Version of Record: https://www.sciencedirect.com/science/article/pii/S0021999119305972
Manuscript_e767512fd8606dcfc9660f1f91270f9f

High order direct Arbitrary-Lagrangian-Eulerian (ALE) $P_N P_M$ schemes with WENO Adaptive-Order reconstruction on unstructured meshes

Walter Boscheri[a], Dinshaw S. Balsara[b]

[a]*Department of Mathematics and Computer Science, University of Ferrara, Via Machiavelli 30, 44121 Ferrara, Italy*
[b]*Physics Department, University of Notre Dame, 225 Nieuwland Science Hall, Notre Dame, IN 46556, USA*

Abstract

In this work we present a conservative WENO Adaptive Order (AO) reconstruction operator applied to an explicit one-step Arbitrary-Lagrangian-Eulerian (ALE) discontinuous Galerkin (DG) method. The spatial order of accuracy is improved by reconstructing higher order piecewise polynomials of degree $M > N$, starting from the underlying polynomial solution of degree N provided by the DG scheme. High order of accuracy in time is achieved by the ADER approach, making use of an element-local space-time Galerkin finite element predictor that arises from a one-step time integration procedure. As a result, space-time polynomials of order $M + 1$ are obtained and used to perform the time evolution of the numerical solution adopting a fully explicit DG scheme.

To maintain algorithm simplicity, the mesh motion is restricted to be carried out using straight lines, hence the old mesh configuration at time t^n is connected with the new one at time t^{n+1} via space-time segments, which result in space-time control volumes on which the governing equations have to be integrated in order to obtain the time evolution of the discrete solution. Our algorithm falls into the category of *direct* Arbitrary-Lagrangian-Eulerian (ALE) schemes, where the governing PDE system is directly discretized relying on a space-time conservation formulation and which already takes into account the new grid geometry *directly* during the computation of the numerical fluxes. A local rezoning strategy might be used in order to locally optimize the mesh quality and avoiding the generation of invalid elements with negative determinant. The proposed approach reduces to direct ALE finite volume schemes if $N = 0$, while explicit direct ALE DG schemes are recovered in the case of $N = M$.

In order to stabilize the DG solution, an *a priori* WENO based limiting technique is employed, that makes use of the numerical solution inside the element under consideration and its neighbor cells to find a less oscillatory polynomial approximation. By using a *modal basis* in a reference element, the evaluation of the oscillation indicators is very easily and efficiently carried out, hence allowing higher order modes to be properly limited, while leaving the zero-*th* order mode untouched for ensuring conservation.

Numerical convergence rates for $2 \leq N, M \leq 4$ are presented as well as a wide set of benchmark test problems for hydrodynamics on moving and fixed unstructured meshes.

Keywords: Arbitrary-Lagrangian-Eulerian (ALE), $P_N P_M$ schemes, reconstructed DG schemes, WENO-AO reconstruction, ADER time discretization, high order of accuracy in space and time, moving unstructured meshes, local rezoning, hyperbolic PDE, Euler equations

1. Introduction

This work is concerned with the application of a reconstruction operator to the discontinuous Galerkin (DG) method on unstructured moving meshes. The idea of improving the accuracy of the DG scheme by performing a reconstruction step was first proposed in [31, 32], where the reconstruction was used as a post-processing technique. On the contrary, in [43] the reconstruction operator was applied for the first time at the beginning of each time step in the computation, thus increasing the formal order of accuracy of the underlying DG method. In [97] the diffusion

Email addresses: walter.boscheri@unife.it (Walter Boscheri), dbalsara@nd.edu (Dinshaw S. Balsara)

equations have been discretized by means of a DG scheme combined with a reconstruction algorithm, while nonlinear reconstruction operators of WENO-type have been used as limiters for DG schemes in [60, 3, 102, 89, 88]. Recent contributions to reconstructed DG schemes can be found in [28, 25, 75].

In [44] a *linear* reconstruction operator has been used in the context of DG schemes for the solution of hyperbolic balance laws on unstructured meshes. Specifically, a unified framework for finite volume (FV) and DG methods has been developed and the reconstruction step has been used for generating a new family of algorithms called $P_N P_M$ schemes, that embed as special cases both FV and DG methods. The DG solution is given in terms of piecewise polynomials of degree N, whereas M denotes the degree of the reconstruction operator employed for enhancing the overall accuracy of the scheme. Therefore, if $N = 0$ finite volume methods are recovered, otherwise fully DG schemes are constructed in the case of $N = M$. In [44] no limiters were used for the DG scheme, hence using $N = 0$ when shocks or strong discontinuities were present in the large suit of test cases run in that work. For $1 < N < M$ new classes of $P_N P_M$ schemes were found in [44] whose accuracy was comparable to that of DG schemes; i.e. the $P_N P_M$ schemes. This finding confirmed ones intuition that most of the significant variation in a DG scheme is carried by the lower moments, and usually by the first moment [3], with the result that the higher moments could be reconstructed at the beginning of each time step to retrieve the high formal order of accuracy. A von Neumann stability analysis of $P_N P_M$ schemes showed that they could operate stably with a limiting CFL that was closer to that of $P_N P_M$ schemes. Since $P_N P_M$ schemes can have limiting CFL numbers that are substantially larger than $P_N P_M$ schemes when $N \ll M$, this represented a major increase in permitted time steps with minimal sacrifice of accuracy. Recently, Balsara and Käppeli [8, 9] have presented $P_N P_M$ schemes for constraint-preserving MHD and CED and the same finding with respect to accuracy and time step stability was re-confirmed for this very different class of schemes. This recent work significantly enlarges the finding in [44].

DG schemes have also gained interest in the field of Lagrangian algorithms, where the mesh moves with the fluid flow. A lot of research has been carried out for devising moving mesh algorithms, because of the excellent properties achieved by these numerical methods in the resolution of moving material interfaces and contact waves. Finite element algorithms for Lagrangian hydrodynamics and the equations of nonlinear elasto-plasticity have been presented in [84, 91, 40, 38, 39], while Lagrangian DG methods have been proposed for the first time in [78, 54, 52, 53, 70]. A Taylor basis was used in [58] for the development of a discontinuous Galerkin spectral finite element method. In [54, 52, 53, 74] a so-called *total Lagrangian* approach is adopted, in which the governing equations also account for the time evolution of the the Jacobian matrix associated to the transformation from the Lagrangian coordinates to the reference coordinates related to the initial configuration of the flow. Consequently, the mesh is kept fixed, and the information about the grid motion is contained in the mapping. Recently in [101], an *updated Lagrangian* DG scheme has been developed for hydrodynamics, where the mesh physically moves with the fluid flow. Lagrangian DG methods can also be found in [81, 73], while reconstructed DG algorithms for compressible flows have been presented in [99, 86]. An ALE formulation which is based on a reconstructed DG scheme for the solution of hydrodynamics equations has been recently forwarded in [98].

Arbitrary-Lagrangian-Eulerian (ALE) methods [90, 12, 67, 71, 66, 11] also belong to the category of moving mesh algorithms, and they are characterized by more flexibility in the choice of the mesh velocity. Indeed, in the ALE context the grid velocity can be chosen independently from the local fluid velocity, hence recovering fully Eulerian algorithms on fixed grids, in the special case where the mesh velocity is set to zero. In [18] a direct ALE DG scheme was proposed, with the mesh motion taken into account *directly* in the numerical flux computation. This approach derives from previous contributions in the field of finite volume schemes [14, 13, 20].

In this paper we propose to develop $P_N P_M$ schemes in the framework of direct ALE methods on unstructured meshes. The nonlinear WENO Adaptive-Order (AO) reconstruction [7, 6] will be employed for obtaining higher accuracy within the reconstruction operator applied to the DG solution, while a WENO limiter for DG is used following the work presented in [106]. Unlike the *a posteriori* finite volume limiters designed in [51, 49, 92, 35, 18], an *a priori* WENO limiter is employed for the first time in the context of direct ALE. We use the flattener detector [5] as troubled cell indicator in order to mark the elements that need to be limited, then the shifting polynomial technique devised in [6] allows the WENO limiting to be carried out on the basis of the piecewise DG polynomials defined on each computational cell. The WENO AO reconstruction operator is then performed on the top of the limited DG solution, thus generating a reconstructed DG approximation of the numerical solution of higher accuracy. The stability of the DG scheme is ensured via our novel *a priori* WENO limiter, while the nonlinear WENO AO strategy makes the reconstruction polynomial also stable. Therefore, the ALE $P_N P_M$ method proposed in this work can be used for

running computations involving shocks and other discontinuities on both moving and fixed unstructured meshes. To our best knowledge, this is the *first time* that the $P_N P_M$ method (also known as "reconstructed DG") is used for the development of high order schemes on moving computational domains.

$P_N P_M$ schemes combine the well-known high resolution properties of DG schemes with the robustness of a reconstruction operator that is commonly used in the finite volume framework. Thus, such methods can achieve higher resolution compared to finite volume schemes while being computationally less expensive than DG algorithms. Moreover, we aim at developing an ALE numerical scheme which can handle both fixed and moving computational domains within the same framework, hence allowing more flexibility for general applications. Specifically, a direct ALE method will be derived, that grants the computational mesh to be moved with more freedom compared to fully Lagrangian approaches. In this way, any remapping step is avoided, hence saving computational time, and reducing any further numerical dissipation coming from the projection of the numerical solution from the Lagrangian grid to the remapped mesh. This can only be possible if one considers a fully space-time discretization of the governing equations that automatically takes into account the geometry changes in the definition of the space-time control volumes. As such, ADER time stepping is likely to be used, because it provides a one-step time discretization of the PDE system, starting from a high order spatial data representation obtained at the aid of the WENO-AO operator. Furthermore, the direct ALE approach permits to directly insert a mesh optimization or rezoning procedure within the mesh motion step, thus improving the robustness of the method.

The outline of this article is as follows: all the details regarding the numerical method are contained in Section 2, including the WENO limiter presented in Section 2.1.1 as well as the WENO AO reconstruction which is described in Section 2.1.2. Section 3 shows numerical convergence rates from third up to fifth order of accuracy in space and time for a smooth problem as well as a wide set of benchmark test problems considering the Euler equations of compressible gas dynamics. Finally, we give some concluding remarks and an outlook to possible future work in Section 4.

2. The $P_N P_M$ method on moving unstructured meshes

The governing equations are given by nonlinear hyperbolic conservation laws which can be written in space-time divergence form as

$$\tilde{\nabla} \cdot \tilde{\mathbf{F}} = \mathbf{S}, \qquad \mathbf{x} \in \Omega(t) \subset \mathbb{R}^d, \quad t \in \mathbb{R}_0^+, \quad \mathbf{Q} \in \Omega_{\mathbf{Q}} \subset \mathbb{R}^\nu, \tag{1}$$

where $\tilde{\nabla} = \left(\frac{\partial}{\partial x}, \frac{\partial}{\partial y}, \frac{\partial}{\partial z}, \frac{\partial}{\partial t}\right)^T$ denotes the space-time divergence operator, $\tilde{\mathbf{F}} = (\mathbf{f}, \mathbf{g}, \mathbf{h}, \mathbf{Q})$ is the space-time flux tensor and $\mathbf{S} = \mathbf{S}(\mathbf{Q})$ represents an algebraic source term. In the ALE framework the computational domain $\Omega(t)$ is dependent on time t and is defined in $d \in [2, 3]$ space dimensions by the spatial coordinate vector $\mathbf{x} = (x, y, z)$. Then, $(\mathbf{f}, \mathbf{g}, \mathbf{h})$ is the purely conservative flux tensor that contains the flux vectors for each spatial component (x, y, z), respectively, while the term $\mathbf{Q}$ represents the vector of conserved variables defined in the space of the admissible states $\Omega_{\mathbf{Q}} \subset \mathbb{R}^\nu$.

At any time level t^n, the computational domain is discretized by a total number of N_E non-overlapping simplex control volumes T_i^n that constitute the *current mesh configuration* $\mathcal{T}_\Omega^n$, that is

$$\mathcal{T}_\Omega^n = \bigcup_{i=1}^{N_E} T_i^n. \tag{2}$$

Furthermore, let $\tilde{T}_i = T_i^n \times \Delta t$ be the space-time control volume related to cell T_i, which is nothing but the time evolution of T_i over one time step $\Delta t = t^{n+1} - t^n$. As depicted in Figure 1, the space-time control volume $\tilde{T}_i$ is bounded by a space-time surface $\partial \tilde{T}_i$ that involves a total number of $d + 2$ space-time faces: the bottom and the top surfaces are given by the cell configuration T_i^n at time t^n and T_i^{n+1} at time t^{n+1}, respectively, while the volume is closed laterally by considering the remaining space-time faces, that involve the *Neumann neighborhood* $\mathcal{D}_i$, which is the set of directly adjacent neighbors T_j that share a common face ∂T_{ij} with element T_i. Finally, $\tilde{\mathbf{n}}$ denotes the outward pointing space-time unit normal vector that is defined on the space-time surface $\partial \tilde{T}_i$.

Within each control volume, the numerical solution for the state vector $\mathbf{Q}$ in (1) is expressed as

$$\mathbf{u}_h(\mathbf{x}, t^n) = \sum_{l=1}^{\mathcal{N}} \phi_l(\boldsymbol{\xi})\, \hat{\mathbf{u}}_l^n \qquad \mathbf{x} \in T_i^n, \tag{3}$$

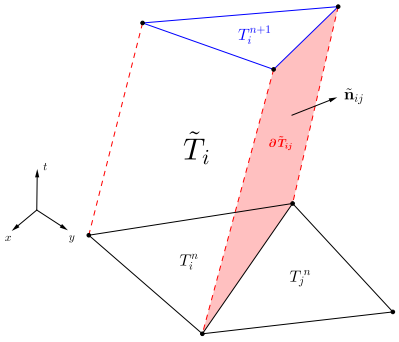

Figure 1: Space-time evolution of element T_i^n within one time step Δt. The space-time surface $\partial\tilde{T}_i$ involves the old (black) T_i^n and the new (blue) T_i^{n+1} element configuration, and the lateral faces (red) ∂T_{ij} which are shared with each Neumann neighbor T_j^n. The outward pointing unit normal vector $\tilde{\mathbf{n}}_{ij}$ is also drawn for face ∂T_{ij}.

where $\mathbf{u}_h(\mathbf{x}, t^n)$ denote piecewise polynomials defined in the space $\mathcal{U}_h$. $\phi_l(\boldsymbol{\xi})$ is a set of spatial basis functions used to span the space $\mathcal{U}_h$ up to degree N and $\hat{\mathbf{u}}_l^n$ are the corresponding expansion coefficients, whose total number $\mathcal{N}$ depends both on N and d, and is explicitly given by

$$\mathcal{N} = \mathcal{H}(N, d) = \frac{1}{d!} \prod_{m=1}^{d} (N + m). \tag{4}$$

The basis functions depend on a reference coordinate system with vector $\boldsymbol{\xi} = (\xi, \eta, \zeta)$, where also the reference element T_E is defined, see Figure 2. The mapping from the physical to the reference space as well as the definition of T_E will be detailed later (see Eqn.(15)). Analogously, let us define a *reconstruction* solution $\mathbf{w}_h(\mathbf{x}, t^n)$ that is the result of the P_M operator, thus it constitutes a piecewise polynomial approximation of degree $M \geq N$ of the numerical solution $\mathbf{u}_h(\mathbf{x}, t^n)$. It reads

$$\mathbf{w}_h(\mathbf{x}, t^n) = \sum_{l=1}^{\mathcal{M}} \psi_l(\boldsymbol{\xi})\, \hat{\mathbf{w}}_l^n \qquad \mathbf{x} \in T_i^n, \tag{5}$$

with the total number $\mathcal{M}$ of expansion coefficients $\hat{\mathbf{w}}_l^n$ computed again using (4) with polynomial degree M, i.e. $\mathcal{M} = \mathcal{H}(M, d)$. Finally, a *space-time* representation of the numerical solution within the space-time control volume $\tilde{T}_i$ is provided by piecewise polynomials $\mathbf{q}_h(\mathbf{x}, t)$ as

$$\mathbf{q}_h(\mathbf{x}, t) = \sum_{l=1}^{\mathcal{L}} \theta_l(\tilde{\boldsymbol{\xi}})\, \hat{\mathbf{q}}_l \qquad (\mathbf{x}, t) \in \tilde{T}_i, \tag{6}$$

where $\theta_l(\tilde{\boldsymbol{\xi}})$ are space-time basis functions of degree M with a total number of $\mathcal{L}$ expansion coefficients $\hat{\mathbf{q}}_l$. They are also defined in a space-time reference coordinate system given by $\tilde{\boldsymbol{\xi}} = (\xi, \eta, \zeta, \tau)$. The number of degrees of freedom $\mathcal{L}$ is evaluated via (4) with $\mathcal{L} = \mathcal{H}(M, d + 1)$, since time counts as an additional dimension. Both basis functions ϕ_l and ψ_l in (3) and (5) are chosen to be a *modal basis* of degree N and M, respectively, while a *nodal basis* is used for $\mathbf{q}_h(\mathbf{x}, t)$ in (6) with $\theta_l(\tilde{\boldsymbol{\xi}})$. Explicit definitions of the basis functions up to degree 3 are detailed in Appendix A.

The reconstruction polynomials $\mathbf{w}_h(\mathbf{x}, t^n)$ are the result of a high order WENO-type reconstruction procedure that is presented in Section 2.1, while the space-time polynomials $\mathbf{q}_h(\mathbf{x}, t)$ arise from an element-local predictor step that is based on a weak formulation of the governing PDE (1) and will be briefly illustrated in Section 2.2.

An explicit discretization of the PDE (1) with a DG method derives from a multiplication by a test function ϕ_k and

a subsequent integration over a space-time control volume $\tilde{T}_i$, hence obtaining

$$\int_{\partial\tilde{T}_i} \phi_k \tilde{\mathbf{F}} \cdot \tilde{\mathbf{n}}\, dS\, dt - \int_{\tilde{T}_i} \tilde{\nabla}\phi_k \cdot \tilde{\mathbf{F}}\, d\mathbf{x}dt = \int_{\tilde{T}_i} \phi_k \cdot \mathbf{S}\, d\mathbf{x}dt, \tag{7}$$

after application of Gauss' theorem. Inserting the definitions (3) and (6), the above formulation becomes

$$\left(\int_{T_i^{n+1}} \phi_k\phi_l\, d\mathbf{x}\right)\hat{\mathbf{u}}_l^{n+1} = \left(\int_{T_i^n} \phi_k\phi_l\, d\mathbf{x}\right)\hat{\mathbf{u}}_l^n - \int_{\partial\tilde{T}_i} \phi_k\, \mathcal{G}\left(\mathbf{q}_h^-, \mathbf{q}_h^+\right)\cdot\tilde{\mathbf{n}}\, dS\, dt + \int_{\tilde{T}_i} \tilde{\nabla}\phi_k \cdot \tilde{\mathbf{F}}(\mathbf{q}_h)\, d\mathbf{x}dt + \int_{\tilde{T}_i} \phi_k \cdot \mathbf{S}(\mathbf{q}_h)\, d\mathbf{x}dt, \tag{8}$$

where $\mathcal{G}$ represents an *Arbitrary-Lagrangian-Eulerian* numerical flux function that provides the coupling between neighbor elements. It depends on the high order boundary-extrapolated data $(\mathbf{q}_h^-, \mathbf{q}_h^+)$ coming from the left and right piecewise polynomials (6) defined in $\tilde{T}_i$ and $\tilde{T}_j$, and it is projected in space-time normal direction $\tilde{\mathbf{n}}$. The Riemann solver $\mathcal{G}$ makes use of the eigenstructure of the system of hyperbolic PDE, thus let us introduce the ALE Jacobian matrix in space normal direction $\mathbf{A}_{\mathbf{n}}^{\mathbf{V}}(\mathbf{Q})$, which reads

$$\mathbf{A}_{\mathbf{n}}^{\mathbf{V}}(\mathbf{Q}) := \left(\sqrt{\tilde{n}_x^2 + \tilde{n}_y^2 + \tilde{n}_z^2}\right)\left[\frac{\partial \mathbf{F}}{\partial \mathbf{Q}}\cdot\mathbf{n} - (\mathbf{V}\cdot\mathbf{n})\,\mathbf{I}\right], \qquad \mathbf{n} = \frac{(\tilde{n}_x, \tilde{n}_y, \tilde{n}_z)^T}{\sqrt{\tilde{n}_x^2 + \tilde{n}_y^2 + \tilde{n}_z^2}}. \tag{9}$$

$\mathbf{I}$ is the identity matrix and $\mathbf{V}\cdot\mathbf{n}$ denotes the local normal mesh velocity, while $\mathbf{n}$ is the spatial part of the space-time normal vector $\tilde{\mathbf{n}}$. A simple and robust ALE Rusanov-type scheme [16] is given by

$$\mathcal{G}\left(\mathbf{q}_h^-, \mathbf{q}_h^+\right)\cdot\tilde{\mathbf{n}} = \frac{1}{2}\left(\tilde{\mathbf{F}}(\mathbf{q}_h^+) + \tilde{\mathbf{F}}(\mathbf{q}_h^-)\right)\cdot\tilde{\mathbf{n}}_{ij} - \frac{1}{2}s_{\max}\left(\mathbf{q}_h^+ - \mathbf{q}_h^-\right), \tag{10}$$

with $s_{\max}$ representing the maximum eigenvalue of $\mathbf{A}_{\mathbf{n}}^{\mathbf{V}}(\mathbf{Q})$, while the less dissipative Osher-type numerical flux function [85, 50, 17] is written as

$$\mathcal{G}\left(\mathbf{q}_h^-, \mathbf{q}_h^+\right)\cdot\tilde{\mathbf{n}} = \frac{1}{2}\left(\tilde{\mathbf{F}}(\mathbf{q}_h^+) + \tilde{\mathbf{F}}(\mathbf{q}_h^-)\right)\cdot\tilde{\mathbf{n}}_{ij} - \frac{1}{2}\left(\int_0^1 \left|\mathbf{A}_{\mathbf{n}}^{\mathbf{V}}(\boldsymbol{\Psi}(s))\right| ds\right)\left(\mathbf{q}_h^+ - \mathbf{q}_h^-\right). \tag{11}$$

The left and the right state across the discontinuity is assumed to be connected by a straight-line segment $\boldsymbol{\Psi}(s) = \mathbf{q}_h^- + s\left(\mathbf{q}_h^+ - \mathbf{q}_h^-\right)$ with $s \in [0, 1]$. The absolute value of the dissipation matrix in (11) is evaluated as usual as

$$|\mathbf{A}_{\mathbf{n}}^{\mathbf{V}}| = \mathbf{R}|\boldsymbol{\Lambda}|\mathbf{R}^{-1}, \qquad |\boldsymbol{\Lambda}| = \mathrm{diag}\left(|\lambda_1|, |\lambda_2|, ..., |\lambda_\nu|\right), \tag{12}$$

with $\mathbf{R}$ and $\mathbf{R}^{-1}$ representing the right eigenvector matrix and its inverse, respectively. A novel numerical flux function based on the FORCE scheme [94, 95] has been recently developed in [20], in which the dissipation matrix only depends on geometric quantities, hence no information about the eigenstructure of the governing system is required any more.

High order of accuracy in space and time is granted by the scheme in one single time step Δt, that is computed under a classical Courant-Friedrichs-Levy number (CFL) stability condition, i.e.

$$\Delta t = \mathrm{CFL}\,\frac{1}{(2N+1)}\min_{T_i^n}\frac{h_i}{|\lambda_{\max,i}|}, \qquad \forall T_i^n \in \Omega^n, \tag{13}$$

where h_i denotes a characteristic element size, either the incircle or the insphere diameter for triangles or tetrahedra, respectively, while $|\lambda_{\max,i}|$ is taken to be the maximum absolute value of the eigenvalues computed from the current solution $\mathbf{Q}_i^n$ in T_i^n. The CFL number has to obey the condition $\mathrm{CFL} \leq \frac{1}{d}$ in multiple space dimensions on unstructured meshes.

Remark. If the mesh velocity is set to zero, i.e. $\mathbf{V} = \mathbf{0}$, then an Eulerian scheme on fixed grid is recovered. In this case, the space-time control volume $\tilde{T}_i$ is simply given by the extrusion of each element T_i^n over the time step Δt, hence generating an space-time control volume which is *orthogonal* w.r.t. time, so that $\tilde{n}_t = 0$ and $\tilde{\mathbf{n}}_{x,y,z} = \mathbf{n}_{x,y,z}$.

2.1. WENO limiter and WENO Adaptive-Order reconstruction for $P_N P_M$ schemes

The ALE $P_N P_M$ scheme (8) is composed of three main building blocks, namely the numerical solution $\mathbf{u}_h$ of order $N+1$, the spatial reconstruction solution $\mathbf{w}_h$ of order $M+1$ and the space-time predictor solution $\mathbf{q}_h$ of order $M+1$. If high order of accuracy is used within the proposed scheme, i.e. $N > 0$ and $M \geq N > 0$, according to Godunov theorem [56], Gibbs phenomena at shock waves or other discontinuities typically occur while solving nonlinear hyperbolic systems of the form (1). To avoid such oscillatory behavior, a *nonlinear hybridization* must be introduced in the algorithm, so that a so-called *limiter* has to be employed for stabilizing both the DG solution $\mathbf{u}_h$ and the reconstruction solution $\mathbf{w}_h$. That is not needed for the predictor solution $\mathbf{q}_h$ because it consists in a local evolution of the PDE which starts from $\mathbf{w}_h$ as initial condition, which is supposed to be non-oscillatory and already limited.

The *modal* basis functions ψ_l used for the expansions (3) and (5) are the orthogonal Dubiner-type basis functions [41, 62, 30], which constitute a hierarchical tensor-type basis. They are derived as a product among two or three principle functions (in 2D or in 3D, respectively) expressed in terms of the Jacobi polynomials. The modal basis functions ψ_l are then defined on the reference element T_E, thus $\psi_l = \psi_l(\boldsymbol{\xi})$, with $\boldsymbol{\xi} = (\xi, \eta, \zeta)$ representing the reference coordinate system. The reference element is depicted in Figure 2 and is either the unit triangle in 2D or the unit tetrahedron in 3D with nodes

$$\left.\begin{array}{l} \boldsymbol{\xi}_1^e = (\xi_1^e, \eta_1^e) = (0,0) \\ \boldsymbol{\xi}_2^e = (\xi_2^e, \eta_2^e) = (1,0) \\ \boldsymbol{\xi}_3^e = (\xi_3^e, \eta_3^e) = (0,1) \end{array}\right\} T_E \in \text{2D}, \qquad \left.\begin{array}{l} \boldsymbol{\xi}_1^e = (\xi_1^e, \eta_1^e, \zeta_1^e) = (0,0,0) \\ \boldsymbol{\xi}_2^e = (\xi_2^e, \eta_2^e, \zeta_2^e) = (1,0,0) \\ \boldsymbol{\xi}_3^e = (\xi_3^e, \eta_3^e, \zeta_3^e) = (0,1,0) \\ \boldsymbol{\xi}_4^e = (\xi_4^e, \eta_4^e, \zeta_4^e) = (0,0,1) \end{array}\right\} T_E \in \text{3D}. \tag{14}$$

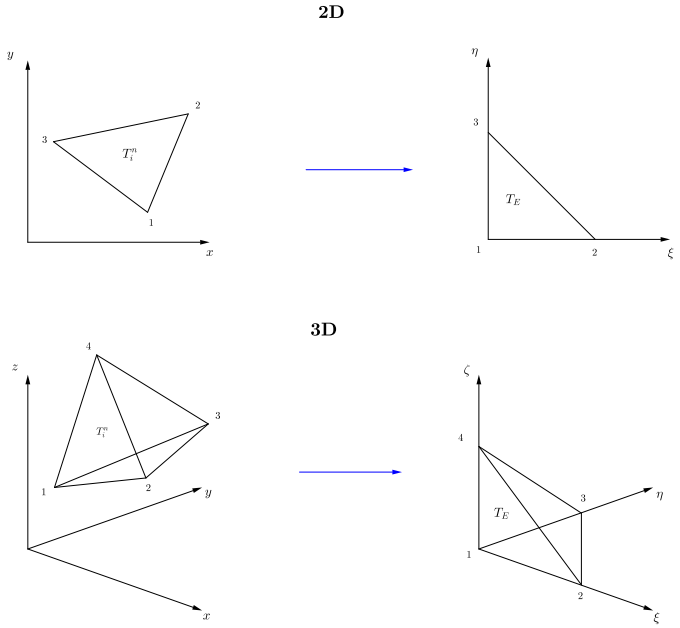

Figure 2: Physical element T_i^n with position vector $\mathbf{x} = (x, y, z)$ and reference element T_E with position vector $\boldsymbol{\xi} = (\xi, \eta, \zeta)$ in 2D (top) and in 3D (bottom).

The mapping from the reference coordinate system $\boldsymbol{\xi}$ to the physical coordinate system $\mathbf{x}$ reads

$$\mathbf{x} = \mathbf{X}_{1,i}^n + \left(\mathbf{X}_{2,i}^n - \mathbf{X}_{1,i}^n\right)\xi + \left(\mathbf{X}_{3,i}^n - \mathbf{X}_{1,i}^n\right)\eta + \left(\mathbf{X}_{4,i}^n - \mathbf{X}_{1,i}^n\right)\zeta, \tag{15}$$

where $\mathbf{X}^n_{k,i} = (X^n_{k,i}, Y^n_{k,i}, Z^n_{k,i})$ represents the vector of physical spatial coordinates of the k-th vertex of element T^n_i.

2.1.1. WENO limiter for P_N

If $N > 0$ the numerical solution $\mathbf{u}_h$ must undergo a process of limiting, in order to produce a stable and essentially non-oscillatory solution at the new time level t^{n+1} via the $P_N P_M$ scheme (8).

Recent efforts have been put in the development of finite volume type limiters [51, 103, 49, 92, 35], which make use of a finite volume scheme at a subgrid level for updating the numerical solution across discontinuities. Specifically, in *a posteriori* limiters [51, 103, 49] a *candidate solution* is first computed by running the DG scheme without any limiting. Then, such solution is verified against a set of physical and numerical detection criteria, such as the positivity of pressure and density, the absence of floating point errors (NaN) and the satisfaction of a relaxed discrete maximum principle (DMP) in the sense of polynomials. If a cell does not satisfy at least one of the above criteria, it is flagged as *troubled* and its numerical solution is *recomputed* on a finer subgrid with a robust finite volume scheme. Such limiter has also been successfully applied in the context of high order ALE DG numerical schemes on unstructured meshes [18]. On the other hand, in [92, 35] the troubled cells are flagged *a priori*, hence without computing any candidate solution. In those cells the DG polynomial $\mathbf{u}_h$ is projected onto a sub-domain composed of subcells and the solution is then updated with a finite volume scheme. Once the discontinuity has passed away, the DG polynomial is recovered from the subcell averages of the finite volume scheme by a standard conservative finite volume reconstruction operator.

In our approach no finite volume scheme is applied, but we use the algorithm discussed in [106], where the entire polynomials of the DG solutions from the troubled cell and its Neumann neighbors are used in a WENO-type procedure, in order to compute a convex combination based on oscillation indicators and nonlinear weights like in classical high order finite volume schemes [59, 4, 104, 48, 47, 93, 96]. Differently from what is presented in [106], such *a priori* WENO limiter is used on *moving* meshes with a set of basis functions defined in a reference system, which will simplify the procedure of shifting the neighbor polynomials to the troubled cell in the computation of the oscillation indicators.

In the following, two goals will be pursued: the first goal is to describe a flattener function that identifies troubled cells, whereas the second goal is to explain how the "DG-like" coefficients in Eqn. (3) are to be limited.

The first objective is to detect troubled cells, i.e. those regions of the computational domain $\Omega(t)$ which are characterized by strong shocks. The flattener variable described in [5] is used as *shock indicator*, hence yielding an *a priori* limiter. According to [5], a shock can be identified by comparing the divergence of the velocity field $\nabla \cdot \mathbf{v}^n$ with the minimum of the sound speed $c^n_{i,\min}$ obtained by considering the element T^n_i itself as well as its Neumann neighborhood $\mathcal{D}_i$. Thus,

$$\nabla \cdot \mathbf{v}^n = \frac{1}{|T^n_i|} \sum_{T^n_j \in \mathcal{D}_i} |\partial T^n_{ij}| \left(\mathbf{v}^n_j - \mathbf{v}^n_i\right) \cdot \mathbf{n}^n_{ij}, \qquad c^n_{i,\min} = \min_{T^n_j \in \mathcal{D}_i} \left(c^n_i, c^n_j\right), \tag{16}$$

with $|\partial T^n_{ij}|$ denoting the surface shared between elements T^n_i and T^n_j, while $|T^n_i|$ is the volume of cell T^n_i. The associated sound speeds (for hydrodynamics and ideal equation of state) are evaluated as $c^n_{i,j} = \sqrt{\frac{\gamma p^n_{i,j}}{\rho^n_{i,j}}}$ and are a function of the pressure and the density, with γ being the ratio of specific heats of the gas. The divergence of the velocity field is estimated from the *cell-averaged* states $\mathbf{Q}^n_{i,j}$ which correspond to the zeroth order expansion coefficients $\hat{\mathbf{u}}^n_1$ in (3).

The flattener variable f^n_i is evaluated as done in [5] as

$$f^n_i = \min\left[1, \max\left(0, -\frac{\nabla \cdot \mathbf{v}^n + k_1 c^n_{i,\min}}{k_1 c^n_{i,\min}}\right)\right], \tag{17}$$

with the coefficient k_1 that is set to the value of $k_1 = 0.1$ for all our computations. For the sake of safety, even those elements that are about to be crossed by a shock, but have still to enter the wave, should be marked as troubled by the flattener. Therefore, according to what already done in [17], we also compute a *node based* flattener $\tilde{f}^n_k$ by considering the Voronoi neighborhood $\mathcal{V}_k$ of node k:

$$\tilde{f}^n_k = \max_{j \in \mathcal{V}_k} f^n_j. \tag{18}$$

Then, the node based flattener is gathered back into a cell based flattener simply by taking again the maximum value among the set $\mathcal{K}_i$ of the vertices that define the element T_i^n, i.e.

$$f_i^n = \max_{k \in \mathcal{K}_i} \tilde{f}_k^n. \tag{19}$$

Here, the flattener variable is interpreted as an *indicator*, therefore the cell is flagged as troubled if $f_i^n > 0$. Note that in the case of rarefaction waves, where the divergence of the velocity field is positive in (17), and when shocks of modest strength occur, that is $-k_1 c_{i,\min}^n \leq \nabla \cdot \mathbf{v}^n \leq 0$, the flattener variable is $f_i^n = 0$. Thus, the high order DG polynomial remains untouched and those cells are not limited.

Now we will discuss the details related to the second objective, which is to limit the polynomials in the troubled elements using a WENO technique along the line of [106]. The WENO stencil $\mathcal{S}_i$ for cell T_i is composed of the element itself and its Neumann neighbors T_j^n, that is

$$\mathcal{S}_i = T_i^n \ \cup \left(T_j^n \in \mathcal{D}_i \right). \tag{20}$$

Let $\mathbf{u}_{h,i}$ and $\mathbf{u}_{h,j}$ be the $(N+1)$-th order numerical solution in T_i^n and T_j^n, respectively. To ensure conservation, the integral average of all polynomials belonging to $\mathcal{S}_i$ must be equal to the integral average of cell T_i^n, as usually required by finite volume reconstruction techniques. Because of the use of hierarchical basis functions ψ_l on the reference system T_E, the following relation holds:

$$\frac{1}{|T_i^n|} \int_{T_i^n} \mathbf{u}_{h,i} \, dV = \hat{\mathbf{u}}_{1,i}^n, \tag{21}$$

and all the remaining higher order modes vanish. As a consequence, the neighbor polynomials $\mathbf{u}_{h,j}$ can be shifted to the cell T_i^n simply by substituting their cell averages with the one of the troubled cell and leaving untouched the higher order modes. Such modified polynomials are denoted by $\tilde{\mathbf{u}}_{h,j}$, they have the *same* mean value of T_i^n and are expressed in terms of the expansion coefficients as

$$\left.\begin{array}{lll} \tilde{\mathbf{u}}_{l,j}^n = \hat{\mathbf{u}}_{l,i}^n & \text{if} & l = 1 \\ \tilde{\mathbf{u}}_{l,j}^n = \hat{\mathbf{u}}_{l,j}^n & \text{if} & l \in [2, \mathcal{N}] \end{array}\right\} \qquad \forall T_j^n \in \mathcal{D}_i. \tag{22}$$

To be consistent with the notation, let us also define $\tilde{\mathbf{u}}_{h,i}^n = \hat{\mathbf{u}}_{l,i}^n$, so that all shifted polynomials which are used in the WENO limiter have the symbol $\tilde{}$. The limited DG solution $\mathbf{u}_h$ for cell T_i is computed by weighting the above-defined stencil polynomials $\tilde{\mathbf{u}}_h$ in a nonlinear way, where the nonlinearity is introduced in the WENO weights ω_s

$$\tilde{\omega}_s = \frac{\lambda_s}{(\sigma_s + \epsilon)^r}, \qquad \omega_s = \frac{\tilde{\omega}_s}{\sum_q \tilde{\omega}_q}, \tag{23}$$

through the oscillation indicators σ_s with $s \in [1, \mathcal{D}_i + 1]$. The oscillation indicators are computed in the *reference system* as

$$\sigma_s = \Sigma_{lm,s} \, \tilde{\mathbf{u}}_{l,s}^n \, \tilde{\mathbf{u}}_{m,s}^n, \qquad l, m \in [2, \mathcal{N}], \tag{24}$$

with the oscillation indicator matrix

$$\Sigma_{lm} = \sum_{1 \leq \alpha+\beta+\gamma \leq N} \int_{T_E^j} \frac{\partial^{\alpha+\beta+\gamma} \phi_l(\xi,\eta,\zeta)}{\partial \xi^\alpha \partial \eta^\beta \partial \zeta^\gamma} \cdot \frac{\partial^{\alpha+\beta+\gamma} \phi_m(\xi,\eta,\zeta)}{\partial \xi^\alpha \partial \eta^\beta \partial \zeta^\gamma} d\xi d\eta d\zeta, \tag{25}$$

which does not depend on the mesh, but only on the reconstruction basis functions. As such, it can be conveniently precomputed once in a pre-processing stage, and used throughout the entire computation. This holds for moving meshes as well, since the reference element does not change. Finally, the volume of the element does not need to be taken into account in (25) because scaling is already taken out of the problem thanks to the formulation in the reference system. The integral appearing in (25) is evaluated on element T_E^j, which is defined by the reference coordinates $\boldsymbol{\xi}(T_i^n)$ of element T_i^n mapped with respect to element T_j^n in (15), as explained in Figure 3. In other words,

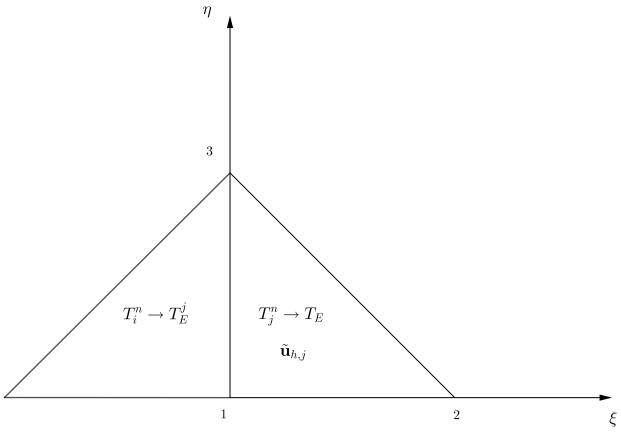

Figure 3: Reference configuration for the computation of the oscillation indicator matrix. Element T_j^n is mapped to the reference element T_E, then the oscillation indicator matrix corresponding to the DG solution $\tilde{\mathbf{u}}_{h,j}$ is evaluated over the *mapped* reference element T_E^j, which is nothing but element T_i^n mapped w.r.t. element T_j^n.

the oscillation indicators are computed by shifting the DG solution from element T_j^n to element T_i^n, which means that the DG polynomial $\tilde{\mathbf{u}}_{h,j}$ is extended over cell T_i^n for computing the oscillation indicators in (25).

Here, we use $\epsilon = 10^{-12}$, $r = 4$, and the linear weights

$$\begin{aligned} \lambda_s &= 1 - \left(10^{-2} \cdot \mathcal{D}_i\right) \quad \text{if} \quad s = 1, \\ \lambda_s &= (1 - \lambda_1) \cdot \mathcal{D}_i^{-1} \quad \text{if} \quad s \in [2, \mathcal{D}_i + 1], \end{aligned} \tag{26}$$

thus ensuring $\sum_s \lambda_s = 1$. The final nonlinear DG-*limited* polynomial $\mathbf{u}_h$ at time t^n is then given by

$$\mathbf{u}_h(\mathbf{x}, t^n) = \sum_{s=1}^{\mathcal{D}_i+1} \omega_s \, \tilde{\mathbf{u}}_{h,j}. \tag{27}$$

2.1.2. WENO Adaptive-Order reconstruction for P_M

Once the DG polynomials $\mathbf{u}_h$ have passed the WENO limiting procedure previously described, two different set of cells can be identified. Those cells, which have not been detected as troubled, have been left with its unlimited numerical solution, while the remaining ones, for which the flattener detector has been activated, have been limited by a WENO-type technique. Thus, at this point the current numerical solution $\mathbf{u}_h$ can be assumed as *valid* within each control volume, that is all elements are given a non-oscillatory polynomial.

According to [44], on the top of these data a reconstruction operator is applied in order to achieve an even higher order of accuracy for the spatial discretization at time t^n. Such reconstruction polynomials $\mathbf{w}_h$ are of degree $M \geq N$ and will constitute the starting point in the local predictor step (presented in next Section 2.2) for achieving a complete space-time accurate discretization of the numerical solution $\mathbf{q}_h$.

The reconstruction procedure is carried out through the following steps:

- definition of the reconstruction stencils that count one big central stencil and other sectorial stencils which span all space directions around a cell;
- computation of an unlimited high order reconstruction polynomial for each stencil;
- estimation of the oscillation of each polynomial in order to seek the less oscillatory polynomial;

- computation of the nonlinear weights which depend on the oscillation indicator of each polynomial;
- hybridization of the stencil polynomials that results in the final limited (WENO-AO) reconstruction polynomial.

The goal of this Sub-section is to describe how the higher order coefficients in Eqn. (5) are obtained. In other words, since the basis functions $\psi_l(\mathbf{x})$ in (5) extend the set of basis functions $\phi_l(\mathbf{x})$ in (3), we wish to describe how the additional expansion coefficients of the extended basis functions are obtained. In this work we use a slightly modified version of the WENO Adaptive-Order (AO) reconstruction [7], recently extended to fixed and moving unstructured meshes in [6] for finite volume numerical schemes. The reconstruction polynomial is computed as a nonlinear hybridization between a large, centered, high accuracy stencil and a lower order WENO scheme in order to provide stability. Physically meaningful extrema can be nonetheless captured and such schemes are computationally less expensive than the WENO reconstruction algorithms developed in [46, 48, 47], where a total number of six (in 2D) or nine (in 3D) stencils with polynomial degree M have to be computed. A similar approach has also been proposed in [69, 33, 34, 45] with Central WENO (CWENO) schemes, which make use of one central high order accurate stencil and a total number of $\mathcal{D}_i$ lateral second order accurate stencils which are then nonlinearly hybridized for stabilizing the reconstruction scheme. Due to the compactness of the reconstruction stencils and to their high robustness, such CWENO finite volume schemes have also been used as *a posteriori* subcell limiters for DG schemes in [23]. A central ENO scheme is presented in [24] for computing a reconstruction polynomial of order M on tetrahedral meshes, that can be seen as a P_0P_M method.

The starting point of the reconstruction is the computation of an *optimal* polynomial $\mathbf{P}_{\text{opt}}$ of degree M with a total number of unknown expansion coefficients $\mathcal{M}$, according to (4). Such a polynomial is expressed using again the Dubiner-type basis functions ψ_l in the reference system $\boldsymbol{\xi}$ on T_E, thus

$$\mathbf{P}_{\text{opt}}(\mathbf{x}, t^n) = \sum_{l=1}^{\mathcal{M}} \psi_l(\boldsymbol{\xi})\, \hat{\mathbf{p}}_{l,i}^n, \tag{28}$$

where the mapping to the reference coordinate system is given by (15) and $\hat{\mathbf{p}}_{l,i}$ denote the unknown expansion coefficients. The reconstruction procedure is based on L_2-projection, thus leading to a conservation property due to the choice of hierarchical orthogonal basis functions. Indeed, $\mathbf{P}_{\text{opt}}$ is evaluated by imposing that the first $\mathcal{N}$ expansion coefficients match *exactly* those ones of element T_i^n, so that the underlying DG solution $\mathbf{u}_h$ is conserved in cell T_i^n even by the higher order polynomial $\mathbf{P}_{\text{opt}}$ of degree M. As a consequence, the unknowns are the remaining $\mathcal{M} - \mathcal{N}$ degrees of freedom which must be determined to define $\mathbf{P}_{\text{opt}}$. To this purpose, a *central* reconstruction stencil $\mathcal{S}_i^0$ is built, which contains a total number of elements n_e, that should theoretically be equal to the smallest number $\frac{\mathcal{M}}{\mathcal{N}}$ needed to reach the formal order of accuracy $M+1$. However, in [10, 83, 63] it has been shown that this choice is not appropriate for unstructured meshes. Therefore n_e is typically taken to be $n_e = d \cdot \frac{\mathcal{M}}{\mathcal{N}}$ and the central stencil is then defined as

$$\mathcal{S}_i^0 = \bigcup_{k=1}^{n_e} T_{j(k)}^n, \tag{29}$$

where $j = j(k)$ denotes a mapping from the set of integers $k \in [1, n_e]$ to the global indices j of the cells in the mesh. The first element in the stencil $\mathcal{S}_i^0$ is always the element itself T_i^n, hence $j(1) = i$, while the remaining elements are added in a recursive manner until the desired number n_e is reached. The unknown degrees of freedom $\hat{\mathbf{p}}_{l,i}^n$ for $l \in [\mathcal{N}+1, \mathcal{M}]$ are determined by solving the following *overdetermined* linear system, since $n_e > \frac{\mathcal{M}}{\mathcal{N}}$:

$$\begin{aligned}
\mathbf{P}_{\text{opt}} &= \underset{\mathbf{p}\in\mathcal{P}_i}{\operatorname{argmin}} \sum_{T_j^n \in \mathcal{S}_i^0} \left(\frac{1}{|T_j^n|} \int_{T_j^n} \mathbf{u}_h(\mathbf{x}) d\mathbf{x} - \frac{1}{|T_j^n|} \int_{T_j^n} \mathbf{p}(\mathbf{x}) d\mathbf{x} \right)^2, \quad \text{with} \\
\mathcal{P}_i &= \left\{ \mathbf{p} \in \mathbb{P}_M : \frac{1}{|T_i^n|} \int_{T_i^n} \mathbf{u}_h(\mathbf{x}) d\mathbf{x} = \frac{1}{|T_i^n|} \int_{T_i^n} \mathbf{p}(\mathbf{x}) d\mathbf{x} \right\} \subset \mathbb{P}_M,
\end{aligned} \tag{30}$$

with $\mathbb{P}_M$ representing the set of all polynomials of degree at most M. The polynomial $\mathbf{P}_{\text{opt}}$ must recover exactly the cell average on the cell T_i^n (which corresponds to the first degree of freedom in the expansion (3)), and it matches all

the other cell averages in the stencil in the least-squares sense. The integrals in (30) are computed using Gaussian quadrature rules of suitable order and the reconstruction matrix arising from system (30) is entirely defined in the reference system $\boldsymbol{\xi}$, so that ill-conditioning due to scaling effects is prevented. A constrained least-squares technique (LSQ) is employed for the solution of (30), following [47].

The WENO Adaptive-Order technique requires the computation of an additional *third order* accurate polynomial evaluated on a *smaller* central stencil and a total number of $\mathcal{D}_i$ *sectorial* polynomials of the same degree $M^s = 2$. Therefore, a total number of $N_p = (\mathcal{D}_i + 1)$ additional polynomials has to be determined by considering N_p stencils S_i^s with $s \in [1, N_p]$. Each sectorial polynomial is again defined in the reference system with the Dubiner-type basis functions $\psi_l(\boldsymbol{\xi})$, counting a total number of degrees of freedom $\mathcal{M}^s$, evaluated according to (4) with M^s. Thus, stencil S_i^s is recursively filled with a total number of $\hat{n}_e = d \cdot \frac{\mathcal{M}^s}{\mathcal{N}}$ elements and includes again the central cell T_i^n as first element. The lateral stencils contains only those elements whose barycenter lies in the open cone defined by one vertex and the opposite face of element T_i^n, as done in [6, 45]. Then, for each stencil S_i^s a third order polynomial $\mathbf{P}_s \in \mathbb{P}_2$ is computed by solving an analogous constrained linear system of (30).

As done in [6], a set of linear weights λ_s is defined in such a way that $\sum_s \lambda_s = 1$. Specifically, we set

$$\begin{aligned} &\lambda_s = 1 - \left(10^{-2} \cdot \mathcal{D}_i\right) && \text{if} \quad s = 0, \\ &\lambda_s = (1 - \lambda_0)\, \lambda_{Lo} && \text{if} \quad s = 1, \\ &\lambda_s = (1 - \lambda_0)\,(1 - \lambda_{Lo}) \cdot \mathcal{D}_i^{-1} && \text{if} \quad s \in [2, \mathcal{D}_i + 1], \end{aligned} \tag{31}$$

with $\lambda_{Lo} = 0.95$. λ_0 refers to the central high order stencil, λ_1 refers to the central smaller third order stencil and the remaining λ_s weights are used for the lateral third order $\mathcal{D}_i$ stencils. Let us define a polynomial $\mathbf{P}_0$ as

$$\mathbf{P}_0 = \frac{1}{\lambda_0}\left(\mathbf{P}_{\text{opt}} - \sum_{l=1}^{N_p} \lambda_l \mathbf{P}_l\right) \in \mathbb{P}_M, \tag{32}$$

so that the polynomial $\mathbf{P}_{\text{opt}}$ is exactly recovered by a linear combination of the polynomials $\mathbf{P}_0, \ldots, \mathbf{P}_{N_p}$. The reconstruction polynomial $\mathbf{w}_h(\mathbf{x}, t^n)$ for cell T_i^n is computed in a standard WENO-type manner by nonlinearly hybridizing all polynomials $\mathbf{P}_s$ with $s \in [0, N_p]$ as

$$\mathbf{w}_h(\mathbf{x}, t^n) = \sum_{l=0}^{N_p} \omega_s \mathbf{P}_s(\mathbf{x}). \tag{33}$$

The nonlinear weights ω_s are given by (23) with the same choice of the parameters $\epsilon = 10^{-12}$ and $r = 4$. The oscillation indicators are given by (24) with the oscillation matrix evaluated with (25) on the reference element T_E (not on the shifted reference element T_E^j) for each polynomial $\mathbf{P}_s$. In smooth regions where $\omega_s \simeq \lambda_s$ the optimal polynomial $\mathbf{P}_{\text{opt}}$ is recovered, i.e. $\mathbf{w}_h \simeq \mathbf{P}_{\text{opt}}$, while if high oscillatory behavior is detected by the oscillation indicators, then $\omega_s \simeq 0$ and the non-oscillatory property is granted to the reconstruction polynomial $\mathbf{w}_h(\mathbf{x}, t^n)$ by using only the least oscillatory sectorial lower order polynomials $\mathbf{P}_s$.

If the mesh is fixed, then all reconstruction matrices in (30) are pre-computed and stored once and for all in the preprocessing stage, while in the ALE context they must be assembled and inverted on-the-fly at each time step for the solution of the associated linear systems. Nevertheless, the use of a WENO Adaptive-Order strategy reduces the number of reconstruction stencils as well as the number of elements contained in each stencil if compared to standard WENO reconstruction techniques [46, 48, 47, 93, 96, 14, 17]. Another reconstruction strategy is given by the *a posteriori* MOOD paradigm [29, 36, 37, 76, 22, 21, 13], which has proved to be very efficient because it only requires *one reconstruction stencil* for both 2D and 3D computational grids.

Eqns. (32) and (33) describe the smallest/simplest nonlinear hybridization that can be made between a higher order reconstruction polynomial and a set of lower order polynomials. However, other alternatives have been recently suggested that would allow the reconstructed polynomial to achieve a sequence of orders. To that end, please see Section from [7] and also [105, 1, 68]. In this work we content ourselves with the simplest nonlinear hybridization because it provides a simple strategy that works. It can, indeed, be upgraded to a more sophisticated adaptive order WENO formulation if one is willing to incorporate more computational complexity in the numerical scheme. In [6] the computational efficiency of the WENO-AO reconstruction is compared against a standard WENO procedure, showing the superior performance of the WENO-AO approach.

2.2. Local space-time Galerkin predictor in the ALE framework

The reconstruction polynomials $\mathbf{w}_h(\mathbf{x}, t^n)$ are now used as starting point for a *local* evolution over one single time step Δt of the numerical solution within each space-time control volume $\tilde{T}_i$. The result will then be stored as piecewise space-time polynomials $\mathbf{q}_h(\mathbf{x}, t^n)$. This strategy has been first introduced for the Eulerian framework in [44, 46], and subsequently extended to ALE schemes on moving meshes in [16, 17, 14]. It is based on a local weak formulation of (1), that is

$$\int_{\tilde{T}_i} \theta_k \frac{\partial \mathbf{q}_h}{\partial t} \, d\mathbf{x} \, dt + \int_{\tilde{T}_i} \theta_k \, \nabla \cdot \mathbf{F}(\mathbf{q}_h) \, d\mathbf{x} \, dt = \int_{\tilde{T}_i} \theta_k \cdot \mathbf{S}(\mathbf{q}_h) \, d\mathbf{x} \, dt, \tag{34}$$

where $\theta_k(\mathbf{x}, t)$ is a space-time basis function which is chosen to be equal to the one used for the expansion of the predictor solution $\mathbf{q}_h$ in (6). The *nodal basis* is composed by Lagrange interpolation polynomials passing through a set of space-time nodes that are explicitly defined in [44]. Specifically, the first $\mathcal{M}$ degrees of freedom are defined at time t^n and coincide with the ones of the reconstruction polynomials, therefore

$$\hat{\mathbf{q}}_l = \hat{\mathbf{w}}_l \qquad \forall l \in [1, \mathcal{M}], \qquad \forall T_i \in \Omega_m. \tag{35}$$

The remaining $\mathcal{L} - \mathcal{M}$ unknown degrees of freedom in the approximation (6) must be determined by solving the nonlinear system (34) with Picard iterations.

In the ALE context, the mesh motion has also to be taken into account when integrating the weak formulation (34) over a space-time moving control volume. To this purpose, the geometry of $\tilde{T}_i$ is evolved *together* with the numerical solution $\mathbf{q}_h$ by considering the trajectory equation

$$\frac{d\mathbf{x}}{dt} = \mathbf{V}(\mathbf{x}, t), \tag{36}$$

with $\mathbf{V}(\mathbf{x}, t) = (U, V, W)$ being the local mesh velocity. The same basis functions θ_l used for the solution $\mathbf{q}_h$ are also adopted for approximating the mesh velocity, hence yielding

$$\mathbf{V}_h(\mathbf{x}, t) = \sum_{l=1}^{\mathcal{L}} \theta_l(\mathbf{x}, t) \, \hat{\mathbf{V}}_l \qquad (\mathbf{x}, t) \in \tilde{T}_i. \tag{37}$$

In this way an *isoparametric* approach is used, where the space-time control volume $\tilde{T}_i$ is allowed to involve *curved* boundaries in both space and time for $t^{n+1} \leq t < t^n$. The ODE for the mesh velocity (36) is solved together with the nonlinear system for the numerical solution (34) *iteratively*, until the residuals of the two systems are less than a prescribed tolerance *tol* (typically $tol \approx 10^{-12}$). From numerical experiments it turned out that at most $d \cdot (M + 1)$ iterations are needed to reach convergence. A complete and fully detailed description about the local space-time predictor on moving meshes can be found in [14].

Since no neighbor information is taken into account at this predictor stage, the element volume T_i^{n+1} might also lead to an overlapping or discontinuous mesh configuration $\mathcal{T}_\Omega^{n+1}$.

2.3. Mesh motion

The mesh motion procedure aims at defining a valid mesh configuration $\mathcal{T}_\Omega^{n+1}$ at the new time level, which means that the grid must be always *conforming* and *continuous*. This is why the element-local space-time configurations obtained by the predictor strategy with the local evolution of the geometry (36) are not acceptable, but must be *corrected* in such a way that the final mesh is valid.

As a consequence, a *nodal solver* is typically adopted for moving mesh schemes [79, 55, 26], whose task is the definition of a *unique* velocity vector $\bar{\mathbf{V}}_k$ that must be assigned to each vertex k of the mesh. Then, the old $\mathbf{x}_k^n$ and the new $\mathbf{x}_k^{n+1}$ vertex position are simply connected by the following relation

$$\mathbf{x}_k^{n+1} = \mathbf{x}_k^n + \bar{\mathbf{V}}_k^n \cdot \Delta t. \tag{38}$$

In this work we rely on a mass weighted nodal solver, that follows the work done in [26, 72, 27]. The nodal velocity is evaluated as a mass weighted average velocity among the Voronoi neighborhood $\mathcal{V}_k$ of node k:

$$\bar{\mathbf{V}}_k^n = \frac{1}{\mu_k} \sum_{T_j^n \in \mathcal{V}_k} \mu_{k,j} \bar{\mathbf{V}}_{k,j}, \tag{39}$$

with

$$\mu_k = \sum_{T_j^n \in \mathcal{V}_k} \mu_{k,j}, \qquad \mu_{k,j} = \rho_j^n |T_j^n|. \tag{40}$$

The local velocity contributions $\bar{\mathbf{V}}_{k,j}$ are given by the integration over the time step Δt of the high order vertex-extrapolated velocity coming from the local predictor solution, that is

$$\mathbf{V}_{k,j} = \int_{t^n}^{t^{n+1}} \theta_l(\mathbf{x}_k, t)\, \hat{\mathbf{V}}_{l,j}\, dt, \tag{41}$$

where the expansion (37) has been used for the mesh velocity. Finally, $\mu_{k,j}$ in (40) denotes the mass of the Voronoi neighbor T_j, given by multiplying the cell density ρ_j^n by the cell volume $|T_j^n|$.

If the new vertex location $\mathbf{x}_k^{n+1}$ yields a bad quality element configuration, i.e. a highly stretched or compressed or twisted element shape, a *rezoning* [65, 55] algorithm might be applied in order to improve the local mesh quality. Here, for severe test cases with strong shock waves or complex flow patterns, we rely on the local rezoning strategy proposed in [65] and used in the context of direct ALE schemes in [17].

For a comparison on different nodal solvers the interested reader is referred to [19], while the use of genuinely multidimensional Riemann solvers for the definition of the mesh velocity is presented in [15].

Once a unique velocity vector has been computed for each node of the entire computational mesh, the new mesh configuration $\mathcal{T}_\Omega^{n+1}$ is known and it is connected by *straight lines* with the old mesh configuration $\mathcal{T}_\Omega^n$. Therefore, the space-time control volumes $\tilde{T}_i$ can be constructed and the $P_N P_M$ scheme (8) can be run to perform one time evolution of the numerical solution up to the next time level $\mathbf{u}_h^{n+1}$.

Because the scheme belongs to the category of Arbitrary-Lagrangian-Eulerian (ALE) methods, we underline that the space-time order of accuracy of the scheme is not spoiled by handling the mesh motion with straight segments. Indeed, a pure Eulerian configuration is achieved by setting $\bar{\mathbf{V}}_k^n = \mathbf{0}$ at each node k, hence assuming that $T_i^{n+1} = T_i^n$, i.e. simplex elements remain as such even at the new time level, and performing a time-orthogonal mesh motion, where again the old and new element configurations are connected by straight lines. Similarly, if the mesh moves, the *same* procedure applies, with the only difference that the mesh motion is no longer orthogonal to the time coordinate. Nevertheless, the $P_N P_M$ scheme (8) is not assigned any modification and the mesh motion is *directly* taken into account in the space-time integration of the governing equations over volume $\tilde{T}_i$ (see Figure 1) as well as in the Jacobian matrix of the system (9). In other words, the mesh motion is *independent* from the numerical scheme (8), which can easily adapt to any arbitrary grid velocity.

3. Numerical results

We present a set of benchmark test problems in order to validate the accuracy and the robustness of the high order ALE $P_N P_M$ ("reconstructed DG") schemes discussed in this work. To this purpose, let us consider the Euler equations of compressible gas dynamics, which can be cast into form (1) with the following definitions:

$$\mathbf{Q} = \begin{pmatrix} \rho \\ \rho u \\ \rho v \\ \rho w \\ \rho E \end{pmatrix}, \quad \mathbf{f} = \begin{pmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ \rho uw \\ u(\rho E + p) \end{pmatrix}, \quad \mathbf{g} = \begin{pmatrix} \rho v \\ \rho uv \\ \rho v^2 + p \\ \rho wv \\ v(\rho E + p) \end{pmatrix} \quad \mathbf{h} = \begin{pmatrix} \rho w \\ \rho uw \\ \rho vw \\ \rho w^2 + p \\ w(\rho E + p) \end{pmatrix}, \quad \mathbf{S} = \mathbf{0}. \tag{42}$$

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

$$p = (\gamma - 1)\left(\rho E - \frac{1}{2}\rho(u^2 + v^2 + w^2)\right), \tag{43}$$

with γ representing the ratio of specific heats. The fluid density and pressure are denoted by ρ and p, respectively, while the fluid velocity vector is $\mathbf{v} = (u, v, w)$. The total energy density is given by ρE and the sound speed reads $c = \sqrt{\frac{\gamma p}{\rho}}$. If not specified, we will use the ALE Rusanov-type flux (10) and the initial condition will be expressed in terms of primitive variables $\mathbf{U} = (\rho, u, v, w, p)$. All test cases are run in a quasi-Lagrangian setting, i.e. with $\bar{\mathbf{V}} = \mathbf{v}$, apart from the double Mach reflection problem and the forward facing step problem, where we have set $\bar{\mathbf{V}} = \mathbf{0}$. The mesh velocity is computed by means of the nodal solver (39).

3.1. Numerical convergence studies

The numerical convergence of the new ALE $P_N P_M$ schemes is studied on a smooth isentropic vortex test case [57], that is concerned with a hydrodynamic vorticial flow which is furthermore convected with constant velocity $\mathbf{v}_c = (1, 1, 0)$. The initial computational domain is the box $\Omega(0) = [0; 10]^d$, with periodic boundaries, and is discretized with a sequence of refined unstructured meshes composed of triangles and tetrahedra. The initial condition is given as a homogeneous background field on the top of which some perturbations are added, thus

$$\mathbf{U} = (1 + \delta\rho, 1 + \delta u, 1 + \delta v, 0, 1 + \delta p), \tag{44}$$

with the perturbations for temperature δT, density $\delta\rho$, velocity $(\delta u, \delta v)$ and pressure δp that read

$$\begin{aligned} \delta T &= -\frac{(\gamma-1)\epsilon^2}{8\gamma\pi^2} e^{1-r^2}, & \delta\rho &= (1+\delta T)^{\frac{1}{\gamma-1}} - 1, \\ \begin{pmatrix} \delta u \\ \delta v \end{pmatrix} &= \frac{\epsilon}{2\pi} e^{\frac{1-r^2}{2}} \begin{pmatrix} -(y-5) \\ (x-5) \end{pmatrix}, & \delta p &= (1+\delta T)^{\frac{\gamma}{\gamma-1}} - 1. \end{aligned} \tag{45}$$

The simulation is run until the final time $t_f = 0.1$ and the exact solution is given by the time-shifted initial condition, i.e. $\mathbf{U}_e(\mathbf{x}, t_f) = \mathbf{U}(\mathbf{x} - \mathbf{v}_c t_f, 0)$. The characteristic mesh size $h(\Omega(t_f))$ is assumed to be the maximum diameter of the circumcircles among all the elements of the final grid configuration $\Omega(t_f)$. Tables 1 and 2 contain the errors in L_2 norm referred to the fluid density from third up to fifth order of accuracy. The correct convergence behavior is observed for all schemes spanning from P_2P_2 to P_4P_4. For comparison purposes, we also report in Table 3 the total computational time per element update τ_{EU} for each simulation run in two and three space dimensions. Unlike what discussed in [44], the convergence studies carried out here show the behavior of general $P_N P_M$ schemes when nonlinear hybridization is included. High order finite volume schemes (P_0P_M) are less accurate than mixed $P_N P_M$ schemes, even though the formal order of accuracy is always achieved. Furthermore, one can note that for $N < M$ with N close to M, the errors are almost of the same magnitude as the pure DG scheme, especially for the two-dimensional case. However, the gain in terms of efficiency is not negligible. Indeed, the $P_N P_M$ scheme is much more efficient: for example, the P_3P_4 and P_4P_4 schemes in 2D with $h = 1.63 \cdot 10^{-1}$ need approximately $7.2 \cdot 10^3$ and $10.2 \cdot 10^3$ seconds of computational time, for obtaining the *same* accuracy.

3.2. The Kidder problem

The Kidder problem considers the isentropic compression of a portion of a shell filled with an ideal gas. This is a benchmark for Lagrangian schemes because it allows to verify whether the scheme produces spurious entropy during the isentropic compression, or not. The initial computational domain is bounded by $r_i(t) \le r \le r_e(t)$, with $r = \sqrt{\mathbf{x}^2}$ representing the generic radial coordinate, $r_i(t)$ and $r_e(t)$ being the time-dependent internal and external radius of the shell, respectively. At the initial time $t = 0$ we set $r_{i,0} = 0.9$ and $r_{e,0} = 1.0$, and the gas is initially assigned a uniform entropy $s_0 = \frac{p_0}{\rho_0^\gamma} = 1$ with the adiabatic index $\gamma = 2$ in 2D and $\gamma = 5/3$ in 3D and the initial condition

$$\begin{pmatrix} \rho_0(r) \\ \mathbf{v}_0(r) \\ p_0(r) \end{pmatrix} = \begin{pmatrix} \left(\frac{r_{e,0}^2 - r^2}{r_{e,0}^2 - r_{i,0}^2}\rho_{i,0}^{\gamma-1} + \frac{r^2 - r_{i,0}^2}{r_{e,0}^2 - r_{e,0}^2}\rho_{e,0}^{\gamma-1}\right)^{\frac{1}{\gamma-1}} \\ 0 \\ s_0\rho_0(r)^\gamma \end{pmatrix}. \tag{46}$$

Table 1: Isentropic vortex test case in 2D. Numerical convergence results for the two-dimensional compressible Euler equations using the direct ALE $P_N P_M$ schemes from first up to fifth order of accuracy. The errors are evaluated in L_2 norm and refer to the variable ρ (density) at time $t = 0.1$.

$h(\Omega(t_f))$	ϵ_{L_2}	$O(L_2)$	ϵ_{L_2}	$O(L_2)$	ϵ_{L_2}	$O(L_2)$	ϵ_{L_2}	$O(L_2)$	ϵ_{L_2}	$O(L_2)$
$\mathcal{O}1$	P_0P_0									
6.71E-01	7.12E-02	-								
3.26E-01	3.23E-02	1.1								
2.48E-01	2.27E-02	1.3								
1.63E-01	1.56E-02	0.9								
$\mathcal{O}2$	P_0P_1		P_1P_1							
6.71E-01	3.22E-02	-	2.67E-02	-						
3.26E-01	9.75E-03	1.7	7.13E-03	1.8						
2.48E-01	5.82E-03	1.9	4.06E-03	2.0						
1.63E-01	3.19E-03	1.4	1.77E-03	2.0						
$\mathcal{O}3$	P_0P_2		P_1P_2		P_2P_2					
6.71E-01	7.60E-02	-	1.55E-02	-	4.75E-03	-				
3.26E-01	1.05E-02	2.7	2.05E-03	2.8	6.03E-04	2.9				
2.48E-01	4.06E-03	3.4	7.63E-04	3.6	2.56E-04	3.1				
1.63E-01	1.29E-03	2.8	2.09E-04	3.1	7.04E-05	3.1				
$\mathcal{O}4$	P_0P_3		P_1P_3		P_2P_3		P_3P_3			
6.71E-01	3.99E-02	-	1.21E-02	-	2.55E-03	-	8.47E-04	-		
3.26E-01	2.62E-03	3.8	8.06E-04	3.7	1.47E-04	4.0	5.39E-05	3.8		
2.48E-01	8.81E-04	4.0	2.36E-04	4.5	4.73E-05	4.1	1.63E-05	4.3		
1.63E-01	1.92E-04	3.7	4.86E-05	3.8	9.80E-06	3.8	3.68E-06	3.6		
$\mathcal{O}5$	P_0P_4		P_1P_4		P_2P_4		P_3P_4		P_4P_4	
6.71E-01	4.90E-02	-	4.79E-03	-	1.67E-03	-	5.27E-04	-	2.41E-04	-
3.26E-01	2.60E-03	4.1	2.04E-04	4.4	7.80E-05	4.2	4.02E-05	3.6	2.95E-05	2.9
2.48E-01	6.15E-04	5.2	4.58E-05	5.4	2.22E-05	4.6	1.35E-05	4.0	1.04E-05	3.8
1.63E-01	9.84E-05	4.4	6.28E-06	4.8	3.01E-06	4.8	1.61E-06	5.1	1.30E-06	5.0

Table 2: Isentropic vortex test case in 3D. Numerical convergence results for the three-dimensional compressible Euler equations using the direct ALE $P_N P_M$ schemes from first up to fifth order of accuracy. The errors are evaluated in L_2 norm and refer to the variable ρ (density) at time $t = 0.1$.

$h(\Omega(t_f))$	ϵ_{L_2}	$O(L_2)$	ϵ_{L_2}	$O(L_2)$	ϵ_{L_2}	$O(L_2)$	ϵ_{L_2}	$O(L_2)$	ϵ_{L_2}	$O(L_2)$
$\mathcal{O}1$	P_0P_0									
1.01E-00	2.19E-01	-								
5.92E-01	1.35E-01	0.9								
3.62E-01	9.21E-02	0.8								
2.13E-01	6.47E-02	0.8								
$\mathcal{O}2$	P_0P_1		P_1P_1							
1.01E-00	1.50E-01	-	1.56E-01	-						
5.92E-01	5.25E-02	2.0	4.84E-02	2.2						
3.62E-01	2.36E-02	1.6	2.09E-02	1.7						
2.13E-01	1.23E-02	1.5	9.30E-03	1.8						
$\mathcal{O}3$	P_0P_2		P_1P_2		P_2P_2					
1.01E-00	3.10E-01	-	1.86E-01	-	4.46E-02	-				
5.92E-01	7.43E-02	2.7	1.98E-02	4.2	8.15E-03	3.2				
3.62E-01	1.98E-02	2.7	5.45E-03	2.6	2.04E-03	2.8				
2.13E-01	5.51E-03	2.9	1.59E-03	2.7	5.78E-04	2.8				
$\mathcal{O}4$	P_0P_3		P_1P_3		P_2P_3		P_3P_3			
1.01E-00	2.31E-01	-	8.17E-02	-	3.52E-02	-	1.40E-02	-		
5.92E-01	3.48E-02	3.5	9.07E-03	4.1	3.39E-03	4.4	1.29E-03	4.4		
3.62E-01	5.92E-03	3.6	1.64E-03	3.5	6.24E-04	3.4	2.04E-04	3.7		
2.13E-01	1.15E-03	3.7	3.16E-04	3.7	1.13E-04	3.8	3.36E-05	4.0		
$\mathcal{O}5$	P_0P_4		P_1P_4		P_2P_4		P_3P_4		P_4P_4	
1.01E-00	2.27E-01	-	5.98E-02	-	2.43E-02	-	1.09E-02	-	4.18E-03	-
5.92E-01	2.76E-02	3.9	4.28E-03	4.9	1.60E-03	5.1	6.50E-04	5.2	2.29E-04	5.4
3.62E-01	3.37E-03	4.3	4.83E-04	4.4	1.74E-04	4.5	7.53E-05	4.4	2.63E-05	4.4
2.13E-01	4.15E-04	4.7	6.49E-05	4.5	2.28E-05	4.5	1.01E-05	4.5	3.26E-06	4.7

Table 3: Isentropic vortex test case. Computational time per element update for $P_N P_M$ schemes in 2D and 3D with $[N, M] \in [0, 4]$.

	2D τ_{EU}					**3D** τ_{EU}				
	P_0P_0					P_0P_0				
	3.23E-03					1.23E-03				
$\mathcal{O}1$	9.64E-04					5.01E-04				
	3.18E-04					3.52E-04				
	1.89E-04					3.26E-04				
	P_0P_1	P_1P_1				P_0P_0	P_1P_1			
	3.42E-03	2.33E-03				5.61E-03	6.88E-03			
$\mathcal{O}2$	8.46E-04	7.80E-04				1.82E-03	3.12E-03			
	3.97E-04	5.93E-04				1.54E-03	2.71E-03			
	2.31E-04	3.58E-04				1.52E-03	2.51E-03			
	P_0P_2	P_1P_2	P_2P_2			P_0P_2	P_1P_2	P_2P_2		
	3.64E-03	3.48E-03	7.34E-03			1.29E-02	2.14E-02	2.53E-02		
$\mathcal{O}3$	1.18E-03	1.97E-03	3.35E-03			7.51E-03	1.07E-02	1.27E-02		
	6.18E-04	1.15E-03	1.48E-03			5.06E-03	9.91E-03	1.22E-02		
	3.93E-04	8.31E-04	1.23E-03			7.72E-03	1.27E-02	1.38E-02		
	P_0P_3	P_1P_3	P_2P_3	P_3P_3		P_0P_3	P_1P_3	P_2P_3	P_3P_3	
	4.25E-03	6.77E-03	1.03E-02	1.45E-02		3.35E-02	1.12E-01	1.14E-01	1.37E-01	
$\mathcal{O}4$	1.54E-03	4.55E-03	4.72E-03	4.80E-03		1.71E-02	6.46E-02	6.51E-02	7.72E-02	
	9.58E-04	3.09E-03	3.36E-03	3.65E-03		1.61E-02	6.29E-02	1.88E-02	2.26E-02	
	6.81E-04	2.76E-03	2.90E-03	3.26E-03		1.47E-02	6.23E-02	1.81E-02	2.21E-02	
	P_0P_4	P_1P_4	P_2P_4	P_3P_4	P_4P_4	P_0P_4	P_1P_4	P_2P_4	P_3P_4	P_4P_4
	5.93E-03	2.22E-02	2.64E-02	2.22E-02	2.42E-02	9.91E-02	6.92E-01	6.93E-01	7.13E-01	8.77E-01
$\mathcal{O}5$	2.58E-03	1.76E-02	1.75E-02	1.85E-02	1.99E-02	5.86E-02	4.28E-01	4.08E-01	1.34E-01	1.59E-01
	1.67E-03	1.22E-02	1.25E-02	1.29E-02	1.40E-02	5.57E-02	4.23E-01	1.37E-01	1.31E-01	5.35E-01
	1.30E-03	1.17E-02	1.18E-02	1.22E-02	1.33E-02	5.05E-02	4.20E-01	4.10E-01	4.43E-01	5.13E-01

The initial values of density read $\rho_{i,0} = 1$ and $\rho_{e,0} = 2$, defined at $r_{i,0}$ and $r_{e,0}$, respectively. According to [64], the final time is evaluated as $t_f = \frac{\sqrt{3}}{2}\tau$ with the focalisation time $\tau^{2D} = 0.217944947177$ and $\tau^{3D} = 0.254345722569$. A self-similar analytical solution $R(r,t)$ has been provided in [64], with the exact location of the shell at the final time bounded by $0.45 \leq R \leq 0.5$. Consequently, the absolute error $|err|$ between analytical and numerical solution is easily computed and reported in Table 4, while L_1, L_2 and L_∞ norms are shown in Table 5. Sliding wall boundary conditions have been set on the lateral faces, while on the internal and on the external frontier a space-time dependent state is assigned according to the exact analytical solution $R(r,t)$. Figure 4 shows the density distribution and the mesh configuration at the initial and final time of the simulation as well as the evolution of the internal and external frontier. In Figure 5 a scatter plot of density is depicted for the two-dimensional simulation at output times $t = 0.09$ and $t = 0.18$, while for the three-dimensional case at output times $t = 0.11$ and $t = 0.22$. A P_2P_4 scheme with the Osher-type (11) numerical flux has been used in 2D, whereas in 3D we set $N = M = 3$, hence recovering a pure DG scheme. No cells are affected by the WENO limiter because the Kidder problem does not involve any discontinuity.

2D P_2P_4		
r_{ex}	r_{num}	$\|err\|$
0.450000	0.44999	6.6E-6
0.500000	0.50007	7.7E-5

3D P_3P_3		
r_{ex}	r_{num}	$\|err\|$
0.450000	0.44730	2.7E-3
0.500000	0.50076	7.6E-4

Table 4: Kidder problem. Absolute error for the internal and external radius location between exact (r_{ex}) and numerical (r_{num}) solution.

2D P_2P_4	L_1	L_2	L_∞
ρ	1.2386E-4	6.4975E-4	7.2564E-3
u	5.1999E-5	3.0618E-4	1.7031E-2

3D P_3P_3	L_1	L_2	L_∞
ρ	3.3686E-6	3.2567E-5	3.7945E-3
u	5.8022E-7	6.2654E-6	1.6739E-3

Table 5: Kidder problem. L_1, L_2 and L_∞ norms for density ρ and horizontal velocity u.

3.3. The Saltzman problem

A rectangular channel $\Omega(0) = [0;1] \times [0;0.1]$ is filled with a perfect gas with $\gamma = \frac{5}{3}$. The left side of the computational domain is a piston which moves with velocity $\mathbf{v}_p = (1,0,0)$ towards the right direction, thus compressing the gas. A strong shock wave is then generated, traveling faster than the piston. According to [42], the computational mesh is composed by $N_E = 2000$ right triangles, that have been drawn as follows:

- first a Cartesian mesh with 100×10 square elements is built [79, 72];
- each square element is then split into two right triangles;
- finally, the uniform grid, defined by the coordinate vector $\mathbf{x} = (x, y)$, is skewed with the mapping

$$\begin{aligned} x' &= x + (0.1 - y)\sin(\pi x), \\ y' &= y, \end{aligned} \tag{47}$$

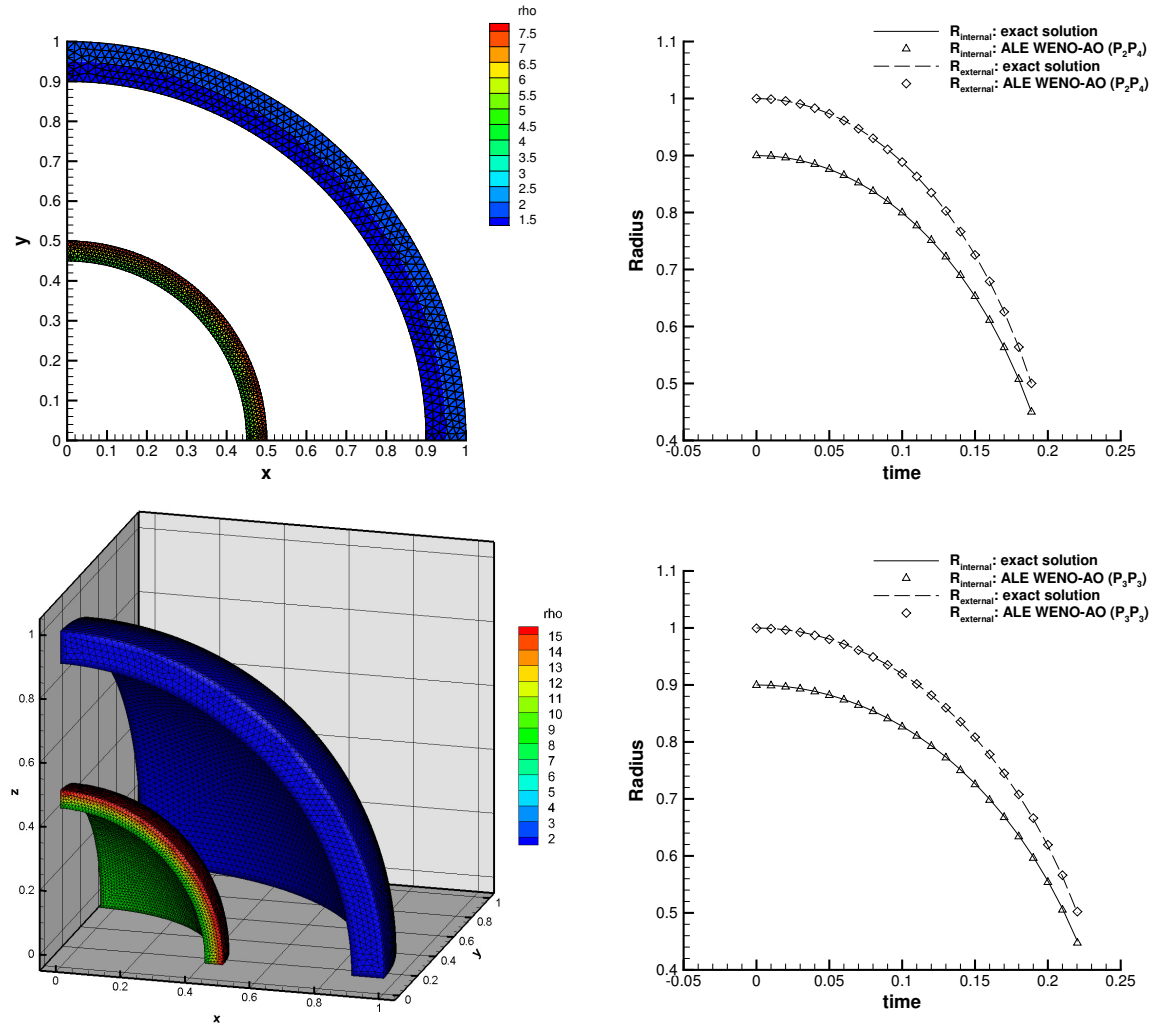

Figure 4: Kidder problem in 2D (top row) and in 3D (bottom row). Left: initial and final computational domain with density distribution. Right: evolution of the internal and external radius of the shell and comparison between analytical and numerical solution.

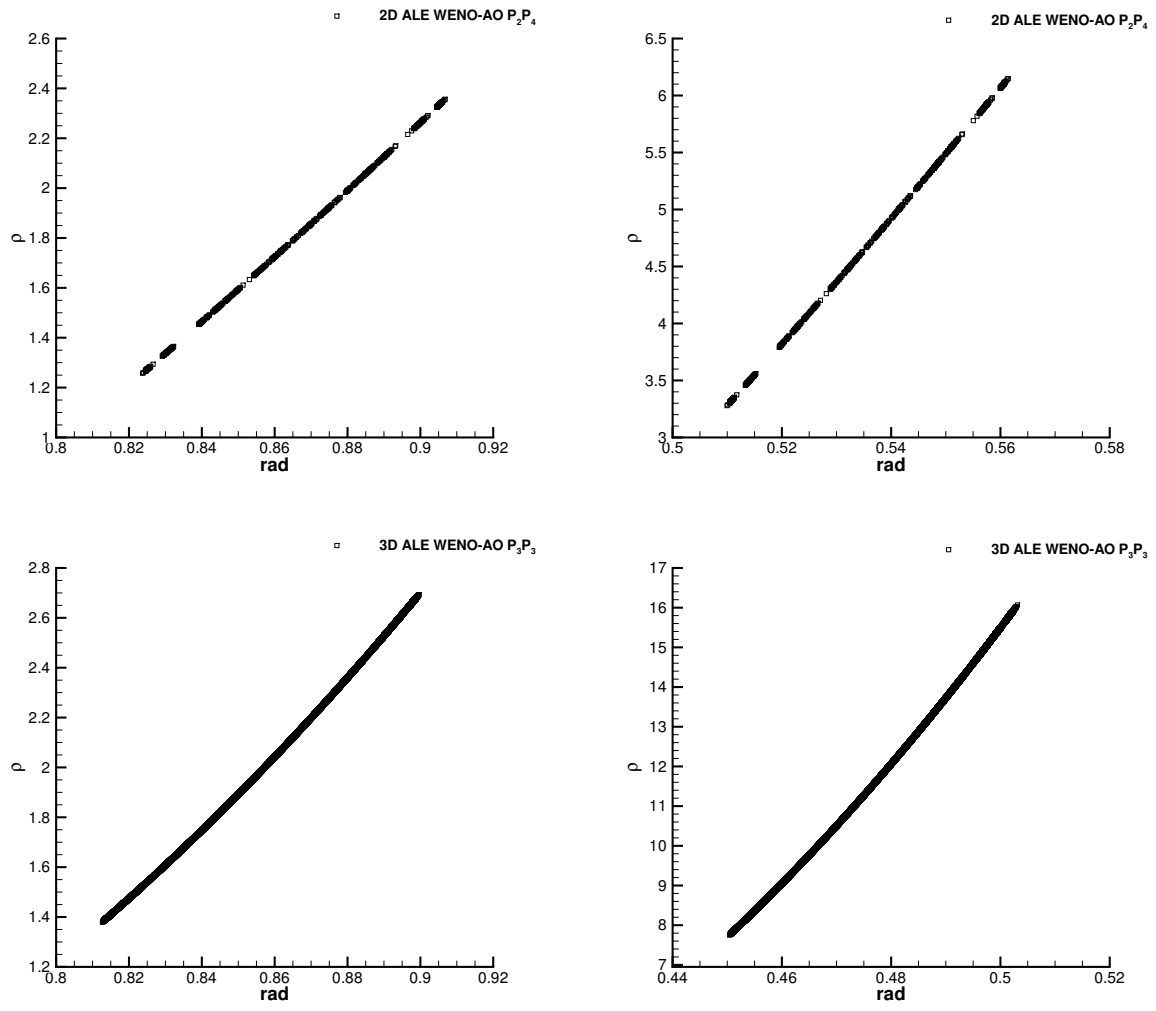

Figure 5: Scatter plot of density distribution for the Kidder problem. Top: 2D with P_2P_4 at output times $t = 0.09$ (left) and $t = 0.18$ (right). Bottom: 3D with P_3P_3 at output times $t = 0.11$ (left) and $t = 0.22$ (right).

where x' and y' represent the deformed coordinates, respectively. We impose sliding wall boundary conditions on all sides, except for the piston, where a moving slip wall boundary condition is considered. The gas is initially at rest, with density $\rho_0 = 1$ and pressure $p_0 = 10^{-4}$ [72]. A fifth order scheme P_2P_4 is used to run the Saltzman problem. A scatter plot of the cell density and velocity is plot in Figure 6 at the final time $t_f = 0.6$, together with a comparison against the exact solution, which is given by a one-dimensional infinite strength shock wave with a post shock density of $\rho_e = 4.0$ and the shock front located at $x = 0.8$, see [16]. The shock wave is precisely identified by the troubled cell indicator, as shown in Figure 6, while the mesh configuration is of reasonably good quality thanks to the rezoning strategy [55, 14]. Furthermore, we show the map of troubled cells, the density distribution and the mesh configuration at output times $t = 0.6$, $t = 0.75$ and $t = 0.9$ in Figure 7: the shock is again properly identified by the flattener indicator, even for times beyond $t = 0.6$. Specifically, at time $t = 0.75$ the shock wave has reached the final wall of the domain, hence being reflected towards the piston, which is hit at time $t = 0.9$.

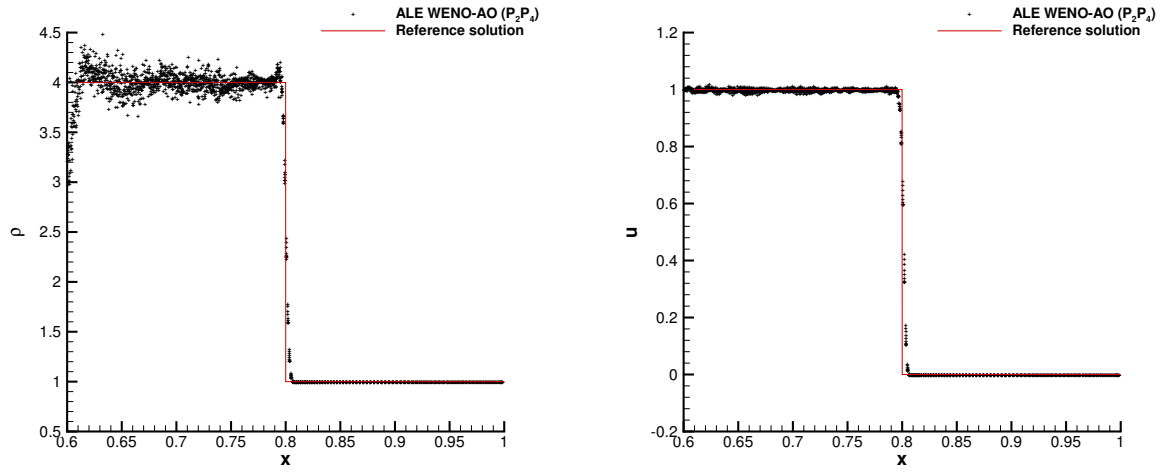

Figure 6: Saltzman problem in 2D with P_2P_4: scatter plot of the cell density (left) and horizontal velocity (right) as a function of cell horizontal coordinate x versus the exact solution.

3.4. The Sedov problem

The Sedov problem involves the evolution of a blast wave with cylindrical symmetry. The initial computational domain is a circle of radius $R_0 = 1.2$ with transmissive boundaries, which has been discretized with a total number of $N_E = 11482$ triangles. This test case constitutes a benchmark in literature [79, 80, 77] since it allows the algorithm to be tested against strong element compressions produced by the diverging shock wave. The initial condition in primitive variable simply reads $\mathbf{U}_0 = (1, 0, 0, 0, p_0)$, where the initial pressure is $p_0 = 10^{-6}$ everywhere except for the cells c_{or} containing the origin $\mathbf{O} = (\mathbf{x}) = (0)$ of the domain where we assign

$$p_{or} = (\gamma - 1)\rho_0 \frac{E_{tot}}{V_{or}} \quad \text{with} \quad E_{tot} = 0.979264. \tag{48}$$

The total energy density is E_{tot} and induces the shock wave, while V_{or} is the volume of the cells that share the origin. The ratio of specific heats is $\gamma = 1.4$ and the final time of the simulation is set to $t_f = 1$. The exact solution based on self-similarity arguments has been computed in [61] and consists in a shock wave with a density peak of $\rho = 6$ located at radius $R = \sqrt{\mathbf{x}^2} = 1$. A fifth order P_2P_4 scheme is employed with the Rusanov-type flux (10). For the three-dimensional case, a cubic domain $\Omega(0) = [0; 1]^3$ with a total number of $N_E = 40000$ tetrahedra is considered, with symmetric boundary conditions on the internal walls and transmissive boundaries elsewhere. In 3D we set $E_{tot} = 0.851072$ and a P_2P_2 scheme is used with the Rusanov numerical flux (10). Figure 8 plots a scatter plot of cell density and a map of the troubled cells, which are only detected across the shock wave. Furthermore, the symmetry of

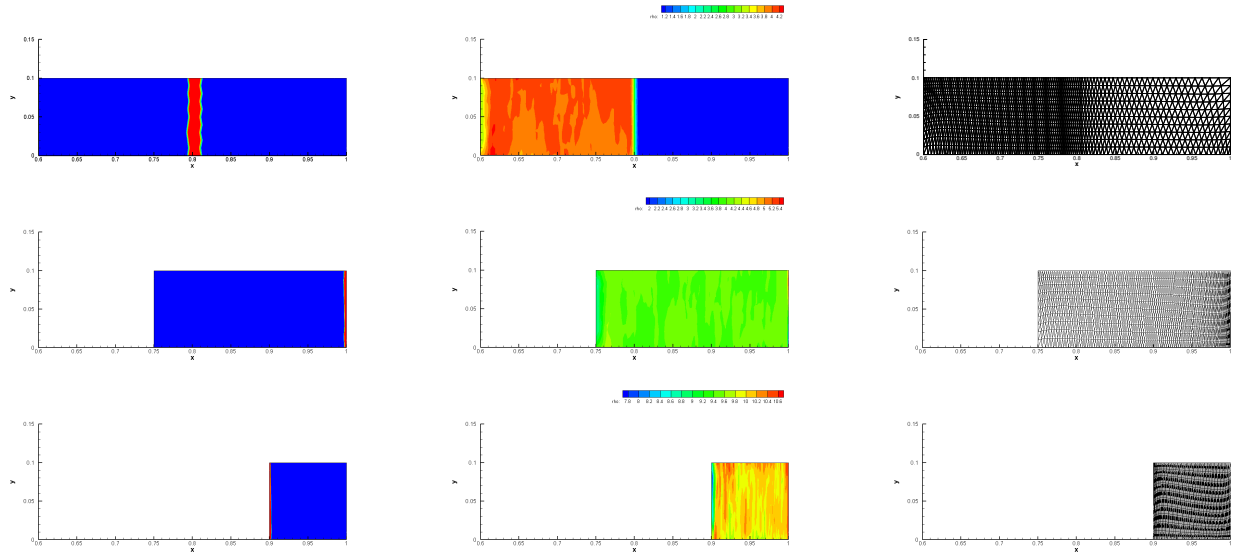

Figure 7: Saltzman problem in 2D with P_2P_4: map of troubled cell indicator (left), density distribution (middle) and final mesh configuration (right) at output times $t = 0.6$ (top), $t = 0.75$ (middle) and $t = 0.9$ (bottom).

the cylindrical shock wave is very well preserved, without spreading the numerical solution, as evident from the cell density plot.

3.5. *The Noh problem*

The Noh problem [82] allows any Lagrangian scheme to be validated in the regime of strong shock waves. The initial computational domain is given by $\Omega(0) = [0; 1]^2$ and the characteristic mesh size is $h = 1/40$. The computational grid is composed quadrilateral elements that have been split into two triangles each. Thus, the total number of simplex control volumes counts $N_E = 3200$. A gas with $\gamma = 5/3$ is initially assigned with a unity density $\rho_0 = 1$ and a unity radial velocity which is moving the gas towards the origin of the domain $O = (0, 0)$. The initial velocity components are initialized with

$$u_0 = -\frac{x}{r}, \qquad v_0 = -\frac{y}{r}, \tag{49}$$

with r denoting the radial position $r = \sqrt{x^2 + y^2}$. The pressure is initially set equal to $p = 10^{-6}$ everywhere. As time advances, an outward moving shock wave is generated which travels with velocity $v_{sh} = 1/3$ in radial direction. The final time is $t_f = 0.6$ according to [82, 79, 80], so that the shock wave is located at radius $R = 0.2$ and the maximum density value is $\rho_f = 16$, which occurs on the plateau behind the shock wave. We impose no-slip wall boundary conditions on all sides of the domain which contains the origin $O = (0, 0)$, while moving boundaries have been used on the remaining sides.

Figure 9 shows the final mesh configuration with the map of troubled cells as well as a scatter plot of the final density distribution compared with the exact solution. A P_2P_4 scheme with the Rusanov-type numerical flux (10) has been used to run the Noh problem. The rezoning strategy proposed in [55] has been adopted in order to maintain a good mesh quality. The entire region before the shock frontier is marked as troubled, since condition (17) is always verified in that zone of the computational domain, due to the initial velocity field (49) which gives rise to a strong compression.

3.6. *2D Taylor-Green vortex*

The Taylor-Green problem consists of a smooth vortex, hence representing a vortical flow problem for which an analytical solution is available [54]. The computational domain is defined by the square $\Omega(0) = [0; 1] \times [0; 1]$ and it is

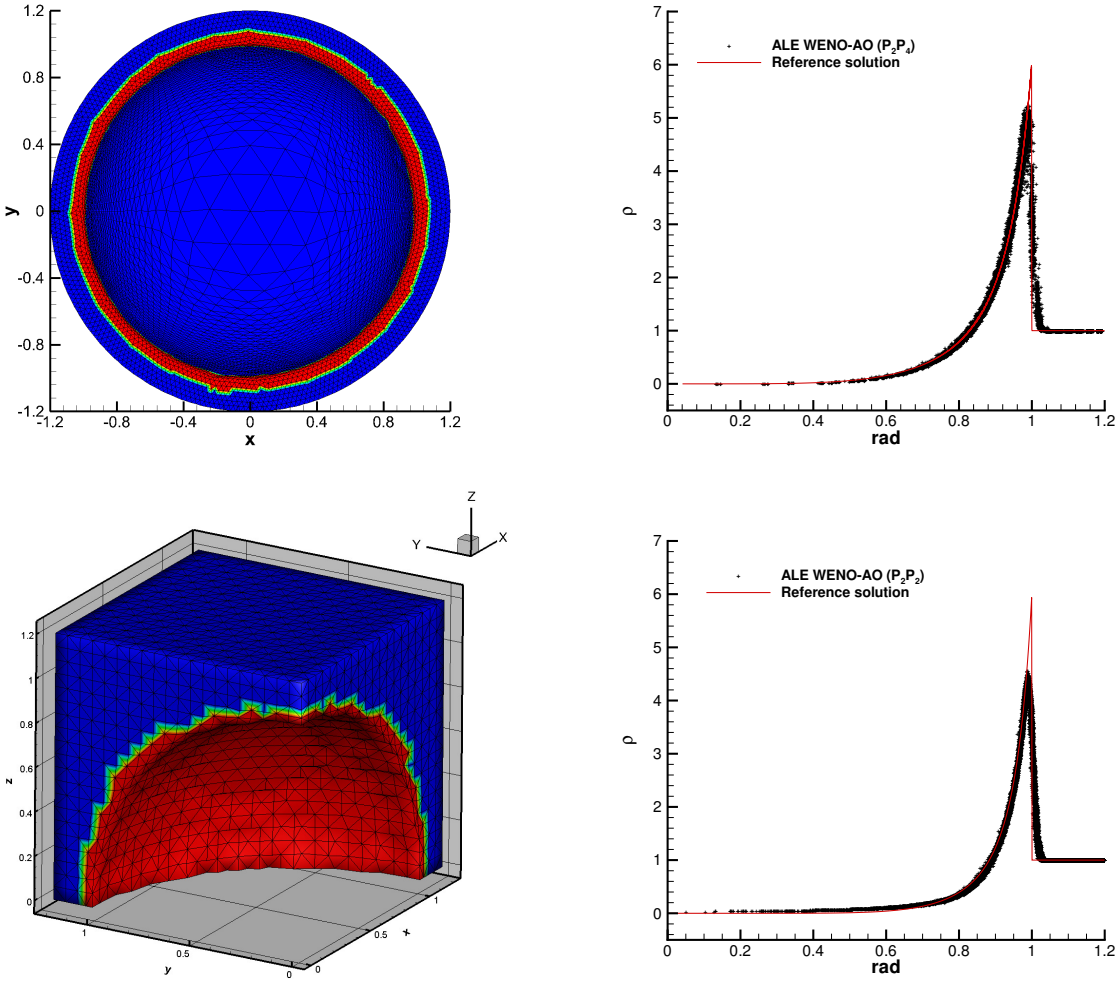

Figure 8: Sedov problem in 2D with P_2P_4 (top row) and in 3D with P_2P_2 (bottom row). Left: mesh configuration at the final time $t_f = 1$ with map of troubled cell indicator. Right: scatter plot of cell density and comparison against the exact solution.

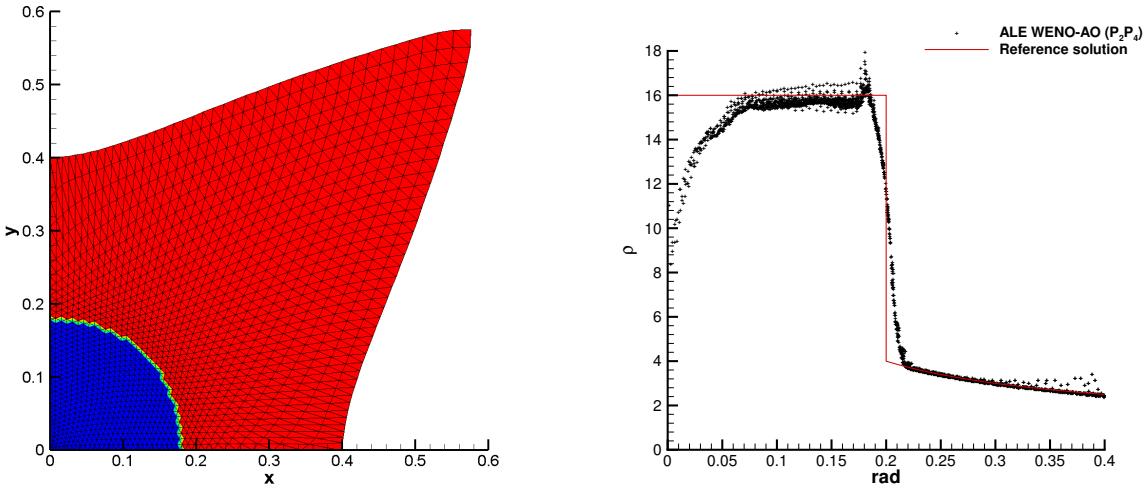

Figure 9: Noh problem in 2D with P_2P_4 (top row) and in 3D with P_2P_3 (bottom row). Left: two-dimensional mesh configuration at the final time $t_f = 0.6$ with map of troubled cell indicator. Right: scatter plot of cell density and comparison against the exact solution.

filled with a gas with $\gamma = 1.4$. The initial flow field reads

$$\mathbf{U}(\mathbf{x},0) = \begin{cases} \rho_0 &= 1 \\ u_0 &= [\sin(\pi x)\cos(\pi y)] \\ v_0 &= [-\cos(\pi x)\sin(\pi y)] \\ p_0 &= \frac{1}{4}[\cos(2\pi x) + \cos(2\pi y)] + 1 \end{cases} . \tag{50}$$

In order to satisfy the total energy equation of the Euler system (42), an additional source term has to be defined for the last equation, hence obtaining

$$\mathbf{S}_E = \frac{\pi}{4(\gamma-1)}\left[\cos(3\pi x)\cos(\pi y) - \cos(3\pi y)\cos(\pi x)\right]. \tag{51}$$

We impose Dirichlet boundary conditions everywhere and we study the numerical convergence of the P_2P_3 and P_2P_4 schemes on a sequence of refined meshes with characteristic mesh size $h = 1/10$, $h = 1/20$, $h = 1/30$. Tables 6, 7, 8 collect the convergence studies at output times $t = 0.2$, $t = 0.4$ and $t = 0.6$, showing that the formal order of accuracy is retrieved for density, horizontal velocity and pressure in L_2 norm. Figure 10 displays the mesh configuration as well as the pressure distribution on each mesh at the final time $t_f = 0.75$, highlighting the strong deformation induced by the vortical motion.

3.7. Double Mach reflection problem

This challenging test case involves a very strong shock wave moving along the $x-$direction of the computational domain, where a ramp with angle $\alpha = \frac{\pi}{6}$ is located. The shock Mach number is $M_s = 10$, thus small-scale structures arise from the interaction between the shock with the ramp. The computational domain $\Omega(0)$ is discretized with a characteristic mesh size of $h = 1/200$, yielding a total number of $N_E = 672680$ triangles. According to [100], the initial condition is given in terms of primitive variables and reads

$$\mathbf{U}(\mathbf{x},0) = \begin{cases} (8.0, 8.25, 0, 116.5), & \text{if} \quad x < x_0, \\ (1.4, 0, 0, 1.0), & \text{if} \quad x \geq x_0, \end{cases} \tag{52}$$

with the initial discontinuity located at $x_0 = 0$. Transmissive boundaries are set on the left and right side of the domain, whereas sliding wall boundary conditions have been imposed on the upper and the lower edges. The ratio of specific

P_2P_3

$h(\Omega(t_f))$	ρ_{L_2}	$O(\rho)$	u_{L_2}	$O(u)$	p_{L_2}	$O(p)$
8.02E-02	5.03E-05	-	4.55E-05	-	6.20E-05	-
3.91E-02	5.63E-06	3.1	4.92E-06	3.1	7.36E-06	3.0
2.77E-02	1.33E-06	4.2	1.32E-06	3.8	1.72E-06	4.2

P_2P_4

$h(\Omega(t_f))$	ρ_{L_2}	$O(\rho)$	u_{L_2}	$O(u)$	p_{L_2}	$O(p)$
8.02E-02	4.55E-05	-	3.46E-05	-	5.68E-05	-
3.91E-02	5.46E-06	3.3	4.51E-06	2.8	7.14E-06	2.9
2.77E-02	8.62E-07	5.4	8.40E-07	4.9	1.10E-06	5.4

Table 6: 2D Taylor-Green vortex. Convergence rates for density ρ, horizontal velocity u and pressure p at time $t = 0.2$.

P_2P_3

$h(\Omega(t_f))$	ρ_{L_2}	$O(\rho)$	u_{L_2}	$O(u)$	p_{L_2}	$O(p)$
8.43E-02	8.85E-04	-	1.12E-04	-	1.17E-04	-
4.04E-02	9.01E-06	3.1	1.24E-05	3.0	1.37E-05	2.9
2.97E-02	2.39E-06	4.3	3.58E-06	4.0	3.81E-06	4.1

P_2P_4

$h(\Omega(t_f))$	ρ_{L_2}	$O(\rho)$	u_{L_2}	$O(u)$	p_{L_2}	$O(p)$
8.43E-02	5.24E-05	-	7.87E-05	-	1.07E-04	-
4.04E-02	3.92E-06	3.5	5.73E-06	3.6	7.38E-06	3.6
2.97E-02	8.34E-07	5.0	1.21E-06	5.1	1.44E-06	5.3

Table 7: 2D Taylor-Green vortex. Convergence rates for density ρ, horizontal velocity u and pressure p at time $t = 0.4$.

P_2P_3

$h(\Omega(t_f))$	ρ_{L_2}	$O(\rho)$	u_{L_2}	$O(u)$	p_{L_2}	$O(p)$
9.13E-02	2.46E-04	-	3.31E-04	-	3.31E-04	-
4.27E-02	4.55E-05	2.2	5.65E-05	2.3	5.52E-05	2.4
3.25E-02	1.36E-05	4.4	1.91E-05	4.0	1.87E-05	4.0

P_2P_4

$h(\Omega(t_f))$	ρ_{L_2}	$O(\rho)$	u_{L_2}	$O(u)$	p_{L_2}	$O(p)$
9.13E-02	2.40E-04	-	3.53E-04	-	4.50E-04	-
4.27E-02	2.42E-05	3.0	3.70E-05	3.0	5.11E-05	2.9
3.25E-02	4.47E-06	6.2	6.55E-06	6.3	8.41E-06	6.6

Table 8: 2D Taylor-Green vortex. Convergence rates for density ρ, horizontal velocity u and pressure p at time $t = 0.6$.

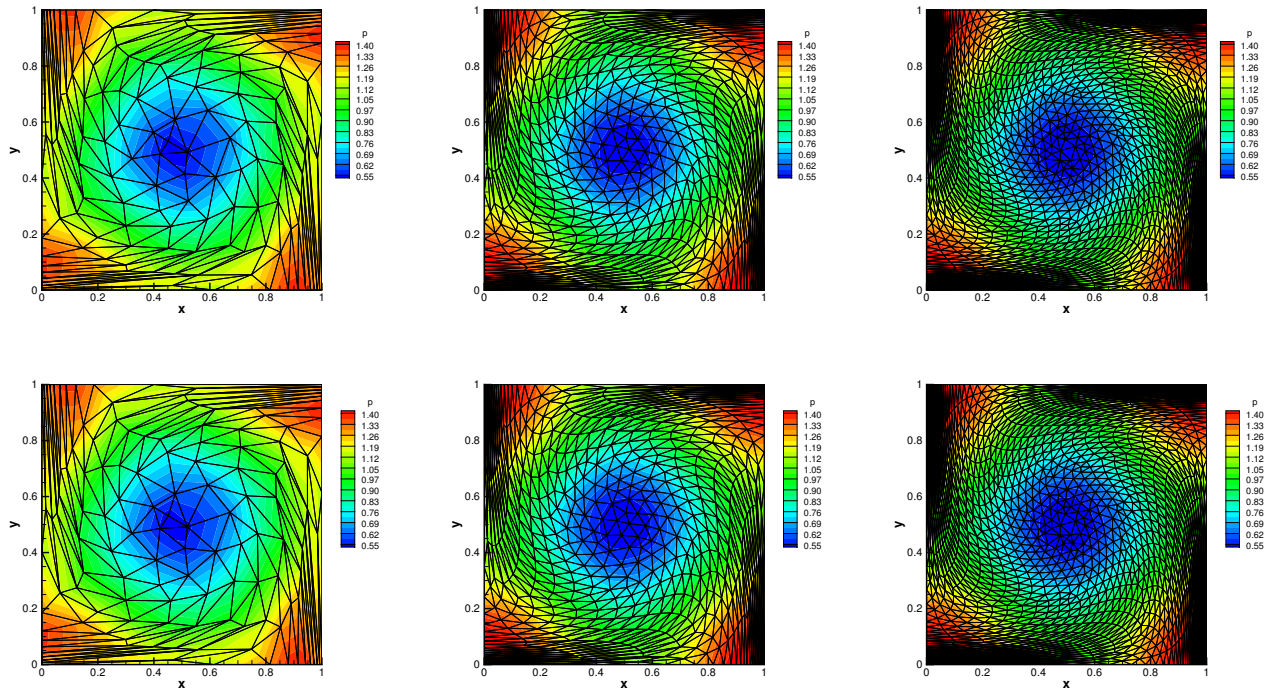

Figure 10: Taylor-Green vortex in 2D with P_2P_3 (top row) and P_2P_4 (bottom row). Mesh configuration and pressure distribution at time $t = 0.75$ for characteristic mesh size $h = 1/10$ (left column), $h = 1/20$ (middle column) and $h = 1/30$ (right column).

heats is $\gamma = 1.4$ and the simulation stops at time $t_f = 0.2$. The computation is run using a P_2P_4 scheme on a fixed mesh and the results are depicted in Figure 11: the troubled cell indicator correctly identifies only the cells crossed by shock waves and substructures are captured by the high order scheme.

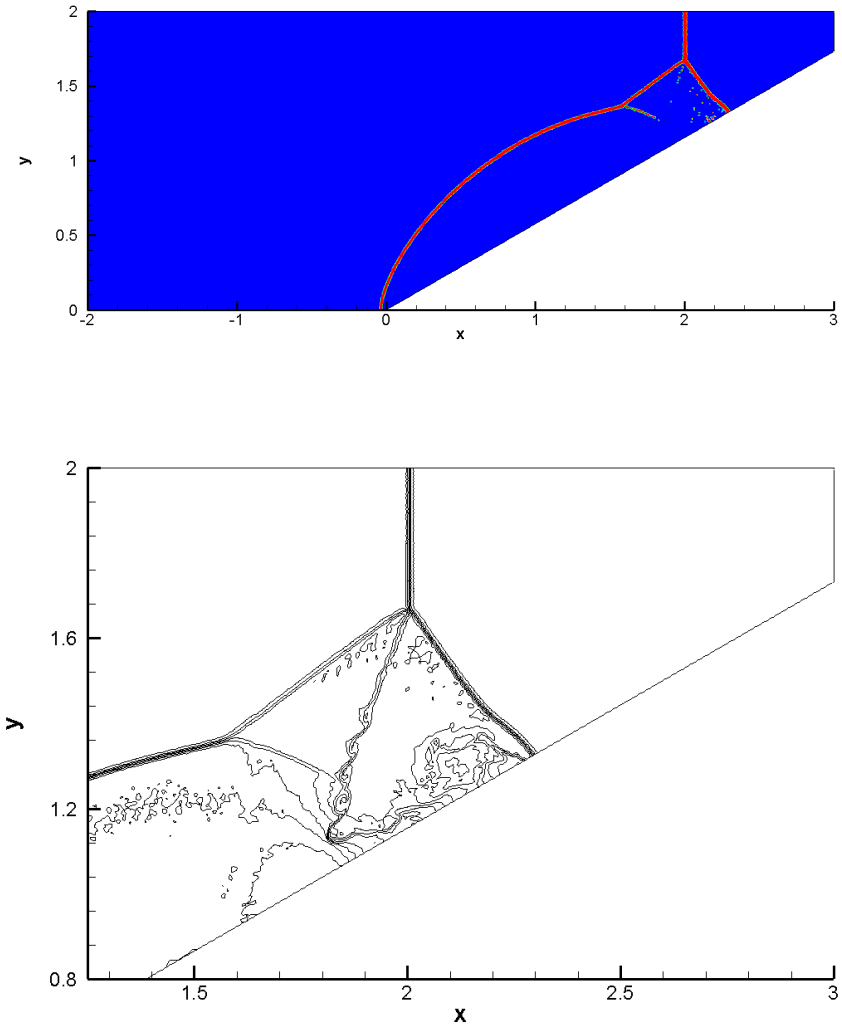

Figure 11: Double Mach reflection problem with P_2P_4 at final time $t_f = 0.2$. Top: map of troubled cell indicator. Bottom: zoom on the roll-up of the Mach stem with 20 equally spaced density contour lines from 1.5 to 22.

3.8. Forward facing step problem

We consider a two-dimensional tunnel of dimension $\Omega(0) = [0; 3] \times [0; 1]$, in which a step of length $L = 0.2$ is located at position $x_0 = 0.6$, see [100] for details. The characteristic mesh size is $h = 1/200$ and we impose transmissive boundary conditions at the left and right side of the domain, while reflective wall boundaries are set elsewhere. The initial condition is given by a perfect gas with $\gamma = 1.4$ with Mach number $M_s = 3$ and the simulation is run until the final time $t_f = 2.5$ with a P_2P_4 scheme with Rusanov-type flux (10). Even in this case the mesh velocity is set to zero, hence recovering a standard Eulerian algorithm. As shown in Figure 12, a bow shock wave is generated

and it spreads towards the upper wall boundary of the computational domain. As a result, a normal shock and a slip surface are emanated by the reflecting wall and the interaction among all waves yields a triple point structure in the flow. Physical instabilities and roll-up of the contact line are well recovered by our fifth order scheme and the WENO limiter is active only where needed, as depicted in the troubled cell indicator map of Figure 12.

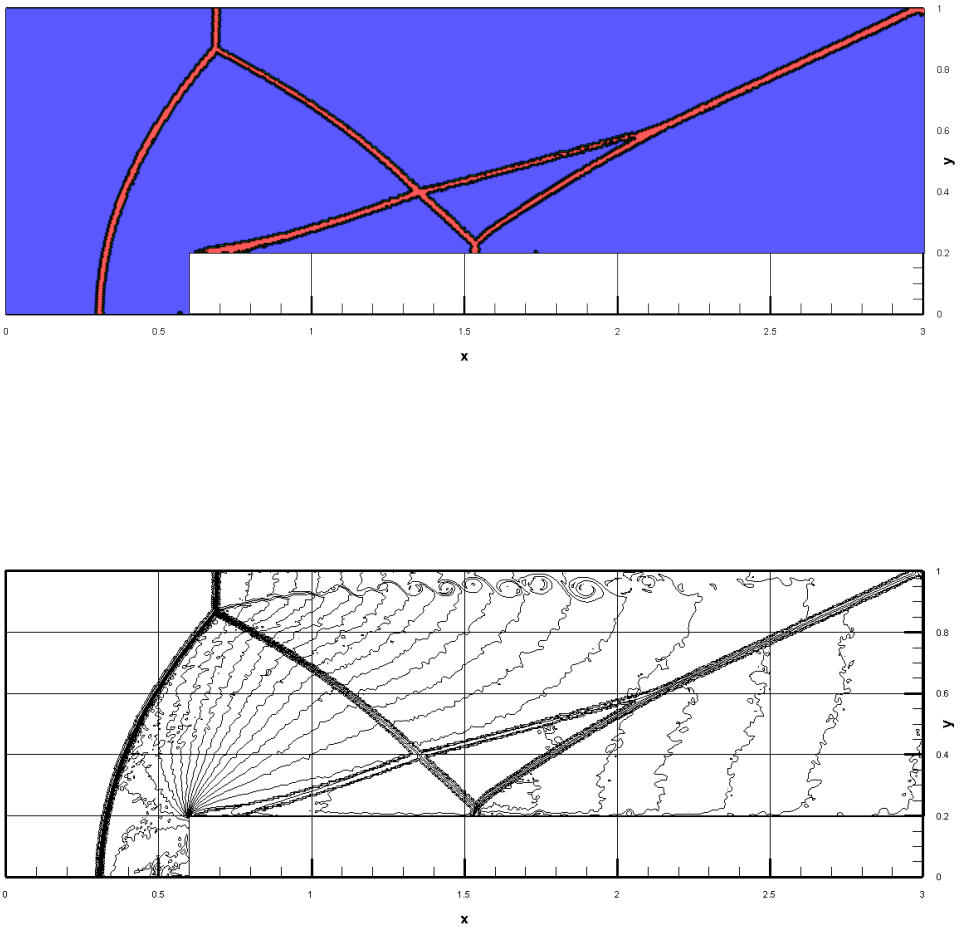

Figure 12: Forward facing step problem with P_2P_4 at final time $t_f = 2.5$. Top: map of troubled cell indicator. Bottom: 20 equally spaced density contour lines from 0.5 to 6.5.

4. Conclusions

In this work we have presented a new family of P_NP_M in the framework of direct Arbitrary-Lagrangian-Eulerian (ALE) methods on moving unstructured multidimensional meshes. A conservative nonlinear WENO Adaptive Order (AO) reconstruction operator is applied to a one-step explicit DG scheme, based on a high order space-time predictor computed relying on the ADER methodology. The WENO-AO strategy is performed by means of the shifting polynomial technique, which becomes easier in our approach because the entire reconstruction procedure is carried out

on a reference system. A WENO limiter is used for stabilizing the DG solution and a novel indicator for identifying troubled cells is proposed for the first time in the context of ALE DG algorithms.

Convergence studies demonstrate the space-time accuracy of the new schemes up to fifth order of accuracy and a wide range of benchmark test cases have been run, in order to assess the validity and the robustness of the method for both Eulerian and quasi-Lagrangian simulations.

Future work will regard the extension of the presented scheme to non-conservative systems and stiff source terms, thus considering multi-phase flows with Baer-Nunziato model [2] as well as nonlinear hyperelasticity in the framework of [87]. The ideal and resistive equations of magnetohydrodynamics (MHD) are also planned to be solved using the proposed approach supplemented by a constraint preserving scheme for ensuring a divergence free magnetic field at the discrete level.

Acknowledgments

The Authors acknowledge the HLRS supercomputing center in Stuttgart (Germany) for awarding access to the *HazelHen* supercomputer. WB has been funded by INdAM (Italian National Institute of High Mathematics) under the program *Young researchers funding 2018*. DSB acknowledges support via NSF grants NSF-ACI-1533850, NSF-DMS-1622457, NSF-ACI-1713765 and NSF-DMS-1821242.

The Authors would also like to thank the two anonymous referees for their constructive comments and for suggesting additional test cases. This was highly appreciated, since it helped to improve the quality and the clarity of the manuscript.

Appendix A. Basis functions

The *modal* basis functions ϕ_l and ψ_l used in (3)-(5) as well as the *nodal* basis functions θ_l adopted in (6) are explicitly written up to degree 3 for both triangles and tetrahedra.

Modal basis functions in space.
The orthogonal Dubiner-type basis functions [41, 62, 30] are used for ϕ_l and ψ_l, which constitute a hierarchical tensor-type basis. They are defined on the reference element T_E shown in Figure 2, therefore they depend on the reference coordinate system defined by the vector $\boldsymbol{\xi} = (\xi, \eta, \zeta)$, i.e. $\phi_l = \phi_l(\boldsymbol{\xi})$ and $\psi_l = \psi_l(\boldsymbol{\xi})$. The basis functions ϕ_l are listed hereafter for degree $N = 0$, $N = 1$, $N = 2$, $N = 3$.

The modal basis functions ϕ_l of degree N on *triangles* (2D) write as follows:

- $N = 0$

$$\phi_1 = 1$$

- $N = 1$

$$\begin{aligned} \phi_1 &= 1 \\ \phi_2 &= 2\xi - 1 + \eta \\ \phi_3 &= -1 + 3\eta \end{aligned}$$

- $N = 2$

$$\begin{aligned} \phi_1 &= 1 \\ \phi_2 &= 2\xi - 1 + \eta \\ \phi_3 &= -1 + 3\eta \\ \phi_4 &= 1 - 2 * \eta + \eta^2 - 6 * \xi + 6 * \xi * \eta + 6 * \xi^2 \\ \phi_5 &= 5\eta^2 + 10\xi\eta - 6\eta - 2\xi + 1 \\ \phi_6 &= 1 - 8\eta + 10\eta^2 \end{aligned}$$

- $N = 3$

$$
\begin{aligned}
\phi_1 &= 1 \\
\phi_2 &= 2\xi - 1 + \eta \\
\phi_3 &= -1 + 3\eta \\
\phi_4 &= 1 - 2 * \eta + \eta^2 - 6 * \xi + 6 * \xi * \eta + 6 * \xi^2 \\
\phi_5 &= 5\eta^2 + 10\xi\eta - 6\eta - 2\xi + 1 \\
\phi_6 &= 1 - 8\eta + 10\eta^2 \\
\phi_7 &= -1 + 3\eta + 12\xi - 3\eta^2 - 24\xi\eta - 30\xi^2 + \eta^3 + 12\xi\eta^2 + 30\xi^2\eta + 20\xi^3 \\
\phi_8 &= 7\eta^3 - 15\eta^2 + 42\xi\eta^2 - 48\xi\eta + 42\xi^2\eta + 9\eta + 6\xi - 6\xi^2 - 1 \\
\phi_9 &= 21\eta^3 + 42\xi\eta^2 - 33\eta^2 - 24\xi\eta + 13\eta - 1 + 2\xi \\
\phi_{10} &= -1 + 15\eta - 45\eta^2 + 35\eta^3
\end{aligned}
$$

The modal basis functions ϕ_l of degree N on *tetrahedra* (3D) write as follows:

- $N = 0$

$$
\phi_1 = 1
$$

- $N = 1$

$$
\begin{aligned}
\phi_1 &= 1 \\
\phi_2 &= 2\xi - 1 + \eta + \zeta \\
\phi_3 &= -1 + 3\eta + \zeta \\
\phi_4 &= -1 + 4\zeta
\end{aligned}
$$

- $N = 2$

$$
\begin{aligned}
\phi_1 &= 1 \\
\phi_2 &= 2\xi - 1 + \eta + \zeta \\
\phi_3 &= -1 + 3\eta + \zeta \\
\phi_4 &= -1 + 4\zeta \\
\phi_5 &= \zeta^2 - 2\zeta + 2\eta\zeta + 1 - 2\eta + \eta^2 - 6\xi + 6\xi\eta + 6\zeta\xi + 6\xi^2 \\
\phi_6 &= 5\eta^2 + 10\xi\eta + 6\eta\zeta - 6\eta + \zeta^2 + 2\zeta\xi - 2\zeta - 2\xi + 1 \\
\phi_7 &= 1 - 2\zeta + \zeta^2 - 8\eta + 8\eta\zeta + 10\eta^2 \\
\phi_8 &= -\eta + 6\eta\zeta - 2\xi + 12\zeta\xi + 6\zeta^2 - 7\zeta + 1 \\
\phi_9 &= 6\zeta^2 - 7\zeta + 18\eta\zeta + 1 - 3\eta \\
\phi_{10} &= 1 - 10\zeta + 15\zeta^2
\end{aligned}
$$

- $N = 3$

$$
\begin{aligned}
\phi_1 &= 1 \\
\phi_2 &= 2\xi - 1 + \eta + \zeta \\
\phi_3 &= -1 + 3\eta + \zeta \\
\phi_4 &= -1 + 4\zeta \\
\phi_5 &= \zeta^2 - 2\zeta + 2\eta\zeta + 1 - 2\eta + \eta^2 - 6\xi + 6\xi\eta + 6\zeta\xi + 6\xi^2 \\
\phi_6 &= 5\eta^2 + 10\xi\eta + 6\eta\zeta - 6\eta + \zeta^2 + 2\zeta\xi - 2\zeta - 2\xi + 1 \\
\phi_7 &= 1 - 2\zeta + \zeta^2 - 8\eta + 8\eta\zeta + 10\eta^2 \\
\phi_8 &= -\eta + 6\eta\zeta - 2\xi + 12\zeta\xi + 6\zeta^2 - 7\zeta + 1 \\
\phi_9 &= 6\zeta^2 - 7\zeta + 18\eta\zeta + 1 - 3\eta \\
\phi_{10} &= 1 - 10\zeta + 15\zeta^2 \\
\phi_{11} &= -1 + 3\eta + 12\xi - 3\eta^2 - 24\xi\eta - 30\xi^2 + \eta^3 + 12\xi\eta^2 + 30\xi^2\eta + 20\xi^3 - 6\eta\zeta + 3\zeta\eta^2 \\
&+ 3\zeta^2\eta + 3\zeta - 3\zeta^2 + 12\zeta^2\xi - 24\zeta\xi + 24\zeta\eta\xi + \zeta^3 + 30\zeta\xi^2 \\
\phi_{12} &= 7\eta^3 - 15\eta^2 + 15\zeta\eta^2 + 42\xi\eta^2 + 48\zeta\eta\xi + 42\xi^2\eta + 9\eta + 9\zeta^2\eta - 48\xi\eta - 18\eta\zeta - 3\zeta^2 \\
&+ 6\zeta\xi^2 - 6\xi^2 + \zeta^3 + 6\zeta^2\xi + 6\xi - 1 - 12\zeta\xi + 3\zeta \\
\phi_{13} &= 21\eta^3 + 33\zeta\eta^2 - 33\eta^2 + 42\xi\eta^2 + 13\zeta^2\eta - 26\eta\zeta + 13\eta + 24\zeta\eta\xi - 24\xi\eta + 3\zeta - 4\zeta\xi + 2\zeta^2\xi - 1 - 3\zeta^2 + 2\xi + \zeta^3 \\
\phi_{14} &= -1 + 15\eta - 45\eta^2 + 35\eta^3 - 30\eta\zeta + 45\zeta\eta^2 + 15\zeta^2\eta + 3\zeta - 3\zeta^2 + \zeta^3 \\
\phi_{15} &= -\eta^2 + 8\zeta\eta^2 + 2\eta - 6\xi\eta + 16\zeta^2\eta - 18\eta\zeta + 48\zeta\eta\xi + 8\zeta^3 - 54\zeta\xi - 6\xi^2 + 6\xi - 1 + 10\zeta + 48\zeta^2\xi + 48\zeta\xi^2 - 17\zeta^2 \\
\phi_{16} &= 40\zeta\eta^2 - 5\eta^2 + 80\zeta\eta\xi - 54\eta\zeta + 48\zeta^2\eta - 10\xi\eta + 6\eta + 16\zeta^2\xi + 10\zeta - 1 + 8\zeta^3 + 2\xi - 17\zeta^2 - 18\zeta\xi \\
\phi_{17} &= 8\zeta^3 - 17\zeta^2 + 64\zeta^2\eta + 10\zeta - 72\eta\zeta + 80\zeta\eta^2 - 1 - 10\eta^2 + 8\eta \\
\phi_{18} &= \eta - 14\eta\zeta + 28\zeta^2\eta + 56\zeta^2\xi + 28\zeta^3 + 15\zeta - 1 - 28\zeta\xi + 2\xi - 42\zeta^2 \\
\phi_{19} &= 28\zeta^3 - 42\zeta^2 + 84\zeta^2\eta + 15\zeta - 42\eta\zeta - 1 + 3\eta \\
\phi_{20} &= -1 + 18\zeta - 63\zeta^2 + 56\zeta^3
\end{aligned}
$$

Nodal basis functions in space and time.
The space-time basis functions $\theta_l(\tilde{\boldsymbol{\xi}})$ are given by the Lagrange interpolation polynomials passing through a set of space-time nodes $\tilde{\boldsymbol{\xi}} = (\xi, \eta, \zeta, \tau)$. These nodes are explicitly defined in [44] on a reference space-time element $\tilde{T}_E$, which simply results in $T_E \times [0, 1]$ with the reference time coordinate $\tau \in [0, 1]$. The space-time nodes $\tilde{\boldsymbol{\xi}}_l$ and the basis functions θ_l are listed hereafter for degree $N = 0$, $N = 1$, $N = 2$, $N = 3$.

The nodal degrees of freedom $\tilde{\boldsymbol{\xi}}_l$ and the associated basis functions θ_l of degree N on space-time *triangles* (2D) write as follows:

- $N = 0$

$$
\begin{aligned}
\tilde{\boldsymbol{\xi}}_1 &= (1/3, 1/3, 1) \\
\theta_1 &= 1
\end{aligned}
$$

- $N = 1$

$$
\begin{aligned}
\tilde{\boldsymbol{\xi}}_1 &= (0, 0, 0) \\
\tilde{\boldsymbol{\xi}}_2 &= (1, 0, 0) \\
\tilde{\boldsymbol{\xi}}_3 &= (0, 1, 0) \\
\tilde{\boldsymbol{\xi}}_4 &= (0, 0, 1)
\end{aligned}
$$

$$
\begin{aligned}
\theta_1 &= 1-\tau-\xi-\eta \\
\theta_2 &= \xi \\
\theta_3 &= \eta \\
\theta_4 &= \tau
\end{aligned}
$$

- $N = 2$

$$
\begin{aligned}
\tilde{\boldsymbol{\xi}}_1 &= (0,0,0) \\
\tilde{\boldsymbol{\xi}}_2 &= (1/2,0,0) \\
\tilde{\boldsymbol{\xi}}_3 &= (1,0,0) \\
\tilde{\boldsymbol{\xi}}_4 &= (0,1/2,0) \\
\tilde{\boldsymbol{\xi}}_5 &= (1/2,1/2,0) \\
\tilde{\boldsymbol{\xi}}_6 &= (0,1,0) \\
\tilde{\boldsymbol{\xi}}_7 &= (0,0,1/2) \\
\tilde{\boldsymbol{\xi}}_8 &= (1,0,1/2) \\
\tilde{\boldsymbol{\xi}}_9 &= (0,1,1/2) \\
\tilde{\boldsymbol{\xi}}_{10} &= (0,0,1)
\end{aligned}
$$

$$
\begin{aligned}
\theta_1 &= 2\eta\tau - 3\tau - 3\xi - 3\eta + 4\eta\xi + 2\tau\xi + 2\eta^2 + 2\tau^2 + 2\xi^2 + 1 \\
\theta_2 &= 4\xi - 4\eta\xi - 4\xi^2 \\
\theta_3 &= 2\xi^2 - 2\tau\xi - \xi \\
\theta_4 &= 4\eta - 4\eta\xi - 4\eta^2 \\
\theta_5 &= 4\eta\xi \\
\theta_6 &= 2\eta^2 - 2\eta\tau - \eta \\
\theta_7 &= 4\tau - 2\eta\tau - 2\tau\xi - 4\tau^2 \\
\theta_8 &= 2\tau\xi \\
\theta_9 &= 2\eta\tau \\
\theta_{10} &= 2\tau^2 - \tau
\end{aligned}
$$

- $N = 3$

$$
\begin{aligned}
\tilde{\xi}_1 &= (0,0,0) \\
\tilde{\xi}_2 &= (1/3,0,0) \\
\tilde{\xi}_3 &= (2/3,0,0) \\
\tilde{\xi}_4 &= (1,0,0) \\
\tilde{\xi}_5 &= (0,1/3,0) \\
\tilde{\xi}_6 &= (1/3,1/3,0) \\
\tilde{\xi}_7 &= (2/3,1/3,0) \\
\tilde{\xi}_8 &= (0,2/3,0) \\
\tilde{\xi}_9 &= (1/3,2/3,0) \\
\tilde{\xi}_{10} &= (0,1,0) \\
\tilde{\xi}_{11} &= (0,0,1/3) \\
\tilde{\xi}_{12} &= (1/2,0,1/3) \\
\tilde{\xi}_{13} &= (1,0,1/3) \\
\tilde{\xi}_{14} &= (0,1/2,1/3) \\
\tilde{\xi}_{15} &= (1/2,1/2,1/3) \\
\tilde{\xi}_{16} &= (0,1,1/3) \\
\tilde{\xi}_{17} &= (0,0,2/3) \\
\tilde{\xi}_{18} &= (1,0,2/3) \\
\tilde{\xi}_{19} &= (0,1,2/3) \\
\tilde{\xi}_{20} &= (0,0,1)
\end{aligned}
$$

$$
\begin{aligned}
\theta_1 &= (45\eta\tau)/4 - (11\tau)/2 - (11\xi)/2 - (11\eta)/2 + 18\eta\xi + (45\tau\xi)/4 - (9\eta\tau^2)/2 - (27\eta^2\tau)/4 - (27\eta\xi^2)/2 - (27\eta^2\xi)/2 \\
&- (27\tau\xi^2)/4 - (9\tau^2\xi)/2 + 9\eta^2 - (9\eta^3)/2 + 9\tau^2 - (9\tau^3)/2 + 9\xi^2 - (9\xi^3)/2 - (27\eta\tau\xi)/2 + 1 \\
\theta_2 &= 9\xi - (45\eta\xi)/2 - (27\tau\xi)/4 + 27\eta\xi^2 + (27\eta^2\xi)/2 + (27\tau\xi^2)/4 - (45\xi^2)/2 + (27\xi^3)/2 + (27\eta\tau\xi)/4 \\
\theta_3 &= (9\eta\xi)/2 - (9\xi)/2 - (27\tau\xi)/4 - (27\eta\xi^2)/2 + (27\tau\xi^2)/4 + 18\xi^2 - (27\xi^3)/2 + (27\eta\tau\xi)/4 \\
\theta_4 &= \xi + (9\tau\xi)/4 - (27\tau\xi^2)/4 + (9\tau^2\xi)/2 - (9\xi^2)/2 + (9\xi^3)/2 \\
\theta_5 &= 9\eta - (27\eta\tau)/4 - (45\eta\xi)/2 + (27\eta^2\tau)/4 + (27\eta\xi^2)/2 + 27\eta^2\xi - (45\eta^2)/2 + (27\eta^3)/2 + (27\eta\tau\xi)/4 \\
\theta_6 &= 27\eta\xi - 27\eta\xi^2 - 27\eta^2\xi \\
\theta_7 &= (27\eta\xi^2)/2 - (9\eta\xi)/2 - (27\eta\tau\xi)/4 \\
\theta_8 &= (9\eta\xi)/2 - (27\eta\tau)/4 - (9\eta)/2 + (27\eta^2\tau)/4 - (27\eta^2\xi)/2 + 18\eta^2 - (27\eta^3)/2 + (27\eta\tau\xi)/4 \\
\theta_9 &= (27\eta^2\xi)/2 - (9\eta\xi)/2 - (27\eta\tau\xi)/4 \\
\theta_{10} &= \eta + (9\eta\tau)/4 + (9\eta\tau^2)/2 - (27\eta^2\tau)/4 - (9\eta^2)/2 + (9\eta^3)/2 \\
\theta_{11} &= 9\tau - 12\eta\tau - 12\tau\xi + 9\eta\tau^2 + 6\eta^2\tau + 6\tau\xi^2 + 9\tau^2\xi - (45\tau^2)/2 + (27\tau^3)/2 + 12\eta\tau\xi \\
\theta_{12} &= 12\tau\xi - 12\tau\xi^2 - 12\eta\tau\xi \\
\theta_{13} &= 6\tau\xi^2 - 9\tau^2\xi \\
\theta_{14} &= 12\eta\tau - 12\eta^2\tau - 12\eta\tau\xi \\
\theta_{15} &= 12\eta\tau\xi \\
\theta_{16} &= 6\eta^2\tau - 9\eta\tau^2 \\
\theta_{17} &= (3\eta\tau)/2 - (9\tau)/2 + (3\tau\xi)/2 - (9\eta\tau^2)/2 - (9\tau^2\xi)/2 + 18\tau^2 - (27\tau^3)/2 \\
\theta_{18} &= (9\tau^2\xi)/2 - (3\tau\xi)/2 \\
\theta_{19} &= (9\eta\tau^2)/2 - (3\eta\tau)/2 \\
\theta_{20} &= \tau - (9\tau^2)/2 + (9\tau^3)/2
\end{aligned}
$$

The nodal degrees of freedom $\tilde{\boldsymbol{\xi}}_l$ and the associated basis functions θ_l of degree N on space-time *tetrahedra* (3D) write as follows:

- $N = 0$

$$
\begin{aligned}
\tilde{\boldsymbol{\xi}}_1 &= (1/4, 1/4, 1/4, 1) \\
\theta_1 &= 1
\end{aligned}
$$

- $N = 1$

$$
\begin{aligned}
\tilde{\boldsymbol{\xi}}_1 &= (0,0,0,0) \\
\tilde{\boldsymbol{\xi}}_2 &= (1,0,0,0) \\
\tilde{\boldsymbol{\xi}}_3 &= (0,1,0,0) \\
\tilde{\boldsymbol{\xi}}_4 &= (0,0,1,0) \\
\tilde{\boldsymbol{\xi}}_5 &= (0,0,0,1)
\end{aligned}
$$

$$
\begin{aligned}
\theta_1 &= 1 - \tau - \xi - \zeta - \eta \\
\theta_2 &= \xi \\
\theta_3 &= \eta \\
\theta_4 &= \zeta \\
\theta_5 &= \tau
\end{aligned}
$$

- $N = 2$

$$
\begin{aligned}
\tilde{\xi}_1 &= (0,0,0,0)\\
\tilde{\xi}_2 &= (1/2,0,0,0)\\
\tilde{\xi}_3 &= (1,0,0,0)\\
\tilde{\xi}_4 &= (0,1/2,0,0)\\
\tilde{\xi}_5 &= (1/2,1/2,0,0)\\
\tilde{\xi}_6 &= (0,1,0,0)\\
\tilde{\xi}_7 &= (0,0,1/2,0)\\
\tilde{\xi}_8 &= (1/2,0,1/2,0)\\
\tilde{\xi}_9 &= (0,1/2,1/2,0)\\
\tilde{\xi}_{10} &= (0,0,1,0)\\
\tilde{\xi}_{11} &= (0,0,0,1/2)\\
\tilde{\xi}_{12} &= (1,0,0,1/2)\\
\tilde{\xi}_{13} &= (0,1,0,1/2)\\
\tilde{\xi}_{14} &= (0,0,1,1/2)\\
\tilde{\xi}_{15} &= (0,0,0,1)
\end{aligned}
$$

$$
\begin{aligned}
\theta_1 &= 2\eta\tau - 3\tau - 3\xi - 3\zeta - 3\eta + 4\eta\xi + 4\eta\zeta + 2\tau\xi + 2\tau\zeta + 4\xi\zeta + 2\eta^2 + 2\tau^2 + 2\xi^2 + 2\zeta^2 + 1\\
\theta_2 &= 4\xi - 4\eta\xi - 4\xi\zeta - 4\xi^2\\
\theta_3 &= 2\xi^2 - 2\tau\xi - \xi\\
\theta_4 &= 4\eta - 4\eta\xi - 4\eta\zeta - 4\eta^2\\
\theta_5 &= 4\eta\xi\\
\theta_6 &= 2\eta^2 - 2\eta\tau - \eta\\
\theta_7 &= 4\zeta - 4\eta\zeta - 4\xi\zeta - 4\zeta^2\\
\theta_8 &= 4\xi\zeta\\
\theta_9 &= 4\eta\zeta\\
\theta_{10} &= 2\zeta^2 - 2\tau\zeta - \zeta\\
\theta_{11} &= 4\tau - 2\eta\tau - 2\tau\xi - 2\tau\zeta - 4\tau^2\\
\theta_{12} &= 2\tau\xi\\
\theta_{13} &= 2\eta\tau\\
\theta_{14} &= 2\tau\zeta\\
\theta_{15} &= 2\tau^2 - \tau
\end{aligned}
$$

- $N = 3$

$$
\begin{aligned}
\tilde{\xi}_1 &= (0,0,0,0)\\
\tilde{\xi}_2 &= (1/3,0,0,0)\\
\tilde{\xi}_3 &= (2/3,0,0,0)\\
\tilde{\xi}_4 &= (1,0,0,0)\\
\tilde{\xi}_5 &= (0,1/3,0,0)\\
\tilde{\xi}_6 &= (1/3,1/3,0,0)\\
\tilde{\xi}_7 &= (2/3,1/3,0,0)\\
\tilde{\xi}_8 &= (0,2/3,0,0)\\
\tilde{\xi}_9 &= (1/3,2/3,0,0)\\
\tilde{\xi}_{10} &= (0,1,0,0)\\
\tilde{\xi}_{11} &= (0,0,1/3,0)\\
\tilde{\xi}_{12} &= (1/3,0,1/3,0)\\
\tilde{\xi}_{13} &= (2/3,0,1/3,0)\\
\tilde{\xi}_{14} &= (0,1/3,1/3,0)\\
\tilde{\xi}_{15} &= (1/3,1/3,1/3,0)\\
\tilde{\xi}_{16} &= (0,2/3,1/3,0)\\
\tilde{\xi}_{17} &= (0,0,2/3,0)\\
\tilde{\xi}_{18} &= (1/3,0,2/3,0)\\
\tilde{\xi}_{19} &= (0,1/3,2/3,0)\\
\tilde{\xi}_{20} &= (0,0,1,0)\\
\tilde{\xi}_{21} &= (0,0,0,1/3)\\
\tilde{\xi}_{22} &= (1/2,0,0,1/3)\\
\tilde{\xi}_{23} &= (1,0,0,1/3)\\
\tilde{\xi}_{24} &= (0,1/2,0,1/3)\\
\tilde{\xi}_{25} &= (1/2,1/2,0,1/3)\\
\tilde{\xi}_{26} &= (0,1,0,1/3)\\
\tilde{\xi}_{27} &= (0,0,1/2,1/3)\\
\tilde{\xi}_{28} &= (1/2,0,1/2,1/3)\\
\tilde{\xi}_{29} &= (0,1/2,1/2,1/3)\\
\tilde{\xi}_{30} &= (0,0,1,1/3)\\
\tilde{\xi}_{31} &= (0,0,0,2/3)\\
\tilde{\xi}_{32} &= (1,0,0,2/3)\\
\tilde{\xi}_{33} &= (0,1,0,2/3)\\
\tilde{\xi}_{34} &= (0,0,1,2/3)\\
\tilde{\xi}_{35} &= (0,0,0,1)
\end{aligned}
$$

$$
\begin{aligned}
\theta_{1} &= (45\eta\tau)/4 - (11\tau)/2 - (11\xi)/2 - (11\zeta)/2 - (11\eta)/2 + 18\eta\xi + 18\eta\zeta + (45\tau\xi)/4 + (45\tau\zeta)/4 + 18\xi\zeta - (9\eta\tau^2)/2 \\
&- (27\eta^2\tau)/4 - (27\eta\xi^2)/2 - (27\eta^2\xi)/2 - (27\eta\zeta^2)/2 - (27\eta^2\zeta)/2 - (27\tau\xi^2)/4 - (9\tau^2\xi)/2 - (27\tau\zeta^2)/4 - (9\tau^2\zeta)/2 \\
&- (27\xi\zeta^2)/2 - (27\xi^2\zeta)/2 + 9\eta^2 - (9\eta^3)/2 + 9\tau^2 - (9\tau^3)/2 + 9\xi^2 - (9\xi^3)/2 + 9\zeta^2 - (9\zeta^3)/2 \\
&- (27\eta\tau\xi)/2 - (27\eta\tau\zeta)/2 - 27\eta\xi\zeta - (27\tau\xi\zeta)/2 + 1 \\
\theta_{2} &= 9\xi - (45\eta\xi)/2 - (27\tau\xi)/4 - (45\xi\zeta)/2 + 27\eta\xi^2 + (27\eta^2\xi)/2 + (27\tau\xi^2)/4 + (27\xi\zeta^2)/2 + 27\xi^2\zeta - (45\xi^2)/2 \\
&+ (27\xi^3)/2 + (27\eta\tau\xi)/4 + 27\eta\xi\zeta + (27\tau\xi\zeta)/4 \\
\theta_{3} &= (9\eta\xi)/2 - (9\xi)/2 - (27\tau\xi)/4 + (9\xi\zeta)/2 - (27\eta\xi^2)/2 + (27\tau\xi^2)/4 - (27\xi^2\zeta)/2 + 18\xi^2 - (27\xi^3)/2 \\
&+ (27\eta\tau\xi)/4 + (27\tau\xi\zeta)/4 \\
\theta_{4} &= \xi + (9\tau\xi)/4 - (27\tau\xi^2)/4 + (9\tau^2\xi)/2 - (9\xi^2)/2 + (9\xi^3)/2 \\
\theta_{5} &= 9\eta - (27\eta\tau)/4 - (45\eta\xi)/2 - (45\eta\zeta)/2 + (27\eta^2\tau)/4 + (27\eta\xi^2)/2 + 27\eta^2\xi + (27\eta\zeta^2)/2 + 27\eta^2\zeta - (45\eta^2)/2 \\
&+ (27\eta^3)/2 + (27\eta\tau\xi)/4 + (27\eta\tau\zeta)/4 + 27\eta\xi\zeta \\
\theta_{6} &= 27\eta\xi - 27\eta\xi^2 - 27\eta^2\xi - 27\eta\xi\zeta \\
\theta_{7} &= (27\eta\xi^2)/2 - (9\eta\xi)/2 - (27\eta\tau\xi)/4 \\
\theta_{8} &= (9\eta\xi)/2 - (27\eta\tau)/4 - (9\eta)/2 + (9\eta\zeta)/2 + (27\eta^2\tau)/4 - (27\eta^2\xi)/2 - (27\eta^2\zeta)/2 + 18\eta^2 \\
&- (27\eta^3)/2 + (27\eta\tau\xi)/4 + (27\eta\tau\zeta)/4 \\
\theta_{9} &= (27\eta^2\xi)/2 - (9\eta\xi)/2 - (27\eta\tau\xi)/4 \\
\theta_{10} &= \eta + (9\eta\tau)/4 + (9\eta\tau^2)/2 - (27\eta^2\tau)/4 - (9\eta^2)/2 + (9\eta^3)/2 \\
\theta_{11} &= 9\zeta - (45\eta\zeta)/2 - (27\tau\zeta)/4 - (45\xi\zeta)/2 + 27\eta\zeta^2 + (27\eta^2\zeta)/2 + (27\tau\zeta^2)/4 + 27\xi\zeta^2 + (27\xi^2\zeta)/2 \\
&- (45\zeta^2)/2 + (27\zeta^3)/2 + (27\eta\tau\zeta)/4 + 27\eta\xi\zeta + (27\tau\xi\zeta)/4 \\
\theta_{12} &= 27\xi\zeta - 27\xi\zeta^2 - 27\xi^2\zeta - 27\eta\xi\zeta \\
\theta_{13} &= (27\xi^2\zeta)/2 - (9\xi\zeta)/2 - (27\tau\xi\zeta)/4 \\
\theta_{14} &= 27\eta\zeta - 27\eta\zeta^2 - 27\eta^2\zeta - 27\eta\xi\zeta \\
\theta_{15} &= 27\eta\xi\zeta \\
\theta_{16} &= (27\eta^2\zeta)/2 - (9\eta\zeta)/2 - (27\eta\tau\zeta)/4 \\
\theta_{17} &= (9\eta\zeta)/2 - (9\zeta)/2 - (27\tau\zeta)/4 + (9\xi\zeta)/2 - (27\eta\zeta^2)/2 + (27\tau\zeta^2)/4 - (27\xi\zeta^2)/2 + 18\zeta^2 - (27\zeta^3)/2 \\
&+ (27\eta\tau\zeta)/4 + (27\tau\xi\zeta)/4 \\
\theta_{18} &= (27\xi\zeta^2)/2 - (9\xi\zeta)/2 - (27\tau\xi\zeta)/4 \\
\theta_{19} &= (27\eta\zeta^2)/2 - (9\eta\zeta)/2 - (27\eta\tau\zeta)/4 \\
\theta_{20} &= \zeta + (9\tau\zeta)/4 - (27\tau\zeta^2)/4 + (9\tau^2\zeta)/2 - (9\zeta^2)/2 + (9\zeta^3)/2 \\
\theta_{21} &= 9\tau - 12\eta\tau - 12\tau\xi - 12\tau\zeta + 9\eta\tau^2 + 6\eta^2\tau + 6\tau\xi^2 + 9\tau^2\xi + 6\tau\zeta^2 + 9\tau^2\zeta - (45\tau^2)/2 + (27\tau^3)/2 \\
&+ 12\eta\tau\xi + 12\eta\tau\zeta + 12\tau\xi\zeta \\
\theta_{22} &= 12\tau\xi - 12\tau\xi^2 - 12\eta\tau\xi - 12\tau\xi\zeta \\
\theta_{23} &= 6\tau\xi^2 - 9\tau^2\xi \\
\theta_{24} &= 12\eta\tau - 12\eta^2\tau - 12\eta\tau\xi - 12\eta\tau\zeta \\
\theta_{25} &= 12\eta\tau\xi \\
\theta_{26} &= 6\eta^2\tau - 9\eta\tau^2 \\
\theta_{27} &= 12\tau\zeta - 12\tau\zeta^2 - 12\eta\tau\zeta - 12\tau\xi\zeta \\
\theta_{28} &= 12\tau\xi\zeta \\
\theta_{29} &= 12\eta\tau\zeta \\
\theta_{30} &= 6\tau\zeta^2 - 9\tau^2\zeta \\
\theta_{31} &= (3\eta\tau)/2 - (9\tau)/2 + (3\tau\xi)/2 + (3\tau\zeta)/2 - (9\eta\tau^2)/2 - (9\tau^2\xi)/2 - (9\tau^2\zeta)/2 + 18\tau^2 - (27\tau^3)/2 \\
\theta_{32} &= (9\tau^2\xi)/2 - (3\tau\xi)/2 \\
\theta_{33} &= (9\eta\tau^2)/2 - (3\eta\tau)/2 \\
\theta_{34} &= (9\tau^2\zeta)/2 - (3\tau\zeta)/2 \\
\theta_{35} &= \tau - (9\tau^2)/2 + (9\tau^3)/2
\end{aligned}
$$

[1] T. Arbogast, C.S. Huang, and X. Zhao. Accuracy of WENO and adaptive order WENO reconstructions for solving conservation laws. *SIAM J. Numer. Anal.*, 56:1818 – 1847, 2018.

[2] 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.

[3] D. Balsara, C. Altmann, C.D. Munz, and M. Dumbser. A sub-cell based indicator for troubled zones in RKDG schemes and a novel class of hybrid RKDG+HWENO schemes. *Journal of Computational Physics*, 226:586–620, 2007.

[4] 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.

[5] D.S. Balsara. Self-adjusting, positivity preserving high order schemes for hydrodynamics and magnetohydrodynamics. *Journal of Computational Physics*, 231:7504 – 7517, 2012.

[6] D.S. Balsara, S. Garain, V. Florinski, and W. Boscheri. An Efficient Class of WENO Schemes with Adaptive Order for Unstructured Meshes. *Journal of Computational Physics*. submitted to.

[7] D.S. Balsara, S. Garain, and C.-W. Shu. An efficient class of WENO schemes with adaptive order. *Journal of Computational Physics*, 326:780 – 804, 2016.

[8] D.S. Balsara and R. Käppeli. von neumann stability analysis of globally constraint-preserving dgtd and pnpm schemes for the maxwell equations using multidimensional riemann solvers. *Journal of Computational Physics*, 376:1108 – 1137, 2019.

[9] D.S. Balsara and Roger Käppeli. Von neumann stability analysis of globally divergence-free rkdg schemes for the induction equation using multidimensional riemann solvers. *Journal of Computational Physics*, 336:104 – 127, 2017.

[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] 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.

[12] 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.

[13] W. Boscheri. An efficient high order direct ALE ADER finite volume scheme with a posteriori limiting for hydrodynamics and magnetohydrodynamics. *International Journal for Numerical Methods in Fluids*, 134-135:111–129, 2016.

[14] W. Boscheri. High Order Direct Arbitrary-Lagrangian-Eulerian (ALE) Finite Volume Schemes for Hyperbolic Systems on Unstructured Meshes. *Archives of Computational Methods in Engineering*, 24:1–51, 2017.

[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. 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*, 275:484 – 523, 2014.

[18] W. Boscheri and M. Dumbser. Arbitrary-Lagrangian-Eulerian Discontinuous Galerkin schemes with a posteriori subcell finite volume limiting on moving unstructured meshes. *Journal of Computational Physics*, 346:449–479, 2017.

[19] 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.

[20] W. Boscheri, M. Dumbser, and M. Righetti. FORCE schemes on moving unstructured meshes for hyperbolic systems. *Computers and Mathematics with Applications*, 78:362–380, 2019.

[21] W. Boscheri and R. Loubère. High order accurate direct Arbitrary-Lagrangian-Eulerian ADER-MOOD finite volume schemes for non-conservative hyperbolic systems with stiff source terms. *Communications in Computational Physics*, 21:271–312, 2017.

[22] W. Boscheri, R. Loubère, and M. Dumbser. Direct Arbitrary-Lagrangian-Eulerian ADER-MOOD finite volume schemes for multidimensional hyperbolic conservation laws. *Journal of Computational Physics*, 292:56–87, 2015.

[23] W. Boscheri, M. Semplice, and M. Dumbser. Central WENO subcell finite volume limiters for ADER discontinuous Galerkin schemes on unstructured meshes. *Communications in Computational Physics*, 25:311 – 346, 2019.

[24] M.R.J. Charest, C.P.T. Groth, and P.Q. Gauthier. A High-Order Central ENO Finite-Volume Scheme for Three-Dimensional Low-Speed Viscous Flows on Unstructured Mesh. *Communications in Computational Physics*, 17:615 – 656, 2015.

[25] J. Cheng, T. Liu, and H. Luo. A hybrid reconstructed discontinuous Galerkin method for compressible flows on arbitrary grids. *Computers and Fluids*, 139:68 – 79, 2016.

[26] 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.

[27] J. Cheng and C.W. Shu. A cell-centered Lagrangian scheme with the preservation of symmetry and conservation properties for compressible fluid flows in two-dimensional cylindrical geometry. *Journal of Computational Physics*, 229:7191–7206, 2010.

[28] J. Cheng, F. Zhang, and T. Liu. A high order compact least-squares reconstructed discontinuous Galerkin method for the steady-state compressible flows on hybrid grids. *Journal of Computational Physics*, 362:95 – 113, 2018.

[29] S. Clain, S. Diot, and R. Loubère. A high-order finite volume method for systems of conservation lawsmulti-dimensional optimal order detection (MOOD). *Journal of Computational Physics*, 230(10):4028 – 4050, 2011.

[30] B. Cockburn, G. E. Karniadakis, and C.W. Shu. *Discontinuous Galerkin Methods*. Lecture Notes in Computational Science and Engineering. Springer, 2000.

[31] B. Cockburn, M. Luskin, C. W. Shu, and E. Suli. Enhanced accuracy by post-processing for finite element methods for hyperbolic equations. *Mathematics of Computation*, 72:577–606, 2003.

[32] B. Cockburn, M. Luskin, C. W. Shu, and E. Suli. Extension of a post-processing technique for the discontinuous galerkin method for hyperbolic equations with applications to an aeroacoustic problem. *SIAM Journal on Scientific Computing*, 26:821–843, 2005.

[33] I. Cravero, G. Puppo, M. Semplice, and G. Visconti. CWENO: uniformly accurate reconstructions for balance laws. *Mathematics of Computation*, 87:1689–1719, 2018.
[34] I. Cravero and M. Semplice. On the accuracy of WENO and CWENO reconstructions of third order on nonuniform meshes. *Journal of Scientific Computing*, 67:1219–1246, 2016.
[35] J.N. de La Rosa and C.-D. Munz. Hybrid DG/FV schemes for magnetohydrodynamics and relativistic hydrodynamics. *Computer Physics Communications*, 222:113–135, 2018.
[36] 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.
[37] 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.
[38] 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.
[39] V.A. Dobrev, T.E. Ellis, T.V. Kolev, and R.N. Rieben. High-order curvilinear finite elements for axisymmetric lagrangian hydrodynamics. *Computers & Fluids*, 83(0):58 – 69, 2013.
[40] 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.
[41] M. Dubiner. Spectral methods on triangles and other domains. *Journal of Scientific Computing*, 6:345–390, 1991.
[42] J.K. Dukovicz and B. Meltz. Vorticity errors in multidimensional lagrangian codes. *Journal of Computational Physics*, 99:115 – 134, 1992.
[43] M. Dumbser. *Arbitrary High Order Schemes for the Solution of Hyperbolic Conservation Laws in Complex Domains*. Shaker Verlag, Aachen, 2005.
[44] 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.
[45] M. Dumbser, W. Boscheri, M. Semplice, and G. Russo. Central WENO schemes for hyperbolic conservation laws on fixed and moving unstructured meshes. *SIAM Journal on Scientific Computing*, 39:A2564 – A2591, 2017.
[46] 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.
[47] M. Dumbser and M. Kaeser. Arbitrary high order non-oscillatory finite volume schemes on unstructured meshes for linear hyperbolic systems. *Journal of Computational Physics*, 221:693 – 723, 2007.
[48] 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.
[49] M. Dumbser and R. Loubère. A simple robust and accurate a posteriori sub-cell finite volume limiter for the discontinuous Galerkin method on unstructured meshes. *Journal of Computational Physics*, 319:163–199, 2016.
[50] 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.
[51] M. Dumbser, O. Zanotti, R. Loubère, and S. Diot. A posteriori subcell limiting of the discontinuous Galerkin finite element method for hyperbolic conservation laws. *Journal of Computational Physics*, 278:47–75, 2014.
[52] F.Vilar. Cell-centered discontinuous Galerkin discretization for two-dimensional Lagrangian hydrodynamics. *Computers and Fluids*, 64:64–73, 2012.
[53] 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.
[54] 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.
[55] 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.
[56] S.K. Godunov, A. Zabrodine, M. Ivanov, A. Kraiko, and G. Prokopov. *Résolution numérique des problèmes multidimensionnels de la dynamique des gaz*. Mir, 1979.
[57] C. Hu and C.W. Shu. A high-order weno finite difference scheme for the equations of ideal magnetohydrodynamics. *Journal of Computational Physics*, 150:561 – 594, 1999.
[58] Z. Jia and S. Zhang. A new high-order discontinuous Galerkin spectral finite element method for Lagrangian gas dynamics in two-dimensions. *Journal of Computational Physics*, 230:2496 – 2522, 2011.
[59] G.S. Jiang and C.W. Shu. Efficient implementation of weighted eno schemes. *Journal of Computational Physics*, pages 202–228, 1996.
[60] J.Qiu and C-W.Shu. Hermite weno schemes and their application as limiters for runge-kutta discontinuous galerkin method: One-dimensional case. *J. Comput. Phys.*, 193(1):115–135, January 2004.
[61] J.R. Kamm and F.X. Timmes. On efficient generation of numerically robust sedov solutions. *Technical Report LA-UR-07-2849,LANL*, 2007.
[62] G. E. Karniadakis and S. J. Sherwin. *Spectral/hp Element Methods in CFD*. Oxford University Press, 1999.
[63] M. Käser and A. Iske. Ader schemes on adaptive triangular meshes for scalar conservation laws. *Journal of Computational Physics*, 205:486 – 508, 2005.
[64] R.E. Kidder. Laser-driven compression of hollow shells: power requirements and stability limitations. *Nucl. Fus.*, 1:3 – 14, 1976.
[65] 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.
[66] 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.
[67] 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.

[68] R. Kumar and P. Chandrashekar. Simple smoothness indicator and multi-level adaptive order WENO scheme for hyperbolic conservation laws. *Journal of Computational Physics*, 375:1059 – 1090, 2018.
[69] D. Levy, G. Puppo, and G. Russo. Central WENO schemes for hyperbolic systems of conservation laws. *M2AN Math. Model. Numer. Anal.*, 33(3):547–571, 1999.
[70] 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.
[71] R. Liska, M.J. Shashkov P. Váchal, and B. Wendroff. Synchronized flux corrected remapping for ALE methods. *Computers and Fluids*, 46:312–317, 2011.
[72] 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.
[73] X. Liu, N. Morgan, and D. Burton. A high-order Lagrangian discontinuous Galerkin hydrodynamic method for quadratic cells using a subcell mesh stabilization scheme. *Journal of Computational Physics*, 386:110 – 157, 2019.
[74] X. Liu, N.R. Morgan, and D.E. Burton. A Lagrangian discontinuous Galerkin hydrodynamic method. *Computers and Fluids*, 163:68 – 85, 2018.
[75] J. Lou, L. Li, H. Luo, and H. Nishikawa. Reconstructed discontinuous Galerkin methods for linear advection-diffusion equations based on first-order hyperbolic system. *Journal of Computational Physics*, 369:103 – 124, 2018.
[76] 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.
[77] 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.
[78] R. Loubère, J. Ovadia, and R. Abgrall. A Lagrangian Discontinuous Galerkin-type method on unstructured meshes to solve hydrodynamics problems. *JInternational Journal for Numerical Methods in Fluids*, 44:645 – 663, 2004.
[79] 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.
[80] 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.
[81] N. Morgan, X. Liu, and D. Burton. Reducing spurious mesh motion in Lagrangian finite volume and discontinuous Galerkin hydrodynamic methods. *Journal of Computational Physics*, 372:35 – 61, 2018.
[82] 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.
[83] 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.
[84] 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.
[85] S. Osher and F. Solomon. Upwind difference schemes for hyperbolic conservation laws. *Math. Comput.*, 38:339–374, 1982.
[86] A. Pandare, C. Wang, H. Luo, and M. Shashkov. A Reconstructed Discontinuous Galerkin Method for Compressible Multiphase Flows in Lagrangian Formulation. *AIAA*, 0595, 2018.
[87] I. Peshkov and E. Romenski. A hyperbolic model for viscous Newtonian flows. *Continuum Mechanics and Thermodynamics*, 28:85–104, 2016.
[88] J. Qiu and C.W. Shu. Hermite WENO schemes and their application as limiters for Runge-Kutta discontinuous Galerkin method II: two dimensional case. *Computers and Fluids*, 34:642–663, 2005.
[89] J. Qiu and C.W. Shu. Runge-Kutta discontinuous Galerkin method using WENO limiters. *SIAM Journal on Scientific Computing*, 26:907–929, 2005.
[90] 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.
[91] G. Scovazzi. Lagrangian shock hydrodynamics on tetrahedral meshes: A stable and accurate variational multiscale approach. *Journal of Computational Physics*, 231:8029–8069, 2012.
[92] M. Sonntag and C.-D. Munz. Shock capturing for discontinuous galerkin methods using finite volume subcells. *Springer Proceedings in Mathematics and Statistics*, 78:945–953, 2014.
[93] V.A. Titarev, P. Tsoutsanis, and D. Drikakis. WENO schemes for mixed–element unstructured meshes. *Communications in Computational Physics*, 8:585–609, 2010.
[94] E.F. Toro and S. J. Billet. Centered TVD schemes for hyperbolic conservation laws. *IMA Journal of Numerical Analysis*, 20:44–79, 2000.
[95] E.F. Toro, A. Hidalgo, and M. Dumbser. FORCE schemes on unstructured meshes I: Conservative hyperbolic systems. *Journal of Computational Physics*, 228:3368–3389, 2009.
[96] 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.
[97] B. van Leer and S. Nomura. Discontinuous Galerkin for diffusion. In *Proceedings of 17th AIAA Computational Fluid Dynamics Conference (June 6–9 2005), AIAA-2005-5108*.
[98] C. Wang, H. Luo, and A. Kashi. Reconstructed Discontinuous Galerkin Methods for Compressible Flows in ALE Formulation. *AIAA*, 0596, 2019.
[99] C. Wang, M. Shashkov, and H. Luo. A Reconstructed Discontinuous Galerkin Method for Compressible Flows in Lagrangian Formulation. *Journal of Computational Physics*. submitted.
[100] P. Woodward and P. Colella. The numerical simulation of two-dimensional fluid flow with strong shocks. *Journal of Computational Physics*, 54:115–173, 1984.
[101] T. Wu, M. Shashkov, N. Morgan, D. Kuzmin, and H. Luo. An updated Lagrangian discontinuous Galerkin hydrodynamic method for gas dynamics. *Computers and Mathematics with Applications*, 2018.

[102] X.Zhong and C-W.Shu. A simple weighted essentially nonoscillatory limiter for runge-kutta discontinuous galerkin methods. *J. Comput. Phys.*, 232(1):397–415, January 2013.

[103] O. Zanotti, F. Fambri, M. Dumbser, and A. Hidalgo. Space–time adaptive ADER discontinuous Galerkin finite element schemes with a posteriori sub–cell finite volume limiting. *Computers and Fluids*, 118:204–224, 2015.

[104] Y.T. Zhang and C.W. Shu. Third order WENO scheme on three dimensional tetrahedral meshes. *Communications in Computational Physics*, 5:836–848, 2009.

[105] J. Zhu and C.W. Shu. A new type of multi-resolution WENO schemes with increasingly higher order of accuracy. *Journal of Computational Physics*, 375:659 – 683, 2018.

[106] J. Zhu, X. Zhong, C.W. Shu, and J. Qiu. Runge-Kutta discontinuous Galerkin method using a new type of WENO limiters on unstructured meshes. *Journal of Computational Physics*, 248:200–220, 2013.