The MOOD method in the three-dimensional case: Very-High-Order Finite Volume Method for Hyperbolic Systems.

Steven Diot, Raphaël Loubère, Stéphane Clain

INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
Int. J. Numer. Meth. Fluids 2012; **00**:1–34
Published online in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/fld

The MOOD method in the three-dimensional case: Very-High-Order Finite Volume Method for Hyperbolic Systems.

S. Diot*[,1], R. Loubère[1], S. Clain[1,2]

[1]*Institut de Mathématiques de Toulouse, Université de Toulouse, France.*
[2]*Departamento de Matemática e Aplicações, Campus de Gualtar - 4710-057 Braga, Campus de Azurém - 4800-058 Guimarães, Portugal.*

SUMMARY

The Multi-dimensional Optimal Order Detection (MOOD) method for two-dimensional geometries has been introduced in "A high-order finite volume method for hyperbolic systems: Multi-dimensional Optimal Order Detection (MOOD)", J. Comput. Phys. 230 (2011), and enhanced in "Improved Detection Criteria for the Multi-dimensional Optimal Order Detection (MOOD) on unstructured meshes with very high-order polynomials", Comput. & Fluids 64 (2012). We present in this paper the extension to 3D mixed meshes composed of tetrahedra, hexahedra, pyramids and prisms. In addition, we simplify the $u2$ detection process previously developed and show on a relevant set of numerical tests for both the convection equation and the Euler system that the optimal high-order of accuracy is reached on smooth solutions while spurious oscillations near singularities are prevented. At last, the intrinsic positivity-preserving property of the MOOD method is confirmed in 3D and we provide simple optimizations to reduce the computational cost such that the MOOD method is very competitive compared to existing high-order Finite Volume methods.

Received . . .

KEY WORDS: Finite Volume; high-order; conservation law; polynomial reconstruction; 3D; unstructured; Euler; MOOD; positivity-preserving

CONTENTS

*Correspondence to: steven.diot@math.univ-toulouse.fr

1. INTRODUCTION

First introduced in [5], the Multi-dimensional Optimal Order Detection (MOOD) method proposes a new strategy to provide third-order approximations to hyperbolic scalar or vectorial problems for two-dimensional geometry with structured meshes. The author then gave an extension in [11] to general unstructured 2D meshes where they achieved a sixth-order convergence in space introducing new detection-limitation procedure. The issue we address in the present paper is to extend the MOOD method to three-dimensional geometries with general polyhedral unstructured meshes for the scalar advection equation and the hydrodynamics Euler system. The method casts in the generic framework of the finite volume method but fundamentally differs from the traditional techniques by the specific detection-limitation procedure implemented by the authors. Indeed, classical high-order polynomial reconstruction-based schemes such as MUSCL [21, 35, 36, 22, 18, 3, 23] or ENO/WENO methods [15, 16, 29, 39, 27] rely on an *a priori* limiting procedure to achieve some stability properties. For instance, in MUSCL-like methods unlimited slopes are reduced through the use of limiters to respect some Discrete Maximum Principle or Total Variation Diminishing properties. In the same way, ENO/WENO-like methods employ an essentially non-oscillatory polynomial which provides an accurate solution while preventing undesirable oscillations from appearing.

We state that such limitation strategies are *a priori* in the sense that only the data at time t^n are used to first perform the limitation procedure and then compute an approximation at time t^{n+1}. Generally, the "worst case scenario" (speculative approach) has to be considered as plausible and, consequently a "precautionary principle" is applied. It results that most of the time the limitation mechanism unnecessarily operates and may reduce the scheme accuracy due to restrictive assessments. The MOOD principle lies in a different approach since we first compute a candidate solution for time t^{n+1} and use this *a posteriori* information to check if the proposed approximation is valid. Roughly speaking, we compute a candidate solution without any limitation using local polynomial reconstructions to provide accurate approximation of the flux (the degree is set to a prescribed maximal value). We then detect if this solution locally fails to fulfill some stability criteria (detection of problematic cells) and further decrement polynomial degree only on problematic regions (limitation step) before recomputing a new candidate solution. An iterative procedure (the MOOD algorithm) is carried out by successively decrementing the degree to provide the *optimal* local polynomial reconstruction for each cell to satisfy the given stability criteria. At the end of the MOOD algorithm, the candidate solution is eligible and turns out to be the approximation at time t^{n+1}. The *a posteriori* strategy brings new benefits. We dramatically reduce the number of polynomial reconstructions regarded to the ENO/WENO method since our technique only requires one polynomial function for each cell. Most of the time, the polynomial with maximal degree is employed since the limitation mechanism is only activated for problematic cells (objective approach). From a physical point of view, the positivity preserving property (for the Euler equations as instance) is simply guaranteed by the *a posteriori* strategy applying a simple detection procedure which checks the physical admissibility of the solution.

The paper is organized as follows. In section 2, we detail the concept of the MOOD method, while the detection criteria are developed in section 3 both for the advection equation and the hydrodynamics Euler system of equations. Numerical tests are proposed in section 4 to prove the efficiency of the method: we first consider the scalar advection equation and show effective high-order of accuracy for regular solutions with the fourth- and sixth-order schemes considering meshes made of hexahedra and pyramids. We then propose an H-shaped discontinuous profile in rotation to verify the non-oscillatory property of the MOOD method. Finally, numerical simulations are carried out for the Euler system to test the method with a nonlinear vectorial problem. As preliminary experiments, the classical 1D test cases, namely the Sod and Lax shock tubes and the Shu-Osher and Woodward-Collela problems, are run on 3D tetrahedral and pyramidal meshes. Then the numerical order of accuracy is checked on the 2D isentropic vortex in motion extended by

DOI: 10.1002/fld

invariance for a 3D mesh and a realistic 2D test case (introduced in [11]), namely the *impact of a shock wave on a cylindrical cavity*, is carried out on a mesh made of triangular and quadrangular prisms. At last, we present results for the 3D explosion problem on a pyramidal mesh and the interaction of a shock wave with a quarter of cone on a mesh of 1.1 millions of tetrahedra. We moreover provide computational cost (CPU and memory storage) for the 3D explosion problem for the MOOD method for different polynomial degrees. We conclude with section 5 and delineate some future perspectives.

2. THE MOOD CONCEPT

We consider the generic hyperbolic equation defined on a domain $\Omega \subset \mathbb{R}^3$, $t > 0$ cast in the conservative form

$$\partial_t U + \nabla \cdot F(U) = 0, \tag{1a}$$
$$U(\cdot, 0) = U_0, \tag{1b}$$

where $U = U(\mathbf{x}, t)$ is the vector of unknown functions depending on $\mathbf{x} = (x, y, z) \in \Omega$ and on the time t. We denote by F the so-called physical flux where we shall consider the autonomous case $F = F(U)$ (Euler system as instance) and the non-autonomous situation $F = F(\mathbf{x}, U)$ such as $\nabla_{\mathbf{x}}.F(\mathbf{x}, .) = 0$ (scalar convection case). Function u_0 stands for the initial condition while the boundary conditions will be prescribed in section devoted to the numerical simulations.

2.1. Framework

In order to design the numerical scheme, we introduce the following notation illustrated in Figure 1. The computational domain Ω is assumed to be a polyhedron bounded set of $\mathbb{R}^3$ divided into polyhedral cells K_i, $i \in \mathcal{E}_{el}$ where $\mathcal{E}_{el}$ is the cell index set. For each cell K_i, we denote by $\mathbf{c}_i$ the cell centroid, and define the set $\underline{\nu}(i)$ of all the indexes $j \in \mathcal{E}_{el}$ such that elements K_j share a common face f_{ij} with K_i and the set $\overline{\nu}(i)$ of all the indexes $j \in \mathcal{E}_{el}$ such that $K_i \cap K_j \neq \emptyset$ (see illustrations in Figure 2). Moreover for each face $f_{ij} = K_i \cap K_j$, $\mathbf{n}_{ij}$ stands for the unit normal vector going from K_i to K_j and we denote by $\left(\xi_{ij,r}, q_{ij,r}\right)$, $r = 1, ..., R_{ij}$ the quadrature rule for the numerical integration on f_{ij} where $\xi_{ij,r}$ is the weight associated to the r^{th} quadrature point $q_{ij,r}$ with $\sum_{r=1}^{R} \xi_{ij,r} = 1$, $\forall i \in \mathcal{E}_{el}$ and $\forall j \in \underline{\nu}(i)$ (see Figure 1).

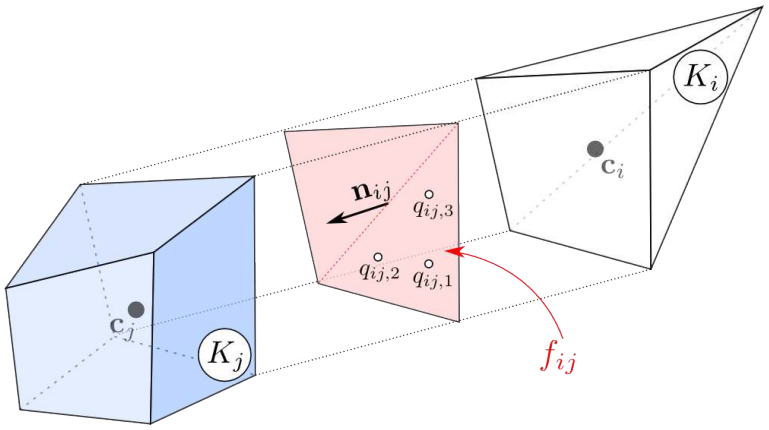

Figure 1. Notation for a three-dimensional mesh: exploded view of the face f_{ij} between two cells K_i and K_j. Centroids are respectively denoted by $\mathbf{c}_i$ and $\mathbf{c}_j$. Three quadrature points $q_{ij,r}, r = 1, 2, 3$, on f_{ij} are drawn for illustration. The unit normal vector pointing from K_i to K_j is denoted $\mathbf{n}_{ij}$.

DOI: 10.1002/fld

To avoid a specific treatment of the boundary faces we introduce the notion of virtual cell. To this end, assuming that cell K_i has a face $f_{ie} = K_i \cap \partial\Omega$ on $\partial\Omega$, we introduce the virtual cell K_j where $j \notin \mathcal{E}_{el}$ obtained by symmetrical transformation of the original cell K_i which represents the exterior side of Ω. We shall denote by $\mathcal{E}_{bd}$ the index set of all virtual cells such that $\widetilde{\mathcal{E}_{el}} = \mathcal{E}_{el} \cup \mathcal{E}_{bd}$ is the index set of all cells (including the virtual ones).

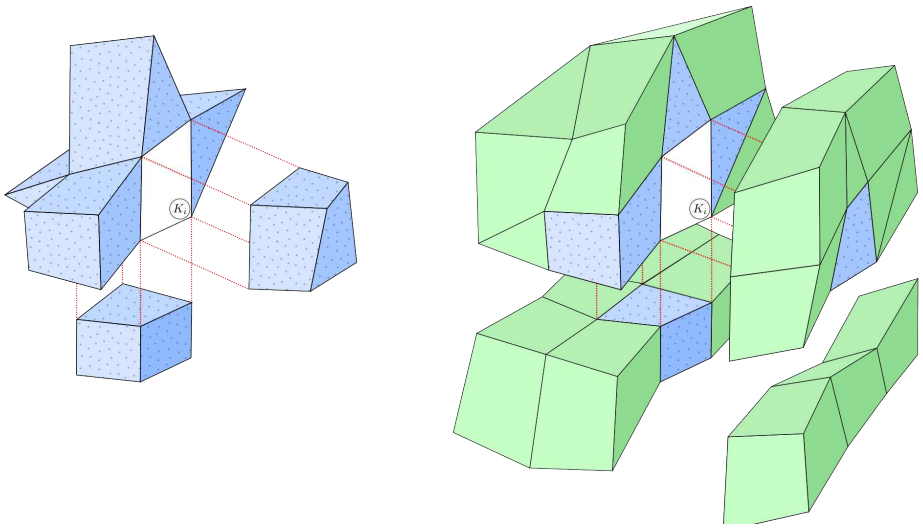

Figure 2. Illustrations for index sets $\underline{\nu}(i)$ (left) and $\overline{\nu}(i)$ (right) in 3D.

The generic first-order Finite Volume scheme associated to equation (1) writes

$$U_i^{n+1} = U_i^n - \Delta t \sum_{j \in \underline{\nu}(i)} \frac{|f_{ij}|}{|K_i|} \mathbb{F}(U_i^n, U_j^n, \mathbf{n}_{ij}), \tag{2}$$

where U_i^n is an approximation of the mean value of U at time t^n on K_i, $\mathbb{F}(U_i^n, U_j^n, \mathbf{n}_{ij})$ is a numerical flux which satisfies the properties of consistency and monotonicity for the scalar case, Δt stands for the time step while $|f_{ij}|$ and $|K_i|$ are the area of face f_{ij} and the volume of cell K_i respectively.

To provide high-order finite volume schemes, we use convex combinations of the initial building-block (2) with better approximations at the quadrature points to compute the numerical flux (see [5, 11] for instance). The high-order schemes are thus obtained from an original first-order Finite Volume and that is of crucial importance from a computational and implementation point of view (re-use of the original first-order code to achieve high-order approximations). We substitute the first-order approximations of the flux integral by higher-order versions, the scheme then writes

$$U_i^{n+1} = U_i^n - \Delta t \sum_{j \in \underline{\nu}(i)} \frac{|f_{ij}|}{|K_i|} \sum_{r=1}^{R_{ij}} \xi_{ij,r} \mathbb{F}(U_{ij,r}^n, U_{ji,r}^n, \mathbf{n}_{ij}), \tag{3}$$

where $U_{ij,r}^n, U_{ji,r}^n$ are high-order approximations of U at quadrature points $q_{ij,r}$ on both side of f_{ij}.

For meshes constituted of tetrahedral cells, all faces are triangles. Consequently R_{ij} and $\xi_{ij,r}$ are independent of i and j and the previous scheme rewrites as a convex combination of the first-order scheme (2)

$$U_i^{n+1} = \sum_{r=1}^{R} \xi_r \left(U_i^n - \Delta t \sum_{j \in \underline{\nu}(i)} \frac{|f_{ij}|}{|K_i|} \mathbb{F}(U_{ij,r}^n, U_{ji,r}^n, \mathbf{n}_{ij}) \right). \tag{4}$$

DOI: 10.1002/fld

Remark 2.1. *When dealing with general polyhedral cells, the quadrature rules may be different from a face to another and such a convex combination is not valid anymore. However, since each polygonal face can be split into triangles, one can recover equation (4) by considering each polyhedron as a polyhedron only constituted by triangular faces (and consequently with more faces than the original one).*

Let denote by $U_h^n = \sum_{i\in\mathcal{E}_{el}} U_i^n \mathbb{1}_{K_i}$ the constant piecewise representation of approximation $(U_i^n)_{i\in\mathcal{E}_{el}}$, we introduce operator $\mathcal{H}^R(U_h^n)$ such that relation (3) rewrites as

$$U_h^{n+1} = U_h^n + \Delta t \mathcal{H}^R(U_h^n). \tag{5}$$

From the original forward Euler discretization in time (5) we derive a high-order approximation in time using a Runge–Kutta 3 TVD method:

$$U_h^{n+1} = \frac{U_h^n + 2U_h^{(3)}}{3} \quad \text{with} \quad \left\{ \begin{array}{rcl} U_h^{(1)} & = & U_h^n + \Delta t\ \mathcal{H}^R(U_h^n) \\ U_h^{(2)} & = & U_h^{(1)} + \Delta t\ \mathcal{H}^R(U_h^{(1)}) \\ U_h^{(3)} & = & \widehat{U}_h^{(2)} + \Delta t\ \mathcal{H}^R(\widehat{U}_h^{(2)}) \end{array} \right. \tag{6}$$

where $\widehat{U}_h^{(2)}$ is the convex combination $(3U_h^n + U_h^{(2)})/4$.

The time discretization introduces a 3^{rd}-order error which makes the whole scheme to be formally 3^{rd}-order accurate. However setting $\Delta t = \Delta \mathbf{x}^{r/3}$ where r is the spatial order of accuracy and $\Delta\mathbf{x}$ is a characteristic length provide same order for spatial and time errors.

2.2. *Reconstruction*

We have formally defined an arbitrary high-order accurate Finite Volume scheme, providing that $U_{ij,r}$ is a high-order accurate point-wise approximation of $U(q_{ij,r})$ computed from cell K_i. In this subsection, we briefly describe the technique to produce such approximations and we refer to [5, 11] and references herein for details. Let us consider a scalar variable u and denote by $\widetilde{u}_i(\cdot, \mathsf{d})$ a local polynomial approximation of degree d reconstructed on cell K_i from the mean values of function u on a set of neighboring cells $\mathcal{S}_i^{\mathsf{d}}$ called stencil. For the sake of conservation, *i.e.* $\frac{1}{|K_i|}\int_{K_i} \widetilde{u}_i(\mathbf{x};\mathsf{d})\ d\mathbf{x} = u_i$, we assume that the polynomial has the following structure

$$\widetilde{u}_i(\mathbf{x};\mathsf{d}) = u_i \ + \sum_{1\le|\boldsymbol{\alpha}|\le\mathsf{d}} \mathcal{R}_i^{\boldsymbol{\alpha}} \left((\mathbf{x}-\mathbf{c})^{\boldsymbol{\alpha}} - \frac{1}{|K|}\int_K (\mathbf{x}-\mathbf{c})^{\boldsymbol{\alpha}}\ d\mathbf{x} \right), \tag{7}$$

where the polynomial coefficients $\mathcal{R}_i^{\boldsymbol{\alpha}}$ are fixed by solving a least-squares problem equivalent to minimizing the functional

$$E = \sum_{j\in\mathcal{S}_i^{\mathsf{d}}} \left(\frac{1}{|K_j|}\int_{K_j} \widetilde{u}_i(\mathbf{x};\mathsf{d})\ d\mathbf{x} - u_j \right)^2.$$

In practice, the polynomial coefficients are obtained by multiplying the pseudoinverse of the least-square problem matrix (that we store in memory) with the vector of mean values on the stencil, see [11] for details. We moreover recall that for the vectorial case, the reconstructions are performed for all the conservative components independently.

Finally, considering that polynomial reconstructions $\widetilde{u}_i(\mathbf{x};\mathsf{d})$ are provided for all cells $K_i, i \in \mathcal{E}_{el}$, we compute the approximation at each quadrature point of each face f_{ij} by $u_{ij,r} = \widetilde{u}_i(q_{ij,r};\mathsf{d})$. The so-called ($\mathsf{d}$-)unlimited scheme (3) is thus defined by employing the reconstructed values in the numerical flux without any restriction (*i.e.* no limitation).

DOI: 10.1002/fld

Remark 2.2. *We recall that the reconstruction process is very time and memory consuming and would like to emphasize that contrarily to WENO methods we consider only one reconstruction stencil per cell and per degree, so that a lot of computational resources are saved.*

2.3. The MOOD concept

It is well-known that the first-order scheme (2) is robust but tremendously diffusive, while unlimited schemes of higher-order produce spurious oscillations in the vicinity of steep gradients. Limitation mechanisms have been developed to prevent the oscillations from appearing, such as the slope or flux limitation in MUSCL methods [21, 35, 36, 22, 18, 3, 23] or the computation of an Essentially Non-Oscillatory polynomial reconstruction in WENO methods [15, 16, 29, 39, 27]. As we mention in the introduction, all the classical techniques act *a priori* in the sense that we determine the limitation process in function of the current data (*i.e.* the solution at time t^n). As a consequence, the *a priori* strategy imposes very drastic accuracy reduction due to strong and unnecessary limitations (the worst case scenario has to be considered). Moreover, computational resources are allocated to perform the limitation process where most of the time it is useless. In the ENO/WENO case for instance, several polynomial reconstructions are required even if the solution is locally regular and can be approximated with only one polynomial function.

In two recent papers [5, 11], we have introduced a new approach based on an *a posteriori* evaluation of the solution to determine if the limitation procedure has to be applied and where. The technique is *a posteriori* in the sense that we compute a candidate solution (a potential approximation at time t^{n+1}) and we use the data of the candidate solution to determine if the solution is valid. More precisely, the detection-limitation mechanism operates in several steps. A candidate solution is first computed with the highest-order unlimited scheme (the polynomials with maximal degree). Then a detection procedure is performed to determine the problematic cells, *i.e.* all cells where the approximation does not respect some given criteria (see next section). For problematic cells, the solution is recomputed with a lower-order unlimited scheme (using polynomials with lower degree) and we repeat the procedure detection-degree decrementing (the MOOD algorithm) till the cell satisfy the detection criteria or the polynomial degree is zero. In the last case, a robust first-order scheme (2) is triggered and a meaningful solution is thus provided. Note that we need to guarantee that the MOOD algorithm stops after a finite number of iterations.

We now set some fundamental notions to define the MOOD method. We name Cell Polynomial Degree, shortened as CellPD and denoted by d_i, the degree of the polynomial reconstruction on cell K_i. We name Face Polynomial Degrees, shortened as FacePD and denoted by d_{ij} and d_{ji}, the degrees of the polynomial reconstructions actually used to compute approximations, $U_{ij,r}$ and $U_{ji,r}$, of the solution on face f_{ij} at quadrature points $q_{ij,r}$ respectively from K_i and K_j. The computation of d_{ij} and d_{ji}, named FacePD strategy, consists in evaluating the FacePD $\mathsf{d}_{ij}, \mathsf{d}_{ji}$ that we employ on both sides of the interface f_{ij} with respect to the CellPD of the neighboring cells. In previous studies (see [5] for details), we have proposed and experimented several strategies and introduced the *upper-limiting* property for a FacePD strategy which states that for any degree $\bar{\mathsf{d}}$, the following property holds

$$\mathsf{d}_i = \bar{\mathsf{d}} \Longrightarrow \mathsf{d}_{ij} \leq \bar{\mathsf{d}} \quad \text{and} \quad \mathsf{d}_{ji} \leq \bar{\mathsf{d}}, \quad \forall j \in \underline{\nu}(i).$$

This guarantees (see [5]) that the MOOD algorithm stops after a finite number of iterations. In practice, we use the simple rule $\mathsf{d}_{ij} = \mathsf{d}_{ji} = \min(\mathsf{d}_i, \mathsf{d}_j)$.

As mentioned above, the detection mechanism is performed on the candidate solution $U_h^\star$ and criteria have to be set to specify what is a good solution. To this end, we denote by $\mathcal{A}$ the set of detection criteria (*e.g.* positivity of a variable or a maximum principle) that the numerical approximation has to respect on each cell and we say that a candidate solution is $\mathcal{A}$-eligible if it fulfills all the criteria of $\mathcal{A}$. If the candidate solution is not $\mathcal{A}$-eligible on cell K_i, then we decrement the polynomial degree. However the solution may not be $\mathcal{A}$-eligible regardless of the set $\mathcal{A}$ even if the polynomial degree is zero for the cell. Consequently, we shall consider the solution *acceptable*

DOI: 10.1002/fld

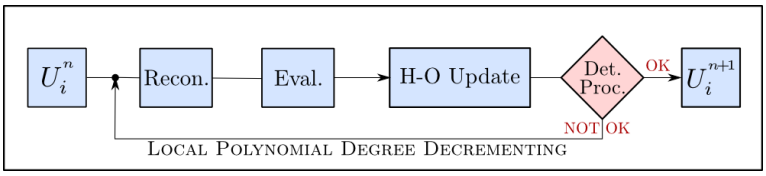

Figure 3. Flowchart of the MOOD algorithm: Recon. stands for polynomial reconstruction, Eval. for high-order evaluations at quadrature points, H-O Update for high-order update of the solution and Det. Proc. for detection process.

on the cell if either it is $\mathcal{A}$-eligible, or is a first-order solution (*i.e.* CellPD has been decremented to zero). To sum up, the MOOD algorithm for the explicit discretization in time consists in the following stages depicted in Figure 3:

0. Initialize $\mathsf{d}_i = \mathsf{d}_{max}, \quad \forall i \in \mathcal{E}_{el}$.
1. Compute polynomial reconstruction of degree $\mathsf{d}_i, \quad \forall i \in \mathcal{E}_{el}$.
2. Compute FacePD d_{ij} and d_{ji} and evaluate high-order approximations at quadrature points on face $f_{ij}, \quad \forall j \in \underline{\nu}(i), \ \forall i \in \mathcal{E}_{el}$.
3. Compute candidate solution mean values through unlimited scheme (3), $\forall i \in \mathcal{E}_{el}$.
4. Detection process: decrement CellPD of cells where solution is not acceptable.
5. Stop if the solution is acceptable else go to stage 1.

Following [5, 11] we extend the MOOD algorithm initially designed for a one-time step scheme to the RK3-TVD scheme by applying it to each sub-step of the RK3-TVD (6) procedure. The MOOD method is now completely defined except from the detection criteria that have to be suited to the problem we intend to solve. Such a difficult task requires the complete next section.

3. DETECTION CRITERIA

The crucial point of the MOOD method is the elaboration of the detection criteria set $\mathcal{A}$ which characterizes the properties we want the numerical solution to fulfill. A fundamental purpose of the detection criteria is to obtain higher-order of accuracy for regular solutions while preventing numerical oscillations in the vicinity of discontinuous profiles. This would consequently provide an efficient and robust method. We face several difficulties to design such a set since accuracy and robustness are antagonist objectives. Moreover, in the Euler problem, a physically admissible solution is mandatory since the positivity of the density and the pressure is required to compute the numerical flux. It results that the detection criteria would cover a wide spectrum of properties and restrictions. A key point we shall detail in the following is the notion of "numerical regularity" in the sense that we have to determine if, for a local stencil and a set of data (for instance the mean values), we can associate a regular or a irregular function. This point is really important since the choice of the reconstruction (namely the polynomial degree) depends on it.

The present section intends to extend and improve detection criteria initially introduced in [11] to evaluate the local "numerical regularity" of the approximation. We first begin the study for the advection equation in section 3.1 and address the hydrodynamics Euler system in section 3.2.

3.1. Advection equation

The scalar advection problem is characterized by the physical flux $F(U) = VU$ where $V \in \mathbb{R}^3$ stands for the velocity that we assume to be a regular function on Ω and satisfies $\nabla_{\mathbf{x}} V(\mathbf{x}) = 0$ while $U = U(t, \mathbf{x}) \in \mathbb{R}$ is the passive scalar quantity transported by the fluid.

When dealing with a constant velocity, the exact solution is simply given by $U(\mathbf{x}, t) = U_0(\mathbf{x} - Vt)$ and clearly fulfills a maximum principle, *e.g.* the minimum of the solution can not be lower

DOI: 10.1002/fld

than the initial condition minimum (and a similar property for the maximum). Consequently, it seems natural to impose such a condition at the numerical level and, as proposed in [5], we integrate in the set $\mathcal{A}$ the Discrete Maximum Principle (DMP) on mean values for the candidate solution $U_h^\star$ formulated like this:

$$\min_{j\in\overline{\nu}(i)}\left(U_i^n, U_j^n\right) \leq U_i^\star \leq \max_{j\in\overline{\nu}(i)}\left(U_i^n, U_j^n\right). \tag{8}$$

A solution is $\mathcal{A}$-eligible if condition (8) is satisfied for all the cells and we have proved in [5] that the scheme equipped with such set $\mathcal{A}$ provides a numerical solution which, under first-order scheme CFL condition, satisfies the DMP. However a strict application of relation (8) at smooth extrema unavoidably reduces the scheme accuracy to two. It suggests that relation (8) is too restrictive and should be relaxed.

In [11] we have relaxed the condition on cells which violate the DMP. More specifically, the relation (8) has been supplemented with a new criteria, the so-called $u2$ detection criteria which provides an effective arbitrary high-order of accuracy. As mention in the beginning of the section, the key point is to determine if the numerical solution is regular enough to be approximated by a high-order polynomial reconstruction and avoid the Gibbs phenomena. To this end, let assume that the candidate solution does not satisfy the DMP criteria on cell K_i. A first step consists in reconstructing quadratic polynomials on K_i denoted by $\widetilde{U}_i$ and on its neighbors K_j for $j \in \overline{\nu}(i)$ denoted by $\widetilde{U}_j$. In a second step, we define approximations to the local minimal and maximal curvatures, namely

$$\mathcal{X}_i^{min} = \min_{j\in\overline{\nu}(i)}\left(\partial_{xx}\widetilde{U}_i,\ \partial_{xx}\widetilde{U}_j\right) \quad \text{and} \quad \mathcal{X}_i^{max} = \max_{j\in\overline{\nu}(i)}\left(\partial_{xx}\widetilde{U}_i,\ \partial_{xx}\widetilde{U}_j\right), \tag{9}$$

$$\mathcal{Y}_i^{min} = \min_{j\in\overline{\nu}(i)}\left(\partial_{yy}\widetilde{U}_i,\ \partial_{yy}\widetilde{U}_j\right) \quad \text{and} \quad \mathcal{Y}_i^{max} = \max_{j\in\overline{\nu}(i)}\left(\partial_{yy}\widetilde{U}_i,\ \partial_{yy}\widetilde{U}_j\right), \tag{10}$$

$$\mathcal{Z}_i^{min} = \min_{j\in\overline{\nu}(i)}\left(\partial_{zz}\widetilde{U}_j,\ \partial_{zz}\widetilde{U}_j\right) \quad \text{and} \quad \mathcal{Z}_i^{max} = \max_{j\in\overline{\nu}(i)}\left(\partial_{zz}\widetilde{U}_i,\ \partial_{zz}\widetilde{U}_j\right), \tag{11}$$

where we emphasize that the second derivatives are constant and naturally referred to as *curvatures*. The $u2$ detection criterion holds in the following definition.

Definition 3.1 ($u2$ detection criterion)**.** *A candidate solution $U_i^\star$ in cell K_i which violates the DMP is nonetheless eligible if the following holds*

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

$$\textit{and} \quad \mathcal{Y}_i^{max}\mathcal{Y}_i^{min} > 0 \quad \textit{and} \quad \left|\frac{\mathcal{Y}_i^{min}}{\mathcal{Y}_i^{max}}\right| \geq 1-\varepsilon,$$

$$\textit{and} \quad \mathcal{Z}_i^{max}\mathcal{Z}_i^{min} > 0 \quad \textit{and} \quad \left|\frac{\mathcal{Z}_i^{min}}{\mathcal{Z}_i^{max}}\right| \geq 1-\varepsilon,$$

where ε is a smoothness parameter.

The definition derives from the idea that the comparison of local second derivatives of the quadratic reconstructions on a neighborhood provides a relevant information on the numerical smoothness of the underlying solution. More precisely, we consider that the underlying solution (characterized by the piecewise constant mean value) is (ε-)smooth if for each direction the curvatures have the same sign (no oscillation or inflection point) and are (ε-)*close* enough to each-other. Such a definition lies in a fitting of the parameter ε the value of which defines the threshold

DOI: 10.1002/fld

between what is considered as smooth extrema or as discontinuity. Therefore the determination of ε is of crucial importance since it rules the decrementing process activation.

From a practical point of view, the $u2$ detection criteria operates in two stages. First the test on the sign of curvatures (left inequalities) is performed. If oscillations are detected, *i.e.* the product is negative, the cell is considered as problematic and the decrementing procedure must be applied. The second stage is performed only if the product is positive. It consists in computing the ratios between minimal and maximal curvatures and comparing it to $1-\varepsilon$ (right inequalities). If the curvatures ratio does not respect the inequality, the cell is considered as problematic and the decrementing process must be applied.

In [11], we propose a parameter ε depending on a local characteristic length and on the spatial dimension of the domain. This was a first attempt to the determination of ε and deeper investigations have shown that a simpler definition provides same quality results. To set the ε value, we extend the parameter as a new function $\varepsilon_x = \varepsilon_x\left(\dfrac{\mathcal{X}_i^{min}}{\mathcal{X}_i^{max}}\right)$ (for the x-direction) with respect to the curvatures which have to satisfy the restriction

$$\frac{\mathcal{X}_i^{min}}{\mathcal{X}_i^{max}} \geq 1-\varepsilon_x. \tag{12}$$

The goal is to determine a relevant function ε_x which enables high-order approximation and robustness. We first note that the curvatures ratio ranges between zero and one so that ε_x must range in $[0,1]$ to make sense. Moreover, the ratio is expected to be close to zero on discontinuities and close to one on smooth functions which are the two extreme cases. For a non-smooth function, we expect that the limiting procedure operates and since the closer to zero ε is, the less smooth the underlying function is considered, ε_x is expected to be close to zero on discontinuities, *i.e.* $\lim\limits_{r\to 0^+}\varepsilon_x(r)=0^+$. On the other hand, a ratio close to one indicates smooth functions, so we expect $\lim\limits_{r\to 1^-}\varepsilon_x(r)=1^-$ to relax the restriction. We thus propose to define ε_x as a continuous increasing function of the curvatures ratio such that $\varepsilon_x(0)=0$ and $\varepsilon_x(1)=1$. After several attempts, it appears that the simple function $\varepsilon_x(r)=r$ is an excellent choice. When substituting expression of $\varepsilon_x = \mathcal{X}_i^{min}/\mathcal{X}_i^{max}$ in relation (12), the x-direction curvatures criterion becomes

$$\frac{\mathcal{X}_i^{min}}{\mathcal{X}_i^{max}} \geq 1-\frac{\mathcal{X}_i^{min}}{\mathcal{X}_i^{max}},$$

and yields

$$\frac{\mathcal{X}_i^{min}}{\mathcal{X}_i^{max}} \geq 1/2.$$

Finally we apply the same reasoning for y- and z-directions and obtain

$$\frac{\mathcal{Y}_i^{min}}{\mathcal{Y}_i^{max}} \geq 1/2 \quad \text{and} \quad \frac{\mathcal{Z}_i^{min}}{\mathcal{Z}_i^{max}} \geq 1/2.$$

The linearity of function ε_x simplifies the final inequalities and leads to the constant value $\varepsilon = 1/2$ in definition 3.1.

Remark 3.2. *The definition of ε is really simpler than the one proposed in [11]. However numerous numerical test cases have been carried out and no change in the quality of results have been reported.*

Remark 3.3. *Numerical experiments show that the choice of the neighborhood where the curvatures are computed should define a convex hull which contains the reference cell K_i. To constitute such a stencil, we used the index set of cells $\underline{\nu}(i)$ in 2D (see [11]) but this choice is not relevant for three-dimensional meshes and we use the index set $\overline{\nu}(i)$ in equations (9)-(11) to provide the expected results even for large form factor meshes.*

DOI: 10.1002/fld

To conclude the section we propose in Figure 4 an algorithmic view of the complete detection process [DMP$\rightarrow u2$] for the advection equation constituted of the DMP of equation (8) relaxed by the $u2$ detection criteria of definition 3.1. We emphasize that the algorithm is given in the case of a cell K_i with $U_i^\star$ its associated candidate solution mean value.

If $\left[U_i^\star \leq \min\limits_{j\in\overline{\nu}(i)} (U_i^n, U_j^n)\right]$ **or** $\left[U_i^\star \geq \max\limits_{j\in\overline{\nu}(i)} (U_i^n, U_j^n)\right]$ **then**

- decrement the CellPD d_i

else

$\rightarrow$ compute the curvatures $\mathcal{X}_i^{min}, \mathcal{X}_i^{max}, \mathcal{Y}_i^{min}, \mathcal{Y}_i^{max}, \mathcal{Z}_i^{min}, \mathcal{Z}_i^{max}$

If $\left[\mathcal{X}_i^{min}\mathcal{X}_i^{max} < 0\right]$ **or** $\left[\mathcal{Y}_i^{min}\mathcal{Y}_i^{max} < 0\right]$ **or** $\left[\mathcal{Z}_i^{min}\mathcal{Z}_i^{max} < 0\right]$ **then**

- decrement the CellPD d_i

else

$\rightarrow$ compute the curvatures ratios $\dfrac{\mathcal{X}_i^{min}}{\mathcal{X}_i^{max}}, \dfrac{\mathcal{Y}_i^{min}}{\mathcal{Y}_i^{max}}, \dfrac{\mathcal{Z}_i^{min}}{\mathcal{Z}_i^{max}}$

If $\left[\dfrac{\mathcal{X}_i^{min}}{\mathcal{X}_i^{max}} < 1-\varepsilon\right]$ **or** $\left[\dfrac{\mathcal{Y}_i^{min}}{\mathcal{Y}_i^{max}} < 1-\varepsilon\right]$ **or** $\left[\dfrac{\mathcal{Z}_i^{min}}{\mathcal{Z}_i^{max}} < 1-\varepsilon\right]$ **then**

- decrement the CellPD d_i

end if

end if

end if

Figure 4. Algorithmic view of the [DMP$\rightarrow u2$] detection process for the advection equation.

3.2. Hydrodynamics Euler system

The Euler system for three-dimensional geometries writes

$$\partial_t \begin{pmatrix} \rho \\ \rho u \\ \rho v \\ \rho w \\ E \end{pmatrix} + \partial_x \begin{pmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ \rho uw \\ u(E+p) \end{pmatrix} + \partial_y \begin{pmatrix} \rho v \\ \rho uv \\ \rho v^2 + p \\ \rho vw \\ v(E+p) \end{pmatrix} + \partial_z \begin{pmatrix} \rho v \\ \rho uw \\ \rho vw \\ \rho w^2 + p \\ w(E+p) \end{pmatrix} = 0, \tag{13}$$

where ρ stands for the density, u, v and w for the velocity components in the x, y and z directions respectively, p for the pressure and E for the total energy. This system is closed by the Equation Of State (EOS) of a perfect gas $p = (\gamma - 1)\rho\epsilon$, where ϵ is the specific internal energy, γ the ratio of specific heats and the total energy is constituted of the internal and kinetic energy

$$E = \rho\left((u^2 + v^2 + w^2)/2 + \epsilon\right).$$

DOI: 10.1002/fld

At last, vector $U = (\rho, \rho u, \rho v, \rho w, E)$ represents the conservative variables of the system while $W = (\rho, u, v, w, p)$ are the primitive ones. Note that contrarily to WENO methods we do not use the characteristic variables.

To provide accurate and oscillation-free solutions we use on the one hand polynomial reconstruction and apply, on the other hand the MOOD algorithm. We mention that all the polynomial reconstructions are performed on conservative variables and only one CellPD is used for all variables (see [11] for the motivations and justifications). We now turn to the detection-limitation procedure and we have to design specific detection criteria for the Euler problem.

Following [11], a first and also mandatory detection criteria corresponds to ensuring the physical meaningfulness of the primitive variables. We then introduce the Physical Admissibility Detection (PAD in short) which considers that the candidate solution on a cell K_i is not valid if we have $\rho_i^\star$ or $p_i^\star$ are negative (after having computed pressure $p_i^\star$). We underline the important property that a high-order scheme (whichever the degree of the polynomial reconstruction) equipped with the PAD and a first-order scheme which preserves the positivity (of density and pressure) under a CFL condition is automatically positivity preserving. This property straightforwardly derives from the *a posteriori* nature of the MOOD method and has been proved in [11].

However the PAD detection process does not prevent spurious oscillations from appearing and we turn to the adaptation of the [DMP$\rightarrow u2$] detection process proposed in [11]. Initially defined for scalar quantity, we apply the [DMP$\rightarrow u2$] on the density ρ (detection) and recall that the decrementing is performed for all variables (limitation). Note that the smoothness parameter ε is still set as $1/2$ in the $u2$ definition as in previous section.

The set of constraints $\mathcal{A}$ for Euler system is thus constituted by the PAD followed by the [DMP$\rightarrow u2$] detection process applied to the density variable since we first check the PAD and if the cell is valid we continue with the [DMP$\rightarrow u2$]detection. In Figure 5 we give an algorithmic view of the complete detection process [PAD$\rightarrow$DMP$\rightarrow u2$] for the hydrodynamics Euler system constituted of the PAD detection criteria, the DMP of equation (8) on the density relaxed by the $u2$ detection criteria of definition 3.1. We emphasize that the algorithm is given in the case of a cell K_i with $U_i^\star = \left(\rho_i^\star, (\rho u)_i^\star, (\rho v)_i^\star, (\rho w)_i^\star, E_i^\star\right)$ its associated candidate solution mean value and that the candidate pressure $p_i^\star$ has to be computed.

We now highlight some implementation aspects about the detection process which enable to improve the solution accuracy. Actually in the above algorithm, the [PAD$\rightarrow$DMP$\rightarrow u2$] performs well but does not, in some cases, fully reach the optimal order of accuracy for smooth solutions. Deeper investigations on the isentropic vortex in motion problem have shown that the detection process inappropriately decrements some cells of the flat region while it operates well in the area where curvatures are not negligible. The undesirable limitation derives from the extra-small *curvatures* treatment by the $u2$ detection where some spurious micro-oscillations take place on the flat area and wrongly activate the curvature sign detection. It results that the sign criterion is not relevant when all the *curvatures* sizes are too small with respect to a mesh parameter δ. To overcome the over-detection phenomena, we introduce a relaxation parameter in the $u2$ criterion to fix the problem.

Definition 3.4 (**u2** detection criterion)**.** *A candidate solution $U_i^\star$ in cell K_i for which the density $\rho_i^\star$ violates the DMP is nonetheless eligible if*

$$\mathcal{X}_i^{max}\mathcal{X}_i^{min} > -\delta \quad \textit{and} \quad \left(\max\left(|\mathcal{X}_i^{max}|, |\mathcal{X}_i^{min}|\right) < \delta \quad \textit{or} \quad \left|\frac{\mathcal{X}_i^{min}}{\mathcal{X}_i^{max}}\right| \geq 1/2\right),$$

$$\textit{and} \quad \mathcal{Y}_i^{max}\mathcal{Y}_i^{min} > -\delta \quad \textit{and} \quad \left(\max\left(|\mathcal{Y}_i^{max}|, |\mathcal{Y}_i^{min}|\right) < \delta \quad \textit{or} \quad \left|\frac{\mathcal{Y}_i^{min}}{\mathcal{Y}_i^{max}}\right| \geq 1/2\right),$$

DOI: 10.1002/fld

$$\textit{and} \quad \mathcal{Z}_i^{max}\mathcal{Z}_i^{min} > -\delta \quad \textit{and} \quad \left(\max\left(|\mathcal{Z}_i^{max}|, |\mathcal{Z}_i^{min}|\right) < \delta \quad \textit{or} \quad \left|\frac{\mathcal{Z}_i^{min}}{\mathcal{Z}_i^{max}}\right| \geq 1/2\right),$$

where δ is the greatest length of geometrical entity of dimension one defined by the length of the cells in $\mathbb{R}$, the maximal length of the cell interfaces in $\mathbb{R}^2$ and the maximal length of edges of the cell interface for three-dimensional meshes.

The correction only damps extra-small oscillations such that minimal and maximal curvatures product satisfies the left condition. When maximal curvatures are larger than δ, the condition on the ratios of curvatures implies that the underlying function will be considered as non-smooth.

Remark 3.5. *The value of δ has been determined after numerous simulation experiments. It enables to fully reach the optimal order for the Euler system but does not affect the method in wisely capturing discontinuous profiles. The correction has even been tested for the convection equation and accuracy losses have not been reported.*

In the same way, we slightly relax the DMP criteria to reduce the computational effort to avoid the waste of resources when performing the $u2$ detection criterion on plateaus. We consider that a DMP violation is not relevant if

$$\max_{j\in\overline{\nu}(i)}(\rho_i^{RK}, U_j^{RK}) - \min_{j\in\overline{\nu}(i)}(\rho_i^{RK}, U_j^{RK}) < \delta^3$$

where index ^{RK} corresponds to one of the Runge-Kutta sub-steps.

The MOOD method for the Euler hydrodynamics system is now completely defined and numerical simulations are carried out for three-dimensional geometries presented in section 4.

If $[\rho_i^\star \leq 0]$ **or** $[p_i^\star \leq 0]$ **then**
- decrement the CellPD d_i

else
- **If** $\left[\rho_i^\star \leq \min_{j\in\overline{\nu}(i)}(\rho_i^n, \rho_j^n)\right]$ **or** $\left[\rho_i^\star \geq \max_{j\in\overline{\nu}(i)}(\rho_i^n, \rho_j^n)\right]$ **then**
  - decrement the CellPD d_i
- **else**
  - $\rightarrow$ compute the curvatures $\mathcal{X}_i^{min}, \mathcal{X}_i^{max}, \mathcal{Y}_i^{min}, \mathcal{Y}_i^{max}, \mathcal{Z}_i^{min}, \mathcal{Z}_i^{max}$ from density ρ at t^n
  - **If** $\left[\mathcal{X}_i^{min}\mathcal{X}_i^{max} < 0\right]$ **or** $\left[\mathcal{Y}_i^{min}\mathcal{Y}_i^{max} < 0\right]$ **or** $\left[\mathcal{Z}_i^{min}\mathcal{Z}_i^{max} < 0\right]$ **then**
    - decrement the CellPD d_i
  - **else**
    - $\rightarrow$ compute the curvatures ratios $\frac{\mathcal{X}_i^{min}}{\mathcal{X}_i^{max}}, \frac{\mathcal{Y}_i^{min}}{\mathcal{Y}_i^{max}}, \frac{\mathcal{Z}_i^{min}}{\mathcal{Z}_i^{max}}$
    - **If** $\left[\frac{\mathcal{X}_i^{min}}{\mathcal{X}_i^{max}} < 1-\varepsilon\right]$ **or** $\left[\frac{\mathcal{Y}_i^{min}}{\mathcal{Y}_i^{max}} < 1-\varepsilon\right]$ **or** $\left[\frac{\mathcal{Z}_i^{min}}{\mathcal{Z}_i^{max}} < 1-\varepsilon\right]$ **then**
      - decrement the CellPD d_i
    - **end if**
  - **end if**
- **end if**

end if

Figure 5. Algorithmic view of the [PAD$\rightarrow$DMP$\rightarrow u2$] detection process for the Euler system.

DOI: 10.1002/fld

3.3. Implementation and optimizations

To conclude the section we detail two important and simple optimizations that we apply to drastically improve the efficiency of the MOOD method.

Local re-updating. The MOOD method may seem computationally expensive since the MOOD algorithm we run for each time step, recompute the candidate solution several times whereas polynomial degrees have been only modified for a small number cells. At the first stage, an initial candidate solution is computed on all cells. Then the MOOD algorithm successively detects and limits the problematic cells. The evaluation of a new candidate solution during the MOOD algorithm by means of scheme (4) only involves the fluxes at the interfaces of corrected cells. Consequently only problematic cells and their neighbors by face must be recomputed. It drastically reduces the computational effort since in most cases the solution is acceptable on more than 80–90% of cells, even when shocks are present.

Reduced polynomial degree decrementing. The original decrementing procedure consists in dropping one-by-one polynomial degrees until zero is reached. Such an approach may be both costly in CPU and memory resources since reconstruction matrices must be stored for all degrees. It would nonetheless still be less memory consuming than for the WENO method due to the large number of polynomial functions involved in the WENO technique. Moreover numerical experiments suggest the following alternative: whether the solution is very smooth, whether the solution presents some discontinuities. To take advantage of it, we change the decrementing strategy by starting from the highest degree, reducing to degree 2 if any and setting degree equal to 0 if the candidate solution is still not $\mathcal{A}$-eligible. We then manage to reduce the number of decrementing stages and save computational resources. We point out that the size of the reconstruction stencil is also an important parameter since a large stencil (required for the maximal degree) will be influenced by a discontinuity located in the second or third layer of cells around the reference one while a more compact one (for a $\mathbb{P}_2$ reconstruction) still preserves the local regularity of the underlying function. Another reason to use the $\mathbb{P}_2$ reconstruction is that it is also used for the $u2$ detection process and always has to be stored.
Therefore in practice, we only store two reconstruction matrices per cell, one for the maximal degree and one for the degree two. It is thus important to remark that the storage cost of the matrix for degree two is always much lower than the one for the maximal degree. Indeed for two-dimensional situations, the memory cost of the pseudoinverse matrix associated to polynomial of degree 2 represents about 10 times 5 elements, about 16 times 9 for $\mathbb{P}_3$ reconstruction and around 28 times 20 for $\mathbb{P}_5$. Analogically for three-dimensional situations, the $\mathbb{P}_2$ reconstruction matrix represents about 16 times 9 elements while it is about 38 times 19 for $\mathbb{P}_3$ and 110 times 55 for $\mathbb{P}_5$.

To conclude this section, we would like to draw some remarks about the potentiality of the MOOD method to be parallelized. Within the MOOD algorithm, only classical unlimited schemes are used without modification so that the parallelization of this part of the method can be done as efficiently as the state-of-the-art methods (WENO method for instance). The only novelty brought by the MOOD method is the iterative process constituting the MOOD algorithm. A potential difficulty comes from the fact that the number of cells on which the numerical scheme acts changes from an iteration of the MOOD algorithm to another, since the procedure is only applied to problematic cells. However it may not dramatically affect the parallelization efficiency: firstly, because an efficient treatment of the list of problematic cells can be achieved and secondly, because the time spent to recompute new candidate solutions is negligible compared to the time to compute the initial one since the number of problematic cells is (in general) very low compared to the total number of cells. The parallelization capacity of the MOOD method is thus as good as the state-of-the-art higher-order finite volume methods.

DOI: 10.1002/fld

4. NUMERICAL RESULTS

The MOOD method has been implemented into a 3D unstructured code dealing with polyhedra having coplanar faces: tetrahedron, hexahedron, pyramid and prism. The polynomial reconstruction procedure is implemented independently of the degree d_{max} and we provide in the present paper numerical results up to $\mathsf{d}_{max} = 5$. Following [11] and remarks in section 3.3, we use the decrementing sequence $\mathbb{P}_{\mathsf{d}_{max}} - \mathbb{P}_2 - \mathbb{P}_0$. The reconstruction matrices are computed and stored in a preprocessing step since they only depend on geometry. Moreover fluxes across faces are approximated by the mean of Gaussian quadrature formulae on a triangular decomposition of the faces (see Figure 1). At last concerning the time discretization, the first-order time step Δt is controlled by a CFL coefficient equal to 0.5. For the convergence studies on smooth solutions we use the time step $\Delta t = \Delta x^{r/3}$ to achieve a global r^{th}-order of accuracy and compute the relative L^1 and L^∞ errors for a a bounded, L^1 function φ by

$$L^1 \text{ error: } \frac{\sum_{i\in\mathcal{E}_{el}} |\varphi_i^N - \varphi_i^{ex}||K_i|}{\sum_{i\in\mathcal{E}_{el}} |\varphi_i^{ex}||K_i|} \quad \text{and} \quad L^\infty \text{ error: } \frac{\max_{i\in\mathcal{E}_{el}} |\varphi_i^N - \varphi_i^{ex}|}{\max_{i\in\mathcal{E}_{el}} |\varphi_i^{ex}|},$$

where $(\varphi_i^{ex})_{i\in\mathcal{E}_{el}}$ and $(\varphi_i^N)_{i\in\mathcal{E}_{el}}$ are respectively the exact and the approximated cell mean values at final time $t = t_{\text{final}}$.

4.1. Advection equation

For the scalar advection equation, the MOOD method is employed with the [DMP$\rightarrow u2$] detection process and two test cases are carried out: the Triple Sine Translation (TST) to assess the effective very high-order of accuracy and the rotation of a discontinuous H-like shape to test its ability to damp the spurious oscillations.

4.1.1. Triple Sine Translation Let Ω be the unit cube. We consider a constant translation velocity $V = (1, 1, 1)$ and the C^∞ initial condition

$$U_0(x, y, z) = \sin(2\pi x)\sin(2\pi y)\sin(2\pi z).$$

The final time is $t_{\text{final}} = 2.0$ and periodic boundary conditions imply that the exact final solution coincides with the initial one. The computations are first carried out on a series of successively refined regular hexahedral meshes from 8^3 to 64^3 cells. To underline the capability of the MOOD method to handle mixed element meshes, we also consider a series of meshes built from a series of regular hexahedral meshes from 4^3 to 48^3 cells into which we regularly split half of cells into 6 pyramids (see top line of Figure 6).

In Figure 6, we display the convergence curves for the L^1 and L^∞ errors of the MOOD-$\mathbb{P}_2$, MOOD-$\mathbb{P}_3$ and MOOD-$\mathbb{P}_5$ methods and give in Table I, the corresponding errors and rates of convergence. As expected, the optimal rate of convergence is achieved. Notice that on the coarsest meshes the initial mean values are not representative of the underlying smooth function and are coherently handled by the method as discontinuous profiles. As such the sine function is under-resolved; for instance in 1D, averaging the function $\sin(2\pi x)$ or an Heaviside-like function on $[0; 1]$ using four cells provides to same mean values.

DOI: 10.1002/fld

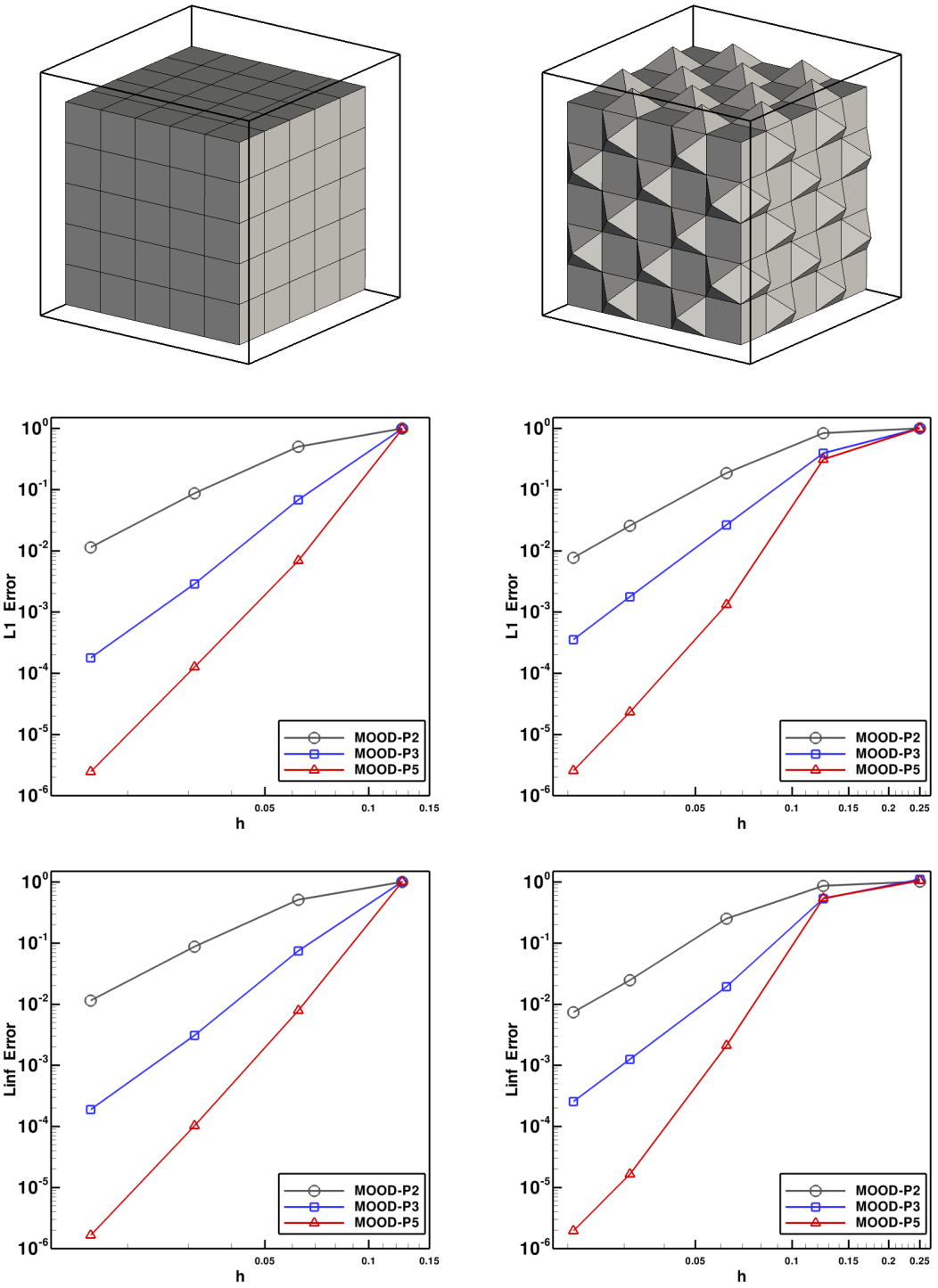

Figure 6. Triple sine translation: convergence curves for L^1 (middle) and L^∞ (bottom) errors for series of hexahedral (left) and hexahedral/pyramidal (right) meshes. Examples of such meshes are given on top line.

DOI: 10.1002/fld

MOOD on hexahedra							
Deg.	h	L^1 error		L^2 error		L^∞ error	
$\mathbb{P}_2$	0.125	9.9682484294e-1	—	9.9680699621e-1	—	1.0038484635	—
	0.0625	5.0100730661e-1	0.99	5.0179199583e-1	0.99	5.1327069642e-1	0.97
	0.03125	8.6946371321e-2	2.53	8.6934402652e-2	2.53	8.7634409525e-2	2.55
	0.015625	1.1429255953e-2	2.93	1.1424251727e-2	2.93	1.1459635442e-2	2.93
Expected order			3		3		3
$\mathbb{P}_3$	0.125	9.8673015719e-1	—	9.8688657294e-1	—	1.0110942145	—
	0.0625	6.8019691177e-2	3.86	6.9068026293e-2	3.84	7.4325569597e-2	3.77
	0.03125	2.8693653411e-3	4.57	2.7741019990e-3	4.64	3.0819084097e-3	4.59
	0.015625	1.7856449887e-4	4.01	1.5709922281e-4	4.14	1.8874795858e-4	4.03
Expected order			4		4		4
$\mathbb{P}_5$	0.125	9.7842521971e-1	—	9.7924733246e-1	—	1.0169454936	—
	0.0625	6.9230110414e-3	7.14	6.9967747247e-3	7.13	7.9478234947e-3	7.00
	0.03125	1.2666634416e-4	5.77	1.1021776542e-4	5.99	1.0247433118e-4	6.28
	0.015625	2.4614368833e-6	5.69	2.0386571852e-6	5.76	1.6605387870e-6	5.95
Expected order			6		6		6
MOOD on mixed hexahedra/pyramids							
Deg.	h	L^1 error		L^2 error		L^∞ error	
$\mathbb{P}_2$	0.25	1.0000027468	—	1.0000052406	—	1.0143912168	—
	0.125	8.3799247906e-1	0.25	8.3412664416e-1	0.26	8.6799172420e-1	0.20
	0.0625	1.8662020042e-1	2.17	1.8646014762e-1	2.16	2.5210598518e-1	1.78
	0.03125	2.5647018453e-2	2.86	2.4729005004e-2	2.91	2.4798614346e-2	3.35
	0.020833	7.6897099615e-3	2.97	7.4071918780e-3	2.97	7.3982102275e-3	2.98
Expected order			3		3		3
$\mathbb{P}_3$	0.25	9.9952627605e-1	—	1.0018017690	—	1.1083447073	—
	0.125	3.9219135702e-1	1.35	4.1718119801e-1	1.26	5.3531820180e-1	1.05
	0.0625	2.6501056786e-2	3.89	2.2797888150e-2	4.19	1.9364138004e-2	4.79
	0.03125	1.7829686262e-3	3.90	1.5178093945e-3	3.91	1.2521397100e-3	3.95
	0.020833	3.5401059785e-4	3.99	3.0038884987e-4	4.00	2.5551614705e-4	3.92
Expected order			4		4		4
$\mathbb{P}_5$	0.25	1.0009285907	—	1.0025919881	—	1.0496962436	—
	0.125	3.1141644019e-1	1.68	4.0086400280e-1	1.32	5.4249781220e-1	0.95
	0.0625	1.3246287256e-3	7.88	1.1541322861e-3	8.44	2.1098853389e-3	8.00
	0.03125	2.3443624169e-5	5.82	2.0015998229e-5	5.85	1.6596915121e-5	6.99
	0.020833	2.0207215760e-6	6.05	1.7188485947e-6	6.05	1.5127171717e-6	5.91
Expected order			6		6		6

Table I. L^1, L^2 and L^∞ errors and convergence rates for the TST problem with the MOOD-$\mathbb{P}_2$, MOOD-$\mathbb{P}_3$ and MOOD-$\mathbb{P}_5$ methods. Top lines: hexahedral meshes. Bottom lines: mixed hexahedral/pyramidal meshes.

DOI: 10.1002/fld

4.1.2. H-like shape rotation We now turn to the rotation of an H-like shape in the unit cube Ω. The initial shape is given by

$$U_0(x,y,z) = \begin{cases} 1 & \text{if } (|x-0.5| > 0.1) \text{ or } (|y-0.5| < 0.1), \\ 0 & \text{elsewhere,} \end{cases}$$

in the cube $[0.2;0.8]^3$ and 0 elsewhere. The rotation axis is the diagonal line joining the origin $(0,0,0)$ and the point $(1,1,1)$. We stop the simulation after one full rotation when the shape is back to its original position. Note that the velocity depends on the spatial position but is divergence-free so that the maximum principle also applies in that case. Numerical simulations are carried out on a 86215 tetrahedra mesh generated by the free mesher Gmsh. Results are displayed with an extruded view on the cut plane $z = 1/2$. Initialization details are illustrated in Figure 7.

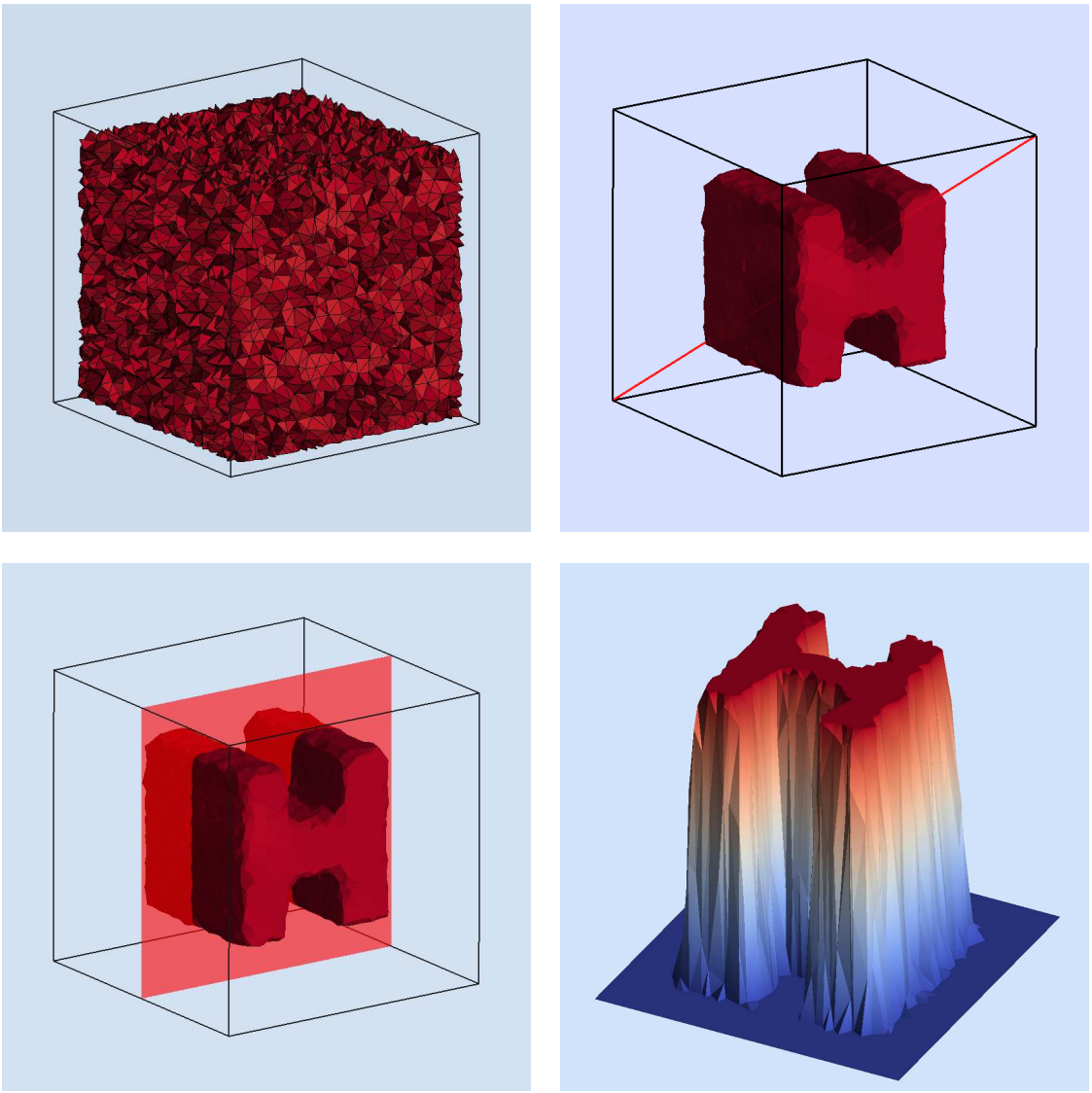

Figure 7. Initialization of the H-like shape rotation problem. Top left: interior view of the tetrahedral mesh. Top right: initialization of the H-like shape (isosurface $1/2$, rotation axis is the red line). Bottom left: cut plane $z = 1/2$. Bottom right: extruded initial values from the cut plane.

DOI: 10.1002/fld

We plot in Figure 8 the solution on the cut plane $z = 1/2$ for the unlimited $\mathbb{P}_3$ and $\mathbb{P}_5$ schemes and the MOOD-$\mathbb{P}_3$, MOOD-$\mathbb{P}_5$ methods. We notice that the unlimited schemes produce oscillations, depicted in green in the figure, whereas the MOOD method provides an oscillation-free solution even for polynomials of degree 5. It highlights the capacity of the $[\text{DMP} \rightarrow u2]$ detection process to correctly treat discontinuous shapes on genuinely unstructured 3D meshes.

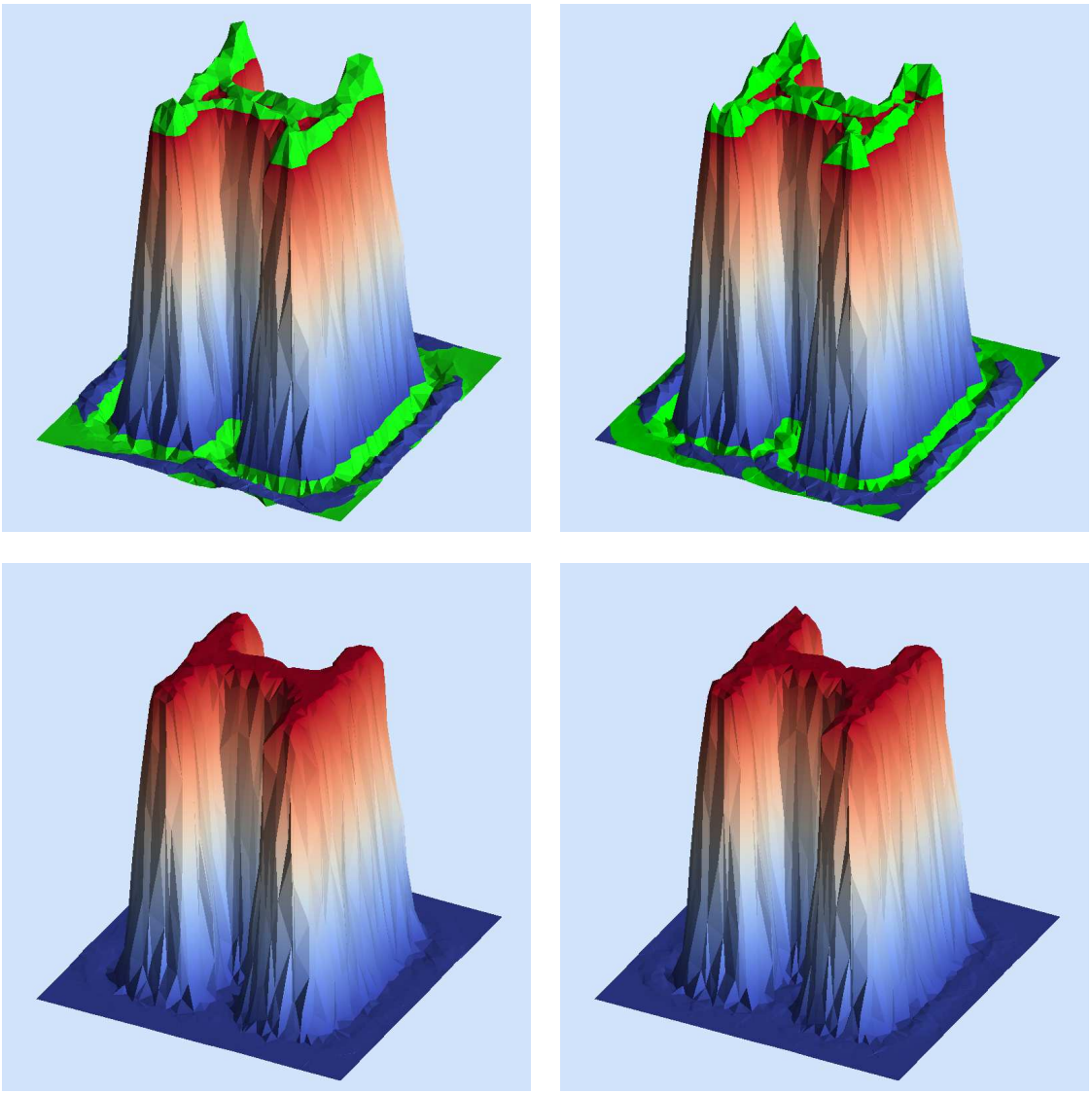

Figure 8. Results of the H-like shape rotation problem on the cut plane $z = 1/2$ for the unlimited $\mathbb{P}_3$ and $\mathbb{P}_5$ schemes (top line) and for the MOOD $\mathbb{P}_3$ and $\mathbb{P}_5$ methods (bottom line). The highlighted green cells correspond to values below 0 or above 1.

DOI: 10.1002/fld

4.2. *Euler system*

We now consider the three-dimensional hydrodynamics Euler system on unstructured meshes. The first test cases proposed in section 4.2.1 deal with the Sod and Lax shock tubes following the Ox axis (invariant with respect to the other directions). The simulations are carried out on a tetrahedral mesh to study the MOOD method capacity to handle simple waves. Section 4.2.2 is dedicated to the Shu-Osher and Blastwave problems approximated on pyramidal cells which respectively involve a complex oscillatory solution and strong interactions between simple waves along the Ox direction. We address in section 4.2.3 the effective numerical accuracy of the method with the isentropic vortex problem for which an exact smooth solution exists. Then in section 4.2.4, we assess the ability of the MOOD method to simulate complex realistic physics on a mesh of triangular and quadrangular prisms by carrying out the impact of a shock wave on a cylindrical cavity proposed in [11]. At last, we provide the MOOD method results for two genuinely three-dimensional test cases. First we compare the behavior and computational cost (CPU and memory storage) of the MOOD method with different degrees and detection processes by simulating the so-called explosion problem [33] using unstructured pyramidal meshes in section 4.2.5; Then in section 4.2.6, we consider the interaction of a shock wave with a quarter of cone on a mesh of 1.1 millions of tetrahedra with the 4^{th}-order MOOD method.

4.2.1. Sod and Lax shock tubes The original Sod [31] and Lax [24] problems concern one-dimensional Riemann shock tubes whose solutions consist of a left-moving rarefaction fan, a right-moving contact discontinuity and a right-moving shock wave. In the three-dimensional context, we reproduce the expansion following the Ox axis setting initial condition invariant in y, z and we prescribe reflecting boundary conditions on the cylinder sides. The domain is filled with an ideal gas with $\gamma = 1.4$ and the discontinuity is located in $x = 0.5$ at $t = 0$. The initial density/velocity/pressure values and final time t_{final} are given by

- Sod: $(\rho, u, p)_L = (1.0, 0.0, 1.0)$ and $(\rho, u, p)_R = (0.125, 0.0, 0.1)$, $t_{\text{final}} = 0.2$,
- Lax: $(\rho, u, p)_L = (0.445, 0.698, 3.528)$ and $(\rho, u, p)_R = (0.5, 0.0, 0.571)$, $t_{\text{final}} = 0.13$.

The computational domain we consider is a cylinder of unit length and radius $R = 0.025$ with $0x$ line as symmetry axis which is paved with 7517 unstructured tetrahedra as shown in figure 9.

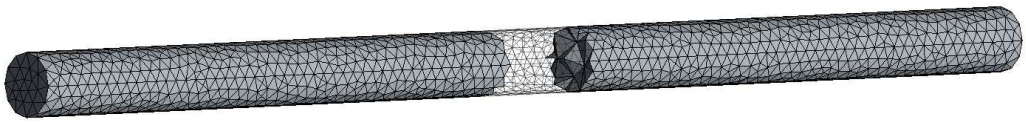

Figure 9. Mesh constituted of 7517 tetrahedra used for the Sod and Lax problems. Some cells are drawn non-opaque to see some interior tetrahedra.

We display in Figure 10 the numerical approximations of the density computed with the MOOD-$\mathbb{P}_3$ method using the [PAD$\rightarrow$DMP$\rightarrow u2$] detection process and the exact solution (red line). In order to provide a clear and relevant representation of the solution along the Ox axis, we slice the whole cylinder in 100 uniform cylinders (since the average characteristic length is 10^{-2}) and plot the average of the solution on each of them. As expected the MOOD-$\mathbb{P}_3$ method provides a very good approximation of the solution and maintains sharp discontinuities. In particular, we underline the very few numbers of points in the contact discontinuity.

4.2.2. Shu-Osher and Blastwave problems The Shu-Osher problem has been introduced in [29] to test the ability of a scheme to capture both small-scale smooth flow along with shock wave. The one-dimensional computational domain is $\Omega = [-5; 5]$ and the final time is $t_{\text{final}} = 1.8$. An initial x-directional shock wave located at $x = -0.4$ separates the domain into a left post-shock state

DOI: 10.1002/fld

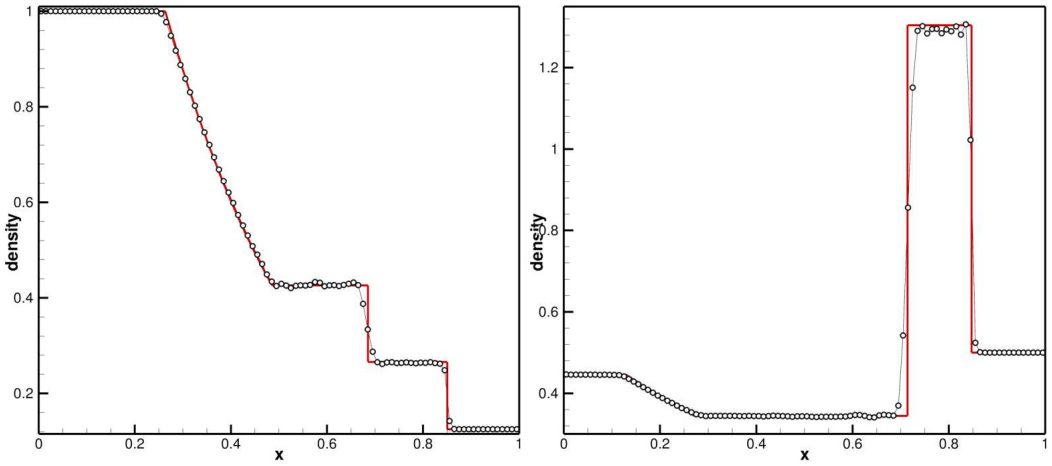

Figure 10. The MOOD-$\mathbb{P}_3$ density results are displayed for the Sod (left) and Lax (right) problems on tetrahedral mesh *vs* the exact solution (red line).

$(\rho, u, p)_L = (3.857143, 2.629369, 10.333333)$ and a right state $(\rho, u, p)_R = (1 + 0.2\sin(5x), 0, 1.0)$. We consider a perfect gas with $\gamma = 1.4$. Reflecting boundary conditions are used to preserve the invariance following axis Oy, Oz except from the left boundary condition which is an inflow one.

The Blastwave problem has been introduced by Collela and Woodward in [10] to test the performance of numerical schemes on problems involving strong and thin shock structures. The initial conditions consist of two parallel planar flow discontinuities on domain $\Omega = [0, 1]$ separated by the planes $x_1 = 0.1$ and $x_2 = 0.9$. The density is unity on the whole domain and the gas is assumed initially at rest. The pressure is given by $p_L = 1000$ on the left, $p_C = 0.01$ in the center and $p_R = 100$ on the right. Reflecting boundary conditions are prescribed and the final time is $t_{\text{final}} = 0.038$.

We consider a 21600 regular pyramids mesh (see Figure 12-top right for a pattern example) obtained from a $400 \times 3 \times 3$ regular hexahedral mesh for which each cell is split into six pyramids. The original hexahedral mesh is built by setting $\Delta x = \Delta y = \Delta z$ with $\Delta x = 0.075$ for Shu-Osher problem and $\Delta x = 0.0075$ for the Blastwave problem. Since there is no exact solution for both tests we have computed reference solutions using a first-order finite volume scheme with very fine meshes. As in the previous simulations, the solutions are plotted following the Ox direction considering an underlying 400 points uniform one-dimensional mesh and circles in Figure 11 represent the mean density on three-dimensional slices of thickness Δx.

Density approximations obtained with the MOOD-$\mathbb{P}_3$ are presented in Figure 11 and compared to the reference solution (red line). For the Shu-Osher problem (left) we report that the [PAD$\rightarrow$ DMP$\rightarrow u2$] detection criteria does not over-smooth the oscillatory solution and accurately capture the high-frequencies waves. On the other hand, for the Blastwave problem (right) we observe sharp contact discontinuities and shock waves are well-preserved. No spurious oscillations are generated and the central structure of the solution is very well approximated.

4.2.3. Isentropic vortex The isentropic vortex problem was initially introduced for the two-dimensional space [28, 38] to test the accuracy of numerical methods since the exact solution is smooth and has an analytical expression. We simply extend the original problem for the three-dimensional situation taking the two-dimensional solution invariant following Oz. Let us consider the computational domain $\Omega = [-5, 5] \times [-5, 5] \times [0, z_{max}]$ and an ambient flow characterized with $\rho_\infty = 1.0,\ u_\infty = 1.0,\ v_\infty = 1.0,\ w_\infty = 1.0,\ p_\infty = 1.0$, with a normalized ambient temperature

DOI: 10.1002/fld

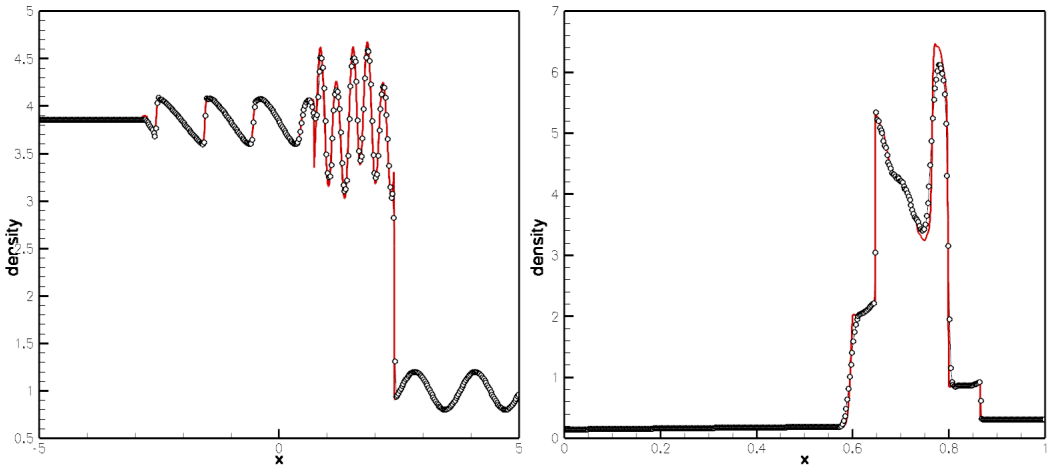

Figure 11. Results for the Shu-Osher (left) and Blastwave (right) problems on pyramids. MOOD-$\mathbb{P}_3$ density results are displayed on the left and right columns respectively *vs* the reference solution (red line).

$T^*_\infty = 1.0$ computed with the perfect gas equation of state and $\gamma = 1.4$.
A z-invariant vortex is centered on the axis line $\mathbf{x}_{\text{vortex}} = (x_{\text{vortex}}, y_{\text{vortex}}, z) = (0, 0, z)\ z \in \mathbb{R}$ and supplemented to the ambient gas at the initial time $t = 0$ with the following conditions $u = u_\infty + \delta u$, $v = v_\infty + \delta v$, $T^* = T^*_\infty + \delta T^*$ where

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

with $r = \sqrt{x'^2 + y'^2}$ and $x' = x - x_{\text{vortex}}, y' = y - y_{\text{vortex}}$. The vortex strength is given by $\beta = 5.0$ and the initial density follows relation

$$\rho = \rho_\infty \left(\frac{T^*}{T^*_\infty}\right)^{\frac{1}{\gamma-1}} = \left(1 - \frac{(\gamma-1)\beta}{8\gamma\pi^2} \exp\left(1-r^2\right)\right)^{\frac{1}{\gamma-1}}. \tag{14}$$

The domain is paved either with $N \times N \times 4$ hexahedra , $N = 20,\ 40,\ 60,\ 80,\ 120$ or with $N \times N \times 24$ pyramids (each hexahedron from the previous mesh is split into 6 pyramids, see Figure 12). To reduce the computational effort, only four cells are considered in the z-direction and z_{max} is taken such that $\Delta x = \Delta y = \Delta z$, that is to say $z_{max} = 4\Delta x = 40/N$. The minimal/maximal number of cells is $1600/57600$ hexahedra and $9600/153600$ pyramids. We prescribe periodic boundary conditions everywhere.

In Figure 12 we display the convergence curves for the L^1 and L^∞ errors on the density approximations for MOOD-$\mathbb{P}_2$, MOOD-$\mathbb{P}_3$, MOOD-$\mathbb{P}_5$ methods, while we provide in Table II the corresponding errors and convergence rates. We report effective orders corresponding to the expected optimal rates of convergence for both types of meshes and underline the MOOD method capacity to provide effective high-order of accuracy on a smooth but non-trivial solution for the three-dimensional Euler system.

DOI: 10.1002/fld

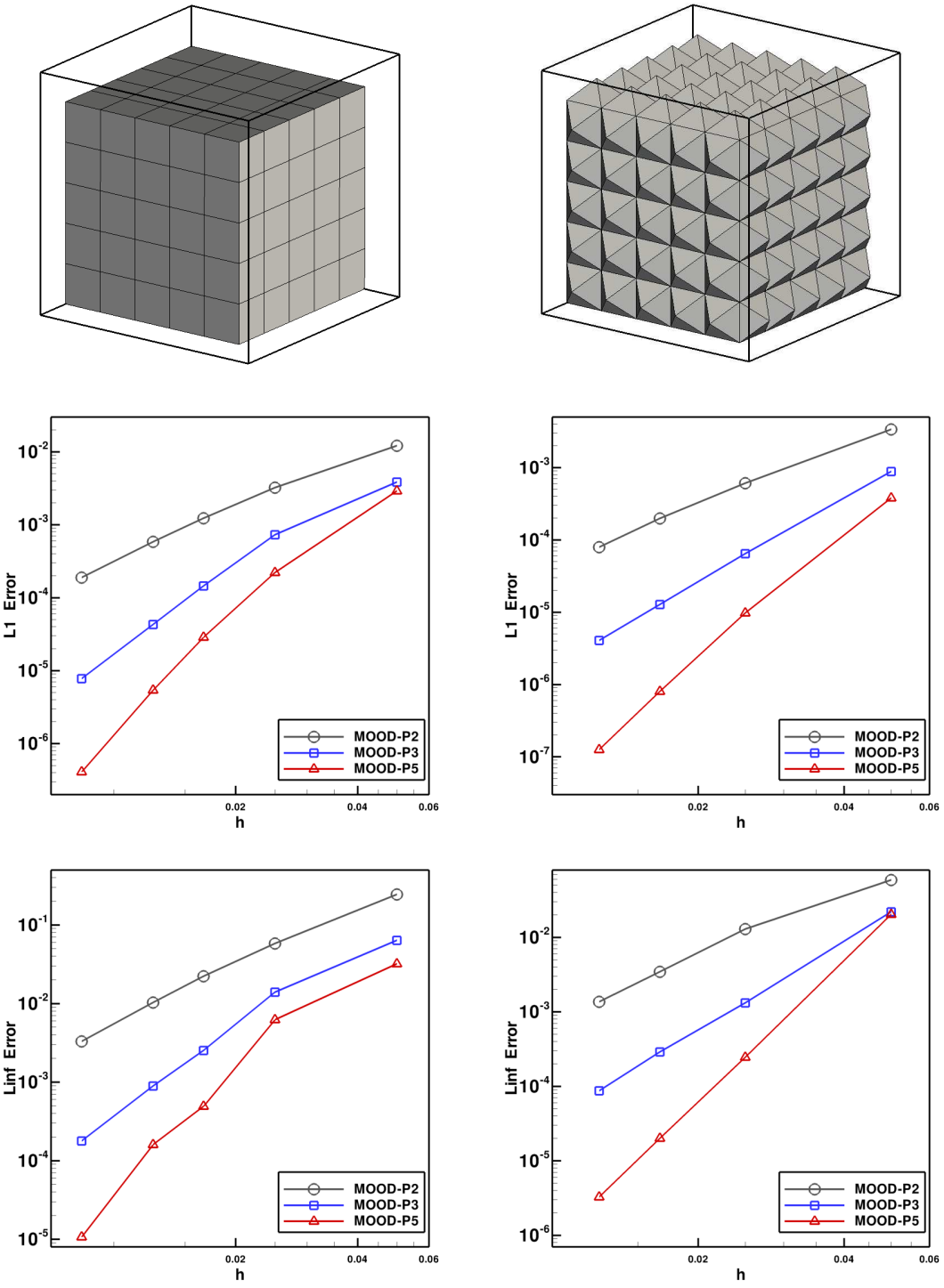

Figure 12. Isentropic vortex in motion: convergence curves for L^1 (middle) and L^∞ (bottom) errors for series of hexahedral (left) and pyramidal (right) meshes. Examples of such meshes are given on top line.

DOI: 10.1002/fld

MOOD on hexahedra							
Deg.	Cell nb	L^1 error		L^2 error		L^∞ error	
$\mathbb{P}_2$	$20 \times 20 \times 4$	1.2149125472e-2	—	3.1900165943e-2	—	2.4441600308e-1	—
	$40 \times 40 \times 4$	3.2245099895e-3	1.91	8.0410260168e-3	1.98	5.8071095773e-2	2.07
	$60 \times 60 \times 4$	1.2274201731e-3	2.38	3.0952199533e-3	2.35	2.2241472168e-2	2.37
	$80 \times 80 \times 4$	5.8449248920e-4	2.57	1.4891720330e-3	2.54	1.0333244490e-2	2.66
	$120 \times 120 \times 4$	1.8870676632e-4	2.78	4.8726522430e-4	2.75	3.2966795901e-3	2.82
Expected order			3		3		3
$\mathbb{P}_3$	$20 \times 20 \times 4$	3.8426301161e-3	—	9.2866634542e-3	—	6.3860302401e-2	—
	$40 \times 40 \times 4$	7.2909293814e-4	2.39	1.6463735019e-3	2.49	1.3928173243e-2	2.19
	$60 \times 60 \times 4$	1.4537313954e-4	3.97	3.4725838178e-4	3.84	2.5316808689e-3	4.20
	$80 \times 80 \times 4$	4.3014601762e-5	4.23	1.1403422006e-4	3.87	8.9234157884e-4	3.62
	$120 \times 120 \times 4$	7.7653485653e-6	4.22	2.0827671718e-5	4.19	1.7793186459e-4	3.98
Expected order			4		4		4
$\mathbb{P}_5$	$20 \times 20 \times 4$	2.8991068920e-3	—	4.8543664172e-3	—	3.2038381504e-2	—
	$40 \times 50 \times 4$	2.2151699683e-4	3.71	5.5851141683e-4	3.12	6.2194475329e-3	2.36
	$60 \times 60 \times 4$	2.8610132561e-5	5.04	7.5286576723e-5	4.94	4.9068468256e-4	6.26
	$80 \times 80 \times 4$	5.4168534310e-6	5.78	1.5519206048e-5	5.49	1.5955744462e-4	3.90
	$120 \times 120 \times 4$	4.0840597698e-7	6.38	1.1795674119e-6	6.36	1.0709587465e-5	6.66
Expected order			6		6		6
MOOD on pyramids							
Deg.	Cell nb	L^1 error		L^2 error		L^∞ error	
$\mathbb{P}_2$	$20 \times 20 \times 24$	3.3660908651e-3	—	8.2020368268e-3	—	5.8966752971e-2	—
	$40 \times 40 \times 24$	6.0800306087e-4	2.47	1.4780372369e-3	2.47	1.2917288297e-2	2.19
	$60 \times 60 \times 24$	1.9831385885e-4	2.76	5.0256415975e-4	2.66	3.4489695638e-3	3.25
	$80 \times 80 \times 24$	7.9096059248e-5	3.19	2.0028509695e-4	3.19	1.3642153624e-3	3.22
Expected order			3		3		3
$\mathbb{P}_3$	$20 \times 20 \times 24$	8.8005733635e-4	—	2.0405839361e-3	—	2.2060839273e-2	—
	$40 \times 40 \times 24$	6.4460987694e-5	3.77	1.4763173293e-4	3.78	1.3204082077e-3	4.06
	$60 \times 60 \times 24$	1.2809782719e-5	3.98	2.9223354775e-5	3.99	2.8960192576e-4	3.74
	$80 \times 80 \times 24$	4.0713141263e-6	3.98	9.3121356054e-6	3.97	8.6899534957e-5	4.18
Expected order			4		4		4
$\mathbb{P}_5$	$20 \times 20 \times 24$	3.7944742185e-4	—	9.8016940506e-4	—	2.0273963181e-2	—
	$40 \times 40 \times 24$	9.7451977113e-6	5.28	2.4664732108e-5	5.31	2.4540502338e-4	6.36
	$60 \times 60 \times 24$	8.0304771455e-7	6.15	2.0569058735e-6	6.12	2.0022941712e-5	6.18
	$80 \times 80 \times 24$	1.2520658320e-7	6.46	3.1436119294e-7	6.53	3.2667224251e-6	6.30
Expected order			6		6		6

Table II. L^1, L^2 and L^∞ errors and convergence rates for the isentropic vortex problem with the MOOD-$\mathbb{P}_2$, MOOD-$\mathbb{P}_3$ and MOOD-$\mathbb{P}_5$ methods. Top lines: hexahedral meshes. Bottom lines: pyramidal meshes.

DOI: 10.1002/fld

4.2.4. Impact of a shock wave on a cylindrical cavity Based on the experiment proposed in [30], we have introduced this test case in [11] for the two-dimensional case. We here extend it to 3D by invariance along the z-direction. It consists in a planar shock wave which impacts a cylindrical cavity creating complex structures and instabilities. The original purpose of this stringent numerical test is to prove the ability of the MOOD method to capture physics in realistic conditions. In this paper it moreover assesses the capacity of the MOOD method to deal with mixed triangular and quadrangular prisms since the mesh is obtained by extrusion (only two layers) along the Oz axis of a 2D mesh containing 101127 cells (triangles and quadrangles). We moreover point out that important differences between cell sizes are present in the domain, since the largest characteristic length is 0.008 and the smallest one is 0.00015. At last, we run the simulation on the lower half part of the domain but plot a full domain using a symmetry argument. Details of the mesh are provided in Figure 13.

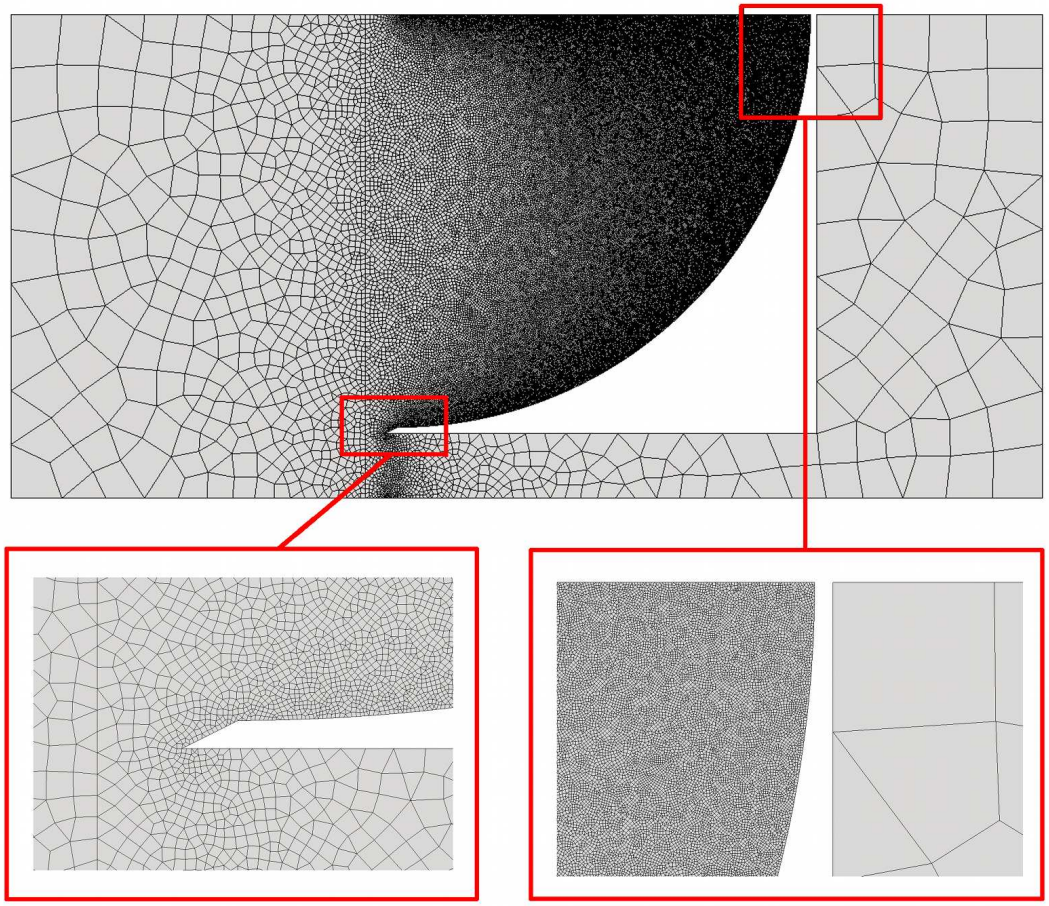

Figure 13. Impact of a shock on a cylindrical cavity: details of the mesh containing 202254 triangular and quadrangular prisms.

The detailed configuration and boundary conditions are provided in [11], and we recall that we consider the case of a nominal incident shock Mach number of 1.33 in ambient air (with $\gamma = 1.4$) at 0.95 bar pressure and that the variables initialization consists in the pre-shock values $(\rho, u, v, w, p) = (1.1175, 0.0, 0.0, 0.0, 95000.0)$ and the post-shock ones $(\rho, u, v, w, p) = (1.7522, 166.3435, 0.0, 0.0, 180219.75)$ leading to conditions of [30] at temperature $T = 296.15K$.

DOI: 10.1002/fld

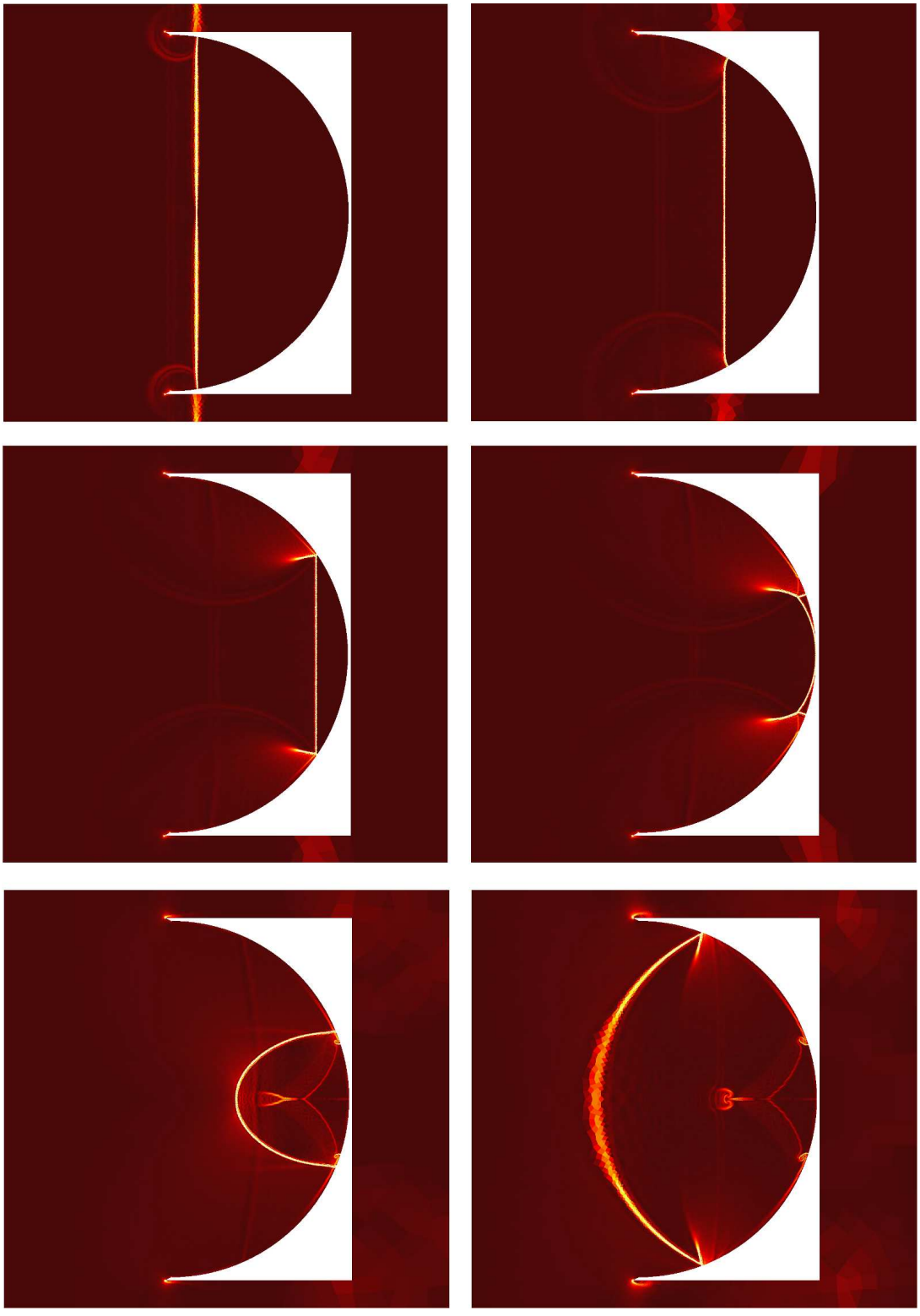

Figure 14. Impact of a shock on a cylindrical cavity: magnitude of the density gradient at different times from left to right and top to bottom.

DOI: 10.1002/fld

In Figure 14, we plot the magnitude of the density gradient computed with the MOOD $\mathbb{P}_2$ method equipped with the $[\text{PAD} \rightarrow \text{DMP} \rightarrow u2]$ detection process at different times of the simulation in order to give an overview of the physical phenomena. We emphasize that the instabilities along the cylindrical wall are very well captured. Finally in Figure 15, we provide a zoom of the final solution on the created instabilities which perfectly match the experimental results of [30].

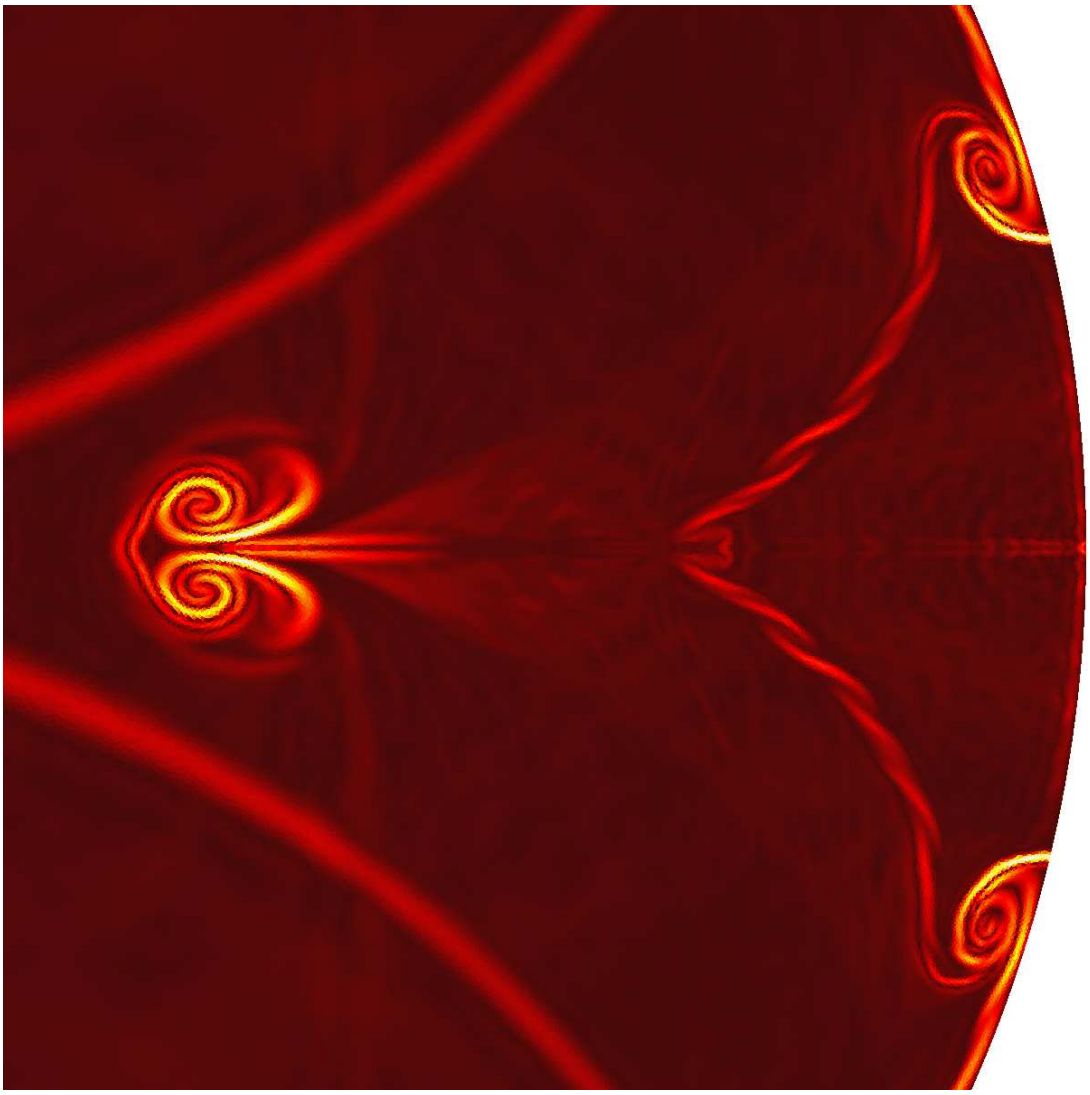

Figure 15. Impact of a shock on a cylindrical cavity: Zoom on the created instabilities at final time.

4.2.5. The explosion problem We consider the so-called explosion problem [33] given by a gas initially at rest in the unit cube where a quarter of the ball of radius $r_c = 0.4$ centered at the origin has a density $\rho_b = 1.0$, a pressure $p_b = 1.0$ whereas the exterior is characterized by $\rho_e = 0.125$, $p_e = 0.1$. The domain is partitioned into 20^3 hexahedral cells for which each hexahedron is split into 6 pyramids leading to a mesh of 48000 pyramids. Simulations are carried out till the final time $t_{\text{final}} = 0.25$. A reference solution has been computed with a two-dimensional cylindrical staggered numerical Lagrangian scheme [25].

DOI: 10.1002/fld

We report in Figure 16 the density approximations in function of the radius for a classical MUSCL scheme [26], the MOOD-$\mathbb{P}_2$, MOOD-$\mathbb{P}_3$, MOOD-$\mathbb{P}_5$ methods equipped with the [PAD$\rightarrow$ DMP$\rightarrow u2$] detection and the reference solution. Note that we use the same type of representation than for the previous test cases by slicing the radius in 100 uniform cells.

The solution shape is well reproduced by all methods and the higher the polynomial degree is the sharper the contact discontinuity and the shock wave are. In a three-dimensional context with discontinuous solutions, the MOOD-$\mathbb{P}_3$ method seems to be the right balance between accuracy and cost. The slight improvement gained by the MOOD-$\mathbb{P}_5$ compared to MOOD-$\mathbb{P}_3$ may not justify the computational over-cost (see further). We also notice that the head of the rarefaction wave is badly resolved by the MUSCL method whereas the MOOD-$\mathbb{P}_2$ and especially the MOOD-$\mathbb{P}_3$ method give accurate approximations.

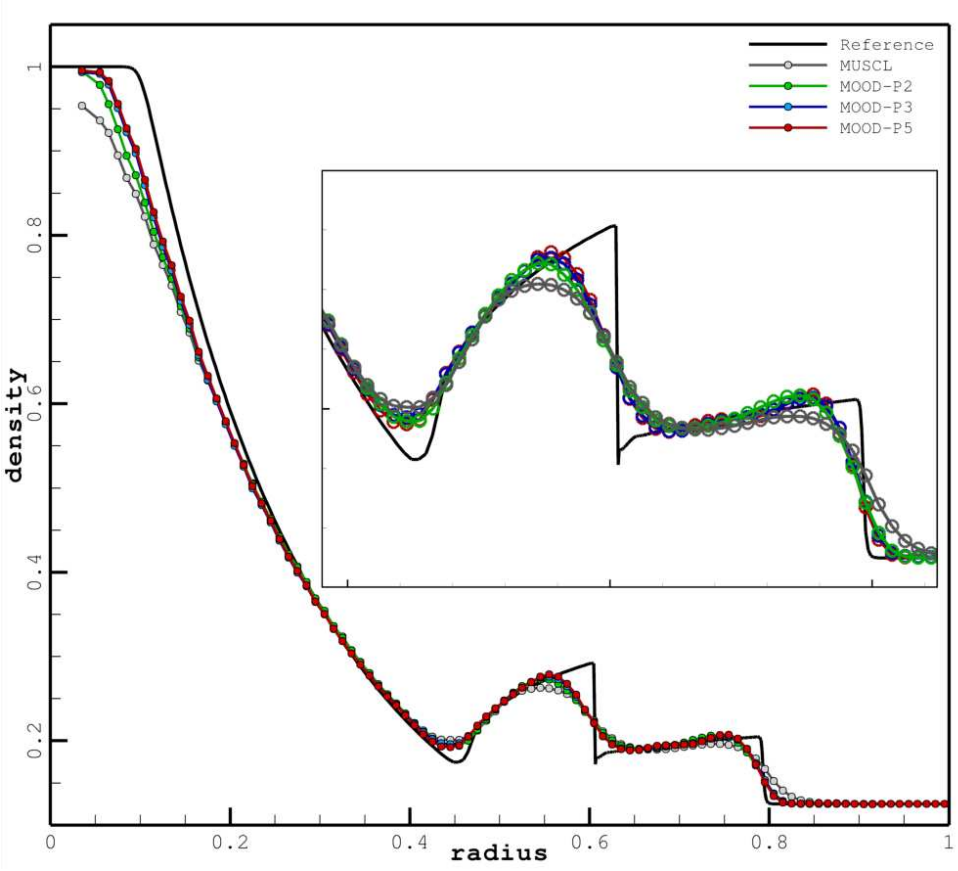

Figure 16. Density results for the explosion problem in 3D. Comparison between a classical MUSCL method and the MOOD-$\mathbb{P}_2$, MOOD-$\mathbb{P}_3$ and MOOD-$\mathbb{P}_5$ methods with [PAD$\rightarrow$DMP$\rightarrow u2$] detection process on tetrahedral mesh. The straight line corresponds to the reference solution.

To compare the different detection strategies, we present in Figure 17 the final solutions obtained by the MOOD method with the PAD alone and the [PAD$\rightarrow$DMP$\rightarrow u2$] detection processes using $\mathbb{P}_5$ polynomial reconstructions. Note that contrary to previous figures, we plot the density values for all cells by associating them with the radius corresponding to the cell centroid. As expected, the PAD detection process does not damp spurious oscillations close to the shock wave (see the zoom panel) and extra oscillations are also visible on the head of the rarefaction. We recall that the numerical approximation using the PAD detection process is the most accurate one on smooth solutions since only the physical admissibility of the solution is required so that few numerical diffusion is produced. On the opposite, the [PAD$\rightarrow$DMP$\rightarrow u2$] detection process damps the oscillations close

DOI: 10.1002/fld

to the shock and to the head of rarefaction but also maintain a very good accuracy with a slight non monotonic behavior.

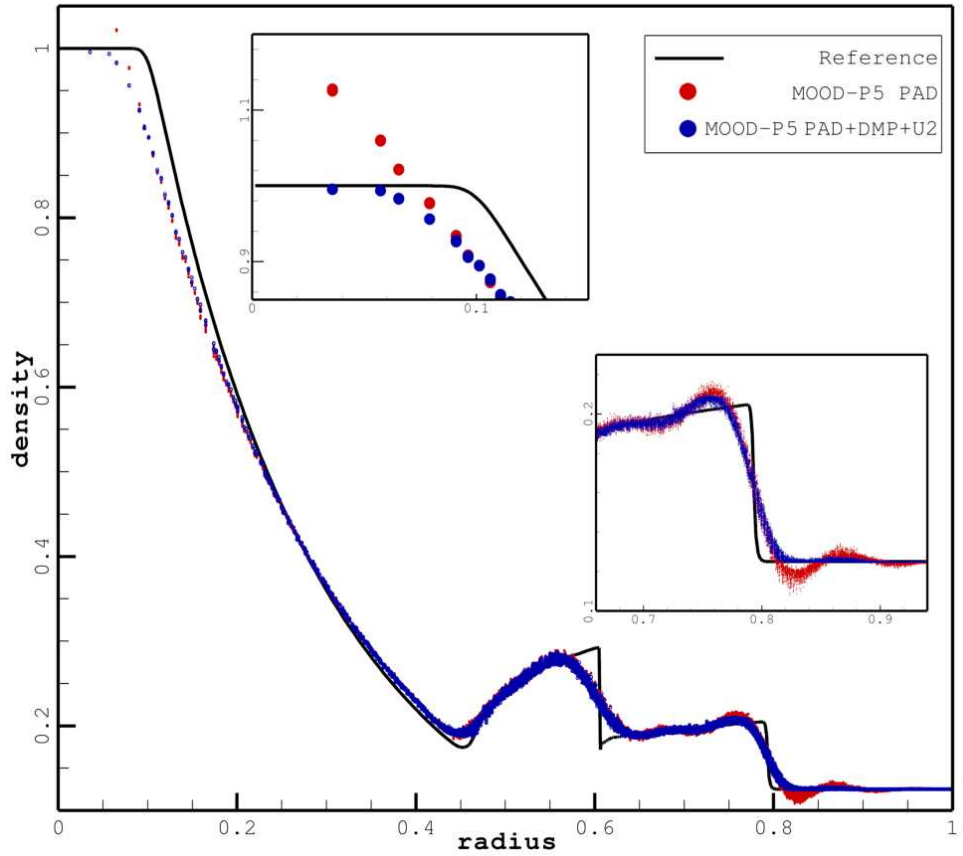

Figure 17. Density results for the explosion problem in 3D obtained with the MOOD-$\mathbb{P}_5$ method. Comparison between the PAD alone and [PAD$\rightarrow$DMP$\rightarrow u2$] detection process. The straight line corresponds to the reference solution and the symbols represent mean values of all cells.

Finally we provide in Tables III the computational cost of the MOOD method for this test case in this particular configuration when running on a single core of the three following machines (using -O3 flag for gfortran compiler):

M1: server with two Intel Xeon E5335 (4 cores) @ 2.00Ghz, 8MB of L2 Cache, 16GB of RAM
M2: laptop with Intel Core2Duo P7550 (2 cores) @ 2.26GHz, 3MB of L2 Cache, 8GB of RAM
M3: desktop with Intel Core i5 2500 (4 cores) @ 3.30GHz, 6MB of L2 Cache, 8GB of RAM

Note that the same three machines have been used in [11] to assess the computational cost of the MOOD method for two-dimensional geometries.

MOOD with [**PAD**$\rightarrow$**DMP**$\rightarrow u2$]	**Machine 1** Intel Xeon E5335 @ 2.00Ghz	**Machine 2** Intel Core2Duo P7550 @ 2.26GHz	**Machine 3** Intel Core i5 2500 @ 3.30GHz	**Memory storage**
MOOD-$\mathbb{P}_2$	66μs/it./cell	57μs/it./cell	30μs/it./cell	0.4 GB
MOOD-$\mathbb{P}_3$	163μs/it./cell	136μs/it./cell	69μs/it./cell	0.8 GB
MOOD-$\mathbb{P}_5$	439μs/it./cell	385μs/it./cell	185μs/it./cell	3.0 GB

Table III. CPU time in microseconds per iteration per cell and memory storage in Gigabytes for the MOOD-$\mathbb{P}_k$ methods ($k = 2, 3, 5$) with the [PAD$\rightarrow$DMP$\rightarrow u2$] detection process on three different computers.

DOI: 10.1002/fld

We first observe that the memory storage doubles when the polynomial degree is increased by one: 0.4 for $\mathbb{P}_2$, 0.8 for $\mathbb{P}_3$, 1.6 for $\mathbb{P}_4$ (not presented in the table) and 3.0 for $\mathbb{P}_5$. Notice that the memory consumption is very low since only two reconstruction pseudoinverse matrices (for $\mathbb{P}_2$ and $\mathbb{P}_{\mathsf{d}_{max}}$) per cell are effectively stored. The CPU cost increases by a factor about 2.4 from $\mathbb{P}_2$ to $\mathbb{P}_3$ and about 2.7 from $\mathbb{P}_3$ to $\mathbb{P}_5$.

By extrapolation of these results we estimate the cost of the MOOD method for larger meshes. As instance for one million cells mesh and 1000 time steps the method cost should be:

- MOOD-$\mathbb{P}_2$ is 66000 seconds on M1, that is to say $\sim$ 18 hours ($\sim$ 16 hours on M2 and $\sim$ 8.3 hours on M3) with about 8 Gb of memory storage,
- MOOD-$\mathbb{P}_3$ is 163000 seconds, $\sim$ 2 days on M1 ($\sim$ 1.5 day and $\sim$ 19 hours on M2 and M3) with about 16 Gb of memory storage,
- MOOD-$\mathbb{P}_5$ is 439000 seconds, $\sim$ 5 days on M1 ($\sim$ 4.5 days and $\sim$ 2.1 days on M2 and M3) with about 62 Gb of memory storage.

Consequently simulations with nowadays sequential computers with a one million cells mesh (assuming one thousand time steps) can be obtained for about one day of computation with MOOD-$\mathbb{P}_3$ method. The MOOD method is thus a very competitive very high-order finite volume method, and these results shall be improved by an efficient parallelization.

4.2.6. Interaction of a shock wave with a quarter of cone To conclude the numerical tests section, we run the test case named *interaction of a shock wave with a quarter of cone* with the 4^{th}-order MOOD-$\mathbb{P}_3$ method equipped with the [PAD$\rightarrow$DMP$\rightarrow u2$] detection process. This 3D extension of the so-called *interaction of a shock wave with a wedge* has been proposed in [14] as instance.

The domain consists in a quarter on cylinder of radius $R = 2.25$ centered on the Ox axis which covers the interval $[-1.1; 3.0]$ in the x-direction. Note that three modifications have been made in comparison to [14] in order to reduce the computational cost: the test is run on a quarter of cylinder instead of a half one, the initial interface is placed at $x = -0.2$ instead of $x = -1.0$ and the domain covers in the x-direction the interval $[-1.1; 3.0]$ instead of $[-1.5; 3.0]$. Finally the mesh obtained by the free mesher Gmsh contains 1161854 tetrahedra in three refinement zones and exactly matches the initial interface, see top of Figure 19.

We recall that the circular cone under consideration is such that its length is 1, its tip and foot radii are 0.02 and 0.5 respectively while its tip is placed at the origin. Moreover wall boundary conditions are prescribed everywhere except from the top and bottom of the quarter of cylinder where the exact solution according to the Rankine-Hugoniot conditions is imposed. At last the initial pre- and post shock conditions are given by $(\rho, u, v, w, p) = (2.122, 0.0, 0.0, 0.0, 1.805)$ and $(\rho, u, v, w, p) = (1.4, 0.0, 0.0, 0.0, 1.0)$ respectively with $\gamma = 1.4$ and the final time is chosen such that it corresponds to the final time of [14].

In Figure 18, we propose numerical Schlieren-type images on the solution in the $Ox - Oy$ and $Ox - Oz$ planes. We remark that the symmetry is very well conserved since both images are almost identical and that all waves that are present in results of [14] are also resolved here although much less cells (more than 3.5 times less) are considered. This proves that the MOOD method performs very well on 3D unstructured meshes. Finally on bottom of Figure 19, we provide a 3D view for which isosurfaces have been chosen to represent the principal waves in the whole domain. It is thus clear that the method properly reproduces the cylindrical symmetry even on this fully unstructured 3D tetrahedral mesh.

DOI: 10.1002/fld

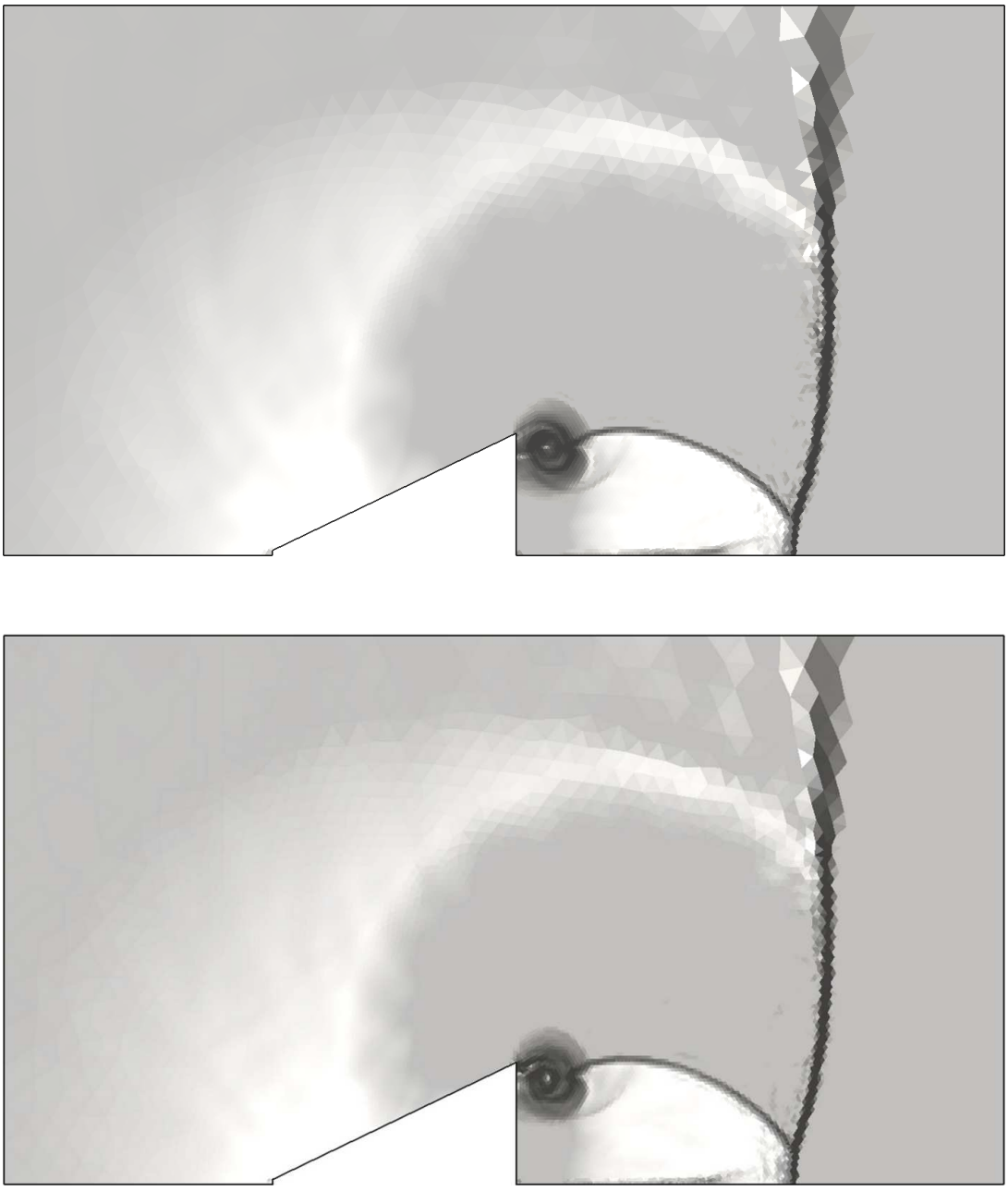

Figure 18. Interaction of a shock wave with a half cone: on top, numerical Schlieren-type image on the $Ox - Oy$ plane; on bottom, numerical Schlieren-type image on the $Ox - Oz$ plane.

DOI: 10.1002/fld

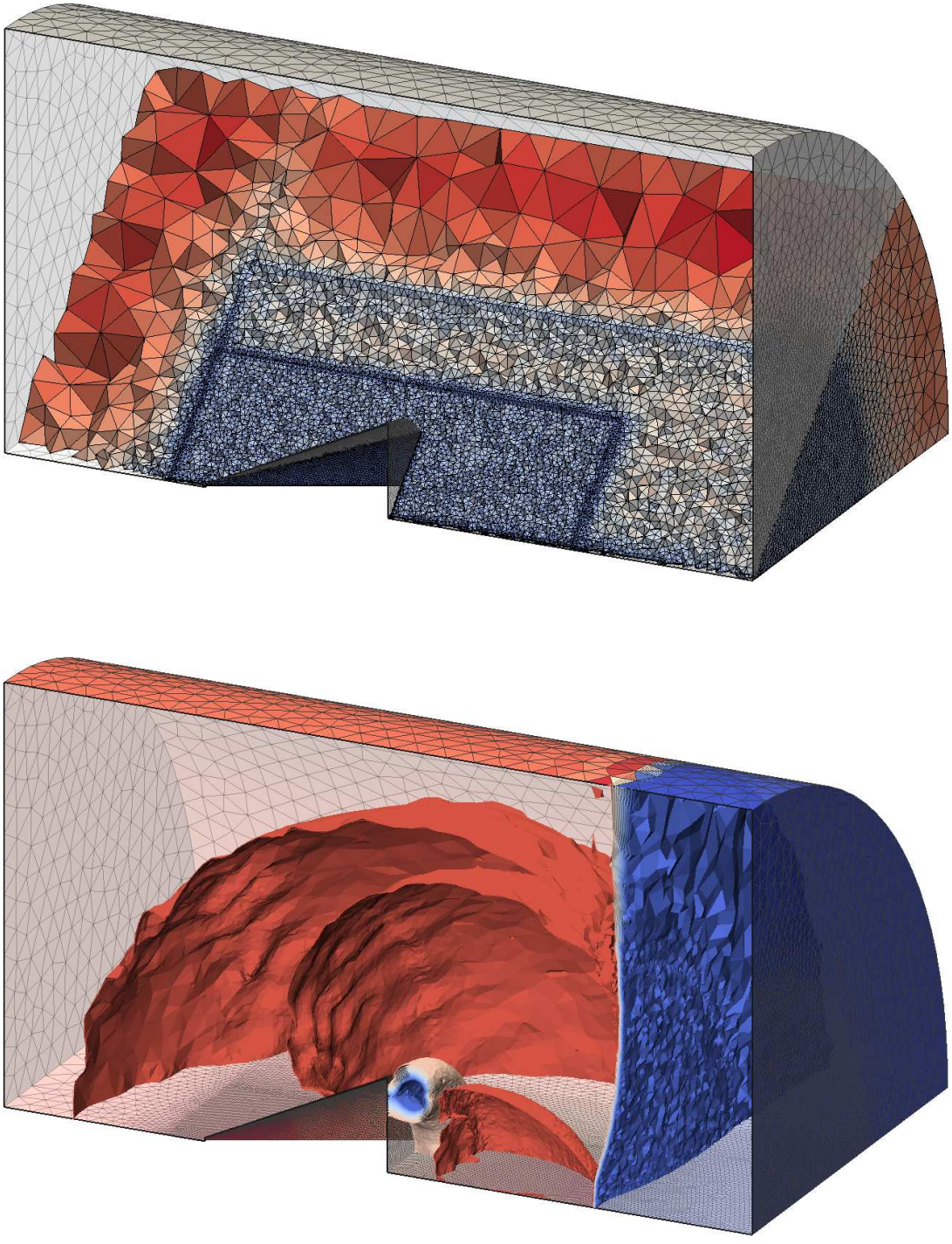

Figure 19. Interaction of a shock wave with a half cone: on top, view of the interior of the tetrahedral mesh with the different zones of refinement; on bottom, isosurfaces corresponding to the principal waves.

DOI: 10.1002/fld

5. CONCLUSION

In this paper we have proposed the three-dimensional extension of the so-called MOOD method [5, 11]. The *Multi-dimensional Optimal Order Detection* expression refers to an original way of determining the optimal local polynomial degree to be used in the reconstruction step of a classical high-order unlimited scheme. To each cell corresponds a polynomial reconstruction for which we *a posteriori* determine the degree according to given criteria (positivity as instance) against which we test each candidate solution. The detection criteria is based on a relaxed version of the discrete maximum principle (DMP) associated with a so-called $u2$ detection procedure which analyses the numerical curvatures in the neighborhood of a DMP violating cell and determine if the underlying function is regular or not. In the latter case the polynomial degree in the associated cell is decremented and the solution is locally recomputed. We have detailed the numerical method for three-dimensional unstructured meshes and improved the detection criteria both for the advection equation and the Euler system. Moreover some optimizations for the three dimensional case have been provided to significantly improve the efficiency of the method.

The MOOD method has been implemented on several kinds of unstructured meshes with $\mathbb{P}_k$ polynomial reconstructions (k varying from 1 to 5). We have provided some sanity checks with simple configurations and performed more advanced full three-dimensional tests to assess the ability of the MOOD method to accurately capture waves on real unstructured meshes. For the scalar convection equation with a regular initial shape the method gives an effective high-order of accuracy corresponding to the optimal one and we have shown that spurious oscillations are damped when discontinuous profiles are convected. The results for unidirectional problems for the Euler system with three-dimensional unstructured meshes show that small scaled structures are captured while shock waves are resolved within few cells. For the isentropic vortex test case extended to the three-dimensional context with non-trivial exact solution, effective high-orders of accuracy are measured and optimal orders are reported for $\mathbb{P}_2, \mathbb{P}_3$ and $\mathbb{P}_5$ polynomials. We prove that the MOOD method is able to capture the realistic physics of the impact of a shock wave on a cylindrical cavity on a non trivial mesh made of a mix of triangular and quadrangular prisms. At last, the three-dimensional explosion problem has been carried out to show the improvement gained with the use of high-order MOOD methods and the slight numerical diffusion generated by the $u2$ detection process which enables to prevent spurious numerical oscillations from appearing. We have also provided the solution computed with the PAD detection process alone to support the intrinsic positivity-preserving property of the MOOD method and measures of the CPU cost to underline that the MOOD method is effective on nowadays personal computers. Finally the interaction of a shock wave on a quarter of cone with the 4^{th}-order MOOD method proves that the MOOD method provides a very good reproduction of the physics on a unstructured non-regular 3D mesh of 1.1 millions of tetrahedra.

In a near future, we plan to adapt the MOOD within an ADER technique to avoid the multiple time steps of the Runge-Kutta approach and overcome the third-order accuracy restriction. Furthermore although the MOOD method significantly reduces the necessary computational resources (CPU and memory storage), a parallelized version is of crucial importance to treat huge size simulations. Finally the application of the MOOD method to more complex physics (multi-material, multi-phase, etc.) is also an important challenge that has to be tackled.

REFERENCES

1. R. Abgrall On Essentially Non-oscillatory Schemes on Unstructured Meshes: Analysis and Implementation, J. Comput. Phys. 114 (1994) 45–58.
2. R. Abgrall, Essentially non-oscillatory Residual Distribution schemes for hyperbolic problems, J. Comput. Phys. 214 (2006) 773–808.

DOI: 10.1002/fld

3. T. Buffard, S. Clain, Monoslope and Multislope MUSCL Methods for unstructured meshes, J. Comput. Phys. 229 (2010) 3745–3776.
4. S. Clain, V. Clauzon, L^∞ stability of the MUSCL methods, Numer. Math. 116 (2010) 31–64.
5. S. Clain, S. Diot, R. Loubère, A high-order finite volume method for hyperbolic systems: Multi-dimensional Optimal Order Detection (MOOD), J. Comput. Phys. 230 (2011) 4028–4050.
6. B. Cockburn and C. W. Shu, TVB Runge-Kutta local projection discontinuous Galerkin finite element method for scalar conservation laws II: General framework, Math. Comp. 52 (1989), 411–435.
7. B. Cockburn, S. Y. Lin and C. W. Shu, TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws III: One dimensional systems, J. Comput. Phys. 84 (1989) 90–113.
8. B. Cockburn, S. Hou and C. W. Shu, TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws IV: The multidimensional case, Math. Comp. 54 (1990) 545–581.
9. B. Cockburn, C.-W. Shu, The Runge-Kutta Discontinuous Galerkin Method for Conservation Laws V: Multidimensional Systems J. Comput. Phys. 141 (1998) 199–224.
10. P. Colella and P.R. Woodward, The Piecewise Parabolic Method (PPM) for Gas-Dynamical Simulations, J. Comput. Phys. 54 (1984) 174–201.
11. S. Diot, S. Clain, R. Loubère, Improved Detection Criteria for the Multi-dimensional Optimal Order Detection (MOOD) on unstructured meshes with very high-order polynomials, Comput. & Fluids 64 (2012) 43–63.
12. M. Dumbser, M. Castro, C. Parés, E. F. Toro, ADER schemes on unstructured meshes for nonconservative hyperbolic systems: Applications to geophysical flows, Comput. & Fluids 38 (2009) 1731–1748.
13. M. Dumbser and M. Käser. Arbitrary High Order Non-Oscillatory Finite Volume Schemes on Unstructured Meshes for Linear Hyperbolic Systems, J. Comput. Phys. 221 (2007) 693–723.
14. M. Dumbser, M. Käser, V. A. Titarev and E. F. Toro. Quadrature-Free Non-Oscillatory Finite Volume Schemes on Unstructured Meshes for Nonlinear Hyperbolic Systems, J. Comput. Phys. 226 (2007) 204–243.
15. A. Harten, S. Osher, Uniformly highorder accurate nonoscillatory schemes I, SIAM J. Num. Anal. 24 (1987) 279–309.
16. A. Harten, B. Engquist, S. Osher, S. Chakravarthy, Uniformly highorder accurate nonoscillatory schemes III, J. Comput. Phys. 71 (1987) 279-309.
17. C. Hu, C.W. Shu, Weighted essentially non-oscillatory schemes on triangular meshes, J. Comput. Phys. 150 (1999) 97–127.
18. M. E. Hubbard, Multidimensional slope limiters for MUSCL-type finite volume schemes on unstructured grids, J. Comput. Phys. 155 (1999) 54–74.
19. G.-S. Jiang, C.-W. Shu, Efficient implementation of weighted ENO schemes, J. Comput. Phys. 126 (1996) 202–228.
20. G.-S. Jiang, E. Tadmor, Non-oscillatory central schemes for multidimensional hyperbolic conservative laws, SIAM J. Sci. Comput. 19 (1998) 1892–1917.
21. V. P. Kolgan, Application of the minimum-derivative principle in the construction of finite-difference schemes for numerical analysis of discontinuous solutions in gas dynamics, Transactions of the Central Aerohydrodynamics Institute 3 (1972) 68–77, in Russian.
22. V. P. Kolgan, Finite-difference schemes for computation of three dimensional solutions of gas dynamics and calculation of a flow over a body under an angle of attack, Transactions of the Central Aerohydrodynamics Institute 6 (1975) 1–6, in Russian.
23. V. P. Kolgan, Application of the principle of minimizing the derivative to the construction of finite-difference schemes fr computing discontinuous solutions of gas dynamics, J. Comput. Phys. 230 (2010) 2384–2390.
24. P. D. Lax, Weak solutions of nonlinear hyperbolic equations and their numerical computation, Comm. Pure and Appl. Math., (1954), 7, 159-193.
25. R. Loubère, P.-H. Maire, P. Vachal, Staggered Lagrangian discretization based on cell-centered Riemann solver and associated hydrodynamics scheme, Communication in Computational Physics, 2011, Vol. 10 (2011), Issue 4, pp. 940-978.
26. J. S. Park,S.-H. Yoon, C. Kim, Multi-dimensional limiting process for hyperbolic conservation laws on unstructured grids, J. Comput. Phys. 229 (2010) 788–812.
27. C.-W. Shu, High-order Weighted Essentially Non-Oscillatory schemes for convection dominated problems, SIAM Review 51 (2009) 82–126.
28. C.-W. Shu, Essentially non-oscillatory and weighted essentially non-oscillatory schemes for hyperin Advanced Numerical Approximation of Nonlinear Hyperbolic Equations, B. Cockburn, C. Johnson, C.-W. Shu and E. Tadmor (Editor: A. Quarteroni), Lecture Notes in Mathematics 1697 Springer (1998) 325–432.
29. C.-W. Shu, S. Osher, Efficient implementation of essentially non-oscillatory shock-capturing scheme, J. Comput. Phys. 77 (1988) 439–471.
30. B.W. Skews, H. Kleine, Flow features resulting from shock wave impact on a cylindrical cavity, J. Fluid. Mech. 580 (2007) 481–493.
31. G. A. Sod, A survey of several finite difference methods for systems of non-linear hyperbolic conservation laws, J. Comput. Phys. 27 (1978) 1–31.
32. V.A. Titarev, E. F. Toro, ADER schemes for three-dimensional non-linear hyperbolic systems, J. Comput. Phys. 204 (2005) 715–736.
33. E. F. Toro, Riemann Solvers and Numerical Methods for Fluid Dynamics, 3^{rd} revision, Springer-Verlag Berlin and Heidelberg GmbH & Co. K (2009).
34. E. F. Toro, A. Hidalgo, ADER finite volume schemes for nonlinear reaction-diffusion equations, Applied Numerical Mathematics 59 (2009) 73–100.
35. B. Van Leer, Towards the ultimate conservative difference scheme I. The quest of monotonicity, Proceedings of the Third International Conference on Numerical Methods in Fluid Mechanics, Lecture Notes in Physics 18 (1973) 163–168.
36. B. Van Leer, Towards the ultimate conservative difference scheme II. Monotonicity and conservation combined in a second-order scheme, J. Comput. Phys. 14 (1974) 361–370.

DOI: 10.1002/fld

37. P. Woodward, P. Colella, The numerical simulation of two-dimensional fluid flow with strong shocks, J. Comput. Phys. 54 (1984) 115–173.
38. H. C. Yee, M. Vinokur, M. J. Djomehri, Entropy Splitting and Numerical Dissipation, J. Comput. Phys. 162 (2000) 33–81.
39. Y.-T. Zhang and C.-W. Shu Third-order WENO scheme on three dimensional tetrahedral meshes, Com. Comput. Phys. 5 (2009) 836–848.
40. X. Zhang and C.-W. Shu Maximum-principle-satisfying and positivity-preserving high-order schemes for conservation laws: survey and new developments, Proc. R. Soc. A 467 (2011) 2752-2776.

DOI: 10.1002/fld