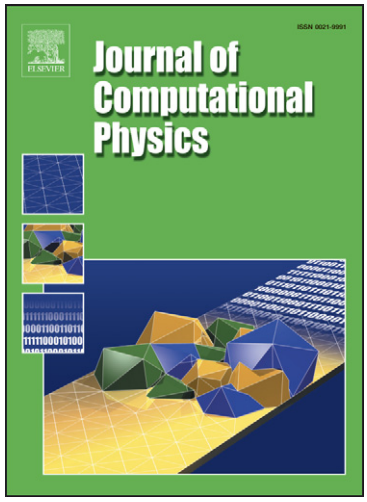
Journal of Computational Physics, Volume 230, Issue 4, 20 February 2011, Pages 1585-1601

Accepted Manuscript



WENO schemes on arbitrary mixed-element unstructured meshes in three space dimensions

P. Tsoutsanis, V.A. Titarev, D. Drikakis

PII: S0021-9991(10)00638-8
DOI: 10.1016/j.jcp.2010.11.023
Reference: YJCPH 3329

To appear in: *Journal of Computational Physics*

Received Date: 27 January 2010
Revised Date: 29 October 2010
Accepted Date: 15 November 2010

Please cite this article as: P. Tsoutsanis, V.A. Titarev, D. Drikakis, WENO schemes on arbitrary mixed-element unstructured meshes in three space dimensions, *Journal of Computational Physics* (2010), doi: 10.1016/j.jcp.2010.11.023

WENO schemes on arbitrary mixed-element unstructured meshes in three space dimensions

P. Tsoutsanis, V.A. Titarev*, D. Drikakis

Department of Fluid Mechanics and Computational Science, Cranfield University, Cranfield, UK, MK43 0AL

Abstract

The paper extends weighted essentially non-oscillatory (WENO) methods to three dimensional mixed-element unstructured meshes, comprising tetrahedral, hexahedral, prismatic and pyramidal elements. Numerical results illustrate the convergence rates and non-oscillatory properties of the schemes for various smooth and discontinuous solutions test cases and the compressible Euler equations on various types of grids. Schemes of up to fifth order of spatial accuracy are considered.

Key words: WENO, unstructured, tetrahedral, hexahedral, prismatic, pyramidal, mixed-element, three space dimensions, very high order, Euler

1. Introduction

Weighted essentially non-oscillatory (WENO) schemes [13, 9] for hyperbolic conservation laws combine the very high order of spatial accuracy in smooth regions of the solution and quasi-monotone behavior at discontinuities. The main idea behind the construction of WENO methods is to combine the low-order reconstruction polynomials with the specially designed weights in such a way that the resulting reconstructed value is of higher order of accuracy than those from each of lower order polynomials for smooth

*Corresponding author

Email addresses: p.tsoutsanis@cranfield.ac.uk (P. Tsoutsanis), v.a.titarev@cranfield.ac.uk, titarev@mail.ru (V.A. Titarev), d.drikakis@cranfield.ac.uk (D. Drikakis)

solutions and in the same time free of spurious oscillations near discontinuities. These properties make WENO methods a good choice for long-time evolution flow problems requiring both shock-capturing ability and uniformly high accuracy in smooth areas with rich structures.

On structured meshes the WENO schemes are relatively easy to implement and computationally not very expensive and thus have been used for a number of studies, e.g. [16, 8]. The extension to unstructured meshes have so far been carried out for triangular/tetrahedral elements only [5, 14, 3, 23]. However, practical applications of high-order methods require the use of mixed-element meshes, consisting of a variety of elements, such as hexahedral and prismatic cells. This is because in the numerical modelling of viscous flows such cells are typically used inside the boundary layers where the rest of the computational domain is discretised by tetrahedrals. Therefore, the development of WENO methods capable of using general mixed-element meshes is vital if this class of schemes is to be applied to real-world problems.

The motivation of the present work is to extend the schemes from [3, 4] to arbitrary three-dimensional mixed-element unstructured meshes, consisting of all four possible element types: hexahedral, tetrahedral, prismatic and pyramidal. The extension is based on a new stencil construction procedure, which is suitable for arbitrarily-shaped cells elements and results in a reduced number of stencils for tetrahedral elements. The presented numerical results for to scalar and nonlinear system cases show that the new schemes achieve the very high order of spatial accuracy across interfaces between cells of different types and in the same time essentially non-oscillatory profiles are produced for discontinuous solutions. Since the goal of the present paper is to develop the basic methodology for the construction of WENO methods on mixed-element meshes, the calculations are limited to Cartesian geometries mostly, except one test case, in which the flow over three-dimensional slender body is considered.

The paper is organised as follows.In Section 2 a detailed explanation of both linear and nonlinear reconstruction for a scalar variable is provided. This reconstruction can be viewed as an extension of the WENO methodology proposed in [3, 4]. The application of the developed techniques to the compressible Euler equations is discussed in Section 3. Section 4 presents numerical results, which demonstrate the very high-order accuracy of the resulting methods, their essentially non-oscillatory properties as well as illustrate the influence of the mesh quality on the accuracy of the calculations. Conclusions are drawn in Section 5.

2. Arbitrary order reconstruction on mixed-element meshes

In this section a reconstruction procedure for mixed-element unstructured meshes in three-space dimensions is described to be used later for the construction of a WENO method. Without loss of generality the idea can be explained as applied to a scalar variable $u(x, y, z)$. Suppose that the spatial computational domain is discretized by conforming elements V_i of the volume $|V_i|$, indexed by a unique mono-index i. The center of the element has coordinates (x_i, y_i, z_i). The elements considered are of hexahedral, tetrahedral, pyramidal and prismatic shapes, as illustrated on Fig. 1. The main goal of the reconstruction procedure is to build high-order polynomial $p_i(x, y, z)$ such that it has the same cell average as u on the target cell V_i

$$\bar{u}_i = \frac{1}{|V_i|} \int\limits_{V_i} u(x, y, z)\, dV = \frac{1}{|V_i|} \int\limits_{V_i} p_j(x, y, z)\, dV \tag{1}$$

and in the same time approximates the point-wise values of u inside the cell with the given order of accuracy r:

$$u(x, y, z) = p_i(x, y, z) + \text{const} \cdot h^r, \quad h \approx |V_i|^{1/3} \tag{2}$$

To build up this polynomial, the reconstruction procedure will use the cell averages of $u(x, y, z)$ on the target cell V_i as well as averages $\bar{u}_m$ from the reconstruction stencil formed by neighboring cells V_m.

In order to simplify the notation, in this section the global spatial index i is omitted and the local numbering of cells is introduced. The reconstruction problem can thus be reformulated as follows: for a target cell V_0 build a high-order polynomial $p(x, y, z)$ so that its spatial average on cell V_0 is equal to u_0 and approximates point-wise values of u with rth order of accuracy.

2.1. Linear (central) reconstruction

In general, the reconstruction can be carried out in the physical coordinates $\mathbf{x} = (x, y, z)$, taking special measures against scaling effects. A more elegant and computationally accurate approach, however, is to use the so-called reference coordinate system (ξ, η, ζ), as was suggested in [3] for triangular (2D) and tetrahedral (3D) elements. Here this transformation technique is extended to deal with general mixed-element mesh elements.

The basic steps of the transformation procedure proposed in the present work can be summarized as follows. Firstly, if the cell V_0 is not tetrahedral, it

is decomposed into tetrahedrons. The number of these tetrahedral depends on the type of element V_0 and is equal to two for a pyramid, three for a prism and finally six for a hexahedral cell. Next, the linear transformation and the corresponding Jacobian matrix from physical coordinate system (x, y, z) into the reference coordinate system (ξ, η, η) are defined using one of the tetrahedral elements resulting from the decomposition. Finally, the cell V_0 and other cells in the reconstruction stencil are transformed into the reference space.

Let $\mathbf{w}_1 = (x_1, y_1, z_1)$, $\mathbf{w}_2 = (x_2, y_2, z_2)$, $\mathbf{w}_3 = (x_3, y_3, z_3)$, $\mathbf{w}_4 = (x_4, y_4, z_4)$ be the four vertices of one of the tetrahedral elements the target element V_0 consists of. The transformation from the Cartesian coordinates (x, y, z) into a reference space (ξ, η, ζ) is defined as

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} x_1 \\ y_1 \\ z_1 \end{pmatrix} + J \cdot \begin{pmatrix} \xi \\ \eta \\ \zeta \end{pmatrix}, \quad J = \begin{bmatrix} x_2 - x_1 & x_3 - x_1 & x_4 - x_1 \\ y_2 - y_1 & y_3 - y_1 & y_4 - y_1 \\ z_2 - z_1 & z_3 - z_1 & z_4 - z_1 \end{bmatrix} \tag{3}$$

with J is the Jacobian matrix If the target cell V_0 is in fact tetrahedral, then (3) reduces to the mapping introduced in [3]. Eq. 3 defines both the direct and inverse mappings from $\boldsymbol{\xi} = (\xi, \eta)$ into $\boldsymbol{x} = (x, y)$, which are denoted as

$$\mathbf{x} = \boldsymbol{x}(\boldsymbol{\xi}), \quad \boldsymbol{\xi} = \boldsymbol{\xi}(\mathbf{x}) \tag{4}$$

Via the inverse mapping the element V_0 can be transformed to the element V_0' in the reference coordinate system. Note that for the uniform (Cartesian) hexahedral mesh the transformed element V_0' is just a unit cube in the reference space (ξ, η, ζ), whereas for a general hexahedral element as well as for prismatic and pyramidal cells four of the vertices of the transformed element will be from the unit cube.

For performing the reconstruction on the target element V_0, the so-called central reconstruction stencil $\mathcal{S}$ is formed which consists of $M + 1$ elements, including the target element V_0:

$$\mathcal{S} = \bigcup_{m=0}^{M} V_m$$

where the local index m counts the elements in the stencil $\mathcal{S}$. The central stencil is build up recursively by adding the direct side neighbors of the element V_0 and all of the elements already existing in the stencil, until the

desired number of elements is reached. The neighbors are added irrespective of their shape. The inverse mapping (4) is then applied to all the elements V_m from the reconstruction stencil $\mathcal{S}$ and the transformed elements and stencil are denoted as E'_m and $\mathcal{S}'$, respectively:

$$\mathcal{S}' = \bigcup_{m=0}^{M} V'_m$$

Fig. 2 shows an examples of a central stencil the physical coordinate system for the linear third order reconstruction; the target cell V_0 has direct neighbors of all four types.

The reconstruction polynomial at the transformed cell V'_0 is sought as an expansion over local polynomial basis functions $\phi_k(\xi, \eta, \zeta)$:

$$p(\xi, \eta, \zeta) = \sum_{k=0}^{K} a_k \phi_k(\xi, \eta, \zeta) = \bar{u}_0 + \sum_{k=1}^{K} a_k \phi_k(\xi, \eta, \zeta) \tag{5}$$

where a_k are degrees of freedom and the upper index in the summation of expansion K is related to the degree of the polynomial r by the expression

$$K = \frac{1}{2}(r+1)(r+2)(r+3) - 1.$$

The basis functions ϕ_k are constructed in such a way that condition (2) is satisfied identically irrespective of values of degrees of freedom:

$$\begin{gathered} \phi_k(\xi, \eta, \zeta) \equiv \psi_k(\xi, \eta, \zeta) - \frac{1}{|V'_0|} \int_{V'_0} \psi_k \, d\xi d\eta d\zeta, \quad k = 1, 2, \dots \\ \{\psi_k\} = \xi, \quad \eta, \quad \zeta, \quad \xi^2, \quad \eta^2, \quad \zeta^2, \quad \xi \cdot \eta, \dots \end{gathered} \tag{6}$$

The unknown degrees of freedom a_k are found by requiring that for each cell V'_m, $m = 1, \dots M$, from the stencil $\mathcal{S}'$ the cell average of the reconstruction polynomial $p(\xi, \eta, \zeta)$ be equal to the cell average of the solution $\bar{u}_m$:

$$\frac{1}{|V'_m|} \int_{V'_m} p(\xi, \eta, \zeta) \, d\xi d\eta d\zeta = \bar{u}_0 + \frac{1}{|V'_m|} \sum_{k=1}^{K} \int_{V'_m} a_k \phi_k \, d\xi d\eta d\zeta = u_m. \tag{7}$$

In general, in order to compute the degrees of freedom a_k we need at least K cells in the stencil, different from the target cell V_0. However, the use of the

minimum possible number of cells in the stencil $M \equiv K$ results in a scheme which may become unstable on general meshes. It is therefore recommended to use more cells in the stencil then the minimal required number [2, 11, 3]. Although it is usually sufficient to use 50% more cells, on mixed-element meshes it may be safer to increase the stencil further. We typically select $M = (1.5 \dots 2) \cdot K$. The resulting over-determined system (7) is solved by means of the least-square procedure.

2.2. WENO reconstruction on mixed-element meshes

The WENO reconstruction proposed in the present work is an extension of the approach from [3] to mixed-element meshes, consisting of elements of arbitrary shapes. The WENO reconstruction stencils is a union of several reconstruction stencils $\mathcal{S}_m$, $m = 0, 1, \dots, m_s$. These are one central stencil and several one-sided, or sectorial, stencils. The construction of the central stencil $\mathcal{S}_0$ was outlined in the previous sections. The sectorial stencils are obtained by adding only those neighboring cells, centres of which lie inside the given sector. In [3] eight sectorial stencils are used for a tetrahedral element, which include four primary sectors and four additional so-called reverse sectors. This is already quite computationally demanding for tetrahedral elements, which have four faces, and becomes too expensive for other type of elements, which have more faces. In the present work a different procedure for defining the directional sectors is adopted, which is approximately two times more efficient. The direction stencils are formed by the barycenter and edges of the faces as illustrated in Fig. 3. Namely, the planes $C12$, $C13$ and $C23$ define the directional stencil sector for the triangular face whereas the planes $C12$, $C23$, $C34$, $C41$ make the sector for the quadrilateral face. Here C is the barycentre of the element. In other words, the number of sectorial stencils in our schemes is usually equal to the number of faces of the cell $m_s = L$. Note, that the number of stencils may be smaller near solid boundaries. Overall, our reconstruction procedure thus uses a significantly smaller number of stencils compared to the original construction [3].

Figs. 4 – 6 illustrate the directional WENO stencils for a particular cell V_0 of the mixed-element unstructured mesh. Alls stencils are plotted in the physical coordinate system and correspond to the central stencil, shown on Fig 2. Here, the target element V_0 is hexahedral and is marked on the plots of directional stencils. Note that there are in total six mixed-element directional stencils, attached to each of six faces of the element V_0.

The WENO reconstruction polynomial is now defined as a non-linear combination of reconstruction polynomials $p_m(\xi,\eta,\zeta)$, obtained by using individual stencils $\mathcal{S}_m$:

$$p_{\text{weno}} = \sum_{m=0}^{m_s} \omega_m p_m(\xi,\eta,\zeta) \tag{8}$$

Substituting the form of the individual polynomial (5) corresponding to the stencil $\mathcal{S}_m$

$$p_m(\xi,\eta,\zeta) = \sum_{k=0}^{K} a_k^{(m)} \phi_k(\xi,\eta,\zeta)$$

and further reordering yields the final expression for p_{weno}:

$$p_{\text{weno}} = \bar{u}_0 + \sum_{k=1}^{K} \left(\sum_{m=0}^{m_s} \omega_m a_k^{(m)} \right) \phi_k(\xi,\eta,\zeta) \equiv \bar{u}_0 + \sum_{k=1}^{K} \tilde{a}_k \phi_k(\xi,\eta,\zeta) \tag{9}$$

Here $\tilde{a}_k$ are the new values of degrees of freedom, modified according to the WENO procedure. As is usual in WENO methods [9, 14], the nonlinear weights ω_m are defined as

$$\omega_m = \frac{\gamma_m}{\sum_{m=0}^{m_s} \gamma_m}, \quad \gamma_m = \frac{d_m}{(\varepsilon + IS_m)^p}$$

where d_m are the so-called linear weights, IS_m are smoothness indicators, ε is a small number used to avoid division by zero and finally p is an integer parameter, controlling how fast the non-linear weights decay for non-smooth stencils. We typically use $\varepsilon = 10^{-6}$ and $p = 4$.

The selection of linear weights d_m is based on [3]. The central stencil is assigned a large linear weight $d_0 = 10^2 \ldots 10^3$ whereas the sectorial stencils are assigned smaller weights $d_m = 1$. This selection of the weights is motivated by the fact that for smooth solutions the central stencil is usually the most accurate one. The oscillation indicators IS_m of each stencil is a measurement of how smooth the solution is on this stencil. Due to the use of the reference coordinate system, scaling is already taken out of the problem and IS_m can be computed in a mesh-independent manner as

$$IS_m = \sum_{1<|\beta|<r} \int_{V_0'} \left(D^\beta p_m(\xi,\eta,\zeta) \right)^2 d\xi d\eta d\zeta \tag{10}$$

where β is a multi-index, r is the order of the polynomial and D is the derivative operator. The general form of B in three space dimensions can be found in [3]. It is easily seen that the smoothness indicators (10) are quadratic functions of the degrees of freedom $a_k^{(m)}$ and thus the expression (10) can be rewritten in terms of the so-called universal oscillation indicator matrix [3]. For purely tetrahedral meshes this matrix does not depend on the element. For general elements it will, however, depend on the element. For efficiency, it can be precomputed and stored at the beginning of the calculations for each element V_i.

3. Application to the compressible Euler equations

3.1. The framework

The developed WENO reconstruction technique is applied to solve numerically the three-dimensional compressible Euler equations of the form

$$\frac{\partial}{\partial t}\mathbf{U} + \frac{\partial}{\partial x}\mathbf{F}(\mathbf{U}) + \frac{\partial}{\partial y}\mathbf{G}(\mathbf{U}) + \frac{\partial}{\partial z}\mathbf{H}(\mathbf{U}) = \mathbf{0}, \tag{11}$$

where $\mathbf{U}$ is the vector of conserved variables, $\mathbf{F}$, $\mathbf{G}$, $\mathbf{H}$ are flux vectors in x, y and z coordinate directions respectively, given by

$$\mathbf{U} = \begin{pmatrix} \rho \\ \rho u \\ \rho v \\ \rho w \\ E \end{pmatrix}, \quad \mathbf{F} = \begin{pmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ \rho uw \\ (E+p)u \end{pmatrix}, \tag{12}$$

$$\mathbf{G} = \begin{pmatrix} \rho v \\ \rho vu \\ \rho v^2 + p \\ \rho vw \\ (E+p)v \end{pmatrix}, \quad \mathbf{H} = \begin{pmatrix} \rho w \\ \rho wu \\ \rho wv \\ \rho w^2 + p \\ (E+p)w \end{pmatrix}. \tag{13}$$

Here ρ is density, u,v,w velocity components in the x, y and z directions, respectively, p pressure, $E = p/(\gamma-1) + (1/2)\rho(u^2+v^2+w^2)$ total energy, γ is the ratio of specific heats; $\gamma = 1.4$ is used throughout.

Spatial integration of (11) over a mesh element V_i yields the following semi-discrete finite-volume method:

$$\begin{gathered} \frac{d}{dt}\mathbf{U}_i + \frac{1}{|V_i|}\oint\limits_{\partial V_i} \mathbf{F}_n dA = \mathbf{0}, \\ \mathbf{F}_n\left(\mathbf{U}\right) = \mathbf{F}\left(\mathbf{U}\right) n_x + \mathbf{G}\left(\mathbf{U}\right) n_y + \mathbf{H}\left(\mathbf{U}\right) n_z, \end{gathered} \tag{14}$$

where $\mathbf{n} = (n_x, n_y, n_z)$ is outward unit normal vector to cell faces, $\mathbf{U}_i(t)$ are the cell averages of the solution at time t, $\mathbf{F}_n$ - projection of the flux tensor on the normal direction. The integral over the element boundary ∂V_i is split into the sum of integrals over each face A_j resulting in the following expression:

$$\frac{d}{dt}\mathbf{U}_i = \mathbf{R}_i, \quad \mathbf{R}_i = -\frac{1}{|V_i|}\sum_{j=1}^{L}\int\limits_{A_j} \mathbf{F}_{n,j} dA = -\frac{1}{|V_i|}\sum_{j=1}^{L}\mathbf{K}_{ij} \tag{15}$$

Here the numerical flux $\mathbf{K}_{ij}$ corresponding to the face j of the cell V_i is the surface integral of the projection of the tensor of fluxes onto the outward unit vector $\mathbf{n}_j$ for face A_j. In a numerical method the exact integral expression for the numerical flux $\mathbf{K}_{ij}$ for the face j of a cell V_i is approximated by a suitable Gaussian numerical quadrature:

$$\mathbf{K}_{ij} = \int\limits_{A_j} \mathbf{F}_{n,j} dA = \sum_{\beta} \mathbf{F}_{n,j}\left(\mathbf{U}(\mathbf{x}_\beta, t)\right) \omega_\beta |A_j| \tag{16}$$

where the subscript β corresponds to different Gaussian integration points $\mathbf{x}_\beta$ and weights ω_β over the face A_j.

The temporal derivative in (14) is approximated by the third-order TVD Runge-Kutta method [9]:

$$\begin{aligned} \mathbf{U}_i^{(n+1/3)} &= \mathbf{U}_i^n + \Delta t\, \mathbf{R}_i(\mathbf{U}^n), \\ \mathbf{U}_i^{(n+2/3)} &= \frac{3}{4}\mathbf{U}_i^n + \frac{1}{4}\mathbf{U}_i^{(n+1/3)} + \frac{1}{4}\Delta t\, \mathbf{R}_i(\mathbf{U}^{(n+1/3)}), \\ \mathbf{U}_i^{n+1} &= \frac{1}{3}\mathbf{U}_i^n + \frac{2}{3}\mathbf{U}_i^{(n+2/3)} + \frac{2}{3}\Delta t\, \mathbf{R}_i(\mathbf{U}^{(n+2/3)}). \end{aligned} \tag{17}$$

The time step Δt is selected according to the formula

$$dt = K \min_i \frac{h_i}{S_i} \tag{18}$$

where S_i is an estimate of the maximum (in absolute value) propagation speed in cell V_i, $K \leq 1/3$ is the CFL number, h_i is the characteristic length of the element V_i.

The description of the scheme is complete once a reconstruction procedure to calculate the point-wise values from cell averages and a numerical flux (building block) of the scheme are specified.

3.2. Reconstruction for systems

Calculation of a numerical flux (16) through the face A_j of a cell V_i requires the knowledge of point-wise values of the conserved vector $\mathbf{U}$ at the Gaussian points. These values are obtained by the WENO reconstruction procedure on mixed-element meshes, developed in the previous section and extended here in to vector variables, which are solutions of the compressible Euler equations. The reconstruction produces the high-order vector polynomials $\mathbf{P}_i(\xi, \eta, \zeta)$ defined in the local reference coordinate system of each element V_i. The simplest approach to the construction of these polynomials is to apply the scalar reconstruction procedure (9) to each component of the conserved vector $\mathbf{U}$. However, for systems better results are obtained if the reconstruction procedure is carried out in characteristic variables. Our approach for extending the scalar reconstruction to the characteristic-based reconstruction is very similar to that of [4], although different in some respects, and thus we only outline the main steps.

Consider the cell V_i and the corresponding set of directional stencils in the local reference coordinate system $\{S'_m\}$, $m = 0, 1, \dots m_s$. The vector of degrees of freedom $\mathbf{A}_{ik}^{(m)}$ for each stencil are calculated using the scalar reconstruction procedure in the component-wise fashion. Then, the corresponding polynomials are given by

$$\mathbf{P}_{im}(\xi, \eta, \zeta) = \sum_{k=0}^{K} \mathbf{A}_{ik}^{(m)} \phi_{ik}(\xi, \eta, \zeta) = \bar{\mathbf{U}}_i + \sum_{k=1}^{K} \mathbf{A}_{ik}^{(m)} \phi_{ik}(\xi, \eta, \zeta), \qquad (19)$$

where ϕ_{ik} are basis functions for cell V_i in the local reference coordinate system.

Define at the arithmetic average of the conserved vector $\mathbf{U}_i$ and the conserved vector $\mathbf{U}_{i'}$, corresponding to the computational cell, adjacent to the face A_j of the current cell V_i:

$$\mathbf{U}'_n = \frac{1}{2}(\mathbf{U}_i + \mathbf{U}_{i'}).$$

Let $\mathbf{R}_j$, $\mathbf{L}_j$ be the matrices containing the right and left eigenvectors of the Jacobian matrix $\mathbf{H}_j$, corresponding to the normal projection of the flux tensor calculated at this average state

$$\mathbf{H}_j = \frac{\partial \mathbf{F}_n}{\partial \mathbf{U}}|_{\mathbf{U}=\mathbf{U}'_n},$$

where $\mathbf{F}_n$ is defined in eq. (14). The characteristic projections of vector degrees of freedom of each stencil S_m, including the cell averaged value $\mathbf{U}_I$, are computed as

$$\mathbf{B}^{(m)}_{ikj} = \mathbf{L}_j \mathbf{A}^{(m)}_{ik}, \quad m = 0, \dots, m_s, \quad k = 0, \dots K.$$

The scalar WENO reconstruction algorithm is applied to each component of the projected degrees of freedom. The resulting modified degrees of freedom $\tilde{\mathbf{B}}^{(m)}_{ikj}$ are projected back to by multiplying them by $\mathbf{R}_j$. The WENO reconstruction polynomial for the face A_j is then given by

$$\mathbf{P}_{ij}(\xi, \eta, \zeta) = \bar{\mathbf{U}}_i + \sum_{k=1}^{K} \tilde{\mathbf{A}}_{ikj} \phi_{ik}(\xi, \eta, \zeta), \quad \tilde{\mathbf{A}}_{ikj} = \mathbf{R}_j \mathbf{B}_{ikj}. \tag{20}$$

Note, that the degrees of freedom in eq. (20) depend on the face index j. Finally, the reconstructed values at Gaussian integration points are then given by

$$\mathbf{P}_i(\xi_\beta, \eta_\beta, \zeta_\beta) = \bar{\mathbf{U}}_i + \sum_{k=1}^{K} \tilde{\mathbf{A}}_{ikj} \phi_{ik}(\xi_\beta, \eta_\beta, \zeta_\beta), \tag{21}$$

where (ξ_β, η_β, ζ_β) are the coordinates of Gaussian points in the reference coordinate system for the face A_j of the cell V_i. This completes the reconstruction step.

An additional step in the reconstruction process was used in [4]. Namely, for each cell V_i the least oscillatory of all $\mathbf{P}_{ij}$ is taken as the unique reconstruction polynomial $\mathbf{P}_i$ and then used for all faces. In the present work this part of the characteristic-wise reconstruction process is omitted in order to reduce the computational cost.

3.3. Numerical flux

After the reconstruction is carried out, for each computational cell the point-wise values of the conserved vector $\mathbf{U}$ are represented by high-order

reconstruction polynomials. Since these polynomials are different, at each Gaussian point β in the expression for the numerical flux (16) for the face A_j of cell V_i two approximate values for the conserved vector $\mathbf{U}$ exist. The first value $\mathbf{U}_\beta^-$ corresponds to the spatial limit to the cell boundary from inside the cell V_i and is given by the reconstruction polynomial $\mathbf{P}_i$. The second value $\mathbf{U}_\beta^+$ corresponds to the spatial limit from outside the element and is obtained by using the reconstruction polynomial of the neighboring element $V_{i'}$. The values $\mathbf{U}_\beta^\pm$ are usually called left and right boundary extrapolated values. In upwind Godunov-type methods the resulting discontinuity is resolved by replacing the physical flux at each Gaussian integration point by a monotone function of left and right boundary extrapolated values so that (22) can be rewritten as

$$\mathbf{K}_{ij} \approx \sum_\beta \mathbf{F}_{n,j}\left(\mathbf{U}_\beta^-, \mathbf{U}_\beta^+\right) \omega_\beta |A_j| \tag{22}$$

The function $\tilde{\mathbf{F}}_{n,j}\left(\mathbf{U}_\beta^-, \mathbf{U}_\beta^+\right)$ is called the Riemann solver [6], or a building block of a high-order scheme.

Review of existing exact and approximate Riemann solvers for various hyperbolic systems can be found in [12, 19]. Using the concept of the rotational invariance [19], for each face A_j the normal projection of the flux tensor $\mathbf{F}_{n,j}$ is replaced by

$$\mathbf{F}_{n,j} = \mathbf{T}^{-1}\mathbf{F}\left(\mathbf{T}_j \mathbf{U}\right) \tag{23}$$

where $\mathbf{T}_j$ is the (constant) rotation matrix for face j. Then the expression (22) for $\mathbf{K}_{ij}$ is rewritten as

$$\mathbf{K}_{ij} = \sum_\beta \mathbf{F}_{n,j}\left(\mathbf{U}_\beta^-, \mathbf{U}_\beta^+\right) \omega_\beta |A_j| = \sum_\beta \mathbf{T}^{-1}\mathbf{F}\left(\hat{\mathbf{U}}_L, \hat{\mathbf{U}}_R\right) \omega_\beta |A_j| \tag{24}$$

where $\hat{\mathbf{U}}_j$ is the rotated conserved variable and

$$\hat{\mathbf{U}}_L = \mathbf{T}_j \mathbf{U}_\beta^-, \quad \hat{\mathbf{U}}_R = \mathbf{T}_j \mathbf{U}_\beta^+$$

It follows from (24) that the flux function for the Gaussian point β can be computed from the augmented one-dimensional Riemann problem

$$\frac{\partial}{\partial t}\hat{\mathbf{U}} + \frac{\partial}{\partial s}\hat{\mathbf{F}} = \mathbf{0}, \quad \hat{\mathbf{F}} = \mathbf{F}(\hat{\mathbf{U}}), \quad \hat{\mathbf{U}}(s,0) = \begin{cases} \hat{\mathbf{U}}_L, & s < 0, \\ \hat{\mathbf{U}}_R, & s > 0 \end{cases} \tag{25}$$

Assuming a three-wave structure with wave speed estimates S_L, S_* and S_R the HLLC flux is given by

$$\hat{\mathbf{F}}^{HLLC} = \begin{cases} \hat{\mathbf{F}}_L, & \text{if} \quad 0 \le S_L\,, \\ \hat{\mathbf{F}}_{*L} = \hat{\mathbf{F}}_L + S_L(\hat{\mathbf{U}}_{*L} - \hat{\mathbf{U}}_L)\,, & \text{if} \quad S_L \le 0 \le S_*\,, \\ \hat{\mathbf{F}}_{*R} = \hat{\mathbf{F}}_R + S_R(\hat{\mathbf{U}}_{*R} - \hat{\mathbf{U}}_R)\,, & \text{if} \quad S_* \le 0 \le S_R\,, \\ \hat{\mathbf{F}}_R, & \text{if} \quad 0 \ge S_R\,, \end{cases} \tag{26}$$

where

$$\hat{\mathbf{U}}_{*K} = \rho_K \left(\frac{S_K - u_K}{S_K - S_*}\right) \begin{bmatrix} 1 \\ S_* \\ v_K \\ w_K \\ \dfrac{E_K}{\rho_K}(S_* - u_K)[S_* + \dfrac{p_K}{\rho_K(S_K - u_K)}] \end{bmatrix}$$

for $K = L$ and $K = R$. The wave speeds S_L, S_* and S_R must be estimated. Here the procedure for pressure-velocity estimates of Sect. 10.5.2 of [19] are used.

It is worth noting that HLLC flux contains all waves in the Riemann problem solution, does not use linearization of the equations and works well for low-density problems and sonic points without any fixes. The HLLC flux has been recently used in a number of very high-order methods, with good results, see e.g. [22, 18, 21].

4. Numerical examples

In this section the numerical results of the new schemes are presented as applied to both smooth and discontinuous solutions in three space dimensions of the schemes of up to fifth order accuracy. The schemes of spatial order r are denoted as WENO$-r$, e.g. the spatially fifth order scheme is denoted as WENO-5. Since only third-order accurate time evolution method (17) is used, the resulting methods are of only third order of formal accuracy. For the numerical schemes up to 3^{rd}-order of spatial accuracy all convergence tests are run with a fixed Courant number $K = 0.3$. However for higher than

third order numerical schemes for convergence studies with smooth solutions the time-step size is reduced according to (27)

$$\Delta t = K \cdot \Delta^{r/3} \tag{27}$$

where Δ is a characteristic cell size, r stands for the order of the scheme for $r > 3$. This allows the spatial-order of accuracy to dominate the computation. If uniformly fourth and high order of accuracy is required under the conventional time-step restriction, then an ADER-type time evolution approach is recommended [20, 21, 3, 4, 15].

One of the advantages of explicit methods used here is that they can be easily parallelized based on domain decomposition. The mesh decomposition process is the preprocessor step which is one of the most essential elements for an equal load balance between processes in terms of memory requirements and communications between processes. In the present work it is carried out using the METIS software package [10], which has been shown to work well for very high order accurate unstructured solvers on tetrahedral meshes [4]. The process chosen for decomposition is to convert the mesh into a nodal graph rather than using a dual graph in order to derive a partitioning of the nodes. The load balancing achieved with the METIS software package for the meshes used ranges between 1.00 and 1.07 for any type of unstructured mesh. of the scheme employed. The calculations were carried out on the HPC Facility 'Astral' of the Cranfield university. Astral is a Hewlett Packard HPC, comprising 856 Intel Woodcrest cores with 2Gb of RAM per core and clock frequency of 3.0GHz. With the present implementation overall parallel efficiency of our methods is above 90% for up to 128 processors used.

4.1. Linear advection with a smooth initial condition

The methods are applied to the linear constant-coefficient advection equation

$$\frac{\partial u}{\partial t} + \frac{\partial u}{\partial x} + \frac{\partial u}{\partial y} + \frac{\partial u}{\partial z} = 0 \tag{28}$$

with a smooth initial condition defined in a cube $[0, 1]^3$

$$u_0(x, y, z) = \sin(2\pi x) \cdot \sin(2\pi y) \cdot \sin(2\pi z) \tag{29}$$

The periodic boundary conditions are applied. The unstructured meshes used in calculations are constructed as follows. First the number of cells N_{edge} over each edge of the cube is specified. Then, for each N_{edge} four

sequences of meshes are constructed, corresponding to purely hexahedral mesh, purely tetrahedral mesh, purely prismatic mesh and a mixed-element mesh, consisting of cells of all element types. The purely hexahedral mesh is further perturbed in order to make it non-uniform. The total number of computational cells in the domain is then denoted as N_{tot} and is a function of both N_{edge} and the type of elements used inside the computational domain. Table 1 provides a summary of the mesh composition for each mesh sequence. Cutaway sections for the case N_{edge} are shown on Fig. 7.

Tables 2 – 5 shows the convergence studies for cell averages for the solution. The errors are calculated the output time $t = 1$ Both linear (central stencil) and non-linear WENO schemes are employed. It is observed that all schemes reach the designed order of accuracy. Note that hexahedral and prismatic meshes $N_{edge} = 10$ do not have enough elements to employ fifth-order schemes since the stencils extend more than half a period in length, leading to incorrect computational results.

The results from the third-order schemes, contained in Tables 2 – 5, makes it possible to study the relative efficiency of different types of unstructured meshes. It is obvious that for a given number of boundary cells N_{edge} the hexahedral mesh leads to less accurate results are compared to meshes made of elements with fewer number of nodes: tetrahedral and prismatic ones. However, in practical applications the numerical error should be taken as a function of the computational time, which is in turn proportional to the total number of cells N_{tot}, the number of time steps N_t, required to reach the selected output time and the computational cost required to compute the numerical fluxes for one cell. Generally speaking, for the WENO schemes the hexahedral element is costlier for flux calculations than tetrahedral, prismatic or pyramidal one due to the larger number of sectorial stencils used. However, it also allows one to run with a significantly large time step. For the linear schemes with the central stencil the cost of the reconstruction for different cell types is very comparable, whereas the differences in time step size remain significant.

It is obvious that for any given value of N_{edge}, the prismatic and tetrahedral meshes are superior to hexahedral meshes in terms of L_1 and L_∞ error norms. This is due to the fact that prismatic and tetrahedral cells are smaller than the corresponding hexahedral ones, resulting in finer mesh resolution, but also leading to 4 to 14 times large computational times for the same value of N_{edge}. If instead the computational error is taken to be function of the total number of elements and the number of time steps required to reach

the output time, then the hexahedral meshes are slightly more efficient than any other elements.

4.2. Two-dimensional vortex evolution problem

The three-dimensional Euler equations (11)–(13) are solved in the computational domain $[0,10]\times[0,10]\times[0,0.5]$ with periodic boundary conditions. The initial condition corresponds to a smooth two-dimensional vortex placed at the centre of the x-y plane $[5,5]$. The vortex is defined as the following isentropic perturbation of unit values of primitive variables [1]:

$$\begin{gathered} u = \frac{\varepsilon}{2\pi} e^{\frac{1-r^2}{2}(5-y)}, \quad v = \frac{\varepsilon}{2\pi} e^{\frac{1-r^2}{2}(x-5)}, \quad w = 0, \\ T = \frac{(\gamma-1)\,\varepsilon^2}{8\gamma\pi^2} e^{\left(1-r^2\right)}, \quad \frac{p}{\rho^\gamma} = 1, \quad r^2 = (x-5)^2 + (y-5)^2 \end{gathered}, \tag{30}$$

where the vortex strength is $\varepsilon = 5$. The exact solution is a vortex movement in the $x-y$ plane with a constant velocity at 45^o to the Cartesian axis. The numerical solution is computed at the output time $t = 10$ (one period) for which the vortex returns to the initial position. Periodic boundary conditions are applied.

The construction of the meshes for this test case is similar to the linear advection convergence study. The edges of the computational in x, y plane are mesh with N_{edge} number of cells, whereas the z direction is discretized with a fixed number of cell equal to 20. Then, the interior is meshed using cells of different types. The summary of the mesh statistics can be found in Table 6.

Tables 7 – 10 show errors and convergence rates in L_1 and L_∞ norm for cell averages of density. It is observed that both linear and WENO schemes achieve the expected convergence rates for all meshes, including the mixed-element ones. For a fixed resolution the fifth-order schemes are more accurate than the third order ones by one to two orders of magnitude.

4.3. Blunted-cone-cylinder-flare test case

Finally, the third-order WENO method is applied to calculate the supersonic flow over a realistic three-dimensional geometry. We consider the so-called the blunted-cone-cylinder-flare geometry, designated HB-2. The geometry has been used extensively in aerodynamic test facilities [7]. The

corresponding high-speed flow is a suitable test of the robustness of the proposed methods as applied to real-life applications.

The geometry of the problem is shown in Fig. 8, where d is the reference length. In the computational setup the x axis is directed along the body. The computational domain (including the wake region) was meshed by two hybrid unstructured meshes of different resolution. Figure 9 illustrate schematically how the computational mesh was constructed. Although most of the computational cells are hexahedral, near the axis the mesh contains prisms and is thus of mixed-element type. In actual computations the mesh consisted of 690040 cells and was partitioned into 32 blocks for parallel computations, see Figure 10.

The computations were performed for the free-stream Mach number equal to 5 at zero angle of attack Fig. 11 shows the pressure distribution along the body, normalized by the post-shock stagnation pressure p_0 and plotted against longitudinal position. The spatial coordinate is normalised by the length of the body L. Also shown is experimental data from [7] (symbols). It is observed that the numerical solution provided by the WENO-3 method agrees very well with the experimental measurements, including fine details. Fig. 12 shows the pressure distribution, computed on the same mesh using a TVD method. It is seen that the TVD method provides lower accuracy even though the flow is dominated by relatively flow structures, such as shock waves.

5. Conclusions

This paper has focused on the development of three-dimensional WENO schemes for arbitrary-element unstructured meshes. The extension is based on a new stencil construction procedure, which is suitable for arbitrarily-shaped cells elements. A detailed description of the reconstruction step of schemes is provided allowing for its practical implementation by a reader.

The new schemes have been applied to a number of well-established test problems with both smooth and discontinuous solutions. The presented numerical results demonstrate that the new schemes achieve the designed order of accuracy for hexahedral, tetrahedral, prismatic and pyramidal elements, which is maintained across interfaces between cells of various types. Also, the computation of discontinuous solutions is not affected by interfaces between neighbouring cells of different types.

The future developments will consist of two directions. First is the extensions to the ADER approach [20, 21, 3, 4, 15], which will allow an arbitrary high-order accurate time evolution as opposed to the third-order accurate TVD Runge-Kutta method used in the present work. Second direction is the incorporation of the second-order derivatives. The preliminary results for the compressible Navier-Stokes equations can be found in [17], where the linear third-order method is presented. In the non-linear method the WENO reconstruction, developed in the present work, should be used for computing the convective flux only whereas the viscous and heat conduction terms are still computed using the linear central stencil reconstruction. The corresponding results will be reported elsewhere.

Acknowledgments

The authors would like to thank Dr. Michael Dumbser, University of Trento, Italy, for various discussions concerning very high order methods on unstructured meshes.

References

[1] Balsara, D., Shu, C.-W., 2000. Monotonicity preserving weighted essentially non-oscillatory schemes with increasingly high order of accuracy. J. Comput. Phys. 160, 405–452.

[2] Barth, T., Frederickson, P., 1990. Higher order solution of the Euler equations on unstructured grids using quadratic reconstruction. In: AIAA paper no. 90-0013, 28th Aerospace Sciences Meeting.

[3] Dumbser, M., Käser, M., 2007. Arbitrary high order non-oscillatory finite volume schemes on unstructured meshes for linear hyperbolic systems. Journal of Computational Physics 221 (2), 693–723.

[4] Dumbser, M., Käser, M., Titarev, V., Toro, E., 2007. Quadrature-free non-oscillatory finite volume schemes on unstructured meshes for nonlinear hyperbolic systems. Journal of Computational Physics 226, 204–243.

[5] Friedrichs, O., 1998. Weighted essentially non-oscillatory schemes for the interpolation of mean values on unstructured grids. Journal of Computational Physics 144 (1), 194–212.

[6] Godunov, S., 1959. A finite difference method for the computation of discontinuous solutions of the equations of fluid dynamics. Mat. Sbornik 47, 357–393.

[7] Gray, J. D., 1964. Summary Report on Aerodynamic Characteristics of Standard Models HB-1 and HB-2. In: Technical Report, AEDC-TDR-64-137.

[8] Hahn, M., Drikakis, D., 2009. Assessment of large-eddy simulation of internal separated flow. Journal of Fluids Engineering 131 (7), 071201.

[9] Jiang, G., Shu, C., 1996. Efficient implementation of weighted ENO schemes. J. Comput. Phys. 126, 202–212.

[10] Karypis, G., Kumar, V., 1998. Multilevel k-way partitioning scheme for irregular graphs. J. Parallel Distrib. Comput. 48, 96–129.

[11] Käser, M., Iske, A., 2004. ADER schemes for the solution of conservation laws on adaptive triangulations. In: Iske, A., Randen, T. (Eds.), Mathematical Methods and Modeling in Hydrocarbon Exploration and Production, Series : Mathematics in Industry. Vol. 7. Springer-Verlag.

[12] Kulikovskii, A., Pogorelov, N., Semenov, A., 2002. Mathematical Aspects of Numerical Solution of Hyperbolic Systems. Chapman and Hall, Monographs and Surveys in Pure and Applied Mathematics, Vol. 118.

[13] Liu, X., Osher, S., Chan, T., 1994. Weighted essentially non-oscillatory schemes. J. Comput. Phys. 115, 200–212.

[14] Shi, J., Hu, C., Shu, C.-W., 2002. A technique for treating negative weights in WENO schemes. J. Comput. Phys. 175, 108–127.

[15] Taube, A., Dumbser, M., Balsara, D., Munz, C.-D., 2007. Arbitrary high-order Discontinuous Galerkin schemes for the magnetohydrodynamic equations. Journal of Scientific Computing 30 (3), 441–464.

[16] Thornber, B., Mosedale, A., Drikakis, D., 2007. On the implicit large eddy simulations of homogeneous decaying turbulence. Journal of Computational Physics 226, 1902–1929.

[17] Titarev, V., Drikakis, D., Uniformly high-order schemes on arbitrary unstructured meshes for advection-diffusion equations. Computers and Fluids, in press.

[18] Titarev, V., Toro, E., 2004. Finite-volume WENO schemes for three-dimensional conservation laws. J. Comput. Phys. 201 (1), 238–260.

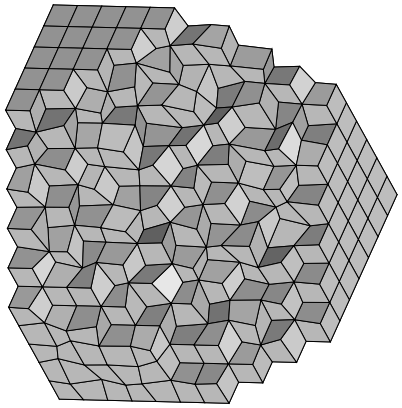
[19] Toro, E., 2009. Riemann solvers and numerical methods for fluid dynamics, 3rd Edition. Springer-Verlag.

[20] Toro, E., Millington, R., Nejad, L., 2001. Towards very high order Godunov schemes. In: Toro, E. F. (Ed.), Godunov Methods. Theory and Applications. Kluwer/Plenum Academic Publishers, pp. 907–940.

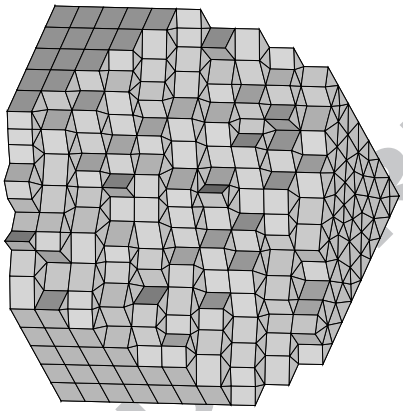
[21] Toro, E., Titarev, V., 2006. Derivative Riemann solvers for systems of conservation laws and ADER methods. J. Comput. Phys. 212 (1), 150–165.

[22] van der Vegt, J., van der Ven, H., 2002. Space-time discontinuous Galerkin finite element method with dynamic grid motion for inviscid compressible flows: I. General formulation. J. Comput. Phys. 182 (2), 546–585.

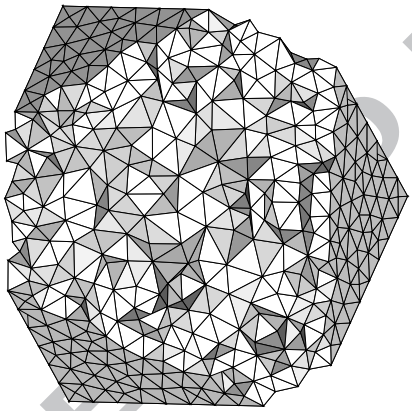
[23] Zhang, Y.-T., Shu, C.-W., 2009. Third order WENO scheme on three dimensional tetrahedral meshes. Communications in Computational Physics 5 (2-4), 836–848.



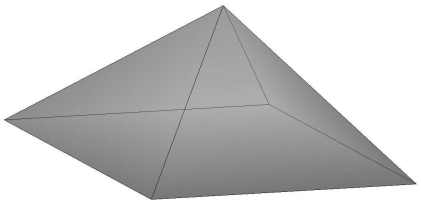
(a) Hexahedral



(b) Prismatic



(c) Tetrahedral



(d) Pyramidal

Figure 1: Four element shapes considered in the preset work

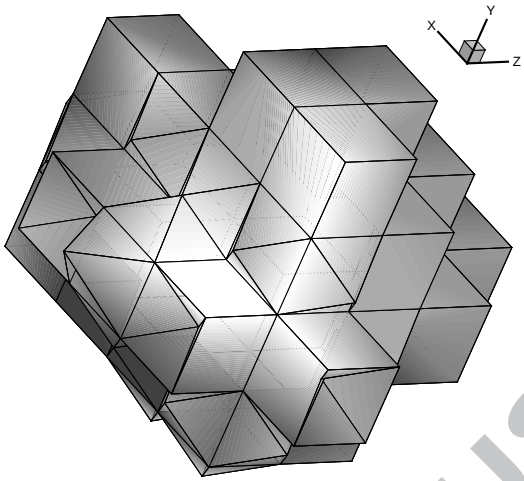


Figure 2: An example of a central stencil of the third-order spatial reconstruction.

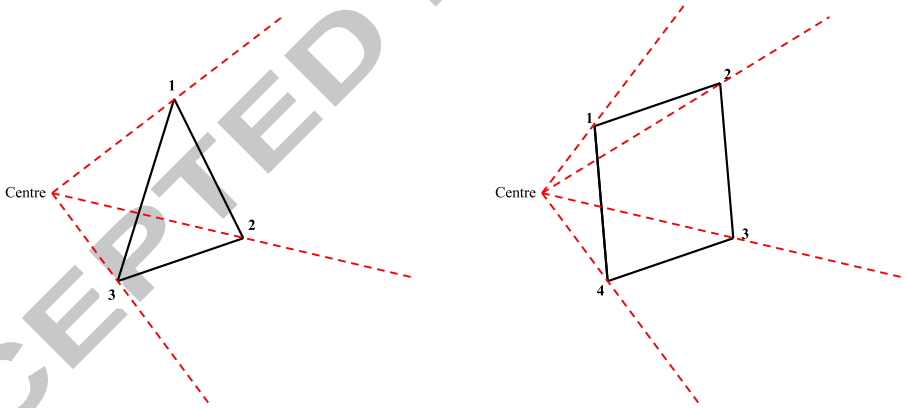


Figure 3: Construction of direction planes for triangular (left) and quadrilateral (right) faces.

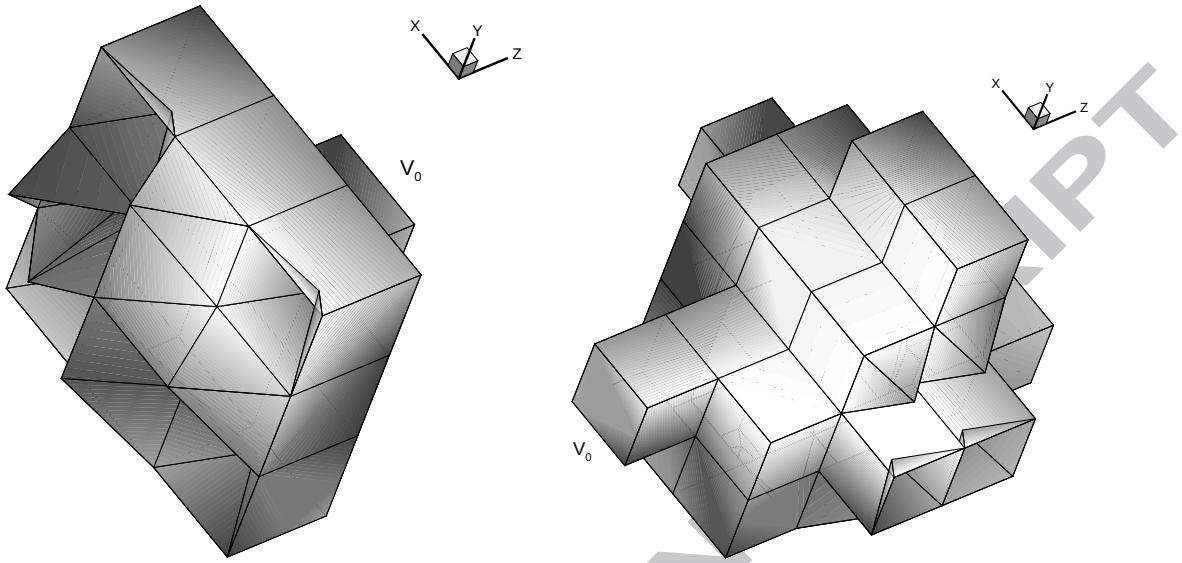


Figure 4: First and second directional WENO stencils

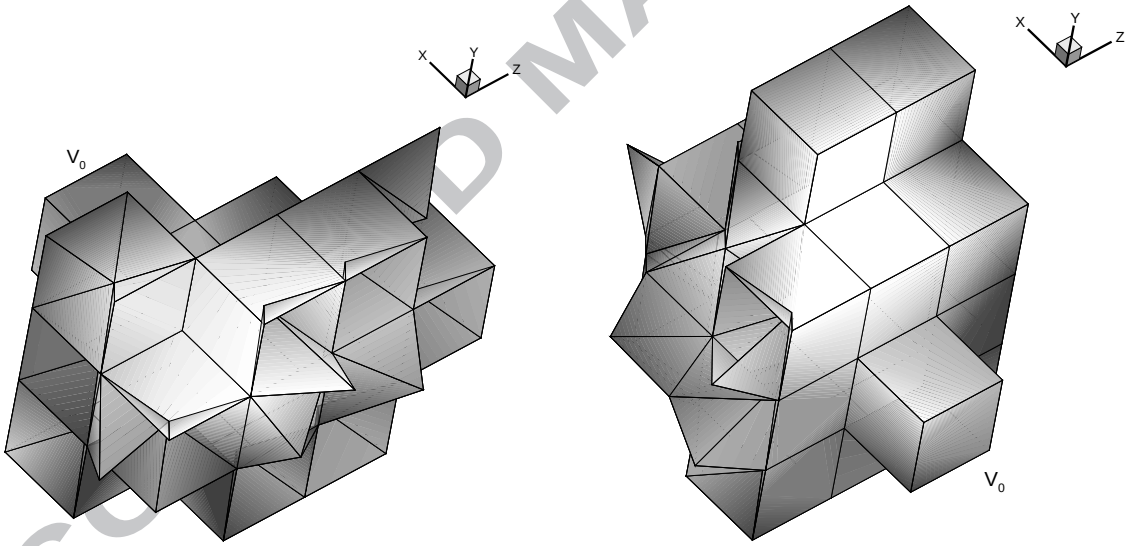


Figure 5: Third and fourth directional WENO stencils

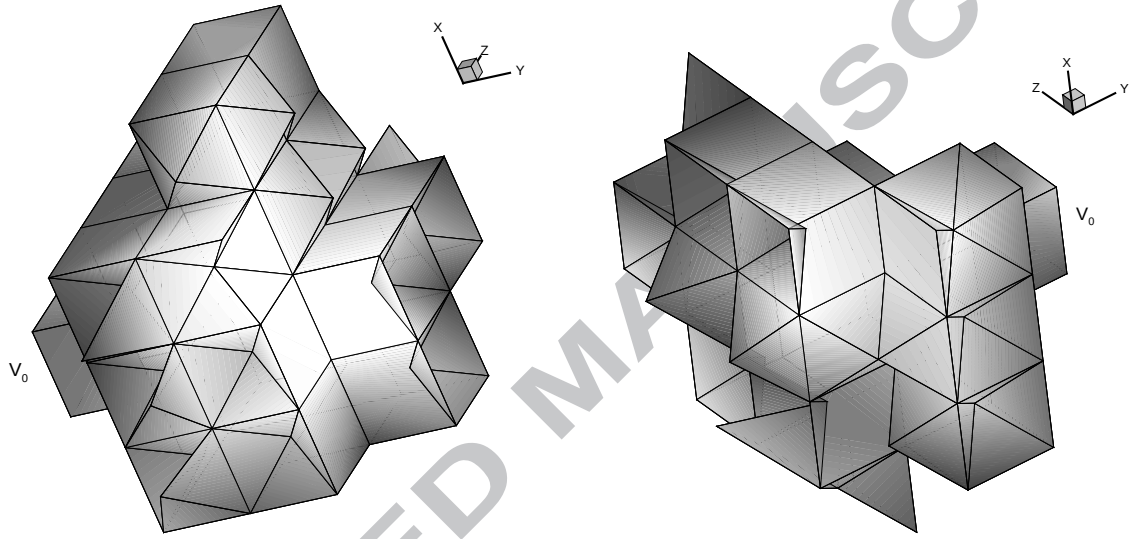
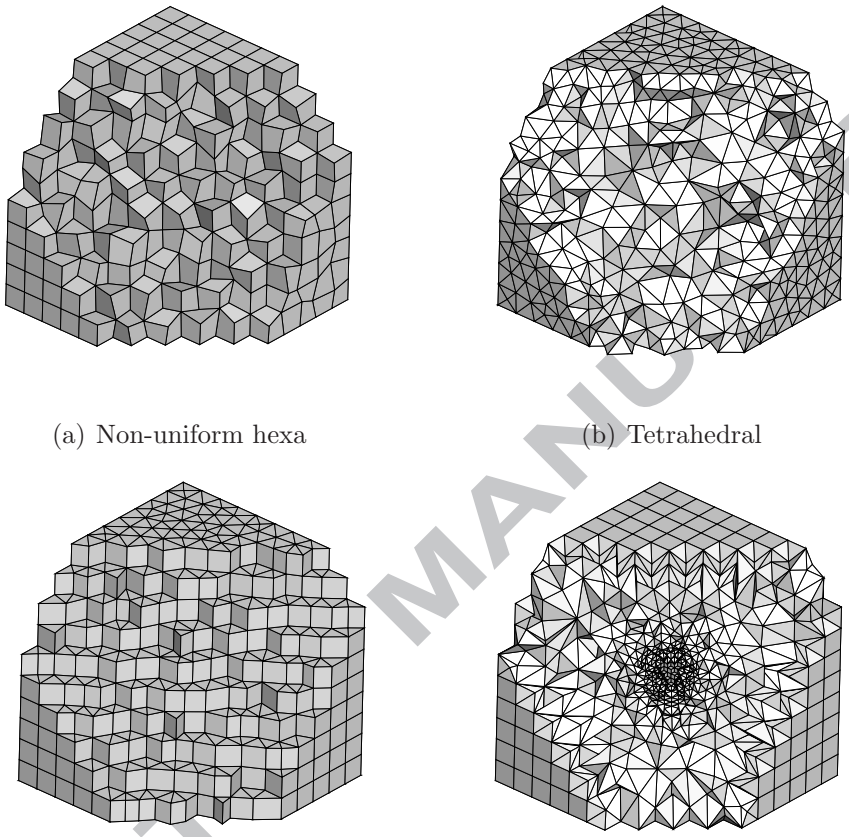


Figure 6: Fifth and sixth directional WENO stencils



(a) Non-uniform hexa

(b) Tetrahedral

(c) Prismatic

(d) Mixed-element

Figure 7: Cutaway sections of different types of unstructured meshes used for convergence study of the model equation (28) with initial conditions (29)

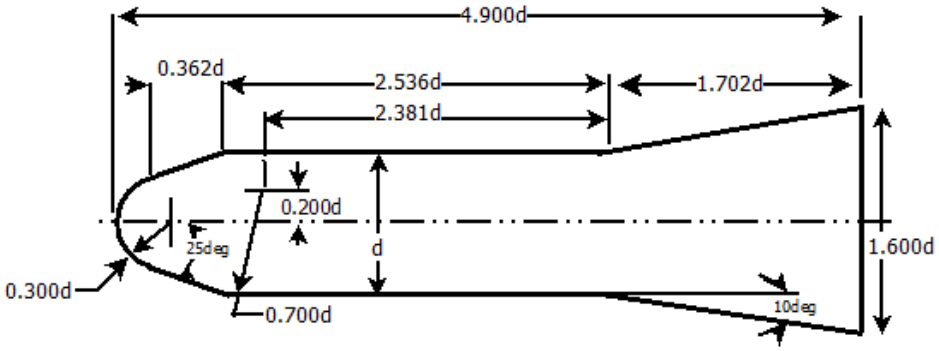


Figure 8: Geometry for for the blunted-cone-cylinder-flare test case.

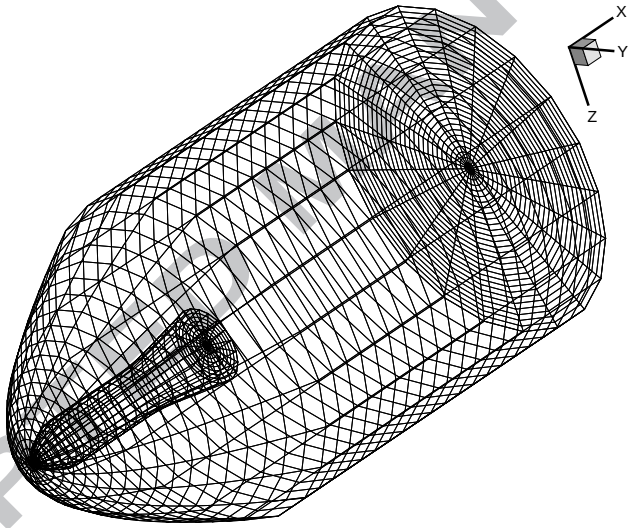


Figure 9: Computational domain and schematics of the mesh construction for the blunted-cone-cylinder-flare test case.

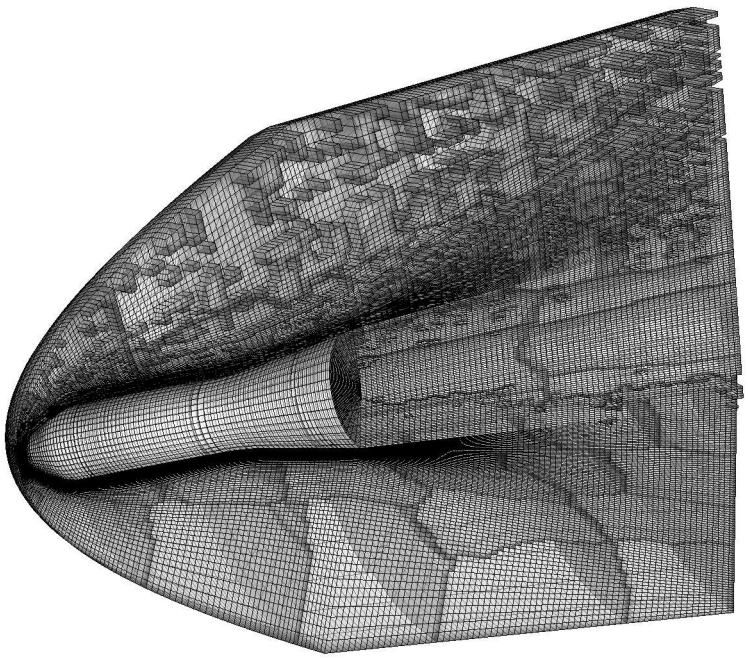


Figure 10: The fine mesh for the blunted-cone-cylinder-flare test case.

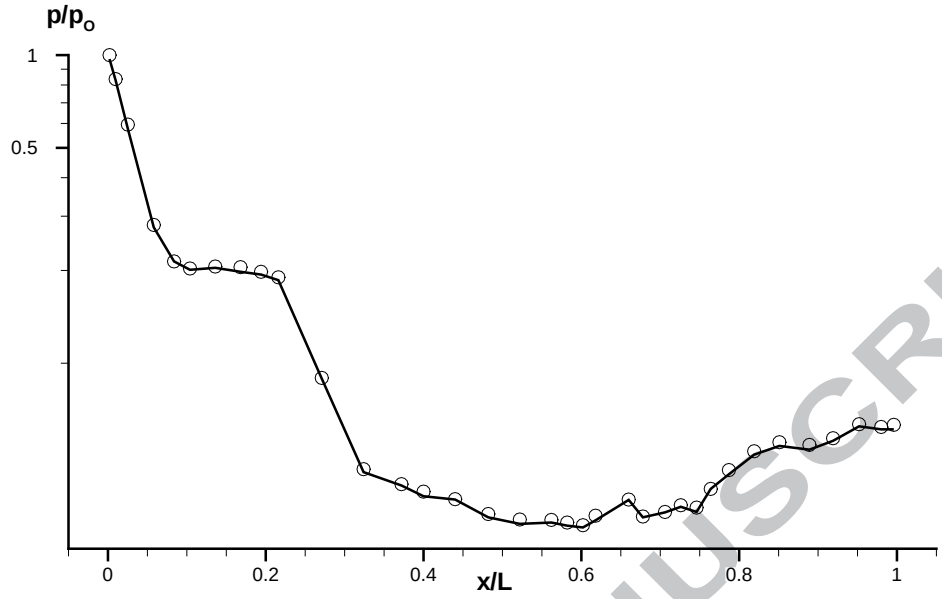


Figure 11: The pressure distribution for the blunted-cone-cylinder-flare test case, computed with WENO-3 method.

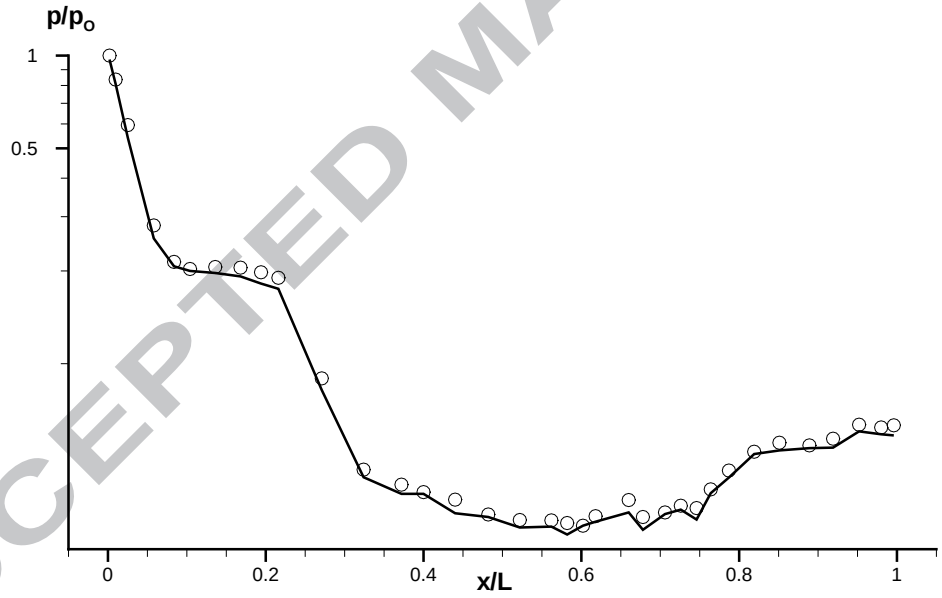


Figure 12: The pressure distribution for the blunted-cone-cylinder-flare test case, computed with TVD method.

Mesh Type	N_{edge}	N_{tetra}	N_{pyra}	N_{prism}	N_{hexa}	N_{tot}
Hexa	20	0	0	0	8000	8000
	40	0	0	0	64000	64000
	80	0	0	0	512000	512000
Tetra	20	84669	0	0	0	84669
	40	507337	0	0	0	507337
	80	3830397	0	0	0	3830397
Prism	20	0	0	28000	0	28000
	40	0	0	219200	0	219200
	80	0	0	1760480	0	1760480
Hybrid	20	218016	40000	16000	8000	282016
	40	1629213	160000	128000	64000	1981213
	80	4844598	640000	1024000	512000	7020598

Table 1: Statistics for the meshes used for the computations of the model equation (28) with initial conditions (29).

Method	N_{edge}	L_1 error	L_1 order	L_∞ error	L_∞ order
Linear-3	20	1.39×10^{-2}	2.857	4.66×10^{-2}	2.962
	40	1.59×10^{-3}	3.127	5.55×10^{-3}	3.059
	80	1.89×10^{-4}	3.072	7.67×10^{-4}	2.865
Linear-5	20	8.56×10^{-4}	N/S	2.09×10^{-3}	N/S
	40	3.34×10^{-5}	4. 679	6.89×10^{-5}	4. 920
	80	1.08×10^{-6}	4.950	1.95×10^{-6}	5.145
WENO-3	20	9.43×10^{-3}	2.579	3.67×10^{-2}	2.622
	40	1.55×10^{-3}	2.604	5.88×10^{-3}	2.641
	80	1.96×10^{-4}	2.983	8.23×10^{-4}	2.836
WENO-5	20	7.73×10^{-4}	N/S	3.47×10^{-3}	N/S
	40	2.69×10^{-5}	4.844	1.27×10^{-4}	4.772
	80	1.03×10^{-6}	4.706	4.23×10^{-6}	4.908

Table 2: Convergence study for various schemes using a unstructured hexahedral mesh as applied to the model equation (28) with initial conditions (29) at output time $t = 1$.

Method	N_{edge}	L_1 error	L_1 order	L_∞ error	L_∞ order
Linear-3	20	9.85×10^{-2}	2.844	3.983×10^{-2}	2.958
	40	9.73×10^{-4}	3.339	6.47×10^{-3}	2.620
	80	1.29×10^{-4}	2.915	7.26×10^{-4}	3.155
Linear-5	20	7.54×10^{-4}	N/S	1.53×10^{-3}	N/S
	40	2.57×10^{-5}	4.874	5.39×10^{-5}	4.828
	80	9.38×10^{-7}	4.776	1.79×10^{-6}	4.912
WENO-3	20	6.93×10^{-3}	2.671	2.49×10^{-2}	2.542
	40	9.27×10^{-4}	2.902	4.73×10^{-3}	2.396
	80	1.41×10^{-4}	2.716	6.77×10^{-4}	2.804
WENO-5	20	6.99×10^{-4}	N/S	1.96×10^{-3}	N/S
	40	2.36×10^{-5}	4.888	8.55×10^{-5}	4.518
	80	9.71×10^{-7}	4.603	2.69×10^{-6}	4.990

Table 3: Convergence study for various schemes using a unstructured prismatic mesh as applied to the model equation (28) with initial conditions (29) at output time $t = 1$.

Method	N_{edge}	L_1 error	L_1 order	L_∞ error	L_∞ order
Linear-3	20	4.32×10^{-3}	2.942	1.66×10^{-2}	2.746
	40	5.96×10^{-4}	2.857	2.38×10^{-3}	2.802
	80	7.12×10^{-5}	3.065	2.91×10^{-4}	3.031
Linear-5	20	1.93×10^{-4}	4.991	7.56×10^{-4}	4.322
	40	8.67×10^{-6}	4.476	2.72×10^{-5}	4.796
	80	2.99×10^{-7}	4.853	1.06×10^{-6}	4.681
WENO-3	20	1.96×10^{-3}	2.899	8.67×10^{-2}	2.541
	40	2.54×10^{-4}	2.947	1.03×10^{-3}	3.073
	80	3.75×10^{-5}	2.759	1.36×10^{-4}	2.918
WENO-5	20	3.72×10^{-4}	4.712	8.63×10^{-4}	4.963
	40	1.69×10^{-5}	4.460	2.97×10^{-5}	4.860
	80	5.33×10^{-7}	4.986	8.94×10^{-7}	5.053

Table 4: Convergence study for various schemes using a unstructured tetrahedral mesh as applied to the model equation (28) with initial conditions (29) at output time $t = 1$.

Method	N_{edge}	L_1 error	L_1 order	L_∞ error	L_∞ order
Linear-3	20	7.96×10^{-4}	2.892	5.78×10^{-3}	2.642
	40	9.23×10^{-5}	3.108	6.99×10^{-4}	3.047
	80	1.17×10^{-5}	2.979	8.32×10^{-5}	3.070
Linear-5	20	4.85×10^{-5}	4.692	2.61×10^{-4}	4.518
	40	1.44×10^{-6}	5.073	8.67×10^{-6}	4.911
	80	4.57×10^{-8}	4.977	2.55×10^{-7}	5.087
WENO-3	20	9.29×10^{-3}	2.638	6.75×10^{-3}	2.778
	40	1.17×10^{-4}	2.990	7.63×10^{-4}	3.145
	80	1.41×10^{-5}	3.050	9.41×10^{-5}	3.019
WENO-5	20	5.62×10^{-5}	4.887	3.43×10^{-4}	4.925
	40	2.35×10^{-6}	4.578	1.08×10^{-5}	4.784
	80	7.56×10^{-8}	4.959	3.62×10^{-7}	4.898

Table 5: Convergence study for various schemes using a hybrid unstructured mesh as applied to the model equation (28) with initial conditions (29) at output time $t = 1$.

Mesh Type	N_{edge}	N_{tetra}	N_{pyra}	N_{prism}	N_{hexa}	N_{tot}
Hexahedral	40	0	0	0	8000	8000
	80	0	0	0	64000	64000
	160	0	0	0	512000	512000
Prismatic	40	110800	0	0	0	110800
	80	440840	0	0	0	440840
	160	1761280	0	0	0	1761280
Tetrahedral	40	0	0	51844	0	51844
	80	0	0	350153	0	350153
	160	0	0	2085769	0	2085769
Hybrid	40	97343	39217	0	8000	137360
	80	351405	109755	0	64000	525160
	160	1041529	486391	0	512000	2039920

Table 6: Statistics for the meshes used for the vortex evolution problem.

Method	Mesh	L_1 error	L_1 order	L_∞ error	L_∞ order
Linear-3	$40^2 \times 20$	1.87×10^{-1}	-	2.45×10^{-1}	-
	$80^2 \times 20$	2.61×10^{-2}	2.837	2.93×10^{-2}	3.061
	$160^2 \times 20$	3.26×10^{-3}	3.001	3.6×10^{-3}	3.027
Linear-5	$40^2 \times 20$	1.24×10^{-1}	-	1.49×10^{-1}	-
	$80^2 \times 20$	3.57×10^{-3}	5.118	4.68×10^{-3}	4. 992
	$160^2 \times 20$	1.08×10^{-4}	5.046	1.47×10^{-4}	4.984
WENO-3	$40^2 \times 20$	1.67×10^{-1}	-	1.96×10^{-1}	-
	$80^2 \times 20$	2.31×10^{-2}	2.853	2.98×10^{-2}	2.717
	$160^2 \times 20$	2.52×10^{-3}	3.196	3.76×10^{-3}	2.986
WENO-5	$40^2 \times 20$	8.65×10^{-2}	-	9.64×10^{-1}	-
	$80^2 \times 20$	2.33×10^{-3}	5.214	3.29×10^{-3}	4.872
	$160^2 \times 20$	8.14×10^{-5}	4.839	9.33×10^{-5}	5.140

Table 7: Convergence study for various schemes using a uniform hexahedral mesh as applied to the vortex evolution problem (30) at output time $t = 10$.

Method	N	L_1 error	L_1 order	L_∞ error	L_∞ order
Linear-3	$40^2 \times 20$	1.96×10^{-1}	-	2.51×10^{-1}	-
	$80^2 \times 20$	2.87×10^{-2}	2.771	3.35×10^{-2}	2.905
	$160^2 \times 20$	4.41×10^{-3}	2.702	4.63×10^{-3}	2.855
Linear-5	$40^2 \times 20$	1.18×10^{-1}	-	1.28×10^{-1}	-
	$80^2 \times 20$	2.86×10^{-3}	5.366	4.17×10^{-3}	4.939
	$160^2 \times 20$	1.11×10^{-4}	4.687	1.56×10^{-4}	4.740
WENO-3	$40^2 \times 20$	1.78×10^{-1}	-	2.57×10^{-1}	-
	$80^2 \times 20$	1.95×10^{-2}	3.190	3.11×10^{-2}	3.046
	$160^2 \times 20$	2.54×10^{-3}	2.864	4.07×10^{-3}	2.933
WENO-5	$40^2 \times 20$	7.99×10^{-2}	-	1.27×10^{-1}	-
	$80^2 \times 20$	2.46×10^{-3}	5.021	4.514×10^{-3}	4.814
	$160^2 \times 20$	7.22×10^{-5}	5.090	1.28×10^{-5}	5.140

Table 8: Convergence study for various schemes using a prismatic mesh as applied to the vortex evolution problem (30) at output time $t = 10$.

Method	N	L_1 error	L_1 order	L_∞ error	L_∞ order
Linear-3	$40^2 \times 20$	2.18×10^{-1}	-	2.84×10^{-1}	-
	$80^2 \times 20$	3.01×10^{-2}	2.857	3.42×10^{-2}	3.051
	$160^2 \times 20$	5.20×10^{-3}	2.533	5.49×10^{-3}	2.641
Linear-5	$40^2 \times 20$	1.41×10^{-1}	-	1.53×10^{-1}	-
	$80^2 \times 20$	2.92×10^{-3}	5.605	4.71×10^{-3}	5.024
	16062×20	1.19×10^{-4}	4.604	1.59×10^{-4}	4.890
WENO-3	$40^2 \times 20$	1.31×10^{-1}	-	2.93×10^{-1}	-
	$80^2 \times 20$	2.13×10^{-2}	3.299	3.26×10^{-2}	3.168
	$160^2 \times 20$	2.80×10^{-3}	2.927	4.49×10^{-3}	2.860
WENO-5	$40^2 \times 20$	8.63×10^{-2}	-	1.44×10^{-1}	-
	$80^2 \times 20$	2.71×10^{-3}	4.989	4.80×10^{-3}	4.096
	$160^2 \times 20$	7.7×10^{-5}	5.141	1.49×10^{-4}	5.014

Table 9: Convergence study for various schemes using a tetrahedral mesh as applied to the vortex evolution problem (30) at output time $t = 10.0$.

Method	N	L_1 error	L_1 order	L_∞ error	L_∞ order
Linear-3	$40^2 \times 20$	1.73×10^{-1}	-	2.28×10^{-1}	-
	$80^2 \times 20$	1.99×10^{-2}	3.120	2.86×10^{-2}	2.994
	$160^2 \times 20$	2.74×10^{-3}	2.864	3.42×10^{-3}	3.063
Linear-5	$40^2 \times 20$	9.38×10^{-2}	-	1.11×10^{-1}	-
	$80^2 \times 20$	2.89×10^{-3}	5.019	4.05×10^{-3}	4.767
	$160^2 \times 20$	9.23×10^{-5}	4.970	1.23×10^{-4}	5.036
WENO-3	$40^2 \times 20$	1.59×10^{-1}	-	1.53×10^{-1}	-
	$80^2 \times 20$	1.97×10^{-2}	3.010	2.80×10^{-2}	2.457
	$160^2 \times 20$	1.92×10^{-3}	3.357	3.71×10^{-3}	2.916
WENO-5	$40^2 \times 20$	6.53×10^{-2}	-	7.12×10^{-1}	-
	$80^2 \times 20$	2.22×10^{-3}	4.870	3.27×10^{-3}	4.442
	$160^2 \times 20$	5.83×10^{-5}	5.258	8.03×10^{-5}	5.350

Table 10: Convergence study for various schemes using a hybrid mesh as applied to the vortex evolution problem (30) at output time $t = 10.0$.