LA-UR-14-23047 (Accepted Manuscript)

A High-Order Vertex-Based Central ENO Finite-Volume Scheme for Three-Dimensional Compressible Flows

Charest, Marc R.
Canfield, Thomas R.
Morgan, Nathaniel R.
Waltz, Jacob I.
Wohlbier, John G.

Provided by the author(s) and the Los Alamos National Laboratory (2015-08-12).

To be published in: COMPUTERS & FLUIDS, Vol.114, p.172-192, JUL 2 2015.

DOI to publisher's version: 10.1016/j.compfluid.2015.03.001

Permalink to record: http://permalink.lanl.gov/object/view?what=info:lanl-repo/lareport/LA-UR-14-23047

Disclaimer:
Approved for public release. Los Alamos National Laboratory, an affirmative action/equal opportunity employer, is operated by the Los Alamos National Security, LLC for the National Nuclear Security Administration of the U.S. Department of Energy under contract DE-AC52-06NA25396. Los Alamos National Laboratory strongly supports academic freedom and a researcher's right to publish; as an institution, however, the Laboratory does not endorse the viewpoint of a publication or guarantee its technical correctness.

A High-Order Vertex-Based Central ENO Finite-Volume Scheme for Three-Dimensional Compressible Flows

Marc R. J. Charest*, Thomas R. Canfield, Nathaniel R. Morgan, Jacob Waltz, John G. Wohlbier

Los Alamos National Laboratory, P.O. Box 1663, Los Alamos, NM, 87545

Abstract

High-order discretization methods offer the potential to reduce the computational cost associated with modelling compressible flows. However, it is difficult to obtain accurate high-order discretizations of conservation laws that do not produce spurious oscillations near discontinuities, especially on multi-dimensional unstructured meshes. A novel, high-order, central essentially non-oscillatory (CENO) finite-volume method that does not have these difficulties is proposed for tetrahedral meshes. The proposed unstructured method is vertex-based, which differs from existing cell-based CENO formulations, and uses a hybrid reconstruction procedure that switches between two different solution representations. It applies a high-order k-exact reconstruction in smooth regions and a limited linear reconstruction when discontinuities are encountered. Both reconstructions use a single, central stencil for all variables, making the application of CENO to arbitrary unstructured meshes relatively straightforward. The new approach was applied to the conservation equations governing compressible flows and assessed in terms of accuracy and computational cost. For all problems considered, which included various function reconstructions and idealized flows, CENO demonstrated excellent reliability and robustness. Up to fifth-order accuracy was achieved in smooth regions and essentially non-oscillatory solutions were obtained near discontinuities. The high-order schemes were also more computationally efficient for high-accuracy solutions, i.e., they took less wall time than the lower-order schemes to achieve a desired level of error. In one particular case, it took a factor of 24 less wall-time to obtain a given level of error with the fourth-order CENO scheme than to obtain the same error with the second-order scheme.

Keywords: Numerical Algorithms; Computational Fluid Dynamics; High-Order Methods; Compressible Flows; Shock Hydrodynamics

1. Introduction

Finite-volume methods are a popular discretization technique for computational fluid dynamics, especially for compressible flows. Numerous formulations exist, and one of the main differing characteristics is the approach used to discretize the computational domain. Either cell- or vertex-based discretizations are typically employed. Cell-based approaches apply conservation laws to the individual elements or cells of the mesh, whereas vertex-based approaches apply them to control volumes constructed surrounding the vertices of the mesh. The choice is not always straightforward, since both techniques are widely used and both have their respective advantages/disadvantages. For example, vertex-based schemes are often favored for use with unstructured tetrahedral meshes since there are approximately 5 to 6 times fewer vertices than elements. But because there are more elements than vertices in a tetrahedral mesh, cell-based schemes have more degrees of freedom. As such, cell-based schemes tend to be slightly more accurate on tetrahedral meshes, although this increased accuracy comes at the expense of additional computational effort [1]. The direct comparison between the two types of schemes is complicated, however, because they

*Corresponding author. Email: `charest@lanl.gov`
LANL report no. LA-UR-14-23047

both use different stencils. The larger, denser stencils that vertex-based schemes use are more robust and accurate per degree of freedom, which may make vertex centered schemes more computationally efficient for a given accuracy [1]. Nonetheless, whatever the chosen finite-volume formulation, current production codes rely mostly on standard first- or second-order accurate discretization schemes. These discretizations are often not practical for physically-complex, multi-dimensional flows with disparate scales as they tend to exhibit excessive numerical dissipation.

High-order discretization methods for conservation laws have the potential to significantly reduce the cost of modelling physically-complex flows. They offer improved numerical efficiency to obtain high-resolution solutions since fewer computational cells are required to achieve a desired level of accuracy [2]. However, this potential is challenging to fully realize as it is difficult to obtain accurate and robust discretizations of hyperbolic conservation laws near discontinuities [3]. Although there are many different high-order schemes for both structured and unstructured meshes that address this issue [3–31], there is still no consensus on the most robust, efficient, and accurate scheme that deals with the aforementioned issues and is also universally applicable to arbitrary meshes.

One promising high-order discretization is the central essentially non oscillatory (CENO) finite-volume approach [32–40]. It was originally developed for two-dimensional structured meshes by Ivan et al. [36–40] and then extended to three-dimensional unstructured meshes by Charest et al. [32–35]. In all formulations, CENO remained both accurate and robust throughout a variety of physically-complex flows. This robustness is provided by a hybrid reconstruction procedure that switches between two algorithms: an unlimited high-order k-exact reconstruction in smooth regions, and a monotonicity-preserving limited piecewise linear reconstruction in regions with discontinuities or shocks. Switching between the two reconstructions is facilitated by a smoothness indicator that measures the ability of the of the k-exact reconstruction to locally resolve the flow. Fixed central stencils are used for both reconstruction algorithms, which makes its extension to arbitrary unstructured meshes straightforward.

Essentially non-oscillatory (ENO) [5, 41] and weighted ENO (WENO) [10, 11, 42–45] finite-volume schemes that can provide accurate and robust solutions on unstructured meshes already exist. However, the CENO approach avoids many of the complexities associated with these ENO and WENO schemes because it does not require a high-order reconstruction on multiple stencils. ENO and WENO have difficulty selecting stencils on general multi-dimensional unstructured meshes [5, 6, 9, 46], and some of these stencils produce poorly conditioned linear systems for solution reconstruction [9, 46].

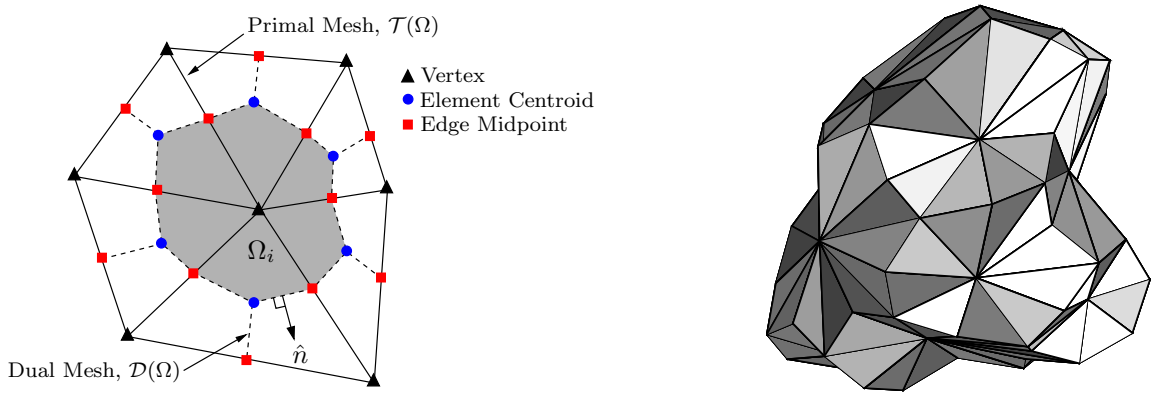
The existing CENO formulations for structured [36–40] and unstructured [32–35] meshes were developed for cell-based finite-volume schemes only. In the present research, CENO was extended to a vertex-based finite-volume discretization for three-dimensional unstructured meshes and applied to solve the equations governing compressible flows. The resulting algorithm was applied to various function reconstructions as well as steady and unsteady flows and then analyzed with respect to accuracy and computational cost. This research was performed using Chicoma, a computational framework for compressible fluid flow, i.e., shock hydrodynamics [47–53]

2. Governing Equations

The Euler equations governing compressible fluid flow were considered for the present research. In three space dimensions, these partial-differential equations (PDEs) are given by

$$\frac{\partial}{\partial t}\mathbf{U}(\mathbf{W}) + \vec{\nabla} \cdot \vec{\mathbf{F}}(\mathbf{W}) = \mathbf{S}(\mathbf{W}) \tag{1}$$

where t is the time, $\mathbf{U}$ and $\mathbf{W}$ are the vectors of conserved and primitive variables, respectively, $\vec{\mathbf{F}}(\mathbf{W}) = [\mathbf{E}, \mathbf{F}, \mathbf{G}]$ is the inviscid solution flux dyad, and $\mathbf{S}(\mathbf{W})$ is a vector of source terms. These terms are defined



(a) Primal and dual mesh (shown in two-dimensions for simplicity).

(b) Example of a three-dimensional control volume.

Fig. 1. Computational mesh and local control volume configuration.

as

$$\mathbf{U} = \left[\rho,\ \rho u,\ \rho v,\ \rho w,\ \rho e_{\mathrm{t}}\right],$$
$$\mathbf{W} = \left[\rho,\ u,\ v,\ w,\ e\right],$$
$$\mathbf{E} = \begin{bmatrix} \rho u \\ \rho u^2 + p \\ \rho u v \\ \rho u w \\ u(\rho e_{\mathrm{t}} + p) \end{bmatrix},\ \mathbf{F} = \begin{bmatrix} \rho v \\ \rho v u \\ \rho v^2 + p \\ \rho v w \\ v(\rho e_{\mathrm{t}} + p) \end{bmatrix},\ \mathbf{G} = \begin{bmatrix} \rho w \\ \rho w u \\ \rho w v \\ \rho w^2 + p \\ w(\rho e_{\mathrm{t}} + p) \end{bmatrix}$$

where ρ is the fluid density, p is the pressure, $\vec{v} = (u, v, w)$ is the fluid velocity vector, e is the internal energy, and e_{t} is the total energy. The total energy is the sum of the internal and kinetic energies, i.e.,

$$e_{\mathrm{t}} = e + \frac{1}{2}(u^2 + v^2 + w^2) \tag{2}$$

Internal energy is related to pressure and density through the following relation for an ideal gas:

$$e = \frac{p}{\rho(\gamma - 1)} \tag{3}$$

where γ is the ratio of specific heats. The sound speed of an ideal gas is given by

$$a = \sqrt{\gamma p / \rho} \tag{4}$$

Although Eqs. (3) and (4) describe an ideal gas, the numerical formulation described herein is designed to support arbitrary analytic or tabular equations of state. For all test cases considered, $\gamma = 1.4$.

The source term vector, $\mathbf{S}$, is typically treated as zero throughout this work. There is one particular case — which will be discussed in the following sections — where it was used to generate a known solution for verification purposes.

3. CENO Finite-Volume Scheme

In the proposed vertex-based finite-volume approach, the physical domain, Ω, was discretized into non-overlapping, finite-sized control volumes, Ω_i, such that

$$\Omega = \cup \Omega_i \tag{5}$$
$$\Omega_i \cap \Omega_j = \varnothing \quad \text{for} \quad i \neq j \tag{6}$$

Table 1. Gauss quadrature rules used for integrating over triangles and tetrahedrons.

Reconstruction	**Number of Points**		**Degree of Precision**
	Triangle	Tetrahedron	
Constant ($k = 0$)	1	1	1
Linear ($k = 1$)	1	1	1
Quadratic ($k = 2$)	3	4	2
Cubic ($k = 3$)	4	8	3
Quartic ($k = 4$)	6	14	4

The individual control volumes were formed by constructing the median dual, $\mathcal{D}(\Omega)$, of a three-dimensional triangulation of the domain, $\mathcal{T}(\Omega)$, which is illustrated in two-dimensions in Fig. 1(a). Only primal meshes composed of tetrahedral elements were considered, and they have a corresponding dual mesh composed of complex polyhedrons with triangular faces. A control volume surrounding a vertex i, Ω_i, was constructed from the polyhedron whose vertices are the centroids of incident tetrahedra and triangles, plus the mid-points of incident edges. A sample control volume surrounding an individual vertex in a three-dimensional tetrahedral mesh is illustrated in Fig. 1(b).

Equation (1) was integrated over each individual control volume to give the following system of ordinary differential equations (ODEs) for control-volume-averaged solution quantities, $\overline{\mathbf{U}}_i$:

$$\frac{\mathrm{d}\overline{\mathbf{U}}_i}{\mathrm{d}t} = -\frac{1}{V_i}\oint_{\partial\Omega_i}\left(\vec{\mathbf{F}}\cdot\hat{n}\right)\,\mathrm{d}\Gamma + \frac{1}{V_i}\int_{\Omega_i}\mathbf{S}\,\mathrm{d}\Omega = \mathbf{R}_i, \qquad i = 1, 2, \ldots, N_\mathrm{v} \tag{7}$$

where N_v is the number of control volumes (i.e., vertices of the primal mesh), V_i is the volume, $\hat{n}$ is the unit vector normal to the surface of the control volume, $\partial\Omega_i$, and $\overline{\mathbf{U}}_i = \frac{1}{V_i}\int_{\Omega_i}\mathbf{U}\,\mathrm{d}\Omega$. Applying Gauss quadrature to evaluate the surface and volume integrals in Eq. (7) produces a set of nonlinear ODEs given by

$$\frac{\mathrm{d}\overline{\mathbf{U}}_i}{\mathrm{d}t} = -\frac{1}{V_i}\sum_{j=1}^{N_\mathrm{f}}\sum_{k=1}^{G_\mathrm{f}}\left[\omega_\mathrm{f}\vec{\mathbf{F}}\cdot\hat{n}\right]_{i,j,k} + \frac{1}{V_i}\sum_{m=1}^{G_\mathrm{v}}\left[\omega_\mathrm{v}\,\mathbf{S}\right]_{i,m} = \mathbf{R}_i \tag{8}$$

where N_f is the number of faces of Ω_i, G_f and G_v are the number of quadrature points for the face and volume integrals, respectively. The corresponding quadrature weights for the face and volume integrals are denoted by ω_f and ω_v, respectively.

In Eq. (8), the number of quadrature points required for each rule is a direct function of the number of spatial dimensions and the reconstruction order — i.e., the quadrature rule must be able to integrate a k-degree polynomial exactly (k-exactness). Integrating over the individual faces of the polyhedral-shaped control volume is relatively straightforward. Since the faces are triangular, standard quadrature rules for triangles were used. However, general quadrature rules for integrating over complex polyhedrons do not exist. As such, numerical integrals over the volume of these complex elements were evaluated by subdividing the polyhedrons into tetrahedrons and applying standard Gauss quadrature rules to each individual tetrahedron. The coefficients for the quadrature rules applied herein are as given by Felippa [54] and summarized in Table 1.

3.1 CENO Reconstruction

Evaluating Eq. (8) requires numerically integrating the fluxes and source terms over the control-volumes, and this numerical integration requires interpolating the solution at quadrature points. Only control-volume averages are known in the proposed finite-volume approach, so the solution at these quadrature points was interpolated using the high-order CENO method [32–40]. The reconstruction was applied to the primitive solution quantities, $\mathbf{W}$, to ensure that both pressure and internal energy remain positive.

3.1.1 k-Exact Reconstruction

The CENO spatial discretization scheme is based on the high-order k-exact least-squares reconstruction technique of Barth and Fredrickson [4, 55]. The k-exact reconstruction algorithm begins by assuming that the solution within each control-volume is represented by piecewise Taylor polynomials. In three space dimensions, the polynomials are defined as

$$u_i^k(x,y,z) = \sum_{p=0}^{p+q+r\leq k}\sum_{q=0}\sum_{r=0}(x-x_i)^p(y-y_i)^q(z-z_i)^r D_{pqr} \tag{9}$$

where u_i^k is the reconstructed solution quantity, $(x_i,\, y_i,\, z_i)$ is the geometric reference point, k is the degree of the piecewise polynomial interpolant, and D_{pqr} are the unknown coefficients of the Taylor series expansion. Any geometric reference point can be chosen; the vertex about which the control volume was constructed was used here.

The following conditions were applied to determine the unknown coefficients: (i) the mean or average value within the computational volume must be preserved; (ii) the solution reconstruction must reproduce polynomials of degree $\leq k$ exactly (i.e., k-exactness); and (iii) the reconstruction must have compact support. The first condition introduces a constraint on the reconstruction which states that

$$\overline{u}_i = \frac{1}{V_i}\int_{\Omega_i} u_i^k(x,y,z)\,\mathrm{d}\Omega \tag{10}$$

where $\overline{u}_i$ is the control-volume average in Ω_i. Additional constraints are introduced by the second condition, requiring that

$$u_i^k(x,y,z) = u_{\text{exact}} + \mathcal{O}(h^{k+1}) \tag{11}$$

in the vicinity of Ω_i. The length scale, h, is defined as the maximum diameter of the control-volume circumspheres in the vicinity of Ω_i. From Eq. (11), the reconstruction polynomial for Ω_i must also recover the averages of neighboring control volumes. That is,

$$\overline{u}_j = \frac{1}{V_j}\int_{\Omega_j} u_i^k(x,y,z)\,\mathrm{d}\Omega + \mathcal{O}(h^{k+1}) \qquad \forall j \in S_{\text{neigh},i} \tag{12}$$

where $S_{\text{neigh},i}$ is the collection of control-volume indices in the vicinity of Ω_i. This property ensures that the difference between the predicted solution states at control-volume interfaces diminishes at a rate proportional to h^{k+1}.

The third condition merely specifies the number and location of neighbors included in the reconstruction. For a compact stencil, the minimum number of neighbors is equal to the number of unknowns minus one (because of the constraint imposed by Eq. (10)). For any type of mesh, the total number of unknown coefficients for a particular order is given by

$$N = \frac{1}{d!}\prod_{n=1}^{d}(k+n) \tag{13}$$

where d represents the number of space dimensions. In three-dimensions, there are four, ten, twenty and thirty-five unknown coefficients for $k=1$, $k=2$, $k=3$ and $k=4$, respectively.

A sample stencil for Ω_i is illustrated in two-dimensions in Fig. 2. The stencil was constructed by recursively selecting nearest neighbors until at least the minimum number of neighbors was met. The closest neighbors were selected first, and then, if more neighbors were required, the next nearest neighbors were selected. This process continued until the stencil was deemed sufficient. Additional neighbors were included to ensure that the stencil was not biased in any particular direction and that the reconstruction remained reliable on poor quality meshes with high aspect ratio cells [55]. In the present research, a stencil

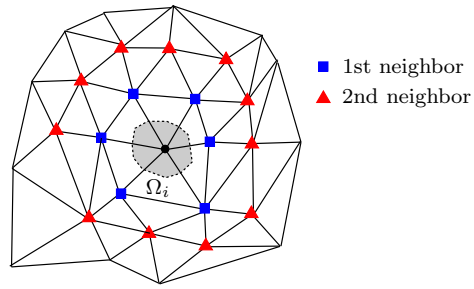


Fig. 2. Supporting stencil for the reconstruction in Ω_i. First- and second-level neighbors are indicated in the figure.

Table 2. Minimum stencil sizes used for reconstructions.

Reconstruction	**Minimum Stencil Size**	
	Theoretical	Actual
Linear ($k = 1$)	3	5
Quadratic ($k = 2$)	9	14
Cubic ($k = 3$)	19	29
Quartic ($k = 4$)	34	51

at least 1.5 times larger than the minimum size was employed. The employed stencil sizes are listed in Table 2.

The constraints given by Eqs. (10) and (12) create an over-determined system of linear equations of the form,

$$\mathbf{AX} = \mathbf{B} \tag{14}$$

where $\mathbf{A}$ is the coefficient matrix, $\mathbf{X}$ is the vector of unknown polynomial coefficients, and $\mathbf{B}$ is a vector which depends on control volume averages. Since the system is over-determined, a least-squares solution for $\mathbf{X}$ was obtained in each control-volume. Equation (10) was strictly enforced, while a minimum-error solution to the remaining constraint equations was sought. The final form of Eq. (14) for each control volume Ω_i was derived from Eqs. (10) and (12). It is given by

$$\begin{bmatrix} w_{1i}\,\widehat{x^0y^0z^1}_{1i} & \cdots & w_{1i}\,\widehat{x^py^qz^r}_{1i} & \cdots & w_{1i}\,\widehat{x^ky^0z^0}_{1i} \\ \vdots & & \vdots & & \vdots \\ w_{ji}\,\widehat{x^0y^0z^1}_{ji} & \cdots & w_{ji}\,\widehat{x^py^qz^r}_{ji} & \cdots & w_{ji}\,\widehat{x^ky^0z^0}_{ji} \\ \vdots & & \vdots & & \vdots \\ w_{ni}\,\widehat{x^0y^0z^1}_{ni} & \cdots & w_{ni}\,\widehat{x^py^qz^r}_{ni} & \cdots & w_{ni}\,\widehat{x^ky^0z^0}_{ni} \end{bmatrix} \cdot \begin{bmatrix} D_{001} \\ \vdots \\ D_{pqr} \\ \vdots \\ D_{k00} \end{bmatrix} = \begin{bmatrix} w_{1i}\,(\overline{u}_1 - \overline{u}_i) \\ \vdots \\ w_{ji}\,(\overline{u}_j - \overline{u}_i) \\ \vdots \\ w_{ni}\,(\overline{u}_n - \overline{u}_i) \end{bmatrix} \tag{15}$$

where n is the number of neighbors in the stencil, $S_{\text{neigh},i}$, and w_{ji} are least-squares weights. The geometric coefficients, $\widehat{x^py^qz^r}_{ji}$, are given by

$$\widehat{x^py^qz^r}_{ji} = \widetilde{x^py^qz^r}_{ji} - \overline{x^py^qz^r}_i \tag{16}$$

where

$$\widetilde{x^p y^q z^r}_{ji} = \frac{1}{V_j} \int_{\Omega_j} (x - x_i)^p (y - y_i)^q (z - z_i)^r \, \mathrm{d}\Omega \tag{17}$$

$$\overline{x^p y^q z^r}_i = \frac{1}{V_i} \int_{\Omega_i} (x - x_i)^p (y - y_i)^q (z - z_i)^r \, \mathrm{d}\Omega \tag{18}$$

To reduce the computational storage requirements, only the geometric moments about each individual control-volume, $\overline{x^p y^q z^r}_i$, were computed and stored prior to solving Eq. (8). The remaining geometric coefficients were computed as needed using a binomial expansion [10, 12]:

$$\widetilde{x^p y^q z^r}_{ji} = \frac{1}{V_j} \int_{\Omega_j} [(x - x_j) + (x_j - x_i)]^p \cdot [(y - y_j) + (y_j - y_i)]^q \cdot [(z - z_j) + (z_j - z_i)]^r \, \mathrm{d}\Omega \tag{19}$$

$$= \sum_{a=0}^{p} \sum_{b=0}^{q} \sum_{c=0}^{r} \binom{p}{a} \binom{q}{b} \binom{r}{c} \cdot (x_j - x_i)^a \cdot (y_j - y_i)^b \cdot (z_j - z_i)^c \cdot \overline{x^{p-a} y^{q-b} z^{r-c}}_j \tag{20}$$

Weighting was applied to each individual constraint equation to improve the locality of the reconstruction [56]. The weights for the reconstruction in Ω_i are

$$w_{ji} = \frac{1}{|\vec{x}_j - \vec{x}_i|^p}, \tag{21}$$

where $\vec{x}_i$ and $\vec{x}_j$ are the vertex locations. The exponent, p, was set equal to 1.

The condition number of the least-squares problem for the reconstruction coefficients was improved via the application of a simple column scaling [10, 57], which effectively makes the condition number independent of the mesh size and control-volume aspect ratio. Scaling the columns of the matrix $\mathbf{A}$ gives the new linear system

$$(\mathbf{AP}) \left(\mathbf{P}^{-1}\mathbf{X}\right) = \mathbf{B} \tag{22}$$

where $\mathbf{P}$ is a diagonal matrix of size $N - 1$ whose entries are the inverse of the largest absolute values of each column of $\mathbf{A}$. The scaling matrix is given by

$$\mathrm{P}_{jj} = \frac{1}{\max\limits_{\forall i} |\mathrm{A}_{ij}|} \qquad i = 1, 2, \ldots, n \tag{23}$$

where P_{jj} and A_{ij} are the individual elements of $\mathbf{P}$ and $\mathbf{A}$, respectively.

A least-squares solution to Eq. (22) was sought using either QR factorization based on Householder transformations or the singular value decomposition (SVD) method [58]. Since the coefficient matrix, $\mathbf{A}$, and the scaling matrix, $\mathbf{P}$, only depend on the mesh geometry, they can be inverted and stored prior to solving Eq. (8) [37]. Thus, when SVD was used, the pseudoinverse was stored and polynomial coefficients were simply determined from the following matrix-vector product at each iteration:

$$\mathbf{X} = \mathbf{P} \left(\mathbf{AP}\right)^{\dagger} \mathbf{B} \tag{24}$$

where $\dagger$ denotes the pseudoinverse and the matrix $\mathbf{P}\left(\mathbf{AP}\right)^{\dagger}$ is the pre-computed and stored result of SVD. This operation was considerably less computationally intensive than performing a full QR factorization or SVD decomposition for each control-volume at every iteration. Once the least-squares solution for $\mathbf{X}$ in Eq. (22) was obtained, the remaining polynomial coefficient, D_{000}, was obtained from Eq. (10).

3.1.2 Reconstruction at Boundaries

To enforce conditions at the boundaries of the computational domain, the least-squares reconstruction was constrained at Gauss quadrature points along the boundary without altering the reconstruction's order of

accuracy [12, 13, 37]. The constraints were implemented as Robin-type boundary conditions and are given by

$$f\left(\vec{x}\right) = a\left(\vec{x}\right) f_{\mathrm{D}}\left(\vec{x}\right) + b\left(\vec{x}\right) f_{\mathrm{N}}\left(\vec{x}\right) \tag{25}$$

where $a\left(\vec{x}\right)$ and $b\left(\vec{x}\right)$ are coefficients which define the contribution of the Dirichlet, $f_{\mathrm{D}}\left(\vec{x}\right)$, and Neumann, $f_{\mathrm{N}}\left(\vec{x}\right)$, components, respectively. These coefficients are simply $(a, b) = (1, 0)$ for Dirichlet- and $(a, b) = (0, 1)$ for Neumann-type boundary conditions. The Dirichlet condition is expressed as

$$f_{\mathrm{D}}\left(\vec{x}_{\mathrm{g}}\right) = u_i^k\left(\vec{x}_{\mathrm{g}}\right) \tag{26}$$

where $\vec{x}_{\mathrm{g}}$ is the location of the Gauss quadrature point. The Neumann condition is

$$\begin{aligned} f_{\mathrm{N}}\left(\vec{x}_{\mathrm{g}}\right) &= \vec{\nabla} u_i^k\left(\vec{x}_{\mathrm{g}}\right) \cdot \hat{n}_{\mathrm{g}} \\ &= \sum_{p+q+r=1}^{p+q+r\leq k} \sum \sum \Delta x^{p-1} \Delta y^{q-1} \Delta z^{r-1} \left[p \Delta y \Delta z n_x + q \Delta x \Delta y n_y + r \Delta x \Delta y n_z\right] D_{pqr} \end{aligned} \tag{27}$$

where $\Delta(\cdot) = (\cdot)_{\mathrm{g}} - (\cdot)_i$ is the distance between the vertex of the control volume adjacent to the boundary and the Gauss quadrature point, and $\hat{n}_{\mathrm{g}}$ is the outward surface normal at the quadrature point.

Exact solutions to the boundary constraints described by Eq. (25) were sought, which adds linear equality constraints to the original over-determined system (Eq. (15)). This resulting equality-constrained least-squares problem was solved using the method of weights [58]. It was solved in the same manner as described in Section 3.1.1, except the original equations in Eq. (15) were multiplied by an additional weight. The new over-determined linear system with boundary constraints is given by

$$\begin{bmatrix} \epsilon \mathbf{A} \\ \mathbf{C} \end{bmatrix} \cdot \left[\mathbf{X}\right] = \begin{bmatrix} \epsilon \mathbf{B} \\ \mathbf{D} \end{bmatrix} \tag{28}$$

where $\mathbf{C}$ and $\mathbf{D}$ are the coefficient matrix and solution vector for the boundary constraints, respectively. A weight, ϵ, equal to 10^{-3} was applied to the original equations defined by Eq. (15), which gives the boundary constraints a large influence.

For boundary conditions where the reconstructed variables are not related, such as inflow/outflow or farfield-type conditions, the constraints were applied separately to each variable. Thus, a separate least-squares problem with equality constraints was set up for each variable and solved independently of the others. For these cases, the flux along the boundary is simply given as

$$\vec{\mathbf{F}}_{\mathrm{b}}(\vec{x}) = \vec{\mathbf{F}}\left(\mathbf{W}(\vec{x})\right) \tag{29}$$

More complex boundary conditions involve linear combinations of solution variables that couple the reconstruction coefficients of different variables. For example, the individual velocity components for reflection or solid wall conditions are coupled because $\vec{v} \cdot \hat{n} = 0$. These types of coupled boundary conditions were handled via constraints in combination with an appropriately prescribed flux [12, 13, 38, 59].

For coupled boundary conditions, the unknown polynomial coefficients for the uncoupled variables were determined independently first, and then the coupled variables were reconstructed together. To illustrate this procedure, consider a reflection boundary condition, which was applied in the present study. Along reflecting boundaries,

$$\begin{gathered} \vec{\nabla} \rho \cdot \hat{n} = 0 \\ \vec{v} \cdot \hat{n} = 0 \\ \vec{\nabla} e \cdot \hat{n} = 0 \end{gathered}$$

where $\hat{n}$ is a unit vector normal to the boundary. Both ρ and e are independent, so they were reconstructed separately by solving Eq. (28), but the three components of velocity are coupled via a linear combination of each other. The constraints for a zero normal velocity at the boundary are given by

$$u\left(\vec{x}\right) n_x\left(\vec{x}\right) + v\left(\vec{x}\right) n_y\left(\vec{x}\right) + w\left(\vec{x}\right) n_z\left(\vec{x}\right) = 0 \tag{30}$$

As such, the coupled, over-determined linear system for the unknown polynomial coefficients of the three velocity components is as follows:

$$\begin{bmatrix} \epsilon\mathbf{A}_u & \mathbf{0} & \mathbf{0} \\ \mathbf{C}_u & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \epsilon\mathbf{A}_v & \mathbf{0} \\ \mathbf{0} & \mathbf{C}_v & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \epsilon\mathbf{A}_w \\ \mathbf{0} & \mathbf{0} & \mathbf{C}_w \\ \multicolumn{3}{c}{\text{Coupled constraints, Eq. (30)}} \end{bmatrix} \cdot \begin{bmatrix} \mathbf{X}_u \\ \mathbf{X}_v \\ \mathbf{X}_w \end{bmatrix} = \begin{bmatrix} \epsilon\mathbf{B}_u \\ \mathbf{D}_u \\ \epsilon\mathbf{B}_v \\ \mathbf{D}_v \\ \epsilon\mathbf{B}_w \\ \mathbf{D}_w \\ \cdots \end{bmatrix} \tag{31}$$

where the subscripts u, v, and w refer to the solution quantities with which the components of the linear system are associated with.

The prescribed flux at each quadrature point along the reflecting boundary is

$$\vec{\mathbf{F}}_{\text{reflect}} \cdot \hat{n} = [0,\ pn_x,\ pn_y,\ pn_z,\ 0] \tag{32}$$

where p is calculated at the wall boundary by extrapolating ρ and e.

Solid walls were treated the same as reflecting boundaries, except that no constraints were applied to ρ and e along the boundary, i.e., only $\vec{v} \cdot \hat{n} = 0$ was enforced.

3.1.3 Smoothness Indicator

After performing a k-exact reconstruction in each control volume, the smoothness indicator was computed for every reconstructed variable to identify under-resolved solution content. It was evaluated for each control volume as [38]

$$S_i = \frac{\sigma}{\max\left[(1-\sigma), \delta\right]} \frac{\text{SOS} - \text{DOF}}{\text{DOF} - 1} \tag{33}$$

where σ is a smoothness parameter, δ is a tolerance to avoid division by zero (equal to 10^{-8}), DOF is the number of degrees of freedom and SOS is the size of the stencil. The factor, $(\text{SOS} - \text{DOF})/(\text{DOF} - 1)$, adjusts σ to account for the number of polynomial coefficients relative to the size of the reconstruction stencil.

The smoothness parameter is based on the coefficient of determination or R^2 parameter, which is a statistical parameter used for assessing how well lines or curves fit data points [60]. For a control-volume Ω_i, the smoothness parameter is given by

$$\sigma_i = 1 - \frac{\displaystyle\sum_{\forall j \in S_{\text{neigh},i}} \left[u_j^k(\vec{x}_j) - u_i^k(\vec{x}_j)\right]^2}{\displaystyle\sum_{\forall j \in S_{\text{neigh},i}} \left[u_j^k(\vec{x}_j) - \overline{u}_i\right]^2} \tag{34}$$

where u is the solution variable of interest. The numerator of the fraction in Eq. (34) measures how well the reconstruction polynomial for Ω_i predicts the values in nearby control volumes, while the denominator in Eq. (34) measures the variance from some reference point — $\overline{u}_i$ in this case — and normalizes σ.

By definition, σ can have a value between negative infinity and one. A value of unity indicates that the solution is smooth whereas a small or negative value indicates large variations in solution content within the reconstruction stencil. An order of magnitude analysis similar to the ones performed by Ivan and Groth [39] and Charest et al. [35] for cell-based CENO formulations confirms the correct behavior of σ with changes in mesh size, h. It follows from Eq. (11) that

$$\sigma \approx 1 - \frac{\left[\mathcal{O}(h^{k+1})\right]^2}{\left[\mathcal{O}(h)\right]^2} \approx 1 - \mathcal{O}(h^{2k}) \tag{35}$$

for smooth solution content. Thus, $\sigma \to 1$ as $\Delta x \to 0$ at a rate much faster than the formal order of accuracy of the scheme. Conversely, when the solution is not smooth, σ is much less than unity because

$$\sigma \approx 1 - \frac{\left[\mathcal{O}(1)\right]^2}{\left[\mathcal{O}(1)\right]^2} \approx 1 - \mathcal{O}(1) \tag{36}$$

This analysis confirms that the switching between the low- and high-order reconstructions occurs because the numerical solution is under-resolved and reconstructions are oscillatory. The smoothness indicator only measures how well nearby recovered polynomials agree with each other. It is not directly related to the size of the computational mesh.

Solutions were deemed smooth when the value of S was above a critical value, S_c. Previous studies found that values for S_c between 1000–5000 provided an excellent balance between stability and accuracy [38]. And because of the form of Eq. (33), S grows rapidly as $\sigma \to 1$. Thus, S tends to be orders of magnitude greater than these cutoff limits in smooth regions. Unless otherwise specified, S_c was equal to 2000. Smaller values of S_c may provide more accurate predictions in some cases, but values much less than 1000 tend to allow small oscillations in the solution.

In cases where the solution was not varying, such as in the free-stream, the smoothness indicator sometimes incorrectly indicated that solutions with small deviations due to numerical noise were under-resolved. This occurred because both the denominator and numerator of the fraction in Eq. (34) approached zero, and σ was close to zero or negative. To alleviate this issue, the solution was automatically deemed smooth if the local variation within the stencil was below a tolerance. That is, if

$$\max_{\forall j \in S_{\mathrm{neigh},i}} |\overline{u}_j - \overline{u}_i| < t_\mathrm{abs} u_\mathrm{ref} + t_\mathrm{rel} \overline{u}_\mathrm{stencil}, \tag{37}$$

the smoothness indicator was not computed and the high-order k-exact reconstruction was used. Here, u_ref is a reference solution, $\overline{u}_\mathrm{stencil}$ is the stencil average, $t_\mathrm{abs} = 10^{-5}$ is an absolute tolerance, and $t_\mathrm{rel} = 10^{-3}$ is a relative tolerance. The reference solution, u_ref, was the average value within the computational domain and was only computed once prior to solving the governing ODEs given by Eq. (8).

3.1.4 Limited Piecewise Linear Reconstruction

In regions where the smoothness indicator was below the critical value, monotonicity was preserved by switching to a limited piecewise linear ($k = 1$) reconstruction on a smaller stencil (see Table 2 for stencil sizes). The least-squares reconstruction procedure described by Barth [61] was used in these regions, since it was found to be more computationally efficient than directly solving Eq. (15) via QR factorization or SVD.

The limited piecewise linear representation in each control volume is given by

$$u^{k=1}(\vec{x}) = \overline{u}_i + \phi_i \vec{\nabla} u \cdot (\vec{x} - \vec{x}_{\mathrm{c},i}) \tag{38}$$

where ϕ_i is the slope limiter and $\vec{x}_{\mathrm{c},i}$ is the location of the control-volume centroid. In this particular case, $k = 1$, the control-volume-averaged solution is equal to the solution at the centroid. It is not equal to the solution at the control volume's associated vertex, since the two locations do not coincide with each other, i.e., $\vec{x}_{\mathrm{c},i} \neq \vec{x}_i$.

For a control volume Ω_i, the new over-determined matrix equation for the unknown polynomial coefficients, which, in this case, are the solution gradients, is given by

$$\begin{bmatrix} w_{i1}\Delta x_{i1} & w_{i1}\Delta y_{i1} & w_{i1}\Delta z_{i1} \\ \vdots & \vdots & \vdots \\ w_{ij}\Delta x_{ij} & w_{ij}\Delta y_{ij} & w_{ij}\Delta z_{ij} \\ \vdots & \vdots & \vdots \\ w_{in}\Delta x_{in} & w_{in}\Delta y_{in} & w_{in}\Delta z_{in} \end{bmatrix} \cdot \begin{bmatrix} \dfrac{\partial u}{\partial x} \\ \dfrac{\partial u}{\partial y} \\ \dfrac{\partial u}{\partial z} \end{bmatrix} = \begin{bmatrix} w_{i1}\left(\overline{u}_1 - \overline{u}_i\right) \\ \vdots \\ w_{ij}\left(\overline{u}_j - \overline{u}_i\right) \\ \vdots \\ w_{in}\left(\overline{u}_n - \overline{u}_i\right) \end{bmatrix} \tag{39}$$

where $\Delta(\cdot)_{ij} = (\cdot)_j - (\cdot)_i$ is the distance between control volume centroids. This system was solved in a least-squares sense using the Gram-Schmidt process outlined in [61].

Uncoupled Dirichlet- and Neumann-type boundary conditions were incorporated by adding constraint equations to Eq. (39) for each quadrature point:

$$\text{Dirichlet:} \qquad \vec{\nabla} u \cdot \Delta\vec{x}_i = u\left(\vec{x}_{\mathrm{g}}\right) - \overline{u}_i \tag{40}$$

$$\text{Neumann:} \qquad \vec{\nabla} u \cdot \Delta\vec{x}_n = \vec{\nabla} u\left(\vec{x}_{\mathrm{g}}\right) \cdot \Delta\vec{x}_n \tag{41}$$

where $\Delta\vec{x}_i = \vec{x}_{\mathrm{g}} - \vec{x}_{\mathrm{c},i}$ and $\Delta\vec{x}_n = (\Delta\vec{x}_i \cdot \hat{n}_{\mathrm{g}})\hat{n}_{\mathrm{g}}$. More complicated boundary conditions were treated using ghost cells to influence the reconstruction. For example, reflecting boundaries or solid walls were treated by reflecting the solution at the control volume's centroid about the boundary and injecting the reflected solution into the ghost cell.

Limiting was performed using the multi-dimensional limiting process (MLP) developed by Park et al. [62] in conjunction with the slope limiter function of Venkatakrishnan [63]. Although MLP was developed specifically for cell-based finite-volume schemes on structured and unstructured meshes, it is easily extended to vertex-based formulations. The general form of the MLP condition states that monotonicity is preserved if the following condition is true for every vertex v_j of a control-volume Ω_i:

$$\overline{u}^{\min}_{i,\mathrm{neigh}} \leq u_{v_j} \leq \overline{u}^{\max}_{i,\mathrm{neigh}} \qquad \forall v_j \in \Omega_i \tag{42}$$

where u_{v_j} is the interpolated value at the vertex v_j, and $\overline{u}^{\min}_{i,\mathrm{neigh}}$ and $\overline{u}^{\max}_{i,\mathrm{neigh}}$ are the minimum and maximum control-volume-averaged values among the control-volumes that share a vertex with Ω_i, respectively. Essentially, the interpolated values at the vertices of the control volume must be bounded by the maximum and minimum $\overline{u}$ of the surrounding control volumes. These vertices, v_j, are the vertices of the dual mesh, $\mathcal{D}(\Omega)$, not the primal mesh, $\mathcal{T}(\Omega)$.

The final MLP slope limiter for the ith control volume is expressed as

$$\phi_i = \min_{\forall v_j \in \Omega_i} \begin{cases} \Phi\left(\dfrac{\overline{u}^{\max}_{i,\mathrm{neigh}} - \overline{u}_i}{u_{v_j} - \overline{u}_i}\right) & \text{if } u_{v_j} - \overline{u}_i > a, \\ \Phi\left(\dfrac{\overline{u}^{\min}_{i,\mathrm{neigh}} - \overline{u}_i}{u_{v_j} - \overline{u}_i}\right) & \text{if } u_{v_j} - \overline{u}_i < -a, \\ 1 & \text{otherwise} \end{cases} \tag{43}$$

where Φ is the Venkatakrishnan limiter function and $a = 10^{-7}$ is tolerance to avoid limiter chatter caused by numerical noise.

3.2 Numerical Flux and Sources

An upwind Godunov-type scheme was used to integrate the inviscid numerical flux, $\vec{\mathbf{F}}$, over the control-volume [64]. Given the left and right solution states, $\mathbf{W_L}$ and $\mathbf{W_R}$, the numerical flux at the interface between two control-volumes is defined as

$$\vec{\mathbf{F}} \cdot \hat{n} = \mathcal{F}\left(\mathbf{W_L}, \mathbf{W_R}, \hat{n}\right) \tag{44}$$

where $\mathcal{F}$ is a flux function which solves a Riemann problem in a direction aligned along the face normal, $\hat{n}$. Both the Rusanov [65, 66] and HLL [67] approximate Riemann solvers were implemented for the numerical flux, $\mathcal{F}$. The two different approximate Riemann solvers each have their own advantages and disadvantages. The HLL flux was found to be slightly more accurate while the Rusanov flux was found to be more stable and less costly.

The left and right solution states at the interface were determined using the CENO reconstruction procedure described in Section 3.1. As a result, the leading truncation error due to the inviscid operator is $\mathcal{O}\left(h^{k+1}\right)$ in smooth regions. When the solution is under-resolved and deemed not smooth, the limited piecewise linear reconstruction was used. In this case, the truncation error of the inviscid operator is between $\mathcal{O}\left(h^2\right)$ (unlimited) and $\mathcal{O}\left(h\right)$ (limited).

The limited linear reconstruction of Section 3.1.4 was only ever applied to the inviscid terms in Eq. (8). Even when the solution was deemed non-smooth, the source terms were still evaluated using the higher-order representation since they don't generally generate instabilities. Thus, the truncation error of the source term operator is $\mathcal{O}\left(h^{k+1}\right)$.

3.3 Transient Continuation and Steady-State Relaxation

Equation (8) defines a finite set of ODEs. For unsteady problems, the temporal derivative was discretized using the classical Runge-Kutta (RK) schemes [68]. Schemes with up to four stages were implemented. For example, the four-stage (RK4) scheme is written as

$$\mathbf{U}^1 = \mathbf{U}^n + \frac{\Delta t}{2}\mathbf{R}^n \tag{45a}$$

$$\mathbf{U}^2 = \mathbf{U}^n + \frac{\Delta t}{2}\mathbf{R}^1 \tag{45b}$$

$$\mathbf{U}^3 = \mathbf{U}^n + \Delta t\mathbf{R}^2 \tag{45c}$$

$$\mathbf{U}^{n+1} = \mathbf{U}^n + \frac{\Delta t}{6}\left(\mathbf{R}^n + 2\mathbf{R}^1 + 2\mathbf{R}^2 + \mathbf{R}^3\right) \tag{45d}$$

where the superscript n denotes the time level.

Steady-state problems where relaxed using the two-stage optimally smoothing scheme of van Leer et al. [69].

$$\mathbf{U}^{n+1} = \mathbf{U}^n + \sum_{\alpha=1}^{n_\mathrm{s}} \beta^\alpha \Delta t \mathbf{R}^\alpha \tag{46}$$

where $n_\mathrm{s} = 2$ is the number of stages, the superscript α denotes the intermediate stage, and β^α are the stage coefficients.

In both cases, steady and unsteady, the time step was determined by considering the inviscid Courant-Friedrichs-Lewy (CFL) stability criteria. The maximum permissible time step for each control volume is given by

$$\Delta t_i = \mathrm{CFL} \cdot \left(\frac{\Delta_i}{\|\vec{v}_i\| + a_i}\right), \qquad i = 1, 2, \ldots, N_\mathrm{v} \tag{47}$$

where $\Delta_i = \sqrt[3]{V_i}$ and CFL is a constant greater than zero. For time-accurate problems, a global time step was used. It is given as

$$\Delta t = \min_{\forall i} \Delta t_i \tag{48}$$

4. Results For Three-Dimensional Unstructured Meshes

The proposed finite-volume scheme was assessed in terms of accuracy, stability, and computational efficiency. Numerical results for smooth and discontinuous function reconstructions, as well as steady and unsteady idealized flows, were obtained on three-dimensional unstructured tetrahedral meshes. All computations were performed on an Hewlett-Packard DL980 G7 compute node with eight Intel Xeon X6550 (2.00GHz) processors and 128 GB of random-access memory (RAM).

Table 3. Convergence rates of the error norms for the spherical cosine function.

k	Formal	L_1	L_2	L_∞
0	1	1.01	1.00	0.95
1	2	2.13	2.11	2.11
2	3	3.06	3.07	2.71
3	4	4.04	4.01	4.01
4	5	5.08	5.05	4.41

Depending on the problem, accuracy was assessed based on the L_1, L_2, and/or L_∞ norms of the error between the exact solution and the numerical solution. The L_p norm of the error evaluated over the entire computational domain is given by

$$L_p = \|\text{Error}\|_p = \left[\frac{1}{\mathcal{V}_\text{T}} \sum_i^{N_\text{v}} \int_{\Omega_i} \left| u_i^k(\vec{x}) - u_\text{exact}(\vec{x}) \right|^p \, \text{d}\Omega \right]^{1/p} \tag{49}$$

where $\mathcal{V}_\text{T}$ is the total volume of the domain and $u_\text{exact}(\vec{x})$ is the exact solution. This integration was performed using the adaptive cubature algorithm developed by Berntsen et al. [70] for integrating functions over a collection of three-dimensional simplices. The algorithm makes use of a 43-point cubature formula of degree 8.

4.1 Spherical Cosine Function

The first case considered was the reconstruction of a smooth spherical cosine function. The function, which is smooth in all directions, is illustrated in Fig. 3(a) and described by

$$u(r) = 1 + \frac{1}{3}\cos(r) \tag{50}$$

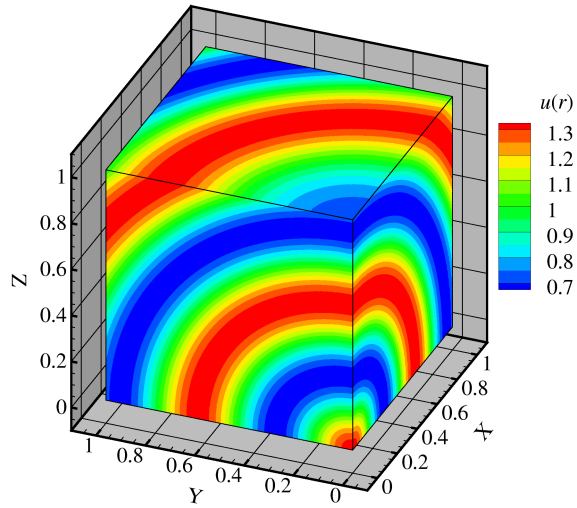
where $r = 10\sqrt{x^2 + y^2 + z^2}$ is the radial position. The solution was computed on a unit cube centered at (0.5, 0.5, 0.5) using grids composed of tetrahedral cells with varying levels of resolution. A sample mesh is illustrated in Fig. 3(b).

Unlimited k-exact reconstructions of the spherical cosine function were obtained on a coarse mesh (995 vertices and 4,515 tetrahedral elements) and are illustrated in Fig. 3(c). As the order of the piecewise polynomial interpolant was increased from $k = 0$ to $k = 4$, the reconstructed solution rapidly approached the exact solution. There is almost no visible difference between the exact solution and the reconstructed solution for $k = 4$.

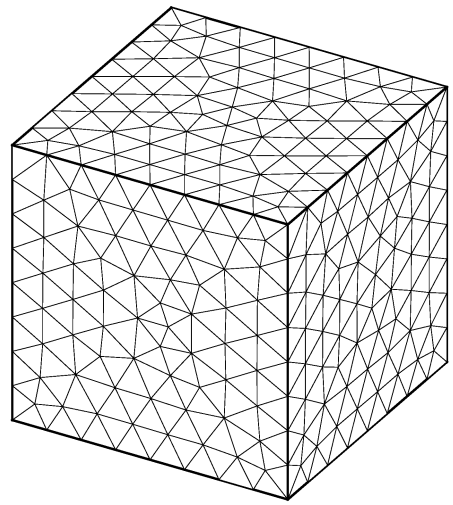
The behavior of the L_2 norm of the discretization error as a function of mesh resolution is illustrated in Fig. 3(d) for various values of k. A large reduction in the global error was observed as the mesh resolution and the order of the polynomial representation were increased. For example, the L_2 error norm obtained on the finest mesh using $k = 4$ was approximately a factor of 3×10^4 smaller than the error obtained using $k = 0$. Figure 3(d) also confirms that k-exact reconstruction of a smooth function yields an order of accuracy equal to k+1. A convergence rate of approximately k+1 was observed in all of the error norms, including the L_∞ error norm. The individual convergence rates as a function of k are listed in Table 3.

4.2 Abgrall's Function

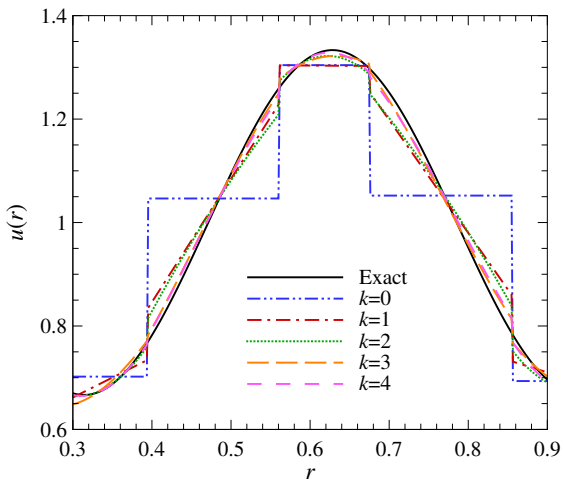
The Abgrall function [71] possesses a number of solution discontinuities that test a high-order spatial discretization's ability to maintain monotonicity. As such, reconstructions of this function were performed using the proposed CENO algorithm to ensure the effectiveness of the smoothness indicator defined in Eq. (33). While the performance of the smoothness indicator was already verified using the Abgrall function on both structured [36] and unstructured [32–35] meshes, it has not been verified for a vertex-based approach. All of the previous CENO approaches applied cell-based finite-volume formulations only.



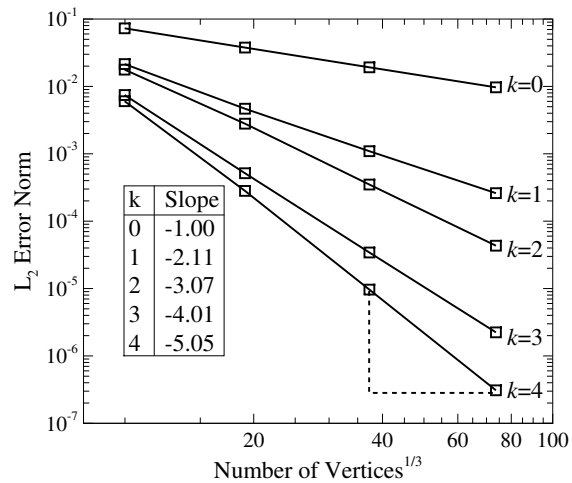
(a) Exact solution.



(b) A sample computational mesh.



(c) Unlimited reconstructions along the diagonal from (0, 0, 0) to (1, 1, 1) on a mesh with 995 vertices.



(d) Reconstruction error as a function of mesh size.

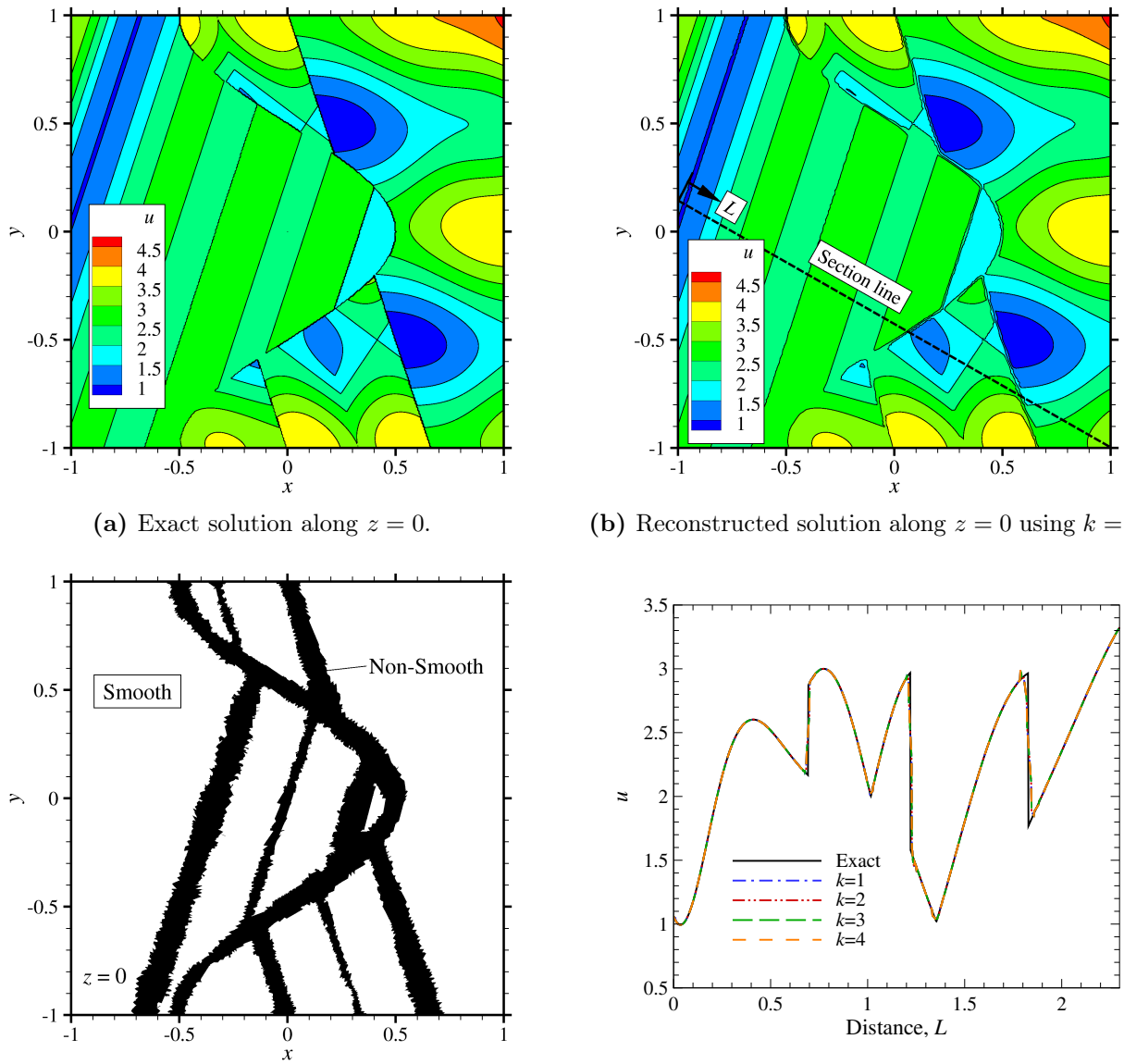
Fig. 3. Results for k-exact reconstruction of the spherical cosine function.

The Abgrall function was originally designed to vary in two space dimensions only. As such, it was modified to include a variation and discontinuity along the third dimension. The resulting three-dimensional discontinuous function is given by

$$u(x,y,z) = g(z) \cdot \begin{cases} f\left[x - \cot\left(\sqrt{\pi/2}\; y\right)\right] & x \le \cos(\pi y)/2 \\ f\left[x + \cot\left(\sqrt{\pi/2}\; y\right)\right] + \cos(2\pi y) & x > \cos(\pi y)/2 \end{cases} \tag{51}$$

where

$$f(r) = \begin{cases} -r \cdot \sin\left(3\pi r^2/2\right) & r \le -1/3 \\ |\sin(2\pi r)| & |r| < 1/3 \\ 2r - 1 + \sin(3\pi r)/6 & r \ge 1/3 \end{cases} \tag{52}$$



(a) Exact solution along $z = 0$.

(b) Reconstructed solution along $z = 0$ using $k = 4$.

(c) Computed smoothness indicator for $k = 4$.

(d) Reconstructed solution along the section line in (b).

Fig. 4. Results for CENO reconstruction of the modified Abgrall function. Numerical results shown here were obtained using a mesh with 3 million vertices and 18 million tetrahedral elements.

and

$$g(z) = \begin{cases} \sin\left(z\pi/2\right)/2 + 1 & z < -1/2 \\ -z/2 + 1 & z \geq -1/2 \end{cases} \tag{53}$$

Eq. (51) was discretized on a cube with length 2 and centered about the origin, using computational meshes similar to those used for the spherical cosine function (Fig. 3(b)). The exact and numerical solutions along $z = 0$ are illustrated in Figs. 4(a) and 4(b), respectively. This reconstruction was preformed using $k = 4$ on a mesh with approximately 3 million vertices and 18 million tetrahedral elements, and, as observed in Fig. 4(b), it was able to accurately represent the Abgrall function without producing spurious oscillations. This is because the smoothness indicator, illustrated in Fig. 4(c), correctly identified the discontinuities in both u and $\vec{\nabla} u$.

The reconstructed solution obtained with $k = 1$ to 4 and a mesh with 18 million tetrahedral elements

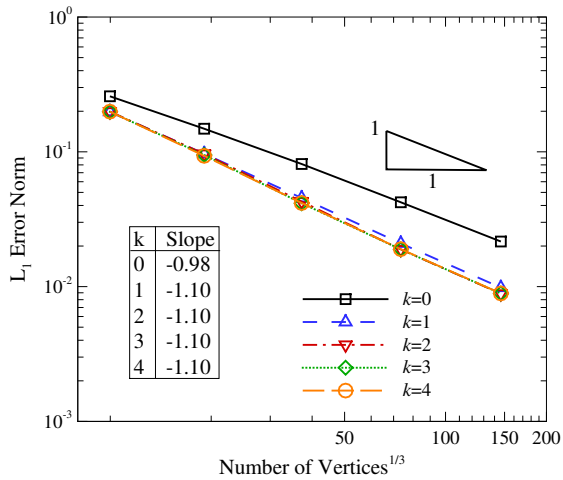


Fig. 5. Effect of grid resolution on solution accuracy for CENO reconstruction of the modified Abgrall function. Accuracy is measured using the L_1 norm of the error.

is compared with the original function along a line in Fig. 4(d). For all values of k, the proposed CENO scheme accurately represented the exact solution and was able to achieve oscillation-free solutions despite the large discontinuities in u and $\vec{\nabla} u$. This oscillation-free behavior confirms the ability of the smoothness indicator to correctly identify under-resolved solution content.

The effect of mesh resolution on the L_1 norm of the solution error is illustrated in Fig. 5. A large improvement in the error was achieved by increasing k. The largest reduction in error occurs when k is increased from 0 to 1. There was a factor of 2.2 reduction in error using $k = 1$ on the finest mesh when compared to the solution error obtained with $k = 0$. Although this improvement became less pronounced as k was increased further, there was still a reduction in error as k was increased above 1. Using $k = 4$ on the same mesh, the error was approximately a factor of 2.4 less than the value obtained using $k = 0$. Even larger reductions in error over the first- and second-order schemes are expected when the high-order ($k > 1$) CENO schemes are used combination with adaptive mesh refinement (AMR) [38].

The convergence rate of the error norms is also provided in Fig. 5. An order of accuracy of 1 was observed for all values of k, which was expected after applying a limited piecewise linear reconstruction near discontinuities. Nonetheless, the main highlight is that the hybrid reconstruction procedure was able to produce non-oscillatory solutions despite the presence of discontinuities, using only a single, central stencil.

4.3 Smooth Supersonic Flow

The spatial accuracy of the proposed finite-volume formulation was verified for smooth flows using the method of manufactured solutions (MMS) [72–74]. In MMS, analytical source terms are derived which, when added to the governing equations, produce a desired solution. The particular MMS approach outlined by Roy et al. [75, 76] was applied to produce sinusoidal solutions of the following form:

$$\rho = \rho_0 + \rho_x \sin\left(a_{\rho_x} \pi x/L\right) + \rho_y \cos\left(a_{\rho_y} \pi y/L\right) + \rho_z \sin\left(a_{\rho_z} \pi z/L\right) \tag{54a}$$

$$u = u_0 + u_x \sin\left(a_{u_x} \pi x/L\right) + u_y \cos\left(a_{u_y} \pi y/L\right) + u_z \cos\left(a_{u_z} \pi z/L\right) \tag{54b}$$

$$v = v_0 + v_x \cos\left(a_{v_x} \pi x/L\right) + v_y \sin\left(a_{v_y} \pi y/L\right) + v_z \sin\left(a_{v_z} \pi z/L\right) \tag{54c}$$

$$w = w_0 + w_x \sin\left(a_{w_x} \pi x/L\right) + w_y \sin\left(a_{w_y} \pi y/L\right) + w_z \cos\left(a_{w_z} \pi z/L\right) \tag{54d}$$

$$p = p_0 + p_x \cos\left(a_{p_x} \pi x/L\right) + p_y \sin\left(a_{p_y} \pi y/L\right) + p_z \cos\left(a_{p_z} \pi z/L\right) \tag{54e}$$

where ϕ_0, ϕ_x, ϕ_y, ϕ_z, a_{ϕ_x}, a_{ϕ_y}, and a_{ϕ_z} are constants for the variable $\phi \in \{\rho, u, v, w, p\}$. The coefficients were chosen based on those given by Roy et al. [75] for smooth, three-dimensional, supersonic inviscid flow. They are provided in Table 4 for completeness. The length scale, L, was chosen as unity.

This particular manufactured solution describes a steady supersonic flow at an angle of approximately 45 degrees to the coordinate axes, and with a Mach number varying between 3 and 6. The flow was modeled

Table 4. The coefficients for the three-dimensional manufactured solution to Eq. (54).

ϕ	ϕ_0	ϕ_x	ϕ_y	ϕ_z	a_{ϕ_x}	a_{ϕ_y}	a_{ϕ_z}
ρ	1	0.15	-0.1	-0.12	1	0.5	1.5
u	800	50	-30	-18	1.5	0.6	0.5
v	800	-75	40	-30	0.5	2/3	1.25
w	800	15	-25	35	1/3	1.5	1
p	1×10^5	0.2×10^5	0.5×10^5	-0.35×10^5	2	1	1/3

on a unit cube over the range $0 \leq x,\ y,\ z \leq 1$; although, any domain could be chosen since the solutions exist for all x, y and z. An example of this smoothly varying solution is illustrated in Fig. 6(a), which depicts the internal energy distribution.

The spatial accuracy was assessed by performing calculations with different mesh sizes and measuring the changes in solution error. Solutions were obtained on meshes of varying resolution, similar to those in Fig. 3(b), with supersonic inflow and outflow boundary conditions at the corresponding upstream and downstream boundaries of the domain. All solutions were relaxed to a steady-state using the two-stage optimally smoothing scheme of van Leer et al. [69] with a CFL = 0.5 and the HLL numerical flux. Any numerical flux can be used for this error analysis, because, according to Eq. (12), the dissipation vanishes with $\mathcal{O}(h^{k+1})$ for smooth solutions.

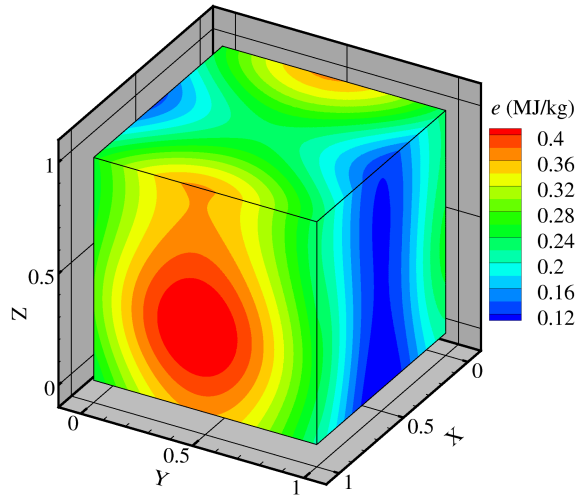
For each calculation, the analytical solution was prescribed at every vertex of the primal mesh and the governing equations were relaxed until all equation residuals were reduced by four orders of magnitude. Tighter tolerances were also tested, but no significant gain in accuracy was observed using them.

The L_2 norm of the error in the predicted internal energy, e, is illustrated in Fig. 6(b). All of the primitive solution quantities displayed the same relationship between mesh size and total solution error, but e displayed the largest errors and was therefore chosen for this analysis. The slopes of the lines in Fig. 6(b) are provided in Fig. 6(c), along with those observed for the other norms, i.e., the L_1 and L_∞ norms. For all values of interest for k, the formal order of accuracy was achieved by the L_2 norm. The other norms also displayed similar convergence characteristics, although some degradation of the slopes of the L_∞ norms were observed as the mesh spacing decreased. This is largely attributed to the finite precision of the adaptive cubature algorithm used to evaluate the numerical errors.

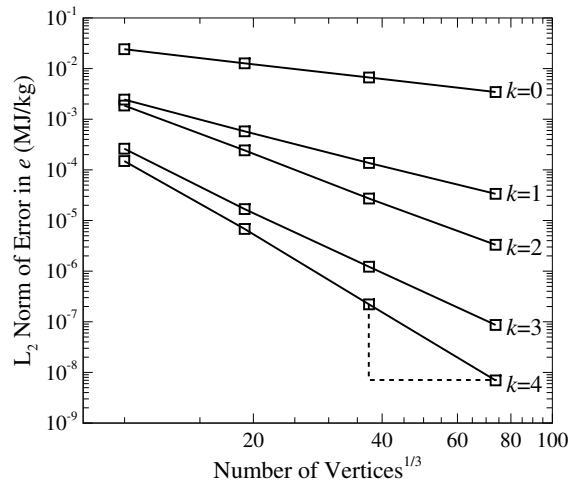
Figure 6(b) illustrates the advantages of using high-order methods on smooth problems. On the finest mesh investigated for this problem, the fifth-order ($k = 4$) scheme achieved an L_2 error norm that was 5×10^5 times smaller than what was obtained using the first-order ($k = 0$) scheme on the same mesh. However, this direct comparison does not account for the added computational cost of using high-order methods.

To compare computational cost, the solution error obtained with each scheme is plotted as a function of the wall-clock time in Fig. 6(d). The high-order schemes became more efficient in terms of accuracy versus computational cost as the target accuracy got smaller. That is, there is a particular range of accuracy for which a particular value of k is the most efficient, and this optimal value of k increases with the desired level of accuracy. For all the meshes considered, the first-order ($k = 0$) scheme was the least efficient, while the second-order ($k = 1$) scheme was the most efficient for a target error above approximately 10^{-3}. Just below this level of error, the fourth-order ($k = 3$) scheme was the most efficient. For example, it took approximately 24 times less wall-time than the second-order scheme to achieve an error of about 3.5×10^{-5}. The computational savings from the fifth-order scheme are much greater. This scheme was the most efficient for errors below 10^{-6}. Extrapolating the results in Fig. 6(d), the fifth-order scheme achieved an error of 7×10^{-9} in a factor of 2×10^6 less wall-time than the second-order scheme.

These results confirm that, for smooth problems, the higher-order CENO schemes are more efficient for higher levels of desired accuracy (i.e., lower errors).



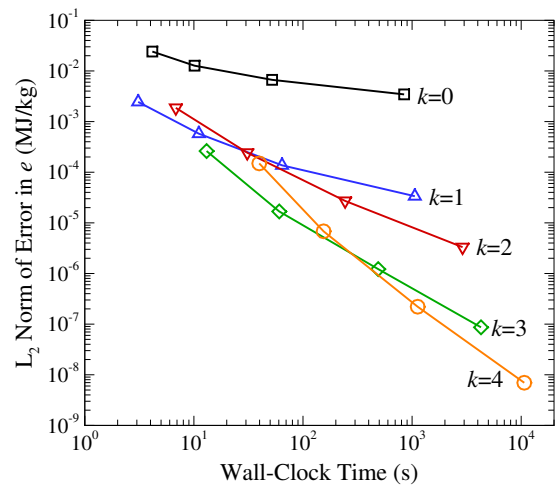
(a) Exact solution.



(b) Error in predicted e.

k	Formal	L_1	L_2	L_∞
0	1	0.97	0.97	0.83
1	2	2.05	2.06	1.88
2	3	3.10	3.09	2.72
3	4	3.93	3.89	3.39
4	5	5.09	5.08	4.35

(c) Convergence of error norms.



(d) Solution time for a given accuracy.

Fig. 6. Results for smooth supersonic flow.

4.4 Ringleb's Flow

Another steady-state problem with a known solution that was used to assess the accuracy of numerical algorithms is Ringleb's flow [77]. It is a hodograph solution to the Euler equations that is parameterized in terms of the total velocity, q, and the stream line constant, ψ, as [78]

$$x(q,\psi) = \frac{1}{2\rho}\left(\frac{2}{\psi^2} - \frac{1}{q^2}\right) - \frac{J}{2} \tag{55}$$

$$y(q,\psi) = \pm\frac{1}{\psi\rho q}\sqrt{1 - \left(\frac{q}{\psi}\right)^2} \tag{56}$$

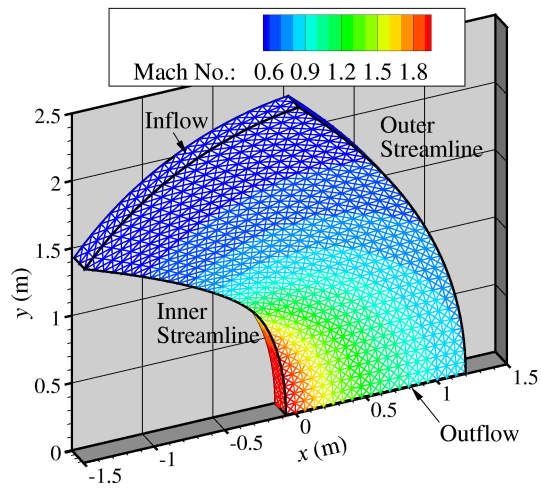


Fig. 7. Mach number contours and sample computational mesh for Ringleb's flow.

where

$$
\begin{aligned}
a &= \sqrt{1 - \frac{\gamma - 1}{2} q^2} \\
\rho &= a^{\left(\frac{2}{\gamma-1}\right)} \\
p &= \frac{1}{\gamma} a^{\left(\frac{2\gamma}{\gamma-1}\right)} \\
J &= \frac{1}{a} + \frac{1}{3a^3} + \frac{1}{5a^5} - \frac{1}{2} \ln \frac{1+a}{1-a}
\end{aligned}
$$

The flow angle ϕ is related to the streamline constant and velocity by

$$\phi = 2\pi - \arcsin\left(q/\psi\right) \tag{57}$$

Here, a fluid with $\gamma = 1.4$ was considered in the quadrant bounded by the streamlines $\psi_{\min} = 0.75$ and $\psi_{\max} = 1.5$. The inflow was situated along the iso-velocity line of $q_0 = 0.5$ and the outflow is situated along $y = 0$ or $q = k$. This configuration results in subsonic flow at the inlet that smoothly accelerates and transitions to supersonic. The Mach number contours for the resulting transonic flow are illustrated in Fig. 7.

For this test case, the spatial accuracy was assessed by measuring the error between the predicted and exact solutions for density. Calculations were performed with both $k = 1$ and $k = 4$ using a series of successfully refined meshes. The coarsest mesh that was used is illustrated in Fig. 7. It was created by first subdividing the domain into hexehedral elements: 31 hexahedra along the streamwise direction, 19 along the transverse direction (i.e., between the streamline boundaries), and 2 in the remaining (z) direction. Each hexahedral element was then split into 28 new tetrahedra to produce a final mesh with 32,984 elements and 7,321 vertices. Finer meshes were generated by refining the original hexahedral mesh and applying the same hexehedra-to-tetrahedra conversion procedure. Since the solution only varies in two directions, x and y, the domain was not refined in the z-direction. Rather, the thickness of the slab was varied to maintain the original aspect ratios of the elements.

As for the previous test case, computations were initialized with the analytic solution and the governing equations were relaxed until all equation residuals were reduced by four orders of magnitude. Tighter tolerances provided no significant gain in accuracy. All solutions were relaxed to a steady-state using the two-stage optimally smoothing scheme of van Leer et al. [69] with a CFL = 0.5 and the HLL numerical flux. Characteristic boundary conditions were applied at upstream and downstream boundaries, while the

Table 5. Global error norms and convergence rates of the predicted density for Ringleb's flow.

Nodes	Elements	Error			Convergence Rate		
		L_1	L_2	L_∞	L_1	L_2	L_∞
$k = 1$							
7,321	32,984	6.74×10^{-5}	1.82×10^{-4}	2.88×10^{-3}			
28,314	129,808	1.44×10^{-5}	3.68×10^{-5}	7.52×10^{-4}	-2.28	-2.36	-1.99
112,257	519,232	3.29×10^{-6}	8.00×10^{-6}	1.89×10^{-4}	-2.15	-2.22	-2.00
439,368	2,041,200	7.98×10^{-7}	1.89×10^{-6}	4.99×10^{-5}	-2.08	-2.12	-1.95
$k = 4$							
7,321	32,984	1.17×10^{-5}	6.31×10^{-5}	1.33×10^{-3}			
28,314	129,808	4.88×10^{-7}	4.18×10^{-6}	1.42×10^{-4}	-4.69	-4.01	-3.31
112,257	519,232	1.22×10^{-8}	1.43×10^{-7}	9.73×10^{-6}	-5.35	-4.90	-3.90
439,368	2,041,200	3.04×10^{-10}	3.76×10^{-9}	3.80×10^{-7}	-5.42	-5.33	-4.75

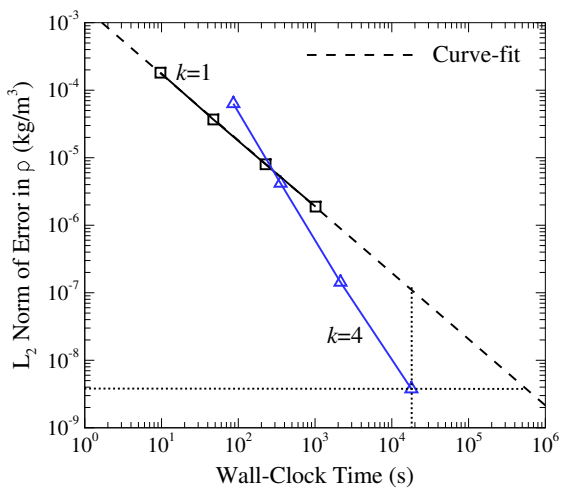


Fig. 8. Solution time for a given accuracy for Ringleb's flow.

exact solution was prescribed along the streamline boundaries. The mesh was assumed periodic in the z-direction.

The global error norms for the predicted density are provided in Table 5. Four different meshes were considered, and, as the meshes were refined, the L_1, L_2 and L_∞ error norms all converged towards their correct asymptotic behavior. Second-order accuracy was observed for $k = 1$ while fifth-order accuracy was observed for $k = 4$. This was confirmed by the computed convergence rates, which are also provided in Table 5.

Similar conclusions as those for the previous test case were reached upon comparing the computational cost for the two schemes as a function of error. As shown in Fig. 8, which depicts the L_2 error in ρ versus wall-clock time, the low-order scheme was more efficient for coarse meshes or higher error levels. But the computational savings of the fifth-order scheme quickly became significant as the desired error was reduced below approximately 10^{-5} in this case. Extrapolating the results for $k = 1$, the fifth-order ($k = 4$) scheme achieves an error approximately 30 times less than the second-order ($k = 1$) scheme for a wall-time of about 2×10^4 s. Additionally, it takes about a factor of 30 less time to get an error of 4×10^{-9} with the high-order scheme than with the low-order one. These results reaffirm the improved computational efficiency and accuracy of the proposed high-order CENO schemes when applied to smooth problems.

4.5 Convection of an Isentropic Vortex

The accuracy of the overall scheme was also verified for transient problems by studying the convection of an isentropic, cylindrical vortex. This is a smooth problem for which an exact solution is known. The flow is initially uniform with a density of $\rho_\infty = 1\,\mathrm{kg/m^3}$, a velocity of $\vec{v}_\infty = (u_\infty, v_\infty, w_\infty) = (1, 1, 0)\,\mathrm{m/s}$, and a temperature of $T_\infty = p_\infty/\rho_\infty = 1\,\mathrm{J/kg}$. At a time $t = 0\,\mathrm{s}$, the flow is perturbed by a cylindrical vortex centered at (x_0, y_0, z) that propagates undisturbed at a velocity of $\vec{v}_\infty$. This initial perturbation is defined by [79]

$$\delta T = \frac{(\gamma-1)\beta^2}{8\gamma\pi^2}\exp\left(1-r^2\right) \tag{58a}$$

$$\delta u = -\frac{\beta}{2\pi}\overline{y}\exp\left(\frac{1-r^2}{2}\right) \tag{58b}$$

$$\delta v = \frac{\beta}{2\pi}\overline{x}\exp\left(\frac{1-r^2}{2}\right) \tag{58c}$$

$$\delta w = 0 \tag{58d}$$

where δT and $(\delta u, \delta v, \delta w)$ are the perturbations in temperature and velocity, respectively, $r = \sqrt{\overline{x}^2 + \overline{y}^2}$ is the radial position relative to the vortex center, and $\beta = 5$ is a parameter which determines the strength of the vortex. The scaled coordinates are given by $\overline{x} = (x - x_0)/r_\mathrm{c}$ and $\overline{y} = (y - y_0)/r_\mathrm{c}$, where $r_\mathrm{c} = 0.1\,\mathrm{m}$ is the radius of the vortex core. Based on the perturbations defined by Eq. (58), the initial conditions for the isentropic flow at a time $t = 0\,\mathrm{s}$ are given by

$$\rho = (T_\infty + \delta T)^{1/(\gamma-1)} \tag{59a}$$

$$u = u_\infty + \delta u \tag{59b}$$

$$v = v_\infty + \delta v \tag{59c}$$

$$w = w_\infty + \delta w \tag{59d}$$

$$p = \rho^\gamma \tag{59e}$$

This solution is translationally invariant, i.e., the initial vortex at (x_0, y_0) is merely translated by a distance $\vec{v}_\infty t$ over a time t. Thus, the exact solution at any time is given by

$$\mathbf{U}(\vec{x}, t) = \mathbf{U}(\vec{x} - \vec{v}_\infty t, 0) \tag{60}$$

In this study, a rectangular domain was considered with $\Omega = [-L_x/2,\ L_x/2] \times [-L_y/2,\ L_y/2] \times [-L_z/2,\ L_z/2]$. The lengths L_x and L_y are equal to $1\,\mathrm{m}$, but, since this is a two dimensional problem, the length L_z was varied to maintain a computational domain that was only two layers of elements thick. At $t = 0\,\mathrm{s}$, the vortex was placed at $(x_0, y_0) = (-0.05, -0.05)\,\mathrm{m}$ and convected until $t_\mathrm{max} = 0.1\,\mathrm{s}$, which results in a final vortex centered at $(0.05, 0.05)\,\mathrm{m}$.

The domain was subdivided into tetrahedral meshes of varying resolutions. A sample mesh configuration along with the initial density contours at $t = 0\,\mathrm{s}$ are illustrated in Fig. 9. Characteristic boundary conditions were applied at all upstream and downstream boundaries while the mesh was assumed periodic in the z-direction.

4.5.1 Temporal Accuracy

First, a temporal refinement study was performed to ensure the asymptotic behavior of the time integration schemes. The study was performed in a similar manner to the spatial error analysis performed in the previous section, i.e., by analyzing the effect of the temporal resolution on the global solution error. For this study, several solutions were obtained using $k = 4$ and the HLL numerical flux on a single mesh with 57,344 uniform tetrahedra and 12,611 vertices. These solutions were obtained using the following time steps: $\Delta t = 1.0$, 0.5, 0.25, and $0.125\,\mu\mathrm{s}$.

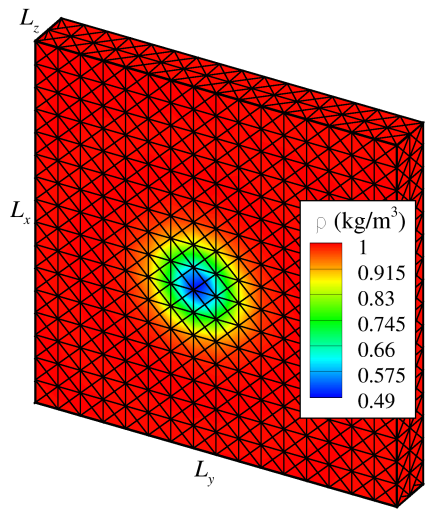
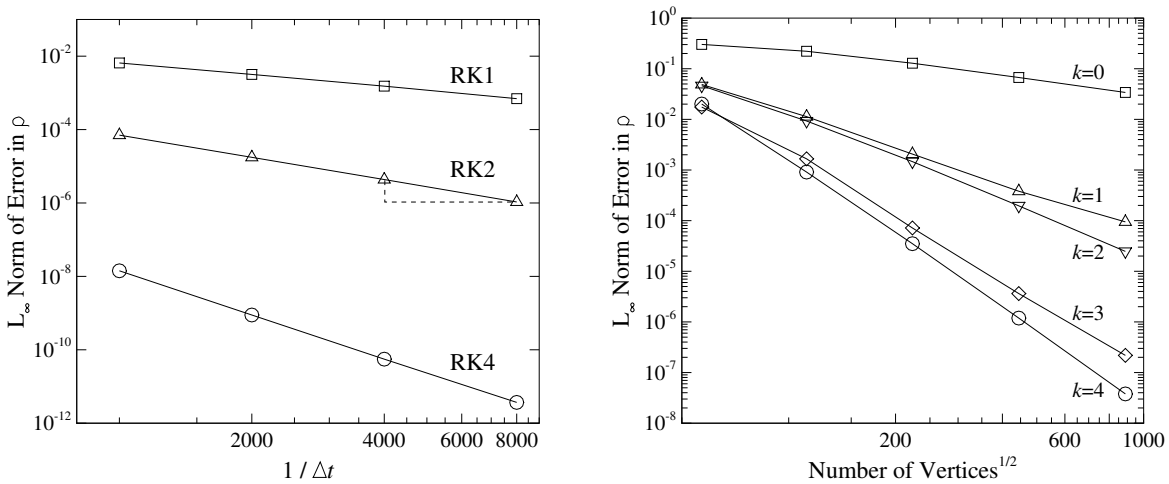


Fig. 9. Sample mesh and contours of density at $t = 0$ for the convection of an isentropic vortex.



(a) Temporal convergence. **(b)** Spatial convergence.

Fig. 10. Convergence of the error norms for the convection of an isentropic vortex.

For each time integration scheme, a reference or "exact" solution was obtained using a small time step and the error was approximated by the norm of the difference between the computed and reference solutions. This was necessary to eliminate the spatial errors and isolate the temporal ones. Here, reference solutions for each temporal discretization were obtained with a small time step of $\Delta t = 0.02\,\mu\text{s}$. The errors obtained with various RK schemes (RK1, RK2, and RK4) at $t = 0.1\,\text{s}$ are plotted in Fig. 10(a) and the convergence rates are provided in Table 6. These results show that the formal temporal accuracy was achieved by all schemes. They also demonstrate the drastic reductions in error that are possible using high-order time-marching schemes.

4.5.2 Spatial Accuracy

In addition to the temporal accuracy, the spatial accuracy was verified for this transient problem using different values of k between 1 and 4. All solutions were integrated in time using a fixed CFL and an s-stage RK scheme. For $k \leq 3$, the number of stages s was selected so that the temporal accuracy matched the spatial accuracy and a constant CFL number of 0.5 was used for all meshes. This ensured that the ratio between the spatial and temporal errors remained constant as the mesh was refined, since this ratio is proportional to $\Delta t^s/h^{k+1}$. However, since a fifth-order RK scheme was not implemented, solutions for

Table 6. Temporal convergence rates of the error norms for the convection of an isentropic vortex.

Scheme	Formal	L_1	L_2	L_∞
RK1	1	1.13	1.13	1.13
RK2	2	2.03	2.03	2.03
RK4	4	3.82	3.98	3.91

Table 7. Spatial convergence rates of the error norms for the convection of an isentropic vortex.

k	Formal	L_1	L_2	L_∞
0	1	0.85	0.86	0.94
1	2	2.38	2.32	2.45
2	3	2.96	2.94	2.91
3	4	4.45	4.41	4.32
4	5	4.97	4.96	4.90

$k = 4$ (fifth-order accurate in space) were obtained with RK4 (fourth-order accurate in time) instead. And because the behavior of the two errors do not match, i.e., temporal and spatial, the CFL number was varied according to the following relation for $k = 4$:

$$\frac{\mathrm{CFL}}{\mathrm{CFL}_0} = \left(\frac{h}{h_0}\right)^{1/4} \tag{61}$$

where $\mathrm{CFL}_0 = 0.5$ is the CFL number used for the coarsest mesh with mesh spacing h_0, and h is the spacing for the mesh in question. This formula ensured that the ratio between the spatial and temporal errors was always maintained constant when using $k = 4$ with RK4.

For the spatial accuracy study, solutions were obtained on a sequence of finer grids with different values of k. Spatial errors were computed using the analytical solution as a reference and are shown in Fig. 10(b). The corresponding convergence rates are provided in Table 7. These results verify that, for a transient problem, the proposed finite-volume formulation achieved the expected formal spatial accuracy for all values of k considered.

To compare the computational costs associated with each high-order scheme, the solution error is plotted as a function of the wall-clock time in Fig. 11. As previously shown for steady problems, the high-order

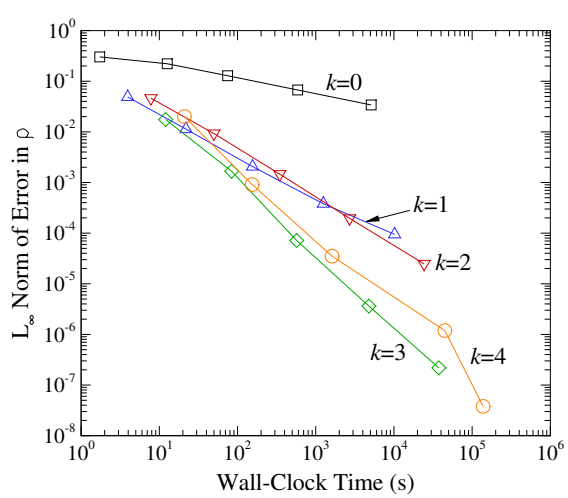


Fig. 11. Solution time to a given level of accuracy for the convection of an isentropic vortex.

Table 8. Mesh sizes used for the shock tube problem.

	Nodes	Elements	Average Edge Length (mm)
Mesh 1	1,484	5,745	16.6
Mesh 2	9,654	45,960	8.16
Mesh 3	69,035	367,680	4.06
Mesh 4	520,821	2,941,440	2.03

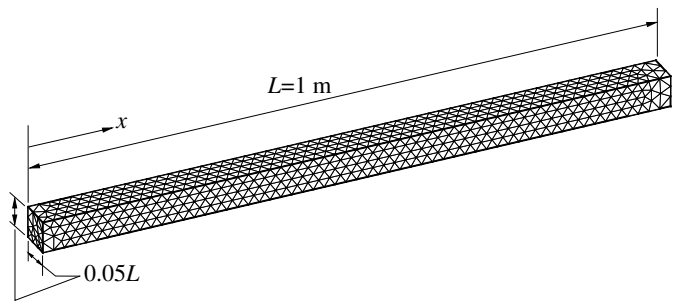


Fig. 12. Sample computational mesh and domain used for shock tube problem.

schemes are more efficient in terms of accuracy versus computational cost when the target accuracy is high. That is, there is a particular range of accuracy for which a particular value of k is the most efficient, and this optimal value of k increases with the desired level of accuracy.

4.6 Shock Tube

The robustness and accuracy of the algorithm was demonstrated for non-smooth problems with a one-dimensional shock-tube [80]. This time-dependent problem was solved on a rectangular domain of length 1 m, with the following initial conditions:

$$\mathbf{W}(x,0) = \begin{cases} \mathbf{W_L} & \text{if } x \leq 0.45\,\text{m}, \\ \mathbf{W_R} & \text{if } x > 0.45\,\text{m} \end{cases} \tag{62}$$

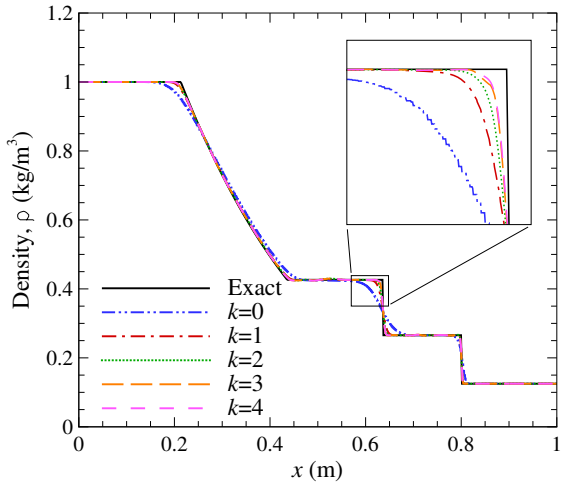
where $\mathbf{W_L} = \left[1\,\text{kg/m}^3,\, 0,\, 0,\, 0,\, 1\,\text{Pa}\right]$ and $\mathbf{W_R} = \left[0.125\,\text{kg/m}^3,\, 0,\, 0,\, 0,\, 0.1\,\text{Pa}\right]$. A sample of the meshes used is illustrated in Fig. 12 along with the dimensions of the computational domain. The sizes and average edge length for each mesh are provided in Table 8.

Solutions were obtained for different values of k, using meshes of increasing resolution. All solutions were integrated in time until $t = 0.2\,\text{s}$ with the RK4 time-marching scheme, a CFL of 0.2, and the HLL numerical flux. Reflection boundary conditions were applied to the surrounding surfaces while the solution was free to vary at both ends of the tube.

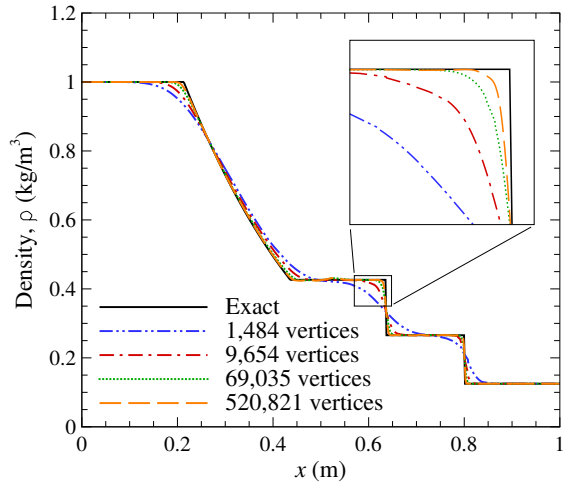
Solutions on the finest mesh considered are compared with the exact solution at $t = 0.2\,\text{s}$ in Fig. 13(a). Overall, there was a distinct improvement in the numerical solution as the polynomial degree was increased, even near discontinuities. This is highlighted for the contact surface in the inset of Fig. 13(a). There was an initial large improvement in the solution as k was increased from 0 to 1. Further increases in k provided smaller and smaller improvements in the solution.

A similar comparison is made in Fig. 13(b), which illustrates the effect of mesh resolution on the fifth-order ($k = 4$) scheme. As expected, increasing the mesh resolution improves the agreement of the numerical solution with the exact solution.

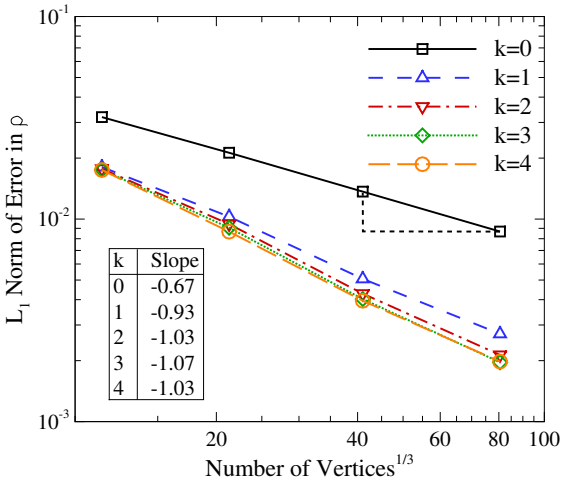
The behavior of the L_1 norm of the error in predicted density is demonstrated in Fig. 13(c). The first-order ($k = 0$) scheme does not quite reach the asymptotic region. It only achieves an order of accuracy of approximately 0.7. The $k = 1$ scheme achieved a significant reduction in error over the $k = 0$ scheme, but



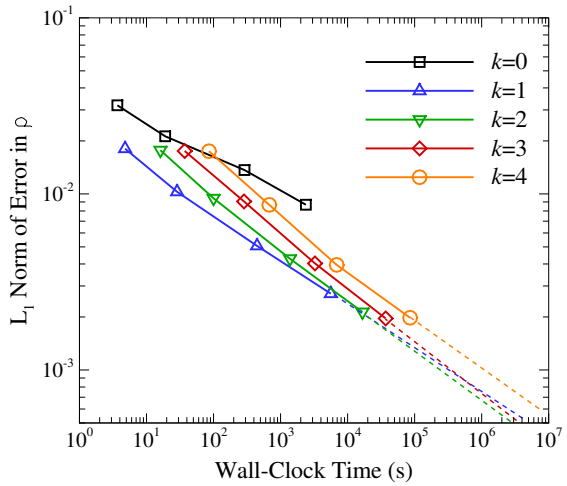
(a) Effect of polynomial degree on predictions using a mesh with 520,821 vertices.



(b) Effect of mesh size on predictions for $k = 4$.



(c) Convergence of the error norms.



(d) Accuracy as a function of solution time. Dashed lines represent extrapolations.

Fig. 13. Results for one-dimensional shock-tube at $t = 2\,\mathrm{s}$.

only converges at a rate of $\mathcal{O}(h)$ due to the discontinuities present in the solution. All higher-order schemes only achieved first-order accuracy as well, which was expected because of the discontinuities in the solution, but there was still a decrease in overall error as k was increased beyond unity. Even though CENO drops to first-order near discontinuities, the size of the region influenced by the discontinuity decreases with mesh size. As such, there is a net reduction in error. For example, using the finest mesh considered for this problem (Mesh 4), the error obtained using the second-order ($k = 1$) scheme is 3.2 times smaller than the error obtained using the first-order ($k = 0$) scheme. The fifth-order ($k = 4$) scheme achieves an even larger reduction in error. It is 4.4 times smaller than the error obtained using the first-order ($k = 0$) scheme.

Similar to the previous test involving smooth supersonic flow (Section 4.3), the computational efficiency was assessed in terms of the wall-clock time to a given level of error in Fig. 13(d). The $k = 1$ scheme was the most efficient over the range of meshes studied. However, extrapolating to lower error levels, the $k = 2$ scheme is expected to be more efficient for error levels below 0.02. For example, extrapolating the results illustrated in Fig. 13(d), the $k = 2$ scheme is expected to take approximately half the wall-time than the $k = 1$ scheme to achieve an error of 5×10^{-4}. The efficiencies of the other high-order schemes, i.e., $k = 3$ and $k = 4$, are expected to improve as the desired error is lowered further.

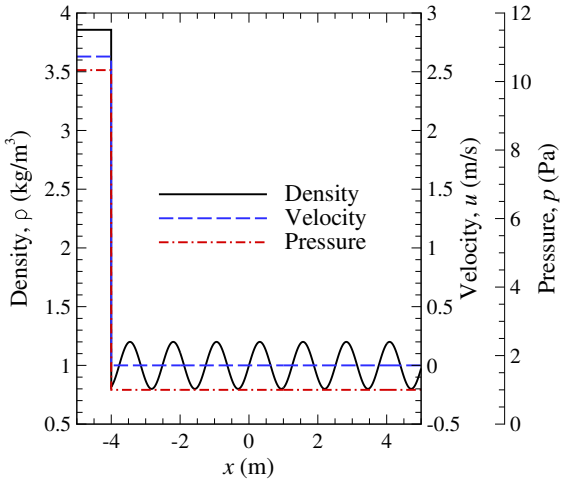


Fig. 14. Initial conditions at time $t = 0$ for the Shu-Osher problem.

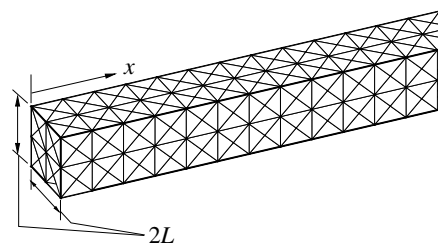


Fig. 15. Sample computational mesh for the Shu-Osher problem.

For problems with discontinuities and no structure in the smooth regions, such as the shock tube problem studied here, the benefits of the proposed scheme in terms of improved global error are not as great as they were for completely smooth problems (Sections 4.3–4.5). As shown in Fig. 13(d), overly fine meshes are required for the proposed high-order schemes to demonstrate improved efficiency. Greater benefits are expected for these types of problems using the high-order schemes in conjunction with AMR. AMR would reduce the local error near the discontinuities and the unlimited high-order reconstruction would reduce the errors in the smooth regions.

4.7 Shu-Osher Problem

The advantages of the proposed high-order scheme are more significant for problems with discontinuities when the smooth regions involve more complex variations. To illustrate this, the one-dimensional Shu-Osher problem [81] was studied. This particular problem involves a shock wave moving at Mach 3 and interacting with a sinusoidal density wave. It is similar to the shock tube studied in the previous section but contains more smoothly varying features. For this problem the initial solution at time $t = 0$ is given by

$$\mathbf{W}(x,0) = \begin{cases} \mathbf{W_L} & \text{if } x < -4 \\ \mathbf{W_R} & \text{if } x \geq -4 \end{cases} \tag{63}$$

where

$$\mathbf{W_L} = \begin{bmatrix} 3.857143 \\ 2.629369 \\ 0 \\ 0 \\ 10.33333 \end{bmatrix} \qquad \mathbf{W_R} = \begin{bmatrix} 1 + 0.2\sin 5x \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

and $\mathbf{W}(x,0)$ is defined on $x \in [-5,\ 5]$. The initial conditions are illustrated in Fig. 14.

This problem was investigated using the three-dimensional tube-shaped mesh illustrated in Fig. 15. The mesh is only two cells wide in the y- and z-directions. It consists of 46,413 vertices and 179,200 elements with an average edge length of 4.5×10^{-3}. Predictions were obtained using both the formally second- ($k = 1$) and fifth-order ($k = 4$) schemes. All solutions were integrated in time until $t = 1.8\,\text{s}$ with the RK4 time-marching scheme, a CFL of 0.2, and the HLL numerical flux. Characteristic boundary conditions were applied at $x = -5$ and $x = 5$ while all other boundaries were assumed periodic. Since an exact solution is not known for this particular problem, a reference solution was computed using a one-dimensional version of the proposed finite-volume algorithm and a finely resolved mesh with 2000 elements.

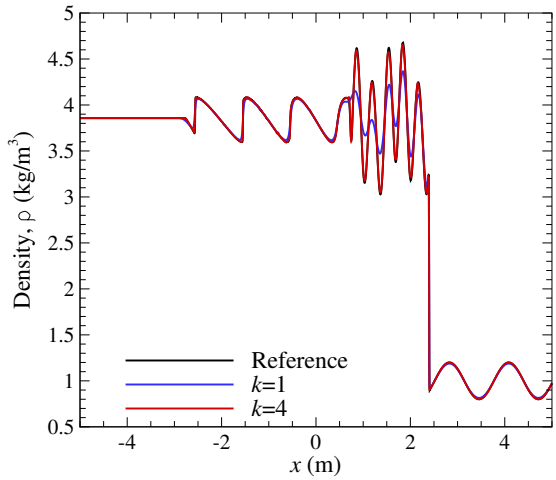


Fig. 16. Predicted density for Shu-Osher problem at $t = 1.8\,\mathrm{s}$. Solutions were obtained with $S_\mathrm{c} = 200$.

Predictions for density are compared with the reference solution at $t = 1.8\,\mathrm{s}$ in Fig. 16. These predictions were obtained with a smoothness indicator cutoff of $S_\mathrm{c} = 200$ since this value produces a result with low diffusion. As illustrated in Fig. 16, both the second- ($k = 1$) and fifth-order ($k = 4$) solutions agree with the reference solution. The high-order discretization does a much better job representing the small-scale variations, i.e., the peaks in front and behind of the shock. There are no oscillations observed across the shock for the high-order CENO predictions either. This demonstrates the ability of the proposed scheme to resolve smoothly varying features with high-order accuracy even in the presence of discontinuities.

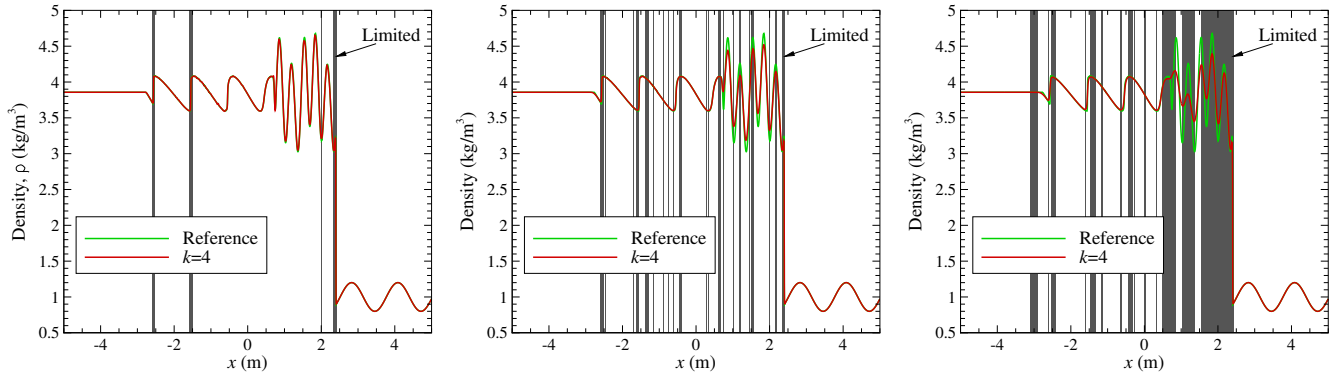
The regions where the reconstructed density was indicated to be non-smooth is illustrated in Fig. 17(a). Only a few small regions were flagged as non-smooth. One small region surrounding the shock was flagged along with two regions begin the shock where the compressed entropy waves begin to steepen. There was also a small region just behind the shock where the density oscillated with a high frequency. While these three limited regions behind the shock do not necessarily possess discontinuities in the PDEs themselves, there were significant jumps in the predicted control-volume averages through these regions. As such, these regions were deemed under-resolved and limited to avoid generating any spurious oscillations.

Most test cases considered up to this point used a slightly larger cutoff than the one used for this problem. A cutoff of $S_\mathrm{c} = 2000$ was used for the previous cases while $S_\mathrm{c} = 200$ was used for the Shu-Osher problem described in this section. It was mentioned in Section 3.1.3 that $S_\mathrm{c} = 2000$ provided the best balance between accuracy and stability for a wide range of problems. Smaller values of S_c offer less numerical diffusion and improved accuracy, but can be less robust depending upon the case. Larger values are more dissipative but they offer improved robustness. Figure 17 illustrates this effect of the cutoff S_c on the predicted density for the Shu-Osher problem. As illustrated in the figure, a significant improvement in the predicted solution was observed as S_c was reduced. Nevertheless, no oscillations were observed in any of the high-order solutions for this problem, even when using more relaxed smoothness requirements.

4.8 Sedov Blast Wave

The next two problems were used to demonstrate the robustness of the proposed algorithm. The first of these two, the Sedov explosion problem [82], involves the evolution of a spherical blast wave from an initial pressure singularity in an otherwise homogeneous medium. This problem challenges a scheme's robustness and ability to remain monotone since any oscillations near the blast front can easily cause negative densities or pressures.

The blast wave is generated by an initial energy source, e_blast, located in a small region of radius r_0 near the origin. In practice, it is difficult to define a small radius r_0 without an overly fine mesh near the origin, especially when using tetrahedral mesh. So the total amount of initial energy e_blast was deposited at $t = 0\,\mathrm{s}$ into a single control volume at the origin only, hereafter referred to as control volume Ω_o. The



(a) $S_c = 200$. **(b)** $S_c = 400$. **(c)** $S_c = 2000$.

Fig. 17. Effect of smoothness indicator on predicted density for Shu-Osher problem. The solutions correspond to a time of $t = 1.8\,\mathrm{s}$.

Table 9. Mesh sizes used for the Sedov blast wave problem.

	Sphere			Cube		
	Nodes	**Elements**	**Average Edge Length (mm)**	**Nodes**	**Elements**	**Average Edge Length (mm)**
Mesh 1	1,999	9,615	98.6	5,631	28,000	84.2
Mesh 2	14,364	76,920	48.6	42,461	224,000	41.9
Mesh 3	108,655	615,360	24.2	140,491	756,000	27.9
Mesh 4	844,701	4,922,880	12.1	329,721	1,792,000	20.9
Mesh 5				640,151	3,500,000	16.7

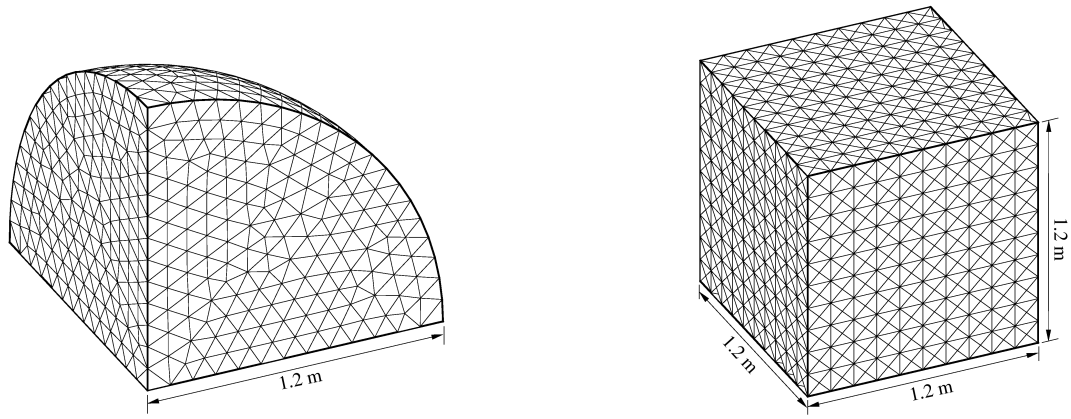
initial conditions at time $t = 0\,\mathrm{s}$ are

$$\rho(\vec{x},\,0) = 1\,\mathrm{kg/m^3}, \qquad \vec{v}(\vec{x},\,0) = \vec{0}\,\mathrm{m/s}, \qquad p(\vec{x},\,0) = \begin{cases} (\gamma-1)\,\rho\,\dfrac{e_{\mathrm{blast}}}{V_{\mathrm{o}}} & \text{if } \vec{x} \in \Omega_{\mathrm{o}} \\ 10^{-5}\,\mathrm{Pa} & \text{otherwise} \end{cases}$$

where $e_{\mathrm{blast}} = 0.851072\,\mathrm{J\,m^3}$, and V_{o} is the volume of the control volume that the energy is deposited into. This configuration gives a blast wave that reaches $r = \sqrt{x^2+y^2+z^2} = 1\,\mathrm{m}$ at $t = 1\,\mathrm{s}$. An exact solution for this spherical blast wave is available using the numerical algorithm described by Kamm [83].

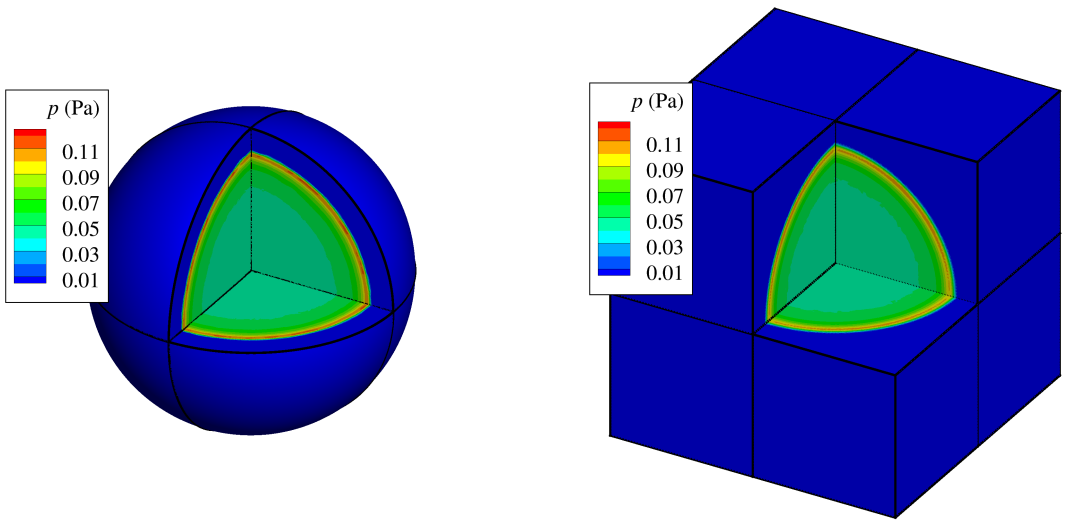
Two different types of domains/meshes were considered to investigate their effects on the accuracy of predicted solutions: a spherical domain with irregular meshes and a cube-shaped domain with regular meshes. Samples of the two types of computational meshes are presented in Fig. 18. In both cases, only an octant of the computational domain was modeled using reflection boundary conditions to enforce symmetry. Thus, V_0 is eight times the volume of the control volume at the origin. The outer surfaces of the octant were also treated as a reflecting wall, but this boundary condition had no effect on the predicted solution since the blast wave never reached the outer surfaces over the simulation time considered. The radius of the sphere and the length of the sides of the cube were both 1.2 m.

All solutions were integrated in time until $t = 1\,\mathrm{s}$ with the RK4 time-marching scheme, a CFL of 0.2, and the Rusanov numerical flux. Simulations were obtained with $k = 0, 1, \ldots, 4$ on a series of different, successively-refined meshes. The number of vertices and tetrahedral elements as well as the average edge length for each mesh considered are provided in Table 9. Note that the finest meshes for the spherical and cube-shaped domains have a similar degree of resolution, i.e., the average edge lengths for the two meshes are of the same order of magnitude.



(a) Sphere. **(b)** Cube.

Fig. 18. Sample computational meshes used for the Sedov problem.

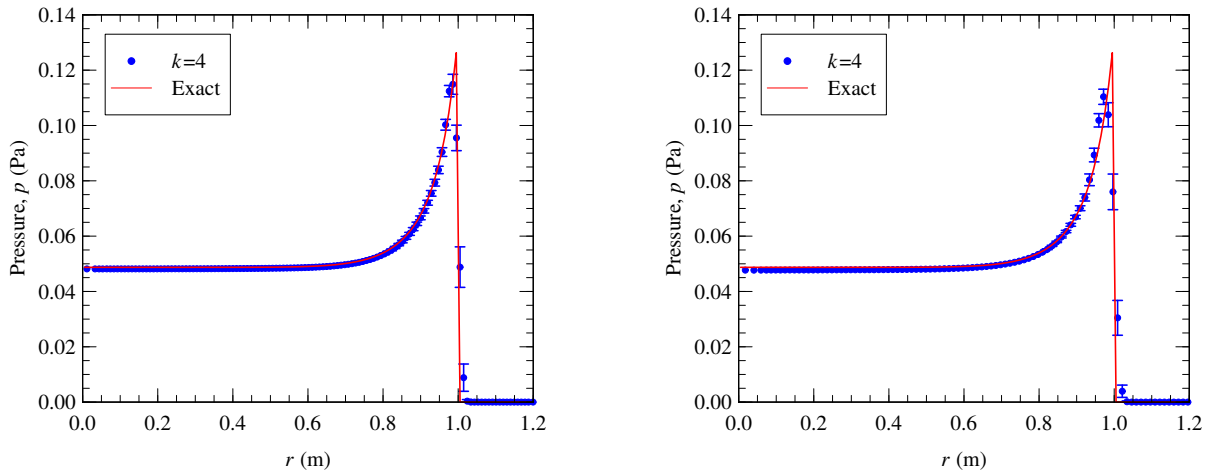


(a) Sphere. **(b)** Cube.

Fig. 19. Predicted pressure contours for the Sedov problem at $t = 1\,\mathrm{s}$. The predictions were obtained with $k = 4$ on the finest mesh (i.e., Mesh 4 for the spherical domain and Mesh 5 for the cube-shaped domain).

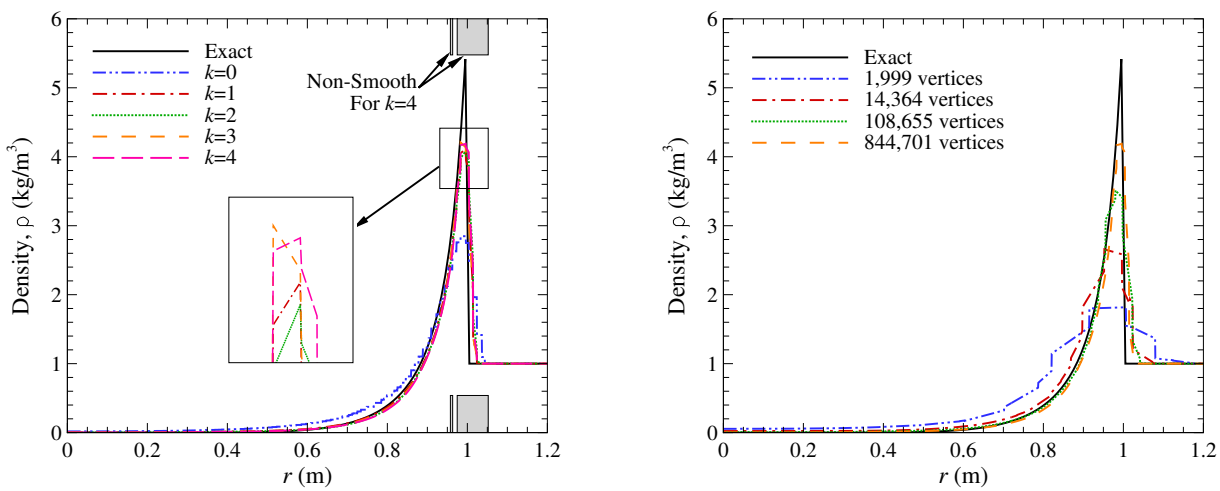
Three dimensional contours of pressure at $t = 1\,\mathrm{s}$ are illustrated in Fig. 19. These predictions were obtained using the finest mesh of both domains (i.e., Mesh 4 for the spherical domain and Mesh 5 for the cube-shaped domain) with the $k = 4$ CENO discretization. As illustrated in Fig. 19, there was no noticeable effect of mesh configuration on the overall solution for pressure. Additionally, both solutions where monotone and displayed an excellent degree of symmetry.

The degree of spherical symmetry in the predicted pressure was quantified by measuring the mean and standard deviation of this quantity as a function of radius. As observed in Fig. 20, which illustrates the variation of the mean and standard deviation of pressure with radius, there was little deviation in pressure at a particular fixed radius. These results were obtained with $k = 4$ on the finest spherical and cube-shaped meshes. In front of and behind the shock, where the solution was relatively smooth, the deviation from the mean was close to zero. The largest deviations occurred at the shock front in both cases, and this deviation was similar for both types of meshes. For example, the maximum standard deviation of the predicted pressure for the finest cube-shaped and spherical meshes were 16.2 and 21.4 %, respectively. And both of these maximums occurred at the shock front, i.e., $r = 1$ m, where the CENO discretization used a limited piecewise linear reconstruction. The limiter does not produce symmetric results since the



(a) Sphere. **(b)** Cube.

Fig. 20. Mean and deviation of the predicted density as a function of radius for the Sedov problem at $t = 1\,\mathrm{s}$. Symbols represent the mean while error bars represent the standard deviation. The predictions were obtained with $k = 4$ on the finest mesh (i.e., Mesh 4 for the spherical domain and Mesh 5 for the cube-shaped domain).



(a) Effect of polynomial degree on predictions obtained using the spherical Mesh 4 (844,701 vertices).

(b) Effect of mesh size on predictions for $k = 4$.

Fig. 21. Predicted density along a line passing through $(0, 0, 0)$ and $(1, 1, 1)$ for the Sedov problem at $t = 1\,\mathrm{s}$. These results were obtained using the irregular spherical meshes.

shock front is not aligned with the mesh. Additionally, larger deviations occurred along the boundaries because the limiter tended to be more conservative near boundaries than in the interior of the domain. This produced asymmetries in the limiter values, which in turn generated asymmetries in the predictions. Nonetheless, the largest standard deviation for each case got smaller as the mesh spacing was decreased.

Predictions for density obtained using the spherical domain are compared with the analytical solution at $t = 1\,\mathrm{s}$ in Fig. 21. These predictions were obtained by interpolating the solution along a line passing through the origin and the coordinate $(x, y, z) = (1, 1, 1)$. For all values of k and meshes employed, no oscillations were observed in front of or behind the shock wave. For the high-order solutions, i.e., $k > 1$, the smoothness indicator correctly identified the large solution discontinuity at the moving shock front. For

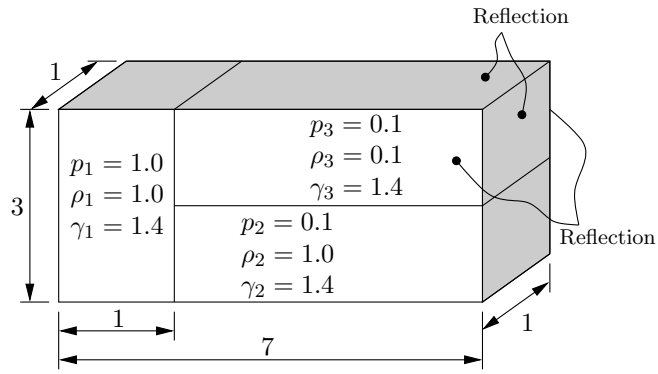


Fig. 22. Computational domain and initial conditions for the triple-point problem.

example, for the $k = 4$ case on the spherical Mesh 4, the region along this line that was deemed non-smooth was confined to the area surrounding the shock, as illustrated in Fig. 21(a). Similar results were obtained using the regular cube-shaped meshes.

Figure 21(a) illustrates the effect of polynomial order on the predicted density for the finest spherical mesh investigated (Mesh 4 with 844,701 vertices and 4,922,880 tetrahedra). As was also observed for the shock tube test problem in Section 4.6, increasing the order of the polynomial from $k = 0$ to $k = 1$ provided a significant improvement in the predicted solution. Although the improvements were less drastic for the higher-order solutions, i.e., $k > 1$, the peak density predicted using $k = 3$ and 4 is noticeably improved over $k = 1$. It is not clear why the peak density predicted for $k = 2$ is slightly less than that predicted with $k = 1$.

The effect of mesh resolution on the predicted density that was obtained using the 5th-order ($k = 4$) CENO reconstruction is illustrated in Fig. 21(b). At low resolutions, the outwardly propagating shock wave was smeared over several cell widths. However, as the mesh resolution was increased, the shock front steepens and the accuracy of the numerical solution improves. Only the region directly surrounding the shock was limited (see Fig. 21(a)).

The most significant result for this test case is that all of the high-order solutions displayed excellent robustness without any oscillations. No negative densities or pressures were ever encountered in any of the numerical solutions.

4.9 Triple-Point Shock Interaction

As a final test of the algorithm's robustness, a three-state Riemann problem was studied. The test problem considered here is a single-material, three-dimensional variant of the original two-dimensional problem proposed by Galera et al. [84]. This problem does not have an exact solution, but it was studied here because of the difficulty in resolving the interaction between shocks and contact discontinuities without generating spurious oscillations. It consists of a computational domain $\Omega = [0,\ 7] \times [0,\ 3] \times [0,\ 1]$ that is subdivided into three separate regions defined by $\Omega_1 = [0,\ 1] \times [0,\ 3] \times [0,\ 1]$, $\Omega_2 = [1,\ 7] \times [0,\ 1.5] \times [0,\ 1]$, and $\Omega_3 = [1,\ 7] \times [1.5,\ 3] \times [0,\ 1]$. As illustrated in Fig. 22, a different set of initial conditions is prescribed for each region at time $t = 0$. The initial discontinuity between Ω_1 and Ω_2 generates a contact discontinuity, a rightward shock wave and leftward rarefaction wave. The same wave pattern is generated by the discontinuity between Ω_1 and Ω_3, except that the shock propagates at a faster speed in Ω_3. This difference in propagation speed creates a strong shear along the initial contact discontinuity located at the interface between Ω_2 and Ω_3 that eventually generates a high-speed vortex.

Two solutions were obtained on a mesh with 1,866,149 vertices and 10,712,865 tetrahedra, one with a second-order polynomial representation ($k = 1$) and one with a fifth-order ($k = 4$) representation. Both solutions were integrated in time until $t = 5\,\mathrm{s}$ with the RK4 time-marching scheme, a CFL of 0.25, and the HLL numerical flux. Reflection conditions were applied to all boundary surfaces as indicated in Fig. 22.

Numerical predictions for density at $t = 5\,\mathrm{s}$ are compared in Fig. 23. Significant differences between

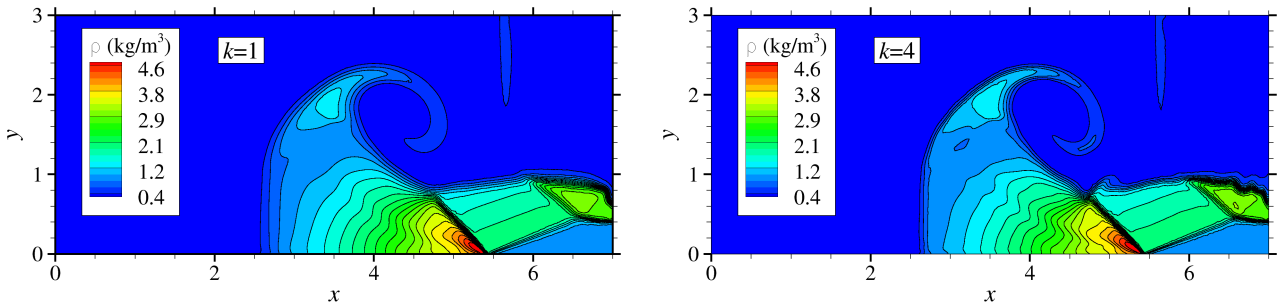


Fig. 23. Predicted density field at $t = 5\,\mathrm{s}$ for triple-point problem.

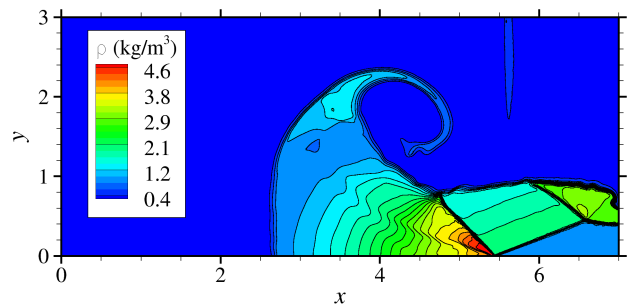


Fig. 24. Reference solution of the density field at $t = 5\,\mathrm{s}$ for triple-point problem. This solution was obtained using CHICOMA [51] on a mesh with 78 million tetrahedra.

the two solutions are visible. Most notably, the high-order CENO solution does a better job resolving the different types of discontinuities. Predicted gradients are steeper and the vortex rolls up further in the high-order prediction. There are also some Kelvin-Helmholtz-like instabilities that develop along the contact discontinuity in the fifth-order solution that are not observed in the second-order predictions. To verify that these instabilities are not spurious oscillations generated by the numerical method, a reference solution was generated using CHICOMA [51] with a much finer mesh of 78 million tetrahedra. The reference solution for density at $t = 5\,\mathrm{s}$ is presented in Fig. 24. As observed in the figure, similar Kelvin-Helmholtz-like instabilities form along the surface of the contact.

These results highlight the robustness of the high-order CENO algorithm since the $k = 4$ scheme was able to reliably obtain a solution without producing any unphysical oscillations. They also highlight the improved accuracy for problems with strong discontinuities. The high-order solution observed in Fig. 23 is in better agreement with the reference solution presented in Fig. 24. More improvements over the second-order scheme are expected when the CENO algorithm is used in conjunction with AMR. AMR will help reduce the control volume sizes near the shock and their contribution to the overall global error.

5. Conclusions

A high-order finite-volume scheme was developed for the mathematical description of compressible fluids on unstructured meshes. It is a vertex-based variant of the cell-based, Godunov-type, finite-volume methods developed by Ivan et al. [36–40] and Charest et al. [32–35], which use a hybrid CENO reconstruction procedure to avoid spurious oscillations. The scheme was assessed in terms of accuracy and computational cost for a variety of problems, including smooth and discontinuous function reconstructions, and solutions to idealized flow problems.

Up to fifth-order accuracy was demonstrated. For smooth flows and function reconstructions, an order of accuracy of $(k+1)$ was achieved using piecewise polynomial representations of degree k. Only first-order accuracy was observed for all problems that contained discontinuities, but there was still a measured advantage provided by the high-order schemes. They displayed lower errors for a given mesh. This advantage

was more distinct for discontinuous problems with structure in the smooth regions, like the Shu-Osher and triple-point problems.

In terms of computational efficiency, i.e., wall time for a given accuracy, there was an optimal value of k which varied depending upon the desired error and the particular problem. The standard second-order scheme was the most efficient for higher error levels, and the high-order schemes became more efficient as the desired error was decreased. This was demonstrated for smooth and discontinuous problems, although the mesh sizes at which the high-order schemes were more efficient was significantly larger for discontinuous problems. In one smooth case, the wall-time to a given level of error was reduced by a factor of approximately 24 when the high-order CENO scheme was compared with the second-order scheme.

For problems with discontinuities and no structure in the smooth regions, such as the shock tube and Sedov blast wave problems, the benefits of the proposed scheme in terms of improved global error and computational efficiency were not as great. This is because the global error for these cases was dominated by the local error near discontinuities, and regions surrounding discontinuities are treated using a limited linear reconstruction in the CENO procedure. As such, the key result of applying the proposed CENO algorithm to these types of problems is that no spurious oscillations were generated and that the solution remained essentially non-oscillatory. Greater benefits are expected for these types of problems using the proposed high-order scheme in conjunction with AMR. AMR would reduce the local error near the discontinuities and the unlimited high-order reconstruction would reduce the errors in the smooth regions.

Overall, this research highlights the main advantages of the CENO finite-volume algorithm. High-order accuracy was achieved in smooth regions, while robust and monotone solutions were maintained near discontinuities and under-resolved solution content. Future work consists of further development and validation of the proposed algorithm, including its extension to multi-material problems, arbitrary equations of state, moving meshes, and adaptive mesh refinement.

Acknowledgments

This research was supported by the United States Department of Energy, through the Advanced Simulation & Computing (ASC) and Metropolis postdoctoral fellowship programs.

References

[1] D.J. Mavriplis. Unstructured-mesh discretizations and solvers for computational aerodynamics. AIAA J., 46(6):1281–1298, 2008.

[2] S. Pirozzoli. On the spectral properties of shock-capturing schemes. J. Comput. Phys., 219(2):489–497, 2006.

[3] A. Harten, B. Engquist, S. Osher, and S.R. Chakravarthy. Uniformly high order accurate essentially non-oscillatory schemes, III. J. Comput. Phys., 71(2):231–303, 1987.

[4] T.J. Barth. Recent developments in high order k-exact reconstruction on unstructured meshes. AIAA Paper 93-0668, 1993.

[5] R. Abgrall. On essentially non-oscillatory schemes on unstructured meshes: analysis and implementation. J. Comput. Phys., 114:45–58, 1994.

[6] T. Sonar. On the construction of essentially non-oscillatory finite volume approximations to hyperbolic conservation laws on general triangulations: polynomial recovery, accuracy and stencil selection. Comp. Meth. Appl. Mech. Eng., pp. 140–157, 1997.

[7] C.F. Ollivier-Gooch. Quasi-ENO schemes for unstructured meshes based on unlimited data-dependent least-squares reconstruction. J. Comput. Phys., 133:6–17, 1997.

[8] G.S. Jiang and C.W. Shu. Efficient implementation of weighted ENO schemes. J. Comput. Phys., 126(1):202–228, 1996.

[9] D. Stanescu and W. Habashi. Essentially nonoscillatory Euler solutions on unstructured meshes using extrapolation. AIAA J., 36:1413–1416, 1998.

[10] O. Friedrich. Weighted essentially non-oscillatory schemes for the interpolation of mean values on unstructured grids. J. Comput. Phys., 144(1):194–212, 1998.

[11] C. Hu and C.W. Shu. Weighted essentially non-oscillatory schemes on triangular meshes. J. Comput. Phys., 150:97–127, 1999.

[12] C.F. Ollivier-Gooch and M. Van Altena. A high-order accurate unstructured mesh finite-volume scheme for the advection-diffusion equation. J. Comput. Phys., 181(2):729–752, 2002.

[13] A. Nejat and C. Ollivier-Gooch. A high-order accurate unstructured finite volume Newton-Krylov algorithm for inviscid compressible flows. J. Comput. Phys., 227(4):2582–2609, 2008.

[14] B. Cockburn and C.W. Shu. TVB Runge-Kutta local projection discontinous Galerkin finite-element method for conservation laws II: General framework. Math. Comp., 52:411, 1989.
[15] B. Cockburn, S. Hou, and C.W. Shu. TVB Runge-Kutta local projection discontinous Galerkin finite-element method for conservation laws IV: The multidimensional case. J. Comput. Phys., 54:545, 1990.
[16] R. Hartmann and P. Houston. Adaptive discontinuous Galerkin finite element methods for the compressible Euler equations. J. Comput. Phys., 183:508–532, 2002.
[17] H. Luo, J.D. Baum, and R. Löhner. A Hermite WENO-based limiter for discontinuous Galerkin method on unstructured grids. J. Comput. Phys., 225:686–713, 2007.
[18] G. Gassner, F. Lörcher, and C.D. Munz. A contribution to the construction of diffusion fluxes for finite volume and discontinuous Galerkin schemes. J. Comput. Phys., 224(2):1049–1063, 2007.
[19] F. Bassi and S. Rebay. A high-order accurate discontinuous finite element method for the numerical solution of the compressible Navier-Stokes equations. J. Comput. Phys., 131:267–279, 1997.
[20] B. Cockburn and C.W. Shu. The local discontinuous Galerkin method for time-dependent convection diffusion system. SIAM J. Numer. Anal., 35(6):2440–2463, 1998.
[21] B. Leervan , M. Lo, and M. Raaltevan . A discontinuous Galerkin method for diffusion based on recovery. AIAA Paper 2007-4083, 2007.
[22] M. Raaltevan and B. Leervan . Bilinear forms for the recovery-based discontinuous Galerkin method for diffusion. Commun. Comput. Phys., 5(2–4):683–693, 2009.
[23] H. Liu and J. Yan. The direct discontinuous Galerkin (DDG) methods for diffusion problems. SIAM J. Numer. Anal., 41(1):675–698, 2009.
[24] Z.J. Wang. Spectral (finite) volume method for conservation laws on unstructured grids – basic formulation. J. Comput. Phys., 178:210–251, 2002.
[25] Z.J. Wang and Y. Liu. Spectral (finite) volume method for conservation laws on unstructured grids – ii. extenstion to two-dimensional scalar equation. J. Comput. Phys., 179:665–697, 2002.
[26] Z.J. Wang, L. Zhang, and Y. Liu. High-order spectral volume method for 2d euler equations. Paper 2003–3534, AIAA, June 2003.
[27] Z.J. Wang and Y. Liu. Spectral (finite) volume method for conservation laws on unstructured grids – iii. one dimensional systems and partition optimization. Journal of Scientific Computing, 20(1):137–157, 2004.
[28] Y. Sun, Z.J. Wang, and Y. Liu. Spectral (finite) volume method for conservation laws on unstructured grids VI: extension to viscous flow. J. Comput. Phys., 215(1):41–58, 2006.
[29] H. Huynh. A flux reconstruction approach to high-order schemes including discontinuous Galerkin methods. AIAA Paper 2007-4079, 2007.
[30] Z.J. Wang and H. Gao. A unifying lifting collocation penalty formulation for the Euler equations on mixed grids. AIAA Paper 2009–401, 2009.
[31] Z.J. Wang and H. Gao. A unifying lifting collocation penalty formulation including the discontinuous Galerkin, spectral volume/difference methods for conservation laws on mixed grids. J. Comput. Phys., 228:8161–8186, 2009.
[32] S.D. McDonald, M.R.J. Charest, and C.P.T. Groth. High-order CENO finite-volume schemes for multi-block unstructured mesh. 20th AIAA Computational Fluid Dynamics Conference, Honolulu, Hawaii, June 27–30 2011. doi: 10.2514/6.2011-3854. AIAA-2011-3854.
[33] M.R.J. Charest, C.P.T. Groth, and P.Q. Gauthier. High-order CENO finite-volume scheme for low-speed viscous flows on three-dimensional unstructured mesh. ICCFD7 - International Conference on Computational Fluid Dynamics, Hawaii, July 9–13 2012. Paper ICCFD7-1002.
[34] M.R.J. Charest and C.P.T. Groth. A high-order central ENO finite-volume scheme for three-dimensional turbulent reactive flows on unstructured mesh. 21st AIAA Computational Fluid Dynamics Conference, San Diego, California, June 24–27 2013. doi: 10.2514/6.2013-2567. AIAA 2013-2567.
[35] M.R.J. Charest, C.P.T. Groth, and P.Q. Gauthier. A high-order central ENO finite-volume scheme for three-dimensional low-speed viscous flows on unstructured mesh. Commun. Comput. Phys., 2015. Accepted for publication.
[36] L. Ivan and C.P.T. Groth. High-order central CENO finite-volume scheme with adaptive mesh refinement. AIAA paper 2007-4323, 2007.
[37] L. Ivan and C.P.T. Groth. High-order solution-adaptive central essentially non-oscillatory (CENO) method for viscous flows. AIAA paper 2011-0367, 2011.
[38] L. Ivan and C.P.T. Groth. High-order central ENO scheme with adaptive mesh refinement for hyperbolic conservation laws. Commun. Comput. Phys., 2013. submitted for publication.
[39] L. Ivan and C.P.T. Groth. High-order solution-adaptive central essentially non-oscillatory (CENO) method for viscous flows. J. Comput. Phys., 257:830–862, 2013.
[40] A. Susanto, L. Ivan, H. De Sterck, and C.P.T. Groth. High-order central ENO finite-volume scheme for ideal MHD. J. Comput. Phys., 250(1):141 – 164, 2013.
[41] A. Harten and S.R. Chakravarthy. Multi-dimensional ENO schemes for general geometries. ICASE Report No. 91-76, 1991.
[42] Y.T. Zhang and C.W. Shu. Third order WENO scheme on three dimensional tetrahedral meshes. Commun. Comput. Phys., 5(2-4):836–848, 2009.
[43] M. Dumbser and M. Käser. Arbitrary high order non-oscillatory finite volume schemes on unstructured meshes for linear

hyperbolic systems. J. Comput. Phys., 221(2):693–723, 2007.
[44] P. Tsoutsanis, V.A. Titarev, and D. Drikakis. WENO schemes on arbitrary mixed-element unstructured meshes in three space dimensions. J. Comput. Phys., 230(4):1585–1601, 2011.
[45] Y. Liu and Y.T. Zhang. A robust reconstruction for unstructured WENO schemes. J. Sci. Comput., 54(2-3):603–621, 2013.
[46] A. Haselbacher. A WENO reconstruction algorithm for unstructed grids based on explicit stencil construction. AIAA paper 2005-0879, 2005.
[47] J. Waltz. Derived data structure algorithms for unstructured finite element meshes. Int. J. Numer. Meth. Engin., 54(7): 945–963, 2002.
[48] J. Waltz. Parallel adaptive refinement for unsteady flow calculations on 3d unstructured grids. Int. J. Numer. Meth. Fluids, 46(1):37–57, 2004.
[49] J. Waltz. Microfluidics simulation using adaptive unstructured grids. Int. J. Numer. Meth. Fluids, 46(9):939–960, 2004.
[50] J. Waltz, T.R. Canfield, N.R. Morgan, L.D. Risinger, and J.G. Wohlbier. Verification of a three-dimensional unstructured finite element method using analytic and manufactured solutions. Comput. Fluids, 2013.
[51] J. Waltz, N.R. Morgan, T.R. Canfield, M.R.J. Charest, L.D. Risinger, and J.G. Wohlbier. A three-dimensional finite element arbitrary Lagrangian-Eulerian method for shock hydrodynamics on unstructured grids. Comput. Fluids, 92: 172–187, 2013.
[52] J. Waltz, N.R. Morgan, T.R. Canfield, M.R.J. Charest, and J.G. Wohlbier. A nodal Godunov method for Lagrangian shock hydrodynamics on unstructured tetrahedral grids. Int. J. Numer. Meth. Fluids, 76(3):129–146, 2014.
[53] N.R. Morgan, J. Waltz, D.E. Burton, M.R.J. Charest, T.R. Canfield, and J.G. Wohlbier. A Godunov-like point-centered essentially Lagrangian hydrodynamic approach. J. Comput. Phys., 281:614–652, 2014.
[54] C.A. Felippa. A compendium of FEM integration formulas for symbolic work. Eng. Computation., 21(8):867–890, 2004.
[55] T.J. Barth and P.O. Fredrickson. Higher order solution of the Euler equations on unstructured grids using quadratic reconstruction. AIAA Paper 90-0013, 1990.
[56] D.J. Mavriplis. Revisiting the least-squares procedure for gradient reconstruction on unstructured meshes. AIAA paper 2003-3986, 2003.
[57] A. Jalali and C. Ollivier-Gooch. Higher-order finite volume solution reconstruction on highly anisotropic meshes. AIAA Paper 2013-2565, 2013.
[58] C.L. Lawson and R.J. Hanson. Solving least squares problems. Prentice-Hall, 1974.
[59] K. Michalak and C. Ollivier-Gooch. Matrix-explicit GMRES for a higher-order accurate inviscid compressible flow solver. AIAA paper 2007-3943, 2007.
[60] N.R. Draper and H. Smith. Applied regression analysis. Wiley, New York, 3rd edition, 1998.
[61] T.J. Barth. A three-dimensional upwind Euler solver for unstructured meshes. AIAA Paper 91-1548, 1991.
[62] J.S. Park, S.H. Yoon, and C. Kim. Multi-dimensional limiting process for hyperbolic conservation laws on unstructured grids. J. Comput. Phys., 229(3):788–812, 2010.
[63] V. Venkatakrishnan. On the accuracy of limiters and convergence to steady state solutions. AIAA Paper 93-0880, 1993.
[64] S.K. Godunov. Fintite-difference method for numerical computations of discontinuous solutions of the equations of fluid dynamics. Mat. Sb., 47:271–306, 1959.
[65] V.V. Rusanov. Calculation of intersection of non-steady shock waves with obstacles. J. Comput. Math. Phys. USSR, 1: 267–279, 1961.
[66] S.F. Davis. Simplified second-order Godunov-type methods. SIAM J. Sci. Stat. Comput., 9(3):445–473, 1988.
[67] A. Harten, P.D. Lax, and B. Leervan . On upstream differencing and Godunov-type schemes for hyperbolic conservation laws. SIAM Rev., 25(1):35–61, 1983.
[68] H. Lomax, T.H. Pulliam, and D.W. Zingg. Fundamentals of Computational Fluid Dynamics. Springer, New York, 2003.
[69] B. Leervan , C. Tai, and K.G. Powell. Design of optimally smoothing multi-stage schemes for the Euler equations. AIAA Paper 89-1933, 1989.
[70] J. Berntsen, R. Cools, and T.O. Espelid. Algorithm 720: An algorithm for adaptive cubature over a collection of 3-dimensional simplices. ACM Trans. Math. Softw., 19(3):320–332, 1993.
[71] R. Abgrall. Design of an essentially non-oscillatory reconstruction procedure on finite-element type meshes. ICASE Contractor Report 189574, 1991.
[72] P.J. Roache and S. Steinberg. Symbolic manipulation and computational fluid dynamics. AIAA J., 22(10):1390–1394, 1984.
[73] W.L. Oberkampf and F.G. Blottner. Issues in computational fluid dynamics code verification and validation. AIAA J., 36(5):687–695, 1998.
[74] P.J. Roache. Code verification by the method of manufactured solutions. J. Fluids Eng., 124(1):4–10, 2001.
[75] C.J. Roy, T.M. Smith, and C.C. Ober. Verification of a compressible cfd code using the method of manufactured solutions. AIAA Paper 2002-3110, 2002.
[76] C.J. Roy, C.C. Nelson, T.M. Smith, and C.C. Ober. Verification of Euler/NavierâĂŞStokes codes using the method of manufactured solutions. Int. J. Numer. Meth. Fluids, 44(6):599–620, 2004.
[77] F. Ringleb. Exakte lösungen der differentialgleichungen einer adiabatischen gasströmung. ZAMM, 20(4):185–198, 1940.
[78] W.J. Coirier and K.G. Powell. An accuracy assessment of Cartesian-mesh approaches for the Euler equations. J. Comput. Phys., 117:121–131, 1995.

[79] H.C. Yee, N.D. Sandham, and M.J. Djomehri. Low-dissipative high-order shock-capturing methods using characteristic-based filters. J. Comput. Phys., 150(1):199–238, 1999.
[80] G.A. Sod. A survey of several finite difference methods for systems of nonlinear hyperbolic conservation laws. J. Comput. Phys., 27(1):1 – 31, 1978.
[81] C.W. Shu and S. Osher. Efficient implementation of essentially non-oscillatory shock-capturing schemes, II. J. Comput. Phys., 83(1):32–78, 1989.
[82] L.I. Sedov. Similarity and Dimensional Methods in Mechanics. Academic Press, New York, 1959.
[83] J. Kamm. Evaluation of the Sedov-von Neumann-Taylor blast wave solution. Technical Report LA-UR-00-6055, Los Alamos National Laboratory, 2000.
[84] S. Galera, P.H. Maire, and J. Breil. A two-dimensional unstructured cell-centered multi-material ALE scheme using VOF interface reconstruction. J. Comput. Phys., 229(16):5755–5787, 2010.