Journal of Computational Physics: X 4 (2019) 100037

Contents lists available at ScienceDirect

Journal of Computational Physics: X

www.elsevier.com/locate/jcpx

Stencil selection algorithms for WENO schemes on unstructured meshes

Panagiotis Tsoutsanis

Centre for Computational Engineering Sciences, Cranfield University, Cranfield MK43 0AL, United Kingdom

ARTICLE INFO

Article history:
Received 22 November 2018
Received in revised form 30 May 2019
Accepted 18 July 2019
Available online 30 August 2019

Dataset link:
https://doi.org/10.17862/cranfield.rd.8983772.v1

Keywords:
WENO
Stencils
Finite-volume
Least-squares

ABSTRACT

In this paper, a family of stencil selection algorithms is presented for WENO schemes on unstructured meshes. The associated freedom of stencil selection for unstructured meshes, in the context of WENO schemes present a plethora of various stencil selection algorithms. The particular focus of this paper is to assess the performance of various stencil selection algorithm, investigate the parameters that dictate their robustness, accuracy and computational efficiency. Ultimately, efficient and robust stencils are pursued that can provide significant savings in computational performance, while retaining the non-oscillatory character of WENO schemes. This is achieved when making the stencil selection algorithms adaptive, based on the quality of the cells for unstructured meshes, that can in turn reduce the computational cost of WENO schemes. For assessing the performance of the developed algorithms well established test problems are employed. These include the least square approximation of polynomial functions, linear advection equation of smooth functions and solid body rotation test problem. Euler and Navier-Stokes equations test problems are also pursued such as the Shu-Osher test problem, the Double Mach Reflection, the supersonic Forward Facing step, the Kelvin-Helmholtz instability, the Taylor-Green Vortex, and the flow past a transonic circular cylinder.

Crown Copyright © 2019 Published by Elsevier Inc. This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/).

1. Introduction

Weighted Essentially non-Oscillatory (WENO) schemes have established themselves as a prevalent technique, for providing non-oscillatory properties, to various high-fidelity high-order numerical frameworks suited for unstructured meshes including the Finite Volume (FV) [1–14] and the Discontinuous Galerkin (DG) [15–21] frameworks. Their goal is to non-linearly combine the reconstructions emerging from a central stencil and various directional stencils. The stencil with the smoothest variation of the variables is going to have the largest influence in the reconstruction process. Although they carry a significant computational cost mostly associated with the reconstruction of polynomials from various stencils, they greatly enhance the robustness of high-order schemes when applied to challenging flow problems including the Large Eddy Simulation (LES) and implicit *i*LES simulation for the time-dependent compressible Navier-Stokes Equations. The presence of regions with smooth flow features and strong non-linear phenomena including shock-waves, the underlying mechanism of the WENO schemes of identifying the spatial regions in the close vicinity of a cell with the best quality -smoothest- information is the catalyst of the robustness of this family of schemes.

E-mail address: panagiotis.tsoutsanis@cranfield.ac.uk.

https://doi.org/10.1016/j.jcpx.2019.100037
2590-0552/Crown Copyright © 2019 Published by Elsevier Inc. This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/).

Due to the inherent increased computational cost of the WENO schemes for unstructured meshes, a recent focus has been to reduce their computational cost by employing optimised numerical libraries [22], or using a compact implementation of them called Compact WENO (CWENO) schemes [8–10] in order to reduce the size of the directional stencils. CWENO schemes are widely used for various problems [23–34], since they are considerably more compact, and efficient due to the simpler stencil selection algorithms. For instance the CWENO of Dumbser et al. [8] on unstructured meshes, employs a large central stencil suitable for the desired high-order of the accuracy, and several directional stencils of lower-order of accuracy. These are then non-linearly combined by considering the smoothness indicators of each of the stencils, although it has to be noted that the smoothness indicators might require some scaling [28] since they are coming from different size stencils and polynomial order. The compactness therefore emanates from the significantly smaller directional stencils compared to the original WENO schemes. Therefore they are considered an exciting alternative towards the traditional WENO schemes, with significant better computational performance due to their compactness. However there are numerous challenges associated with them such as the smoothness indicators scaling (WENO-Z)[28], distinguishing of discontinuous and high frequency waves from stencils with different polynomial orders, the need for reconstruction in characteristics variables as reported by Levy et al. [23] etc. Due to the unstructured meshes, and the nature of the WENO schemes in this context, it can be argued that there is a lot of freedom regarding the choice of stencils, since the arbitrariness of the meshes dictates that generic algorithms that can be employed for any mesh are desirable. The focus of this study is to develop various stencil selection algorithms, by firstly investigating the parameters that can influence the accuracy, robustness and efficiency, and then assessing the performance of the developed algorithms for various realistic test-problems.

The stencil selection algorithms are going to be differentiated in central and directional ones, since different challenges and requirements are encountered for each of these types. Although there is not a panacea in terms of the ideal stencil, due to the fact that realistic unstructured meshes have a variation in their cell-quality metrics, thus there are a lot parameters that need to be considered. In the present study the WENO-FV type of schemes based on the approaches of [2,3,11,35] is considered. It needs to stressed though that the subject stencil algorithms are valuable to other frameworks such as DG or Multi-dimensional Optimal Order Detection (MOOD) [36,37], since the associated freedom of selecting the stencils presents an overwhelming number of approaches to be adopted. It is evident that using only one central stencil and an unlimited scheme in the context of the MOOD approach one might observe a significantly better computational efficiency, and parallel performance compared to a WENO scheme. In practice however the frequent violation of the solution admissibility criteria of the MOOD paradigm arising from a high-order unlimited scheme, can translate in a significant computational overhead of recomputing the solutions for the cells and their neighbours that have been contaminated with invalid solutions. Therefore a central stencil of favourable characteristics in terms of size, quality, compactness and adaptive on the mesh topology can assist towards reducing this overhead.

A usually unfavourable characteristic of the WENO stencils is that they propagate farther away from the considered cell, that is directly related to the complexity of a parallel code, and its increased communication requirements due to the width of the stencils. The ultimate goal of the present approach is to reduce the computational cost, by producing compact stencils for WENO schemes that can provide significant savings in computational cost for realistic flow problems and unstructured meshes, while retaining their favourable non-oscillatory properties. The novelty of this study lies in the fact that adaptive stencil selection algorithms are developed that take into account various parameters, in order to reduce their computational footprint, and therefore make WENO schemes more affordable. To the best of the author's knowledge this is the first systematic approach of assessing the computational efficiency, accuracy and robustness of various developed stencil selection algorithms for WENO schemes on unstructured meshes. All the present stencil selection algorithms are developed in the UCNS3D code [22], which is also employed for all the test-problems.

A series of stringent 2D and 3D test-problems including the Linear advection, Euler and Navier-Stokes equations using unstructured meshes of various elements types are considered for assessing the performance of all the developed stencil selection algorithms. The paper is organized as follows. Section 2 is dedicated to the numerical framework used to describe the high-order finite-volume framework for unstructured meshes, focusing on the central and directional stencil selection algorithms and the parameters that need to be considered, their implementation in terms of pseudo-code algorithms, the fluxes and temporal discretisation. The numerical results obtained for the Linear-advection, Euler and Navier-Stokes equations are presented in Section 3 where the focus is on the performance of the developed algorithms. The conclusions drawn from the present study are outlined in the last section.

2. Numerical framework

Consider a 2D hyperbolic system defined on a 2D domain Ω, written in its conservative form:

$$\frac{\partial \mathbf{U}}{\partial t} + \nabla(F(\mathbf{U})) = 0, \tag{1}$$

where $\mathbf{U} = \mathbf{U}(\mathbf{x}, t)$ is the unknown function, $\mathbf{x} = (x, y)$ denotes the coordinates of a point of the domain Ω, and F is the physical flux. The physical domain Ω in 2D consists of any combination of conforming quadrilateral and triangular elements of surface area S_i, indexed by a unique mono-index i. Integrating Eq. (1) over a mesh element using a high-order explicit finite-volume formulation the following equation is obtained:

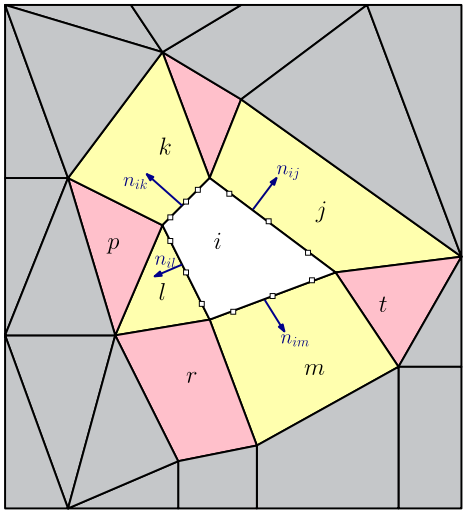


Fig. 1. Drawing illustrating the region around the considered cell i and the union of the direct-side only and vertex-based only neighbours, in yellow and pink colour respectively. The normal vectors for each of its sides, and the quadrature points at the edges in white colour are also depicted.

$$\mathbf{U}_i^{n+1} = \mathbf{U}_i^n - \Delta t \frac{1}{|S_i|} \sum_{j=1}^{N_f} \sum_{\alpha=1}^{N_{qp}} F^{\mathbf{n_{ij}}} \left(\mathbf{U}_{ij,L}^n(\mathbf{x}_{ij,\alpha}, t), \mathbf{U}_{ij,R}^n(\mathbf{x}_{ij,\alpha}, t) \right) \omega_\alpha |E_{ij}|, \tag{2}$$

where $\mathbf{U}_i$ are the volume averaged unknown function, $F^{\mathbf{n_{ij}}}$ is a numerical flux function in the direction normal to the cell interface between cell i and the neighbouring cell at the cell face j, N_f is the number of sides per element, N_{qp} is the number of quadrature points used for approximating the line integrals, $|E_{ij}|$ is the length of the corresponding edge, $\mathbf{U}_{ij,L}^n(\mathbf{x}_{ij,\alpha}, t)$ and $\mathbf{U}_{ij,R}^n(\mathbf{x}_{ij,\alpha}, t)$ are the high-order approximation of the solution at the left (considered cell) and the right (neighbouring cell) of the cell interface respectively; finally α corresponds to different Gaussian integration points $\mathbf{x}_\alpha$ and weights ω_α over the edge as shown in Fig. 1.

The Gauss-Legendre quadrature is employed for this study and the high-order approximation of the unknown function intercell fluxes is obtained by a reconstruction process that employs cell-averaged data. The following sections describe the methodology adopted for space and time discretisation.

2.1. Spatial discretisation

The spatial discretisation entails building a high-order polynomial $p_i(x, y)$ of order r, for each considered cell i, so that it has the same average as a general quantity $\mathbf{U}_i$. This can be formulated as:

$$\mathbf{U}_i = \frac{1}{|S_i|} \int_{S_i} \mathbf{U}(x, y)\, dS = \frac{1}{|S_i|} \int_{S_i} p_i(x, y)\, dS. \tag{3}$$

The reconstruction algorithm used is based upon the approaches [2,3,11,35], that have been applied to smooth and discontinuous flow problems [22,38–47] and only key-components will be presented herein, since the focus is towards the stencil selection algorithms and the reader is referred to [2,3,35] for further details.

For reducing the scaling effects for elements of different size the reconstruction is carried out in a transformed system of coordinates for improving the condition number of the system of equations as suggested by Dumbser et al. [11,12,48]. The main difference being that for non triangular or tetrahedral elements the transformation is done by decomposing each element into triangular or tetrahedral elements and using one of the decomposed elements as the reference element for transforming to the new system of coordinates as suggested by Tsoutsanis et al. [2,3,35]. The reader is referred to [3] for details regarding the decomposition strategy adopted.

Let $\mathbf{v}_{ij}$, $j = 1, 2, \ldots J_i$ be the vertices of the considered 2D general element. By decomposing non triangular elements into triangles and choosing one of them with $\mathbf{w}_1 = (x_1, y_1)$, $\mathbf{w}_2 = (x_2, y_2)$, $\mathbf{w}_3 = (x_3, y_3)$, being its three vertices. The transformation from the Cartesian coordinates x, y into a reference space ξ, η is given by the following equations:

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} x_1 \\ y_1 \end{pmatrix} + J \cdot \begin{pmatrix} \xi \\ \eta \end{pmatrix}, \tag{4}$$

with the Jacobian matrix given by:

$$J = \begin{bmatrix} x_2 - x_1 & x_3 - x_1 \\ y_2 - y_1 & y_3 - y_1 \end{bmatrix}. \tag{5}$$

Using an inverse mapping the element S_i can be transformed to the element S_i' in the reference co-ordinate system as:

$$\mathbf{v}_{ij}' = J^{-1} \cdot \left(\mathbf{v}_{ij} - \mathbf{w}_1\right), \quad j = 1, 2, \dots J_i, \tag{6}$$

and the spatial average of the conserved variable $\mathbf{U}_i$ does not change during transformation

$$\mathbf{U}_i = \frac{1}{|S_i|} \int_{S_i} \mathbf{U}(x, y)\, dS \equiv \frac{1}{|S_i'|} \int_{S_i'} \mathbf{U}(\xi, \eta)\, d\xi d\eta. \tag{7}$$

To perform the reconstruction on the target element S_i, the central stencil $\mathcal{S}$ is formed by recursively adding neighbouring elements, consisting of $M + 1$ elements, including the target element S_i

$$\mathcal{S} = \bigcup_{m=0}^{M} S_m, \tag{8}$$

where the index m refers to the local numbering of the elements in the stencil, with the element with index 0 being the considered cell i.

Remark 1. Different ways to add the M neighbouring elements for the central stencil, how many to add, and what parameters to be considered are investigated in this study.

The considered cell and its corresponding stencil elements are transformed in the reference space,

$$\mathcal{S}' = \bigcup_{m=0}^{M} S_m', \tag{9}$$

and the r^{th} order reconstruction polynomial at the transformed cell S_0' is sought as an expansion over local polynomial basis functions $\phi_k(\xi, \eta)$ given by:

$$p(\xi, \eta) = \sum_{k=0}^{K} a_k \phi_k(\xi, \eta) = \mathbf{U}_0 + \sum_{k=1}^{K} a_k \phi_k(\xi, \eta), \tag{10}$$

where $\mathbf{U}_0$ corresponds to the vector of conserved variables at the considered cell i, and ξ, η are the coordinates in the reference system. a_k are the degrees of freedom and the upper index in the summation of expansion K corresponds to the number of the degrees of freedom.

To find the unknown degrees of freedom a_k for each cell m from the stencil the cell average of the reconstruction polynomial $p(\xi, \eta)$ must be equal to the cell average of the solution $\mathbf{U}_m$:

$$\int_{S_m'} p(\xi, \eta)\, d\xi d\eta = |S_m'|\mathbf{U}_0 + \sum_{k=1}^{K} \int_{S_m'} a_k \phi_k\, d\xi d\eta = |S_m'|\mathbf{U}_m, \quad m = 1, \dots, M. \tag{11}$$

Where S_m' is the surface area of the cell m in the stencil in the transformed coordinate system. Since for general cells the transformation in a unit quadrilateral cell cannot be guaranteed, the basis functions ψ_k used should satisfy the constraint of equation (3) irrespective of the values of degrees of freedom. The basis functions ψ_k for all the elements in the stencil are defined as follows:

$$\phi_k(\xi, \eta) \equiv \psi_k(\xi, \eta) - \frac{1}{|S_0'|} \int_{S_0'} \psi_k\, d\xi d\eta \quad k = 1, 2, \dots, K, \tag{12}$$

where S_0' is the surface area of the considered cell i, in the transformed coordinate system, and in the present study ψ_k are Legendre polynomials basis functions. Denoting the integrals of the basis function k over the cell m in the stencil, and the vector of right-hand side by A_{mk} and b respectively as given by

$$A_{mk} = \int_{S_m'} \phi_k\, d\xi d\eta, \quad b_m = |S_m'|(\mathbf{U}_m - \mathbf{U}_0),$$

the equations for degrees of freedom a_k can be rewritten in a matrix form as:

$$\sum_{k=1}^{K} A_{mk} a_k = b_m, \quad m = 1, 2, \ldots M. \tag{13}$$

The complete linear system is given by:

$$\begin{bmatrix} A_{1,1}(\xi,\eta) & A_{1,2}(\xi,\eta) & A_{1,3}(\xi,\eta) & \cdots & A_{1,K}(\xi,\eta) \\ A_{2,1}(\xi,\eta) & A_{2,2}(\xi,\eta) & A_{2,3}(\xi,\eta) & \cdots & A_{2,K}(\xi,\eta) \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ A_{M,1}(\xi,\eta) & A_{M,2}(\xi,\eta) & A_{M,3}(\xi,\eta) & \cdots & A_{M,K}(\xi,\eta) \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_K \end{bmatrix} = \begin{bmatrix} |S_1'|(\mathbf{U}_1 - \mathbf{U}_0) \\ |S_2'|(\mathbf{U}_2 - \mathbf{U}_0) \\ \vdots \\ |S_M'|(\mathbf{U}_M - \mathbf{U}_0) \end{bmatrix}. \tag{14}$$

The matrix A_{mk} is purely based on the geometry of the elements in the stencils, which remains unchanged for the present study, while the information on the right-hand vector b_m is dependent upon the solution, and the surface area of each of the element in the stencil in the transformed coordinate system. The matrix $A_{km}^T A_{mk}$, is invertible and the equation (13), can be rewritten in the following form:

$$a_k = \left(A_{km}^T A_{mk}\right)^{-1} A_{km}^T b_m = A_{km}^{\dagger} b_m, \tag{15}$$

where the matrix $A_{km}^{\dagger}$ corresponds to the Moore-Penrose pseudo-inverse of A_{mk}. The polynomial coefficients are therefore obtained through a matrix vector multiplication of $A_{km}^{\dagger}$ with b_m. In the present study a QR decomposition based on Householder transformation [49] is used, where the pseudo-inverse is decomposed in the following manner in an orthogonal matrix Q and an upper triangular one R, and the inverse of $R^T R$ is obtained by a forward-substitution followed by a backward-substitution.

$$A_{km}^{\dagger} = \left((Q\,R)^T (Q\,R)\right)^{-1} A_{km}^T = \left(R^T R\right)^{-1} A_{km}^T \tag{16}$$

The main memory requirements of the present least-squares reconstruction is the associated storage of the pseudo-inverse matrix $A_{km}^{\dagger}$, that is only computed once at the preprocessing stage of the simulation.

2.1.1. WENO scheme

The WENO scheme used in this study, employs a non-linear combination of various reconstruction polynomials from the central stencil and additional directional stencils; each polynomial is weighted according to the smoothness of its solution, and it is based on the approaches of [2,3,11,48].

Remark 2. In this study, the number of directional stencils, how to be constructed, and aspects that need to be taken into account are investigated.

The polynomials are given as:

$$p_i(\xi,\eta)^{\text{weno}} = \sum_{s=1}^{s_t} \omega_m p_s(\xi,\eta), \tag{17}$$

where s_t is the total number of WENO stencils. Substituting back to Eq. (10) for $p_s(\xi,\eta)$, we obtain the following expression:

$$p_s(\xi,\eta) = \sum_{k=0}^{K} a_k^{(s)} \phi_k(\xi,\eta). \tag{18}$$

Using the condition that the sum of all weights is unity, yields:

$$\begin{aligned} p_i(\xi,\eta)^{\text{weno}} &= \mathbf{U}_0 + \sum_{k=1}^{K} \left(\sum_{s=0}^{s_t} \omega_s a_k^s \right) \phi_k(\xi,\eta) \\ &\equiv \mathbf{U}_0 + \sum_{k=1}^{K} \tilde{a}_k \phi_k(\xi,\eta), \end{aligned} \tag{19}$$

where $\tilde{a}_k$ are the reconstructed degrees of freedom; and the non-linear weight ω_m is defined as:

$$\omega_s = \frac{\tilde{\omega}_s}{\sum_{s=1}^{s_t} \tilde{\omega}_s} \quad \text{where} \quad \tilde{\omega}_s = \frac{\lambda_m}{(\epsilon + \mathcal{SI}_s)^b}. \tag{20}$$

The smoothness indicator $\mathcal{SI}_m$ is given by:

$$\mathcal{SI}_s = \sum_{1\leq|\beta|\leq r} \int_{V_0'} \left(\mathcal{D}^\beta p_s(\xi,\eta)\right)^2 (d\xi, d\eta), \tag{21}$$

where β is a multi-index, r is the polynomial's order, λ_m is the linear weight. The central stencil is assigned a large linear weight of $\lambda_1 = 1000$ and a value to prevent division by zero of $\epsilon = 10^{-6}$ is used and $\mathcal{D}$ is the derivative operator. The smoothness indicator is a quadratic function of the degrees of freedom (a_k^s) and Eq. (21) can be rewritten as:

$$\mathcal{SI}_s = \sum_{k=1}^{K} a_k^s \left(\sum_{q=1}^{K} \mathcal{OI}_{kq} a_q^s \right), \tag{22}$$

where the oscillation indication matrix $\mathcal{OI}_{kq}$ is given by:

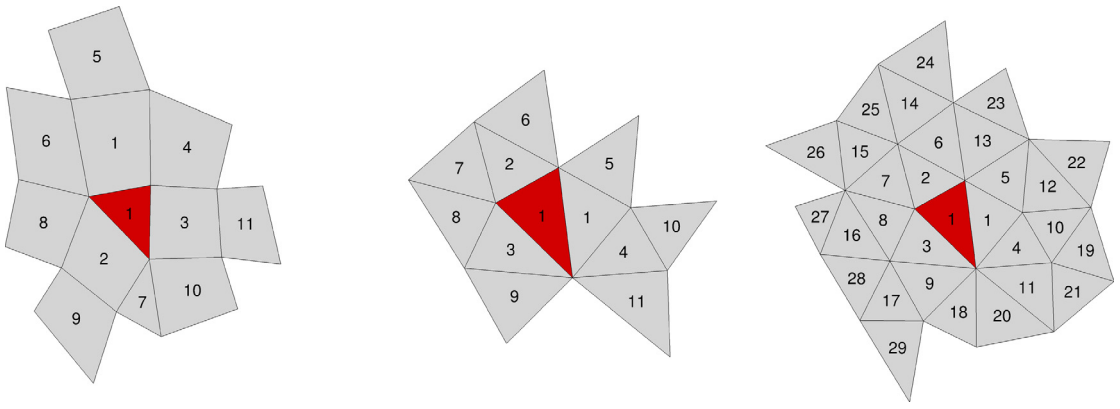
$$\mathcal{OI}_{kq} = \sum_{1\leq|\beta|\leq r} \int_{V_0'} \left(\mathcal{D}^\beta \phi_k(\xi,\eta)\right)\left(\mathcal{D}^\beta \phi_q(\xi,\eta)\right) (d\xi, d\eta), \tag{23}$$

and can be easily precomputed and stored at the initialisation stage of the simulation. Additionally the WENO reconstruction for the present study is carried out with respect to the characteristic variables, and the reader is referred to [2,3,35] and references therein regarding the implementation. The algorithms involved in the construction of the stencils for WENO schemes are going to be differentiated between the central and directional stencils, since each family of stencils are subject to different requirements.

2.2. Central stencil algorithms

Before considering the most important parameters for developing the central stencils, it is of great significance to highlight their role. For the present WENO implementation the central stencil is assigned a large linear weight, thus at smooth regions of the flow the reconstruction polynomial from the central stencil is going to have the largest influence in the resulting non-linear polynomials. Additionally for k-exact least square reconstruction, the central stencil is ultimately employed for obtaining the unlimited gradients for the diffusive terms. Therefore the parameters considered for the central stencil with the greater substance in the context of the present WENO implementation are the following:

- Number of Cells (M): *The number of cells included in the stencil are related to the computational cost and robustness of the scheme. The number of unknowns in the polynomials is given by K where it is* $K = \frac{1}{2}(r+1)(r+2) - 1$ *and* $K = \frac{1}{6}(r+1)(r+2)(r+3) - 1$ *in 2D and 3D respectively. Therefore at least K number of elements are required in the stencil, in order to have at least as many equations as unknowns. In practice, however this is not sufficient it is prevalent in most of the approaches [2,11,36,37,48,50,51] in the k-exact least squares framework tend to use more elements (ranging from* $M = 1.5K$ *to* $M = 3K$*) and thus equations, than the number of the unknowns. The more equations used, the more robust and computationally expensive the scheme becomes. This is due to a lower probability of the cells distribution suffering from alignment resulting in a singular the Moore-Penrose, or at least provide a very large condition number of matrix* A_{mk}*. One approach is adding neighbours in the stencil until a target condition number for the matrix of the linear system has been achieved similarly to the approaches of [36,37]. However one should ensure that some stop criteria regarding the maximum number of cells are introduced, in case the target metric (this could be condition number, error approximation etc) can not be reached. It needs to be stressed that all the stencil formation algorithms and their accompanying quality indicators operators (condition number etc.) are performed in the initialisation phase of the simulation (assuming the mesh remains the same in time) and therefore algorithms should have stop criteria to ensure that this initialisation phase is not prolonged due to unrealistic target not being met. A similar approach of having a max limit of number of cells is followed by Diot [52] as outlined in his thesis where an upper limit of* $M = 1.5K$ *and* $M = 2K$ *is used for 2D and 3D meshes respectively. Another approach could be to add layers of neighbours that surround all the vertices of the considered element, until the required number of layers has been reached [50], although there still might be cases where an excessive number of cells has been used. It is therefore crucial to reduce the computational cost without sacrificing the robustness of the scheme. The task is not as straightforward since there are other parameters such as compactness, cell quality that all of them interact with each other.*
- Compactness: *Independently of the number of cells used, the stencil size should be as compact as possible for better locality when computing polynomials and for an enhanced locality when dealing with the parallelisation of the WENO schemes, and the associated communication overheads when exchanging information for the halo stencil elements. Compactness results in lower errors of polynomial approximations and more importantly in the vicinity of a shock, a compact stencil provides better chance of resolving a shock without producing spurious oscillations compared to a larger stencil. This is one of the cornerstones of the success of the compact WENO schemes (CWENO) on unstructured meshes of Dumbser et al. [8] and Levy et al. [10] where a large central stencil of the desired order of polynomials is combined with several smaller directional stencils of lower-order polynomials, and therefore improvement in computational times, parallelisation efficiency and resolution at shocks is achieved. However compactness can*



(a) $\mathcal{P} = 2$ (b) $\mathcal{P} = 2$ (c) $\mathcal{P} = 4$

Fig. 2. Examples of central stencils using the Naive Cell based algorithm for various polynomial orders $\mathcal{P}$. The considered cell in red colour, and the neighbouring elements are numbered with the order that they are added in the stencil (starting with 1 from the first element), highlighting how the algorithm works.

be pursued only when certain cell quality criteria are satisfied. Due to the multidimensional nature of the present reconstruction, when most of the stencil elements are aligned with any direction, this reduces the robustness of the simulation. Therefore in the presence of highly-skewed, high-aspect ratio cells selecting the closest one for the central stencil will result in linear matrix system of equations with a large condition number and thus reduced robustness and accuracy.

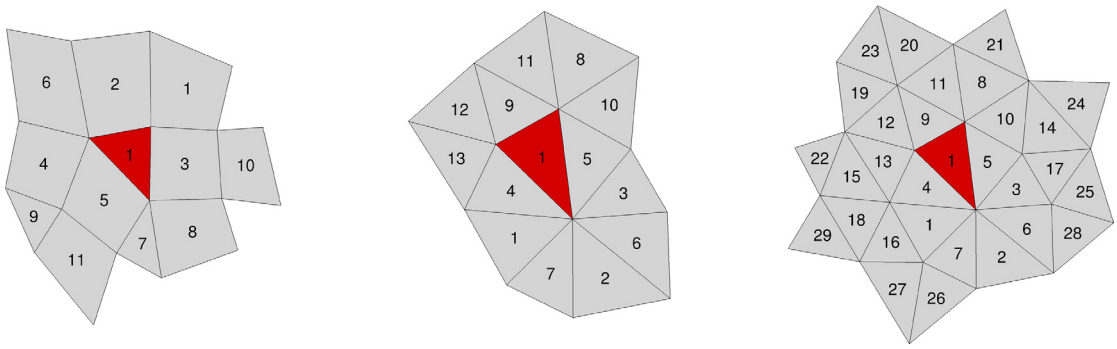
- Cell Quality: *For realistic meshes where there will be combination of regions of good and bad quality cells, it is of paramount importance to take that into account, for the sake of fortifying the stencil selection algorithm. In the course of selecting how many cells to add, if the closest ones to the considered cell can be used, how to add the number of elements etc, the quality of the considered cell, or even its close vicinity needs to be taken into account. There various options that one could explore in order to adapt the stencil selection algorithm. There could be a focus on cell quality characteristics of the considered cell only, the direct side neighbours, or the entire stencil neighbourhood. In meshes of smooth transition from one elements size/shape to another the quality criterion based on a cell or direct side neighbours only might be sufficient to indicate the quality characteristics of the region, whereas in generic meshes where the transitions between element sizes and shapes is rapid the entire stencil neighbourhood quality characteristics need to be considered. If the objective is to resort to more compact algorithms and take advantage of the benefits offered then numerous quality criteria in a cell-level, direct-side level, or stencil-level could be deployed to determine if such an algorithm is feasible and robust within this context. Additionally various cell quality metrics could be used such as aspect-ratio, skewness, maximum angle etc.*

Within the context of these aforementioned parameters there is an overwhelming plethora of combinations, configurations that could be explored ranging from condition number, different cells quality criteria, computational costs etc. In this study a computational framework of four central stencils algorithms is presented by taking into account the most dominant parameters for the set of test-problems presented. It need to be stressed that the stencil selection process for the present WENO algorithm is taking place only once at the initialisation of the UCNS3D code, since the mesh remains fixed during the calculation, therefore the computational cost of the selection algorithms is of limited substance. All the stencil algorithms developed in this study, make use of a "Big Stencil" which consists of mx number of elements ($mx >> M$) -typically ($mx = 4 \cdot M$)- for ensuring that various selection criteria can be employed for defining the selection of the central stencil S_m. Finally the target number of elements M is given by ($M = EX \cdot K$) where EX is the expansion factor.

2.2.1. The Naive Cell based algorithm (NCB)

The first algorithm which seems a natural selection for a cell based solver, is recursively adding the direct side-neighbours of the considered cell, until the target number of cells has been reached. This algorithm does not take into account the quality of the mesh neither the distance of the candidate elements from the considered cell, and is therefore labelled the Naive Cell based algorithm. As it can be seen in Algorithm 1, the Big Stencil and the Sort Store procedures are used as shown in Algorithm 5, for recursively adding the direct side neighbours until mx number of elements have been collected and storing the M number of elements in the stencil S_m respectively. It must be highlighted that the Big Stencil procedure avoids adding duplicate elements in the stencil (line 14 of Algorithm 5). Variables definitions for stencil algorithms are presented in Table 9, Appendix B.

From Fig. 2 typical examples of stencils from the NCB algorithm can be seen. It can be noticed that the first direct side-neighbours of the considered cell are added, and then the direct side neighbours of the first-neighbours as seen by the numbering of the elements. The key-advantage of this algorithm is its simplicity, robustness (provided that sufficient number of cells are used), and reduced computational cost since there is no requirement for any information with respect to the quality of the mesh. The associated low-computational cost can be beneficial when considering meshes that change



(a) $\mathcal{P} = 2$ (b) $\mathcal{P} = 2$ (c) $\mathcal{P} = 4$

Fig. 3. Examples of central stencils using the Vertex based Compact Algorithm for various polynomial orders $\mathcal{P}$. The considered cell in red colour, and the neighbouring elements are numbered with the order that they are added in the stencil (starting with 1 from the first element), highlighting how the algorithm works.

with time. Its key disadvantages are the non-compactness of the stencil and the arbitrary distribution of the elements that can result in non-symmetric stencils.

2.2.2. The vertex based compact algorithm (VBC)

Independently of the nature of the numerical framework (cell-based or vertex-based), a stencil algorithm that is complemented by a process to include the cells closest to the considered cell is beneficial since the resulting stencils can be symmetrical with respect to the considered cell for regions of good mesh quality. The key challenges regarding the number of cells for the VBC algorithm are the following:

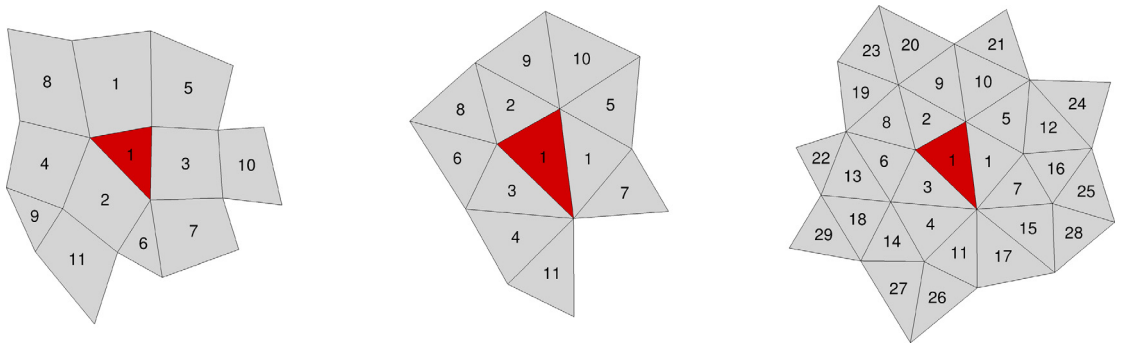
- For low-quality meshes the number of vertex neighbours for the considered cell could become excessive, since there could be vertices that are shared by numerous elements -in particular triangles, tetrahedrals and pyramids.
- For good-quality meshes the number of vertex neighbours for the considered cell, might not be sufficient to reach the target number of cells in the stencil.

Therefore the algorithm is designed in such a way that firstly the vertex-based neighbours of the considered cell are added, and if the required number of elements has not being reached *mx*, then direct-side neighbouring elements are recursively added (starting from the list of the existing vertex based neighbours) until the required number of elements *mx* has been reached as shown in Algorithm 2 and Algorithm 5.

The course of making the stencil more compact is pursued by firstly computing the maximum distance of the centre of all the vertex based neighbours from the cell centre of the considered cell i. Then all the remaining elements in the Big Stencil are sorted with respect to their distance from the considered cell i, and the ones that are within the radius of the farthest vertex based element in the stencil are added until the target number of elements has been reached $M = max(counter3, EX \cdot K)$. This way the robustness of the compactness algorithm is increased by sorting with respect to the distance only the non-vertex based neighbours, and only if the Cells quality criterion has been satisfied. One could use various cell quality criterion including the aspect-ratio, skewness, maximum angles etc. In the subject study any cell with an aspect ratio less than 10 is considered suitable for enabling the compactness function. In Fig. 3 the resulting stencils form the VBC algorithm can be seen for $EX = 2$. It can be noticed that in the $\mathcal{P} = 2$ examples the left one (a) has 8 vertex based neighbours (1-8) and the closest additional three direct-side neighbours (9-11) are added then. Whereas in the right one (b) all the vertex based neighbours are added $M = max(13 + 1, 2 \cdot 6)$ which happen to be more than the number of elements required. Finally for the $\mathcal{P} = 4$ stencil all the remaining direct-side neighbours from $14 - 29$ are selected based on their distance from the considered cell i.

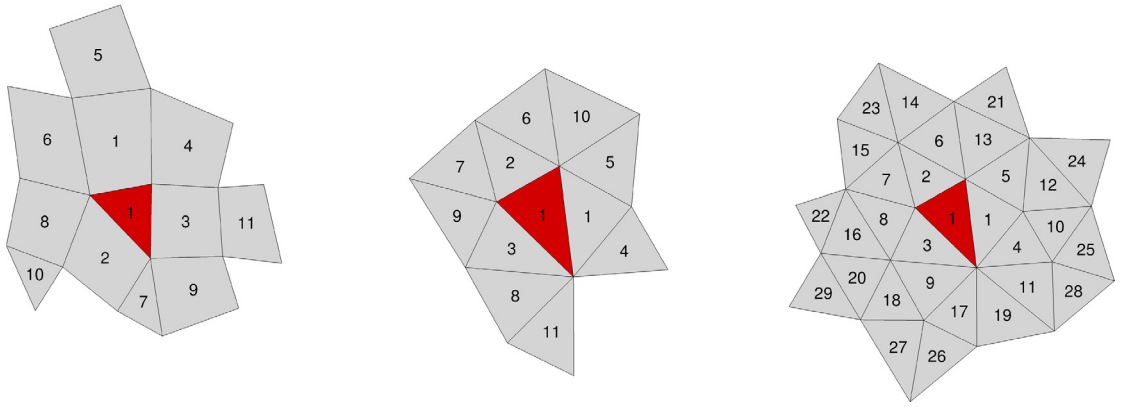
2.2.3. The face based compact algorithm (FBC)

The face based compact algorithm is similar to the (VBC) algorithm in the sense that a specific number of elements is firstly added, and then the remaining added ones are subject to the compactness procedure. Therefore initially the direct-side neighbours are added and then succeeded by the closest ones to the considered cell i for cells of good quality, as it can be seen in Algorithm 3. The key motivation for developing this algorithm is that there might be instances where the direct-side neighbours might result in a shorter maximum distance as opposed to the first-layer of vertex-based neighbours for meshes of poor quality. It is therefore a way to make the resulting stencil more compact, by always selecting only the direct-side neighbours (obtaining their maximum distance d_f from the considered cell i) and then adding as many elements required that are within the radius of d_f until the target number of elements has been reached $M = max(counter3, EX \cdot K)$.



(a) $\mathcal{P} = 2$ (b) $\mathcal{P} = 2$ (c) $\mathcal{P} = 4$

Fig. 4. Examples of central stencils using the Face based Compact Algorithm for various polynomial orders $\mathcal{P}$. The considered cell in red colour, and the neighbouring elements are numbered with the order that they are added in the stencil (starting with 1 from the first element), highlighting how the algorithm works.



(a) $\mathcal{P} = 2$ (b) $\mathcal{P} = 2$ (c) $\mathcal{P} = 4$

Fig. 5. Examples of central stencils using the Stencil based Compact Algorithm for various polynomial orders $\mathcal{P}$. The considered cell in red colour, and the neighbouring elements are numbered with the order that they are added in the stencil (starting with 1 from the first element), highlighting how the algorithm works.

The resulting stencils from the FBC algorithm for $EX = 2$ can be seen in Fig. 4, where it can be noticed that the direct side neighbours are firstly added and then the numbering sequence from 4 and onwards indicates the immediately closest elements. Where from the VBC this sequencing is noticed once the vertex side neighbours have been added.

2.2.4. The stencil based compact algorithm (SBC)

The stencil based compact algorithm, is the only algorithm in the group where the number of elements that are added to the stencil without being sorted with respect to their distance is dependent upon the order of accuracy. In particular as the number of polynomial coefficients (K) increase with the order of accuracy r, the number of elements that are initially added to the stencil without any criterion are also increased since the $counter3 = 1.2 \cdot K$. Then the same process as the other compact algorithm is followed, where the maximum distance d_f within the first $counter3$ elements is found and the closest remaining direct side neighbours within this distance are added until the target number of cells has been reached as seen in Algorithm 4.

The main motivation for developing this algorithm is that for the first set of $counter3$ elements this algorithm features similar robustness to the Naive Cell based algorithm. Since elements are added without any restriction based solely on their direct-side connectivity. For the remaining elements the compactness procedure is applied until the target number of elements has been reached.

From the examples of the stencils built with this algorithm as seen in Fig. 5 it can be noticed that for the $\mathcal{P} = 2$ examples the right one (b) is quite similar to the equivalent VBC, since the compactness algorithm is only applied for the elements 8–11. This algorithm in theory should be the most robust one, since it combines the robustness of the naive for the first $1.2 \cdot K$ elements, and a significant portion of the additional elements subject to the compactness procedure belongs to the vertex-based neighbours of the considered cell i.

2.3. Directional stencil algorithms

The purpose of the directional stencils in the present WENO schemes is to provide additional polynomial reconstructions from different spatial directions, so that in regions of the flows with sharp gradients at least one of the directional stencils will be located in a smooth region. In other words the probability of at least one polynomial arising from a stencil being in a smooth region is greater when using multiple directional stencils. The most dominant-parameters that need to be considered for constructing the directional stencils are as follows:

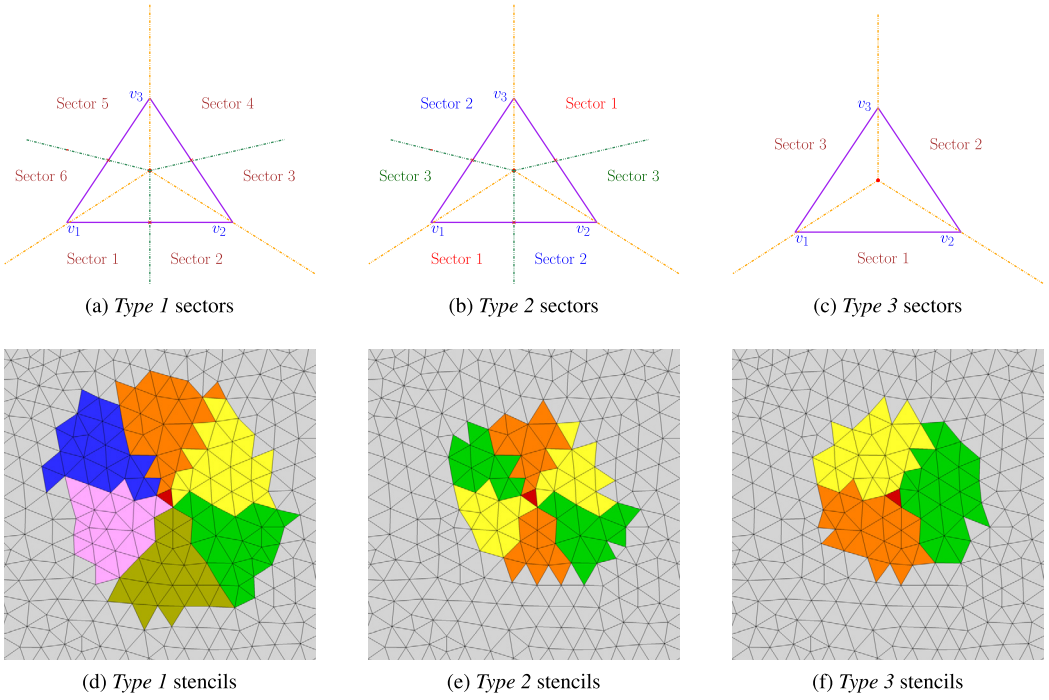
- Width: *The directional stencils ideally should not be defined by narrow geometrical sectors, where elements of bad quality can lead to directional stencils that most of the elements are aligned along a direction, and due to the multidimensional nature of the reconstruction the condition number of the matrices is going to be very high in such cases. The width of the stencil can be described by the parameter* (w_s) *which is the maximum distance of any element -from the list of stencil elements M- from the considered cell i, and will be non-dimensionalised with respect to the radius* (r_i) *of the inscribed circle or sphere size of the considered cell i* $(w_s = \frac{max(d_m)}{r_i})$.
- Symmetry: *Ideally the directional stencils should be as symmetric as possible away from the boundaries with respect to the considered cell and hence more compact, similar to the central stencils with respect to the considered cell for accuracy reasons. The symmetry property however, limits the number of stencils that can be built around cells of various shapes, hence limiting the non-oscillatory properties of the resulting WENO schemes.*
- Number: *The number of the directional stencils is proportional to the robustness and non-oscillatory character of the scheme. On the other hand increasing the number of directional stencils translates in an increase in computational cost, since reconstructions need to be performed for each one of them. Therefore finding a balance between these aspects is far from trivial.*
- Cover: *The directional stencils should be covering the region of a complete circle in 2D, or a sphere in 3D, in order to ensure that all elements from all possible directions within a distance from the considered cell are considered. Ultimately, this will lead in improved robustness of the algorithm at regions of sharp gradients, since all possible directions are considered. The radius* (R_s) *of the associated circle, or sphere as a fraction of the considered cell inscribed circle or sphere size, is dependent upon the polynomial order and stencil algorithm used.*
- Uniqueness: *The number of common elements across various directional stencils for a considered cell is defined as uniqueness. This is one of the most obscure components of the directional stencils, since it is linked with the number of stencils that can be produced, the non-oscillatory properties of the scheme and the compactness of the directional stencils. For example an algorithm that can accept non-unique directional stencils, in other words some elements can be shared with other directional stencils can result in many more and more compact stencils. This might be beneficial in terms of the condition number of the resulting matrices, and the fact that since many more directional stencils can be admitted. On the other hand, the possibility of significantly more stencils with some overlapping might not jeopardise the non-oscillatory properties of the scheme but it will increase the computational cost. Therefore overlapping can be admitted if the number of directional stencils is larger than the number of the sides of the considered cell, whereas in other cases overlapping has not been proven to be beneficial as outlined in the directionality condition of Tsoutsanis [53].*

For the scope of the present study only unique stencils are considered since they have provided a more robust WENO framework so far for various set of applications [2,3,22,38–42,44,46]. The Big Stencil from the central stencil algorithm is employed here, and all the elements from the Big Stencil that satisfy a set of geometrical conditions are added to the directional stencils. In particular the geometrical condition that needs to be satisfied for any selected geometrical sector is the cell centre of a candidate element must lie within this sector. This is achieved by using a triangular or tetrahedral sector in 2D or 3D, consisting of two vertices and the cell centre or three vertices and the cell centre respectively. Once these triangles or tetrahedrals emerging from these separate geometrical sectors as seen throughout Fig. 6, Fig. 7 and Fig. 8 have been defined, they are transformed to the reference co-ordinate system. Subsequently the coordinates (cc) of the cell-centres of any candidate cell are transformed to the reference co-ordinate system based on the same Jacobian matrix and vertices $\mathbf{w}_{1,2,..,N}$ of the pre-defined triangle or tetrahedral sector, and if all of the transformed coordinates $\mathbf{cc}'$ are positive then -the coordinates lie in the pre-defined geometrical sector- the geometrical condition is satisfied.

$$\mathbf{cc}' = J^{-1} \cdot (\mathbf{cc} - \mathbf{w}_1)\,. \tag{24}$$

It needs to be noted that depending on the quality of the cells, similarly to the central stencil algorithms the compactness procedure can be applied to ensure that the closest elements with respect to the considered cell i, that also satisfy the geometrical conditions are selected. A description of the directional stencil algorithms procedure can be found in the Algorithm 6 provided. For this study four different types of directional stencil algorithms are developed and explored as listed below:

- ***Type 1***: *This algorithm uses two sectors per face by combining the cell centre, the vertices of the considered face and one additional point either from the centre of the considered edge (2D) or any of the face edges in 3D, as it can be seen in Fig. 6, Fig. 7 and Fig. 8. This algorithm is characterised by the largest number of directional stencils, and the widest stencils, since the geometrical sectors are quite narrow and as a result they propagate farther in all directions. In principle it is expected to be the most robust one, since*



(a) *Type 1* sectors (b) *Type 2* sectors (c) *Type 3* sectors

(d) *Type 1* stencils (e) *Type 2* stencils (f) *Type 3* stencils

Fig. 6. Schematics of various directional stencil algorithms illustrating the directional sectors, and their corresponding stencils for a triangular considered cell for a scheme using a fourth-order polynomial.

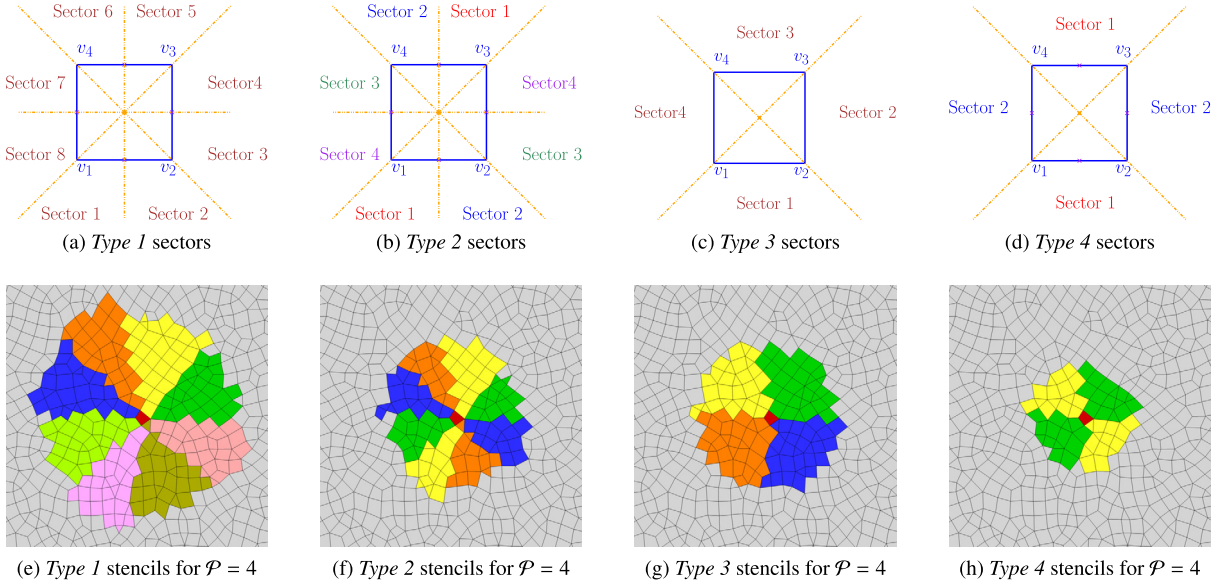
it used the most directional stencils, hence largest possibility of at least one stencil lying in a smooth region. However if the quality of cells is such that these sectors become narrow, there might be instances where even if one of the stencils lies in a smooth region the condition number of the least squares matrix is relatively high due to the alignment of the stencil elements along a direction.

- ***Type 2***: *This algorithm uses the exact same sectors per face as the Type 1, with the main difference that the sectors opposite to each other are combined in such a way that the total number of stencils is equal to the number of faces or edges in 2D as it can be seen in Fig. 6 and Fig. 7. The primary motivation for this developing this algorithm is firstly to reduce the total number of stencils, secondly to increase the compactness of the set of all the stencils, and finally to improve the symmetry of the directional stencils by clustering all the elements around the considered cell. There are two prevalent challenges of this algorithm firstly that the width of the directional stencils is now worse than Type 1 since they span a greater distance, and secondly that the narrow sectors arising from cells of low-quality has not been eliminated.*
- ***Type 3***: *This algorithm is focused on finding a good balance between all the dominant-parameters including width, cover, symmetry and uniqueness. There is only one geometrical sector per face or edge as shown in Fig. 6, Fig. 7 and Fig. 8. It has to be noted that for quadrilateral faces in 3D as shown in Fig. 7, the geometrical sector defined by vertices $(v_1, v_2, v_3, v_4, v_c)$ implies that there are four tetrahedrals (v_1, v_2, v_3, v_c), (v_2, v_3, v_4, v_c), (v_3, v_4, v_1, v_c) and (v_4, v_1, v_2, v_c) that at least one of them must satisfy the geometrical condition set by equation (24). Similar algorithms have been previously used in the context of WENO schemes by [2,3,8,35] with good results in smooth and discontinuous flow problems.*
- ***Type 4***: *This algorithm employs the same sectors as Type 3, with the main difference being that the opposite facing stencils are combined in such a way that the total number of directional stencils is reduced to two for a quadrilateral element as seen in Fig. 7. One of the key challenges of this type is that it is difficult to define suitable symmetrical sectors for triangles and triangular faces for 2D and 3D respectively. The main disadvantage of this scheme is the fact that the number of stencils is very limited and consequently the non-oscillatory properties of the WENO scheme might be impaired if strong gradients from various directions are experienced, on the other hand the computational cost is certainly going to be significantly reduced compared to other types.*

2.4. Fluxes approximation & temporal discretisation

For the evaluation of the intercell numerical flux function $\mathbf{F}^{\mathbf{n}_{ij}}$ for the Euler and Navier-Stokes equations the approximate HLLC (Harten-Lax-van Leer-Contact) Riemann solver of Toro [54] is employed. For the evaluation of the viscous fluxes the unlimited k-exact least square reconstruction is used for the gradients and they are then averaged from two discontinuous states as detailed in [3,9]. For the gradients additionally penalty terms are included following the formulation of Gassner et al. [55] for suppressing odd-even decoupling modes in the numerical solutions [56], in the following manner:

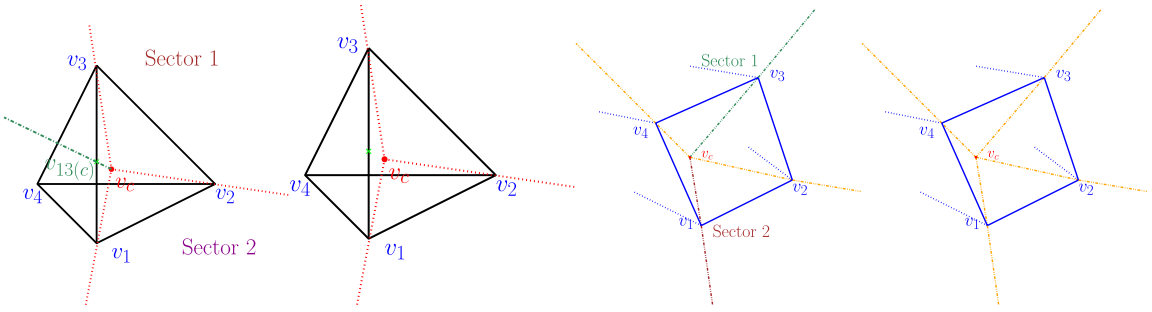
$$\nabla \mathbf{U} = \frac{1}{2}\left(\nabla \mathbf{U}_L + \nabla \mathbf{U}_R\right) + \frac{\alpha}{L_{int}}\left(\mathbf{U}_R - \mathbf{U}_L\right)\vec{n}, \tag{25}$$



(a) *Type 1* sectors (b) *Type 2* sectors (c) *Type 3* sectors (d) *Type 4* sectors

(e) *Type 1* stencils for $\mathcal{P} = 4$ (f) *Type 2* stencils for $\mathcal{P} = 4$ (g) *Type 3* stencils for $\mathcal{P} = 4$ (h) *Type 4* stencils for $\mathcal{P} = 4$

Fig. 7. Schematics of various directional stencil algorithms illustrating the directional sectors, and their corresponding stencils for a quadrilateral considered cell for a scheme using a fourth-order polynomial.

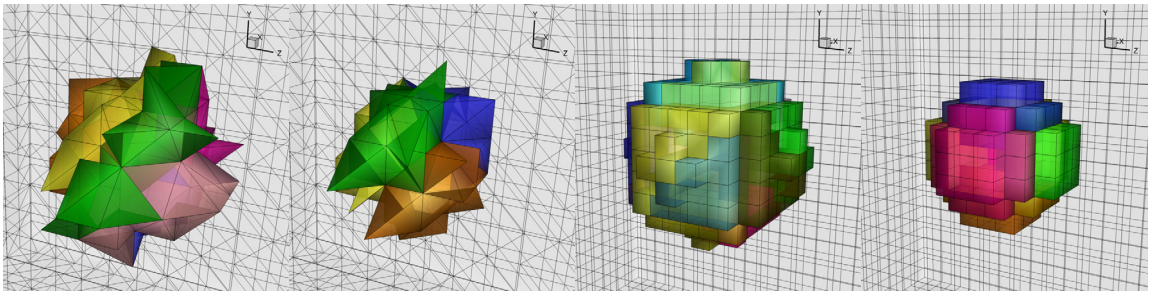


(a) *Type 1* sectors for triangular face, sector 1 consisting of vertices $\left(v_2, v_3, v_{13c(c)}, v_c\right)$ and sector 2 of vertices $\left(v_1, v_2, v_{13c(c)}, v_c\right)$

(b) *Type 3* sectors for triangular face, one sector per face of vertices (v_1, v_2, v_3, v_c)

(c) *Type 1* sectors for quadrilateral face, sector 1 consisting of vertices (v_2, v_3, v_4, v_c) and sector 2 of vertices (v_1, v_2, v_4, v_c)

(d) *Type 3* sectors for quadrilateral face, one sector per face of vertices $(v_1, v_2, v_3, v_4, v_c)$



(e) *Type 1* stencils for triangular faces (f) *Type 3* stencils for triangular faces (g) *Type 1* stencils for quadrilateral faces (h) *Type 3* stencils for quadrilateral faces

Fig. 8. Schematics of the two directional stencil algorithms for triangular and quadrilateral faces illustrating the directional sectors, and their corresponding stencils for a triangular and quadrilateral faces of a considered cell for a scheme using a fourth-order polynomial.

where L_{int} is the distance between the cell centres of adjacent cells, and $\alpha = 4/3$ similarly to previous approaches [56,57]. The solution is advanced in time by the explicit Strong Stability Preserving (SSP) Runge-Kutta 4th-order method of Spiteri and Ruuth [58] which is stable for $CFL \approx 1.50$, with a constant wave speed estimate during the 5 stages, and a CFL of

1.3 is used for all the test-cases in the present study, unless otherwise stated. For all the test problems involved the initial condition is approximated with a 7th-order Gaussian quadrature rule, to ensure that the initial condition is of high-accuracy, and during the simulation the volume/surface/line integrals are approximated by Gaussian quadrature rule suitable for the order of polynomial employed.

2.5. Implementation

The present algorithms and schemes are developed and implemented in the UCNS3D code [22]. The UCNS3D solver, is written in Fortran 2003 programming language, making use of object oriented programming, including abstract data types. It employs the message passing interface (MPI), and the Open Multi-Processing (OpenMP) application programming interface (API). The Metis partitioner [59] is used to decompose the mesh to numerous partitions; the total number of partitions is equal to the number of MPI processes.

Following previous work done under the PRACE project HOVE [22] it has been revealed that the present code is both CPU and memory bounded, due to the nature of the schemes and implementation employed. Following extensive testing in 5 different HPC facilities it was concluded that the major performance bottlenecks where associated with the WENO weights computation (32-59%), the least squares reconstruction (13-36%) and lastly the extrapolation of the reconstructed solutions at the Gaussian quadrature points (7-14%). Following optimisation of the relevant subroutines by formula rewriting, reduction of operations, inclusion of linear system libraries significant speed-up was achieve ranging from 1.5 to 8.5 with very high-order WENO schemes benefiting the most. The reader is referred to [22] for more details regarding the implementation.

Significant improvements in the computational cost associated with the least square reconstruction can be achieved if the linear dependence of the polynomial reconstruction and evaluation on mean values is pursued similarly to Diot [52]. In particular for the WENO schemes only the linear dependence of the polynomial coefficients on neighbour mean values can be used, since the polynomial coefficients from various stencils are non-linearly combined by WENO algorithm to compute the extrapolated boundary reconstructed solutions at the Gaussian quadrature points. This is something that will be explored in the near future to further speedup the reconstruction process, since the use of the pseudo-inverse permits such operations.

Benchmarking performed on three different systems with similar hardware architecture namely the UK National HPC facility ARCHER (KNL) 10 Intel Knights Landing Xeon Phi 7210 processors with 64 cores each, the Cambridge University Service for Data Driven Discovery (CSD3), on the UK national Tier-2 system Peta 4 system consisting of 342 Intel Knights Landing Xeon Phi 7210 processors with 64 cores each and the Marconi A2 at CINECA (IT) with 3600 nodes of Intel Knights Landing Xeon Phi 7250 processors with 68 cores each, has suggested that the subject platform is beneficial for the nature of the UCNS3D code. The increased memory bandwidth of these hardware platforms provides significantly better performance speed-up 8.5 with respect to the original code in comparison to the Intel Xeon E5 platforms (v1,v2,v3) where a maximum speed-up of approximately 5.2 was obtained. Additionally the configuration that resulted in the best performance was in a quad100 mode, with MPI+OMP, and the OMP SCHEDULE GUIDED for all the parallel loops, using 4 MPI processes (1 per quadrant) and 64 OMP threads per MPI process, with 4 hyper threads as documented in the performance report [60]. The single threaded version of intel MKL was used for all the DGEMM computations of the reconstruction process, and the Intel Fortran compiler version 18.0.1 was used with the following settings `-i4 -r8 -O3 -xmic-avx512 -ipo -fp-model precise -qopenmp -qopenmp-link=static`.

It needs to be highlighted that the total computing time could be masking the influence of important parameters such as of CPU hardware, vectorisation implementation, communication patterns, network bandwidth, memory access patterns etc. The focus of the present study is not concerned in such level of analysis and the reader is referred to [22] for a more detailed algorithmic overview of the processes involved. The present computational times provide an indication of the performance of various algorithms under the same hardware, setup (MPI+OMP, number of threads per MPI task), and compiling options. For the present work all the simulations and the corresponding computational times have been obtained at the CSD3 Peta 4 system.

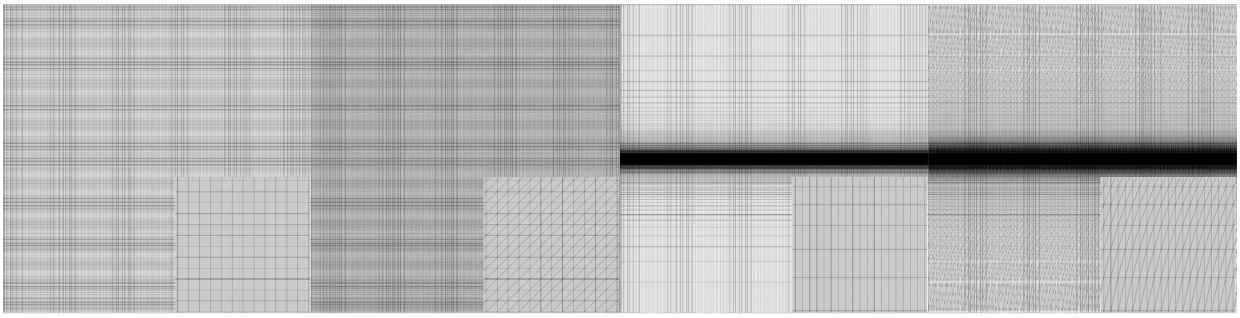
3. Numerical results

3.1. Least squares approximation

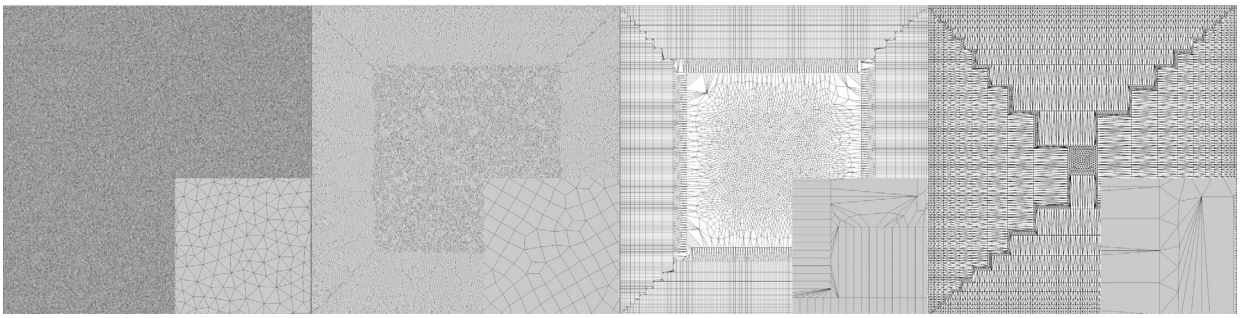
For assessing the quality of the approximation of the least squares reconstruction and the impact of the various stencil selection algorithms a procedure similar to Petrovskaya [61] and Syrakos et al. [62] is pursued. The least squares reconstruction $\tilde{\phi}$ of a polynomial function ϕ of degree equal to the order of the method minus one, leads to some small errors that are good indicators of the matrix approximations and the impact of the stencils selection algorithms. A 2D polynomial function is considered of the following type:

$$\phi(x, y) = (x + y)^{\mathcal{P}}, \tag{26}$$

where $\mathcal{P}$ stands for the polynomial order ($\mathcal{P} = 1$ linear, $\mathcal{P} = 2$ quadratic etc) and a $\mathcal{P} + 1$ order method is assessed through the mean relative errors with respect to the solution and the gradient defined similarly to [61,62] as:



(a) *Grid 1* (b) *Grid 2* (c) *Grid 3* (d) *Grid 4*



(e) *Grid 5* (f) *Grid 6* (g) *Grid 7* (h) *Grid 8*

Fig. 9. Grids used for the least-squares approximation study with a blown-up view of the mesh in the right hand corner.

$$e(x_0, y_0) = \frac{1}{N_c} \sum_{i=1}^{N_c} \frac{|\phi(x_0, y_0)_i - \tilde{\phi}(x_0, y_0)_i|}{|\phi(x_0, y_0)_i|}, \tag{27}$$

and

$$e_{\nabla}(x_0, y_0) = \frac{1}{N_c} \sum_{i=1}^{N_c} \frac{\|\nabla\phi(x_0, y_0)_i - \nabla\tilde{\phi}(x_0, y_0)_i\|}{\|\nabla\phi(x_0, y_0)_i\|}, \tag{28}$$

respectively. N_c being the number of cells in the domain and the gradient norm $\|\nabla\phi(x, y)\|$ being defined as:

$$\|\nabla\phi(x, y)\| = \sqrt{\left(\frac{\partial\phi(x, y)}{\partial x}\right)^2_{|P_0} + \left(\frac{\partial\phi(x, y)}{\partial y}\right)^2_{|P_0}}. \tag{29}$$

The solution point $P_0(x_0, y_0)$ is considered to be the cell-centre. The computational domain considered is given by $[0, 1]^2$ and 8 different types of grids are considered for this study as shown in Fig. 9 with 256 edges per side. For this study the four stencil selection algorithms are assessed, and also the impact of an inverse distance weighted least squares reconstruction is also explored by multiplying both sides of equation (13) with $1/|\vec{r}_M|^2$ where $\vec{r}_M = \vec{r}_M - \vec{r}_0$, $\vec{r}_M$ being the distance of the stencil element M from the considered cell 0.

The obtained results are summarised in Table 1, Table 2, Table 3, Table 4, Table 5 for 2nd-order to 6th-order schemes. It is evident that only for the linear polynomial the approximation of both the gradient and the solution is exact, and for higher orders of polynomials only the solutions exhibit very small errors.

All the algorithms for the linear polynomial exhibit minute differences regarding the weighted and unweighted least squares reconstructions, and the differences become more evident for the *Grid 7, Grid 8*, whereas the weighted least squares provides a better estimate. For the quadratic polynomial it is evident that the FBC algorithm exhibits orders of magnitude larger errors for *Grid 3, Grid 7, Grid 8* with or without weights compared to the other algorithms, while the other algorithms seem to benefit from the weighted least squares mainly for *Grid 7, Grid 8*. As the order of the polynomial and the scheme is increased the weighted least squares provides a marginal improvement on the solution approximation for all grid types and algorithms, but not for the gradient approximation. For linear polynomials and second order schemes the influence of the algorithms is pronounced since the number of the elements in the stencil is considerably smaller compared to higher-order schemes. It can be noticed that for the 2nd-order scheme the VBC and NCB algorithms are marginally better than the others with respect to the errors, and this trend is diminished as the order of accuracy and the polynomial function is increased, since the stencils become significantly larger and hence they do not differ significantly.

Table 1
Mean relative errors for the various central stencil selection algorithms for calculating the solution and the gradient of a linear ($\mathcal{P}=1$) polynomial on a series of grids as shown in Fig. 1 using a 2nd-order reconstruction.

Grid	LSQ	NCB		FBC		VBC		SBC	
	W/U	e	e_∇	e	e_∇	e	e_∇	e	e_∇
1	W	1.0640E-20	3.1502E-16	1.0640E-20	1.5815E-16	1.0640E-20	3.1502E-16	1.0640E-20	3.1453E-16
1	U	1.0640E-20	9.3792E-16	1.0640E-20	1.5815E-16	1.0640E-20	9.3792E-16	1.0640E-20	1.5815E-16
2	W	2.2515E-17	2.3002E-15	2.2515E-17	1.9133E-15	2.2515E-17	2.3002E-15	2.2515E-17	2.3002E-15
2	U	2.2515E-17	1.1380E-15	2.2515E-17	1.9133E-15	2.2515E-17	1.1380E-15	2.2515E-17	1.1380E-15
3	W	6.1511E-17	2.9111E-14	6.1511E-17	2.9190E-14	6.1511E-17	2.9111E-14	6.1511E-17	2.9168E-14
3	U	6.1511E-17	2.9585E-14	6.1511E-17	2.9190E-14	6.1511E-17	2.9585E-14	6.1511E-17	2.9639E-14
4	W	6.0799E-17	5.0646E-14	6.0799E-17	3.1825E-14	6.0799E-17	5.0646E-14	6.0799E-17	5.0646E-14
4	U	6.0799E-17	5.0069E-14	6.0799E-17	3.1825E-14	6.0799E-17	5.0069E-14	6.0799E-17	5.0069E-14
5	W	5.4741E-17	1.3565E-14	5.4741E-17	1.2627E-14	5.4741E-17	1.3565E-14	5.4741E-17	1.3692E-14
5	U	5.4741E-17	1.2805E-14	5.4741E-17	1.2627E-14	5.4741E-17	1.2805E-14	5.4741E-17	1.2884E-14
6	W	8.1188E-17	1.2066E-14	8.1149E-17	1.2282E-14	8.1188E-17	1.2066E-14	8.1140E-17	1.2247E-14
6	U	8.1224E-17	1.1947E-14	8.1149E-17	1.2282E-14	8.1224E-17	1.1947E-14	8.1204E-17	1.2117E-14
7	W	5.7367E-17	4.3377E-15	7.0186E-17	2.3690E-14	5.7367E-17	4.3377E-15	5.7358E-17	4.3701E-15
7	U	7.0087E-17	1.5423E-14	7.0186E-17	2.3690E-14	7.0087E-17	1.5423E-14	7.0102E-17	1.5468E-14
8	W	4.4808E-17	4.5862E-15	4.4808E-17	1.8362E-14	4.4808E-17	4.5862E-15	4.4808E-17	4.6169E-15
8	U	4.4808E-17	1.8523E-14	4.4808E-17	1.8362E-14	4.4808E-17	1.8523E-14	4.4808E-17	1.8680E-14

Table 2
Mean relative errors for the various central stencil selection algorithms for calculating the solution and the gradient of a quadratic ($\mathcal{P}=2$) polynomial on a series of grids as shown in Fig. 1 using a 3rd-order reconstruction.

Grid	LSQ	NCB		FBC		VBC		SBC	
	W/U	e	e_∇	e	e_∇	e	e_∇	e	e_∇
1	W	1.2918E-15	7.0106E-02	1.2919E-15	7.0106E-02	1.2918E-15	7.0106E-02	1.2919E-15	7.0106E-02
1	U	1.2927E-15	7.0106E-02	1.2927E-15	7.0106E-02	1.2927E-15	7.0106E-02	1.2927E-15	7.0106E-02
2	W	1.3061E-15	7.0420E-02	1.3051E-15	7.0420E-02	1.3061E-15	7.0420E-02	1.3061E-15	7.0420E-02
2	U	1.3073E-15	7.0420E-02	1.3073E-15	7.0420E-02	1.3073E-15	7.0420E-02	1.3073E-15	7.0420E-02
3	W	1.2872E-15	5.6015E-02	4.6144E-04	7.7870E-01	1.2872E-15	5.6015E-02	1.2872E-15	5.6015E-02
3	U	1.2871E-15	5.6015E-02	5.5518E-05	2.0492E+00	1.2871E-15	5.6015E-02	1.2871E-15	5.6015E-02
4	W	1.2986E-15	5.6410E-02	1.2988E-15	5.6410E-02	1.2986E-15	5.6410E-02	1.2986E-15	5.6410E-02
4	U	1.2987E-15	5.6410E-02	1.2959E-15	5.6410E-02	1.2987E-15	5.6410E-02	1.2987E-15	5.6410E-02
5	W	1.2877E-15	7.0584E-02	1.2877E-15	7.0584E-02	1.2877E-15	7.0584E-02	1.2877E-15	7.0584E-02
5	U	1.2876E-15	7.0584E-02	1.2877E-15	7.0584E-02	1.2876E-15	7.0584E-02	1.2876E-15	7.0584E-02
6	W	1.2919E-15	6.7704E-02	1.2918E-15	6.7704E-02	1.2919E-15	6.7704E-02	1.2919E-15	6.7704E-02
6	U	1.2919E-15	6.7704E-02	1.2919E-15	6.7704E-02	1.2919E-15	6.7704E-02	1.2919E-15	6.7704E-02
7	W	1.5432E-15	7.1201E-02	2.5434E-04	4.1143E-01	1.5432E-15	7.1201E-02	1.5433E-15	7.1201E-02
7	U	1.2130E-14	7.1201E-02	2.7086E-05	9.5525E-02	1.2130E-14	7.1201E-02	1.2130E-14	7.1201E-02
8	W	1.3212E-15	7.8708E-02	1.5296E-15	7.8708E-02	1.3212E-15	7.8708E-02	1.3209E-15	7.8708E-02
8	U	4.9900E-15	7.8708E-02	5.6759E-15	7.8708E-02	4.9900E-15	7.8708E-02	4.9214E-15	7.8708E-02

Table 3
Mean relative errors for the various central stencil selection algorithms for calculating the solution and the gradient of a cubic ($\mathcal{P}=3$) polynomial on a series of grids as shown in Fig. 1 using a 4th-order reconstruction.

Grid	LSQ	NCB		FBC		VBC		SBC	
	W/U	e	e_∇	e	e_∇	e	e_∇	e	e_∇
1	W	2.1860E-15	1.2794E-01	2.1881E-15	1.2794E-01	2.1860E-15	1.2794E-01	2.1864E-15	1.2794E-01
1	U	2.1812E-15	1.2794E-01	2.1878E-15	1.2794E-01	2.1812E-15	1.2794E-01	2.1828E-15	1.2794E-01
2	W	2.2082E-15	1.2847E-01	2.2082E-15	1.2847E-01	2.2082E-15	1.2847E-01	2.2082E-15	1.2847E-01
2	U	2.2069E-15	1.2847E-01	2.2062E-15	1.2847E-01	2.2069E-15	1.2847E-01	2.2069E-15	1.2847E-01
3	W	2.1800E-15	1.0287E-01	2.4305E-04	6.7213E-01	2.1800E-15	1.0287E-01	2.1800E-15	1.0287E-01
3	U	2.1821E-15	1.0287E-01	1.5773E-04	1.9181E-01	2.1821E-15	1.0287E-01	2.1821E-15	1.0287E-01
4	W	2.2036E-15	1.0355E-01	9.5005E-04	2.4376E+00	2.2036E-15	1.0355E-01	2.2036E-15	1.0355E-01
4	U	2.2215E-15	1.0355E-01	1.3409E-03	1.4946E+00	2.2215E-15	1.0355E-01	2.2215E-15	1.0355E-01
5	W	2.1898E-15	1.2873E-01	2.1898E-15	1.2873E-01	2.1898E-15	1.2873E-01	2.1898E-15	1.2873E-01
5	U	2.1898E-15	1.2873E-01	2.1898E-15	1.2873E-01	2.1898E-15	1.2873E-01	2.1898E-15	1.2873E-01
6	W	2.1893E-15	1.2368E-01	2.1892E-15	1.2368E-01	2.1893E-15	1.2368E-01	2.1893E-15	1.2368E-01
6	U	2.1929E-15	1.2368E-01	2.1974E-15	1.2368E-01	2.1929E-15	1.2368E-01	2.1931E-15	1.2368E-01
7	W	5.8052E-13	1.2930E-01	1.8848E-03	2.9434E-01	5.8052E-13	1.2930E-01	5.8052E-13	1.2930E-01
7	U	6.6583E-11	1.2930E-01	1.6561E-03	3.1559E-01	6.6583E-11	1.2930E-01	6.6583E-11	1.2930E-01
8	W	1.0592E-13	1.4286E-01	2.0123E-03	2.7684E-01	1.0592E-13	1.4286E-01	1.0544E-13	1.4286E-01
8	U	1.4110E-11	1.4286E-01	2.3645E-03	3.9101E-01	1.4110E-11	1.4286E-01	1.4092E-11	1.4286E-01

Table 4
Mean relative errors for the various central stencil selection algorithms for calculating the solution and the gradient of a quartic ($\mathcal{P}=4$) polynomial on a series of grids as shown in Fig. 1 using a 5th-order reconstruction.

Grid	LSQ	NCB		FBC		VBC		SBC	
	W/U	e	e_∇	e	e_∇	e	e_∇	e	e_∇
1	W	1.8042E-15	2.0803E-01	1.8093E-15	2.0803E-01	1.8042E-15	2.0803E-01	1.8076E-15	2.0803E-01
1	U	1.7133E-15	2.0803E-01	1.7966E-15	2.0803E-01	1.7133E-15	2.0803E-01	1.7369E-15	2.0803E-01
2	W	1.8376E-15	2.0873E-01	1.8356E-15	2.0873E-01	1.8376E-15	2.0873E-01	1.8376E-15	2.0873E-01
2	U	1.8290E-15	2.0873E-01	1.8418E-15	2.0873E-01	1.8290E-15	2.0873E-01	1.8290E-15	2.0873E-01
3	W	1.7758E-15	1.4714E-01	2.8472E-03	1.5501E+00	1.7758E-15	1.4714E-01	1.7758E-15	1.4714E-01
3	U	2.0158E-15	1.4714E-01	1.1631E-03	4.3677E-01	2.0158E-15	1.4714E-01	2.0174E-15	1.4714E-01
4	W	1.8219E-15	1.4817E-01	8.4654E-04	3.4397E-01	1.8219E-15	1.4817E-01	1.8219E-15	1.4817E-01
4	U	2.0503E-15	1.4817E-01	4.9884E-04	2.0191E-01	2.0503E-15	1.4817E-01	2.0503E-15	1.4817E-01
5	W	1.8130E-15	2.0909E-01	1.8132E-15	2.0909E-01	1.8130E-15	2.0909E-01	1.8128E-15	2.0909E-01
5	U	1.8185E-15	2.0909E-01	1.8201E-15	2.0909E-01	1.8185E-15	2.0909E-01	1.8188E-15	2.0909E-01
6	W	1.8311E-15	2.0187E-01	1.8339E-15	2.0187E-01	1.8311E-15	2.0187E-01	1.8315E-15	2.0187E-01
6	U	2.0750E-15	2.0187E-01	1.9133E-15	2.0187E-01	2.0750E-15	2.0187E-01	2.0679E-15	2.0187E-01
7	W	1.6439E-09	2.0884E-01	1.5640E-03	3.7262E-01	1.6439E-09	2.0884E-01	1.6439E-09	2.0884E-01
7	U	3.9829E-09	2.0885E-01	2.5027E-03	6.0272E-01	3.9829E-09	2.0885E-01	3.9829E-09	2.0885E-01
8	W	4.9136E-12	2.3044E-01	2.4829E-03	4.9882E-01	4.9136E-12	2.3044E-01	4.9295E-12	2.3044E-01
8	U	3.9701E-10	2.3044E-01	6.2611E-04	6.0950E-01	3.9701E-10	2.3044E-01	3.9702E-10	2.3044E-01

Table 5
Mean relative errors for the various central stencil selection algorithms for calculating the solution and the gradient of a quintic ($\mathcal{P}=5$) polynomial on a series of grids as shown in Fig. 1 using a 6th-order reconstruction.

Grid	LSQ	NCB		FBC		VBC		SBC	
	W/U	e	e_∇	e	e_∇	e	e_∇	e	e_∇
1	W	2.3505E-16	2.4556E-01	2.7572E-16	2.4556E-01	2.3505E-16	2.4556E-01	2.7779E-16	2.4556E-01
1	U	5.0611E-16	2.4556E-01	5.7609E-16	2.4556E-01	5.0611E-16	2.4556E-01	5.5044E-16	2.4556E-01
2	W	3.2102E-16	2.4645E-01	3.3091E-16	2.4645E-01	3.2102E-16	2.4645E-01	3.2102E-16	2.4645E-01
2	U	6.6941E-16	2.4645E-01	8.9724E-16	2.4645E-01	6.6941E-16	2.4645E-01	6.6941E-16	2.4645E-01
3	W	2.2236E-15	1.7646E-01	4.9766E-03	1.7615E+00	2.2236E-15	1.7646E-01	1.6063E-15	1.7646E-01
3	U	2.2531E-14	1.7646E-01	1.2580E-02	1.3384E+00	2.2531E-14	1.7646E-01	2.0451E-14	1.7646E-01
4	W	1.4792E-15	1.7766E-01	7.3896E-04	3.8524E-01	1.4792E-15	1.7766E-01	1.4792E-15	1.7766E-01
4	U	6.9944E-15	1.7766E-01	9.7157E-04	5.0577E-01	6.9944E-15	1.7766E-01	6.9944E-15	1.7766E-01
5	W	3.2363E-16	2.4683E-01	3.2219E-16	2.4683E-01	3.2363E-16	2.4683E-01	3.1891E-16	2.4683E-01
5	U	5.9915E-16	2.4683E-01	5.2399E-16	2.4683E-01	5.9915E-16	2.4683E-01	5.6858E-16	2.4683E-01
6	W	1.5745E-15	2.3804E-01	1.0439E-15	2.3804E-01	1.5745E-15	2.3804E-01	1.7040E-15	2.3804E-01
6	U	1.0120E-14	2.3804E-01	1.8096E-14	2.3804E-01	1.0120E-14	2.3804E-01	1.0893E-14	2.3804E-01
7	W	8.5194E-09	2.4689E-01	7.9739E-03	1.4152E+00	8.5194E-09	2.4689E-01	8.5194E-09	2.4689E-01
7	U	1.9124E-08	2.4691E-01	6.5781E-02	6.4546E+00	1.9124E-08	2.4691E-01	1.9124E-08	2.4691E-01
8	W	2.0976E-10	2.7303E-01	1.0908E-03	3.4390E-01	2.0976E-10	2.7303E-01	2.0963E-10	2.7303E-01
8	U	1.5072E-08	2.7302E-01	6.3635E-03	9.0821E-01	1.5072E-08	2.7302E-01	1.5051E-08	2.7302E-01

It needs to be stressed that exploring the performance of various stencil algorithms on all possible types of meshes, is an open-ended task where an overwhelming number of algorithms can be developed, but one must consider the main use of the targeted algorithms in terms of types of problems and constrain the study to the algorithms that make sense, and can provide robustness and accuracy at the same time. For this reason there have been comprehensive studies in the past focusing on exploring various combinations of gradient approximations [62–65], including weighted and unweighted least squares, cell-based and node-based, augmented stencil mappings etc, however their main focus was on schemes up to 2nd-order accurate. Therefore as seen in this test-problem the differences in stencil construction algorithms are more pronounced for up to second-order schemes, as seen between a NCB and VBC in Fig. 2 and Fig. 3 respectively. Consequently for Reynolds-Averaged Navier-Stokes (RANS) equations where very high-aspect ratio cells can be encountered the stencil selection algorithm used can influence dramatically the approximation of the gradients of the turbulence model in the close vicinity of the wall. In these circumstances, measures of grid quality can be used to switch to different stencil algorithms in these regions, or even other measures for suppressing odd-even decoupling modes in the numerical solutions generated by the gradients approximation such as additionally penalty terms as initially suggested by Gassner et al. [55] and being successfully applied to various test problems by other researchers [55–57].

However very high-order WENO schemes on unstructured meshes present a completely different landscape in terms of grid quality requirements, and grids with qualities of *Grid 7* and *Grid 8* are rarely expected to be encountered for these schemes in the context of LES or implicit *i*LES of real-life problems, where these schemes are mainly targeted for. This does not imply that the WENO schemes cannot be successfully employed in the context of RANS equations, but there are no significant benefits in applying a higher than 3rd-order WENO scheme for RANS on 3D unstructured meshes as a series of previous studies have concluded [3,38,39]. Within the context of this study the inverse weighted least squares reconstruction

translates in an additional memory storage requirement for the right hand side of equation (13) without significant benefits for the grids involved except *Grid 7* and *Grid 8*, and is therefore not considered furthermore in this study.

3.2. Linear advection equation

The 2D linear advection equation of a smooth test function $U(x, y, 0) = sin(2\pi x) \cdot sin(2\pi y)$ is solved on computational domain given by $[0, 1]^2$ with periodic boundary conditions applied on all sides. Initially the result obtained with the various central stencils algorithms are explored for good quality and bad quality meshes. They are then succeeded by the results obtained from the various directional stencil algorithms. Four different mesh types are used including uniform and non-uniform triangular meshes, quadrilateral and hybrid meshes at resolutions corresponding to $16, 32, 64$ and 128 edges per side. The computations are run for one period $t = 1$ where and the e_{L^2} error is computed as follows:

$$e_{L^2} = \sqrt{\frac{\sum_i \int_{\Omega_i} \left(U_e\left(x, t_f\right) - U_c\left(x, t_f\right)\right)^2 dV}{\sum_i |\Omega_i|}}, \tag{30}$$

where $U_c\left(x, t_f\right)$ and $U_e\left(x, t_f\right)$ are the computed and exact solutions at the end of the simulation $t = 1$. The exact solution $U_e\left(x, t_f\right)$ being given by the initial condition itself at $t = 0$.

3.2.1. Central stencils

On the pursue for the most compact stencil algorithm and the associated benefits in terms of reduced computational cost, increased accuracy etc., the impact of the expansion factor $EX = M/K$ where M is the total number of elements in the stencil and K being the total number of polynomial coefficients/degrees of freedom is investigated as well. Ideally an algorithm that can consistently produce accurate results for various types of meshes with lower expansion factor EX values would be preferred. By varying EX from 1.6 to 2 up to 3, it can be seen from Fig. 10, Fig. 11, Fig. 12 and Fig. 13 that no scheme can consistently produce accurate results for all the selected meshes and resolutions for an $EX = 1.6$. On the other hand all the algorithms are robust when an $EX = 2$ and higher is used as supported by previous studies [2,11,35–37,50,51]. As expected the lower the EX value the better the accuracy of the scheme, with a minor reduction in total computational time. It needs to be stressed that for the present 2D test problem, and at the subject resolutions significant differences in terms of computational times are not expected.

It is important to realize that for the subject test problem the meshes used are of good quality, and all the algorithms perform quite similarly producing the expected results and achieving convergence rates close to their theoretical ones. The primary reason that the subject results include schemes only up to fourth-order polynomials is that for higher spatial-order of accuracy the time-step needs to be further reduced to impractical levels in order to achieve higher than 4th-order of accuracy. For this reason the results obtained with such a time-step size restriction for only the SBC algorithm are shown in Fig. 14, where it can be noticed that the errors are significantly reduced for higher polynomial orders.

By keeping the stencil expansion factor fixed at $EX = 2$, which is the most prevalent choice amongst many similar approaches [2,11,35–37,50,51] , and employing a triangular mesh that consists of high-aspect ratio cells as seen in Fig. 16, the robustness of the algorithms at regions of high-aspect ratio cells is explored. Firstly from the central stencils obtained from all the algorithms as shown in Fig. 15 the differences between them becomes more apparent. The NCB selects significant number of cells farther away from the considered cell, the VBC algorithm includes all the vertex neighbours of the considered cell which are significantly more than the ones that emerge from a good quality mesh, and the FBC stencil is the most compact of all since it adds only the direct-side neighbours and then applies the compactness procedure to the rest of the elements. Finally the SBC algorithm combines some of the features of the VBC and NCB algorithm.

Remark 3. There are two striking features that need to be commented in the stencils algorithm when meshes of bad quality are experienced. The first one is that the Cell Quality criterion might not be sufficient since it can not identify regions of transition from good quality to bad quality meshes as seen in this example. Since the considered cell i indicated by red colour is having good quality characteristics. Therefore a more representative quality criterion would be to include a region in the vicinity of the considered cell in order to perform some quality metrics on all the enclosed cells, and based on the findings to activate the most suitable algorithm. This strategy however would loose the local nature of the algorithm, will require substantially more computational effort and might not be suitable for mesh adapting strategies. The second feature being that for realistic meshes for viscous flows that include higher-aspect ratio cells close to the walls, this transition regions will exist and therefore robust central stencil selection algorithms that are as compact as possible are needed in these regions.

It can be seen from the obtained results of Table 6 that the early activation of the compactness procedure of the FBC algorithm leads to a complete failure, and the only algorithm that consistently produces correct results (although not expected to reach the theoretical order of accuracy for such meshes) without crashing is the SBC algorithm. Even the NCB can not produce the correct results for all the resolutions, since a higher stencil expansion factor is required to do so. For the

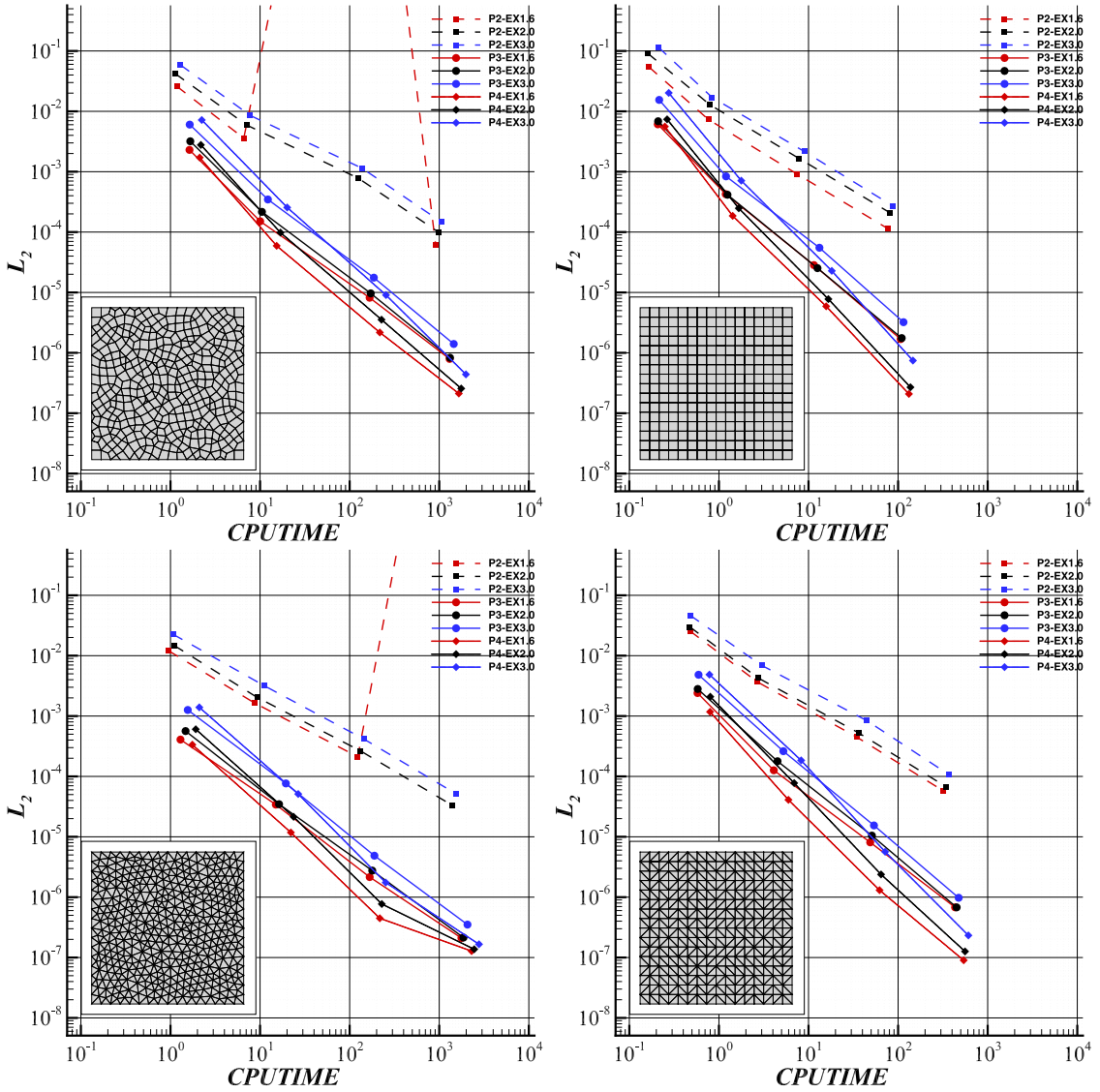


Fig. 10. Plots of $\mathcal{L}_2$ error vs. CPU time measured in (number of cores x seconds) for the linear advection equation with smooth initial conditions at $t = 1.0$. The influence of various polynomial orders $\mathcal{P}$, stencil expansion factors (ranging from 1.6 to 3.0), grid types (quadrilateral and triangular), and grid resolution (16, 32, 64, 128 edges per side) using the Naive Cell based algorithm is explored.

VBC algorithm the compactness procedure is activated quite early for low-number of the vertex-neighbours of the considered cell and as a result the added elements might be aligned with a direction that will eventually impair the robustness of the least-square reconstruction. The robustness of the SBC algorithm is characterised by a slightly increased run time compared to NCB algorithm for both spatial orders examined. Looking closer at the differences of the algorithms in terms of their width w_s seen in Fig. 16 as defined in the previous section, it can be noticed that all the algorithms except SBC have high values of stencil width close to the transition between good quality and bad quality meshes. They reach the maximum value of 12 which is equivalent to central stencils as wide as 12 cells in each direction for a fourth-order polynomial. On the other hand the widest stencil with the SBC algorithm is 7 cells wide, and more uniform-favourable distribution of the width of the stencils throughout the domain is achieved. Therefore from this point onwards the SBC algorithm is going to be used in conjunction with the WENO schemes, unless otherwise specified.

3.2.2. Directional stencils

For investigating the influence of the directional stencil algorithms for the subject test problem a stencil expansion factor of $EX = 2$ is used along with the SBC central stencil algorithm. From the obtained results as seen in Fig. 17 all the directional stencil algorithm produce the same errors as expected, since this is a smooth problem and the central stencil gets assigned the largest non-linear weight hence it has the greatest influence in the obtained results. However the

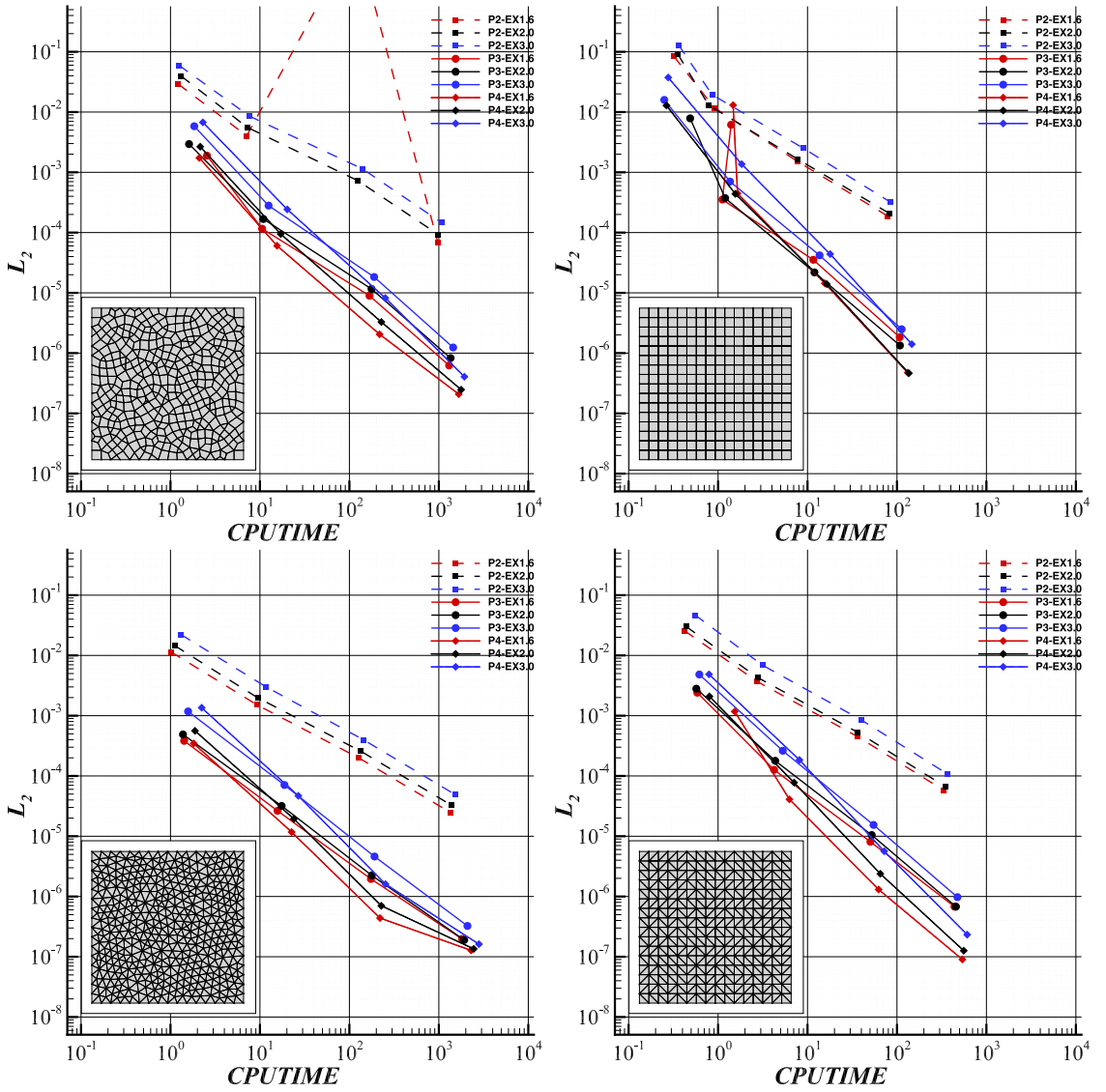


Fig. 11. Plots of $\mathcal{L}_2$ error vs. CPU time measured in (number of cores x seconds) for the linear advection equation with smooth initial conditions at $t = 1.0$. The influence of various polynomial orders $\mathcal{P}$, stencil expansion factors (ranging from 1.6 to 3.0), grid types (quadrilateral and triangular), and grid resolution (16, 32, 64, 128 edges per side) using the Vertex based Compact Algorithm is explored.

Table 6
L^2 error, convergence rate $\mathcal{O}_{L^2}$ and CPU time ($cores \times time(sec)$) for the 2D linear advection equation with smooth initial conditions test problem obtained with various orders of accuracy with the NCB, VBC, FBC, and SBC central stencil algorithms at the final time of $t = 1$ on bad quality meshes. It can be noticed that the only algorithm that is robust enough to complete the simulation is the SBC.

Mesh Elements	NCB $\mathcal{O}=3$			VBC $\mathcal{O}=3$			FBC $\mathcal{O}=3$			SBC $\mathcal{O}=3$		
	e_{L^2}	$\mathcal{O}_{L^2}$	CPU	e_{L^2}	$\mathcal{O}_{L^2}$	CPU	e_{L^2}	$\mathcal{O}_{L^2}$	CPU	e_{L^2}	$\mathcal{O}_{L^2}$	CPU
1368	1.634E-02	–	7.41	1.656E-02	–	5.16	NaN	–	7.84	1.668E-02	–	7.58
3816	NaN	–	60.38	NaN	–	50.21	NaN	–	64.12	1.115E-02	0.58	67.16
10328	NaN	–	392.22	NaN	–	396.35	NaN	–	399.22	1.003E-02	0.15	406.61
Mesh Elements	NCB $\mathcal{O}=5$			VBC $\mathcal{O}=5$			FBC $\mathcal{O}=5$			SBC $\mathcal{O}=5$		
	e_{L^2}	$\mathcal{O}_{L^2}$	CPU	e_{L^2}	$\mathcal{O}_{L^2}$	CPU	e_{L^2}	$\mathcal{O}_{L^2}$	CPU	e_{L^2}	$\mathcal{O}_{L^2}$	CPU
1368	6.952E-04	–	10.98	6.937E-04	–	12.15	NaN	–	15.52	6.792E-04	–	15.41
3816	3.381E-04	1.03	100.35	3.379E-04	1.03	96.82	NaN	–	112.53	3.269E-04	1.05	109.71
10328	NaN	–	613.80	NaN	–	644.99	NaN	–	618.34	4.322E-04	-0.40	626.98

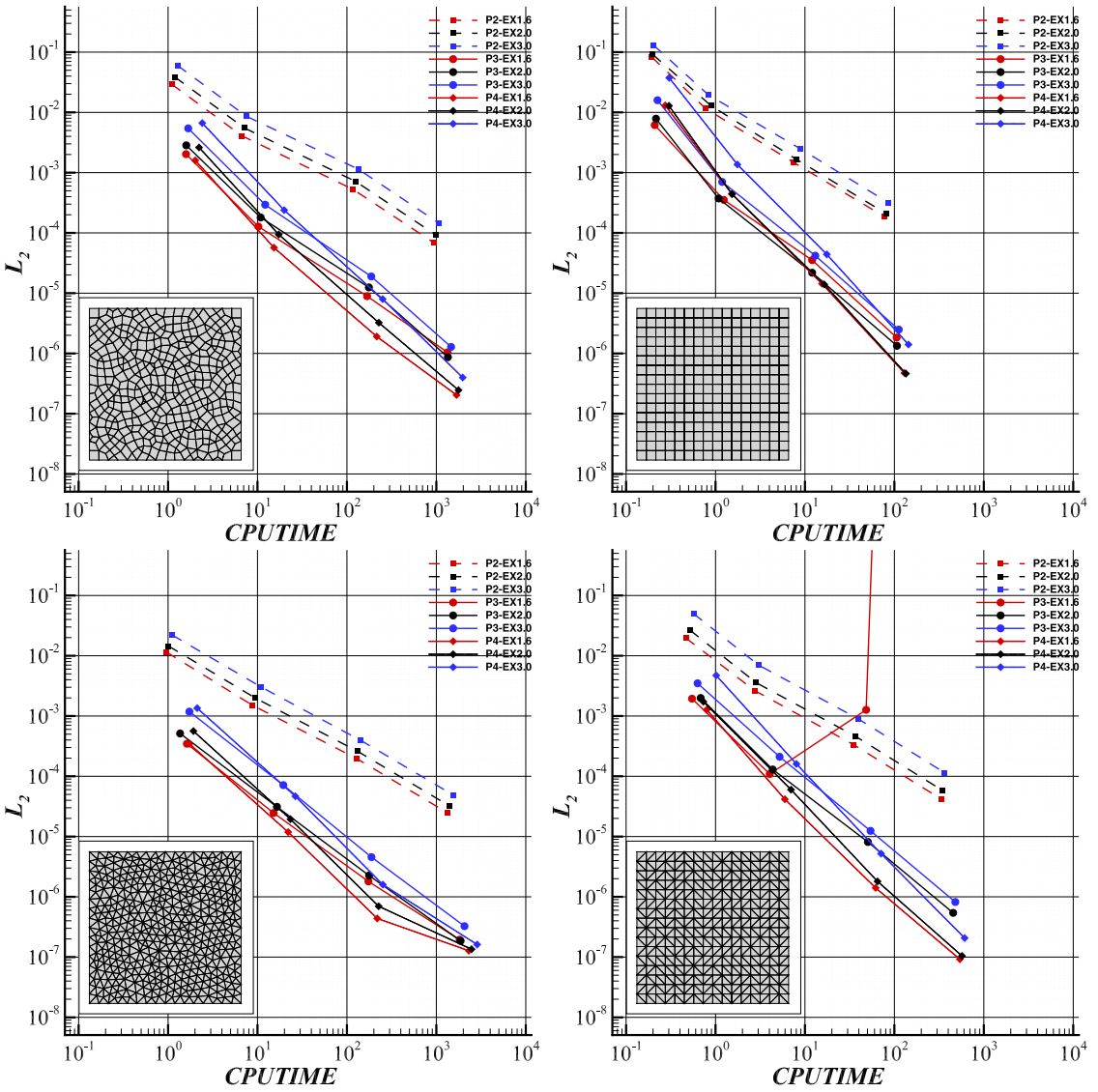


Fig. 12. Plots of $\mathcal{L}_2$ error vs. CPU time measured in (number of cores x seconds) for the linear advection equation with smooth initial conditions at $t = 1.0$. The influence of various polynomial orders $\mathcal{P}$, stencil expansion factors (ranging from 1.6 to 3.0), grid types (quadrilateral and triangular), and grid resolution (16, 32, 64, 128 edges per side) using the Face based Compact Algorithm is explored.

computational cost of the algorithms differs significantly where *Type 4* on quadrilateral meshes is 3 times faster than *Type 1*, and for triangular meshes *Type 3* is approximately 2 times faster than *Type 1* for the same order of accuracy. These trends should be treated with caution since for more complicated test problems the number of variables, and number of operations will increase significantly as highlighted in Table 7.

3.3. 2D solid body rotation

The solid body rotation test of Leveque [66] is employed to assess the impact of the directional stencil algorithms on the correct and accurate revolution of a body. The continuity equation is considered as follows:

$$\frac{\partial U}{\partial t} + \nabla \cdot (\mathbf{v}U) = 0, \tag{31}$$

on a computational domain given by $[0, 1]^2$ with periodic boundary conditions applied on all sides. The velocity $\mathbf{v}$ describing the revolution of the profile around the centre of the domain $(0.5, 0.5)$ is given by:

$$\mathbf{v}(x, y) = (0.5 - y, x - 0.5). \tag{32}$$

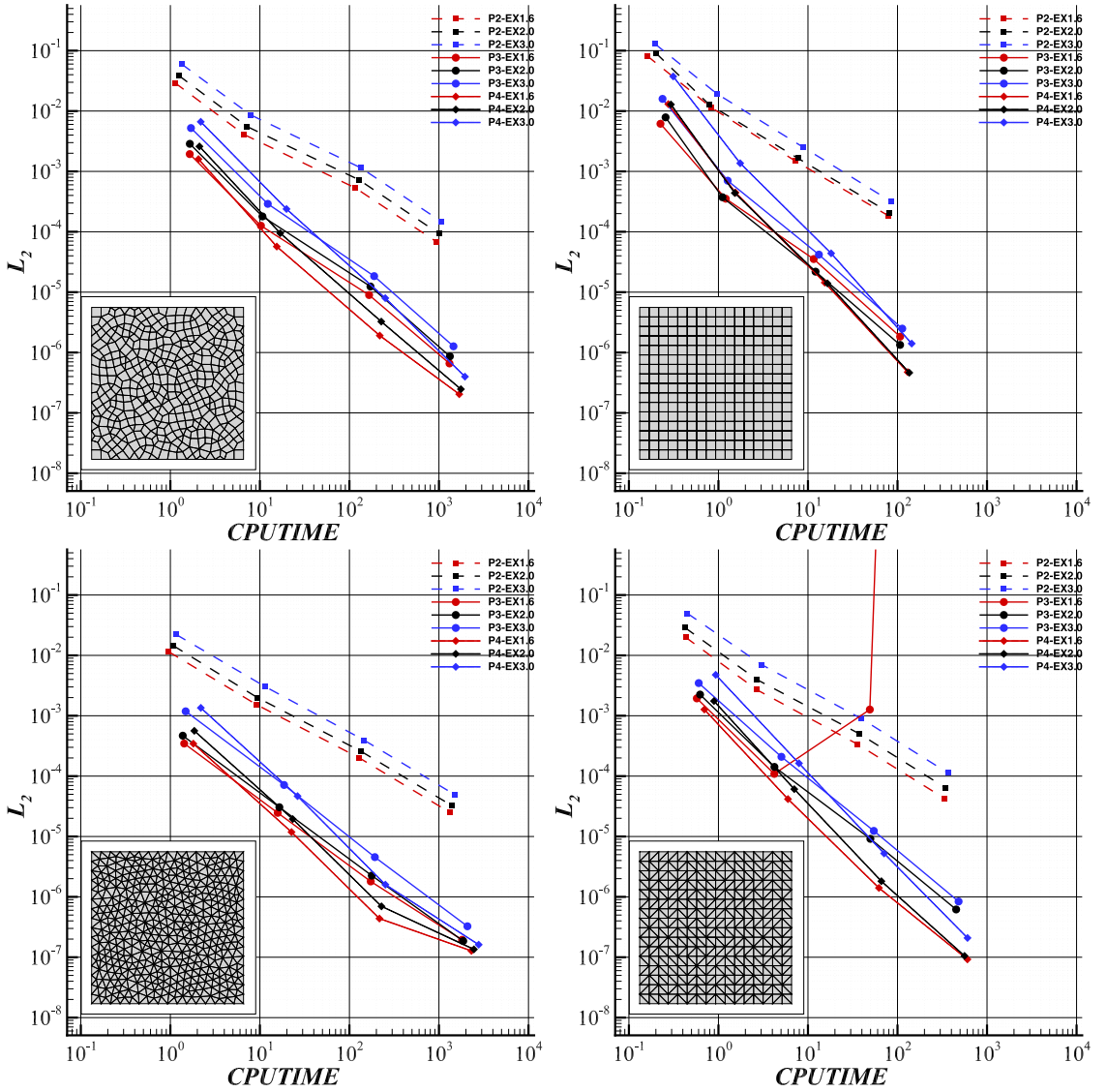


Fig. 13. Plots of $\mathcal{L}_2$ error vs. CPU time measured in (number of cores x seconds) for the linear advection equation with smooth initial conditions at $t = 1.0$. The influence of various polynomial orders $\mathcal{P}$, stencil expansion factors (ranging from 1.6 to 3.0), grid types (quadrilateral and triangular), and grid resolution (16, 32, 64, 128 edges per side) using the Stencil based Compact Algorithm is explored.

Table 7
Number of entries, for all the directional stencils reconstruction matrices A_{mk} per flow variable for 3D meshes with WENO schemes and various directional stencil algorithms on hexahedral meshes.

Order	Type 1	Type 2	Type 3	Type 4
$\mathcal{P} = 1$	216	108	108	54
$\mathcal{P} = 2$	1944	972	972	486
$\mathcal{P} = 3$	8664	4332	4332	2166
$\mathcal{P} = 4$	27744	13872	13872	13872
$\mathcal{P} = 5$	72600	36300	36300	18150
$\mathcal{P} = 6$	165336	82668	82668	41334
$\mathcal{P} = 7$	339864	169932	169932	84966
$\mathcal{P} = 8$	645504	322752	322752	161376

In this particular test three bodies are considered namely a smooth centered at $(x_0 = 0.25, y_0 = 0.5)$, a sharp cone centered at $(x_0 = 0.5, y_0 = 0.25)$ and a slotted cylinder centered at $(x_0 = 0.5, y_0 = 0.75)$ described by the following functions respectively:

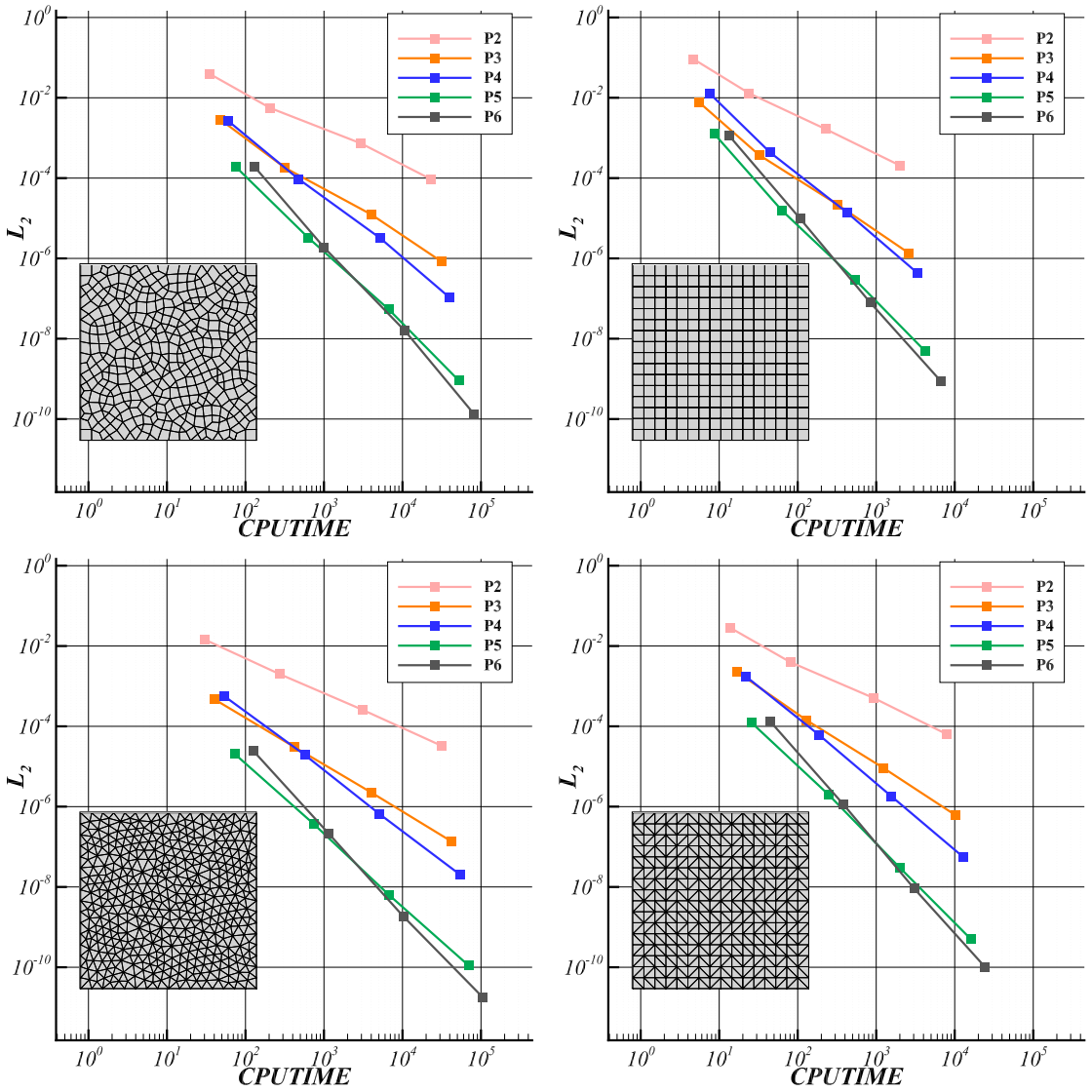
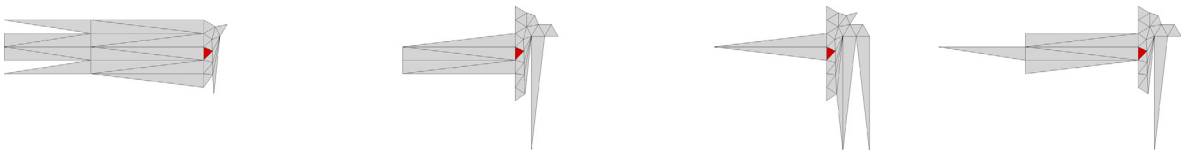


Fig. 14. Plots of $\mathcal{L}_2$ error vs. CPU time measured in (number of cores x seconds) for the linear advection equation with smooth initial conditions at $t = 1.0$ with a CFL=0.01 and a stencil expansion factor of 2. The influence of various polynomial orders $\mathcal{P}$, grid types (quadrilateral and triangular), and grid resolution (16, 32, 64, 128 edges per side) using the Stencil based Compact Algorithm is explored.



(a) NCB (b) VBC (c) FBC (d) SBC

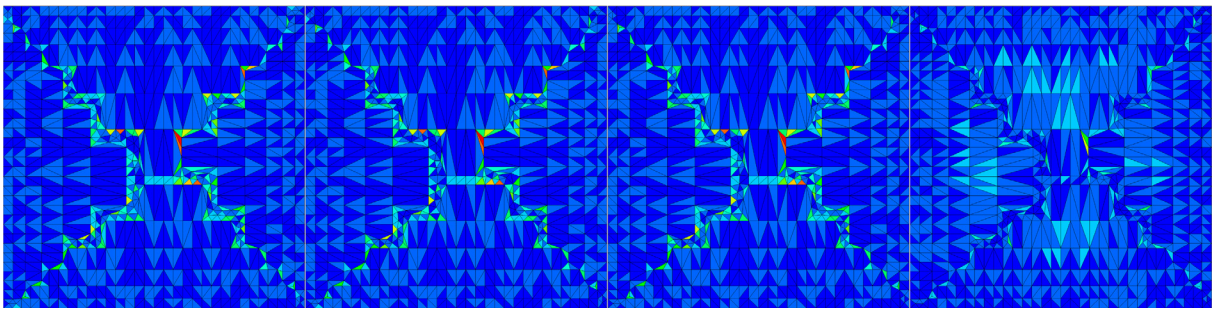
Fig. 15. Examples of central stencils with the considered cell in red colour using various stencil algorithms for meshes of poor quality.

$$\mathbf{f}(x, y) = \frac{1 + cos(\pi r(x, y))}{4}, \tag{33}$$

$$\mathbf{f}(x, y) = 1 - r(x, y), \tag{34}$$

and

$$\mathbf{f}(x, y) = \begin{cases} 1, & \text{if } |x - x_0| \geq 0.025 \text{ or } y \geq 0.85 \\ 0, & \text{if otherwise.} \end{cases} \tag{35}$$



(a) NCB (b) VBC (c) FBC (d) SBC

Fig. 16. Contour plots of the variation of the stencil width parameter w_s (20 equidistant levels used from 1.14 to 12) for various stencil selection algorithms using a $\mathcal{P} = 4$ polynomial for a mesh of poor quality.

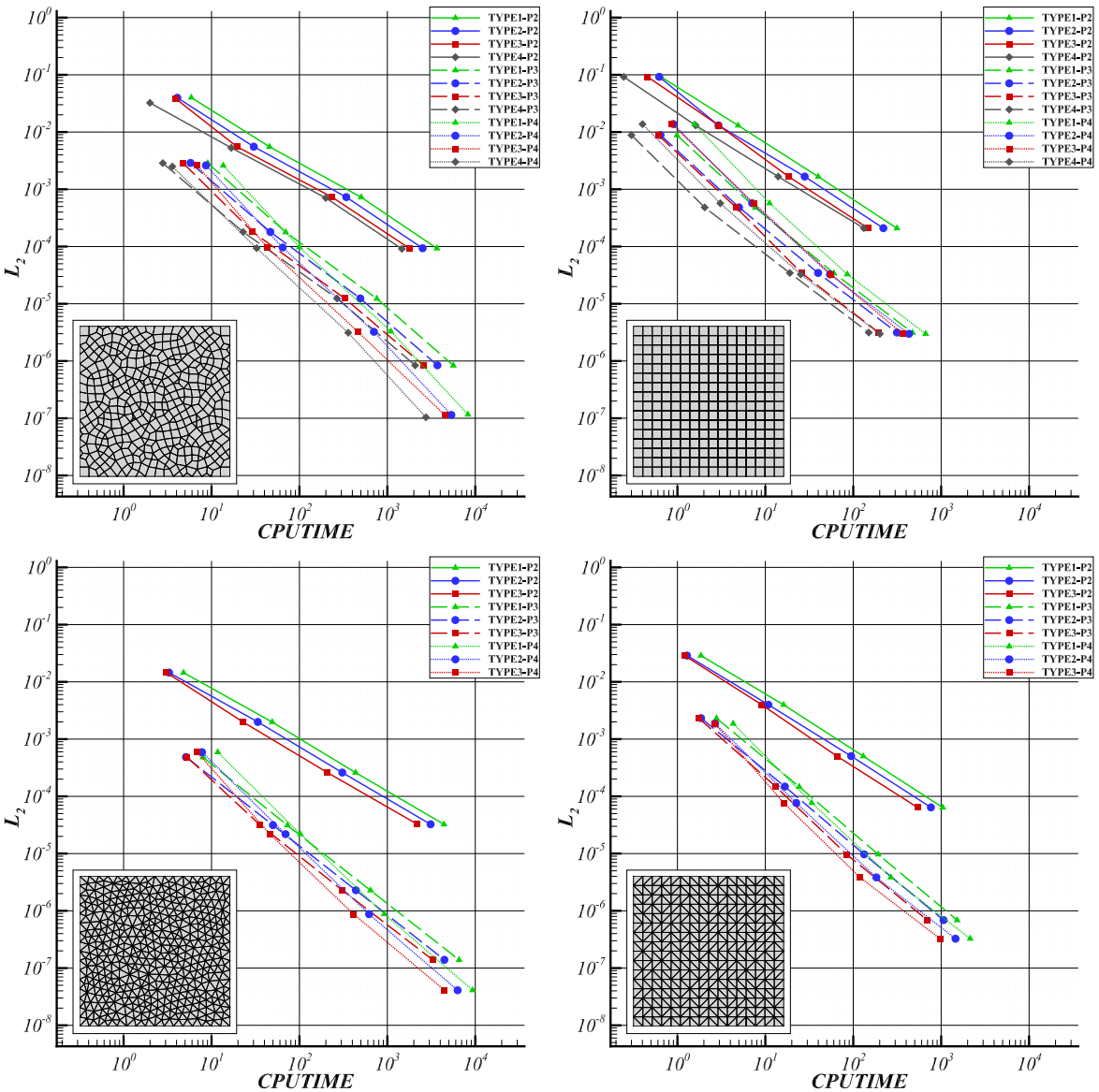
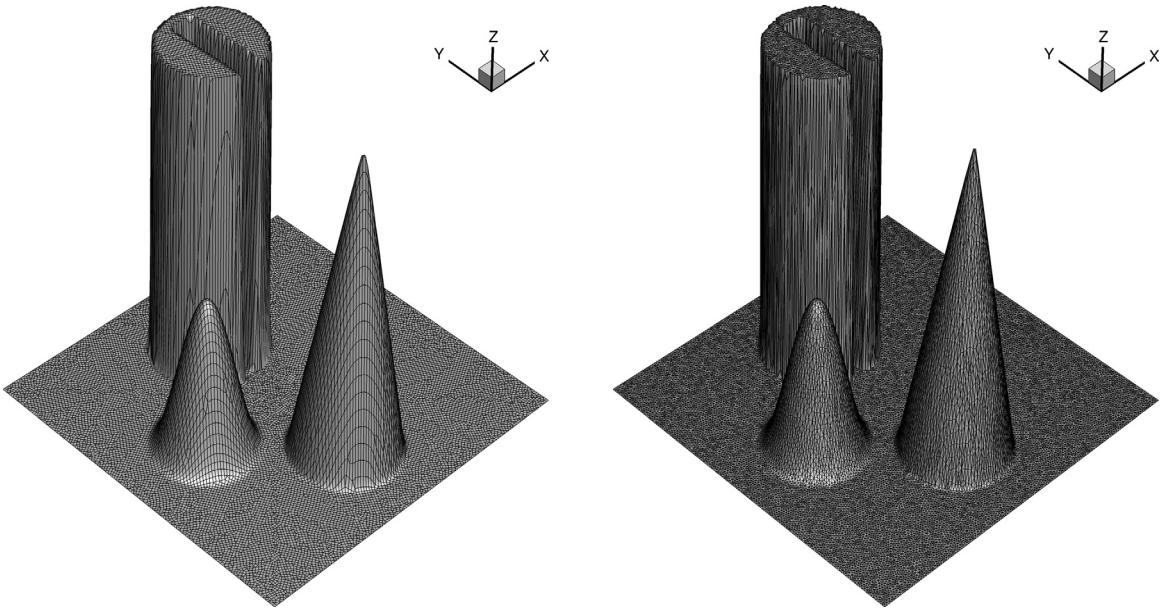


Fig. 17. Plots of $\mathcal{L}_2$ error vs. CPU time measured in (number of cores x seconds) for the linear advection equation with smooth initial conditions at $t = 1.0$. The influence of various polynomial orders $\mathcal{P}$, grid types (quadrilateral and triangular), grid resolution (16, 32, 64, 128 edges per side) and directional stencil selection algorithms are explored.



(a) *Mixed-Element mesh* (b) *Triangular mesh*

Fig. 18. Initial solution for the solid body rotation test problem, using a mixed-element and triangular mesh with 128 edges per side.

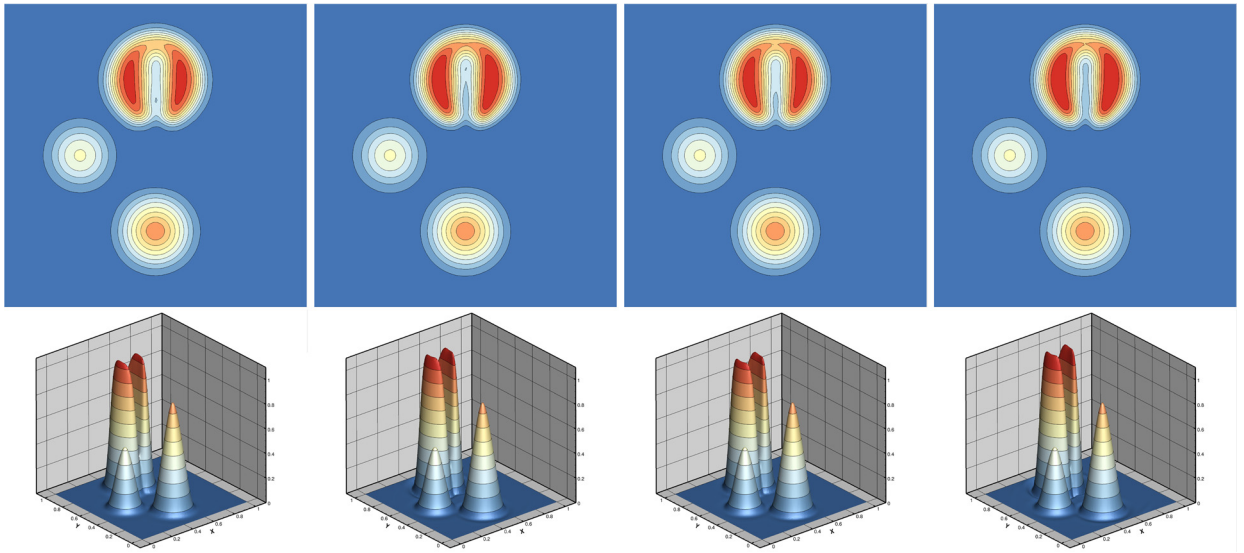


Fig. 19. Computed results after one revolution ($t = 2\pi$) of the solid body rotation test problem, using a mixed-element mesh with 128 edges per side and a WENO 3 scheme with various directional stencil selection algorithms (from left to right *Type 1, Type 2, Type 3, Type 4*). Three-dimensional elevations (bottom) and top views (top) of ten uniformly distributed isolines from 0.1 to 1.0. It can be noticed that all directional stencil algorithms capture the correct shapes without any oscillations, and the *Type 1* being the most dissipative since there are three isolines outside the slotted cylinder, while in the other types there are only two.

Where the normalised distance from the centres (x_0, y_0) is given by:

$$r(x, y) = (1/r_0)\sqrt{(x - x_0)^2 + (y - y_0)^2}, \tag{36}$$

with $r_0 = 0.15$. The rest of the domain the solution is initialised with zero, and after one full revolution $t_f = 2\pi$ the exact solution coincides with the initial solution as shown in Fig. 18. Two unstructured meshes are used a mixed-element one and a triangular one with 128 edges per side of the computational domain.

From the obtained results using the WENO 3 scheme as shown in Fig. 19 and Fig. 21, it is evident that the *Type 1* algorithm is the most dissipative one, since the number of isolines outside the slot is greater than the other types since the exact solution isolines would fit the slot shape. This is to be expected since it has the largest number of directional stencils

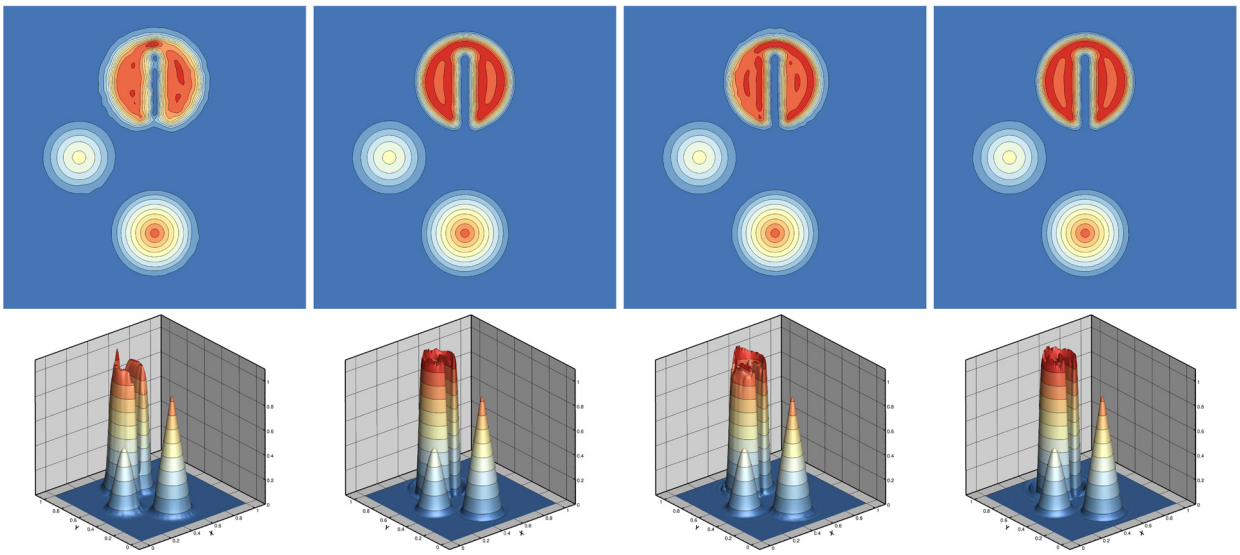


Fig. 20. Computed results after one revolution ($t = 2\pi$) of the solid body rotation test problem, using a mixed-element mesh with 128 edges per side and a WENO 6 scheme with various directional stencil selection algorithms (from left to right *Type 1, Type 2, Type 3, Type 4*). Three-dimensional elevations (bottom) and top views (top) of ten uniformly distributed isolines from 0.1 to 1.0. It can be noticed that all directional stencil algorithms capture the correct shapes but oscillations in the solutions are present. The *Type 2* and *Type 4* symmetric directional algorithms exhibit more symmetric solution profiles of the slotted cylinder as seen in the top views, whereas the non-symmetric ones *Type 1* and *Type 3* do not. The *Type 1* being the most dissipative since there are two isolines outside the slotted cylinder, while in the other types there are none.

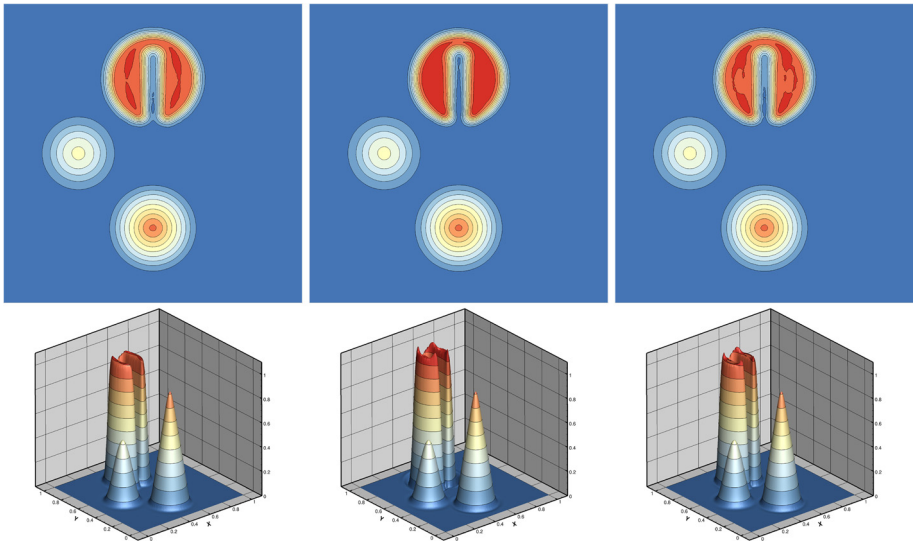


Fig. 21. Computed results after one revolution ($t = 2\pi$) of the solid body rotation test problem, using a triangular mesh with 128 edges per side and a WENO 3 scheme with various directional stencil selection algorithms (from left to right *Type 1, Type 2, Type 3*). Three-dimensional elevations (bottom) and top views (top) of ten uniformly distributed isolines from 0.1 to 1.0. It can be noticed that all directional stencil algorithms capture the correct shapes but some oscillations are present in particularly for the *Type 2* algorithm as seen in the three-dimensional elevations, and the *Type 1* being the most dissipative since there is one isoline outside the slotted cylinder, while in the other types there are none.

and the largest width (smallest compactness) compared to the other ones. For the triangular mesh as shown in Fig. 21 it can be noticed that the results are characterised by some oscillations being more pronounced for the *Type 2* algorithm, compared to the other ones due to the symmetric nature of this algorithm.

Now when switching to the WENO 6 scheme as shown in Fig. 20 and Fig. 22 although the accuracy regarding representing the slot of the slotted cylinder is improved for all types compared to the WENO 3 scheme, the computed solutions exhibit oscillations. These ones are more pronounced for the mixed-element mesh due to the combination of the high-order of accuracy, the coarse grid resolution that translates to having (7-8 cells across the slot) when the stencil sizes have grown compared to the WENO 3 schemes and none of them is being compact enough to fit within the slot section of the cylinder.

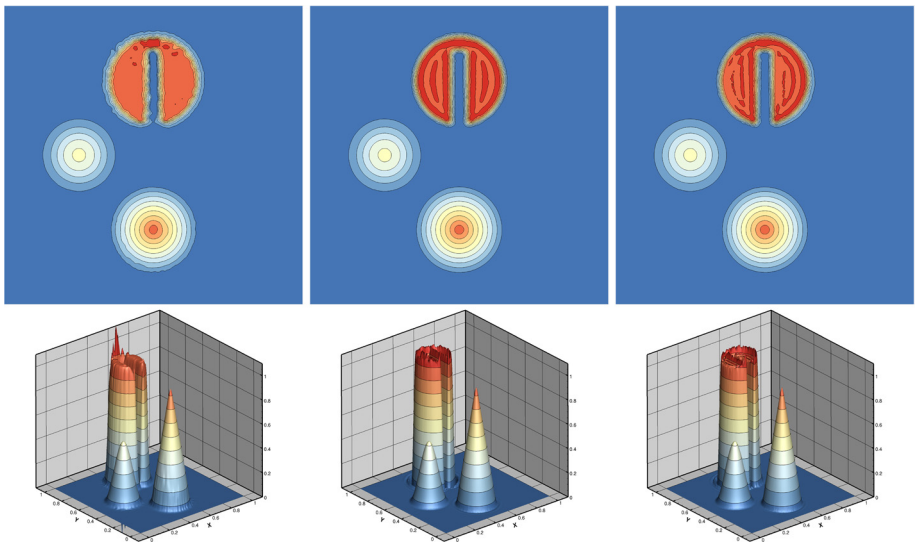


Fig. 22. Computed results after one revolution ($t = 2\pi$) of the solid body rotation test problem, using a mixed-element mesh with 128 edges per side and a WENO 6 scheme with various directional stencil selection algorithms (from left to right *Type 1, Type 2, Type 3*). Three-dimensional elevations (bottom) and top views (top) of ten uniformly distributed isolines from 0.1 to 1.0. It can be noticed that all directional stencil algorithms capture the correct shapes but oscillations in the solutions are present. The *Type 2* symmetric directional algorithm exhibit more symmetric solution profile of the slotted cylinder as seen in the top views, whereas the non-symmetric ones *Type 1* and *Type 3* do not. The *Type 1* being the most dissipative since it exhibits slightly narrower slot for the slotted cylinder compared to the other two.

3.4. Shu-Osher problem

The well established 2D Shu-Osher [67] test problem is employed, which involves the interaction between a shock wave and an entropy wave. The computational domain is $[-5, 5] \times [-0.1, 0.1]$, with periodic boundary conditions in y-axis, a supersonic inflow and outflow on the left and right side of the domain respectively. The initial profile consists of a shock wave $(\rho, u, v, p) = (3.857143, 2.629369, 0, 10.333333)$ on the left when $x < -4$ and an entropy wave in the rest of the domain $(\rho, u, v, p) = (1 + 0.2sin(5x), 0, 0, 1)$.

A 2D quadrilateral mesh is employed with 512 edges for the top and bottom sides with approximately $51,200$ elements. The reference solution is computed with an one-dimensional Euler equations on $10,000$ grid points using a WENO-5th order scheme. The calculation is run until $t = 1.8$ with WENO 5th-order scheme using all four types of directional stencil algorithms. From the density distribution plots at the left image of Fig. 23 it can be noticed that all the stencil algorithms obtained results are in good agreement with the reference solution, although it can be identified that *Type 1* algorithm is the most dissipative in the entropy wave region compared to the rest of the algorithms.

Looking closer at the interaction of the entropy wave with the shock wave region as shown at the right image of Fig. 23, it can be noticed that *Type 2* and *Type 4* exhibit more oscillations from *Type 1* and *Type 3*. This is to be expected since although their stencils are more symmetric, they are wider, hence they have a higher probability of crossing discontinuous regions. *Type 3* preserves a better agreement with the reference solution compared with *Type 1*. *Type 4* is not going to be used for any subsequent test-problems since although it might have a reduced stencil width, the fact that there are only two-directional stencils to choose from constrains its applicability to flow problems with sharp-gradients arising from various directions.

3.5. Double Mach reflection

The widely-used double Mach reflection test problem of Woodward and Colella [68] is employed for assessing the accuracy and non-oscillatory properties of the subject directional stencils algorithms. Initially a shock wave moves with Mach number $M = 10$ at an inclination of $\alpha = 60°$. Ahead of the shock the fluid is at rest with uniform density and pressure $\rho = 1.4$ and $p = 1.0$. Supersonic inflow boundary conditions are prescribed on the left side and supersonic outflow boundary conditions on the right side, and reflecting boundary conditions for the bottom of the domain beginning at $x = 1/6$ and post shock conditions before. At the top boundary the exact solution of an isolated moving oblique shock wave with $M = 10$ is prescribed. For additional details regarding the setup of the problem the reader is referred to the work of Woodward and Colella [68]. The computational domain is given by $[0, 4] \times [0, 1]$ and is discretised by an unstructured mesh of approximately $320,000$ triangular cells, that corresponds to an equivalent resolution of $h = 1/180$. WENO 3rd-, 5th-, and 7th-order schemes are employed with *Type 1*,*Type 2*, and *Type 3* directional stencil algorithms. The calculations are performed until time $t = 0.2$, as seen in Fig. 24.

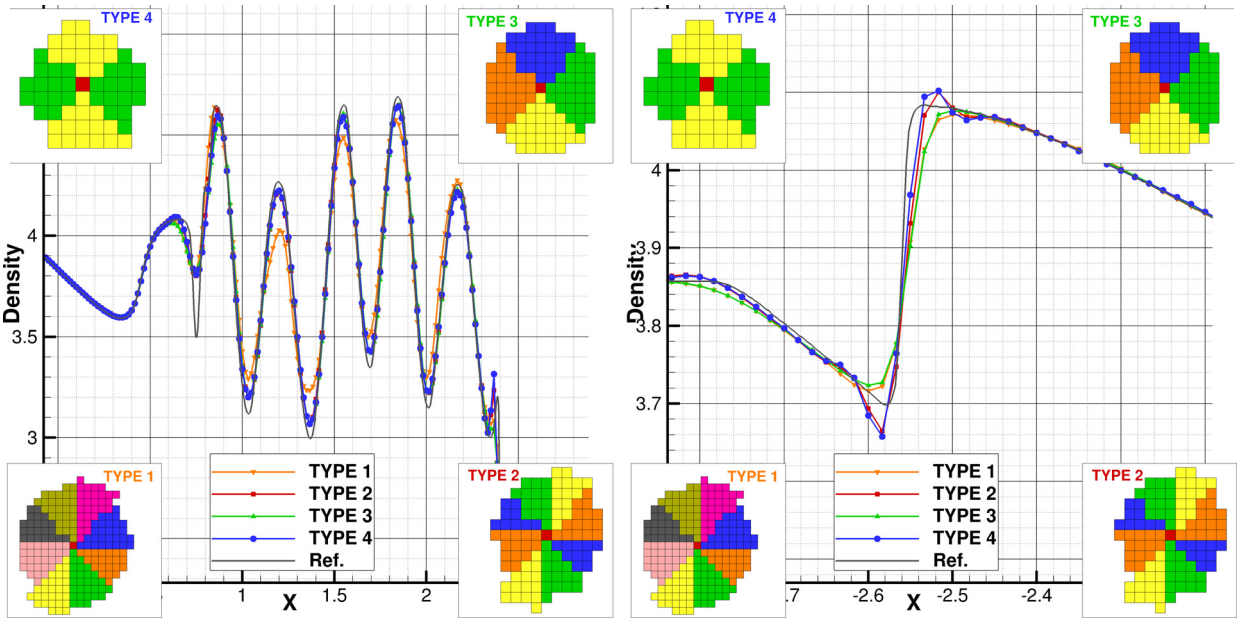


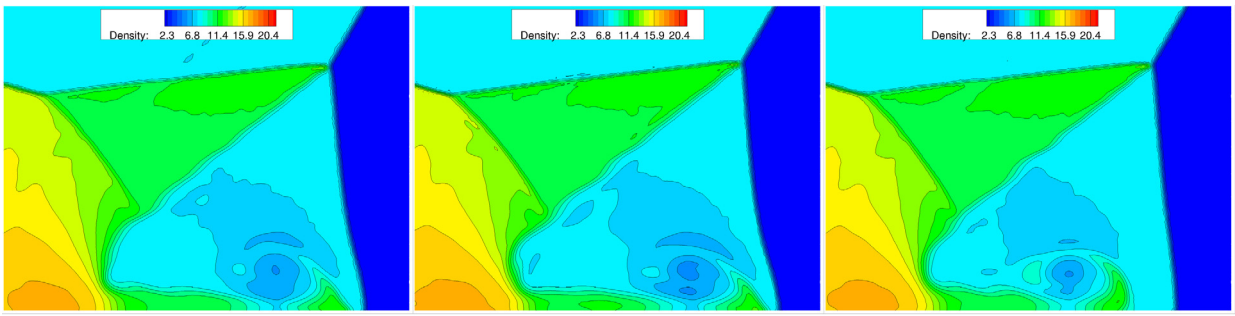
Fig. 23. Density distribution for the Shu-Osher [67] shock tube test problem at the final time $t = 1.8$ using various directional stencil algorithms for a WENO 5th-order scheme on a quadrilateral mesh and compared with the reference solution obtained from the one-dimensional Euler equations on 10,000 grid points using a WENO-5th order scheme. It can be noticed that *Type 2* and *Type 4* exhibit more oscillations from *Type 1* and *Type 3*. *Type 3* preserves a better agreement with the reference solution compared with *Type 1*.

All the stencil algorithms capture the correct flow pattern with the reflected shock, the two Mach stems and two contact discontinuities. However looking closer at the interaction zone there are two aspects that differentiate the results between these types of directional stencil algorithms. The first one is the rolling of the shear waves which is more pronounced for *Type 1* and *Type 3* algorithms for all the spatial orders. The second aspect which is the most crucial is related to the structure of the Mach stem. The curved Mach stem is associated with the carbuncle sensitivity of the scheme, the influence of the inclination of the shock wave to the grid itself and the presence or not of a Triple-Mach White-Reflection TM-WR as documented by other computational and experimental studies [69,70]. The only algorithm that is consistently free from a curved Mach stem is *Type 3* for all spatial orders of accuracy, indicating that it less sensitive to carbuncle and in agreement with results of Woodward and Colella [68]. It is worth noting that the *Type 2* algorithm although it is the least dissipative in smooth regions of the flow due to the symmetric directional stencils with respect to the considered cell, in this test problem it exhibits the least pronounced rolling of the shear waves. This symmetry is translating to reduced compactness of the directional stencils, therefore it is more likely for a larger portion of the total number of directional stencils to cross discontinuous regions compared to the other two algorithms. Hence the activation of the WENO limiting across the shear waves is more likely to occur for *Type 2* algorithm. *Type 1* and *Type 3* algorithms are not so prone to this behaviour due to the larger number of directional stencils and more compact ones respectively.

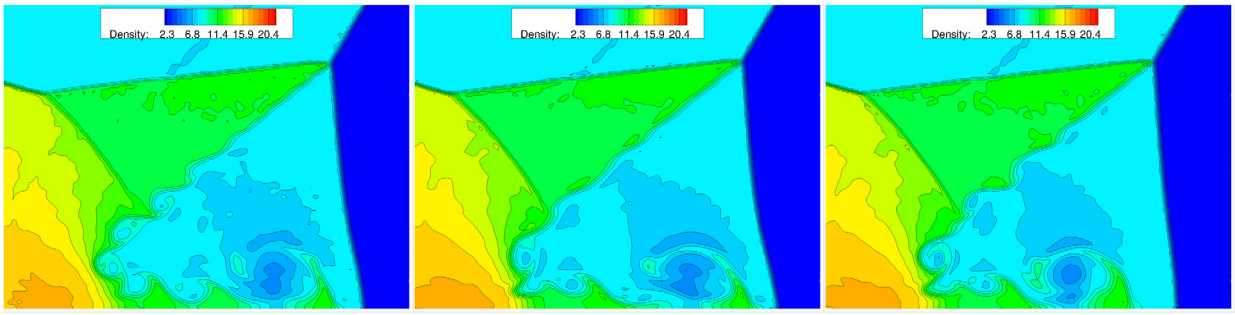
3.6. Mach 3 forward facing step

A well suited test problem for assessing the performance of the stencil algorithms due to the presence of complicated flow structures and interacting shocks is the Mach 3 forward facing step, introduced by Woodward and Colella [68]. The size of the domain is 3 units length with the step placed at 0.6 length units from the left side of the domain, with a height of 0.2 units. An ideal gas is considered with constant density $\rho = 1.4$, velocity $(u, v) = (3, 0)$ and pressure $p = 1.0$. A supersonic inflow condition is applied on the left, a supersonic outflow condition on the right and reflective boundary conditions on the upper and lower sides. The computational domain is discretised by an triangular mesh of approximately 180,000 cells, that correspond to an equivalent resolution of $h = 1/200$. The corner of the step was rounded off as suggested in [71] with a curvature of radius 0.003 in order to avoid the numerical treatment of the singularity point by different means. The 3rd-order and 5th-order numerical schemes are employed with *Type 1*,*Type 2*, and *Type 3* directional stencil algorithms. The calculations are performed until time $t = 4.0$, as seen in Fig. 25.

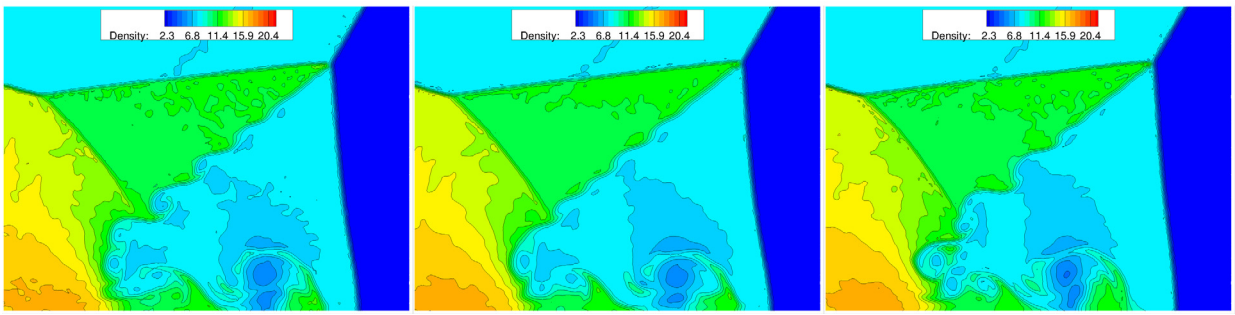
All the algorithms capture the correct flow pattern with the contact discontinuity resulting from the three-shock interaction. As expected the Kelvin-Helmholtz instabilities that develop along the contact discontinuity are better resolved as the order of accuracy is increased, with the *Type 3* algorithm having slightly better resolution than the others. Their key difference lies in the region behind the bow shock and close to the three-shock interaction where some algorithms produce "noisier" results than others which means that they are more prone to some oscillations developing. *Type 3* is the least susceptible to this behaviour, since there are sufficient number of stencils that are more compact than *Type 1* or *Type 2*.



(a) *Type 1* WENO3 (b) *Type 2* WENO3 (c) *Type 3* WENO3



(d) *Type 1* WENO5 (e) *Type 2* WENO5 (f) *Type 3* WENO5



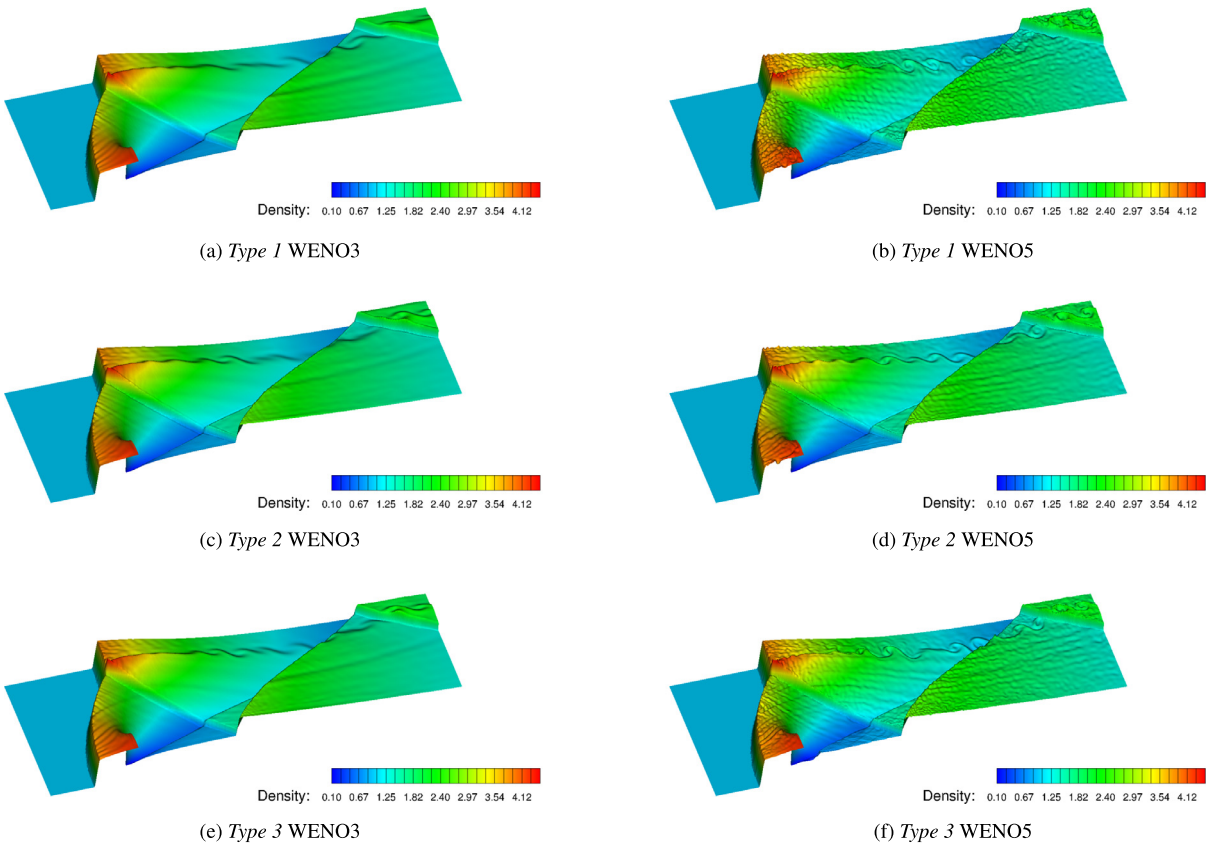
(g) *Type 1* WENO7 (h) *Type 2* WENO7 (i) *Type 3* WENO7

Fig. 24. Density distribution with various directional stencil algorithms for WENO 3rd-, 5th-, and 7th-order schemes for the double Mach reflection problem at $t = 0.2$. Zoomed on the wave interaction zone where it can be noticed that key difference is that *Type 3* algorithm consistently produced a less curved Mach stem, which is a result of the less propagated jet along the reflection axis. All the algorithms produce more pronounced shear waves as the order of accuracy is increased, but only *Type 3* is free from the carbuncle effect revealed from the curved Mach stem.

3.7. Kelvin-Helmholtz instability

The 2D Kelvin-Helmholtz instability and the resulting evolution of small shear layer into chaotic vortical structures is considered. This type of problems is widely used [72–75] for assessing the performance of various numerical methods in terms of accuracy and dissipation characteristics for evolving a linear perturbation into a non linear one. The computational domain is $[-0.5, 0.5]^2$, with periodic boundary conditions. The initial profile is $(\rho, u, v, p) = (1, 0.5, 0.01sin(2\pi x/L), 2.5)$ when $y \geq 0.25$, $(\rho, u, v, p) = (2, -0.5, 0.01sin(2\pi x/L), 2.5)$ when $-0.25 < y < 0.25$ and $(\rho, u, v, p) = (1, 0.5, 0.01sin(2\pi x/L), 2.5)$ when $y \leq -0.25$. An unstructured triangular mesh is employed with 512 edges across each side with 818670 elements. The calculation is run until $t = 5$ with an unlimited and WENO schemes of 3rd- and 6th-order of accuracy. The WENO scheme using *Type 1*, *Type 2* and *Type 3* directional stencil algorithms. Due to the initial conditions in the absence of viscosity, the fine resolution of the mesh and the variation of the numerical methods employed, that a converged solution will not be obtained. Due to the instabilities interacting with each other and forming new vortices the flow becomes chaotic at late times, hence different solutions are obtained with different numerical methods.

From the density distribution plots at various instants as shown in Fig. 27 and Fig. 28 it can be noticed that the *Type 2* and *Type 3* directional stencil algorithms resolve vortices of finer scales compared to the *Type 1* algorithm and therefore result to more pronounced vortex filamentation processes. This trend is valid for both 3rd and 6th-order WENO schemes. Investigating the time evolution of the total kinetic energy E_K as seen in Fig. 26 it can be noticed that *Type 1* is the most



(a) *Type 1* WENO3 (b) *Type 1* WENO5

(c) *Type 2* WENO3 (d) *Type 2* WENO5

(e) *Type 3* WENO3 (f) *Type 3* WENO5

Fig. 25. 3D elevation contour plots of density with various directions stencil algorithms for WENO 3rd-, and 5th-order schemes for the Mach 3 forward facing step problem, at $t = 4$. It can be noticed that all the algorithms when used with WENO 5th-order scheme resolve the contact discontinuity from the three-shock interaction and the resulting Kelvin-Helmholtz instabilities, however all the algorithms are susceptible to some oscillations emanating behind the strong shock, with *Type 1* algorithm being more pronounced than *Type 2* and *Type 3*.

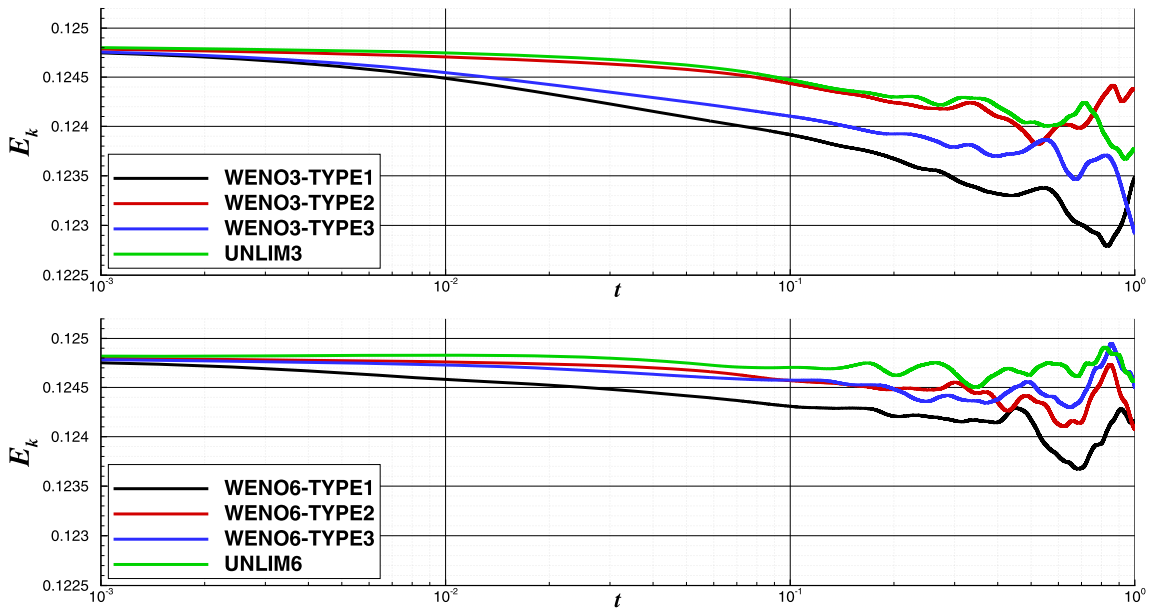
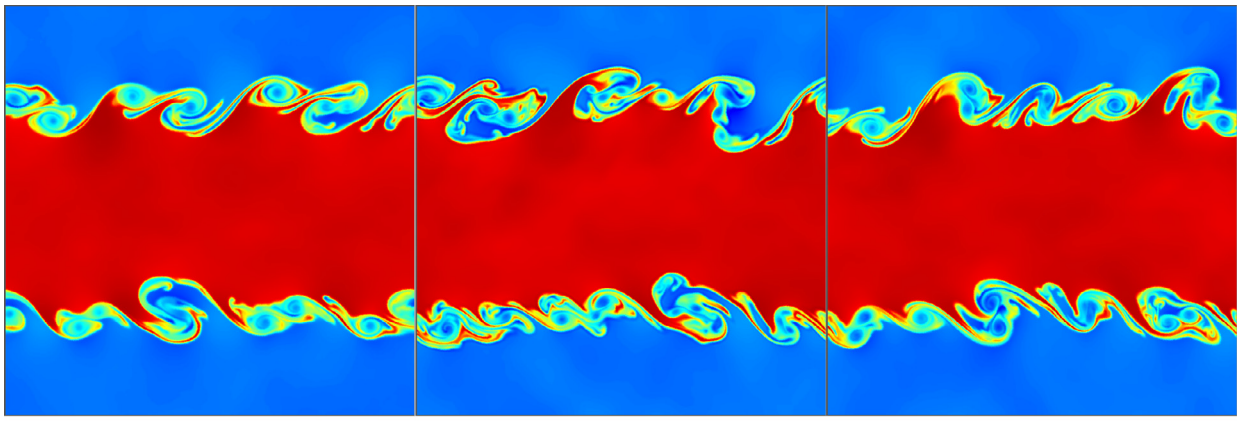
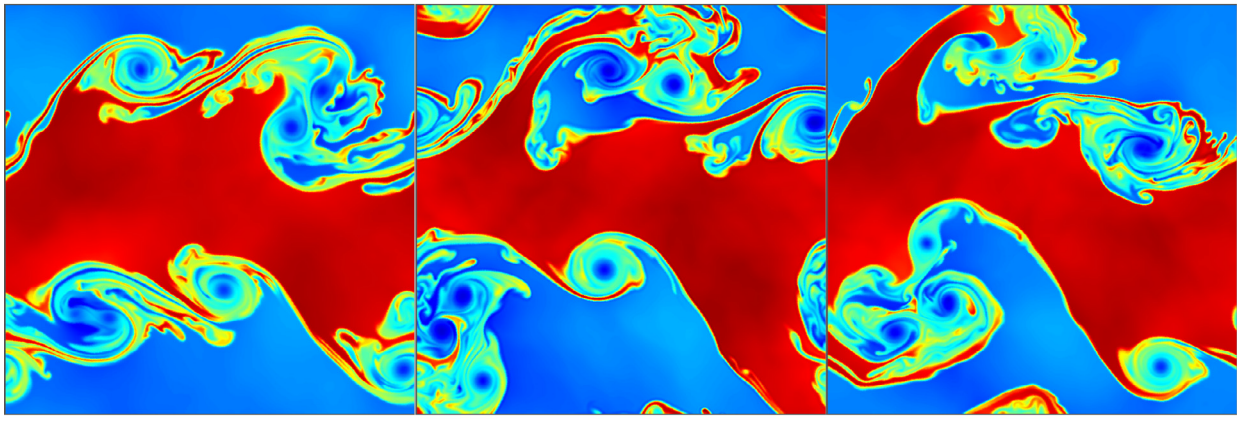


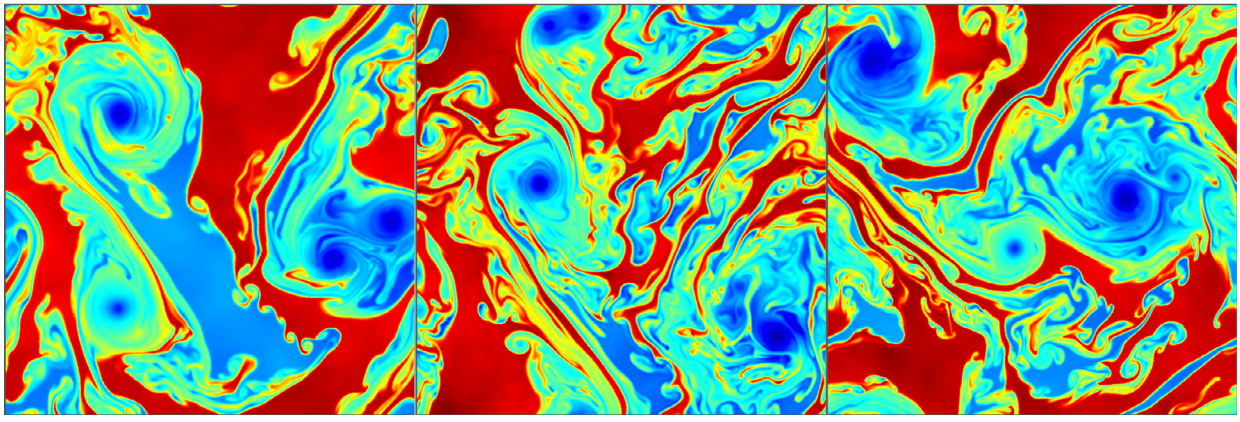
Fig. 26. Time histories of total kinetic energy E_K for the Kelvin-Helmholtz instability obtained with various directional stencil algorithms and an unlimited scheme. It can be noticed that *Type 1* is the most dissipative, while *Type 2* is the least dissipative, however the *Type 3* algorithm is getting closer agreement to *Type 2* in terms of dissipation for the WENO 6th-order scheme.



(a) *Type 1*, $t = 1.0$ (b) *Type 2*, $t = 1.0$ (c) *Type 3*, $t = 1.0$



(d) *Type 1*, $t = 2.5$ (e) *Type 2*, $t = 2.5$ (f) *Type 3*, $t = 2.5$

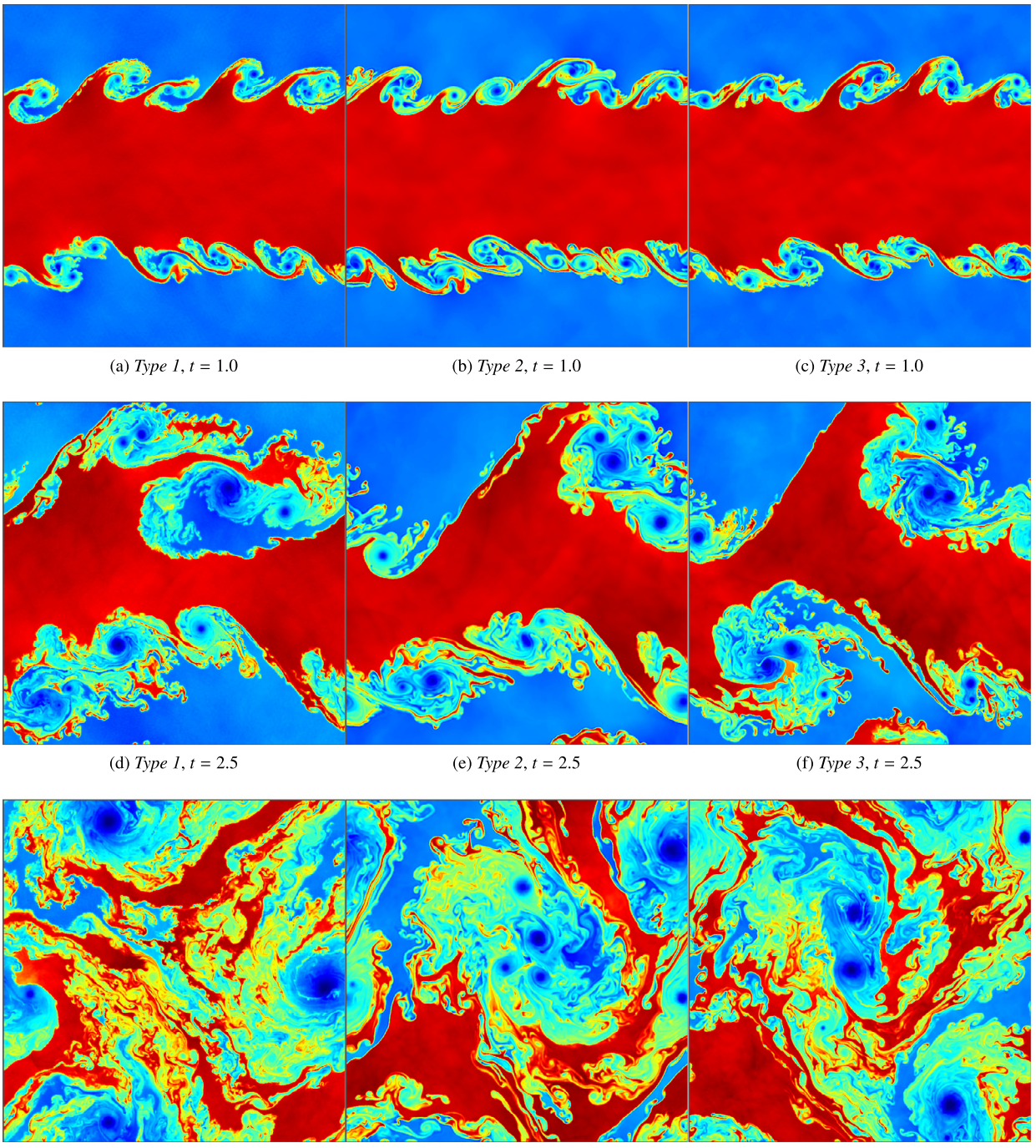


(g) *Type 1*, $t = 5.0$ (h) *Type 2*, $t = 5.0$ (i) *Type 3*, $t = 5.0$

Fig. 27. Contour plots of the density (15 equidistant contours from 0.5 to 2.25) obtained with WENO 3rd-order scheme with various directional stencil algorithms for the Kelvin-Helmholtz instability test problem at various instants. It can be noticed that the *Type 2* and *Type 3* directional stencil algorithms resolve vortices of finer scales compared to the *Type 1* algorithm.

dissipative, while *Type 2* is the least dissipative of the WENO schemes as expected due to the more symmetric directional stencils, however the *Type 3* algorithm is getting a closer agreement to the unlimited scheme at late times in terms of dissipation for 6th-order schemes.

Comparison of the obtained solutions with unlimited and WENO schemes is useful for understanding how the WENO schemes might influence the solution for a smooth flow problem. Ideally the non-linear weight of the central stencil should be approximately equal to 1.0, hence almost only the central stencil contributing to the reconstruction. In practice however,



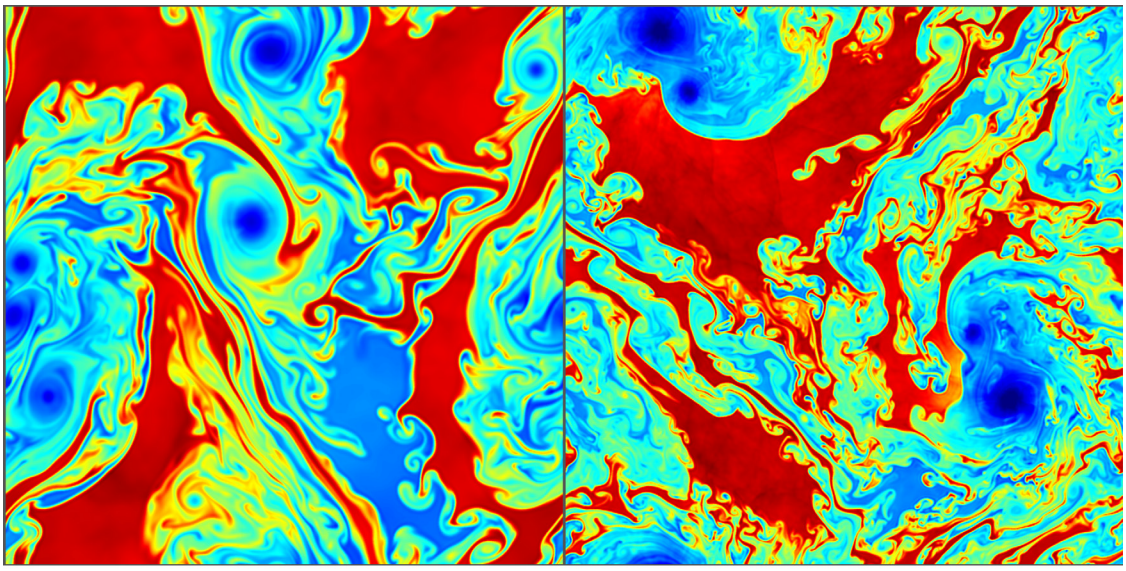
(a) *Type 1*, $t = 1.0$ (b) *Type 2*, $t = 1.0$ (c) *Type 3*, $t = 1.0$

(d) *Type 1*, $t = 2.5$ (e) *Type 2*, $t = 2.5$ (f) *Type 3*, $t = 2.5$

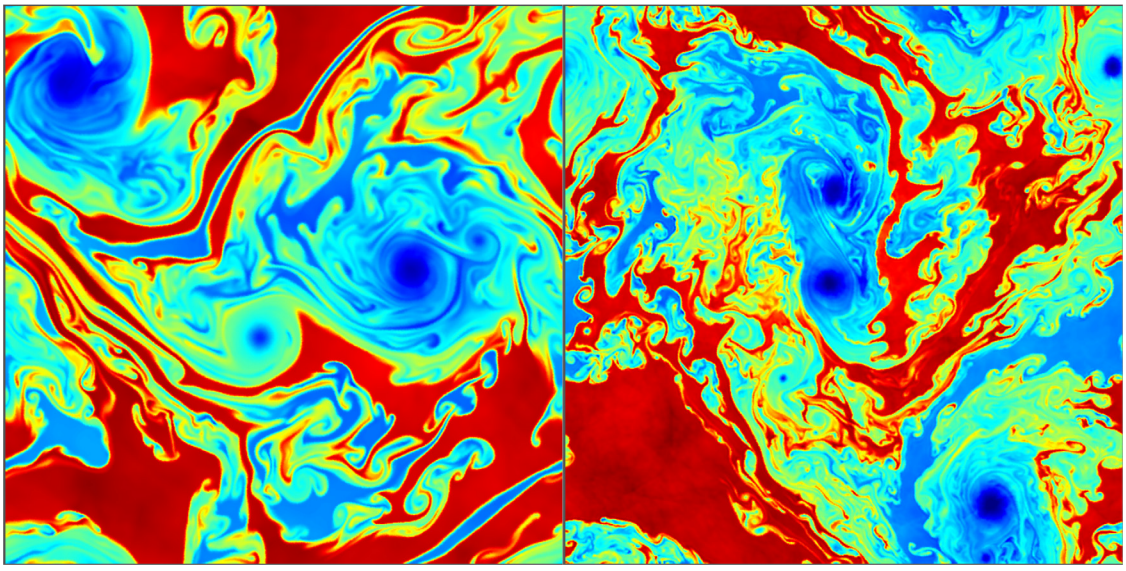
(g) *Type 1*, $t = 5.0$ (h) *Type 2*, $t = 5.0$ (i) *Type 3*, $t = 5.0$

Fig. 28. Contour plots of the density (15 equidistant contours from 0.5 to 2.25) obtained with WENO 6th-order scheme with various directional stencil algorithms for the Kelvin-Helmholtz instability test problem at various instants. It can be noticed that the *Type 2* and *Type 3* directional stencil algorithms resolve vortices of finer scales compared to the *Type 1* algorithm.

this is not the case since the resulting shear layers that trigger the instabilities and the characteristics reconstruction, round-off errors, varying morphology of the unstructured mesh itself can result in a sub-optimal non-linear weight assigned to the central stencil. For this purpose the solutions obtained at late times with unlimited schemes and WENO scheme with *Type 3* are compared in Fig. 29, and in terms of resolved features there are not significant differences between them. Comparing the various directionals stencil algorithms with respect to the percentage of elements with a non-linear weight (for the density) for the central stencil which is less than $\omega_1 < 0.999$ as shown in Fig. 30 reveals that the *Type 2* is superior with respect to



(a) *WENO 3* (b) *WENO 6*

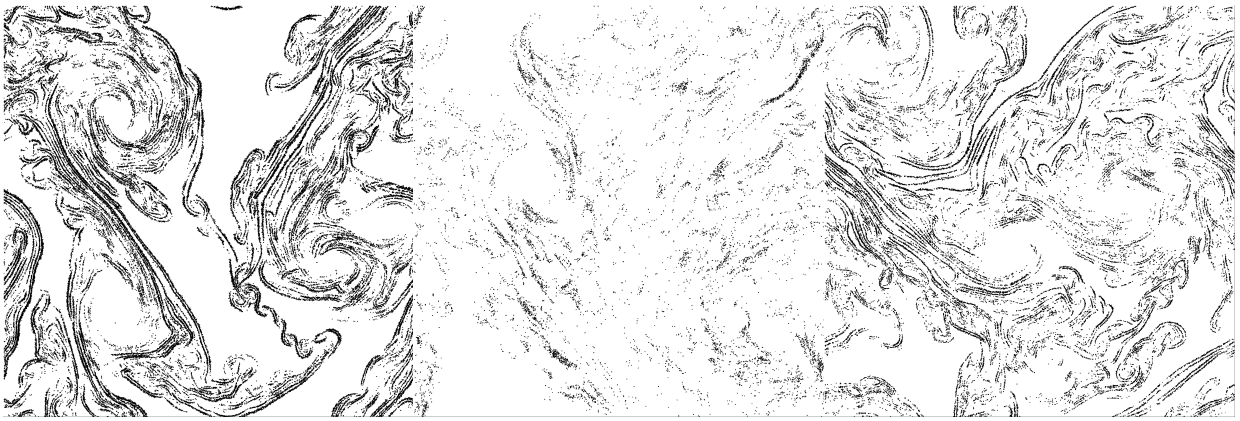


(c) *Unlimited 3* (d) *Unlimited 6*

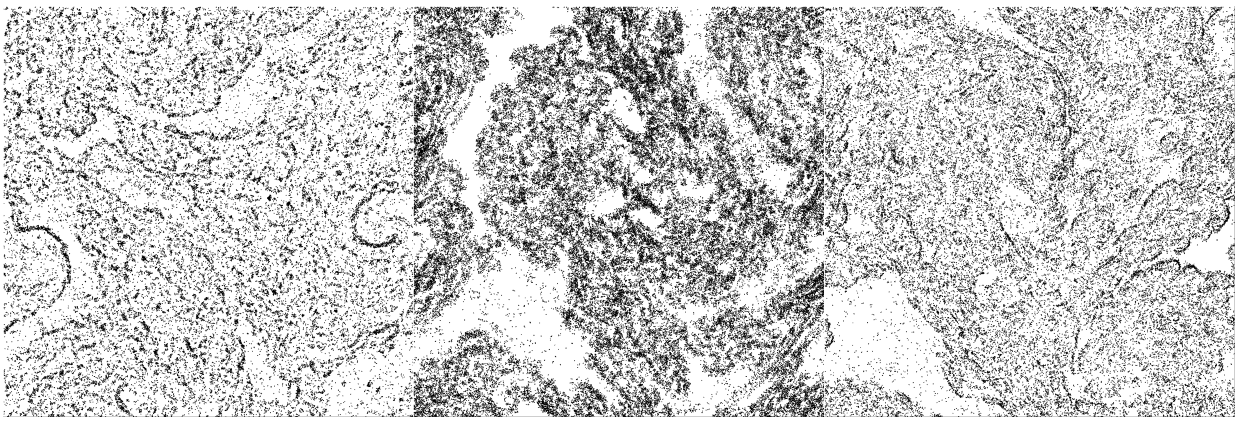
Fig. 29. Contour plots comparison of the density (15 equidistant contours from 0.5 to 2.25) obtained with WENO *Type 3* algorithm and Unlimited scheme for the Kelvin-Helmholtz instability test problem at the final time $t = 5.0$. There are not significant differences between the results obtained by the two approaches.

the other algorithms for 3rd-order WENO schemes. This is again justified by the more symmetric directional stencils which results in just 1.10% being below this threshold. When switching to a 6th-order scheme however, the larger width of the *Type 2* algorithm and its narrower sectors results in significantly more elements being limited by the directional stencils, as opposed by *Type 3* which has the lowest percentage of elements limited 6.78%.

All the simulations where performed on 8 Intel Xeon Phi 7210 processors, and the CPU times required for every timestep of all the schemes are normalised with respect to the fastest scheme which is the Unlimited 3rd-order scheme. For the WENO schemes the computational cost increases by a factor of $(2-3.2)$ and $(4.9-8.0)$ for 3rd-order and 6th-order schemes respectively as shown in as shown in Table 8. Therefore the stencil algorithms selected can have a profound impact on the computational times. Interestingly *Type 2* algorithm has slightly higher computational cost as shown in Table 8 compared to *Type 3* for WENO 3rd-order and WENO 6th-order scheme, due to the stencils having a greater width hence resulting in significantly more elements across interprocessor boundaries. For WENO 6th-order scheme the difference in CPU times between the different types of stencil algorithms is more pronounced with *Type 3* being the fastest and *Type 1* the slowest performing algorithms.



(a) *Type 1*, 8.40% (b) *Type 2*, 1.10% (c) *Type 3*, 4.45%



(d) *Type 1*, 6.90% (e) *Type 2*, 12.1% (f) *Type 3*, 6.78%

Fig. 30. Plot only of the cells (the remaining are blanked) with a WENO weight (for the density) for the central stencil which is less than $\omega_1 < 0.999$ for the Kelvin-Helmholtz instability test problem with a WENO 3rd-order (top) and WENO 6th-order (bottom) scheme at $t = 5.0$ with various directional stencil algorithms. The percentage of the elements in the computational domain with $\omega_1 < 0.999$ provides an additional measure of how each WENO algorithm is limiting the reconstruction arising from the central stencil. It can be noticed that the *Type 2* has the lowest percentage of limited elements for the WENO 3rd-order scheme and for WENO 6th-order scheme *Type 3* algorithm has the lowest percentage.

Table 8
Normalised CPU times for the Kelvin-Helmholtz instability obtained with various directional stencil algorithms and an unlimited scheme.

Scheme	Central	Type 1	Type 2	Type 3
Unlim 3	1	–	–	–
Unlim 6	3.34	–	–	
WENO 3	–	3.2	2.2	2.0
WENO 6	–	8.0	5.6	4.9

3.8. iLES of Taylor Green vortex $Re = 1600$

The *i*LES of the 3D viscous Taylor-Green vortex test problem at $Re = 1600$ is employed, for assessing the influence of the directional stencil algorithms. Primarily it is used for studying the vortex stretching and transition to turbulence and secondly for the validation of developed numerical methods. This test problem when employed at relative coarse-"under-resolved" meshes, reveals the dissipation and dispersion characteristics of non-linear methods at resolutions typically found in regions close to the boundary layer for larger Reynolds number wall bounded flows. It has been widely used by many researchers to assess the behaviour of various numerical schemes [3,46,76–82], in the context of *i*LES. Furthermore there are well-documented DNS results of Brachet et al. [83] to compare against and are used as a reference for this study.

The computational domain is defined as $\Omega = [0, 2\pi]^3$ with periodic boundary conditions. This formulation of the Taylor-Green vortex problem is initialized with the following velocity, density and pressure fields:

$$u(x, y, z, 0) = \sin(kx)\cos(ky)\cos(kz), \tag{37}$$

$$v(x, y, z, 0) = -\cos(kx)\sin(ky)\cos(kz), \tag{38}$$

$$w(x, y, z, 0) = 0, \tag{39}$$

$$\rho(x, y, z, 0) = 1, \tag{40}$$

$$p(x, y, z, 0) = 100 + \frac{\rho}{16}[\cos(2z) + 2] \cdot [\cos(2x) + \cos(2y)]. \tag{41}$$

The initial condition corresponds to an initial Mach number $M \approx 0.08$, with $k = 2\pi/\lambda = 1$ being the wavenumber. Simulations were carried out on a hexahedral mesh of 96^3 using the WENO 4th-order scheme with *Type 1*, and WENO-5th order schemes with *Type 3* directional stencils. The primary reason for this is to demonstrate that for a similar computational cost to a WENO 4th-order with the *Type 1* algorithm, a higher-order scheme (WENO 5th-order) can be used by just switching *Type 3* directional stencil algorithm which is more compact. The simulations were run with a CFL number of 1.3 for the explicit Runge-Kutta 4th-order scheme, up to $t = 20$ for obtaining the dissipation statistics.

From the obtained results as shown in Fig. 31 it can be noticed that the kinetic energy dissipation rate for the WENO 5th-order *Type 3* scheme is in closer agreement with the reference DNS results for $Re = 1600$, than the WENO 4th-order *Type 1* scheme. Furthermore to the previous expected finding, the WENO 5th-order *Type 3* scheme is just 11% more computationally expensive compared to the WENO 4th-order *Type 1*. There are two reasons for that, firstly the stencils are more compact in the sense that they occupy less space, and secondly the computational performance favours fewer and larger matrix multiplications as it has also been mentioned in the WENO performance optimisation study by Tsoutsanis et al. [22]. The WENO reconstruction process is the most computationally intensive part of the algorithm and it mainly involves matrix-matrix and matrix-vector multiplications. Since optimised linear algebra libraries are used in the UCNS3D code [22], using larger and fewer matrices typical of *Type 1* algorithm a higher throughput is achieved, therefore resulting in increased performance and accuracy compared to the WENO 4th-order *Type 1*.

3.9. *iLES of transonic cylinder* $Re = 3900$

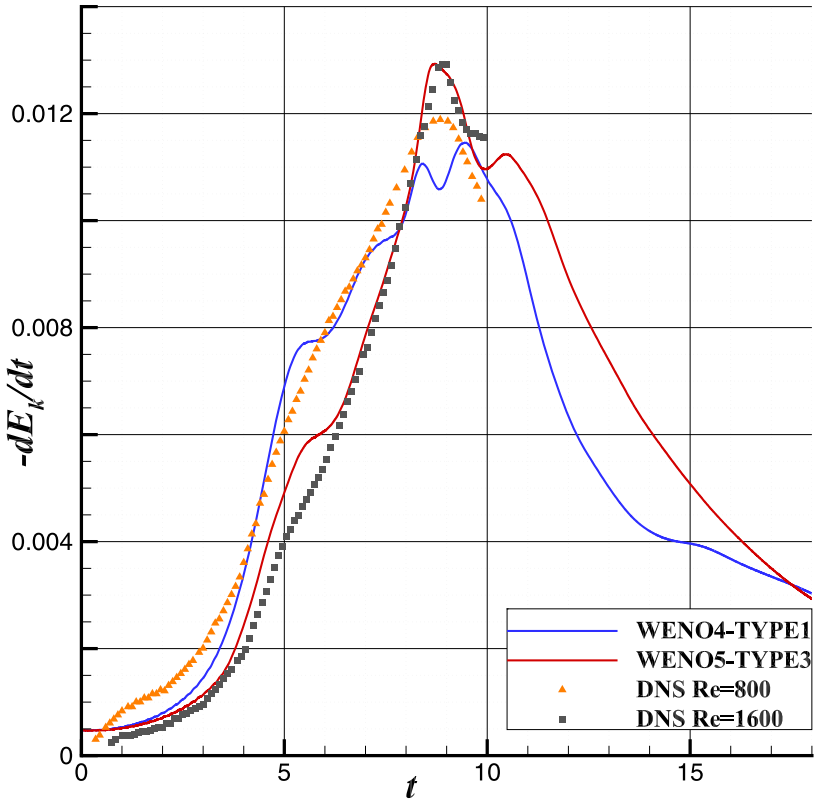
The last test case concerns the *i*LES of a transonic infinite cylinder of diameter $D = 1$ at $Re = 3900$ and $M_{\text{inf}} = 0.8$. At this Reynolds number $Re = 3900$ there are numerous computational results [82,84–91] for subsonic Mach numbers, since the low Reynolds number is suitable for exploring various high-fidelity numerical methods in the context of *i*LES. The main motivation for employing a transonic flow in this configuration is firstly to better justify the use of WENO schemes due to the presence of shock-waves and secondly to highlight the computational performance benefits associated with the utilisation of a more compact stencil algorithm.

For this case a hybrid unstructured mesh of approximately 9.3 million cells is generated, consisting of hexahedral and prismatic cells as illustrated in Fig. 32. The domain extends $10D$ upstream and $25D$ downstream, and $2\pi D$ in the z-direction. The grid resolution at the boundary layer region gives a $y^+ \approx 1$ at the first cell off the surface, 180 cells are used in the z-direction, and a refinement region at the wake of the cylinder is used for $10D$ downstream with an average cell length of 0.05. Periodic boundary conditions are used in the z-direction, no-slip boundary conditions at the surface of the cylinder and free-stream conditions at the farfield. Due to the periodic boundary conditions and the nature of the grid being extruded in the z-direction from a 2D-mesh, there are regions with relatively medium aspect ratio ($4 < AR < 15$) cells close to the wall, and very high-aspect ratio cells ($AR > 1000$).

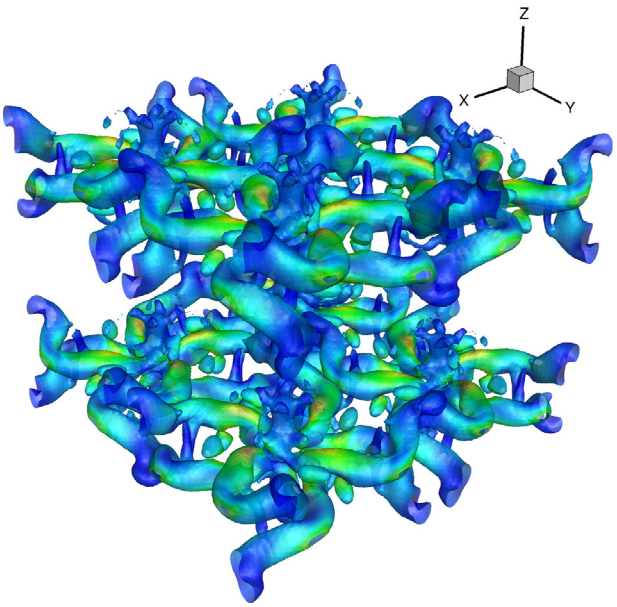
Two numerical methods are used a WENO 4th-order *Type 1* scheme with the NCB algorithm and a WENO 5th-order *Type 3* scheme with the SBC algorithm. A CFL number of 1.3 is set along with the explicit Runge-Kutta 4th-order scheme. The main objective of employing an NCB algorithm which is inferior to the SBC in all the tests performed, is to further highlight the impact that a non-carefully selected stencil algorithm can have in the computational efficiency of a 3D unsteady simulation. The simulations were run for $t = 10t_c$ where $t_c = D/U_\infty$, for obtaining an average drag coefficient, and due to the computational budget available the analysis of turbulent statistics was not pursued, since the focus for this paper was to highlight in a realistic test problem the influence of the stencil algorithm.

From the obtained results as it can be seen in Fig. 33, the typical regime at this Mach number associated with the elongated vortex formation, wake shock wave and the pressure waves aft the wake shock wave are better resolved by the WENO 5th-order *Type 3* scheme with the SBC algorithm and in better agreement with the experimental results of Ackerman et al. [92]. Furthermore, the drag coefficient obtained with the WENO 5th-order scheme $C_D = 1.06$ and the WENO 4th-order scheme $C_D = 1.12$, are within the error threshold of the results of Ackerman et al. [92].

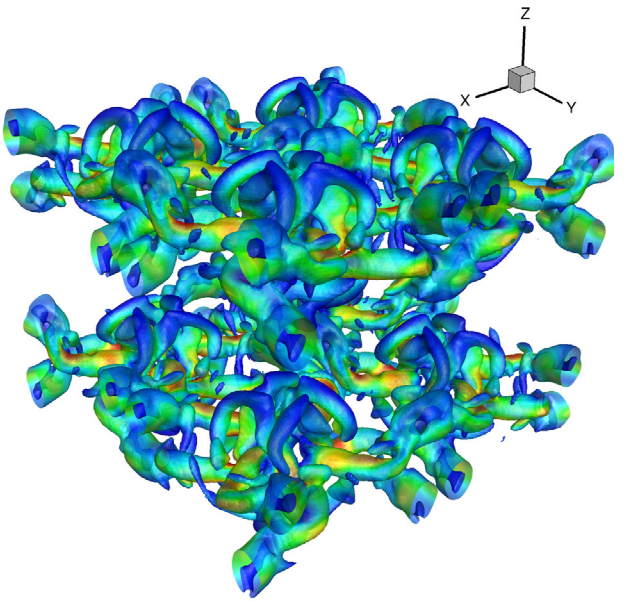
It needs to be highlighted that the WENO 5th-order *Type 3* scheme with the SBC algorithm is just 19% more expensive compared to the WENO 4th-order *Type 3* scheme with the NCB algorithm. This is due to the fact that the central and directional stencils are more compact and therefore they have reduced communication requirements across cpus for halo cells, as well as the better throughput achieved with fewer and larger matrices under the current implementation of the UCNS3D code [22]. This is demonstrated in Fig. 34, where it can be noticed that the WENO 5th-order scheme employs stencils of similar-size if not more compact than the WENO 4th-order scheme. The geometrical sectors for the *Type 1* directional algorithm are narrow and they result in wide stencils, furthermore the NCB central algorithm is not taking



(a) Kinetic energy dissipation rate



(b) Iso-surface of Q criterion coloured by kinetic energy for WENO 4th-order type 1 scheme



(c) Iso-surface of Q criterion coloured by kinetic energy for WENO 5th-order type 3 scheme

Fig. 31. Kinetic Energy dissipation rate and isosurfaces of Q criterion for the viscous *i*LES of the Taylor-Green Vortex $Re = 1600$ obtained with WENO 4th-order *Type 1* scheme and WENO 5th-order *Type 3* scheme on a 96^3 hexahedral mesh, and comparison with the DNS results of Brachet et al. [83]. It can be noticed that the WENO 5th-order *Type 3* scheme is in much closer agreement with the DNS results for $Re = 1600$ and resolves finer scales, compared to WENO 4th-order type 1, while costing 11% more in computational time.

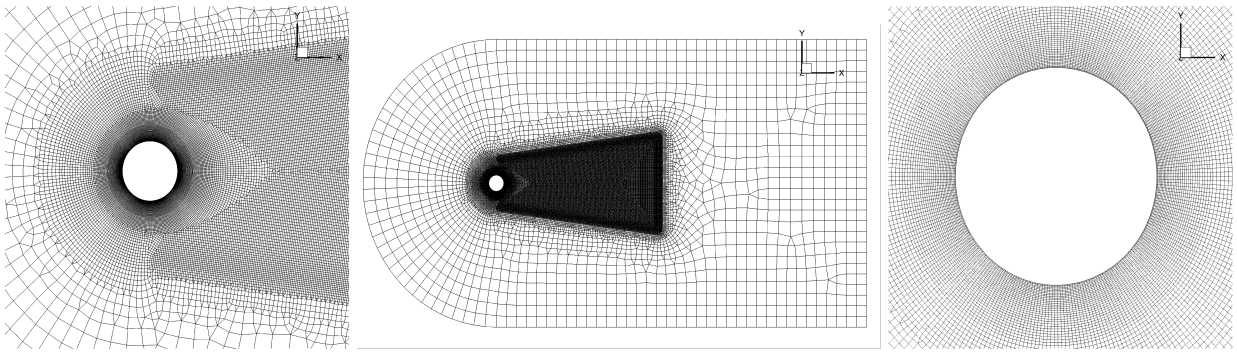


Fig. 32. Overview of the hybrid unstructured mesh used for the transonic flow past an infinite cylinder at $Re = 3900$ and $M_\infty = 0.8$.

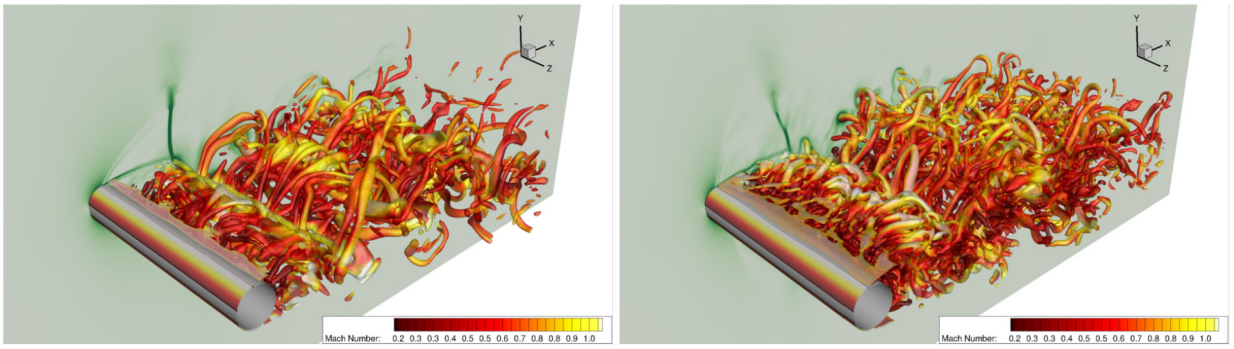


Fig. 33. Instantaneous iso-surfaces of Q criterion (Q=200) coloured by Mach number using a WENO 4th-order *Type 1* scheme with NCB algorithm (left) and a WENO 5th-order *Type 3* scheme with SBC algorithm (right) for the transonic flow past an infinite cylinder at $Re = 3900$ and $M_\infty = 0.8$. It can be noticed that flow structures typical of this regime [92] the wake shock wave, the elongated vortex formation region and the pressure waves are more pronounced with the WENO 5th-order *Type 3* scheme, as well as finer vortical structures are resolved.

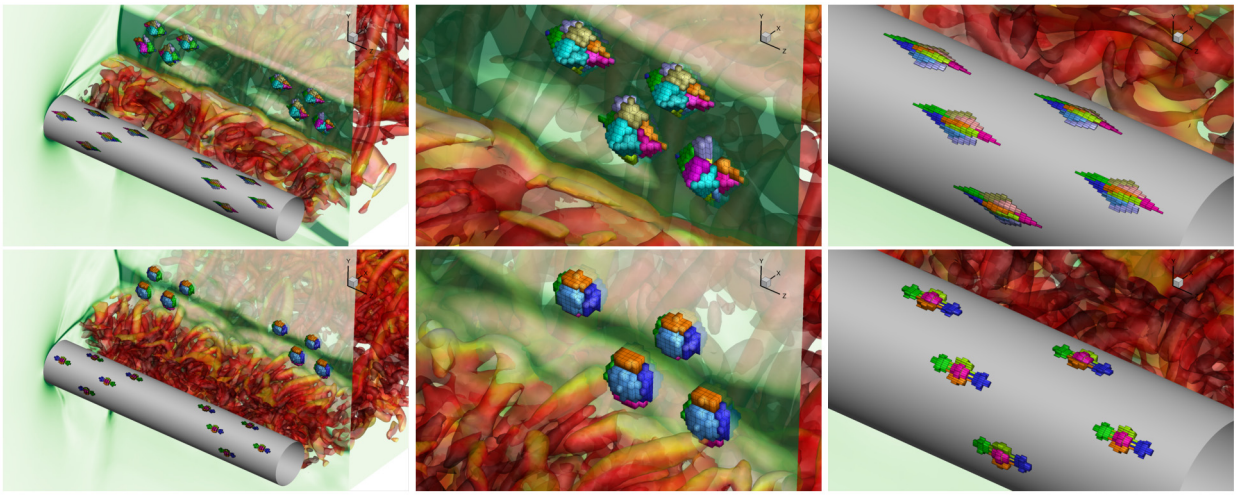


Fig. 34. Plots of stencils at various regions of the computational domain, for the transonic flow past an infinite cylinder at $Re = 3900$ and $M_{\text{inf}} = 0.8$ using a WENO 4th-order *Type 1* scheme with NCB algorithm (top) and a WENO 5th-order *Type 3* scheme with SBC algorithm (bottom). It can be noticed that the stencils arising from the WENO 5th-order *Type 3* scheme have equivalent if not smaller spatial footprint compared to the WENO 4th-order *Type 3* scheme.

into account the distance of the candidate elements from the considered cell, making the situation even worse. The results highlight that higher-order WENO schemes on unstructured meshes can be employed using more compact stencil algorithms.

4. Conclusions

This paper investigates various stencil selection algorithms for WENO schemes for smooth and discontinuous flow problems. The algorithms are differentiated with respect to the central stencil and directional stencils since different parameters need to be considered for each one of them. From the various central stencil algorithms presented the SBC algorithm in its present implementation is the most robust, for good and bad quality meshes, and it is therefore the selected algorithm for the present WENO schemes. Regarding the directional stencils, the situation is more complicated, in the sense that a fine balance between computational cost and non-oscillatory properties of the WENO schemes are pursued. The *Type 3* directional stencil algorithm provides significant computational savings and a satisfactory level of robustness for all the test-problems investigated and remains the algorithm of choice for the present implementation.

For other WENO implementations on unstructured meshes including the CWENO [8,10] and hybrid DG-WENO [17,19], *Type 1* algorithm might be more suitable. CWENO and hybrid DG-WENO schemes have the advantage of employing compact directional stencils of $\mathcal{P} = 1$ or $\mathcal{P} = 2$ polynomial order. Therefore the computational cost associated with them is not as significant as the using high-order polynomial reconstructions for the directional stencils. Hence *Type 1* algorithm might be the most suitable for these implementations due to its enhanced non-oscillatory properties.

Comprehensive studies devoted to the comparison of WENO and CWENO schemes on unstructured meshes are limited, hence there is a need to identify potential benefits of switching to CWENO schemes as part of a future study. It can be expected though that the compactness of the CWENO schemes might prove beneficial close to discontinuities on coarse grids, but in smooth regions of the flow a falsely reduction of the non-linear weight of the central stencil due to the grid topology/morphology will reduce the order of accuracy significantly more than the traditional WENO schemes. Finally expanding the WENO schemes to be able to adapt their stencil algorithm usage in real-time, based on some oscillation indicator metrics, so that the optimal one is used at any instant is a future research direction pursued.

Data availability

Data relating to the results in this manuscript can be downloaded from https://doi.org/10.17862/cranfield.rd.8983772.v1 under a CC-BY 4.0 license.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

The author acknowledges the computing time on the UK national high-performance computing service ARCHER that was provided through the UK Turbulence Consortium in the framework of the EPSRC grant EP/L000261/1 and the computing time on the University of Cambridge Tier-2 national high-performance computing service Peta4 provided through the EPSRC grant EP/P020259/1. The author also acknowledges the computing time at CINECA (Bologna – IT) provided through the Project HPC-EUROPA3 (INFRAIA-2016-1-730897), with the support of the EC Research Innovation Action under the H2020 Programme. Finally the author would like to acknowledge the constructive comments and suggestions from the reviewers that greatly enhanced this manuscript.

Appendix A. Central stencil algorithms

Algorithm 1 NCB.

```
1: procedure NCB(output=i,counter3)
2:     USE Stencil Algorithms
3:     for All cells i do
4:         counter3=0,Icomp=0,Ivert=0
5:         Call Big Stencil
6:         Call Sort Store
7:     end for
8: end procedure
```

Appendix B. Directional stencil algorithms

Algorithm 2 VBC.

```
1: procedure VBC(output=i,counter3)
2:     USE Stencil Algorithms
3:     for All cells i do
4:         Icomp=0
5:         Call Cells Quality
6:         if Qual=1 then
7:             Icomp=1
8:         end if
9:         counter3=0,Ivert=1
10:         Call Big Stencil
11:         Call Sort Store
12:     end for
13: end procedure
```

Algorithm 3 FBC.

```
1: procedure FBC(output=i,counter3)
2:     USE Stencil Algorithms
3:     for All cells i do
4:         Icomp=0
5:         Call Cells Quality
6:         if Qual=1 then
7:             Icomp=1
8:         end if
9:         counter3=number of faces, Ivert=0
10:         Call Big Stencil
11:         Call Sort Store
12:     end for
13: end procedure
```

Algorithm 4 SBC.

```
1: procedure SBC(output=i,counter3)
2:     USE Stencil Algorithms
3:     for All cells i do
4:         Icomp=0
5:         Call Cells Quality
6:         if Qual=1 then
7:             Icomp=1
8:         end if
9:         counter3=1.2 · K, Ivert=0
10:         Call Big Stencil
11:         Call Sort Store
12:     end for
13: end procedure
```

Table 9
Variables definitions for stencil algorithms.

Variable	Definition
AR	Aspect ratio of cell
c_c	List of direct side neighbours
c_v	List of vertex based neighbours
counter	Loop counter
counter2	Counter for number of stored elements in the stencil
counter3	The number of elements in the stencil not to be sorted
d_f	The farthest distance from the considered cell, from the counter3 list of elements
d_f	The farthest distance from the considered cell, from the list of temporary stencil elements excluding the elements in the counter3 list
EX	Expansion factor of the stencil ($M = EX \cdot K$)
Icomp	Compactness flag
Ivert	Vertex based neighbours flag
Qual	Quality criterion flag
K	Number of degrees of freedom of polynomial
mx	Number of elements ($mx >> M$) -typically ($mx = 4 \cdot M$) of "Big Stencil"
S^g	List of temporary stencil elements
S_m	List of finalised stencil elements

Algorithm 5 Stencil Algorithms.

procedure BIG STENCIL(input=counter3,Ivert,output=S^g_{mx},counter3)
  $S^g_1 = i$,counter=1,counter2=0,counter3
  for Loop 1: cell c, counter2=counter2+1 **do**
    If counter=1 and vertex mode is active (Ivert=1) **then**
    List the number of all the vertex-based neighbours of cell c, $c_v = 1, ..., V1$
    for each vertex-based neighbour c_v **do**
      If the element does not belong in the stencil list $S^g_{counter}$
      Store in the next available position, counter=counter+1, counter3=counter3+1
      endif
    end for
    else
    List the number of direct side neighbours of cell c, $c_c = 1, ..., V$
    for each direct side neighbours c_c **do**
      if the element does not belong in the stencil list $S^g_{counter}$
      store in the next available position, counter=counter+1
      endif
    end for
    end if
  end for
  if $counter \leq mx$ **then**
  set $c = S^g_{counter2}$ and repeat process from Loop 1
  else
  exit
  endif
end procedure
procedure SORT STORE(input=S^g_{mx},counter3,Icomp,output=S_m)
  if Compact is active, (Icomp=1) **then**
    Copy the first $counter3$ elements from S^g_{mx} and store in S_m
    Compute the distance d_f of the farthest element in the list for the first counter 3 elements wrt to cell i
    Compute the distance d_n of the elements from counter3 to mx wrt to cell i
    Bubble sort the rest of the elements from counter3 to mx wrt d_n
    for each candidate element in the sorted list S^g_n,$n = counter3, ..., mx$ **do**
      if $d_n \leq d_f$ or $n \leq M$ **then**
        Store to next available position n in S_m until $M = max(counter3, EX \cdot K)$
      end if
    end for
  elseCompact is not active,(Icomp=0)
    Copy the first M elements from S^g_{mx} and store in S_m
  end if
end procedure
procedure CELLS QUALITY(input=i,output=$Qual$)
  $Qual = 1$
  if Aspect Ratio $AR \geq 10$ **then**
    $Qual = 0$
  end if
end procedure

Algorithm 6 WENO stencil algorithm (WSA).

procedure WSA(output=i)
  USE Stencil Algorithms
  for All cells i **do**
    Determine considered cell centre coordinates $\mathbf{c}_i$, inverse jacobian for transformation $J^{-}1$
    for All faces j **do**
      Determine the coordinates of the triangular (2D), or tetrahedral (3D) element from vertices of cell centre $\mathbf{c}_i$ and the vertices of the face considered for the subject directional stencil algorithm $\mathbf{w}_{1,2,....,N}$
      for All candidate elements in the "Big Stencil" $k = 1, ..., mx$ **do**
        Check if the transformed coordinates of the candidate element centre $\mathbf{cc}'$
        Satisfy the condition $\mathbf{cc}' = J^{-1} \cdot (\mathbf{cc} - \mathbf{w}_1) \geq 0$
        If condition is satisfied and the element is not already in the list of the considered directional stencil, store in the next available memory location until the target number of elements M has been reached
      end for
    end for
  end for
end procedure

References

[1] P. Bakhvalov, T. Kozubskaya, Ebr-weno scheme for solving gas dynamics problems with discontinuities on unstructured meshes, Comput. Fluids 157 (2017) 312–324.

[2] P. Tsoutsanis, V. Titarev, D. Drikakis, WENO schemes on arbitrary mixed-element unstructured meshes in three space dimensions, J. Comput. Phys. 230 (4) (2011) 1585–1601.

[3] P. Tsoutsanis, A. Antoniadis, D. Drikakis, WENO schemes on arbitrary unstructured meshes for laminar, transitional and turbulent flows, J. Comput. Phys. 256 (2014) 254–276.

[4] G. Lei, W. Li, Y. Ren, High-order unstructured-grid WENO fvm for compressible flow computation, Jisuan Wuli/Chin. J. Comput. Phys. 28 (5) (2011) 633–640.
[5] W. Li, Y.-X. Ren, High-order k-exact WENO finite volume schemes for solving gas dynamic Euler equations on unstructured grids, Int. J. Numer. Methods Fluids 70 (6) (2012) 742–763.
[6] G. Hu, R. Li, T. Tang, A robust WENO type finite volume solver for steady Euler equations on unstructured grids, Commun. Comput. Phys. 9 (3) (2011) 627–648.
[7] H.-S. Zheng, N. Zhao, J. Zhu, High order finite volume weighted essentially non-oscillatory schemes on two dimensional unstructured meshes, Kōngqì Dònglìxué Xuébào/Acta Aerodyn. Sin. 28 (4) (2010) 446–451.
[8] M. Dumbser, W. Boscheri, M. Semplice, G. Russo, Central weighted eno schemes for hyperbolic conservation laws on fixed and moving unstructured meshes, SIAM J. Sci. Comput. 39 (6) (2017) A2564–A2591, https://doi.org/10.1137/17M1111036.
[9] L. Ivan, C. Groth, High-order solution-adaptive central essentially non-oscillatory (CENO) method for viscous flows, J. Comput. Phys. 257 (PA) (2014) 830–862, https://doi.org/10.1016/j.jcp.2013.09.045.
[10] D. Levy, S. Nayak, C.-W. Shu, Y.-T. Zhang, Central WENO schemes for Hamilton-Jacobi equations on triangular meshes, SIAM J. Sci. Comput. 28 (6) (2006) 2229–2247.
[11] M. Dumbser, M. Kaser, V. Titarev, E. Toro, Quadrature-free non-oscillatory finite volume schemes on unstructured meshes for nonlinear hyperbolic systems, J. Comput. Phys. 226 (1) (2007) 204–243.
[12] M. Dumbser, M. Kaser, Arbitrary high order non-oscillatory finite volume schemes on unstructured meshes for linear hyperbolic systems, J. Comput. Phys. 221 (2) (2007) 693–723.
[13] R. Abgrall, On essentially non-oscillatory schemes on unstructured meshes: analysis and implementation, J. Comput. Phys. 114 (1) (1994) 45–58.
[14] O. Friedrich, Weighted essentially non-oscillatory schemes for the interpolation of mean values on unstructured grids, J. Comput. Phys. 144 (1) (1998) 194–212.
[15] X. Liu, L. Xuan, Y. Xia, H. Luo, A reconstructed discontinuous Galerkin method for the compressible Navier-Stokes equations on three-dimensional hybrid grids, Comput. Fluids 152 (2017) 217–230, https://doi.org/10.1016/j.compfluid.2017.04.027.
[16] J. Zhu, J. Qiu, Hermite WENO schemes and their application as limiters for Runge-Kutta discontinuous Galerkin method, iii: unstructured meshes, J. Sci. Comput. 39 (2) (2009) 293–321.
[17] Z. Xu, Y. Liu, C.-W. Shu, Hierarchical reconstruction for discontinuous Galerkin methods on unstructured grids with a WENO-type linear reconstruction and partial neighboring cells, J. Comput. Phys. 228 (6) (2009) 2194–2212.
[18] M. Dumbser, D. Balsara, E. Toro, C.-D. Munz, A unified framework for the construction of one-step finite volume and discontinuous Galerkin schemes on unstructured meshes, J. Comput. Phys. 227 (18) (2008) 8209–8253.
[19] J. Zhu, J. Qiu, C.-W. Shu, M. Dumbser, Runge-Kutta discontinuous Galerkin method using WENO limiters ii: unstructured meshes, J. Comput. Phys. 227 (9) (2008) 4330–4353.
[20] H. Luo, J. Baum, R. Lohner, A Hermite WENO-based limiter for discontinuous Galerkin method on unstructured grids, J. Comput. Phys. 225 (1) (2007) 686–713.
[21] X. Zhang, Y. Xia, C.-W. Shu, Maximum-principle-satisfying and positivity-preserving high order discontinuous Galerkin schemes for conservation laws on triangular meshes, J. Sci. Comput. (2011) 1–34.
[22] P. Tsoutsanis, A. Antoniadis, K. Jenkins, Improvement of the computational performance of a parallel unstructured weno finite volume cfd code for implicit large eddy simulation, Comput. Fluids 173 (2018) 157–170, https://doi.org/10.1016/j.compfluid.2018.03.012.
[23] D. Levy, G. Puppo, G. Russo, Compact central weno schemes for multidimensional conservation laws, SIAM J. Sci. Comput. 22 (2) (2001) 656–672.
[24] O. Kolb, On the full and global accuracy of a compact third order weno scheme, SIAM J. Numer. Anal. 52 (5) (2014) 2335–2355.
[25] C.-S. Huang, T. Arbogast, C.-H. Hung, A re-averaged weno reconstruction and a third order cweno scheme for hyperbolic conservation laws, J. Comput. Phys. 262 (2014) 291–312.
[26] Y. Guo, T. Xiong, Y. Shi, A maximum-principle-satisfying high-order finite volume compact weno scheme for scalar conservation laws with applications in incompressible flows, J. Sci. Comput. 65 (1) (2015) 83–109.
[27] O. Kolb, On the Full and Global Accuracy of a Compact Third Order Weno Scheme: Part II, Lecture Notes in Computational Science and Engineering, vol. 112, 2016, pp. 53–62.
[28] M. Semplice, A. Coco, G. Russo, Adaptive mesh refinement for hyperbolic systems based on third-order compact weno reconstruction, J. Sci. Comput. 66 (2) (2016) 692–724.
[29] Y. Shi, Y. Guo, A maximum-principle-satisfying finite volume compact-weno scheme for traffic flow model on networks, Appl. Numer. Math. 108 (2016) 21–36.
[30] Q. Zhu, Z. Gao, W. Don, X. Lv, Well-balanced hybrid compact-weno scheme for shallow water equations, Appl. Numer. Math. 112 (2017) 65–78.
[31] J. Zhu, X. Zhong, C.-W. Shu, J. Qiu, Runge-Kutta discontinuous Galerkin method with a simple and compact Hermite weno limiter on unstructured meshes, Commun. Comput. Phys. 21 (3) (2017) 623–649.
[32] K. Cooley, J. Baeder, A central compact-reconstruction weno method for hyperbolic conservation laws, no. 210059, 2018.
[33] P. Li, W.-S. Don, C. Wang, Z. Gao, High order positivity- and bound-preserving hybrid compact-weno finite difference scheme for the compressible Euler equations, J. Sci. Comput. 74 (2) (2018) 640–666.
[34] Y. Guo, Y. Shi, Seventh order compact-weno scheme for hyperbolic conservation laws, Comput. Fluids 176 (2018) 193–209.
[35] V. Titarev, P. Tsoutsanis, D. Drikakis, WENO schemes for mixed-element unstructured meshes, Commun. Comput. Phys. 8 (3) (2010) 585–609.
[36] S. Diot, S. Clain, R. Loubére, 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, https://doi.org/10.1016/j.compfluid.2012.05.004.
[37] S. Clain, S. Diot, R. Loubére, A high-order finite volume method for systems of conservation laws-multi-dimensional optimal order detection (MOOD), J. Comput. Phys. 230 (10) (2011) 4028–4050, https://doi.org/10.1016/j.jcp.2011.02.026.
[38] A. Antoniadis, P. Tsoutsanis, D. Drikakis, Numerical accuracy in rans computations of high-lift multi-element airfoil and aircraft configurations, in: 53rd AIAA Aerospace Sciences Meeting, 2015, 0317.
[39] A. Antoniadis, P. Tsoutsanis, D. Drikakis, High-order schemes on mixed-element unstructured grids for aerodynamic flows, in: 42nd AIAA Fluid Dynamics Conference and Exhibit, 2012, 2833.
[40] A. Antoniadis, P. Tsoutsanis, I. Kokkinakis, Z. Rana, D. Drikakis, Azure: an advanced CFD software suite based on high-resolution and high-order methods, in: 53rd AIAA Aerospace Sciences Meeting, 2015, 0813.
[41] A. Antoniadis, D. Drikakis, I.W. Kokkinakis, P. Tsoutsanis, Z. Rana, High-order methods for hypersonic shock wave turbulent boundary layer interaction flow, in: 20th AIAA International Space Planes and Hypersonic Systems and Technologies Conference, 2015, 3524.
[42] P. Tsoutsanis, I. Kokkinakis, L. Konozsy, D. Drikakis, R. Williams, D. Youngs, Comparison of structured- and unstructured-grid, compressible and incompressible methods using the vortex pairing problem, Comput. Methods Appl. Mech. Eng. 293 (2015) 207–231, https://doi.org/10.1016/j.cma.2015.04.010.
[43] P. Tsoutsanis, H. Srinivasan, Adaptive mesh refinement techniques for high-order finite-volume WENO schemes, in: ECCOMAS Congress 2016, Crete, Greece, 2016.
[44] P. Tsoutsanis, N. Simmonds, A. Gaylard, Implementation of a low-Mach number modification for high-order finite-volume schemes for arbitrary hybrid unstructured meshes, in: ECCOMAS Congress 2016, Crete, Greece, 2016.

[45] P. Tsoutsanis, D. Drikakis, Addressing the challenges of implementation of high-order finite-volume schemes for atmospheric dynamics on unstructured meshes, in: ECCOMAS Congress 2016, Crete, Greece, 2016.
[46] N. Simmonds, P. Tsoutsanis, A. Antoniadis, K. Jenkins, A. Gaylard, Low-Mach number treatment for finite-volume schemes on unstructured meshes, Appl. Math. Comput. 336 (2018) 368–393.
[47] P. Tsoutsanis, Extended bounds limiter for high-order finite-volume schemes on unstructured meshes, J. Comput. Phys. 362 (2018) 69–94.
[48] M. Dumbser, M. Castro, C. Pares, E. Toro, Ader schemes on unstructured meshes for nonconservative hyperbolic systems: applications to geophysical flows, Comput. Fluids 38 (9) (2009) 1731–1748.
[49] G.W. Stewart, Matrix Algorithms, vol. 1: Basic Decompositions, Society for Industrial and Applied Mathematics SIAM, 1998.
[50] A. Jalali, C. Ollivier-Gooch, Higher-order finite volume solution reconstruction on highly anisotropic meshes, in: 21st AIAA Computational Fluid Dynamics Conference, 2013.
[51] X. Nogueira, L. Cueto-Felgueroso, I. Colominas, F. Navarrina, M. Casteleiro, A new shock-capturing technique based on moving least squares for higher-order numerical schemes on unstructured grids, Comput. Methods Appl. Mech. Eng. 199 (37–40) (2010) 2544–2558.
[52] S. Diot, La méthode MOOD la premiére approache a posteriori aux méthodes volumes finis d'ordre trés élevé, PhD Thesis, 2012, pp. 1–191, http://thesesups.ups-tlse.fr/1736/.
[53] P. Tsoutsanis, Very High-Order Methods for 3d Arbitrary Unstructured Grid, PhD Thesis 1, 2009, pp. 1–129, https://dspace.lib.cranfield.ac.uk/handle/1826/4511.
[54] E. Toro, M. Spruce, W. Speares, Restoration of the contact surface in the hll-Riemann solver, Shock Waves 4 (1) (1994) 25–34.
[55] G. Gassner, F. Lorcher, C. Munz, A contribution to the construction of diffusion fluxes for finite volume and discontinuous Galerkin schemes, J. Comput. Phys. 224 (2) (2007) 1049–1063.
[56] A. Jalali, M. Sharbatdar, C. Ollivier-Gooch, Accuracy analysis of unstructured finite volume discretization schemes for diffusive fluxes, Comput. Fluids 101 (2014) 220–232.
[57] H. Nishikawa, Robust and accurate viscous discretisation via upwind scheme-i: basic principle, Comput. Fluids 49 (2011) 62–86.
[58] R.J. Spiteri, S.J. Ruuth, A new class of optimal high-order strong-stability-preserving time-stepping schemes, SIAM J. Numer. Anal. 40 (2) (2002) 469–491.
[59] G. Karypis, V. Kumar, Multilevel k-way partitioning scheme for irregular graphs, J. Parallel Distrib. Comput. 48 (1998) 96–129.
[60] P. Tsoutsanis, Knl performance comparison: Ucns3d, ARCHER performance report, www.archer.ac.uk/community/benchmarks/archer-knl/KNL_perf_UCNS3D.pdf.
[61] N. Petrovskaya, The accuracy of least-squares approximation on highly stretched meshes, Int. J. Comput. Methods 5 (3) (2008) 449–462, https://doi.org/10.1142/S0219876208001558.
[62] A. Syrakos, S. Varchanis, Y. Dimakopoulos, A. Goulas, J. Tsamopoulos, A critical analysis of some popular methods for the discretisation of the gradient operator in finite volume methods, Phys. Fluids 29 (12) (2017), https://doi.org/10.1063/1.4997682.
[63] D. Mavriplis, Revisiting the least-squares procedure for gradient reconstruction on unstructured meshes, in: 16th AIAA Computational Fluid Dynamics Conference, 2003, 3986.
[64] B. Diskin, J. Thomas, E. Nielsen, H. Nishikawa, J. White, Comparison of node-centered and cell-centered unstructured finite-volume discretizations: viscous fluxes, AIAA J. 48 (7) (2010) 1326–1338, https://doi.org/10.2514/1.44940.
[65] B. Diskin, J. Thomas, Comparison of node-centered and cell-centered unstructured finite volume discretizations: inviscid fluxes, AIAA J. 49 (4) (2011) 836–854, https://doi.org/10.2514/1.J050897.
[66] R. Leveque, High-resolution conservative algorithms for advection in incompressible flow, SIAM J. Numer. Anal. 33 (2) (1996) 627–665, https://doi.org/10.1137/0733033.
[67] C.-W. Shu, S. Osher, Efficient implementation of essentially non-oscillatory shock-capturing schemes, ii, J. Comput. Phys. 83 (1) (1989) 32–78.
[68] P. Woodward, P. Colella, The numerical simulation of two-dimensional fluid flow with strong shocks, J. Comput. Phys. 54 (1) (1984) 115–173.
[69] H. Li, G. Ben-Dor, Analysis of double-Mach-reflection wave configurations with convexly curved Mach stems, Shock Waves 9 (1999) 319–326.
[70] A.N. Semenov, M.K. Berezkina, I.V. Krassovskaya, Classification of pseudo-steady shock wave reflection types, Shock Waves 22 (2012) 307–316.
[71] M. Nazarov, A. Larcher, Numerical investigation of a viscous regularization of the Euler equations by entropy viscosity, Comput. Methods Appl. Mech. Eng. 317 (2017) 128–152.
[72] O. San, K. Kara, Evaluation of Riemann flux solvers for weno reconstruction schemes: Kelvin-Helmholtz instability, Comput. Fluids 117 (2015) 24–41.
[73] O. San, K. Kara, Numerical assessments of high-order accurate shock capturing schemes: Kelvin-Helmholtz type vortical structures in high-resolutions, Comput. Fluids 89 (2014) 254–276.
[74] J. Stone, T. Gardiner, P. Teuben, J. Hawley, J. Simon, Athena: a new code for astrophysical mhd, Astrophys. J. 178 (2008) 137–177.
[75] A. Frank, T. Jones, D. Ruy, J. Gaalaas, The magnetohydrodynamic Kelvin-Helmholtz instability: a two-dimensional numerical study, Astrophys. J. 460 (1996) 777–793.
[76] D. Drikakis, C. Fureby, F. Grinstein, D. Youngs, Simulation of transition and turbulence decay in the Taylor-Green vortex, J. Turbul. 8 (2007) 1–12.
[77] J. Bull, A. Jameson, Simulation of the Taylor-Green vortex using high-order flux reconstruction schemes, AIAA J. 53 (9) (2015) 2750–2761.
[78] M. Dumbser, I. Peshkov, E. Romenski, O. Zanotti, High order ADER schemes for a unified first order hyperbolic formulation of continuum mechanics: viscous heat-conducting fluids and elastic solids, J. Comput. Phys. 314 (2016) 824–862.
[79] J.-B. Chapelier, M. de la Llave Plata, E. Lamballais, Development of a multiscale LES model in the context of a modal discontinuous Galerkin method, Comput. Methods Appl. Mech. Eng. 307 (2016) 275–299.
[80] A. Sifounakis, S. Lee, D. You, A conservative finite volume method for incompressible Navier-Stokes equations on locally refined nested Cartesian grids, J. Comput. Phys. 326 (2016) 845–861.
[81] C.-W. Shu, W.-S. Don, D. Gottlieb, O. Schilling, L. Jameson, Numerical convergence study of nearly incompressible, inviscid Taylor-Green vortex flow, J. Sci. Comput. 24 (1) (2005) 569–595.
[82] B. Vermeire, F. Witherden, P. Vincent, On the utility of gpu accelerated high-order methods for unsteady flow simulations: a comparison with industry-standard tools, J. Comput. Phys. 334 (2017) 497–521.
[83] M. Brachet, D. Meiron, B. Nickel, R. Morf, U. Frisch, S. Orszag, Small-scale structure of the Taylor-Green vortex, J. Fluid Mech. 130 (1983) 411–452, https://doi.org/10.1017/S0022112083001159.
[84] A. Kravchenko, P. Moin, Numerical studies of flow over a circular cylinder at re=3900, Phys. Fluids 12 (2) (2000) 403–417, https://doi.org/10.1063/1.870318.
[85] L. Ong, J. Wallace, The velocity field of the turbulent very near wake of a circular cylinder, Exp. Fluids 20 (6) (1996) 441–453, https://doi.org/10.1007/BF00189383.
[86] M. Breuer, Large eddy simulation of the subcritical flow past a circular cylinder: numerical and modeling aspects, Int. J. Numer. Methods Fluids 28 (9) (1998) 1281–1302.
[87] X. Ma, G.-S. Karamanos, G. Karniadakis, Dynamics and low-dimensionality of a turbulent near wake, J. Fluid Mech. 410 (2000) 29–65, https://doi.org/10.1017/S0022112099007934.
[88] J. Kim, H. Choi, Distributed forcing of flow over a circular cylinder, Phys. Fluids 17 (3) (2005) 033103, https://doi.org/10.1063/1.1850151.
[89] J. Franke, W. Frank, Large eddy simulation of the flow past a circular cylinder at re=3900, J. Wind Eng. Ind. Aerodyn. 90 (10) (2002) 1191–1206, https://doi.org/10.1016/S0167-6105(02)00232-5.

[90] A. Beck, T. Bolemann, D. Flad, H. Frank, G. Gassner, F. Hindenlang, C.-D. Munz, High-order discontinuous Galerkin spectral element methods for transitional and turbulent flow simulations, Int. J. Numer. Methods Fluids 76 (8) (2014) 522–548, https://doi.org/10.1002/fld.3943.
[91] D. Lysenko, I. Ertesvag, K. Rian, Large-eddy simulation of the flow over a circular cylinder at Reynolds number 3900 using the openfoam toolbox, Flow Turbul. Combust. 89 (4) (2012) 491–518, https://doi.org/10.1007/s10494-012-9405-0.
[92] J. Ackerman, J. Gostelow, A. Rona, W. Carscallen, Measurements of fluctuating pressures on a circular cylinder in subsonic crossflow, AIAA J. 47 (9) (2009) 2121–2131, https://doi.org/10.2514/1.40954.



https://dspace.org/

Stencil selection algorithms for WENO schemes on unstructured meshes

Tsoutsanis, Panagiotis

2019-08-30
Elsevier
https://doi.org/10.1016/j.jcpx.2019.100037
Tsoutsanis P. (2019) Stencil selection algorithms for WENO schemes on unstructured meshes. Journal of Computational Physics: X, Volume 4, September 2019, Article number 100037

Downloaded from a DSpace repository. https://dspace.org/