proposed a one-dimensional high order Lagrangian ADER finite volume scheme, where a WENO reconstruction technique is adopted to guarantee high order of accuracy in space and the local space-time Galerkin predictor method [49, 66] to achieve high order of accuracy in time. Furthermore they have also considered the case of stiff source terms. This algorithm can be seen as the starting point of the family of high order direct ALE ADER-WENO schemes developed in a series of paper by Boscheri and Dumbser: in [16, 48] the algorithm presented in [54] has been extended to conservative and non-conservative hyperbolic systems on unstructured triangular meshes, while in [20] three-dimensional applications on moving tetrahedral meshes for several hyperbolic balance laws are shown. In [18] three different node solver algorithms are compared and used for the mesh motion, solving the Lagrangian hydrodynamics equations as well as ideal and relativistic magnetohydrodynamics (MHD) equations. In

[4, 6, 9, 7, 8] Balsara et al. have developed a new family of genuinely multidimensional HLL-type Riemann solvers, which have been used to evaluate the numerical fluxes as well as the node velocities in a moving framework [15]. Lagrangian algorithms typically are more demanding in terms of computational efforts rather than Eulerian methods. This is because the geometry is continuously changing in time, hence requiring the geometric quantities (e.g. normal vectors, volumes, side lengths, face surfaces, *etc.*) to be updated and recomputed at *each* timestep. Indeed we can no longer store all geometry-related quantities and operators once and for all in a preprocessing stage. In [20] it is shown that about 90% of the total computational time is needed for the high order WENO reconstruction algorithm and the Gaussian quadrature of the numerical flux integrals. In order to improve the efficiency of the algorithm, in [15] the use of a genuinely multi-dimensional HLL-type Riemann solver [4, 6, 9] led to larger timesteps, hence yielding a computationally more efficient scheme compared to a method based on classical one-dimensional Riemann solvers. A second attempt for reducing the computational efforts has been presented in [46, 19], where a local time stepping scheme on moving meshes is presented. The local time stepping allows the algorithm to be free from a global CFL stability condition, because each control volume reaches the final time of the simulation using its own optimal timestep. Following [52], in a very recent work [17] we presented a quadrature-free flux computation on moving unstructured meshes in two and three space dimensions leading to an important improvement of the overall algorithm efficiency. Up to our knowledge, no work has been done so far to optimize directly the reconstruction procedure itself, within the framework of better than second order accurate direct ALE-ADER finite volume schemes. In [51] the authors used a WENO reconstruction technique where the reconstruction matrix is calculated and stored for all elements once and for all in a preprocessing step. Unfortunately, as we already mentioned, this is not possible anymore in a moving mesh framework. Yet, another possibility of improving the overall efficiency of the direct ALE numerical method is to reconsider the use of a WENO technique. The WENO approach requires the blending of seven (2D) or nine (3D) polynomial reconstructions per cell per variable using nonlinear weights to obtain essentially-non-oscillatory reconstructions. This computational cost is almost impossible to reduce if the WENO paradigm is not questioned.

In the finite volume context a new concept has been recently proposed, namely the Multi-dimensional Optimal Order Detection approach (MOOD), which is an *a posteriori* approach to the problem of limiting. Indeed, the key idea of this paradigm is to run a spatially unlimited high-order finite volume scheme in order to produce a so-called *candidate solution*. Then, the validity of the candidate solution is tested against a set of pre-defined *admissibility criteria*. Some cells are marked as 'acceptable' and are therefore valid. Some others may be locally marked as 'problematic' if they do not pass the detection process. These cells are consequently *locally recomputed* using polynomial reconstructions of a lower degree. Thus, after decrementing the polynomial degree and locally recomputing the solution, a new candidate solution is obtained. That solution is again tested for validity and the decrementing procedure re-applies, if necessary. Such degree decrementing can occur several times within one time step for the same cell, but it will always halt after a finite number of steps: either the cell is valid for a polynomial degree greater than 0, or the degree zero is reached. In the latter case, which is the worst case scenario, the cell is updated with the robust and stable first order accurate Godunov finite volume scheme, that is supposed to produce always valid (monotone and positivity-preserving) solutions under a CFL stability condition. This iterative *a posteriori* detection and decrementing loop is called the *MOOD loop*. We refer the reader to [29, 38, 39, 82] for more details. Contrarily to WENO, the MOOD paradigm does not require several polynomial reconstructions per cell: only one central stencil is considered to perform the high order reconstruction. Moreover the detection of problematic cells is made *a posteriori*, that is to say on a candidate solution at time t^{n+1} . Consequently it drastically eases the test of the candidate solution against any desirable properties (positivity, mesh validity, *etc.*). In addition, a list of problematic cells that need to be re-updated, can be constructed from the previous checks. If the detection criteria as well as the decrementing procedure are well designed, for an M th order accurate ALE-MOOD scheme we may expect to retrieve at least the same accuracy and ENO behavior than the equivalent M th order accurate ALE-WENO scheme. On the other hand we could expect genuine

gains in CPU time and memory consumption in favor of MOOD. Thereby the main purpose of this paper is to design a MOOD approach within our existing high order Arbitrary-Lagrangian-Eulerian ADER finite volume framework. The new algorithm, which will be addressed with ALE-MOOD, is expected to be as accurate as the original ALE-WENO formulation, but more efficient and more robust. The rest of this paper is organized as follows. The second section briefly presents the direct high order accurate unlimited ADER Arbitrary-Lagrangian-Eulerian (ALE) scheme which was previously designed in [20]. Then in section 3 the MOOD technique in replacement of the WENO procedure is fully described in this ALE framework, leading to a direct *a posteriori* limited ADER-MOOD ALE scheme. Detection criteria, decrementing technique as well as implementation issues and developer choices (cascade of schemes, parachute bulletproof scheme) are detailed in the same section. Next, section 4 gathers all numerical results for a large set of different test cases run in 2D and 3D. The goal is to assess the validity and the robustness of our high order accurate ALE scheme supplemented with the *a posteriori* MOOD detection and decrementing technique. Smooth, irregular and complex flows involving shock waves, contact discontinuities and rarefaction waves in interaction are simulated on unstructured grids. Efficiency of the overall scheme is measured and further compared to the ALE-WENO version of the code. Finally conclusions and perspectives are drawn in section 5.

2. High order accurate unlimited Arbitrary-Lagrangian-Eulerian ADER scheme

In this paper we consider general nonlinear systems of hyperbolic conservation laws in multiple space dimensions, and in particular we apply our new scheme to the Euler equations of compressible gas dynamics. For the sake of completeness we present in the following the three dimensional formulation of the governing equations, knowing that the two dimensional equations can be obtained by neglecting the z component and its associated variables.

Let $\mathbf{Q} = (\rho, \rho u, \rho v, \rho w, \rho E)$ be the vector of conserved variables with ρ the fluid density, $\mathbf{v} = (u, v, w)$ the velocity vector and ρE the total energy density. A general formulation that is suitable to write the Euler system reads

$$\frac{\partial \mathbf{Q}}{\partial t} + \nabla \cdot \mathbf{F}(\mathbf{Q}) = 0, \quad \mathbf{x} \in \Omega \subset \mathbb{R}^3, t \in \mathbb{R}_0^+, \quad (1)$$

where $\mathbf{F} = (\mathbf{f}, \mathbf{g}, \mathbf{h})$ is the conservative nonlinear flux tensor given by

$$\mathbf{f} = \begin{pmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ \rho uw \\ u(\rho E + p) \end{pmatrix}, \quad \mathbf{g} = \begin{pmatrix} \rho v \\ \rho uv \\ \rho v^2 + p \\ \rho vw \\ v(\rho E + p) \end{pmatrix}, \quad \mathbf{h} = \begin{pmatrix} \rho w \\ \rho uw \\ \rho vw \\ \rho w^2 + p \\ w(\rho E + p) \end{pmatrix}. \quad (2)$$

The system is closed by the equation of state for an ideal gas:

$$p = (\gamma - 1) \left(\rho E - \frac{1}{2} \rho \mathbf{v}^2 \right) = (\gamma - 1) \rho \varepsilon, \quad (3)$$

where p is the fluid pressure and γ the ratio of specific heats. Let furthermore $\varepsilon = \left(E - \frac{1}{2} \mathbf{v}^2 \right)$ be the specific internal energy. Provided that an admissible fluid state is considered, that is to say $\rho > 0$ and $\varepsilon > 0$, then the speed of sound is given by $c = \sqrt{\frac{\gamma p}{\rho}} = \sqrt{\gamma(\gamma - 1)\varepsilon}$.

In our moving framework the computational domain $\Omega(t) \subset \mathbb{R}^3$ is discretized at any time level t^n by a set of moving and deforming tetrahedral elements T_i^n . N_E denotes the total number of elements and the union of all elements is referred to as the tetrahedrization \mathcal{T}_Ω^n of the domain:

$$\mathcal{T}_\Omega^n = \bigcup_{i=1}^{N_E} T_i^n. \quad (4)$$

We assume that the mesh configuration continuously changes in time. Consequently we define the mapping between the physical element T_i^n to a reference element T_e via a local reference coordinate system $\xi - \eta - \zeta$, see Fig. 1. The spatial reference element T_e is the unit tetrahedron defined by the vertices

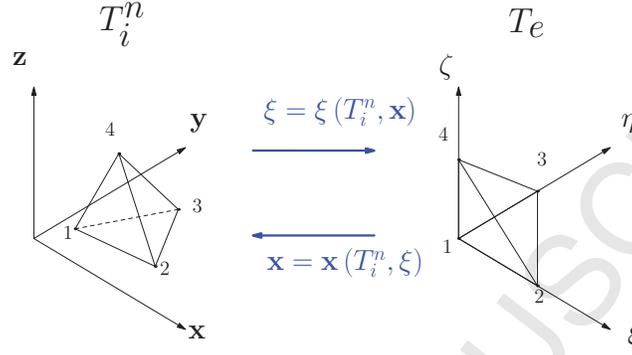


Figure 1: Spatial mapping from the physical element T_i^n defined with $\mathbf{x} = (x, y, z)$ to the unit reference tetrahedron T_e in $\xi = (\xi, \eta, \zeta)$.

$\xi_{e,1} = (\xi_{e,1}, \eta_{e,1}, \zeta_{e,1}) = (0, 0, 0)$, $\xi_{e,2} = (\xi_{e,2}, \eta_{e,2}, \zeta_{e,2}) = (1, 0, 0)$, $\xi_{e,3} = (\xi_{e,3}, \eta_{e,3}, \zeta_{e,3}) = (0, 1, 0)$ and $\xi_{e,4} = (\xi_{e,4}, \eta_{e,4}, \zeta_{e,4}) = (0, 0, 1)$, where $\xi = (\xi, \eta, \zeta)$ and $\mathbf{x} = (x, y, z)$ are the vector of the spatial coordinates in the reference system and the position vector in the physical system, respectively. Let $\mathbf{X}_{k,i}^n = (X_{k,i}^n, Y_{k,i}^n, Z_{k,i}^n)$ be the vector of physical spatial coordinates of the k -th vertex of tetrahedron T_i^n , then the linear mapping from T_i^n to T_e is given by

$$\mathbf{x} = \mathbf{X}_{1,i}^n + (\mathbf{X}_{2,i}^n - \mathbf{X}_{1,i}^n)\xi + (\mathbf{X}_{3,i}^n - \mathbf{X}_{1,i}^n)\eta + (\mathbf{X}_{4,i}^n - \mathbf{X}_{1,i}^n)\zeta. \quad (5)$$

For any finite volume scheme data are represented at a given time t^n by piecewise constant cell averages. As a consequence we define at each time level t^n within the control volume T_i^n the mean value of the state vector \mathbf{Q}_i^n as

$$\mathbf{Q}_i^n = \frac{1}{|T_i^n|} \int_{T_i^n} \mathbf{Q}(\mathbf{x}, t^n) d\mathbf{x}, \quad (6)$$

where $|T_i^n|$ denotes the volume of element T_i^n .

In the next Section 2.1 a polynomial reconstruction technique is described and used to obtain *unlimited* piecewise high order reconstruction polynomials $\mathbf{w}_h(\mathbf{x}, t^n)$ from the known cell averages \mathbf{Q}_i^n . Contrarily to [55, 20] we do *not* employ a WENO technique here to limit the reconstruction, instead we use an iterative *a posteriori* MOOD loop described later in Section 3. High order of accuracy in time is further achieved in Section 2.2 by applying a local space-time Galerkin predictor method starting from the high order reconstruction polynomials $\mathbf{w}_h(\mathbf{x}, t^n)$.

2.1. Polynomial reconstruction

The reconstruction operator generates piecewise polynomials $\mathbf{w}_h(\mathbf{x}, t^n)$ of degree M which are computed for each element T_i^n considering the so-called reconstruction stencil \mathcal{S}_i and its associated known cell averages. The reconstruction stencil \mathcal{S}_i is composed of a number of n_e elements, including the element T_i^n itself and a set of neighbor elements. The number of stencil elements n_e must be larger than the smallest possible number $\mathcal{M} = (M+1)(M+2)(M+3)/6$ that would be needed to reach the nominal order of accuracy $M+1$, according to linear stability considerations [10, 96, 74]. As suggested in [51, 50] for an unstructured mesh we usually take $n_e \simeq 2\mathcal{M}$ in 2D and $n_e \simeq 3\mathcal{M}$ in three space dimensions. The stencil is recursively constructed: first the Voronoi neighborhood of the current cell T_i^n is listed in stencil \mathcal{S}_i , then, if the number

of elements has not reached the prescribed number n_e , the Voronoi neighborhood of any cell already stored in \mathcal{S}_i is added to this list, etc.

In the WENO approach adopted in [55, 20], nine reconstruction stencils are first determined and further used to compute nine different polynomials for each cell of the computational domain. These stencils are supposed to cover sufficiently enough “directions” for the polynomials in order to “catch” local phenomena. Next, these nine polynomials are blended with nonlinear weights to obtain the actual high order nonlinear WENO reconstruction polynomials $\mathbf{w}_h(\mathbf{x}, t^n)$.

In this work only *one single central stencil* is considered with n_e elements, hence we compute only *one* high order reconstruction polynomial per cell. Let us call this unique stencil \mathcal{S}_i , hence

$$\mathcal{S}_i = \bigcup_{j=1}^{n_e} T_{m(j)}^n, \quad (7)$$

where $1 \leq j \leq n_e$ is a local index counting the elements in the stencil and $m(j)$ is a mapping from the local index j to the global index of the element in \mathcal{T}_Ω^n .

We rely on the orthogonal Dubiner-type basis functions $\psi_l(\xi, \eta, \zeta)$ [44, 73, 34], defined on the reference tetrahedron T_e , to explicitly write the high order reconstruction polynomial as

$$\mathbf{w}_h(\mathbf{x}, t^n) = \sum_{l=1}^{\mathcal{M}} \psi_l(\xi) \hat{\mathbf{w}}_{l,i}^n := \psi_l(\xi) \hat{\mathbf{w}}_{l,i}^n, \quad (8)$$

where the mapping to the reference coordinate system is given by (5) and the $\hat{\mathbf{w}}_{l,i}^n$ denote the unknown degrees of freedom, also called expansion coefficients, of the reconstruction polynomial on stencil \mathcal{S}_i for element T_i^n at time t^n . In the rest of the paper we will use the tensor index notation based on the Einstein summation convention implying summation over two equal indices.

We require integral conservation for the reconstruction on each element T_j^n belonging to stencil \mathcal{S}_i , hence

$$\frac{1}{|T_j^n|} \int_{T_j^n} \psi_l(\xi) \hat{\mathbf{w}}_{l,i}^n d\mathbf{x} = \mathbf{Q}_j^n, \quad \forall T_j^n \in \mathcal{S}_i. \quad (9)$$

This leads to an overdetermined linear system of equations for the unknowns $\hat{\mathbf{w}}_{l,i}^n$ that can be solved using either a least squares technique (LSQ), see [51], or a more sophisticated singular value decomposition (SVD) algorithm. The so-called reconstruction matrix, which is given by the integrals of the linear system (9), depends on the geometry of the tetrahedral elements in stencil \mathcal{S}_i which are moving in an ALE framework. As a consequence the reconstruction matrix can not be stored and the linear system can not be pre-inverted in the preprocessing step as done in the Eulerian framework [82]. Here, system (9) is solved on-the-fly at *each* timestep. For further details on the reconstruction technique we refer the reader to [51, 20].

2.2. Local space-time Galerkin predictor on moving curved tetrahedra

We refer the reader to [16, 17] for a detailed and exhaustive description of the local space-time Continuous Galerkin predictor on moving curved triangular and tetrahedral meshes, used also here without any modification.

Such a technique allows the reconstructed polynomials $\mathbf{w}_h(\mathbf{x}, t^n)$ to be evolved during one time step Δt *locally* within each element $T_i(t)$ without requiring any neighbor interaction. As a result, the local space-time Galerkin procedure generates an *element-local predictor* for the numerical solution \mathbf{q}_h , for the fluxes $\mathbf{F}_h = (\mathbf{f}_h, \mathbf{g}_h, \mathbf{h}_h)$ and also for the mesh velocity \mathbf{V}_h , which are approximating as

$$\mathbf{q}_h = \mathbf{q}_h(\tilde{\xi}) = \theta_l(\tilde{\xi}) \widehat{\mathbf{q}}_{l,i}, \quad \mathbf{F}_h = \mathbf{F}_h(\tilde{\xi}) = \theta_l(\tilde{\xi}) \widehat{\mathbf{F}}_{l,i}, \quad \mathbf{V}_h = \mathbf{V}_h(\tilde{\xi}) = \theta_l(\tilde{\xi}) \widehat{\mathbf{V}}_{l,i}. \quad (10)$$

Here the space-time basis functions $\theta_l = \theta_l(\tilde{\xi}) = \theta_l(\xi, \eta, \zeta, \tau)$ are defined in the space-time reference system and they are given by the Lagrange interpolation polynomials passing through the set of space-time nodes $\tilde{\xi}_m = (\xi_m, \eta_m, \zeta_m, \tau_m)$ explicitly specified in [47, 55, 20]. Further details on the local space-time Galerkin predictor can be found in the aforementioned references.

2.3. Mesh motion

The aim of any ALE scheme is to follow as closely as possible the flow motion. The fluid flow typically generates highly deformed control volumes that may drastically reduce the admissible timestep, according to (22). In order to guarantee good resolution properties for contact waves and material interfaces together with a good mesh quality without folded elements, the mesh velocity must be chosen carefully. For this reason a suitable Lagrangian *node solver* technique [37, 87] is used to assign to each node k a *unique* velocity vector $\bar{\mathbf{V}}_k$ accurately representing the “true” fluid velocity, therefore leading to the new vertex *Lagrangian coordinates* \mathbf{X}_k^{Lag} :

$$\mathbf{X}_k^{Lag} = \mathbf{X}_k^n + \Delta t \bar{\mathbf{V}}_k, \quad (11)$$

where \mathbf{X}_k^n represent the coordinates of node k at the current time level t^n . To maintain an overall acceptable geometrical mesh quality we rely on a local *rezoning algorithm* [126, 76] which produces a new vertex position \mathbf{X}_k^{Rez} that does not take into account the underlying flow features. In this work we use the algorithm proposed in [76] which carries out a minimization of a node-based local objective function. The final node position is given by a weighted linear combination between the Lagrangian coordinates \mathbf{X}_k^{Lag} and the rezoned coordinates \mathbf{X}_k^{Rez} using the relaxation algorithm of Galera et al. [62], that is

$$\mathbf{X}_k^{n+1} = \mathbf{X}_k^{Lag} + \omega_k (\mathbf{X}_k^{Rez} - \mathbf{X}_k^{Lag}), \quad (12)$$

where ω_k is a node-based coefficient associated to the deformation of the Lagrangian grid over the time step Δt , see [62] for the details. We refer the reader to [18, 20] for an exhaustive description of the entire process.

Since we are dealing with a direct ALE formulation we have all the freedom to set the mesh velocity, hence the new vertex position. As a consequence we could run the ALE code in a pure Eulerian regime by setting $\mathbf{X}_k^{n+1} = \mathbf{X}_k^0$ or in an almost Lagrangian regime with $\mathbf{X}_k^{n+1} = \mathbf{X}_k^{Lag}$. We could also force $\omega_k = 1$ leading to a pure geometrical rezoning with coordinates $\mathbf{X}_k^{n+1} = \mathbf{X}_k^{Rez}$ or we might even prescribe any user-given mesh velocity.

2.4. Finite volume scheme

In order to develop a finite volume scheme on a moving mesh, we adopt the same approach presented in [16, 20]. There, the governing PDE (1) is reformulated more compactly using a space-time divergence operator $\tilde{\nabla}$ yielding

$$\tilde{\nabla} \cdot \tilde{\mathbf{F}} = \mathbf{0}, \quad \tilde{\nabla} = \left(\frac{\partial}{\partial x}, \frac{\partial}{\partial y}, \frac{\partial}{\partial z}, \frac{\partial}{\partial t} \right)^T, \quad (13)$$

with the space-time flux tensor $\tilde{\mathbf{F}} = (\mathbf{f}, \mathbf{g}, \mathbf{h}, \mathbf{Q})$. The state vector at the new time level \mathbf{Q}^{n+1} is evaluated by integrating the balance law (13) over a *four-dimensional* space-time control volume $C_i^n = T_i(t) \times [t^n, t^{n+1}]$, i.e.

$$\int_{C_i^n} \tilde{\nabla} \cdot \tilde{\mathbf{F}} dxdt = \mathbf{0}. \quad (14)$$

Gauss theorem allows the space-time volume integral (14) to be expressed as the sum of the fluxes computed over the *three-dimensional* manifold ∂C_i^n (the boundary of the 4D space-time control volume) given by the initial and final configuration of the tetrahedron at times t^n and t^{n+1} , respectively, as well as by the evolution of each lateral face of element $T_i(t)$ within the timestep. It reads

$$\int_{\partial C_i^n} \tilde{\mathbf{F}} \cdot \tilde{\mathbf{n}} dS = \mathbf{0}, \quad (15)$$

where $\tilde{\mathbf{n}} = (\tilde{n}_x, \tilde{n}_y, \tilde{n}_z, \tilde{n}_t)$ denotes the outward pointing space-time unit normal vector on the space-time sub-volume $\partial C_i^n j$, which will be immediately defined. The space-time volume ∂C_i^n needed in the previous integral involves a total number of six space-time sub-manifolds:

$$\partial C_i^n = \left(\bigcup_{T_j(t) \in \mathcal{N}_i} \partial C_{ij}^n \right) \cup T_i^n \cup T_i^{n+1}. \quad (16)$$

Indeed T_i^n and T_i^{n+1} represent the tetrahedron configuration at times t^n and t^{n+1} , respectively, while ∂C_i^n is composed by four space-time sub-volumes ∂C_{ij}^n , each of them defined for each face ij of tetrahedron $T_i(t)$. Here we consider \mathcal{N}_i the *Neumann neighborhood* of tetrahedron $T_i(t)$, which is the set of directly adjacent neighbors $T_j(t)$ that share a common face $\partial T_{ij}(t)$ with tetrahedron $T_i(t)$.

Discretization of (15) gives the following high order direct ALE finite volume scheme:

$$|T_i^{n+1}| \mathbf{Q}_i^{n+1} = |T_i^n| \mathbf{Q}_i^n - \sum_{T_j \in \mathcal{N}_i} \int_{\partial C_{ij}^n} \tilde{\mathbf{F}}_{ij} \cdot \tilde{\mathbf{n}}_{ij} dx dt. \quad (17)$$

The integral on the right hand side of the previous equation is computed using Gaussian quadrature, see [113], of sufficient precision on a reference element whose parametrization is fully described in [16, 20]. In (17) the discontinuity of the predictor solution \mathbf{q}_b at the space-time sub-volume ∂C_{ij}^n is resolved by a numerical flux function $\tilde{\mathbf{F}}_{ij} \cdot \tilde{\mathbf{n}}_{ij}$. We adopt two types of numerical flux: either a robust Rusanov type scheme [105] or a less diffusive Osher type flux [53].

2.5. Timestep constraint

The timestep Δt , which is needed for the discretization of the governing equations given by (17), is computed taking into account two different criteria, namely a classical CFL stability condition and a user-defined geometrical limitation.

The Courant-Friedrichs-Levy (CFL) stability condition requires the maximum admissible timestep to be determined by

$$\Delta t_{CFL} = \text{CFL} \min_{T_i^n} \frac{d_i}{|\lambda_{\max,i}|}, \quad \forall T_i^n \in \Omega^n, \quad (18)$$

with d_i denoting the insphere diameter of tetrahedron T_i^n and $|\lambda_{\max,i}|$ corresponding to the maximum absolute value of the eigenvalues computed from the solution \mathbf{Q}_i^n in T_i^n . On general unstructured meshes the CFL stability condition must satisfy the inequality $\text{CFL} \leq \frac{1}{d}$, with d representing the number of space dimensions.

The second criterion is based on the limitation of the rate of change of the element volume within one timestep, i.e. the volume of each cell T_i^n is not allowed either to increase or to decrease more than a certain threshold which is provided by the user at the beginning of the computation. Such a limitation is typically adopted in the Lagrangian and ALE framework [87, 92]. As clearly stated by (18), the timestep tends to become very small when the elements are highly compressed or stretched. Therefore, when compression occurs, we impose the condition

$$|T_i^{n+1}| \leq C_v |T_i^n|, \quad (19)$$

where C_v is a coefficient which sets the maximum admissible variation of volume for the cell. For our applications we set $C_v = 0.8$ and the volume at the new time level $|T_i^{n+1}|$ is conveniently estimated by using the *current* vertex velocities $\bar{\mathbf{V}}_k^n$ for each vertex k of T_i^n to approximate the new vertex positions \mathbf{X}_k^{n+1} , hence obtaining

$$\mathbf{X}_k^{n+1} \approx \mathbf{X}_k^n + \Delta t_{V,i} \bar{\mathbf{V}}_k^n \quad \forall k \in T_i^n, \quad (20)$$

with $\Delta t_{V,i}$ denoting the unknown timestep which satisfies the volume criterion (19). By using expression (20) to explicitly derive a formula for the new cell volume $|T_i^{n+1}|$, the volume restriction (19) constitutes a second

or third order algebraic equation for the unknown $\Delta t_{V,i}$, in two and three space dimensions, respectively. The final timestep will be given by taking as always the minimum between all $\Delta t_{V,i}$, i.e.

$$\Delta t_V = \min_{T_i^n} \Delta t_{V,i}. \quad (21)$$

According to [87] the final expression for the next timestep Δt^{n+1} is given by

$$\Delta t^{n+1} = \min(\Delta t_{CFI}, \Delta t_V, C_M \Delta t^n), \quad (22)$$

where Δt^n is the current timestep and $C_M = 1.01$ is a coefficient which allows the timestep to smoothly increase avoiding large and quick modifications of Δt .

3. Direct Arbitrary-Lagrangian-Eulerian ADER-MOOD scheme with *a posteriori* limiting

The direct ADER ALE method described previously is of high order of accuracy both in space and time through the construction of space and time polynomials of high accuracy. Without any limiter, the designed order of accuracy of the scheme is formally $M + 1$ for sufficiently smooth solutions. However, to ensure stability and robustness of the scheme some mechanism must be added to artificially dissipate the well known Gibbs phenomenon that arises when steep fronts or shock waves are present. In [20] the limiting/dissipation was introduced by the use of a WENO reconstruction technique, see [70, 69, 50, 74, 51, 118, 122]. The numerical results in [20] show that this approach maintains the overall accuracy and provides the necessary robustness of the scheme.

However, some drawbacks can be pointed out within this ALE-WENO formulation. First, the problem of positivity preservation that requires a special technique [5] where a flattener variable is computed to smear out the oscillations and to bring back density and pressure values to a physically admissible range. Second, the WENO reconstruction technique requires several polynomial reconstructions for each cell. Typically in 2D/3D seven/nine polynomials must be reconstructed per cell and per variable by varying the stencils around the current cell. The determination of the polynomial coefficients was one of the most expensive parts of our original ADER-WENO ALE algorithm. Therefore, any improvement or reduction of this number of stencils could create a gain of the overall efficiency.

In this paper we present an alternative technique to WENO in this ALE context using the so-called *a posteriori* MOOD (Multi-dimensional Optimal Order Detection) limiter method. MOOD has been designed originally on fixed grids for the compressible Euler equations [29, 38, 39, 30]. Recently, in [82] it has been successfully substituted to the WENO reconstruction within a high order 3D ADER finite volume scheme designed for fixed grids and solving different systems of hyperbolic conservation laws. Even more recently the *a posteriori* MOOD concept has been successfully used as an *a posteriori* subcell limiter for high order accurate Discontinuous Galerkin schemes in [129], or as an efficient high-order finite volume solver for convection-diffusion problems [32, 31, 101] or, lastly, to construct all-entropy finite volume schemes [124, 123]. Note that in an indirect ALE context, for the so-called remap phase, an *a posteriori* slope limiter has been proposed in [99].

3.1. MOOD paradigm as an *a posteriori* stabilization technique

The *a posteriori* MOOD paradigm is based on the fact that it is relatively easy to check *a posteriori* the validity of a discrete numerical solution at the end of a timestep given physical and/or numerical detection criteria. From this check we can also easily extract a list of truly problematic cells. By 'truly' we mean cells which have not pass the checks. Then going back in time at the beginning of the timestep, we can re-update only these cells with a more viscous, dissipative and robust numerical scheme. As such a new candidate solution, locally updated with two different schemes, is available to be tested against the validity criteria.

If problematic cells are still detected, then, at last, a very dissipative and extremely robust first order finite volume scheme is used on these cells. These ones are re-updated for the second time still within the current timestep.

This iterative MOOD loop ends with cells updated either with a high accurate but less robust scheme or with a low accurate but more robust one.

The main three entities which must be given to an iterative MOOD loop are:

1. The *Detection criteria*. The detection criteria are a list of properties which are checked to assess if a numerical solution in a cell is acceptable. The first set of criteria is based on the physics underlying the simulation, these correspond to properties that must be fulfilled to ensure physical admissibility of a numerical solution. They are called the Physical Admissible Detection (PAD) criteria. For the hydrodynamics system of equations the PAD criteria are the positivity of the density and the specific internal energy (or pressure). By definition the PAD criteria depend on the PDE system to be solved. The second set of criteria is based on numerical properties. These are called Numerical Admissible Detection (NAD) criteria and they ensure that the numerical solution is essentially non-oscillatory. In previous works [29, 38, 39] the NAD criteria are based on a some sort of relaxed discrete maximum principle (DMP). Also we check if the computed solution is an acceptable data, that is to say we check for *Not-a-Number* (NaN) situations. If either the PAD or the NAD criteria are not fulfilled, then, locally some action must be taken to supplement the scheme with more dissipation.
2. The *cascade*. The cascade of schemes is a list of ordered numerical schemes, from the most accurate one up to the least accurate but robust one [82]. This sequence is related to the accuracy which the polynomial reconstructions are done with. We set a maximal polynomial degree d_{\max} meaning that, in an ideal situation, the corresponding scheme is a d_{\max} -th order accurate scheme in space and time. Then several intermediate polynomial degrees for the reconstructions can be tried, for instance we can use the following cascade: $\mathbb{P}_{d_{\max}} \rightarrow \mathbb{P}_2 \rightarrow \mathbb{P}_1 \rightarrow \mathbb{P}_0$. Other decrementing procedures involving less schemes can be also used, for instance $\mathbb{P}_{d_{\max}} \rightarrow \mathbb{P}_0$, or more advanced reconstructions with limiters, $\mathbb{P}_{d_{\max}} \rightarrow \mathbb{P}_1^{\text{LM}} \rightarrow \mathbb{P}_0$.
3. The *parachute* or *bulletproof* scheme. The last scheme of the cascade is called the *parachute* to express the fact that, in any difficult case, when the detection criteria are not fulfilled then it is used as a last resort scheme. As such the candidate numerical solution it provides must always be considered as a valid one. In most works involving a MOOD loop, for instance [29, 38, 39, 30, 82], a first-order Godunov-type finite volume scheme is used, to ensure properties like positivity when extreme phenomena occur.

In the next subsections we present in detail how this MOOD paradigm can be used as an *a posteriori* stabilization technique in the high-order direct ALE-ADER finite volume context on moving unstructured meshes.

In our direct ALE framework we have constructed a nominally high-order space-time scheme. In section 2.1 we have presented a polynomial reconstruction technique with no extra dissipation, meaning that the discrete numerical solution and the displaced mesh at t^{n+1} in in general not free from spurious oscillations.

In this section we provide the detection criteria that determine which solution is considered acceptable. Then we justify the chosen cascade of schemes and the parachute/bulletproof scheme. Finally, we focus on some implementation details.

3.2. *A posteriori* Detection criteria

Here we describe the detection criteria which are used for all test cases to validate if a candidate solution \mathcal{Q}_i^* for cell i is acceptable or not. The superscript $*$ refers to the fact that the numerical solution \mathcal{Q}_i is at discrete time t^{n+1} but it is not yet the final accepted solution. It will become acceptable only after it escapes the MOOD loop.

PAD. The Physical Admissible Detection criteria are based on the system of conservation laws which are solved. For a numerical solution to be valid in the case of Euler equations with perfect gas equation of state, the solution must obey the following positivity criteria:

$$\rho_i^* > 0, \quad \varepsilon_i^* = E_i^* - \frac{1}{2} |\mathbf{v}_i^*|^2 > 0. \quad (23)$$

Furthermore in a moving mesh framework the volume of any cell i must also be strictly positive that is:

$$|T_i^*| > 0. \quad (24)$$

NAD. The Numerical Admissible Detection criteria are based on the relaxed Discrete Maximum Principle (DMP) with the so-called u2 criterion [39] applied on each conservative variable. Let us briefly describe the DMP+u2 detection process on a generic variable A and a candidate solution A_i^* at time t^{n+1} in cell i for a given set of neighbor cells with index $j \in \mathcal{V}_i$. The set of vertex neighbors \mathcal{V}_i contains all neighbors of cell T_i that have a common vertex with T_i . First, if A_i^* fulfills the DMP, that is

$$\min_{j \in \mathcal{V}_i} (A_j^n, A_i^n) \leq A_i^* \leq \max_{j \in \mathcal{V}_i} (A_j^n, A_i^n), \quad (25)$$

then the cell is valid for this variable. If the DMP is not fulfilled, then one checks the u2 criterion [38, 39], which determines if this new extremum is smooth or not. A candidate solution A_i^* in cell T_i^* which violates the DMP is nonetheless eligible if the following holds

$$\mathcal{X}_i^{max} \mathcal{X}_i^{min} > 0 \quad \text{and} \quad \left| \frac{\mathcal{X}_i^{min}}{\mathcal{X}_i^{max}} \right| \geq 1 - \varepsilon, \quad (26)$$

where ε is a smoothness parameter set to $1/2$, \mathcal{X}_i^* represents a “measure” of local discrete directional curvature, for instance the second derivative in x direction of the third order polynomial reconstruction. Moreover we set

$$\mathcal{X}_i^{min} = \min_{j \in \mathcal{V}_i} (\mathcal{X}_i^*, \mathcal{X}_j^*) \quad \text{and} \quad \mathcal{X}_i^{max} = \max_{j \in \mathcal{V}_i} (\mathcal{X}_i^*, \mathcal{X}_j^*). \quad (27)$$

The same check is done for the y and z components. Another alternative is to compute the true local curvatures, see [82]. Note that the detection of smooth local extrema has also been discussed in the context of extremum preserving PPM schemes [35] and MPWENO schemes [114, 3].

In our implementation if any of the conservative variables does not fulfill the detection criteria then one considers that all variables need correction on the next MOOD iteration. This implies that the unique cell polynomial degree will be decremented according to the cascade (see below) for all variables. Finally we would like to emphasize that the detection criteria are the core of the MOOD paradigm in the sense that they must be well designed by the developer in order to make clever choices whenever difficult situations occur; not enough dissipation would lead to instability, whereas too much dissipation would lead to a lack of accuracy.

3.3. Cascade and parachute scheme

In this ALE framework we have simply mimicked the cascade already used in the Eulerian framework [29, 38, 39, 82]. Namely we employ the following one: $\mathbb{P}_{d_{max}} \rightarrow \mathbb{P}_1^{PLM} \rightarrow \mathbb{P}_0$. We have taken $d_{max} = 4$ but other values could be used as well. The \mathbb{P}_1^{PLM} scheme uses \mathbb{P}_1 reconstructions with Barth & Jespersen slope limiting [11].

The MOOD loop first computes the unlimited $\mathbb{P}_{d_{max}}$ candidate solution \mathbf{Q}_i^* for each cell i , and checks if any cell is problematic according to the detection criteria. Then all invalid cells, that is all cells which have not passed the detection criteria, are recomputed with \mathbb{P}_1^{PLM} reconstructions. For the numerical flux

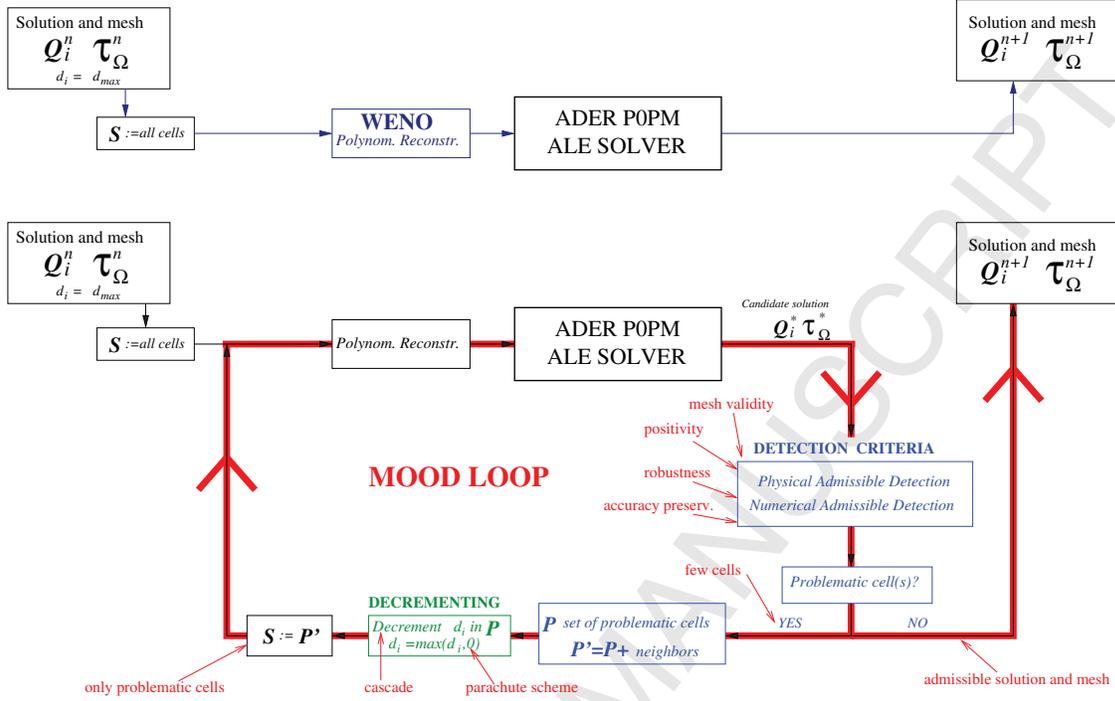


Figure 2: Sketch of the direct ALE ADER- $\mathbb{P}_0\mathbb{P}_M$ simulation code. Top: original ALE-WENO solver. Bottom: sketch of the MOOD loop embracing the existing ALE ADER- $\mathbb{P}_0\mathbb{P}_M$ solver.

evaluation we also need \mathbb{P}_1^{LM} reconstructions for each direct neighbor T_j of T_i . The reason is to assure that the cell is updated with fluxes of the same order of accuracy for each face. This new candidate solution is checked again for validity. Such a candidate solution may enjoy $\mathbb{P}_{d_{\max}}$ -updated cells and \mathbb{P}_1^{LM} -updated ones. Nonetheless, some cells can be invalid again because the \mathbb{P}_1^{LM} may still produce some non-admissible states. Then, for these still invalid cells, we rely on the parachute scheme \mathbb{P}_0 . Under the assumption of being a bulletproof scheme, any remaining problematic cells are presumably properly updated, in the sense that the PAD criteria are fulfilled. In the worst case scenario all cells in the domain are updated with the first-order \mathbb{P}_0 scheme. Contrarily in an ideal situation all cells are updated with the unlimited $\mathbb{P}_{d_{\max}}$ scheme. Note that the MOOD loop always converges to an acceptable discrete solution provided that the parachute scheme produces such a solution.

3.4. MOOD loop

In the present implementation as well as in the Eulerian context [82], the MOOD loop simply embraces the main evolution routines of the high order ADER method and iterates to recompute those cells marked as problematic by the detection criteria. In Figure 2 we have represented a sketch of the MOOD loop. The efficiency of the *a posteriori* MOOD paradigm is mainly due to the fact that usually few cells need a cell polynomial degree decrementing, see the numerical section 4 for several examples. Therefore the extra-work needed to recompute a new candidate solution on problematic cells is low. Moreover, for a given polynomial degree d only one polynomial reconstruction per variable and per cell is computed, which reduces the CPU time but also the memory consumption compared to an *a priori* WENO reconstruction. The flexibility of the *a posteriori* MOOD paradigm is based on:

1. any robust and preferred parachute scheme can be kept as the last and safest scheme;
2. only light modifications are usually needed to implement a MOOD loop within an existing high-order finite volume code because it simply embraces the existing solver; in other words it is non-invasive, see Figure 2;
3. any constraint or property can be added to the list of detection criteria, should it be based on physical or numerical requirements.

In the next section we present a test case suite to verify and validate the implementation of the MOOD loop within the 2D/3D ALE ADER finite volume code on unstructured meshes. We present quantitative efficiency and accuracy measures to illustrate how the *a posteriori* MOOD approach behaves.

4. Numerical experiments

The *a posteriori* MOOD ALE finite volume scheme has been implemented within an MPI parallel 2D and 3D Lagrangian ADER finite volume code on moving unstructured triangular and tetrahedral grids, see [16, 48, 20]. For each test case presented in this section we use a 5th order ALE scheme, i.e. $M = 4$ in \mathbb{P}_M . The polynomial reconstruction on conservative variables is limited with the *a posteriori* MOOD loop and by default we adopt the Rusanov flux [105]. In the next sections we will explicitly state when the Osher-type flux [53] is employed.

For the MOOD approach, the cascade of reconstructions is defined as $\mathbb{P}_M \rightarrow \mathbb{P}_1^{lim} \rightarrow \mathbb{P}_0$. Such a scheme will be denoted throughout the rest of the paper by 'ALE-MOOD- $\mathbb{P}_0\mathbb{P}_4$ '. Some test problems are simulated with successively refined meshes, or different values of M (usually $M = 2$ or 3) to illustrate the behavior of the ALE-MOOD schemes. For comparison purposes we also run the ALE-WENO- $\mathbb{P}_0\mathbb{P}_M$ schemes from [16, 20].

For the majority of the test problems we propose to observe the mesh at the final time and the final cell order (cell polynomial degree plus one) map corresponding to the actual cell degrees used for the last time step. One expects that the detection+decrementing procedure only involves cells which are affected by parasitical oscillations, that is to say, cells which are in the vicinity to shock waves, steep gradients or discontinuous profiles.

CPU times as well as memory consumption will be systematically monitored in order to compare this new ALE-ADER-MOOD approach with the original ALE-ADER-WENO one.

The methodology of testing is based on a sequence of mono-material perfect gas test problems which became classical benchmarks in the Lagrangian community. Here we run:

- Smooth isentropic vortex [69] - This test is designed to observe the high-order of accuracy of a numerical method because it involves a smooth flow field. One expects that in practice the MOOD detection procedure will not detect any problematic cell if the mesh is fine enough. Numerical convergence studies are carried out;
- Sod shock tube [120] - this 1D shock tube problem is classical test to assess the ability of a numerical method to deal with simple waves (rarefaction, contact discontinuity and shock wave);
- Sedov test case [110] - This test simulates a diverging cylindrical/spherical shock wave initiated from high energy deposit at the origin of the computational domain. This problem assesses the ability of the numerical method to capture a cylindrical/spherical shock wave on a grid non-aligned to the flow field without spurious oscillations;
- Noh test case [95] - This problem simulates the implosion of a cylinder/sphere subject to an inward pointing velocity field. A shock wave emanates from the origin and propagates outwards. The ability of the scheme to deal with cylindrical/spherical shock waves as well as its sensitivity to wall heating effect are observed;

- Kidder test case [75] - This test consists in an isentropic compression of a portion of a shell. The location of the shell at final time is often computed to validate the accuracy of the scheme;
- Saltzman test case [45] - The Saltzman test is considered as a stress test for any mesh moving technique. It involves a shock wave that is caused by the motion of a piston traveling along the main direction of a rectangular box. The difficulty is generated by the initially skewed and non-aligned mesh with respect to the flow. For moving mesh techniques, parasitical vortical effects may lead to the failure of the simulation due to unexpected mesh tangling;
- Triple point test case [86] - This problem is a three state one species 2D planar/cylindrical Riemann problem in a vessel which is a mono-material variation of the triple point problem from [86]. Due to the discrepancy in density, two shocks propagate with different velocities generating a shear wave along the slide line coupled with a vortex-like motion. Capturing accurately the vorticity part is the difficult part of such simulation involving interacting shocks, rarefactions and contact waves. This problem is well known in Lagrangian and ALE community as it permits to qualitatively estimate the amount of numerical diffusion implied by the numerical method [86, 40, 42]. No exact solution does exist for this problem but one expects that high accurate numerical scheme may nicely capture the vortex motion.

These test problems have either 1D, 2D or 3D analytical solution apart from the triple point problem. In the case of a smooth flow we can measure the errors produced by any ALE scheme and the associated convergence rates. In case of discontinuous flow we can use the analytical solution to assess the ability of the scheme to capture the correct location of waves and plateaus as well as the symmetry of the flow.

We have run all the aforementioned test cases both in 2D and in 3D. However, to ease the readability of the paper, the numerical results for each test case will be presented either in 2D or in 3D, while the data regarding CPU time and memory consumption are collected for *all* test problems in multiple space dimensions. Specifically, the smooth isentropic vortex test case is used to study numerical convergence, therefore it will be employed in 2D as well as in 3D. Then, in 2D we show the Noh problem and the Kidder problem, while in 3D we present the Sedov problem and the Saltzman problem. The triple point problem will be run in multiple space dimensions, while for the well-known Sod shock tube problem we limit us to give the data contained in Tables 3 and 5. Furthermore in Section 4.1.5 we show the 2D mesh tangling endurance problem, that can be run until the prescribed final time using the MOOD schemes but *not* with the original WENO formulation of our algorithm.

4.1. 2D test problems

In this section we present the numerical results for the two-dimensional version of the test cases listed above.

4.1.1. 2D Isentropic vortex in motion

The 2D isentropic vortex problem was initially developed [111] to test the accuracy of numerical methods. This problem has an exact and smooth solution. The computational domain is $\Omega = [0, 10] \times [0, 10]$ and an ambient flow characterized by $\rho_\infty = 1.0$, $u_\infty = 1.0$, $v_\infty = 1.0$, $p_\infty = 1.0$, with a normalized ambient temperature $T_\infty^* = 1.0$.

A vortex is centered at location $(x_{\text{vortex}}, y_{\text{vortex}}) = (5, 5)$ and is perturbed at the initial time $t = 0$ by $u = u_\infty + \delta u$, $v = v_\infty + \delta v$, $w = w_\infty$, $T^* = T_\infty^* + \delta T^*$ where

$$\delta u = -y' \frac{\beta}{2\pi} \exp\left(\frac{1-r^2}{2}\right), \quad \delta v = x' \frac{\beta}{2\pi} \exp\left(\frac{1-r^2}{2}\right), \quad \delta T^* = -\frac{(\gamma-1)\beta^2}{8\gamma\pi^2} \exp(1-r^2),$$

with $r = \sqrt{x'^2 + y'^2}$ and $x' = x - x_{\text{vortex}}, y' = y - y_{\text{vortex}}$. The vortex strength is driven by β which is chosen equal to 5.0. The initial density is given by

$$\rho = \rho_{\infty} \left(\frac{T^*}{T_{\infty}^*} \right)^{\frac{1}{\gamma-1}} = \left(1 - \frac{(\gamma-1)\beta^2}{8\gamma\pi^2} \exp(1-r^2) \right)^{\frac{1}{\gamma-1}}. \quad (28)$$

Periodic boundary conditions are prescribed and the final time is set to $t_{\text{final}} = 1$. An effective high order of accuracy should be reached for this problem because the exact solution is equal to the smooth initial one shifted by the ambient flow during the time interval of the computation. We measure the errors and the convergence rates using the discrete L_2 and L_{∞} error norms between the discrete solution and the exact solution for the density at the final time. Successively refined grids are constructed given a maximal circumscribed diameter h . In Table 1 we report the errors and the corresponding rates of convergence for ALE-MOOD and ALE-WENO schemes using $\mathbb{P}_0\mathbb{P}_3$ and $\mathbb{P}_0\mathbb{P}_4$ polynomial reconstructions. We also have computed the errors and rates of convergence for an unlimited scheme using \mathbb{P}_3 and \mathbb{P}_4 polynomial reconstructions. This latter provides the reference solution in terms of accuracy and CPU time. These data show that the three schemes reach the nominally expected rates of convergence. Moreover these errors are roughly of the same order but MOOD seems slightly more accurate than WENO. No cell is ever detected as problematic by the MOOD process for this problem, since only smooth profiles are present.

A CPU time comparison between ALE-MOOD and ALE-WENO has been carried out using a single CPU core of an Intel i7-2600 processor with 3.4 GHz of clock speed and 16 GB of RAM, in order to assess the pure serial performance, without accounting for the MPI overhead. The CPU time of the unlimited scheme is used as the reference with respect to which all CPU times have been normalized. This scheme is the fastest method of effective high order because it does not waste time by employing any kind of *a priori* or *a posteriori* limiting. In Table 1 we observe that MOOD is about 2% more expensive than the unlimited scheme, assessing that no problematic cell needs decrementing, whereas WENO is on average 2 times more expensive.

In figure 3 we graphically represent in the left panel the convergence rates for the L_2 norms in log scale versus the expected 4th and 5th order lines, the data are taken from Table 1. In the right panel we also show the CPU time as a function of the initial characteristic cell length for ALE-ADER-WENO and ALE-ADER-MOOD schemes using \mathbb{P}_3 and \mathbb{P}_4 polynomial reconstructions. From this figure we can see that, for this problem, the 5th order accurate ALE-MOOD scheme has approximately the same cost than the 4th order accurate ALE-WENO scheme. We can also observe that MOOD costs about 2 times less than WENO in this configuration.

4.1.2. 2D Noh problem

In a quarter of the unit disk a gas ($\gamma = 5/3$) is initiated with $\rho_0 = 1$, $\varepsilon_0 = 0$ and $\mathbf{v}(x, y) = \left(\frac{-x}{\sqrt{x^2+y^2}}, \frac{-y}{\sqrt{x^2+y^2}} \right)$. This so-called Noh problem generates a cylindrical shock wave at the origin which further diverges. The exact solution is given as a function of cell radius r and time t and can be found in [95] or [83]. At the final time $t_{\text{final}} = 0.6$ this solution is given by a shock wave located at radius $r = 0.2$ (the shock velocity is $\frac{1}{3}$) and the pre- and post-shock states given by

$$(\rho, \varepsilon, u_r) = \begin{cases} \left(16, \frac{1}{2}, 0 \right) & \text{if } r < 0.2, \\ \left(\left(1 + \frac{3}{5}\frac{1}{r}\right), 0, 1 \right) & \text{if } r > 0.2. \end{cases} \quad (29)$$

with u_r denoting the radial velocity component.

A mesh is constructed with $N \times N$ squares further split into two right triangles leading to a triangular mesh made of $N_E = 2N^2$ elements. N takes the values 30, 40 and 50 to show the convergence of the ALE-MOOD- $\mathbb{P}_0\mathbb{P}_4$ scheme under mesh refinement. In Figure 4 we display the mesh colored by the density variable on the top row. On the bottom row we show the scatter plot of cell density as a function of cell-center radius versus the exact solution in red. The cylindrical symmetry of the shock wave and the mesh are

2D isentropic vortex problem — Unlimited — ALE-WENO — ALE-MOOD								
	N_E	h	L_2 error	L_∞ error	L_2 order	L_∞ order	Theor.	CPU time
Unlimited- $\mathbb{P}_0\mathbb{P}_3$	1298	3.29E-01	4.39E-03	3.84E-03	—	—	4	1.0
	2292	2.51E-01	1.70E-03	1.55E-03	3.5	3.4		1.0
	5180	1.68E-01	4.21E-04	4.39E-04	3.4	3.1		1.0
	9192	1.28E-01	1.33E-04	1.49E-04	4.2	4.0		1.0
ALE-WENO- $\mathbb{P}_0\mathbb{P}_3$	1298	3.29E-01	6.85E-03	1.04E-02	—	—	4	1.77
	2292	2.51E-01	1.92E-03	1.56E-03	4.7	7.0		1.84
	5180	1.68E-01	4.22E-04	4.41E-04	3.8	3.1		1.89
	9192	1.28E-01	1.33E-04	1.49E-04	4.3	4.0		1.89
ALE-MOOD- $\mathbb{P}_0\mathbb{P}_3$	1298	3.29E-01	4.39E-03	3.84E-03	—	—	4	1.03
	2292	2.51E-01	1.70E-03	1.55E-03	3.5	3.4		1.02
	5180	1.68E-01	4.21E-04	4.39E-04	3.4	3.1		1.02
	9192	1.28E-01	1.33E-04	1.49E-04	4.2	4.0		1.02
Unlimited- $\mathbb{P}_0\mathbb{P}_4$	1298	3.29E-01	4.42E-03	4.42E-03	—	—	5	1.0
	2292	2.51E-01	1.26E-03	1.17E-03	4.6	4.9		1.0
	5180	1.68E-01	2.24E-04	2.58E-04	4.3	3.7		1.0
	9192	1.28E-01	5.77E-05	7.11E-05	5.0	4.8		1.0
ALE-WENO- $\mathbb{P}_0\mathbb{P}_4$	1298	3.29E-01	6.33E-03	1.01E-02	—	—	5	1.90
	2292	2.51E-01	1.31E-03	1.69E-03	5.7	6.5		1.98
	5180	1.68E-01	2.24E-04	2.58E-04	4.4	4.6		2.01
	9192	1.28E-01	5.77E-05	7.11E-05	5.0	4.8		2.03
ALE-MOOD- $\mathbb{P}_0\mathbb{P}_4$	1298	3.29E-01	4.42E-03	4.42E-03	—	—	5	1.02
	2292	2.51E-01	1.26E-03	1.17E-03	4.6	4.9		1.02
	5180	1.68E-01	2.24E-04	2.58E-04	4.3	3.7		1.02
	9192	1.28E-01	5.77E-05	7.11E-05	5.0	4.8		1.02

Table 1: L_2 and L_∞ errors and convergence rates and CPU time for the 2D isentropic vortex problem for the Unlimited, ALE-WENO and ALE-MOOD schemes for $\mathbb{P}_0\mathbb{P}_3$ and $\mathbb{P}_0\mathbb{P}_4$ polynomial reconstructions from top to bottom.

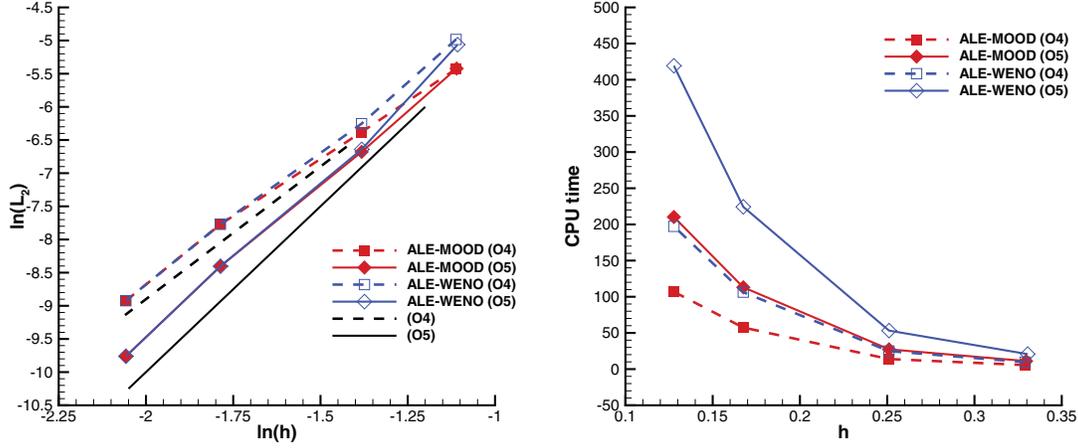


Figure 3: 2D isentropic vortex for ALE-WENO and ALE-MOOD using \mathbb{P}_3 and \mathbb{P}_4 polynomial reconstructions. Left panel: convergence rate for the L_2 norm in log scale versus expected 4th and 5th order lines — Right panel: CPU time in seconds as a function of the final characteristic cell length h .

well preserved. The accuracy for the plateau is also acceptable and one observes the classical wall heating effect close to the origin [102, 121]. A slight overshoot can also be noticed just after the shock wave. The somewhat surprising behaviors around radius 0.35 – 0.4 where the cell density is not anymore accurately located on the exact solution is due to the effect of the boundary conditions for cells on the x and y axis. In Figure 5 is plotted the percentage of cells updated with \mathbb{P}_1^{lim} (red) or \mathbb{P}_0 (blue) polynomial reconstructions as a function of iteration number. The panel on the left presents the first 50 iterations when the explosion occurs. We clearly see that about 50% – 60% of the total number of cells are recomputed with a low order scheme for the first 15 timesteps. This is a large amount of extra work which, nonetheless, is mandatory to stabilize the scheme. But the overall efficiency is not too drastically affected, see the CPU time and memory consumptions provided in section 4.1.6 in Table 3. Then, when the numerical scheme has generated enough numerical dissipation to deal with the initial shock wave, the number of detected and decremented cells drops to few percents. From iteration ~ 100 (see right panel of Figure 5) the number of problematic cells is back to a small amount, about 0.5 – 1.5%. This figure illustrates that Noh problem is difficult to handle at the very beginning, but, as soon as the shock wave has emerged, no more dramatic action has to be taken by the MOOD process. Accordingly to the previous tests we have plotted a cubic fit of these sample points in the right panel taken into account only the sample points after iteration 50.

4.1.3. 2D Kidder problem

A classical benchmark test case for moving mesh techniques is the Kidder problem [75]. It consists in an isentropic compression of a portion of a shell filled with a perfect gas as

$$\begin{pmatrix} \rho_0(r) \\ \mathbf{v}_0(r) \\ p_0(r) \end{pmatrix} = \begin{pmatrix} \left(\frac{r_{e,0}^2 - r^2}{r_{e,0}^2 - r_{i,0}^2} \rho_{i,0}^{\gamma-1} + \frac{r^2 - r_{i,0}^2}{r_{e,0}^2 - r_{e,0}^2} \rho_{e,0}^{\gamma-1} \right)^{\frac{1}{\gamma-1}} \\ 0 \\ s_0 \rho_0(r)^\gamma \end{pmatrix}, \quad (30)$$

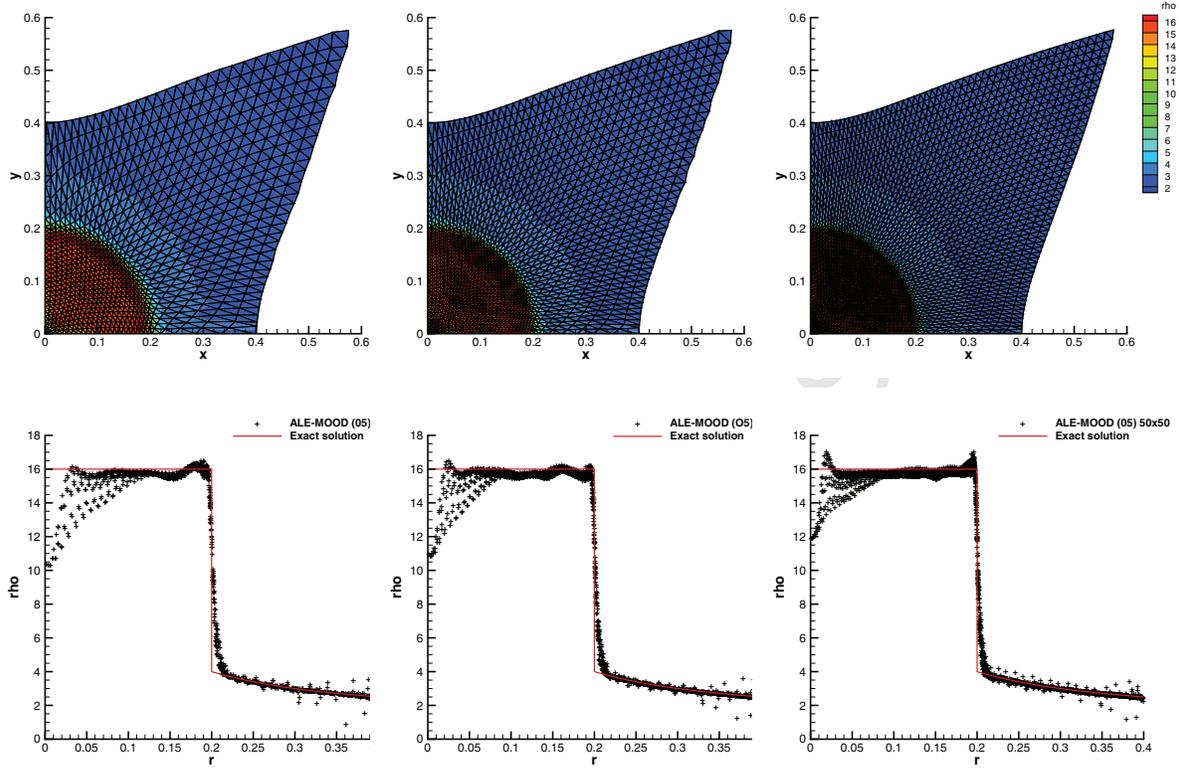


Figure 4: Noh problem — ALE-MOOD- $\mathbb{P}_0\mathbb{P}_4$ with triangular meshes made of $N_E = 2 N^2$ with $N = 30$ (left panels), 40 (middle panels), 50 (right panels) elements — Top-line: mesh and density (color). Bottom-line: scatter plot of cell density as a function of cell-center radius versus the exact solution (red line). All cells are plotted.

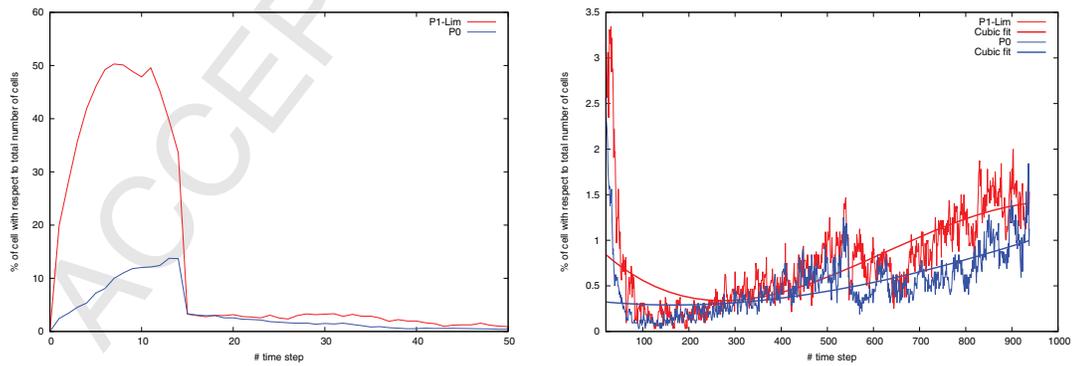


Figure 5: Noh problem — ALE-MOOD- $\mathbb{P}_0\mathbb{P}_4$ — Percentage of cells updated with $\mathbb{P}_1^{\text{Lim}}$ (red) or \mathbb{P}_0 (blue) reconstructions as a function of iteration number. Left panel: first part of the simulation before iteration 50 corresponding to the generation of the shock wave. Right panel: second part of the simulation. Cubic fits are presented on the second panel only.

where $r = \sqrt{x^2 + y^2}$ is the radius and $(r_i(t), r_e(t))$ are the time-dependent internal and external border frontiers of the shell. The densities are set to $\rho_{i,0} = 1$ and $\rho_{e,0} = 2$ and γ is set to $\frac{5}{3}$. Furthermore s_0 denotes the initial and uniform entropy distribution defined by $s_0 = \frac{p_0}{\rho_0} = 1$.

The initial 2D computational domain $\Omega(t = 0)$ is one fourth of the entire shell and is depicted in Figure 6 left panel. Sliding wall boundary conditions are imposed on the lateral faces while a space-time dependent state is set on the remaining boundaries according to the exact analytical solution $R(r, t)$ [75]. This exact solution is defined at the general time t for a fluid particle initially located at radius r as a function of the radius and the homothety rate $h(t)$ as $R(r, t) = h(t)r$, with $h(t) = \sqrt{1 - \frac{t^2}{\tau^2}}$ and τ is the focalisation time

given by the following analytical formula $\tau = \sqrt{\frac{\gamma-1}{2} \frac{(r_{e,0}^2 - r_{i,0}^2)}{c_{e,0}^2 - c_{i,0}^2}}$. Finally $c_{i,e} = \sqrt{\gamma \frac{p_{i,e}}{\rho_{i,e}}}$ represent the internal and external sound speeds.

Following [26, 87], the final time is chosen such that the compression rate is $h(t_{\text{final}}) = 0.5$ with $t_{\text{final}} = \frac{\sqrt{3}}{2} \tau$. Consequently the exact location of the shell is bounded with radii $0.45 \leq R \leq 0.5$.

The computational domain is discretized with a total number of $N_E = 3180$ elements and we use ALE ADER-MOOD- $\mathbb{P}_0\mathbb{P}_4$ scheme together with the Osher-type flux [53] instead of the Rusanov flux.

In Figure 6 the initial and the final meshes along with the density distribution are displayed on top panels. Moreover we present the evolution of the internal and external numerical and exact border locations as a function of time. As can be seen they perfectly match with the exact solution. In Table 2 we provide the absolute error $|err|$ of the radius locations of the internal and external frontiers at final time t_{final} for the Kidder problem in 2D and in 3D. The three dimensional Kidder problem has been run using again an ALE-MOOD- $\mathbb{P}_0\mathbb{P}_4$ scheme. Finally we have computed the number of problematic cells as a function of

r_{ex}	ALE-MOOD- $\mathbb{P}_0\mathbb{P}_4$ 2D		ALE-MOOD- $\mathbb{P}_0\mathbb{P}_4$ 3D	
	r_{num}	$ err $	r_{num}	$ err $
0.450000	0.44999	7.73e-6	0.45045	4.55e-4
0.500000	0.49999	1.01e-5	0.49935	6.44e-4

Table 2: Kidder problem in 2D and 3D — Absolute error for the internal and external radius location between exact (r_{ex}) and numerical (r_{num}) solution.

time to assess that the MOOD detection and decrementing technique is performing as expected on such a smooth flow. Indeed no cell is ever detected as problematic during the entire simulation, meaning that the numerical solution after the MOOD loop is the unlimited one. At last the CPU time consumption and memory consumption can be found in Table 3.

4.1.4. 2D Mono-material triple point problem

This problem is a three state one specie 2D Riemann problem in a vessel which is a variation of the triple point problem from [86]. The triple point problem simulation domain is $\Omega(t = 0) = [0; 7] \times [0; 3]$. Ω is split into three subdomains filled with a perfect gas with $\gamma = 1.4$.

The high pressure high density state in $\Omega_1(t = 0) = [0; 1] \times [0; 3]$ is defined by $(\rho_1, p_1) = (1, 1)$, the low pressure high density state in $\Omega_2(t = 0) = [1; 7] \times [0; 1.5]$ is given by $(\rho_2, p_2) = (0.125, 0.1)$, the low pressure low density in $\Omega_3(t = 0) = [1; 7] \times [1.5; 3]$ is $(\rho_3, p_3) = (1.0, 0.1)$, see Figure 7. The initial gas is at rest and the final time of the simulation is $t_{\text{final}} = 5.5$.

The mesh is made by $N_E = 19098$ Delaunay triangles in such a way that it perfectly matches the discontinuities between the three subdomains. In Figure 8 are presented the density, specific internal energy, cell order and final mesh configuration. The vortex shape is clearly captured and the mesh seems to follow the flow field. This behavior is also illustrated in Figure 9 where we have colored the cells according to the initial subdomain index (1, 2 or 3). For a moving mesh method one expects that the mesh follows the flow with a quasi-vortex like velocity field. Nevertheless the numerical diffusion and rezone part of the ALE

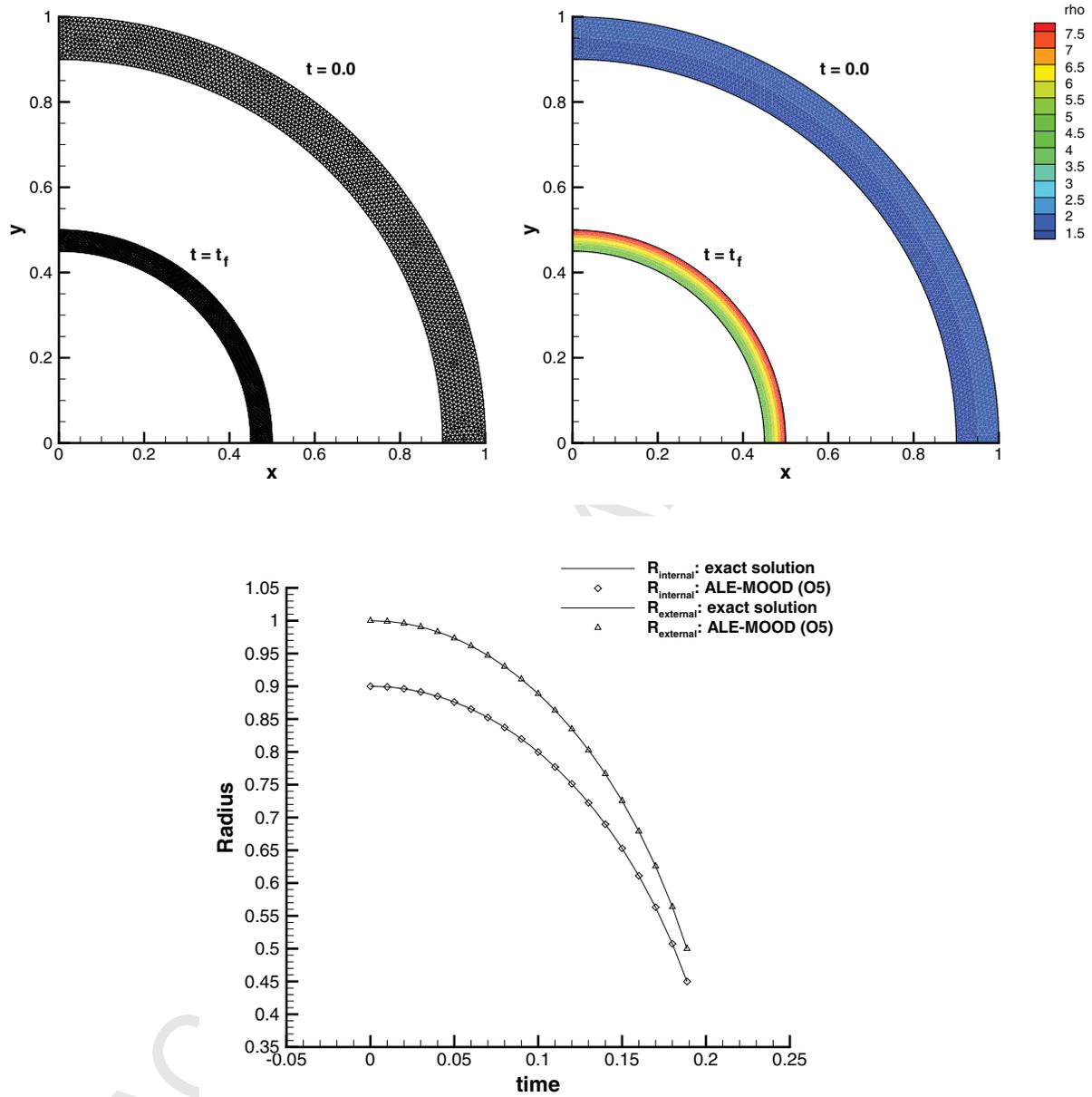


Figure 6: Kidder problem in 2D — ALE-MOOD- $\mathbb{P}_0\mathbb{P}_4$ polynomial reconstruction results — Top-left/right panels: mesh configuration (left) and density (right) of the shell at times $t = 0$ and at $t_{\text{final}} = \frac{\sqrt{3}}{2}\tau$. Bottom panel: evolution of the internal and external radius of the shell versus the analytical solution.

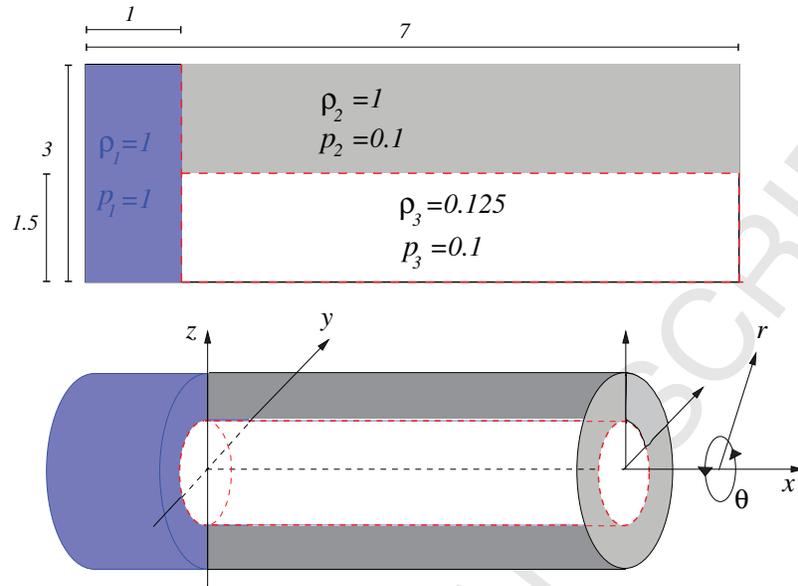


Figure 7: Sketch of the triple point problem initialization. Top: 2D setup — Bottom: 3D axis-symmetric version.

scheme can not allow such a vortex motion as mesh tangling will inexorably occur. This is why we can not observe in Figure 9 a perfect vortex shape. In Figure 10 we show the percentage of cells updated with \mathbb{P}_1^{lim}

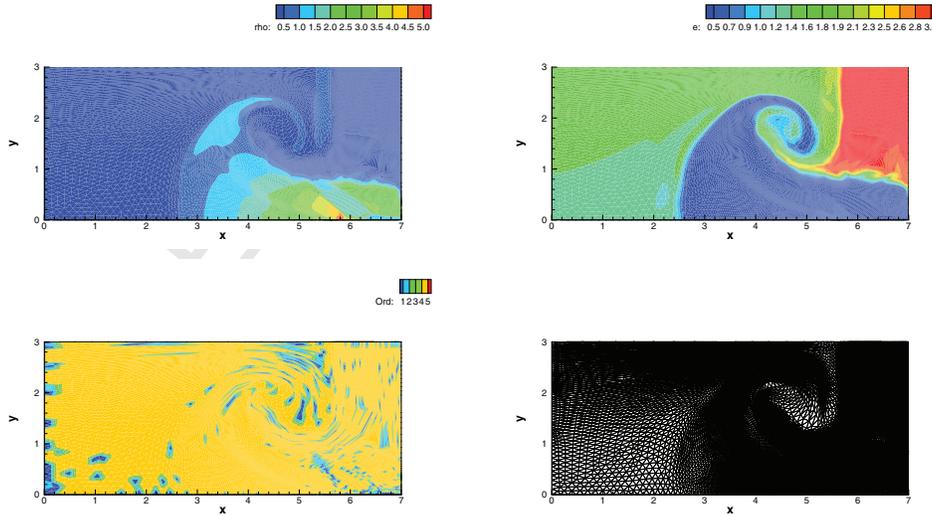


Figure 8: Triple point problem in 2D at $t_{final} = 5.5$ — ALE-MOOD- $\mathbb{P}_0\mathbb{P}_4$ scheme results — From top left to bottom right: cell density, cell specific internal energy, cell order and mesh configuration.

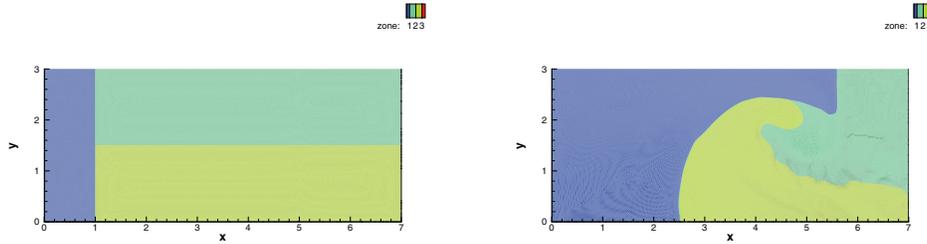


Figure 9: Triple point problem in 2D at $t = 0$ (left) and $t_{\text{final}} = 5.5$ (right) — ALE-MOOD- $\mathbb{P}_0\mathbb{P}_4$ scheme results. Cells colored corresponding to their initial subdomains.

(red) or \mathbb{P}_0 (blue) polynomial reconstructions as a function of the iteration number. For the first iterations about 30% of the cells are problematic and updated with \mathbb{P}_0 (blue) polynomial reconstructions. This number drops to 5% later in the computation. The percentage of cells updated with $\mathbb{P}_1^{\text{lim}}$ oscillates between 2% and 5%.

For comparison purposes we have run the ALE-MOOD- $\mathbb{P}_0\mathbb{P}_4$ scheme in its Eulerian regime by setting

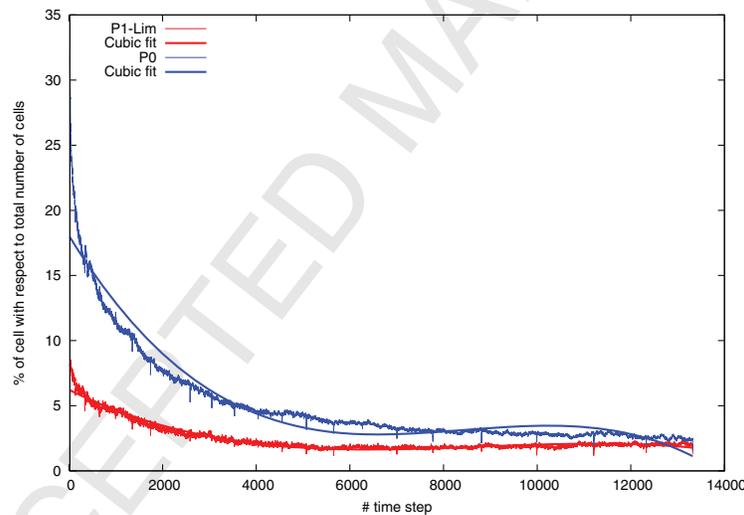


Figure 10: Triple point problem in 2D — ALE-MOOD- $\mathbb{P}_0\mathbb{P}_4$ scheme results — Percentage of cells updated with $\mathbb{P}_1^{\text{lim}}$ (red) or \mathbb{P}_0 (blue) schemes as a function of the iteration number and associated cubic fits. Only one sample point every 5 time steps is plotted for visualization purposes.

for the entire simulation the mesh velocity to zero, i.e. $\mathbf{V} = \mathbf{0}$. Figure 11 is meant to be compared with Figure 8. The mesh is not moving therefore we do not show it in this figure. We can validate the predictive capabilities of the numerical ALE-MOOD approach against the Eulerian regime, which is ultra robust (due to the fact that the mesh can not be tangled) but also more diffusive than the ALE configuration. Here we observe that the Eulerian algorithm produces good numerical results without being too diffusive even if the mesh does not follow the flow field. One reason is that for such a vortical motion a mesh moving scheme with a fixed connectivity is ultimately not able to displace the mesh along with the flow [86]. Then, sooner

or later, the ALE configuration becomes *de facto* an Eulerian one because of mesh stagnation due to the following two competing processes: (i) the vortical physical motion which tends to generate tangled mesh, and (ii) the counter-acting mesh relaxation. The CPU time and memory consumptions for the triple point

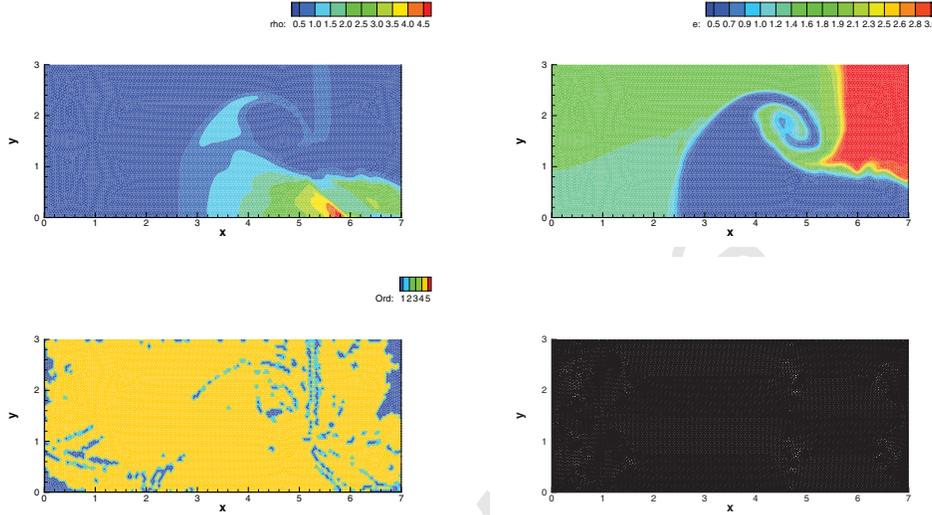


Figure 11: Triple point problem in 2D at $t_{\text{final}} = 5.5$ — ALE-MOOD- $\mathbb{P}_0\mathbb{P}_4$ scheme results run in an Eulerian regime on a fixed grid. From top left to bottom right: density, internal energy, cell order and the fixed Eulerian mesh.

problem can also be found in Table 3.

4.1.5. 2D Mesh tangling endurance problem

The aim of this test case is to highlight and clearly show one possible advantage coming from the use of an *a posteriori* approach w.r.t. classical *a priori* techniques. In fact using the MOOD paradigm we also have the possibility to check and control the quality of the mesh at each timestep, i.e. we can detect whether mesh tangling occurs, or not. The detection process is very easy, since we limit us to compute the volume of a cell using the *candidate* configuration of a cell at the new time level, $T_i^{n+1,*}$. To avoid mesh tangling, condition (24) must hold. If this is *not* satisfied, then the cell T_i is marked as problematic and during the next iteration of the MOOD loop (see Figure 2) we impose a relaxation coefficient of $\omega_k = 1$ in (12), so that the pure rezoned coordinates $\mathbf{X}_k^{\text{Rez}}$ are used as new coordinates \mathbf{X}_k^{n+1} for all vertices of cell T_i .

The initial computational domain for the 2D mesh tangling problem is the rectangle $\Omega(t=0) = [-2, 2] \times [-1, 1]$, which is filled with a perfect gas with $\gamma = 1.4$. The fluid is initially at rest and is assigned with an initial density and pressure of unity, i.e. $\rho_0 = 1$ and $p_0 = 1$. The mesh is composed of $N_E = 1816$ triangles with a characteristic mesh size of $h = 1/10$ and is depicted in Figure 12. Time-dependent boundaries are set everywhere: starting from time $t = 0$ on the left side of the domain we set the velocity $\mathbf{u}_L = (-1, 0)$ and on the right side we impose $\mathbf{u}_R = (1, 0)$, so that the mesh is stretched along the x -direction and rarefaction waves are generated. Up to time $t = \tilde{t} = 0.1$ the upper and lower boundaries are considered as walls, while for $t > \tilde{t}$ we prescribe the velocity for both the top and the bottom side, hence $\mathbf{u}_T = (-1, 0)$ and $\mathbf{u}_B = (1, 0)$, respectively. Such a motion is compressing the grid vertically, whose elements have already been stretched by the previous horizontal motion of the lateral boundaries. The final time of the simulation is set to $t_f = 0.9$ and the final computational domain is $\Omega(t = t_f) = [-2 - t_f, 2 + t_f] \times [-1 + (t_f - \tilde{t}), 1 - (t_f - \tilde{t})]$, as shown in Figure 12. We use the fourth order version of both ALE-ADER-MOOD and ALE-ADER-WENO schemes

with the Rusanov flux and the results are displayed in Figure 13. A comparison between WENO (left column) and MOOD (right column) is depicted at time $t_c = 0.75542$ before the WENO algorithm crashes due to mesh tangling which can already be seen on the zoom. We can notice that at this time most of the problematic cells detected by MOOD are the ones which have been highly stretched and compressed by the boundary motion.

Because the MOOD algorithm allows the mesh quality to be checked *a posteriori* and maintained throughout the simulation by local improvements with a proper use of the rezoning algorithm [62, 76, 18, 20], the simulation is performed until $t_f = 0.9$, see Figure 14. The density and cell orders are displayed (the final mesh can be seen on Figure 12). The numerical results are nicely symmetric along the x and y axes, as expected. The last panel in Figure 14 shows the distribution of problematic cells as a function of the timestep, in red/blue color the percentage of cells updated with $\mathbb{P}_1^{lim}/\mathbb{P}_0$ scheme and a cubic fit of the data. This endurance or fatigue test has been designed to be demanding for numerical methods, we observe that MOOD detection is active, roughly 6 to 15% of cells are updated with a low order scheme leading to a cumulative percentage of bad cells of about 10% up to 30%. Nonetheless, the MOOD algorithm is able to produce a valid answer.

In conclusion, in some cases illustrated by this mesh tangling test problem, an *a posteriori* treatment might lead to a more robust algorithm.

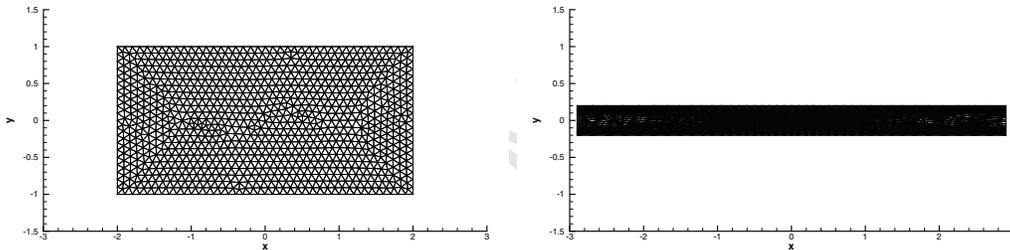


Figure 12: Mesh tangling problem in 2D - Initial mesh (left) and final mesh (right) obtained by the fourth order accurate ALE-ADER-MOOD scheme.

4.1.6. 2D CPU time and memory consumption

In this section we summarize the CPU time and memory consumption for the 2D tests by comparing ALE-WENO (see [20] for a full description) and ALE-MOOD both with $\mathbb{P}_0\mathbb{P}_3$ and $\mathbb{P}_0\mathbb{P}_4$. These schemes are of the same nominal order of accuracy and are implemented within the same framework. All the data listed in Table 3 have been collected running each of the test case using a single CPU core of an Intel i7-2600 processor with 3.4 GHz of clock speed and 16 GB of RAM, in order to assess the pure serial performance, without accounting for the MPI overhead.

From Table 3 we observe that the gain is systematically in favor of a MOOD approach in our ALE framework for these test cases. On average the acceleration is about 1.9 for $\mathbb{P}_0\mathbb{P}_4$ meaning that MOOD needs almost 2 times less CPU time than WENO for the same mesh configuration. This gain seems to be of the same order for the $\mathbb{P}_0\mathbb{P}_3$ schemes.

The saving in terms of memory is about 1.6 in favor of MOOD in 2D.

In [82] for a fixed grid Eulerian MOOD and WENO schemes the ratio was different, namely a bigger memory saving (ratio 3) and a smaller acceleration (1.3), both in favor of MOOD. Contrarily to an Eulerian context, here in our ALE framework we can not pre-compute and store all WENO reconstruction matrices; indeed they change at each time step according to the geometry evolution. Then in this ALE context more

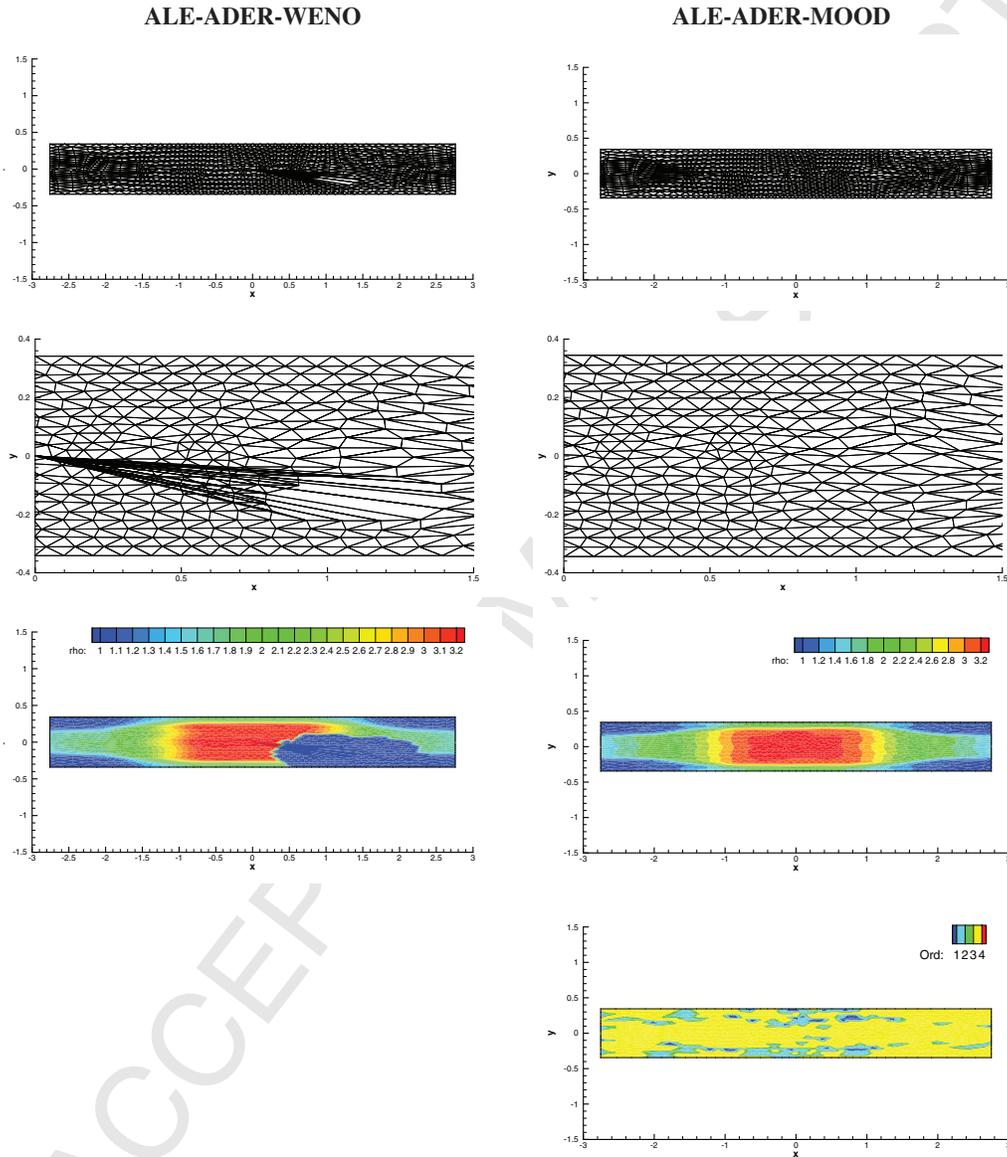


Figure 13: Mesh tangling problem in 2D at intermediate times $t_c = 0.75542$. Fourth order accurate numerical results obtained with ALE-ADER-WENO (left column) and ALE-ADER-MOOD (right column) schemes. From top to bottom: mesh configuration at the WENO crashing time and zoom on the tangled region, density distribution and cell order for MOOD results.

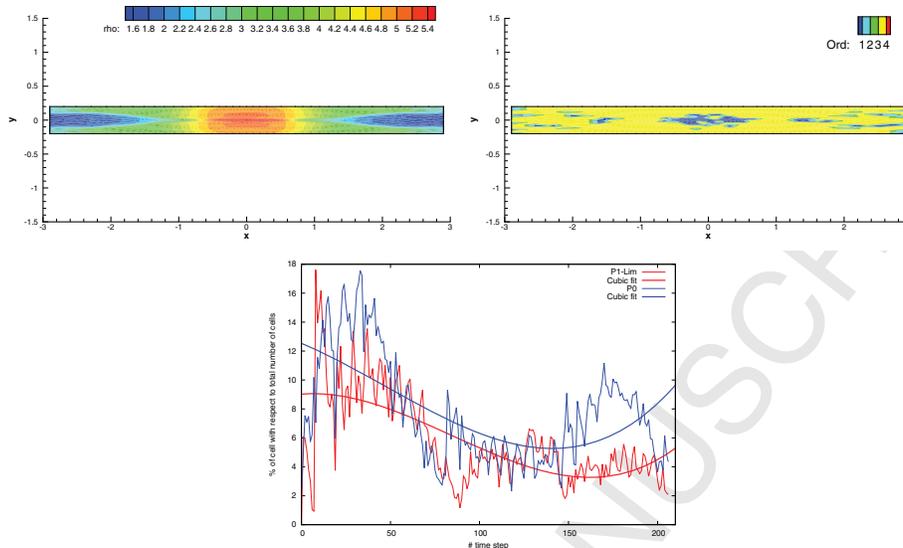


Figure 14: Mesh tangling problem in 2D at final time $t_f = 0.9$. Fourth order accurate numerical results obtained with ALE-ADER-MOOD schemes. Density distribution (top-left) and cell orders (top-right) at final time. The final mesh is shown in Figure 13. Bottom panel: Percentage of cells updated with \mathbb{P}_1^{lim} (red) or \mathbb{P}_0 (blue) schemes as a function of the iteration number and associated cubic fits.

computation and less storing is performed by WENO approach. On the other hand the MOOD approach, which always needs less reconstructions than WENO, needs less on-the-fly computations to solve the linear reconstruction systems (9).

This table only provides a general idea of the gain brought by the use of an *a posteriori* MOOD treatment in replacement of a WENO reconstruction and limiting. Both approaches have different user/developer parameters that may be ticked to improve the general efficiency. We would like to emphasize the fact that this comparison refers to our own implementation of both approaches in one and the same computer code.

4.2. 3D test problems

In this section we report the numerical results obtained for the 3D version of the test problems. Exhaustive 3D description of these tests can be found in the following references: the isentropic vortex problem [39], Sedov problem [85, 20], Saltzman problem [25, 22, 92, 20] and the triple point problem [42].

In 3D we have run all test cases in parallel using MPI on the massively parallel SuperMUC machine of the *Leibniz Rechenzentrum* (LRZ) in Munich, Germany. Therefore all CPU times reported in this section suffer from the MPI overhead and our own way of implementing the parallelization routines. As a consequence, they should be understood as a rough information rather than a genuine comparative measure of the efficiency between ALE-WENO and ALE-MOOD schemes. Contrarily the CPU times provided in the 2D section were all obtained with a dedicated serial machine, and, as such, represent a fair comparative measure.

4.2.1. 3D isentropic vortex problem

The 3D setup of the isentropic vortex is the same as in 2D (see section 4.1.1). The vortex convection velocity is $\mathbf{v} = (1, 1, 0)$ and the computational domain is $\Omega(t = 0) = [0; 10] \times [0; 10] \times [0; 5]$. Periodic

2D Tests	N_E	ALE-WENO- $\mathbb{P}_0\mathbb{P}_3$		ALE-MOOD- $\mathbb{P}_0\mathbb{P}_3$		Ratio WENO/MOOD	
		CPU time $\times 10^{-4}$ s	Memory MB	CPU time $\times 10^{-4}$ s	Memory MB	Acceleration	Saving
Sod	1800	1.1067	48.54	0.6352	30.31	1.83	1.60
Sedov	1800	2.2465	41.67	1.2774	27.41	1.76	1.52
Noh	3200	4.3273	61.24	2.6165	35.21	1.65	1.74
Kidder	3180	4.4953	61.83	2.0964	36.21	2.14	1.71
Saltzman	2000	3.6360	43.32	1.9137	26.72	1.90	1.62
Triple point	19098	21.10	236.80	12.83	156.80	1.65	1.51
Average						1.8	1.6

	N_E	ALE-WENO- $\mathbb{P}_0\mathbb{P}_4$		ALE-MOOD- $\mathbb{P}_0\mathbb{P}_4$		Ratio WENO/MOOD	
		CPU time $\times 10^{-4}$ s	Memory MB	CPU time $\times 10^{-4}$ s	Memory MB	Acceleration	Saving
Sod	1800	2.3018	62.97	1.1882	40.42	1.94	1.56
Sedov	1800	4.7396	53.76	2.7457	35.67	1.73	1.51
Noh	3200	8.6977	79.97	4.5598	47.96	1.91	1.67
Kidder	3180	8.9423	78.54	4.1643	47.53	2.15	1.65
Saltzman	2000	7.2889	56.23	3.5914	36.04	2.03	1.56
Triple point	19098	39.65	307.20	23.32	192.00	1.70	1.60
Average						1.9	1.6

Table 3: Summary of CPU time needed to update one cell within one timestep ($\times 10^{-4}$ in seconds) and memory consumption (in MB) for the two dimensional test cases of this paper. Top-lines: ALE-WENO- $\mathbb{P}_0\mathbb{P}_3$ versus ALE-MOOD- $\mathbb{P}_0\mathbb{P}_3$ schemes run on the same mesh. Bottom-lines: ALE-WENO- $\mathbb{P}_0\mathbb{P}_4$ versus ALE-MOOD- $\mathbb{P}_0\mathbb{P}_4$ schemes. The last two columns show the ratio between the CPU times (“acceleration”) and the memory consumption (“saving”).

boundary conditions are imposed everywhere. Successively refined 3D tetrahedral meshes are employed and the errors in L_2 and L_∞ norms are computed to get the actual convergence rates of the numerical schemes.

In Table 4 we report the errors and the corresponding rates of convergence for ALE-MOOD and ALE-WENO with $\mathbb{P}_0\mathbb{P}_3$ and $\mathbb{P}_0\mathbb{P}_4$ reconstructions. We observe that the nominally expected rates of convergence in 3D on unstructured meshes are almost reached for both methods. We also report the unlimited $\mathbb{P}_0\mathbb{P}_3$ and $\mathbb{P}_0\mathbb{P}_4$ scheme results. This provides the reference solution in terms of accuracy and CPU time. As in 2D (but in a more pronounced way) the MOOD approach seems to produce more accurate results than the WENO approach. The reason being that no cell is ever detected as problematic by the MOOD process for this problem. As a consequence the unlimited and MOOD results are basically the same. Therefore it is not surprising to observe an optimal ratio of 1 for the CPU time between the ALE-MOOD and the unlimited schemes, whereas this ratio is about 2 to 4 for the ALE-WENO scheme. This means that the ALE-WENO approach for this test problem is about 2 to 4 times more expensive than the ALE-MOOD scheme. Next in Figure 15 we graphically represent in the left panel the convergence rates for the L_2 norm in log scale versus the expected 4th and 5th order lines, the data are taken from Table 4. In the right panel we also show the CPU time as a function of the initial characteristics mesh size h for ALE-WENO- $\mathbb{P}_0\mathbb{P}_3/\mathbb{P}_0\mathbb{P}_4$ and ALE-MOOD- $\mathbb{P}_0\mathbb{P}_3/\mathbb{P}_0\mathbb{P}_4$ schemes. These figures show a better general behavior of the ALE-MOOD approach both in terms of accuracy/convergence (left panel) and performance (right panel).

4.2.2. 3D Sedov problem

Next we consider the spherically symmetric Sedov problem, which describes the evolution of a diverging shock wave generated at the origin of the initial computational domain given by one eighth of

3D isentropic vortex problem — Unlimited — ALE-WENO — ALE-MOOD								
	N_E	h	L_2 error	L_∞ error	L_2 order	L_∞ order	Theor.	CPU time
Unlimited- $\mathbb{P}_0\mathbb{P}_3$	60157	2.89E-01	4.12E-03	3.53E-03	—	—	4	1.0
	137317	2.17E-01	1.37E-03	1.34E-03	3.8	3.3		1.0
	431462	1.52E-01	3.27E-04	3.71E-04	4.1	3.6		1.0
	1052264	1.13E-01	9.80E-05	1.06E-04	4.1	4.2		1.0
ALE-WENO- $\mathbb{P}_0\mathbb{P}_3$	60157	2.89E-01	1.51E-02	2.89E-01	—	—	4	2.90
	137317	2.17E-01	2.78E-03	8.96E-03	5.8	12.0		2.80
	431462	1.52E-01	3.58E-04	1.37E-03	5.8	5.3		2.47
	1052264	1.13E-01	9.87E-05	1.95E-04	4.3	6.6		1.91
ALE-MOOD- $\mathbb{P}_0\mathbb{P}_3$	60157	2.89E-01	4.11E-03	3.53E-03	—	—	4	1.03
	137317	2.17E-01	1.37E-03	1.34E-03	3.8	3.3		1.03
	431462	1.52E-01	3.27E-04	3.70E-04	3.9	3.5		1.03
	1052264	1.13E-01	9.80E-05	1.05E-04	4.1	4.3		1.03
Unlimited- $\mathbb{P}_0\mathbb{P}_4$	60157	2.89E-01	2.27E-03	1.69E-03	—	—	5	1.0
	137317	2.17E-01	6.60E-04	7.68E-01	4.3	2.8		1.0
	431462	1.52E-01	1.23E-04	1.76E-04	4.8	4.1		1.0
	1052264	1.13E-01	2.93E-05	4.79E-05	4.8	4.4		1.0
ALE-WENO- $\mathbb{P}_0\mathbb{P}_4$	60157	2.89E-01	1.42E-02	6.46E-02	—	—	5	4.21
	137317	2.17E-01	2.12E-03	2.45E-02	6.6	3.4		4.17
	431462	1.52E-01	1.46E-04	1.11E-03	7.6	8.8		2.94
	1052264	1.13E-01	3.10E-05	2.22E-05	5.2	5.4		2.90
ALE-MOOD- $\mathbb{P}_0\mathbb{P}_4$	60157	2.89E-01	2.27E-03	1.69E-03	—	—	5	1.03
	137317	2.17E-01	6.61E-04	7.67E-05	4.6	4.9		1.03
	431462	1.52E-01	1.24E-04	1.76E-04	4.3	3.7		1.03
	1052264	1.13E-01	2.97E-05	4.82E-05	5.0	4.8		1.03

Table 4: L_2 and L_∞ errors and convergence rates and CPU time for the 3D isentropic vortex problem for the Unlimited, ALE-WENO and ALE-MOOD schemes for $\mathbb{P}_0\mathbb{P}_3$ and $\mathbb{P}_0\mathbb{P}_4$ polynomial reconstructions from top to bottom.

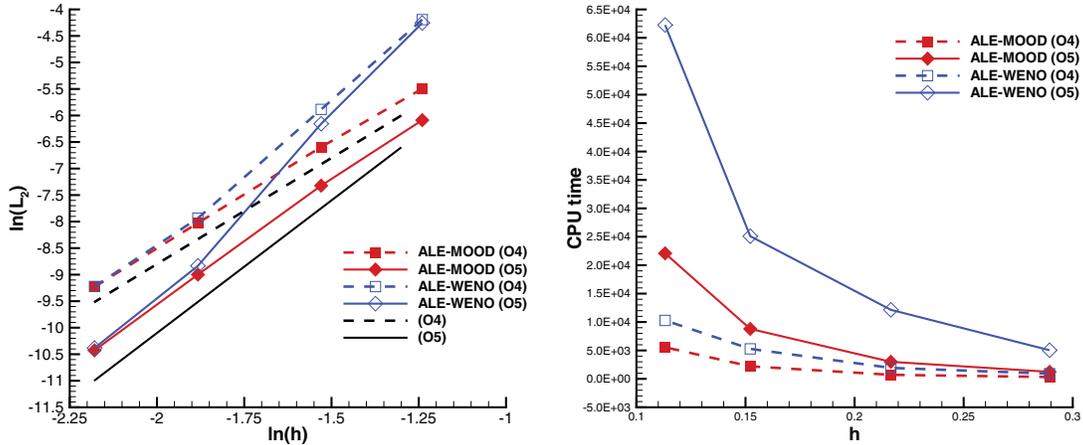


Figure 15: 3D isentropic vortex for ALE-WENO and ALE-MOOD with $\mathbb{P}_0\mathbb{P}_3$ and $\mathbb{P}_0\mathbb{P}_4$ polynomial reconstructions. Left panel: convergence rate for the L_2 norm in log scale versus expected 4th and 5th order lines — Right panel: CPU time in seconds as a function of the final cell characteristics length h .

$\Omega = [0.0; 1.2]^3$. This is a well-known test case for moving mesh schemes [110, 87, 92, 85] that measures the preservation of spherical symmetry on non symmetric grid. An analytical solution based on self-similarity arguments is available, see Kamm et al. [71]. For the 3D Sedov problem a mesh is constructed with $N^3 = 40^3$ hexahedra which are further split into 5 tetrahedra leading to a total number of elements $N_E = 32 \times 10^4$. Wall boundaries have been set everywhere. The computational domain is filled with a perfect gas with $\gamma = 1.4$, which is initially at rest and is assigned with a uniform density $\rho_0 = 1$. The total energy of the explosion is concentrated at the origin and has magnitude $E_{\text{total}} = 0.851072$ similar to [103]. Numerically E_{tot} is concentrated only in the cells containing the origin \mathbf{O} , therefore the initial pressure is given by

$$p_{or} = (\gamma - 1)\rho_0 \frac{E_{\text{tot}}}{8 \cdot V_{or}}, \quad (31)$$

where V_{or} is the volume of the region which is composed by five tetrahedra and the factor $\frac{1}{8}$ takes into account the spherical symmetry. At the final time $t_{\text{final}} = 1.0$ the exact location of the spherically symmetric diverging shock wave is at radius $r = \sqrt{x^2 + y^2 + z^2} = 1$ with a density peak $\rho = 6.0$. The numerical results have been obtained using the ALE-MOOD- $\mathbb{P}_0\mathbb{P}_4$ version of the method.

In Figure 16 the final mesh and density (top panels), the cell orders (bottom-left panel) and the density as a function of cell radius for all cells (bottom-right panel) are presented. The exact solution has been obtained thanks to Kamm [72]. The mesh and the density seem to preserve well the sphericity of the shock wave. We observe that the mesh is highly compressed after the shock wave and expanded close to the origin as expected. The preservation of spherical symmetry is furthermore evident on the plot of the density as a function of cell radius as the numerical solution matches almost perfectly the exact solution, meaning that all cells at a given radius r have the same density.

On the figure of cell orders (cell polynomial degrees plus one) we can also observe that few cells are actually problematic and they are located on the shock front. This implies that most of the cells are updated with the unlimited scheme. To emphasize the gain in accuracy we compare these results with the ALE-WENO- $\mathbb{P}_0\mathbb{P}_4$ ones obtained with the algorithm described in [20] in Figure 17. The density as a function of cell radius for

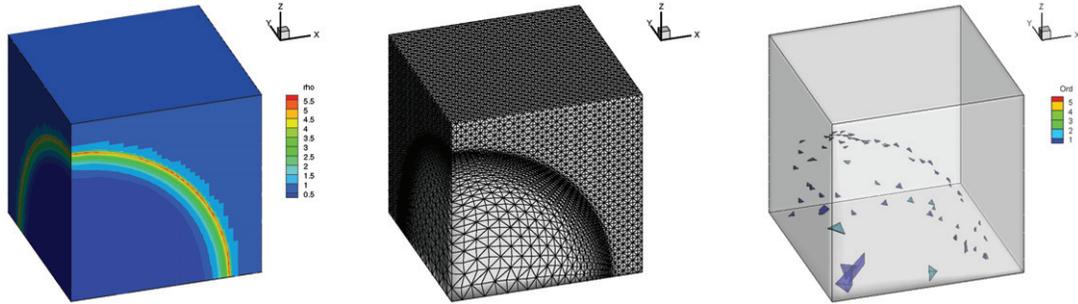


Figure 16: Sedov problem in 3D at $t_{\text{final}} = 1.0$ — ALE-MOOD- $\mathbb{P}_0\mathbb{P}_4$ results — Density, mesh and cell order.

all cells is reported for both schemes against the exact solution in red line. One can clearly notice that the density peak is better retrieved by the MOOD approach. We have plotted in Figure 18 the percentage of cells

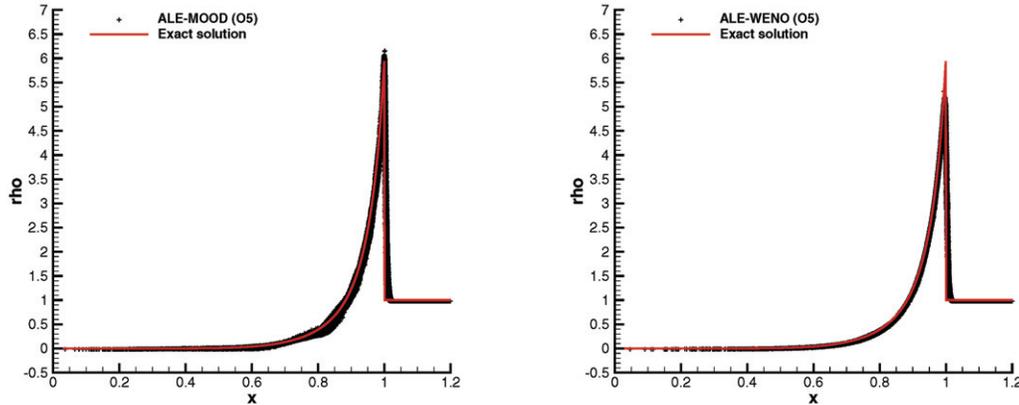


Figure 17: Sedov problem in 3D at $t_{\text{final}} = 1.0$ — ALE-MOOD- $\mathbb{P}_0\mathbb{P}_4$ results (left) and ALE-WENO- $\mathbb{P}_0\mathbb{P}_4$ results (right) — Density distribution as a function of cell radius for all cells vs the exact solution (red line).

updated with $\mathbb{P}_1^{\text{lim}}$ (red) or \mathbb{P}_0 (blue) polynomial reconstructions as a function of the iteration number. We deduce that, for this problem, very few cells (at most 50) are detected as problematic during one timestep, hence explaining the very low percentage of problematic cells in the graphic. This also means that most of the cells are updated with unlimited high order accurate polynomial reconstructions. Last a comparison of CPU time and memory consumptions between the ALE-MOOD and ALE-WENO approaches is reported in Table 5.

4.2.3. 3D Saltzman problem

In this problem [106, 45] a domain $\Omega(t = 0) = [0; 1] \times [0; 0.1] \times [0; 0.1]$ is filled with a perfect gas ($\gamma = 5/3$) at rest with data $\rho = 1$, $p = \epsilon$ where $\epsilon = 10^{-4}$ and the final time is set to $t_{\text{final}} = 0.6$ [81]. All boundaries are perfect walls, apart from the left boundary face which is a right-moving piston with velocity $\mathbf{v} = (1, 0, 0)^t$. In the initial time steps the scheme must obey a geometric CFL condition, i.e. the

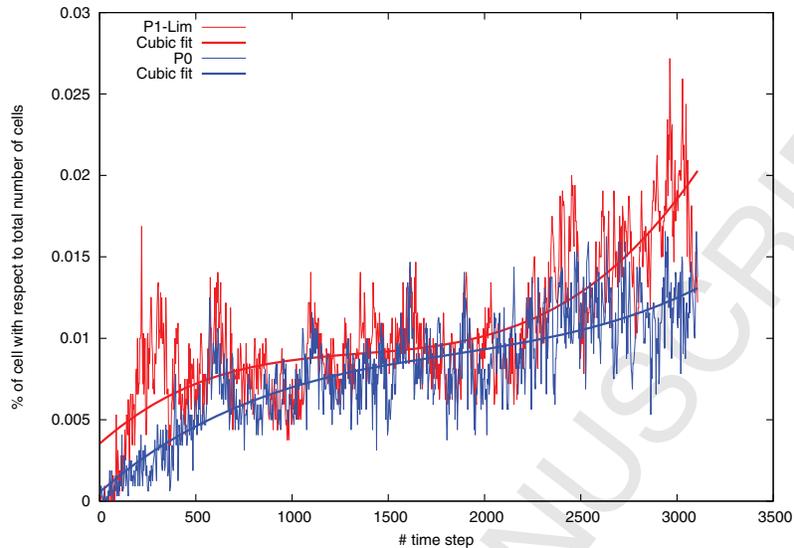


Figure 18: Sedov problem in 3D — ALE-MOOD with \mathbb{P}_4 polynomial reconstruction results — Percentage of cells updated with \mathbb{P}_1^{lim} (red) or \mathbb{P}_0 (blue) reconstructions as a function of iteration number.

piston must not move more than one element per time step. The piston sends a straight shock wave into the computational domain. This shock wave ultimately bounces onto the fixed right wall and onto the piston back and forth.

An exact solution is defined by the value of the plateaus behind the shock, [16, 120], it reads

$$\mathbf{Q}_{ex}(\mathbf{x}, t_{\text{final}}) = \begin{cases} (4, 4, 0, 0, 4) & \text{if } x \leq x_f, \\ (1, 0, 0, 0, \epsilon) & \text{if } x > x_f, \end{cases} \quad (32)$$

where $x_f = 0.8$ is the shock location at time $t_{\text{final}} = 0.6$.

This problem usually tests the robustness of Lagrangian numerical methods by using the Saltzman mesh which is built according to the procedure originally given in [106, 25, 24, 92] and adapted to tetrahedra in [20]. The domain is initially meshed with a uniform Cartesian grid composed by $100 \times 10 \times 10$ cubic elements which are further split into five tetrahedra. We finally use the mapping given in [25, 92] to transform this uniform grid, defined by the coordinate vector $\mathbf{x} = (x, y, z)$, to the skewed Saltzman-like configuration $\mathbf{x}' = (x', y', z')$:

$$\begin{aligned} x' &= x + (0.1 - z)(1 - 20y) \sin(\pi x) & \text{for } 0 \leq y \leq 0.05, \\ x' &= x + z(20y - 1) \sin(\pi x) & \text{for } 0.05 < y \leq 0.1, \end{aligned}$$

and $y' = y, z' = z$. The domain is finally discretized with a total number of $N_E = 50000$ tetrahedra and the final time is $t_{\text{final}} = 0.6$.

For this test case the ALE-MOOD- $\mathbb{P}_0\mathbb{P}_4$ version of the method is adopted.

In Figure 19 we present the final grid and cell order (top line). The original skewness of the mesh has not generated any spurious oscillations or lack of symmetry and the mesh is nicely shaped. Moreover the cell orders are almost at maximal value and few cells are decremented, meaning that the 5th order reconstruction is used almost everywhere. The bottom panels present the density (left) and x component of the velocity (right) as a function of x for all cells versus the exact solution (red line). Apart from the

classical wall heating effect close to the moving boundary condition on the right of the figure and a slight overshoot after the shock wave, the results are in good agreement with the exact solution. We have plotted

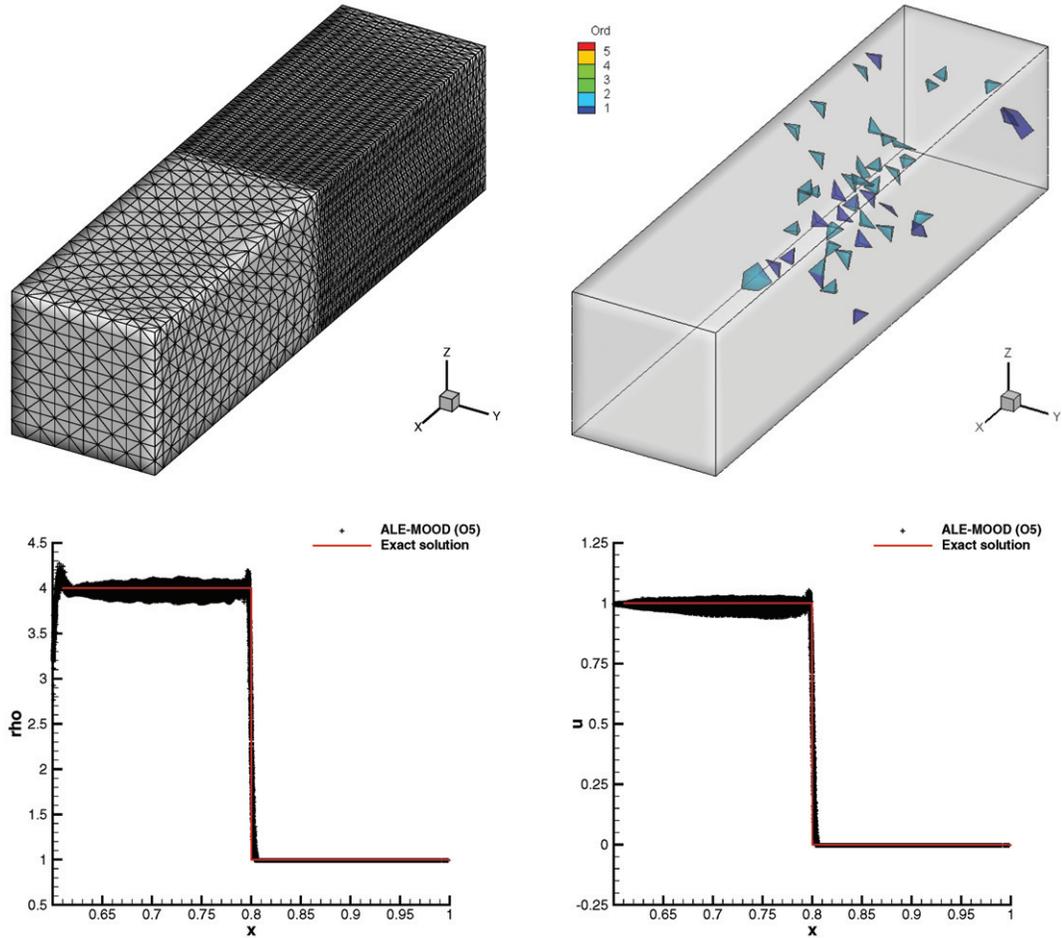


Figure 19: Saltzman problem in 3D at $t_{\text{final}} = 0.6$ — ALE-MOOD with \mathbb{P}_4 polynomial reconstruction results — Top: final mesh and cell orders — Bottom: density/ x component of the velocity as a function of x for all cells.

in Figure 20 the percentage of cells updated with $\mathbb{P}_1^{\text{lim}}$ (red) or \mathbb{P}_0 (blue) polynomial reconstructions as a function of the iteration number. We deduce that, for this problem very few cells are updated using $\mathbb{P}_1^{\text{lim}}$ or \mathbb{P}_0 reconstructions, reaching at most 100 cells. In Table 5 we report the comparison of CPU time and memory consumptions between the ALE-MOOD and ALE-WENO schemes.

4.2.4. 3D mono-material triple point problem

The 3D triple point problem [40, 42] is constructed by considering the 2D setup from section 4.1.4 as a 2D axi-symmetric r, z setup, see Figure 7. Consequently subdomain #3 becomes an internal cylinder to cylindrical subdomain #2. Subdomain #1 (blue colored) can be seen as a cylindrical cap to these embedded cylinders. The grid is made by meshing the three subdomains by exactly matching the interfaces in between

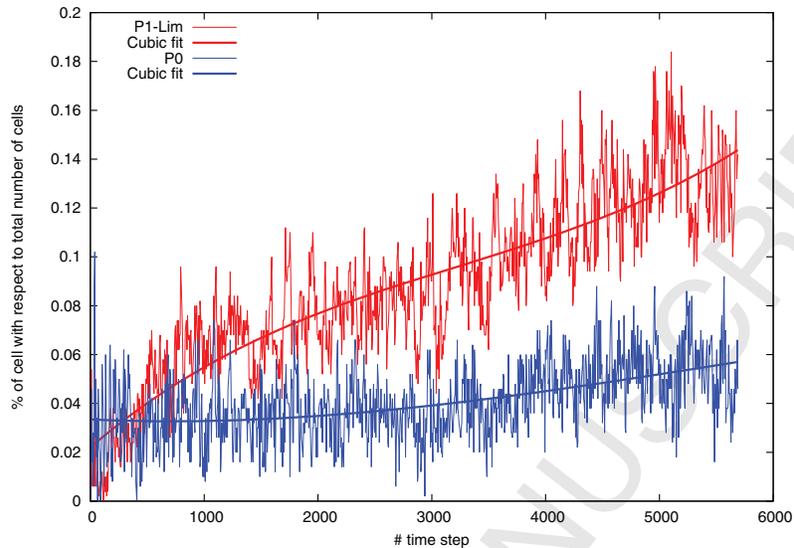


Figure 20: Saltzman problem in 3D — ALE-MOOD with \mathbb{P}_4 polynomial reconstruction results — Percentage of cells updated with \mathbb{P}_1^{lim} (red) or \mathbb{P}_0 (blue) reconstructions as a function of iteration number.

them. It is constituted by $N_E = 304246$ tetrahedra. The ALE-MOOD- $\mathbb{P}_0\mathbb{P}_4$ version is used to simulate this problem up to final time $t_{\text{final}} = 5.5$. In Figures 21 and 22 we present the specific internal energy (top), internal views of the mesh (bottom-left) and of the cell orders (bottom-right) at final time. Two 3D views of the specific internal energy are displayed: one on the left showing the boundary faces, the second one, on the right, the resulting configuration for all cells located below the plane $z = x$. The mesh and cell order figures adopt the same view. Figure 21 presents the results obtained when a strong mesh relaxation is applied fixing the relaxation factor $\omega_k = 0.9$ in (12), leading to an almost Eulerian simulation. Contrarily by using the automatic relaxation algorithm of Galera et al. [62], we collect the results displayed in Figure 22 where the mesh is clearly following the flow field more strictly compared to the previous results. Nonetheless the specific internal energy results are comparable between these two runs. In both cases the final cell orders reach the maximal value for almost all cells.

At last we present the percentage of cells updated with \mathbb{P}_1^{lim} (red) or \mathbb{P}_0 (blue) reconstructions as a function of iteration number. Left panel presents the first 100 iterations whereas the right panel displays the remaining iterations along with cubic fits of the data in straight lines. From these data we observe that for both simulations, for any timestep, less than 1% of cells are decremented, apart from the first 10 iterations when the shocks emanate from the discontinuities. Finally Table 5 presents the comparison of CPU time and memory consumptions between the ALE-MOOD and ALE-WENO schemes. This problem is a difficult one for moving mesh schemes with fixed connectivity, as the vortex motion of the flow implies that sooner or later invalid or highly stretched elements will appear. At this point, to avoid the failure of the code, a stronger mesh relaxation is mandatory leading to a more Eulerian-like simulation.

4.2.5. 3D CPU time and memory consumption

In this section we summarize the CPU time and memory consumption for the 3D tests by comparing ALE-WENO and ALE-MOOD with $\mathbb{P}_0\mathbb{P}_4$ polynomial reconstructions. These schemes are of the same nominal order of accuracy and are implemented within the same framework. All data listed in Table 5 have been collected running each of the test problems in parallel on 1024 processors. The information provided

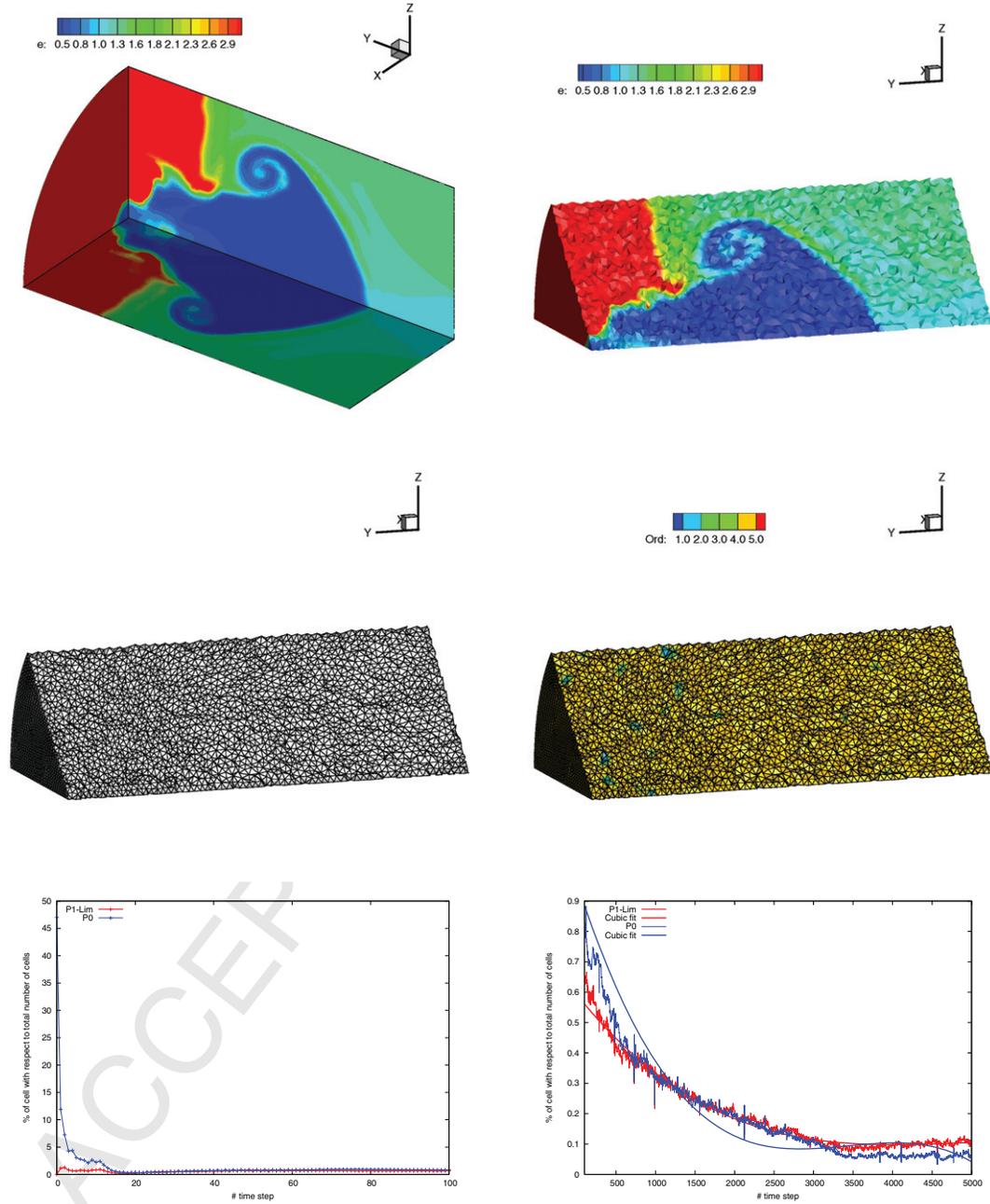


Figure 21: 3D Mono-material triple point problem — ALE-MOOD- $\mathbb{P}_0\mathbb{P}_4$ results using the relaxation parameter $\omega_k = 0.9$ — Top: specific internal energy — Middle: mesh and cell orders — Bottom: percentage of cells updated with \mathbb{P}_1^{lim} (red) or \mathbb{P}_0 (blue) reconstructions as a function of iteration number. Left panel presents the first 100 iterations whereas the right panel displays the remaining iterations along with cubic fits of the data in straight lines.

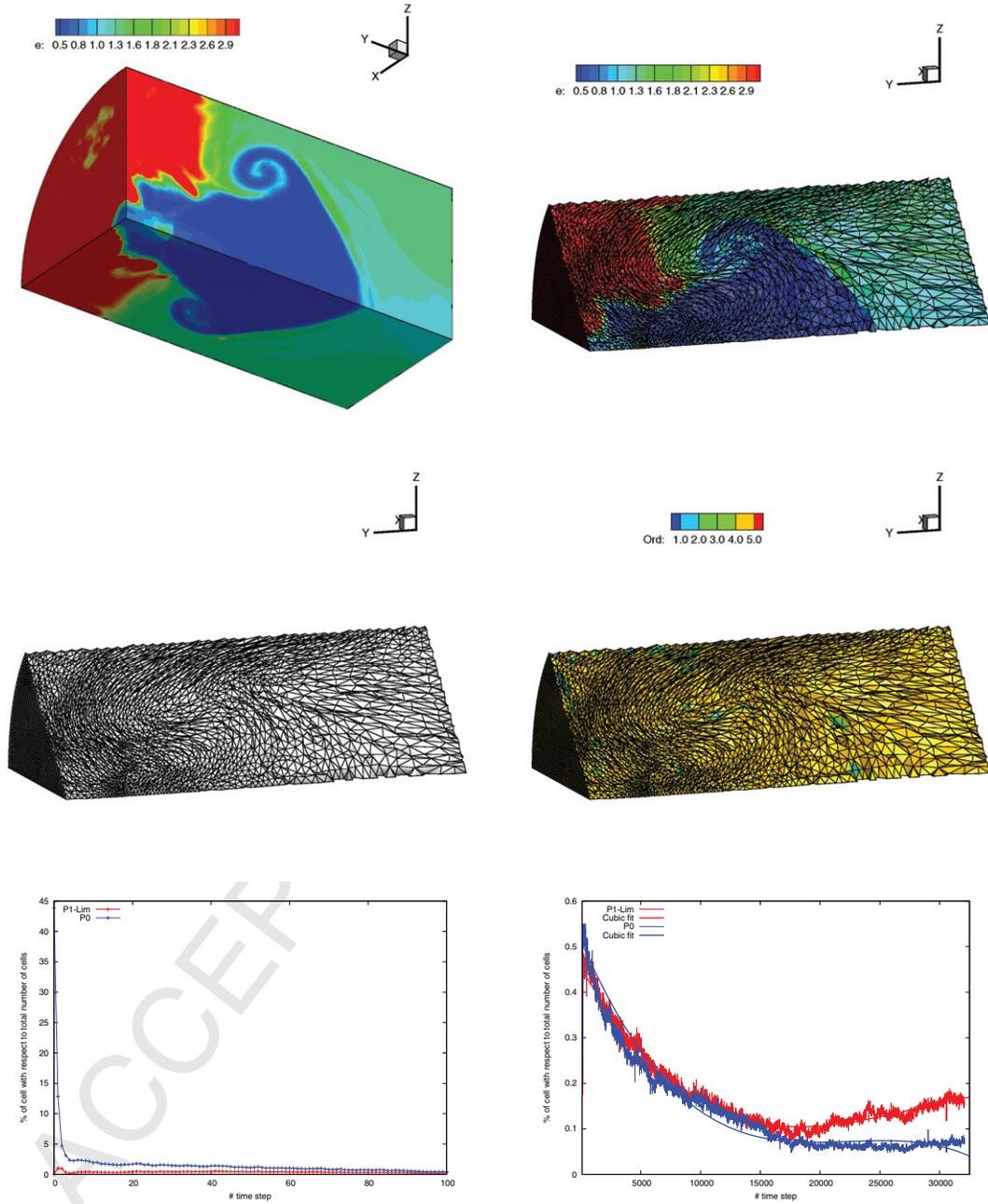


Figure 22: 3D Mono-material triple point problem — ALE-MOOD- $\mathbb{P}_0\mathbb{P}_4$ results using the automatic relaxation parameter algorithm — Top: specific internal energy — Middle: mesh and cell orders — Bottom: percentage of cells updated with \mathbb{P}_1^{lim} (red) or \mathbb{P}_0 (blue) reconstructions as a function of iteration number. Left panel presents the first 100 iterations whereas the right panel displays the remaining iterations along with cubic fits of the data in straight lines.

in this table is consequently affected by the MPI overhead.

We present the CPU time needed to update one cell within one timestep ($\times 10^{-2}$ in seconds) and memory consumption in MB. The last two columns show the ratio between WENO and MOOD for the CPU times (“acceleration”) and the memory consumption (“saving”). From these data we can conclude that, in 3D, our implementation of ALE-ADER-MOOD is on average 2.6 times faster than the ALE-ADER-WENO version. Moreover the factor of memory savings is about 1.8.

3D Tests	N_E $\times 10^4$	ALE-WENO- $\mathbb{P}_0\mathbb{P}_4$		ALE-MOOD- $\mathbb{P}_0\mathbb{P}_4$		Ratio WENO/MOOD	
		CPU time $\times 10^{-2}s$	Memory $\times 10^4MB$	CPU time $\times 10^{-2}s$	Memory $\times 10^4MB$	Acceleration	Saving
Sod	7.1	37.89	1.55	15.16	0.85	2.50	1.83
Sedov	32.0	7.67	2.59	3.26	1.52	2.35	1.70
Noh	32.0	6.83	2.55	3.12	1.54	2.19	1.66
Kidder	11.2	7.92	1.84	2.19	0.98	3.64	1.87
Saltzman	5	10.53	0.050	4.14	0.027	2.54	1.85
Triple point	30.4	34.10	3.28	14.09	1.94	2.42	1.69
Average						2.6	1.8

Table 5: Summary of CPU time needed to update one cell within one timestep ($\times 10^{-2}$ in seconds) and memory consumption (in MB) for the three dimensional test cases of this paper. ALE-WENO- $\mathbb{P}_0\mathbb{P}_4$ versus ALE-MOOD- $\mathbb{P}_0\mathbb{P}_4$ schemes. The last two columns show the ratio between the CPU times (“acceleration”) and the memory consumption (“saving”).

5. Conclusion and Perspectives

In this paper we have applied the novel *a posteriori* Multi-dimensional Optimal Order Detection (MOOD) approach to the context of direct Arbitrary-Lagrangian-Eulerian (ALE) ADER schemes solving the hydrodynamics equations in multi-dimensions. Starting from an unlimited ALE-ADER scheme previously designed in [20] to solve 3D hydrodynamics problems on unstructured tetrahedral grids, we have replaced the *a priori* WENO polynomial limiting technique by the *a posteriori* MOOD paradigm [29, 38, 39, 82]. In this framework MOOD can be thought as an *a posteriori* approach to the problem of limiting high order accurate polynomial reconstructions. Each candidate solution is computed using polynomial reconstructions given by a cell polynomial degree map initially set to a maximal degree $M > 0$. Next, the obtained unlimited candidate solution at t^{n+1} is tested against user given detection criteria which determine whether a numerical cell centered value is admissible or not. For instance the Physical Admissible Detection criteria (PAD) for the Euler equations contain the positivity of density and specific internal energy, whereas the Numerical Admissible Detection criteria (NAD) ensure the essentially non oscillatory behavior of the solution as well as the local mesh admissibility. If a cell of the candidate solution is detected as problematic, it is re-updated starting again at t^n after decrementing the polynomial degree of the reconstructions according to the previously fixed cascade of schemes. In this paper we have consistently chosen the following cascade: $\mathbb{P}_M \rightarrow \mathbb{P}_1^{LM} \rightarrow \mathbb{P}_0$.

This iterative process stops either when all cells have an admissible numerical solution, or, when the polynomial degree has reached the last value corresponding to the parachute bulletproof scheme. Here we have chosen a first order Godunov-type scheme that adopts \mathbb{P}_0 reconstructions.

These ALE-MOOD schemes have been implemented within a 2D/3D MPI parallel code dedicated to solve hyperbolic partial differential equations on unstructured grids, see [16, 48, 20]. A large set of tests in 2D and 3D have been simulated and results are reported in this work. The ALE-MOOD numerical method produces accurate results on these test problems. An evolution of the number of problematic cells is systematically provided in order to visualize the extra-work brought by the use of the MOOD paradigm. Indeed, for these test cases, only few cells are problematic at each time step leading, as a side effect, to large savings in CPU

time and memory. To estimate these savings we have compared the efficiency of our MOOD approach with the previous approach using WENO reconstructions [20]. For all test cases, in 2D and 3D, the CPU time and memory consumption have been reported and these data show that the MOOD approach in this direct ALE context is about 1.8 times faster in 2D (2.6 in 3D) and saves about 38% of memory in 2D (45% in 3D) for, at *minima*, the same accuracy. **Moreover we have shown on a 2D endurance test case that the MOOD approach can reach the prescribed final time while our implementation of ALE with WENO reconstruction is not able to maintain the mesh integrity, hence leading to a severe code failure. The *a posteriori* mesh validity check along with the possibility to recompute some bad cells with a more viscous scheme, *de facto* increases the range of robustness of the MOOD approach.**

In the future we plan to investigate the extension of the ALE-MOOD approach to more complex systems of conservation laws involving stiff source terms and non-conservative products such as multi-fluid [1] and multi-phase [104] models, ideal and relativistic magneto-hydrodynamics models [64, 68, 100, 128, 2], or hyperbolic models of nonlinear elasticity and elasto-plasticity [125, 63, 65, 115].

Acknowledgments

M.D. and W.B. have been financed by the European Research Council (ERC) under the European Union’s Seventh Framework Programme (FP7/2007-2013) with the research project *STiMulUs*, ERC Grant agreement no. 278267. R.L. has been partially funded by the ANR under the JCJC project “ALE INC(ubator) 3D”. The authors would like to acknowledge PRACE for awarding access to the SuperMUC supercomputer based in Munich, Germany at the Leibniz Rechenzentrum (LRZ). Parts of the material contained in this work have been elaborated, gathered and tested while W.B. visited the Mathematical Institute of Toulouse for three months.

- [1] M.R. Baer and J.W. Nunziato. A two-phase mixture theory for the deflagration-to-detonation transition (DDT) in reactive granular materials. *J. Multiphase Flow*, 12:861–889, 1986.
- [2] D. Balsara. Total variation diminishing scheme for relativistic magneto-hydrodynamics. *The Astrophysical Journal Supplement Series*, 132:83–101, 2001.
- [3] D. Balsara and C.W. Shu. Monotonicity preserving weighted essentially non-oscillatory schemes with increasingly high order of accuracy. *Journal of Computational Physics*, 160:405–452, 2000.
- [4] D.S. Balsara. Multidimensional HLLC Riemann solver: Application to Euler and magnetohydrodynamic flows. *Journal of Computational Physics*, 229:1970–1993, 2010.
- [5] D.S. Balsara. Self-adjusting, positivity preserving high order schemes for hydrodynamics and magnetohydrodynamics. *Journal of Computational Physics*, 231:7504–7517, 2011.
- [6] D.S. Balsara. A two-dimensional HLLC Riemann solver for conservation laws: Application to Euler and magnetohydrodynamic flows. *Journal of Computational Physics*, 231:7476–7503, 2012.
- [7] D.S. Balsara. Multidimensional Riemann problem with self-similar internal structure. Part I – Application to hyperbolic conservation laws on structured meshes. *Journal of Computational Physics*, 277:163–200, 2014.
- [8] D.S. Balsara and M. Dumbser. Multidimensional Riemann problem with self-similar internal structure. Part II – Application to hyperbolic conservation laws on unstructured meshes. *Journal of Computational Physics*, 2015. in press: DOI:10.1016/j.jcp.2014.11.004.
- [9] D.S. Balsara, M. Dumbser, and R. Abgrall. Multidimensional HLLC Riemann Solver for Unstructured Meshes - With Application to Euler and MHD Flows. *Journal of Computational Physics*, 261:172–208, 2014.
- [10] T.J. Barth and P.O. Frederickson. Higher order solution of the euler equations on unstructured grids using quadratic reconstruction. *28th Aerospace Sciences Meeting*, pages AIAA paper no. 90-0013, January 1990.
- [11] T.J. Barth and D.C. Jespersen. The design and application of upwind schemes on unstructured meshes. *AIAA Paper 89-0366*, pages 1–12, 1989.
- [12] D. J. Benson. Momentum advection on a staggered mesh. *Journal of Computational Physics*, 100(1):143 – 162, 1992.
- [13] M. Berndt, J. Breil, S. Galera, M. Kucharik, P.H. Maire, and M. Shashkov. Two-step hybrid conservative remapping for multimaterial arbitrary Lagrangian-Eulerian methods. *Journal of Computational Physics*, 230:6664–6687, 2011.
- [14] P. Bochev, D. Ridzal, and M.J. Shashkov. Fast optimization-based conservative remap of scalar fields through aggregate mass transfer. *Journal of Computational Physics*, 246:37–57, 2013.
- [15] W. Boscheri, D.S. Balsara, and M. Dumbser. Lagrangian ADER-WENO Finite Volume Schemes on Unstructured Triangular Meshes Based On Genuinely Multidimensional HLL Riemann Solvers. *Journal of Computational Physics*, 267:112–138, 2014.
- [16] W. Boscheri and M. Dumbser. Arbitrary-Lagrangian-Eulerian One-Step WENO Finite Volume Schemes on Unstructured Triangular Meshes. *Communications in Computational Physics*, 14:1174–1206, 2013.

- [17] W. Boscheri and M. Dumbser. An Efficient Quadrature-Free Formulation for High Order Arbitrary-Lagrangian-Eulerian ADER-WENO Finite Volume Schemes on Unstructured Meshes. *submitted to Journal of Scientific Computing*, 2014.
- [18] W. Boscheri, M. Dumbser, and D.S. Balsara. High Order Lagrangian ADER-WENO Schemes on Unstructured Meshes – Application of Several Node Solvers to Hydrodynamics and Magnetohydrodynamics. *International Journal for Numerical Methods in Fluids*, 76:737–778, 2014.
- [19] W. Boscheri, M. Dumbser, and O. Zanotti. High order cell-centered lagrangian-type finite volume schemes with time-accurate local time stepping on unstructured triangular meshes. *submitted to Journal of Computational Physics*, 2014.
- [20] Walter Boscheri and Michael Dumbser. A direct arbitrary-lagrangian-eulerian ader-weno finite volume scheme on unstructured tetrahedral meshes for conservative and non-conservative hyperbolic systems in 3d. *Journal of Computational Physics*, 275(0):484 – 523, 2014.
- [21] J. Breil, T. Harribey, P.H. Maire, and M.J. Shashkov. A multi-material ReALE method with MOF interface reconstruction. *Computers and Fluids*, 83:115–125, 2013.
- [22] E. J. Caramana and Raphaël Loubère. "curl-q": A vorticity damping artificial viscosity for essentially irrotational lagrangian hydrodynamics calculations. *J. Comput. Physics*, 215(2):385–391, 2006.
- [23] 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*, 146:227–262, 1998.
- [24] E.J. Caramana and R. Loubère. "curl-q": A vorticity damping artificial viscosity for essentially irrotational lagrangian hydrodynamics calculations. *J. Comput. Phys.*, 215:385–391, 2005.
- [25] E.J. Caramana, C.L. Rousculp, and D.E. Burton. A compatible, energy and symmetry preserving lagrangian hydrodynamics algorithm in three-dimensional cartesian geometry. *Journal of Computational Physics*, 157:89 – 119, 2000.
- [26] G. Carré, S. Del Pino, B. Després, and E. Labourasse. A cell-centered Lagrangian hydrodynamics scheme on general unstructured meshes in arbitrary dimension. *Journal of Computational Physics*, 228:5160–5183, 2009.
- [27] J. Cesenek, M. Feistauer, J. Horacek, V. Kucera, and J. Prokopova. Simulation of compressible viscous flow in time-dependent domains. *Applied Mathematics and Computation*, 219:7139–7150, 2013.
- [28] J. Cheng and C.W. Shu. A high order ENO conservative Lagrangian type scheme for the compressible Euler equations. *Journal of Computational Physics*, 227:1567–1596, 2007.
- [29] S. Clain, S. Diot, and R. Loubère. A high-order finite volume method for systems of conservation laws–multi-dimensional optimal order detection (MOOD). *Journal of Computational Physics*, 230(10):4028 – 4050, 2011.
- [30] S. Clain, S. Diot, and R. Loubère. Multi-dimensional optimal order detection (mood) – a very high-order finite volume scheme for conservation laws on unstructured meshes. In Fort Fürst Halama Herbin Hubert (Eds.), editor, *FVCA 6, International Symposium, Prague, June 6-10*, volume 4 of *Series: Springer Proceedings in Mathematics*, 2011. 1st Edition. XVII, 1065 p. 106 illus. in color.
- [31] S. Clain and G. Machado. A very high-order finite volume method for the time-dependent convection–diffusion problem with Butcher Tableau extension. *Computers and Mathematics with Applications*, 68:1292–1311, 2014.
- [32] Stéphane Clain, Gaspar J Machado, JM Nóbrega, and RMS Pereira. A sixth-order finite volume method for multidomain convection–diffusion problem with discontinuous coefficients. *Computer Methods in Applied Mechanics and Engineering*, 267:43–64, 2013.
- [33] A. Claisse, B. Després, E.Labourasse, and F. Ledoux. A new exceptional points method with application to cell-centered Lagrangian schemes and curved meshes. *Journal of Computational Physics*, 231:4324–4354, 2012.
- [34] B. Cockburn, G. E. Karniadakis, and C.W. Shu. *Discontinuous Galerkin Methods*. Lecture Notes in Computational Science and Engineering. Springer, 2000.
- [35] P. Colella and M.D. Sekora. A limiter for PPM that preserves accuracy at smooth extrema. *Journal of Computational Physics*, 227:7069–7076, 2008.
- [36] B. Després and C. Mazeran. Symmetrization of Lagrangian gas dynamic in dimension two and multidimensional solvers. *C.R. Mecanique*, 331:475–480, 2003.
- [37] B. Després and C. Mazeran. Lagrangian gas dynamics in two-dimensions and Lagrangian systems. *Archive for Rational Mechanics and Analysis*, 178:327–372, 2005.
- [38] S. Diot, S. Clain, and R. Loubère. Improved detection criteria for the multi-dimensional optimal order detection (MOOD) on unstructured meshes with very high-order polynomials. *Computers and Fluids*, 64:43 – 63, 2012.
- [39] S. Diot, R. Loubère, and S. Clain. The MOOD method in the three-dimensional case: Very-high-order finite volume method for hyperbolic systems. *International Journal of Numerical Methods in Fluids*, 73:362–392, 2013.
- [40] V. Dobrev, T. Kolev, and R. Rieben. High-order curvilinear finite element methods for lagrangian hydrodynamics. *SIAM Journal on Scientific Computing*, 34(5):B606–B641, 2012.
- [41] V.A. Dobrev, T.E. Ellis, Tz.V. Kolev, and R.N. Rieben. Curvilinear Finite elements for Lagrangian hydrodynamics. *International Journal for Numerical Methods in Fluids*, 65:1295–1310, 2011.
- [42] Veselin A. Dobrev, Truman E. Ellis, Tzanio V. Kolev, and Robert N. Rieben. High-order curvilinear finite elements for axisymmetric lagrangian hydrodynamics. *Computers & Fluids*, 83(0):58 – 69, 2013. Numerical methods for highly compressible multi-material flow problems.
- [43] L. Dubcova, M. Feistauer, J. Horacek, and P. Svacek. Numerical simulation of interaction between turbulent flow and a vibrating airfoil. *Computing and Visualization in Science*, 12:207–225, 2009.
- [44] M. Dubiner. Spectral methods on triangles and other domains. *Journal of Scientific Computing*, 6:345–390, 1991.
- [45] J.K. Dukovicz and B. Meltz. Vorticity errors in multidimensional lagrangian codes. *Journal of Computational Physics*, 99:115

- 134, 1992.
- [46] M. Dumbser. Arbitrary-Lagrangian-Eulerian ADER-WENO Finite Volume Schemes with Time-Accurate Local Time Stepping for Hyperbolic Conservation Laws. *Computational Methods in Applied Mechanics and Engineering*, 280:57–83, 2014.
- [47] M. Dumbser, D. Balsara, E.F. Toro, and C.D. Munz. A unified framework for the construction of one-step finite-volume and discontinuous Galerkin schemes. *Journal of Computational Physics*, 227:8209–8253, 2008.
- [48] M. Dumbser and W. Boscheri. High-order unstructured Lagrangian one-step WENO finite volume schemes for non-conservative hyperbolic systems: Applications to compressible multi-phase flows. *Computers and Fluids*, 86:405–432, 2013.
- [49] M. Dumbser, C. Enaux, and E.F. Toro. Finite volume schemes of very high order of accuracy for stiff hyperbolic balance laws. *Journal of Computational Physics*, 227:3971–4001, 2008.
- [50] M. Dumbser, M. Kaeser, V.A. Titarev, and E.F. Toro. Quadrature-free non-oscillatory finite volume schemes on unstructured meshes for nonlinear hyperbolic systems. *Journal of Computational Physics*, 226:204–243, 2007.
- [51] M. Dumbser and M. Käser. Arbitrary high order non-oscillatory finite volume schemes on unstructured meshes for linear hyperbolic systems. *Journal of Computational Physics*, 221:693–723, 2007.
- [52] M. Dumbser, M. Käser, V.A. Titarev, and E.F. Toro. Quadrature-free non-oscillatory finite volume schemes on unstructured meshes for nonlinear hyperbolic systems. *Journal of Computational Physics*, 226:204–243, 2007.
- [53] M. Dumbser and E. F. Toro. On universal Osher-type schemes for general nonlinear hyperbolic conservation laws. *Communications in Computational Physics*, 10:635–671, 2011.
- [54] M. Dumbser, A. Uuriintsetseg, and O. Zanotti. On Arbitrary–Lagrangian–Eulerian One–Step WENO Schemes for Stiff Hyperbolic Balance Laws. *Communications in Computational Physics*, 14:301–327, 2013.
- [55] Michael Dumbser and Walter Boscheri. High-order unstructured lagrangian one-step (WENO) finite volume schemes for non-conservative hyperbolic systems: Applications to compressible multi-phase flows. *Computers & Fluids*, 86(0):405 – 432, 2013.
- [56] M. Feistauer, J. Horacek, M. Ruzicka, and P. Svacek. Numerical analysis of flow-induced nonlinear vibrations of an airfoil with three degrees of freedom. *Computers and Fluids*, 49:110–127, 2011.
- [57] M. Feistauer, V. Kucera, J. Prokopova, and J. Horacek. The ALE discontinuous Galerkin method for the simulatio of air flow through pulsating human vocal folds. *AIP Conference Proceedings*, 1281:83–86, 2010.
- [58] M.M. Francois, M.J. Shashkov, T.O. Masser, and E.D. Dendy. A comparative study of multimaterial Lagrangian and Eulerian methods with pressure relaxation. *Computers and Fluids*, 83:126–136, 2013.
- [59] F.Vilar. Cell-centered discontinuous Galerkin discretization for two-dimensional Lagrangian hydrodynamics. *Computers and Fluids*, 64:64–73, 2012.
- [60] F.Vilar, P.H. Maire, and R. Abgrall. Cell-centered discontinuous Galerkin discretizations for two-dimensional scalar conservation laws on unstructured grids and for one-dimensional Lagrangian hydrodynamics. *Computers and Fluids*, 46(1):498–604, 2010.
- [61] F.Vilar, P.H. Maire, and R. Abgrall. A discontinuous Galerkin discretization for solving the two-dimensional gas dynamics equations written under total Lagrangian formulation on general unstructured grids. *Journal of Computational Physics*, 276:188–234, 2014.
- [62] S. Galera, P.H. Maire, and J. Breil. A two-dimensional unstructured cell-centered multi-material ale scheme using vof interface reconstruction. *Journal of Computational Physics*, 229:5755–5787, 2010.
- [63] S.L. Gavriluk, N. Favrie, and R. Saurel. Modelling wave dynamics of compressible elastic materials. *Journal of Computational Physics*, 227:2941–2969, 2008.
- [64] B. Giacomazzo and L. Rezzolla. The exact solution of the Riemann problem in relativistic magnetohydrodynamics. *Journal of Fluid Mechanics*, 562:223–259, 2006.
- [65] S.K. Godunov and E.I. Romenski. Nonstationary equations of the nonlinear theory of elasticity in Euler coordinates. *Journal of Applied Mechanics and Technical Physics*, 13:868–885, 1972.
- [66] A. Hidalgo and M. Dumbser. ADER schemes for nonlinear systems of stiff advection-diffusion-reaction equations. *Journal of Scientific Computing*, 48:173–189, 2011.
- [67] C.W. Hirt, A.A. Amsden, and J.L. Cook. An arbitrary Lagrangian-Eulerian computing method for all flow speeds. *J. Comput. Phys.*, 14(3):227–253, March 1974. doi:10.1016/0021-9991(74)90051-5.
- [68] V. Honkkila and P. Janhunen. HLLC solver for ideal relativistic MHD. *Journal of Computational Physics*, 223:643–656, 2007.
- [69] C. Hu and C.W. Shu. Weighted essentially non-oscillatory schemes on triangular meshes. *Journal of Computational Physics*, 150:97–127, 1999.
- [70] G.-S. Jiang and C.W. Shu. Efficient implementation of weighted ENO schemes. *Journal of Computational Physics*, 126:202–228, 1996.
- [71] J.R. Kamm and F.X. Timmes. On efficient generation of numerically robust sedov solutions. *Technical Report LA-UR-07-2849, LANL*, 2007.
- [72] J.R. Kamm and F.X. Timmes. On efficient generation of numerically robust Sedov solutions. Technical Report LA-UR-07-2849, Los Alamos National Laboratory, 2007.
- [73] G. E. Karniadakis and S. J. Sherwin. *Spectral/hp Element Methods in CFD*. Oxford University Press, 1999.
- [74] M. Käser and A. Iske. ADER schemes on adaptive triangular meshes for scalar conservation laws. *Journal of Computational Physics*, 205:486–508, 2005.
- [75] R.E. Kidder. Laser-driven compression of hollow shells: power requirements and stability limitations. *Nucl. Fus.*, 1:3 – 14, 1976.

- [76] P.M. Knupp. Achieving finite element mesh quality via optimization of the jacobian matrix norm and associated quantities. part ii – a framework for volume mesh optimization and the condition number of the jacobian matrix. *Int. J. Numer. Meth. Engng.*, 48:1165 – 1185, 2000.
- [77] M. Kucharik, J. Breil, S. Galera, P.H. Maire, M. Berndt, and M.J. Shashkov. Hybrid remap for multi-material ALE. *Computers and Fluids*, 46:293–297, 2011.
- [78] M. Kucharik and M.J. Shashkov. One-step hybrid remapping algorithm for multi-material arbitrary Lagrangian-Eulerian methods. *Journal of Computational Physics*, 231:2851–2864, 2012.
- [79] Z. Li, X. Yu, and Z. Jia. The cell-centered discontinuous Galerkin method for Lagrangian compressible Euler equations in two dimensions. *Computers and Fluids*, 96:152–164, 2014.
- [80] R. Liska, M.J. Shashkov P. Váchal, and B. Wendroff. Synchronized flux corrected remapping for ALE methods. *Computers and Fluids*, 46:312–317, 2011.
- [81] W. Liu, J. Cheng, and C.W. Shu. High order conservative Lagrangian schemes with Lax-Wendroff type time discretization for the compressible Euler equations. *Journal of Computational Physics*, 228:8872–8891, 2009.
- [82] R. Loubère, M. Dumbser, and S. Diot. A new family of high order unstructured mood and ader finite volume schemes for multidimensional systems of hyperbolic conservation laws. *Communication in Computational Physics*, 16:718–763, 2014.
- [83] R. Loubère, P.-H. Maire, and P. Váchal. Staggered Lagrangian hydrodynamics based on cell-centered Riemann solver. *Communications in Computational Physics*, 10(4):940–978, 2010.
- [84] R. Loubère, P.H. Maire, and P. Váchal. A second-order compatible staggered Lagrangian hydrodynamics scheme using a cell-centered multidimensional approximate Riemann solver. *Procedia Computer Science*, 1:1931–1939, 2010.
- [85] R. Loubère, P.H. Maire, and P. Váchal. 3D staggered Lagrangian hydrodynamics scheme with cell-centered Riemann solver-based artificial viscosity. *International Journal for Numerical Methods in Fluids*, 72:22 – 42, 2013.
- [86] Raphaël Loubère, Pierre-Henri Maire, Mikhail Yu. Shashkov, Jérôme Breil, and Stéphane Galera. Reale: A reconnection-based arbitrary-lagrangian-eulerian method. *J. Comput. Physics*, 229(12):4724–4761, 2010.
- [87] P.H. Maire. A high-order cell-centered lagrangian scheme for two-dimensional compressible fluid flows on unstructured meshes. *Journal of Computational Physics*, 228:2391–2425, 2009.
- [88] P.H. Maire. A high-order one-step sub-cell force-based discretization for cell-centered lagrangian hydrodynamics on polygonal grids. *Computers and Fluids*, 46(1):341–347, 2011.
- [89] P.H. Maire. A unified sub-cell force-based discretization for cell-centered lagrangian hydrodynamics on polygonal grids. *International Journal for Numerical Methods in Fluids*, 65:1281–1294, 2011.
- [90] P.H. Maire, R. Abgrall, J. Breil, and J. Ovardia. A cell-centered lagrangian scheme for two-dimensional compressible flow problems. *SIAM Journal on Scientific Computing*, 29:1781–1824, 2007.
- [91] P.H. Maire and J. Breil. A second-order cell-centered lagrangian scheme for two-dimensional compressible flow problems. *International Journal for Numerical Methods in Fluids*, 56:1417–1423, 2007.
- [92] P.H. Maire and B. Nkonga. Multi-scale Godunov-type method for cell-centered discrete Lagrangian hydrodynamics. *Journal of Computational Physics*, 228:799–821, 2009.
- [93] C.D. Munz. On Godunov-type schemes for Lagrangian gas dynamics. *SIAM Journal on Numerical Analysis*, 31:17–42, 1994.
- [94] J. Von Neumann and R. D. Richtmyer. A method for the numerical calculation of hydrodynamic shocks. *Journal of Applied Physics*, 21:232–237, 1950.
- [95] W.F. Noh. Errors for calculations of strong shocks using an artificial viscosity and an artificial heat flux. *J. Comput. Phys.*, 72:78–120, 1987.
- [96] C. Olliver-Gooch and M. Van Altena. A high-order-accurate unstructured mesh finite-volume scheme for the advection-diffusion equation. *Journal of Computational Physics*, 181:729 – 752, 2002.
- [97] A. López Ortega and G. Scovazzi. A geometrically-conservative, synchronized, flux-corrected remap for arbitrary Lagrangian-Eulerian computations with nodal finite elements. *Journal of Computational Physics*, 230:6709–6741, 2011.
- [98] J.S. Peery and D.E. Carroll. Multi-material ale methods in unstructured grids,. *Computer Methods in Applied Mechanics and Engineering*, 187:591–619, 2000.
- [99] P.Hoch. An arbitrary-lagrangian-eulerian strategie to solve compressible fluid flow. private communication, 2009.
- [100] L. Rezzolla and O. Zanotti. An improved exact riemann solver for relativistic hydrodynamics. *Journal of Fluid Mechanics*, 449:395–411, 2001.
- [101] Stéphane Clain Ricardo Costa, Gaspar J. Machado. Sixth-order finite volume method for the 1d biharmonic operator: application to the intramedullary nail simulation. *International Journal of Applied Mathematics and Computer Science (AMCS)*, 2014.
- [102] William J. Rider. Revisiting wall heating. *J. Comput. Phys.*, 162(2):395–410, August 2000.
- [103] R.Loubère and M.J.Shashkov. A subcell remapping method on staggered polygonal grids for arbitrary-lagrangian-eulerian methods. *J. Comput. Phys.*, 209:105–138, 2005.
- [104] R.Saurel and O.Lemetayer. A multiphase model for compressible flows with interfaces, shocks, detonation waves and cavitation. *Journal of Fluid Mechanics*, 431:239–271, 2001. doi:10.1017/S0022112000003098.
- [105] V. V. Rusanov. Calculation of Interaction of Non-Steady Shock Waves with Obstacles. *J. Comput. Math. Phys. USSR*, 1:267–279, 1961.
- [106] J. Saltzman and P. Colella. Second order upwind transport methods for lagrangian hydrodynamics. Technical report, Los Alamos National Laboratory, 1985.
- [107] S.K. Sambasivan, M.J. Shashkov, and D.E. Burton. A finite volume cell-centered Lagrangian hydrodynamics approach for

- solids in general unstructured grids. *International Journal for Numerical Methods in Fluids*, 72:770–810, 2013.
- [108] S.K. Sambasivan, M.J. Shashkov, and D.E. Burton. Exploration of new limiter schemes for stress tensors in Lagrangian and ALE hydrocodes. *Computers and Fluids*, 83:98–114, 2013.
- [109] G. Scovazzi. Lagrangian shock hydrodynamics on tetrahedral meshes: A stable and accurate variational multiscale approach. *Journal of Computational Physics*, 231:8029–8069, 2012.
- [110] L.I. Sedov. *Similarity and Dimensional Methods in Mechanics*. Academic Press, New York, 1959.
- [111] C.W. Shu. Essentially non-oscillatory and weighted essentially non-oscillatory schemes for hyperbolic Conservation Laws. *NASA/CR-97-206253 ICASE Report No.97-65*, November 1997.
- [112] R.W. Smith. AUSM(ALE): a geometrically conservative arbitrary lagrangian–eulerian flux splitting scheme. *Journal of Computational Physics*, 150:268–286, 1999.
- [113] A.H. Stroud. *Approximate Calculation of Multiple Integrals*. Prentice-Hall Inc., Englewood Cliffs, New Jersey, 1971.
- [114] A. Suresh and H.T. Huynh. Accurate monotonicity-preserving schemes with runge-kutta time stepping. *Journal of Computational Physics*, 136:83–99, 1997.
- [115] V.A. Titarev, E.I. Romenski, and E.F. Toro. MUSTA-type upwind fluxes for non-linear elasticity. *International Journal for Numerical Methods in Engineering*, 73:897–926, 2008.
- [116] V.A. Titarev and E.F. Toro. ADER: Arbitrary high order Godunov approach. *Journal of Scientific Computing*, 17(1-4):609–618, December 2002.
- [117] V.A. Titarev and E.F. Toro. ADER schemes for three-dimensional nonlinear hyperbolic systems. *Journal of Computational Physics*, 204:715–736, 2005.
- [118] V.A. Titarev, P. Tsoutsanis, and D. Drikakis. WENO schemes for mixed–element unstructured meshes. *Communications in Computational Physics*, 8:585–609, 2010.
- [119] E. F. Toro and V. A. Titarev. Derivative Riemann solvers for systems of conservation laws and ADER methods. *Journal of Computational Physics*, 212(1):150–165, 2006.
- [120] E.F. Toro. *Riemann Solvers and Numerical Methods for Fluid Dynamics*. Springer, second edition, 1999.
- [121] E.F. Toro. Anomalies of conservative methods: analysis, numerical evidence and possible cures. *International Journal of Computational Fluid Dynamics*, 11:128–143, 2002.
- [122] P. Tsoutsanis, V.A. Titarev, and D. Drikakis. WENO schemes on arbitrary mixed-element unstructured meshes in three space dimensions. *Journal of Computational Physics*, 230:1585–1601, 2011.
- [123] V.Desveaux. *Contribution à l'approximation numérique des systèmes hyperboliques*. PhD thesis, Université de Nantes, 2013.
- [124] V.Desveaux and C.Berthon. An entropic mood scheme for the euler equations. *International Journal of Finite Volumes*, 2013.
- [125] M.L. Wilkins. Calculation of elastic plastic flow. In B. Alder, S. Fernbach, and M. Rotenberg, editors, *Methods in Computational Physics*, volume 3, pages 211–263. Academic Press, New York, 1964.
- [126] Alan M. Winslow. Numerical solution of the quasilinear poisson equation in a nonuniform triangle mesh. *J. Comput. Phys.*, 135(2):128–138, August 1997.
- [127] Y.V. Yanilkin, E.A. Goncharov, V.Y. Kolobyanin, V.V. Sadchikov, J.R. Kamm, M.J. Shashkov, and W.J. Rider. Multi-material pressure relaxation methods for lagrangian hydrodynamics. *Computers and Fluids*, 83:137–143, 2013.
- [128] L. Del Zanna, N. Bucciantini, and P. Londrillo. An efficient shock-capturing central-type scheme for multidimensional relativistic flows II. magnetohydrodynamics. *Astronomy and Astrophysics*, 400:397–413, 2003.
- [129] O. Zanotti, M. Dumbser, R. Loubère, and S.Diot. A posteriori subcell limiting for Discontinuous Galerkin Finite Element method for hyperbolic system of conservation laws. *Journal of Computational Physics*, 278:47–75, 2014.