

High-Order CENO Finite-Volume Scheme for Low-Speed Viscous Flows on Three-Dimensional Unstructured Mesh

M.R.J. Charest*, C.P.T. Groth* and P.Q. Gauthier**
Corresponding author: charest@utias.utoronto.ca

* University of Toronto Institute for Aerospace Studies
4925 Dufferin Street, Toronto, Ontario, Canada M3H 5T6

** Energy Engineering & Technology, Rolls-Royce Canada Limited
9245 Côte-de-Liesse, Dorval, Québec, Canada H9P 1A5

Abstract: High-order discretization techniques offer the potential to significantly reduce the computational costs necessary to obtain accurate predictions when compared to lower-order methods. However, efficient, universally-applicable, high-order discretizations remain somewhat illusive, especially for more arbitrary unstructured meshes and for incompressible/low-speed flows. A novel, high-order, central essentially non-oscillatory (CENO), cell-centered, finite-volume scheme is proposed for the solution of the conservation equations of viscous, incompressible flows on three-dimensional unstructured meshes. The proposed scheme is applied to the pseudo-compressibility formulation of the steady and unsteady Navier-Stokes equations and the resulting discretized equations are solved with a parallel implicit Newton-Krylov algorithm. For unsteady flows, the temporal derivatives are discretized using the family of high-order backward difference formulas (BDF) and the resulting equations are solved via a dual-time stepping approach. The proposed finite-volume scheme for fully unstructured mesh is demonstrated to provide both fast and accurate solutions for steady and unsteady viscous flows.

Keywords: Numerical Algorithms, Computational Fluid Dynamics, High-Order Methods, Incompressible Flows.

1 Introduction

Computational fluid dynamics (CFD) has proven to be an important enabling technology in many areas of science and engineering. In spite of the relative maturity and widespread success of CFD in aerospace engineering, there is a variety of physically-complex flows which are still not well understood and are very challenging to predict by numerical methods. Such flows include, but are not limited to, multiphase, turbulent, and combusting flows encountered in propulsion systems (e.g., gas turbine engines and solid propellant rocket motors). These flows present numerical challenges as they generally involve a wide range of complicated physical/chemical phenomena and scales.

Many flows of engineering interest are incompressible or can be approximated as incompressible to a good degree of accuracy, i.e. low-speed flows. Incompressible flows can prove challenging to solve numerically because the partial derivative of density with respect to time vanishes. As a result, the governing equations are ill-conditioned. Various methods for solving the incompressible Navier-Stokes equations have been successfully developed to overcome this ill-conditioning [1, 2]. These include but are not limited to the pressure-Poisson [3, 4], fractional-step [5, 6], vorticity-based [7, 8], and pseudo-compressibility methods [9]. The equations governing fully-compressible flows have also been successfully applied to incompressible and low-speed flows using preconditioning techniques [10–15]. The pseudo-compressible formulation [9, 16–23] is attractive because it is easily extended to three dimensions and applied in conjunction with high-order schemes. This method was originally referred to as the artificial compressibility method by Chorin [9] but Chang and Kwak [24] introduced the possibly more accurate name “pseudo-compressibility

method”.

High-order methods have the potential to significantly reduce the cost of modelling physically-complex flows, but this potential is challenging to fully realize. As such, the development of robust and accurate high-order methods remains an active area of research. Standard lower-order methods (i.e, methods up to second order) can exhibit excessive numerical dissipation for multi-dimensional problems and are often not practical for physically-complex flows. High-order methods offer improved numerical efficiency for accurate solution representations since fewer computational cells are required to achieve a desired level of accuracy [25]. For hyperbolic conservation laws and/or compressible flow simulations, the main challenge involves obtaining accurate discretizations while ensuring that discontinuities and shocks are handled reliably and robustly [26]. High-order schemes for elliptic partial differential equations (PDEs) that govern diffusion processes should satisfy a maximum principle, even on stretched/distorted meshes, while remaining accurate [27]. There are many studies of high-order schemes developed for finite-volume [26, 28–37], discontinuous Galerkin [38–42], and spectral finite-difference/finite-volume methods [43–47] on both structured and unstructured mesh. In spite of many advances, there is still no consensus for a robust, efficient, and accurate scheme that fully deals with all of the aforementioned issues and is universally applicable to arbitrary meshes.

Harten et al. [26] originally proposed the essentially non-oscillatory (ENO) high-order finite-volume scheme which achieves monotonicity by avoiding the use of computational stencils that contain discontinuities. Weighted ENO (WENO) schemes attempt to simplify the ENO procedure by adopting a stencil-weighting approach [32, 34, 35]. However, both the ENO and WENO schemes encounter difficulties when selecting appropriate stencils on general multi-dimensional unstructured meshes [29, 30, 33, 48] and using these stencils can produce poor conditioning of the linear systems involved in performing the solution reconstruction [33, 48]. These difficulties, along with the associated computational cost and complexities of the ENO and WENO finite-volume schemes, have somewhat limited the applicable range of ENO and WENO.

Ivan and Groth [49, 50] proposed a high-order Central Essentially Non Oscillatory (CENO), cell-centered, finite-volume scheme that was demonstrated to remain both accurate and robust in a variety of physically-complex flows. The CENO scheme is based on a hybrid solution reconstruction procedure that combines an unlimited high-order k -exact, least-squares reconstruction technique with a monotonicity preserving limited piecewise linear least-squares reconstruction algorithm. Fixed central stencils are used for both the unlimited high-order k -exact reconstruction and the limited piecewise linear reconstruction. Switching between the two reconstruction algorithms is determined by a solution smoothness indicator that indicates whether or not the solution is resolved on the computational mesh. This hybrid approach avoids the complexities associated with reconstruction on multiple stencils that other essentially non-oscillatory (ENO) and weighted ENO schemes can encounter. Originally developed for structured two-dimensional mesh, this scheme has been successfully extended to two- and three-dimensional unstructured mesh by McDonald et al. [51].

The application of high-order solution methods to the pseudo-compressibility approach is not new. Rogers and Kwak [21, 52] and Qian and Zhang [23] employed high-order finite-difference discretizations up to fifth- and sixth-order accuracy, respectively. Using a finite-volume discretization, Chen et al. [53] applied a fifth-order WENO scheme on two-dimensional structured mesh. However, these discretizations are not easily applied to three-dimensional unstructured mesh.

Implicit solution algorithms are commonly applied to improve the stability and convergence of pseudo-compressibility approaches. Implicit algorithms that have been applied to the pseudo-compressible formulation of the Navier-Stokes equations include: approximate factorization [19, 54], LU-SGS/SSOR algorithms [23, 53, 55] and line-relaxation techniques [22, 52]. Due to various approximations and/or linearizations, these schemes are not fully-implicit and their application to unstructured mesh is not straightforward. Jacobian-free Newton-Krylov methods [56–59] offer significant improvements over these types of implicit schemes in terms of rapid convergence. They can robustly handle stiff-wave systems, strong non-linear couplings between equations and offer the potential of quadratic convergence [59, 60].

All of the applications of the pseudo-compressibility formulation discussed previously have focused on constant density flows. They are not directly applicable to more general low-speed flows that involve combustion or multiple fluids/species. Several researchers have applied the standard pseudo-compressibility approach in conjunction with interface-capturing methods to track the discontinuities in density encountered in multi-fluid flows [61, 62]. There are other approaches that are more applicable to the combusting flows encountered in propulsion systems. For example, Riedel [63] applied the pseudo-compressibility approach to reacting flows using an artificial-dissipation-based finite-volume method. A characteristic-based scheme using the pseudo-compressibility approach was derived by Shapiro and Drikakis [64] which is applicable for variable-density, multi-species, isothermal flows. A similar characteristic-based

scheme for constant density flows with heat transfer was developed by Azhdarzadeh and Razavi [65].

In this paper, the high-order CENO finite-volume scheme is extended to solve the equations governing incompressible, viscous, laminar flows with variable density on three-dimensional general unstructured mesh. For steady flows, the equations are solved using the pseudo-compressibility approach coupled with an implicit Newton-Krylov algorithm. The proposed scheme is extended to unsteady flows via a dual-time stepping approach. The resulting algorithm is applied to both steady and unsteady flows and analyzed in terms of accuracy, computational cost, and parallel performance. In particular, the spatial and temporal accuracy of solutions are examined and the influence of mesh resolution on accuracy is assessed for several idealized flow problems. Both the steady flow over an isothermal flat plate and the unsteady decay of Taylor vortices are studied here.

2 Pseudo-Compressibility Approach for Variable Density Low Speed Flows

In the present research, the equations governing viscous, laminar, compressible flows at low Mach numbers are considered. In three space dimensions, the governing partial-differential equations are

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \vec{v}) = 0 \quad (1a)$$

$$\frac{\partial}{\partial t}(\rho \vec{v}) + \nabla \cdot (\rho \vec{v} \vec{v} + p \vec{\mathbf{I}}) = \nabla \cdot \vec{\tau} \quad (1b)$$

$$\frac{\partial}{\partial t}(\rho h) + \nabla \cdot (\rho \vec{v} h) = \nabla \cdot \vec{q} \quad (1c)$$

where t is the time, p is the total pressure, ρ is the fluid density, \vec{v} is the bulk fluid velocity vector, $h = \int_{T_0}^T c_p dT$ is the fluid enthalpy, c_p is the fluid specific heat, T is the temperature, $\vec{q} = -\lambda \nabla T$ is the heat flux vector, and λ is the fluid thermal conductivity. The fluid stress tensor is given by

$$\tau_{ij} = \mu \left[\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) - \frac{2}{3} \delta_{ij} \frac{\partial u_k}{\partial x_k} \right] \quad (2)$$

where μ is the dynamic viscosity. At low speeds, density becomes weakly coupled to pressure via the ideal gas law. Here, we assume that pressure is constant and density is a function of temperature only, $\rho = \rho(T)$.

For low-Mach-number and incompressible flows, the pseudo-compressibility method modifies the partial derivatives of density with respect to time [9, 16, 18, 66]. In the original formulation of Chorin [9] for incompressible flows, a pressure time derivative was added to the steady form of the continuity equation and the primitive form of the governing equations were solved using a time-marching procedure. Turkel [18] derived the conservative form of Chorin's modified governing equations and showed that time derivatives of pressure should also be added to the momentum equations. Applying the pseudo-compressibility approach to Eq. (1), the resulting governing equations are

$$\frac{1}{\beta} \frac{\partial p}{\partial \tau} + \nabla \cdot (\rho \vec{v}) = 0 \quad (3a)$$

$$\rho \frac{\partial \vec{v}}{\partial \tau} + \frac{\alpha \vec{v}}{\beta} \frac{\partial p}{\partial \tau} + \nabla \cdot (\rho \vec{v} \vec{v} + p \vec{\mathbf{I}}) = \nabla \cdot \vec{\tau} \quad (3b)$$

$$\rho \frac{\partial h}{\partial \tau} + \frac{\alpha h}{\beta} \frac{\partial p}{\partial \tau} + \nabla \cdot (\rho \vec{v} h) = \nabla \cdot \vec{q} \quad (3c)$$

where β is the pseudo-compressibility factor, α is a preconditioning parameter, and τ denotes the pseudo-time since the modified equations are no longer time-accurate. The preconditioning parameter, α , controls how the original governing equations are modified. The original pseudo-compressibility method of Chorin [9] corresponds to $\alpha = 0$. When $\alpha = 1$ or 2, the pressure time derivatives are added directly to the conserved or primitive formulation of the governing equations, respectively.

Time-accuracy is regained using a dual-time-stepping approach [21, 22, 67–69]. The time-accurate form of Eq. (3) is

$$\frac{\partial \mathbf{U}}{\partial t} + \mathbf{\Gamma} \frac{\partial \mathbf{W}}{\partial \tau} + \frac{\partial}{\partial x} (\mathbf{E} - \mathbf{E}_v) + \frac{\partial}{\partial y} (\mathbf{F} - \mathbf{F}_v) + \frac{\partial}{\partial z} (\mathbf{G} - \mathbf{G}_v) = 0 \quad (4)$$

where \mathbf{U} and \mathbf{W} are the vectors of conserved and primitive variables, $\vec{\mathbf{F}} = [\mathbf{E}, \mathbf{F}, \mathbf{G}]$ and $\vec{\mathbf{F}}_v = [\mathbf{E}_v, \mathbf{F}_v, \mathbf{G}_v]$ are the inviscid and viscous solution flux dyads, and $\mathbf{\Gamma}$ is the transformation matrix. They are defined as

$$\mathbf{U} = \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ \rho w \\ \rho h \end{bmatrix}, \quad \mathbf{W} = \begin{bmatrix} p \\ u \\ v \\ w \\ T \end{bmatrix}, \quad \mathbf{E} = \begin{bmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ \rho uw \\ \rho uh \end{bmatrix}, \quad \mathbf{F} = \begin{bmatrix} \rho v \\ \rho vu \\ \rho v^2 + p \\ \rho vw \\ \rho vh \end{bmatrix}, \quad \mathbf{G} = \begin{bmatrix} \rho w \\ \rho wu \\ \rho wv \\ \rho w^2 + p \\ \rho wh \end{bmatrix},$$

$$\mathbf{E}_v = \begin{bmatrix} 0 \\ \tau_{xx} \\ \tau_{xy} \\ \tau_{xz} \\ \lambda \frac{\partial T}{\partial z} \end{bmatrix}, \quad \mathbf{F}_v = \begin{bmatrix} 0 \\ \tau_{yx} \\ \tau_{yy} \\ \tau_{yz} \\ \lambda \frac{\partial T}{\partial z} \end{bmatrix}, \quad \mathbf{G}_v = \begin{bmatrix} 0 \\ \tau_{zx} \\ \tau_{yz} \\ \tau_{zz} \\ \lambda \frac{\partial T}{\partial z} \end{bmatrix}, \quad \mathbf{\Gamma} = \begin{bmatrix} \frac{1}{\beta} & 0 & 0 & 0 & 0 \\ \frac{\alpha}{\beta} u & \rho & 0 & 0 & 0 \\ \frac{\alpha}{\beta} v & 0 & \rho & 0 & 0 \\ \frac{\alpha}{\beta} w & 0 & 0 & \rho & 0 \\ \frac{\alpha}{\beta} h & 0 & 0 & 0 & \rho c_p \end{bmatrix}.$$

2.1 Eigenstructure

Based on the analysis conducted by Turkel [18] and the numerical results obtained by Qian et al. [70] and Lee and Lee [71], the optimal value of α is 2. However, Malan et al. [72, 73] and Lee and Lee [71] found that a loss of robustness can occur for $\alpha > 1$ if β is too small. This loss of robustness occurs because the determinant of the modal matrix can be zero when $\alpha > 1$. Since larger values of α display better convergence characteristics [70, 71], $\alpha = 1$ was selected for the current work. For $\alpha = 1$, the Jacobian matrix of the inviscid system with respect to the primitive variables is

$$\mathbf{A} = \mathbf{\Gamma}^{-1} \frac{\partial(\vec{\mathbf{F}} \cdot \hat{n})}{\partial \mathbf{W}} = \begin{bmatrix} 0 & n_x \rho \beta & n_y \rho \beta & n_z \rho \beta & 0 \\ \frac{n_x}{\rho} & q & 0 & 0 & 0 \\ \frac{n_y}{\rho} & 0 & q & 0 & 0 \\ \frac{n_z}{\rho} & 0 & 0 & q & 0 \\ 0 & 0 & 0 & 0 & q \end{bmatrix} \quad (5)$$

where $q = \vec{v} \cdot \hat{n}$ and $c^2 = q^2 + 4\beta$. The resulting matrix of right eigenvectors is

$$\mathbf{R} = \begin{bmatrix} -\frac{\rho}{2}(q-c) & -\frac{\rho}{2}(q+c) & 0 & 0 & 0 \\ n_x & n_x & -n_y & -n_z & 0 \\ n_y & n_y & n_x & 0 & 0 \\ n_z & n_z & 0 & n_x & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad (6)$$

Table 1. Gauss quadrature rules used for cell face integration.

Reconstruction	Tetrahedra		Cartesian / Hexahedra	
	Points	Polynomial Degree	Points	Polynomial Degree
Constant ($k=0$)	1	1	1	1
Linear ($k=1$)	1	1	1	1
Quadratic ($k=2$)	3	2	4	3
Cubic ($k=3$)	4	3	4	3
Quartic ($k=4$)	6	4	9	5

The eigenvalues of the inviscid system defined by Eq. (3) in a particular direction are

$$\lambda = \begin{bmatrix} \frac{1}{2}u_n - \frac{1}{2}\sqrt{u_n^2 + 4\beta} \\ u_n \\ u_n \\ u_n \\ \frac{1}{2}u_n + \frac{1}{2}\sqrt{u_n^2 + 4\beta} \end{bmatrix} \quad (7)$$

where u_n is the velocity of the bulk flow projected onto the direction vector of interest.

3 CENO Finite-Volume Scheme

In the proposed cell-centered finite-volume approach, the physical domain is discretized into finite-sized computational cells and the integral forms of conservation laws are applied to each individual cell. For a cell i , the approach results in the following coupled system of partial differential equations (PDEs) for cell-averaged solution quantities:

$$\frac{d\bar{\mathbf{U}}_i}{dt} + \Gamma_i \frac{d\bar{\mathbf{W}}_i}{d\tau} = -\frac{1}{V_i} \oint (\bar{\mathbf{F}} - \bar{\mathbf{F}}_v) \cdot \hat{n} dA = -\mathbf{R}_i \quad (8)$$

where the overbar denotes cell-averaged quantities, V_i is the cell volume, A is the area of the face and \hat{n} is the unit vector normal to a given face. Applying Gauss quadrature to evaluate the surface integral in Eq. (8) produces a set of nonlinear ordinary differential equations (ODEs) given by

$$\frac{d\bar{\mathbf{U}}_i}{dt} + \Gamma_i \frac{d\bar{\mathbf{W}}_i}{d\tau} = -\frac{1}{V_i} \sum_{l=1}^{N_f} \sum_{m=1}^{N_G} [\omega (\bar{\mathbf{F}} - \bar{\mathbf{F}}_v) \cdot \hat{n} A]_{i,l,m} \quad (9)$$

where N_f is the number of faces (equal to 4 for tetrahedra and 6 for hexahedra), N_G is the number of quadrature points and ω is the corresponding quadrature weight. In Eq. (9), the number of quadrature points required along each face is a function of the reconstruction order and number of spatial dimensions. For tetrahedra and Cartesian (hexahedra with rectangular faces) cells, Gauss quadrature points can be directly mapped from the canonical form to the Cartesian coordinate system. More general hexahedra can have non-rectangular faces or faces composed of vertices that do not all lie on a particular plane. In this case, the Gauss quadrature points are mapped to the Cartesian coordinate system using a trilinear coordinate transformation [74, 75]. The coefficients for the quadrature rules applied here are tabulated by Felippa [76] and summarized in Table 1.

3.1 CENO Reconstruction

Evaluating Eq. (9) requires integration of the numerical flux along the cell faces, but only cell-averaged quantities are known. The high-order CENO method uses a hybrid solution reconstruction process to interpolate the primitive solution state at the Gauss quadrature points along each face [49, 50]. This hybrid approach involves a fixed central

stencil in smooth or fully-resolved regions which is switched to a limited piecewise linear reconstruction when discontinuities in solution content are encountered. This switching provides a means of eliminating spurious oscillations that can occur near regions where the solution is under-resolved. It is facilitated by a parameter called the smoothness indicator which indicates the current level of resolution.

Even though most features of low-speed flows are relatively smooth, there are cases where discontinuities can occur, such as across flame fronts or fluid interfaces. Oscillations can even occur for relatively smooth flows when there is insufficient mesh resolution.

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 [28]. The k -exact higher-order reconstruction algorithm begins by assuming that the solution within each cell can be represented by the following Taylor series expansion in three dimensions:

$$u_i^k(x, y, z) = \sum_{p_1=0}^{(p_1+p_2+p_3) \leq k} \sum_{p_2=0} \sum_{p_3=0} (x - x_i)^{p_1} (y - y_i)^{p_2} (z - z_i)^{p_3} D_{p_1 p_2 p_3} \quad (10)$$

where u_i^k is the reconstructed solution quantity, (x_i, y_i, z_i) are the coordinates of the cell centroid, k is the order of the piecewise polynomial interpolant and $D_{p_1 p_2 p_3}$ are the unknown coefficients of the Taylor series expansion. The summation indices, p_1 , p_2 and p_3 , must always satisfy the condition that $(p_1 + p_2 + p_3) \leq k$.

The following conditions are applied to determine the unknown coefficients: i) the solution reconstruction must reproduce polynomials of degree $N \leq k$ exactly; ii) the mean or average value within the computational cell must be preserved; and iii) the reconstruction must have compact support. The second condition states that

$$\bar{u}_i = \frac{1}{V_i} \iiint_{V_i} u_i^k(x, y, z) dV \quad (11)$$

where \bar{u}_i is the cell average.

The third condition dictates the number and location of neighboring cells 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. (11)). For any type of mesh, the total number of unknown coefficients for a particular order is given by

$$N = \frac{1}{d!} \prod_{n=1}^d (k + n) \quad (12)$$

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. Additional neighbors are included to ensure that the stencil is not biased in any particular direction and that the reconstruction remains reliable on poor quality meshes with high aspect ratio cells [49]. For each neighboring cell, p , a constraint is formed by requiring that

$$\bar{u}_p = \frac{1}{V_p} \iiint_{V_p} u_i^k(x, y, z) dV \quad (13)$$

Since the constraints of Eqs. (11) and (13) result in an over-determined system of linear equations, a least-squares solution for the coefficients, $D_{p_1 p_2 p_3}$, is obtained in each cell. Equation (11) is strictly enforced by Gaussian elimination and a minimum-error solution to the remaining constraint equations is sought. The resulting coefficient matrix of the linear system depends only on the mesh geometry and can be partially calculated and stored prior to computations [36, 77]. Either a Householder QR factorization algorithm or orthogonal decomposition by the SVD method was used to solve the weighted least-squares problem [78]. Weighting is applied here to each control volume to improve the locality of the reconstruction [79]. An inverse distance weighting formula is applied. For the reconstruction in cell i ,

$$w_j = \frac{1}{|\vec{x}_j - \vec{x}_i|}, \quad (14)$$

where \vec{x}_j is the centroid of the neighbor cell j .

3.1.2 Reconstruction at Boundaries

To enforce conditions at physical boundaries, the least-squares reconstruction was constrained in adjacent control volumes without altering the reconstruction order of accuracy [36, 50]. Constraints are placed on the least-squares reconstruction for each variable to obtain the desired value/gradient (Dirichlet/Neumann) at each Gauss integration point. Here we implement them as Robin boundary conditions

$$f(\vec{x}) = a(\vec{x}) f_D(\vec{x}) + b(\vec{x}) f_N(\vec{x}) \quad (15)$$

where $a(\vec{x})$ and $b(\vec{x})$ define the contribution of the Dirichlet, $f_D(\vec{x})$, and Neumann, $f_N(\vec{x})$, components, respectively. In terms of the cell reconstruction, the Dirichlet condition is simply expressed as

$$f_D(\vec{x}_g) = u^k(\vec{x}_g) \quad (16)$$

where \vec{x}_g is the location of the Gauss quadrature point. The Neumann condition is

$$f_N(\vec{x}_g) = \nabla u^k(\vec{x}_g) \cdot \hat{n}_g = \sum_{p_1+p_2+p_3=1} \sum_{(p_1+p_2+p_3) \leq k} \Delta x^{p_1-1} \Delta y^{p_2-1} \Delta z^{p_3-1} \left[p_1 \Delta y \Delta z n_x + p_2 \Delta x \Delta y n_y + p_3 \Delta x \Delta y n_z \right] D_{p_1 p_2 p_3} \quad (17)$$

where $\Delta(\cdot) = (\cdot)_g - (\cdot)_i$, the subscript i denotes the location of the centroid of the cell adjacent to the boundary and g denotes the Gauss quadrature point.

Exact solutions to the boundary constraints, Eq. (15), are sought. This adds linear equality constraints to the original least-squares problem described in Section 3.1.1. Gaussian elimination with full pivoting is first applied to remove the additional boundary constraints and the remaining least-squares problem is solved as described in Section 3.1.1.

For inflow/outflow or farfield-type boundary conditions where the reconstructed variables are not related, the constraints may be applied separately to each variable. More complex boundary conditions involving linear combinations of solution variables, such as symmetry or inviscid solid walls ($\vec{v} \cdot \hat{n} = 0$), can cause the reconstruction coefficients in Eq. (10) for different variables to become coupled. For these types of coupled boundary conditions, the reconstruction for all of the coupled solution variables is performed together [37, 50]. Thus the final matrix \mathbf{A} for the constrained least-squares reconstruction contains the individual constraints for each variable, the relational constraints, and the approximate mean conservation equations for each variable.

3.1.3 Smoothness Indicator

After performing a k -exact reconstruction for each solution variable in each computational cell, the smoothness indicator is computed for every reconstructed variable to identify under-resolved solution content. It is evaluated as

$$S = \frac{\alpha}{\max[(1-\alpha), \delta]} \frac{(\text{SOS} - \text{DOF})}{\text{DOF} - 1} \quad (18)$$

where α 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 smoothness parameter, α , for a cell i is given by

$$\alpha = 1 - \frac{\sum_p \left[u_p^k(x_p, y_p, z_p) - u_i^k(x_p, y_p, z_p) \right]^2}{\sum_p \left[u_p^k(x_p, y_p, z_p) - \bar{u}_i \right]^2} \quad (19)$$

where u is the solution variable of interest, the subscript p refers to the cells in the reconstruction stencil, $u_p^k(x_p, y_p, z_p)$ is the reconstructed solution in cell p evaluated at the cell's centroid (x_p, y_p, z_p) , $u_i^k(x_p, y_p, z_p)$ is the projected value of the reconstruction polynomial for cell i evaluated at (x_p, y_p, z_p) , and \bar{u}_i is the average value for cell i . By definition, α can have a value between negative infinity and one. A value of unity indicates that the solution is smooth whereas small or negative values indicate large variations in solution content within the reconstruction stencil.

The behavior of the smoothness indicator is demonstrated in Fig. 1. As α approaches unity, the smoothness indicator grows rapidly. Solutions are deemed smooth when the value of S is above critical value, S_c . Previous studies found that values for S_c between 1000–5000 provided an excellent balance between stability and accuracy [49].

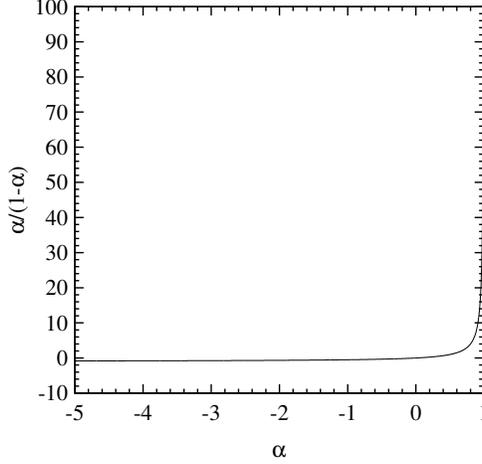


Fig. 1. Variation of smoothness indicator with α .

Typical values for S in smooth regions tend to be orders of magnitude greater than these cutoff limits. Unless otherwise specified, a value for S_c of 1000 was used.

In regions where S is below the critical value, monotonicity is preserved by reverting the high-order k -exact reconstruction to a limited piecewise linear ($k=1$) reconstruction. Limiting is performed using the procedure developed by Park et al. [80] specifically for multiple dimensions in conjunction with the slope limiter function of Venkatakrishnan [81]. Note that the limited linear reconstruction in non-smooth regions is only applied to the inviscid terms in Eq. (9). The viscous flux is still evaluated using the higher-order representation since it is generally the inviscid terms that generate instabilities.

3.2 Numerical Flux

An upwind Godunov scheme is used to integrate the inviscid numerical flux, $\vec{\mathbf{F}}$, over the cell face. Given the left and right solution states, \mathbf{W}_L and \mathbf{W}_R , the numerical flux at the cell interface is defined as

$$\vec{\mathbf{F}} \cdot \hat{n} = \mathcal{F}(\mathbf{W}_L, \mathbf{W}_R, \hat{n}) \quad (20)$$

where \mathcal{F} is a flux function which solves a Riemann problem, \mathcal{R} , in a direction aligned along the face normal, \hat{n} . The left and right solution states at the interface are determined using the k -exact reconstruction procedure described in Section 3.1. As a result, the leading truncation error due to the inviscid operator is $\mathcal{O}(\Delta x^{k+1})$.

The flux function, \mathcal{F} , was derived by applying Roe's approximate Riemann solver [82, 83] to the new modified inviscid eigensystem for Eq. (3). The numerical flux at the interface between two cells is given by

$$\mathbf{F}(\mathcal{R}(\mathbf{W}_L, \mathbf{W}_R)) = \frac{1}{2}(\mathbf{F}_R + \mathbf{F}_L) - \frac{1}{2}|\hat{\mathbf{A}}|\Delta\mathbf{W} \quad (21)$$

where \mathbf{F}_L and \mathbf{F}_R are the inviscid fluxes evaluated based on \mathbf{W}_L and \mathbf{W}_R , $\Delta\mathbf{W} = \mathbf{W}_R - \mathbf{W}_L$, $|\hat{\mathbf{A}}| = \hat{\mathbf{R}}|\hat{\mathbf{A}}|\hat{\mathbf{R}}^{-1}$, $\hat{\mathbf{R}}$ is the matrix of primitive variable right eigenvectors and $\hat{\mathbf{A}}$ is the eigenvalue matrix. The matrix $\hat{\mathbf{A}}$ is the linearized flux Jacobian evaluated at a reference state, $\hat{\mathbf{W}}$. A simple arithmetic average between the left and right states was chosen as the reference state.

The viscous fluxes at each quadrature point are evaluated by averaging the interface state and gradients

$$\mathcal{G}(\mathbf{W}_L, \mathbf{W}_R, \nabla\mathbf{W}_L, \nabla\mathbf{W}_R) = \mathbf{F}_v \left\{ \frac{1}{2}(\mathbf{W}_L + \mathbf{W}_R), \frac{1}{2}(\nabla\mathbf{W}_L, \nabla\mathbf{W}_R) \right\} \quad (22)$$

Because derivatives of the reconstructed polynomial are required, the leading truncation error due to the viscous operator is only $\mathcal{O}(\Delta x^k)$. The degree of the reconstruction polynomial is therefore increased by one to match the

leading truncation error introduced by the inviscid operator. The Gauss quadrature rule is selected to maintain an order of accuracy of $k + 1$ when integrating the fluxes over the cell faces.

For piecewise-linear ($k = 1$) representations, second-order ($k + 1$) accuracy of the viscous operator can be achieved without increasing the degree of the polynomial interpolant. In this case, the average gradient at the interface is evaluated by [84]

$$\nabla \mathbf{W}_{i+1/2} = (\mathbf{W}_n - \mathbf{W}_p) \frac{\hat{n}}{\hat{n} \cdot \vec{r}_{p \rightarrow n}} + \left(\overline{\nabla \mathbf{W}} - \overline{\nabla \mathbf{W}} \cdot \vec{r}_{p \rightarrow n} \frac{\hat{n}}{\hat{n} \cdot \vec{r}_{p \rightarrow n}} \right) \quad (23)$$

where $\overline{\nabla \mathbf{W}}$ is the weighted average of the cell interface

$$\begin{aligned} \overline{\nabla \mathbf{W}} &= \alpha \nabla \mathbf{W}_p + (1 - \alpha) \nabla \mathbf{W}_n \\ \alpha &= V_p / (V_p + V_n) \end{aligned} \quad (24)$$

Equation (23) is second-order accurate if the gradient representation is also second-order accurate. Thus, $k + 1$ reconstruction is not required for $k = 1$.

3.3 Inexact Newton Method for Steady and Unsteady Flows

Integration of the governing equations is performed in parallel to fully take advantage of modern computer architectures. This is carried out by dividing the computational domain up using a parallel graph partitioning algorithm, called Parmetis [85], and distributing the computational cells among the available processors. Solutions for each computational sub-domain are simultaneously computed on each processor. The proposed computational algorithm was implemented using the message passing interface (MPI) library and the Fortran 90 programming language [86]. Ghost cells, which surround an individual local solution domain and overlap cells on neighboring domains, are used to share solution content through inter-block communication.

Newton's method is applied in this work for both steady state relaxation and transient continuation. For transient calculations, a dual-time-stepping approach is used [21, 22, 67–69] with the family of high-order backwards difference formulas to discretize the physical time derivative. In both cases, steady and unsteady, Newton's method is used to relax

$$\mathbf{R}^*(\mathbf{W}) = \mathbf{R} + \frac{d\overline{\mathbf{W}}}{dt} = \mathbf{0} \quad (25)$$

where $\frac{d\overline{\mathbf{W}}}{dt} = \mathbf{0}$ for steady problems.

This particular implementation follows the algorithm developed previously by Groth et al. [87–89] specifically for use on large multi-processor parallel clusters. The implementation makes use of a Jacobian-free inexact Newton method coupled with an iterative Krylov subspace linear solver.

3.3.1 Inexact Newton Method For Steady Problems

For steady problems, a solution to Eq. (25) is sought by iteratively solving a sequence of linear systems given an initial estimate, \mathbf{W}^0 . Successively improved estimates are obtained by solving

$$\left(\frac{\partial \mathbf{R}}{\partial \mathbf{W}} \right)^k \Delta \mathbf{W}^k = \mathbf{J}(\mathbf{W}^k) \Delta \mathbf{W}^k = -\mathbf{R}(\mathbf{W}^k) \quad (26)$$

where $\mathbf{J} = \frac{\partial \mathbf{R}}{\partial \mathbf{W}}$ is the residual Jacobian. The improved solution at step k is then determined from

$$\mathbf{W}^{k+1} = \mathbf{W}^k + \Delta \mathbf{W}^k \quad (27)$$

The Newton iterations proceed until some desired reduction of the norm of the residual is achieved and the condition $\|\mathbf{R}(\mathbf{W}^k)\| < \epsilon \|\mathbf{R}(\mathbf{W}^0)\|$ is met. The tolerance, ϵ , used in this work was typically 10^{-7} for steady problems.

For a system of nonlinear equations, each step of Newton's method requires the solution of the linear problem $\mathbf{J}\mathbf{x} = \mathbf{b}$ where $\mathbf{x} = \Delta \mathbf{W}$ and $\mathbf{b} = -\mathbf{R}(\mathbf{W})$. This system tends to be relatively large, sparse, and non-symmetric for which iterative methods have proven much more effective than direct methods. One effective method for a large variety of problems which is used here is the generalized minimal residual (GMRES) technique of Saad and co-workers [58, 90–92]. This is an Arnoldi-based solution technique which generates orthogonal bases of the Krylov subspace to construct

the solution. The technique is particularly attractive because the matrix \mathbf{J} is not explicitly formed and instead only matrix-vector products are required at each iteration to create new trial vectors. This drastically reduces the required storage. Another advantage is that iterations are terminated based on only a by-product estimate of the residual which does not require explicit construction of the intermediate residual vectors or solutions. Termination also generally only requires solving the linear system to some specified tolerance, $\|\mathbf{R}^k + \mathbf{J}^k \Delta \mathbf{W}^k\|_2 < \zeta \|\mathbf{R}(\mathbf{W}^k)\|_2$, where ζ is typically in the range 0.1–0.5 [56]. We use a restarted version of the GMRES algorithm here, GMRES(m), that minimizes storage by restarting every m iterations.

Right preconditioning \mathbf{J} is performed to help facilitate the solution of the linear system without affecting the solution residual, \mathbf{b} . The preconditioning takes the form

$$(\mathbf{J}\mathbf{M}^{-1})(\mathbf{M}\mathbf{x}) = \mathbf{b} \quad (28)$$

where \mathbf{M} is the preconditioning matrix. A combination of an additive Schwarz global preconditioner and a block incomplete lower-upper (BILU) local preconditioner is used. In additive Schwarz preconditioning, the solution in each block is updated simultaneously and shared boundary data is not updated until a full cycle of updates has been performed on all domains. The preconditioner is defined as follows

$$\mathbf{M}^{-1} = \sum_{k=1}^{N_b} \mathbf{B}_k^T \mathbf{M}_k^{-1} \mathbf{B}_k \quad (29)$$

where N_b is the number of blocks and \mathbf{B}_k is the gather matrix for the k th domain. The local preconditioner, \mathbf{M}_k^{-1} , in Eq. (29) is based on block ILU(p) factorization [92] of the Jacobian for the first order approximation of each domain. The level of fill, p , was maintained at between 0–1 to reduce storage requirements. Larger values of p typically offer improved convergence characteristics for the linear system at the expense of storage. To further reduce computational storage, reverse Cuthill-McKee matrix reordering is used to permute the Jacobian’s sparsity pattern into a band matrix form with a small bandwidth [93].

3.3.2 Implicit-Euler Startup

Newton’s method can fail when initial solution estimates fall outside the radius of convergence. To ensure global convergence of the algorithm, the implicit Euler startup procedure with successive evolution/relaxation (SER) proposed by Mulder and Van Leer [94] was used. Application of this startup procedure to the semi-discrete form of the governing equations gives

$$\left[\frac{\mathbf{\Gamma}}{\Delta\tau^k} + \left(\frac{\partial \mathbf{R}}{\partial \mathbf{W}} \right)^k \right] \Delta \mathbf{W}^k = -\mathbf{R}^k \quad (30)$$

where $\Delta\tau^k$ is the time step. In the SER approach, the time step is varied from some small finite value and gradually increased as the steady state solution is approached. As $\Delta\tau^k \rightarrow \infty$, Newton’s method is recovered.

In the quasi-Newton and SER methods, the time step size was determined by considering the inviscid Courant-Friedrichs-Lewy (CFL) and viscous Von Neumann stability criteria based on the pseudo-compressible system. The maximum permissible time step for each local cell is determined by

$$\Delta\tau^k \leq \text{CFL} \cdot \min \left[\frac{\Delta x}{\lambda^+}, \frac{\rho \Delta x^2}{\mu} \right] \quad (31)$$

where CFL is a constant greater than zero which determines the time step size and $\Delta x = V^{1/3}$ is a measure of the grid size. Using SER, the CFL number for the k th iteration is computed using the following relation:

$$\text{CFL}^k = \text{CFL}_0 \frac{\|\mathbf{R}(\mathbf{W}^0)\|}{\|\mathbf{R}(\mathbf{W}^k)\|} \quad (32)$$

During the startup phase of the Newton calculation, a value for CFL between 10–100 is typically used. The minimum value of β was chosen based on the following formulation proposed by Turkel [18]:

$$\beta = \max \left[2(u^2 + v^2 + w^2), \varepsilon \right] \quad (33)$$

where ε is a smallness parameter.

Table 2. Coefficients for the BDF methods.

$$\alpha^{n+1}u^{n+1} + \alpha^n u^n + \dots = \Delta t f^{n+1}$$

Order	u^{n+1}	u^n	u^{n-1}	u^{n-2}	u^{n-3}
1	1	-1			
2	$\frac{2}{3}$	-2	$\frac{1}{2}$		
3	$\frac{11}{6}$	-3	$\frac{3}{2}$	$-\frac{1}{3}$	
4	$\frac{25}{12}$	-4	3	$\frac{4}{3}$	$\frac{1}{4}$

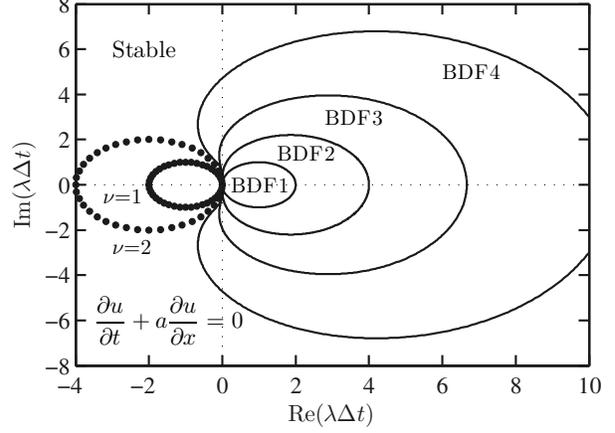


Fig. 2. Stability diagram for the BDF methods applied to the linear advection equation. Lines enclose the unstable regions for each method; symbols are the eigenvalues for a first-order upwind spatial discretization with uniform mesh spacing Δx and periodic boundaries. λ represents the eigenvalues, Δt is the time step size, a is the wave speed, and $\nu = a\Delta t/\Delta x$ is the CFL number.

3.3.3 Dual-Time Stepping Approach for Unsteady Problems

Applying the BDF temporal discretization and Newton's method to the semi-discrete form of the governing equations, Eq. (25), gives

$$\left[\frac{\alpha}{\Delta t^n} \left(\frac{\partial \mathbf{U}}{\partial \mathbf{W}} \right)^{(n+1,k)} + \frac{\mathbf{\Gamma}}{\Delta \tau^k} + \left(\frac{\partial \mathbf{R}}{\partial \mathbf{W}} \right)^{(n+1,k)} \right] \Delta \mathbf{W}^{(n+1,k)} = -\mathbf{R}^{(n+1,k)} - \frac{d\bar{\mathbf{U}}}{dt} \Big|^{(n+1,k)} = -\mathbf{R}^{*(n+1,k)} \quad (34)$$

where n is the outer time level, k is the now the inner iteration level, and α is a constant which depends on the temporal discretization ($\alpha = 1$ for implicit Euler, $3/2$ for BDF2, $11/6$ for BDF3 and $25/12$ for BDF4).

In the dual time-stepping procedure, at each physical time step n , a steady problem is solved using the Newton procedure described in Section 3.3.1. The stability of the unsteady system is now governed by the physical time step size, which is determined based on the CFL criterion:

$$\Delta t^n \leq \text{CFL}_{\text{phys}} \cdot \min \left(\frac{\Delta x}{u^2 + v^2 + w^2} \right) \quad (35)$$

where the CFL_{phys} is the physical CFL number. The convergence tolerance for the inner iteration loop used here was $\epsilon = 10^{-4}$. A tolerance of 10^{-4} was shown to provide a good balance between accuracy and computation time by Tabesh and Zingg [95].

Coefficients for the different BDF schemes are provided in Table 2. The order of accuracy of the temporal discretization scheme was chosen to match the accuracy of the spatial discretization. Note however, it is well known that BDF methods with accuracy higher than second-order can be unstable when the real component of the eigenvalues of the system are negative. A stability analysis of the BDF methods applied to the one-dimensional form of the linear advection equation is shown in Fig. 2. The eigenvalues, λ , for a first-order upwind discretization applied to this system with uniform mesh spacing Δx and periodic boundary conditions are also shown in Fig. 2. BDF4 is stable for a CFL, $\nu = a\Delta t/\Delta x$, up to 2. This condition may be relaxed for systems involving diffusion and relaxation processes since they tend to shift the eigenvalues further into the negative portion of the real $\lambda\Delta t$ -plane. As such, the BDF methods are still suitable for the present application since Eq. (4) involves a system of advection-diffusion equations. No issues related to the stability of the BDF methods were encountered throughout this research.

4 Results For Three-Dimensional Unstructured Mesh

The proposed finite-volume scheme was assessed in terms of accuracy, stability, and computational efficiency. Numerical results for both smooth and discontinuous function reconstructions as well as solutions for steady and unsteady viscous flows on three-dimensional unstructured mesh were obtained. All computations were performed on a high performance parallel cluster consisting of 3,780 Intel Xeon E5540 (2.53GHz) nodes with 16GB RAM per node. The cluster is connected with a high speed InfiniBand switched fabric communications link.

An error analysis is performed whenever exact solutions are present. Accuracy is assessed based on the L_1 and L_2 norms of the error between the exact solution and the numerical solution. The L_p norm of the error is evaluated over all cells, i ,

$$L_p = \|\text{Error}\|_p = \left[\frac{1}{\mathcal{V}_T} \sum_i \iiint_{V_i} |u_i^k(x, y, z) - u_{\text{exact}}(x, y, z)|^p dV \right]^{1/p} \quad (36)$$

where \mathcal{V}_T is the total volume of the domain and $u_{\text{exact}}(x, y, z)$ is the exact solution. This integration is performed using an adaptive cubature algorithm developed by Cools et al. [96, 97] for integrating functions over a collection of N -dimensional hyperrectangles and simplices.

4.1 Spherical Cosine Function

The first case considered is the reconstruction of a smooth spherical cosine function. The function, which is smooth in all directions, is described by

$$u(r) = 1 + \frac{1}{3} \cos(r) \quad (37)$$

where $r = 10\sqrt{x^2 + y^2 + z^2}$ is the radial position. The solution is computed on a unit cube using grids composed of tetrahedral, Cartesian, and irregular hexahedral cells with varying levels of resolution. The irregular hexahedral meshes were generated by randomly perturbing the internal nodes of an initial Cartesian mesh.

The results for the unlimited k -exact reconstruction of the three-dimensional spherical cosine function performed on a coarse mesh with 3,072 tetrahedral elements is illustrated in Fig. 3(a). As the order of the piecewise polynomial interpolant is increased from $k=0$ to $k=3$, the reconstructed solution rapidly approaches the exact solution. There is almost no visible difference between the exact solution and the reconstructed solution for $k=4$ (not shown in figure).

An analysis of the L_2 norm of the error in the numerical solution as the tetrahedral mesh resolution is increased, illustrated in Fig. 3(b) for various values of k , confirms that k -exact reconstruction of a smooth function yields an order of accuracy equal to $k+1$. Similar results for the error analysis are observed for meshes composed of Cartesian, Fig. 3(c), and irregular hexahedrals, Fig. 3(d).

4.2 Abgrall's Function

The Abgrall function [29] possesses a number of solution discontinuities which test a high-order spatial discretization's ability to maintain monotonicity. Reconstructions of this function using the proposed high-order CENO algorithm for unstructured meshes are obtained to ensure the effectiveness of the smoothness indicator defined in Eq. (18). Even though the performance of this formulation for S was already verified using the Abgrall function and structured mesh [49], it has not been fully evaluated for unstructured mesh. McDonald et al. [51] only obtained preliminary results for this function on tetrahedral meshes. The Abgrall function is defined as

$$u(x, y) = \begin{cases} f(x - \cot \sqrt{\pi/2} y) & \text{if } x \leq \cos(\pi y)/2, \text{ and} \\ f(x + \cot \sqrt{\pi/2} y) + \cos(2\pi y) & \text{if } x > \cos(\pi y)/2. \end{cases} \quad (38)$$

where

$$f(r) = \begin{cases} -r \sin(3\pi r^2/2) & \text{if } r \leq -1/3, \\ |\sin(2\pi r)| & \text{if } |r| < 1/3, \text{ and} \\ 2r - 1 + \sin(3\pi r)/6 & \text{if } r \geq 1/3. \end{cases} \quad (39)$$

and $r = \sqrt{x^2 + y^2}$. Here, it is applied in three dimensions to a cube with length 2 by extruding the two-dimensional function along the z axis. The reconstructed solution obtained using the proposed high-order CENO algorithm for

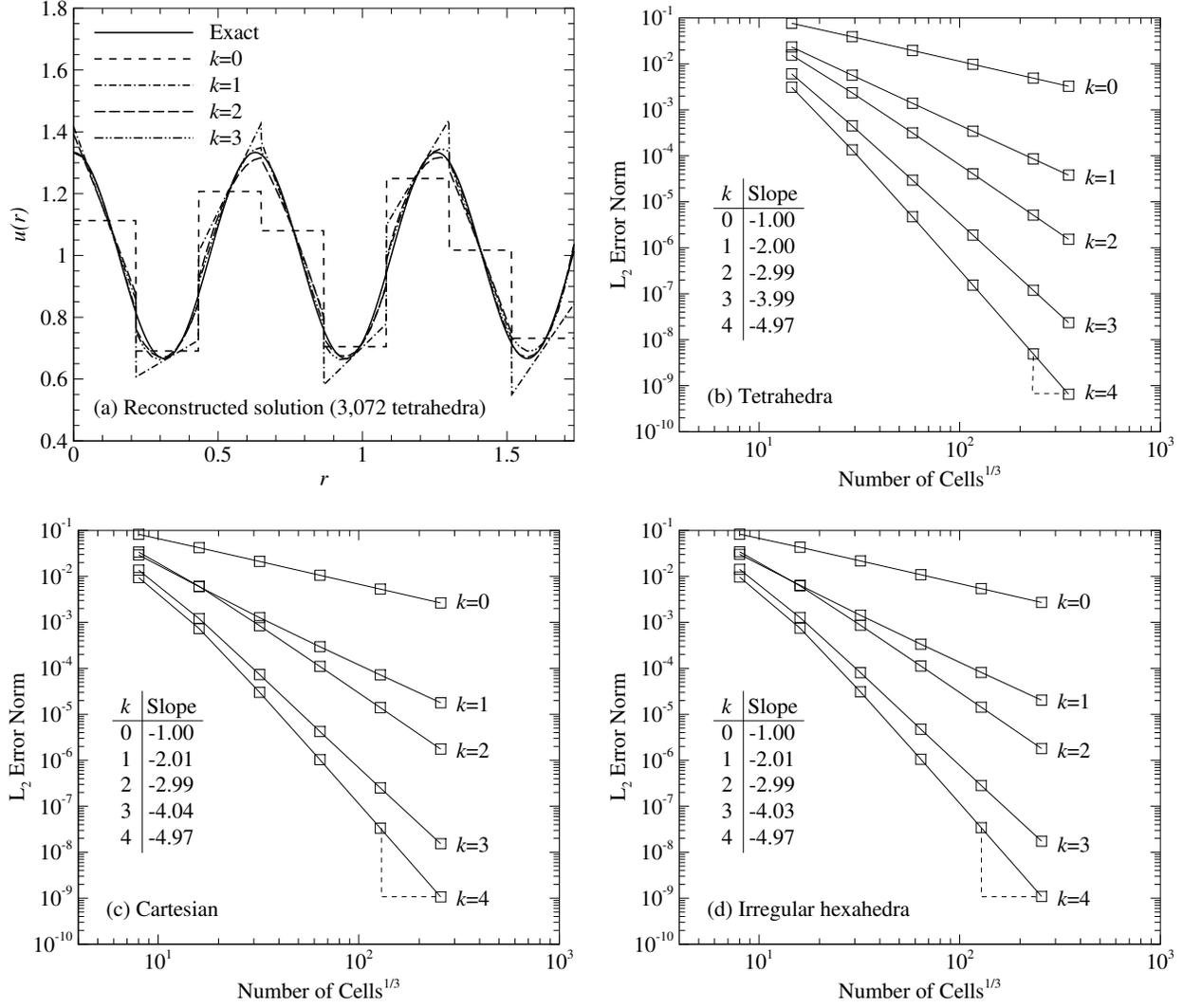


Fig. 3. Results for k -exact reconstruction of the spherical cosine function. (a) Unlimited reconstructed solution along the diagonal from $(0, 0, 0)$ to $(1, 1, 1)$ obtained on a mesh with 3,072 tetrahedral cells; L_2 norm of reconstruction error for (b) tetrahedral, (c) Cartesian, and (d) irregular hexahedral shaped cells.

unstructured meshes with $k = 4$ is compared with the exact solution in Figs. 4(a) and 4(b). The CENO scheme is able to accurately reconstruct the Abgrall function even on a relatively coarse mesh of only 196,608 tetrahedral cells without producing spurious oscillations. For this mesh, there is only approximately 58 cells in each direction with an average edge length of 0.03. The smoothness indicator, illustrated in Fig. 4(c), correctly identified the discontinuities in both $f(r)$ and $\partial f(r)/\partial x$.

The reconstructed solution obtained with $k=0$ to $k=4$ and 196,608 tetrahedral cells along a line is illustrated in Fig. 4(d). The proposed CENO scheme is able to ensure oscillation-free solutions despite the large discontinuities observed. The spikes observed in the numerical solution occur when the line intersects the faces between cells at acute angles.

The effect of mesh resolution on the L_1 norm of the solution error is shown in Fig. 5 for tetrahedral and irregular hexahedral cells. A large improvement in the error is achieved by increasing k from 0 to 1. This improvement becomes less pronounced as k is increased further to 4 since a large portion of the domain possesses discontinuous features. An order of accuracy of 1 is expected for all values of k due to discontinuities in the function and the use of a limited piecewise linear reconstruction in these regions. Nonetheless, the hybrid reconstruction procedure maintains

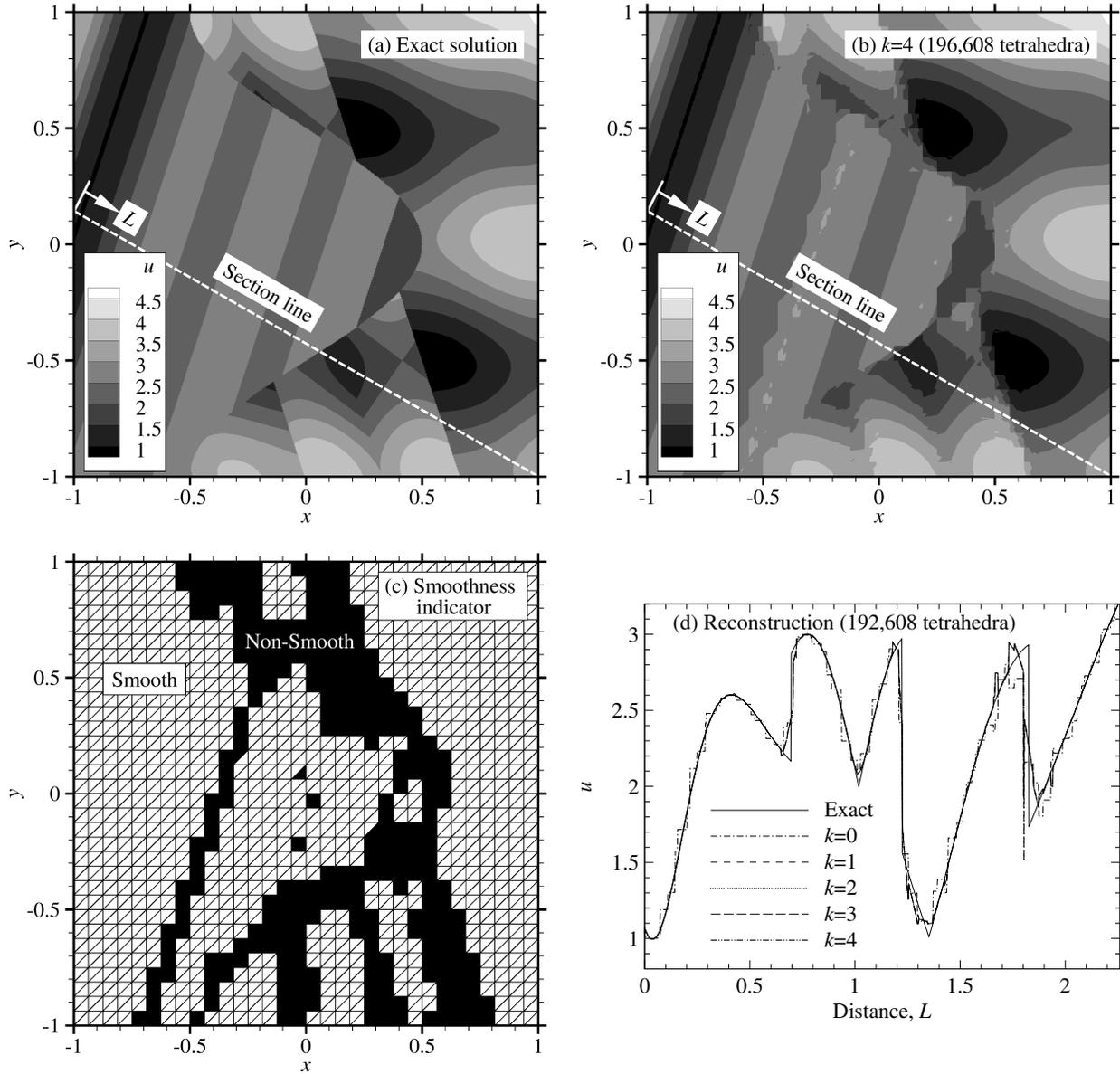


Fig. 4. Results for CENO reconstruction of Abgrall’s function obtained using a three-dimensional mesh with 196,608 tetrahedral cells. (a) Exact solution; (b) numerical solution and (c) computed smoothness indicator for $k=4$; (d) reconstructed solution along the line identified in (a).

the monotonicity of the higher-order solutions. This highlights the robustness of the proposed scheme.

4.3 Isothermal Flow Over a Flat Plate

Numerical results for the steady laminar flow over a flat plate were obtained using the second-order CENO finite volume scheme and compared with the Blasius analytic solution. The computational domain and boundary conditions are illustrated in Fig. 6(a). A rectangular domain was used with dimensions $4L \times 2L \times L/4$. The far-field and upstream of the plate are modeled by a reflection/slip condition while the presence of the plate is simulated using a no-slip condition. A uniform velocity profile is specified at the inlet and a zero-gradient condition on the fluid velocity is applied at the outlet. Pressure is held fixed and equal to 0 at the outlet. The free-stream Mach number and Reynolds

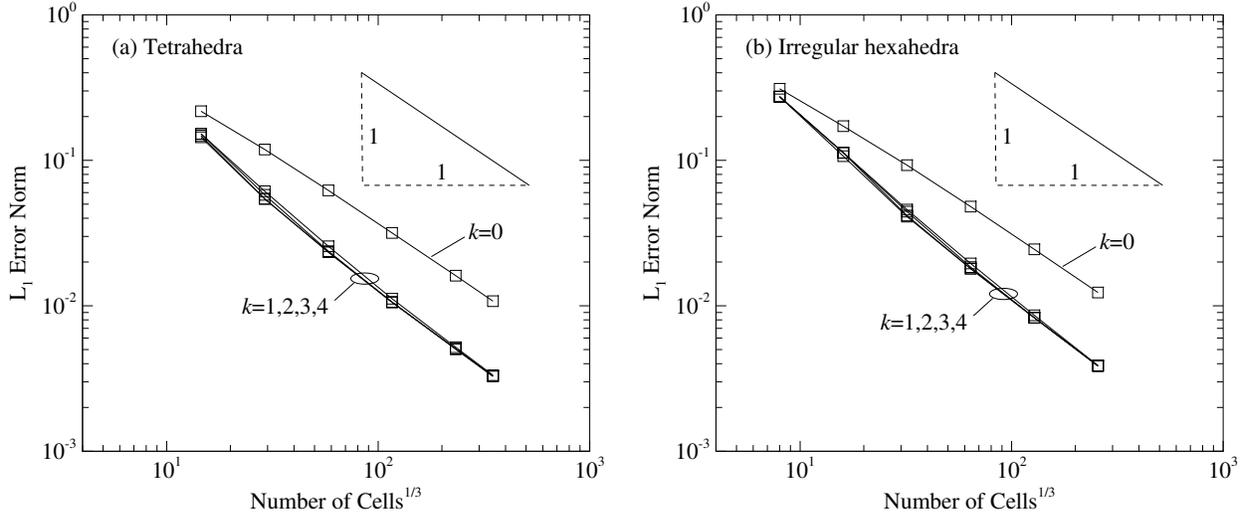


Fig. 5. Effect of grid resolution on solution accuracy for CENO reconstruction of the Abgrall function. Accuracy is measured using the L_1 norm of the error.

number based on the plate length, L , are $M_\infty = 0.2$ and $Re_L = 10,000$, respectively. Solutions were obtained on a stretched mesh with 512,000 Cartesian cells ($160 \times 160 \times 20$) clustered near the leading edge of the plate and near the lower boundary. The CFL was initially equal to 100 for start-up and then increased to 400 using the SER procedure outlined in Section 3.3.2.

The predicted non-dimensionalized boundary layer profile at $Re_x = 8,000$ is compared with the Blasius analytic solution in Fig. 6(c). The non-dimensional distance from the plate, η , is defined as $y\sqrt{Re_x}/x$. Excellent agreement is obtained with the analytic solution.

The convergence history obtained using the second-order CENO scheme is provided in Fig. 6(b). Excellent convergence characteristics are displayed by the parallel Newton-Krylov-Schwarz solution algorithm, requiring only 2500 evaluations of Eq. (25) to reduce the equation residuals by approximately eight orders of magnitude. Note that a non-zero residual is observed for the enthalpy equation even though the flow is isothermal. This is a result of introducing a pseudo-time derivative with respect to pressure to Eq. (1c).

4.4 Taylor-Green Vortex Decay

The Taylor-Green vortex solution [98] is used for testing and validation of the spatial and temporal accuracy of the proposed solution algorithm for the incompressible Navier-Stokes equations. The Taylor-Green vortex solution is given by

$$u = \sin x \cos y \exp^{-2\nu t} \quad (40)$$

$$v = -\cos x \sin y \exp^{-2\nu t} \quad (41)$$

$$p = \frac{\rho}{4} (\cos 2x + \cos 2y) \exp^{-4\nu t} \quad (42)$$

over the domain $0 \leq x, y \leq 2\pi$ where ν is the kinematic viscosity of the fluid. Although the solution is two-dimensional, it was extended to three dimensions by extrapolating the solution in the z -direction and setting $w=0$. Solutions were obtained on a periodic cube domain with length 2π m using varying grid resolutions with either regular hexahedral (Cartesian) or tetrahedral cells and a time interval of $\Delta t=0.05$ s. This time step size corresponds to a CFL_{phys} of 0.1 on the coarsest mesh and 2.0 on the finest mesh. The kinematic viscosity, $\nu = \mu/\rho$, is set to 0.01 m²/s in this case which corresponds to a Reynolds number of 100.

The initial conditions for u at time $t=0$ is illustrated in Fig. 7(a). As time progresses, the magnitude of the velocity decays exponentially through the effects of viscous damping. This decay is apparent in Fig. 7(b) which shows the exact solution along $y=z=\pi$ m at $t=25$ s. The predicted solutions obtained using the CENO finite-volume scheme with

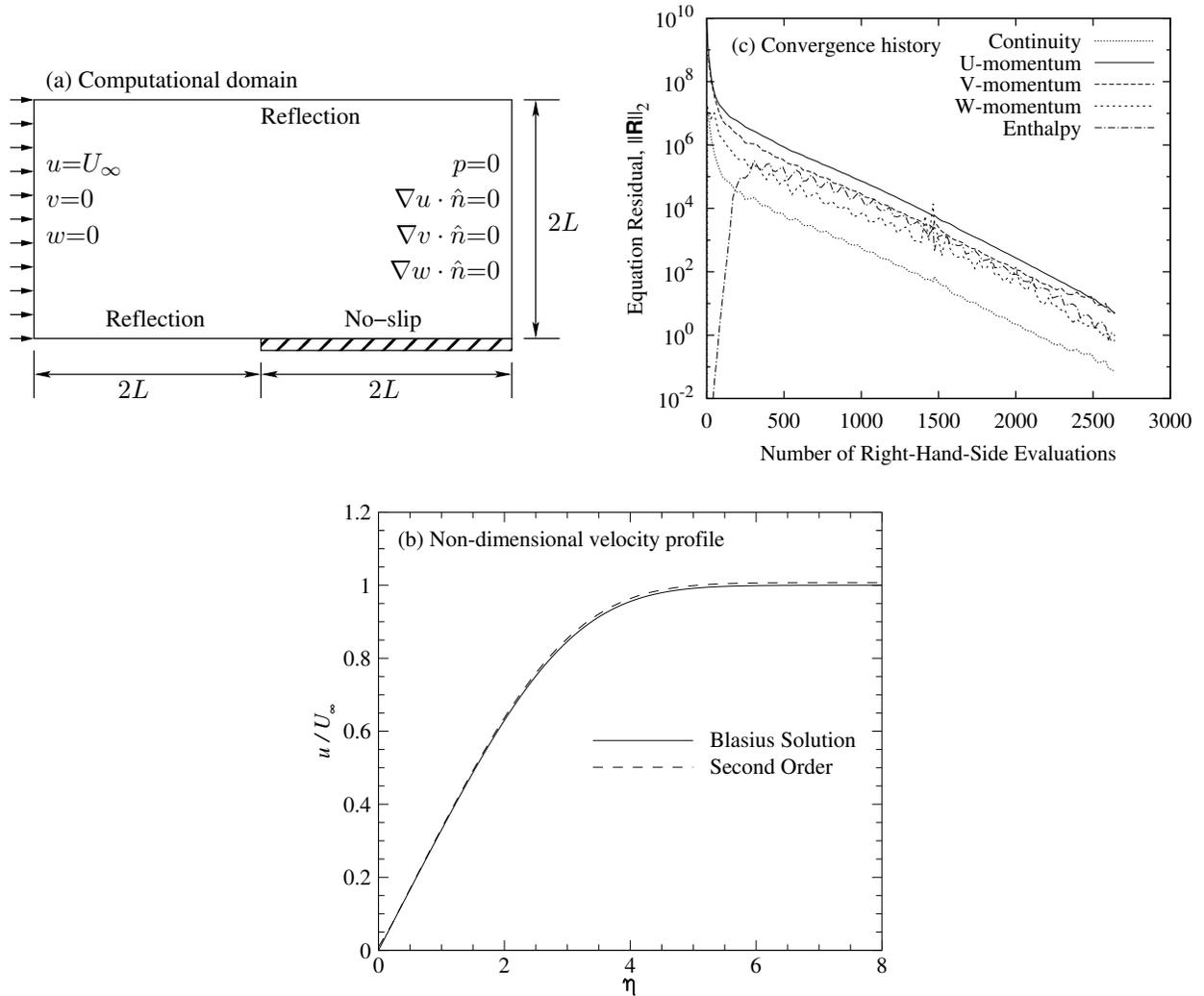


Fig. 6. Numerical results for laminar flow over a flat plate. (a) Computational domain and boundary conditions; (b) convergence history for the second-order accurate solution; and (c) Blasius and numerical solutions for the non-dimensionalized boundary layer profile at $Re_x=8,000$.

different orders of accuracy on a mesh with 64^3 hexahedra are also provided in Fig. 7(b). The first-order solution, which uses a piecewise-constant representation to evaluate the inviscid fluxes and a piecewise-linear representation for the viscous fluxes, greatly over-predicts the exponential decay of the velocity field because of the scheme's dissipative nature. A significant improvement in accuracy is observed when the second-order scheme is used. However, the second-order scheme still cannot properly represent the sinusoidal form of the solution using a piecewise-linear representation. Excellent agreement is obtained between the predicted solutions obtained using the higher-order schemes (third- and fourth-order) and the exact solution.

An error analysis was performed to verify the correct spatial order-of-accuracy and is shown in Fig. 8 for both Cartesian, Fig. 8(a), and tetrahedral, Fig. 8(b), meshes. For both types of mesh, the first-order scheme yields an order-of-accuracy of 0.6, which is less than the expected value of 1. However, it is apparent in Fig. 8 that the finest meshes used in both of these studies has not yet reached the asymptotic regime since the slopes of their corresponding lines is still decreasing. The second-order scheme provides an order-of-accuracy equal to approximately 3 on both Cartesian and tetrahedral meshes. This observed increased slope is also observed for the meshes composed of tetrahedra

When the third- and fourth-order schemes are applied to uniform Cartesian mesh, a similar trend regarding the magnitude of the slopes of the lines corresponding to error vs mesh-size is observed in Fig. 8(a). The fourth-order

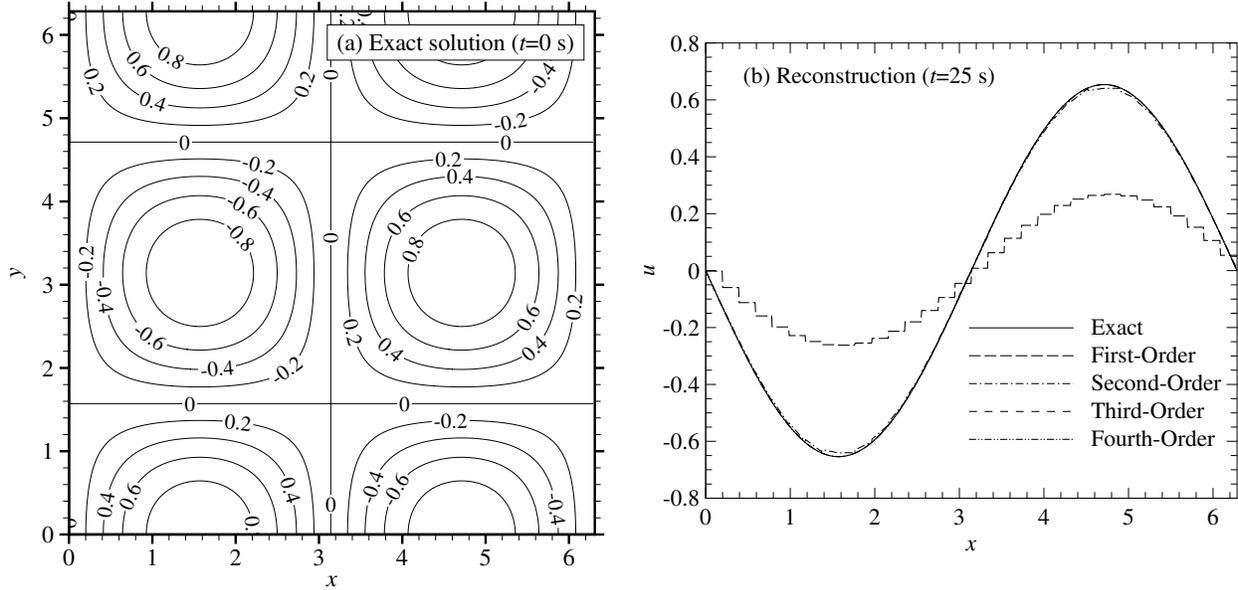


Fig. 7. Results for CENO reconstruction of the Taylor-Green vortex decay. (a) Exact solution for u at $t=0$ s and (b) reconstructed solution for u along $y=z=\pi$ m at $t=25$ s obtained on a mesh composed of 64^3 uniform Cartesian cells.

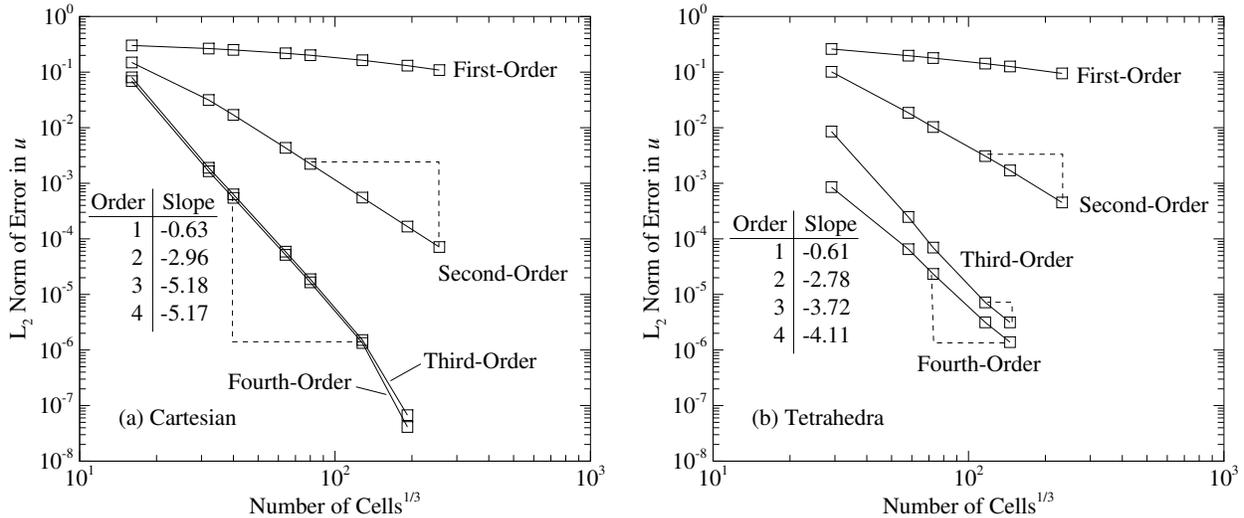


Fig. 8. Effect of mesh size on the L_2 -norm of the error in u for the Taylor-Green vortex decay.

scheme achieves an extra order-of-accuracy while the third-order scheme achieves two extra orders. However, both schemes are expected to yield the same order of accuracy as a result of the cancellation of truncation errors on uniform Cartesian mesh. This suggests that the asymptotic regime has not yet been reached and may explain the unexpected change in slope when the number of computational cells is increased from 128^3 to 192^2 .

With the tetrahedra meshes, Fig. 8(b), fourth-order accuracy is observed for both the third- and fourth-order schemes. Since no cancellation of truncation errors is expected for tetrahedral mesh, this suggests that the largest meshes used in this study may not be sufficiently fine to reach the asymptotic regime. Further study is warranted to clarify these findings. It is certainly clear that the observed convergence rates for all orders either matches or exceeds the theoretical expectations for this unsteady problem.

The computational efficiency is assessed in Fig. 9 which illustrates the error as a function of CPU time for each

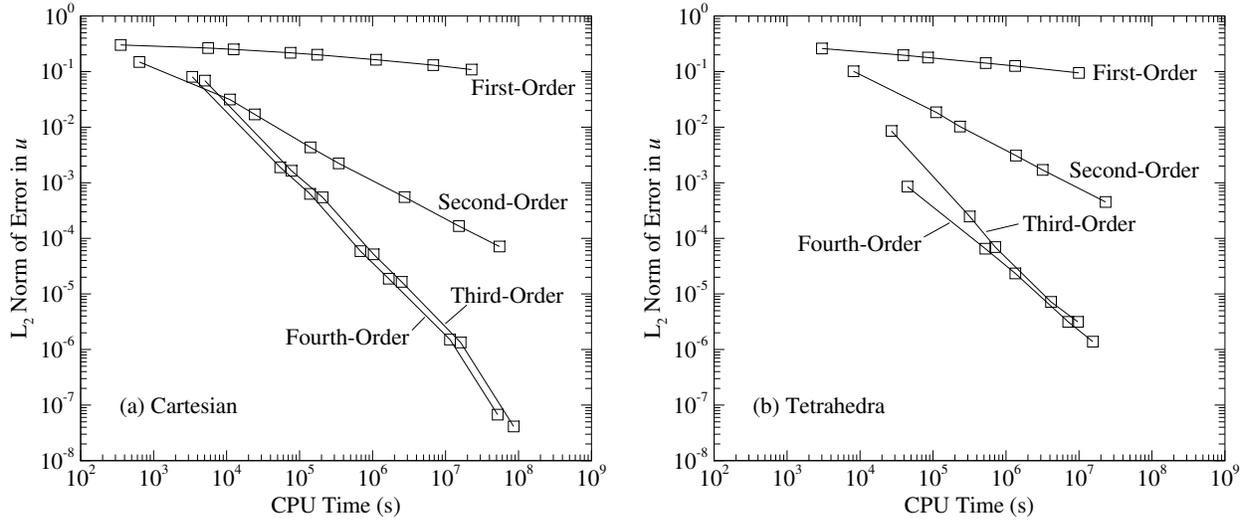


Fig. 9. Required CPU time for a desired level of accuracy for the Taylor-Green vortex decay.

scheme. As expected, lower levels of error have a higher associated computational costs for all the schemes tested here. If the desired level of error is large, around 10^{-1} and higher, the first- and second-order solution methods offer the lowest computational cost. However, the third- and fourth-order schemes become the most cost effective in terms of accuracy vs computational time when higher levels of accuracy are desired. For example, for an L_2 error of approximately 10^{-4} on a uniform Cartesian mesh, the third- and fourth-order schemes are 86 and 62 times faster than the second-order scheme, respectively. This level of error is achieved by both the third- and fourth-order schemes using a mesh 64 times smaller than the mesh required by the second-order scheme. It is apparent that for unstructured meshes of both Cartesian and tetrahedral computational cells, the higher-order schemes (third-order accuracy and higher) offer a superior advantage over the low-order ones in terms of accuracy and computational cost.

5 Conclusions

A high-order finite-volume solver has been developed for the numerical solution of the incompressible Navier-Stokes equations on unstructured meshes using the pseudo-compressibility approach. The scheme is a high-order variant of cell-centered, Godunov-type, finite-volume, methods and uses the hybrid CENO reconstruction method recently proposed by Ivan and Groth [49, 50].

An initial validation of the CENO reconstruction procedure and finite-volume scheme has been performed for a variety of functions and idealized flow problems. For smooth functions, up to forth-order reconstruction accuracy was achieved on general unstructured hexahedral and tetrahedral mesh. Robust and monotone reconstructions were maintained throughout this study even for functions with discontinuities. This demonstrates the effectiveness of the smoothness indicator, which lowers the order of accuracy of the scheme near areas of discontinuities or under-resolved solution content in order to maintain monotonicity.

Further validation of predicted solutions to the three-dimensional Navier-Stokes equations was performed. Numerical results obtained for the unsteady decay of Taylor-Green vortices confirmed that high-order solutions for smooth viscous incompressible fluids are achieved with the proposed finite-volume scheme, even on unstructured mesh. The higher-order schemes also displayed excellent computational efficiency in terms of both accuracy and CPU time. For an L_2 error of approximately 10^{-4} , the third- and fourth-order schemes are 86 and 62 times faster than the second-order scheme, respectively. This level of error was achieved by both schemes using a mesh 64 times smaller than the mesh required by the second-order scheme.

In general, the proposed scheme is able to accurately represent solutions with smooth extrema while robustly handling under-resolved and/or non-smooth solution content. Combined with the parallel Newton-Krylov-Schwarz solution algorithm, which displayed excellent convergence characteristics, the proposed finite-volume scheme is able to obtain fast and accurate solutions to the incompressible Navier-Stokes equations. Future work consists of further devel-

opment and validation of the proposed Newton-Krylov-Schwarz CENO algorithm for three-dimensional unstructured meshes. This includes applying the pseudo-compressibility approach to more complex flows, such as the large-eddy simulation of turbulent flames, and incorporating a multi-block adaptive mesh refinement (AMR) algorithm [99–102]. The applicability of CENO to AMR and the substantial benefits in terms of accuracy and computational savings have already been demonstrated for body-fitted multi-block meshes [49, 50].

Acknowledgments

Financial support for the research described herein was provided by MITACS (Mathematics of Information Technology and Complex Systems) Network, part of the Networks of Centres of Excellence (NCE) program funded by the Canadian government, as well as by Rolls-Royce Canada Inc. This funding is gratefully acknowledged with many thanks. Computational resources for performing all of the calculations reported herein were provided by the SciNet High Performance Computing Consortium at the University of Toronto and Compute/Calcul Canada through funding from the Canada Foundation for Innovation (CFI) and the Province of Ontario, Canada.

References

- [1] D. Drikakis and W. Rider. *High-resolution methods for incompressible and low-speed flows*. Springer-Verlag, Berlin, Heidelberg, 2005.
- [2] D. Kwak, C. Kiris, and J. Housman. *Comput. Fluids*, 41(1):51–64, 2011.
- [3] S.V. Patankar and D.B. Spalding. *Int. J. Heat Mass Transfer*, 15(10):1787–1806, 1972.
- [4] C.M. Rhie and W.L. Chow. *AIAA J.*, 21(11):1525–1532, 1983.
- [5] N.N. Yanenko. *The Method of Fractional Steps*. Springer-Verlag, Berlin, 1971.
- [6] J. Kim and P. Moin. *J. Comput. Phys.*, 59(2):308–323, 1985.
- [7] H. Fasel. *J. Fluid Mech.*, 78(2):355–383, 1976.
- [8] S.C.R. Dennis, D.B. Ingham, and R.N. Cook. *J. Comput. Phys.*, 33(3):325–339, 1979.
- [9] A.J. Chorin. *J. Comput. Phys.*, 2(1):12–26, 1967.
- [10] B. Van Leer, W.T. Lee, and P.L. Roe. AIAA Paper 91-1552, 1991.
- [11] Y.H. Choi and C.L. Merkle. *J. Comput. Phys.*, 105(2):207–223, 1993.
- [12] E. Turkel. *Appl. Numer. Math.*, 12(1-3):257–284, 1993.
- [13] J.M. Weiss and W.A. Smith. *AIAA J.*, 33(11):2050–2057, 1995.
- [14] E. Turkel, R. Radespiel, and N. Kroll. *Comput. Fluids*, 26(6):613–634, 1997.
- [15] S. Venkateswaran, M. Deshpande, and C.L. Merkle. The application of preconditioning to reacting flow computations. In *12th AIAA Computational Fluid Dynamics Conference*, San Diego, CA, Jun. 19-22 1995. AIAA paper 1995-1673.
- [16] J.L. Steger and P. Kutler. *AIAA J.*, 15(4):581–590, 1977.
- [17] D. Kwak, J.L.C. Chang, S.P. Shanks, and S.R. Chakravarthy. *AIAA J.*, 24(3):390–396, 1986.
- [18] E. Turkel. *J. Comput. Phys.*, 72(2):277–298, 1987.
- [19] S.E. Rogers, D. Kwak, and U. Kaul. *Appl. Math. Model.*, 11(1):35–44, 1987.
- [20] C. Kiris, D. Kwak, S. Rogers, and I.D. Chang. *J. Biomed. Eng.*, 119(4):452–460, 1997.

- [21] S.E. Rogers and D. Kwak. *AIAA J.*, 28(2):253–262, 1990.
- [22] S.E. Rogers, D. Kwak, and C. Kiris. *AIAA J.*, 29(4):603–610, 1991.
- [23] Z. Qian and J. Zhang. *Int. J. Numer. Meth. Fluids*, 69(7):1165–1185, 2012.
- [24] J.L.C. Chang and D. Kwak. AIAA Paper 84-0252, 1984.
- [25] S. Pirozzoli. *J. Comput. Phys.*, 219(2):489–497, 2006.
- [26] A. Harten, B. Engquist, S. Osher, and S.R. Chakravarthy. *J. Comput. Phys.*, 71(2):231–303, 1987.
- [27] W.J. Coirier and K.G. Powell. *AIAA J.*, 34(5):938–945, May 1996.
- [28] T.J. Barth. AIAA Paper 93-0668, 1993.
- [29] R. Abgrall. *J. Comput. Phys.*, 114:45–58, 1994.
- [30] T. Sonar. *Comp. Meth. Appl. Mech. Eng.*, pages 140–157, 1997.
- [31] C.F. Ollivier-Gooch. *J. Comput. Phys.*, 133:6–17, 1997.
- [32] G.S. Jiang and C.W. Shu. *J. Comput. Phys.*, 126(1):202–228, 1996.
- [33] D. Stanescu and W. Habashi. *AIAA J.*, 36:1413–1416, 1998.
- [34] O. Friedrich. *J. Comput. Phys.*, 144:194–212, 1998.
- [35] C. Hu and C.W. Shu. *J. Comput. Phys.*, 150:97–127, 1999.
- [36] C.F. Ollivier-Gooch and M. Van Altena. *J. Comput. Phys.*, 181(2):729–752, 2002.
- [37] A. Nejat and C. Ollivier-Gooch. *J. Comput. Phys.*, 227(4):2582–2609, 2008.
- [38] B. Cockburn and C.W. Shu. *Math. Comp.*, 52:411, 1989.
- [39] B. Cockburn, S. Hou, and C.W. Shu. *J. Comput. Phys.*, 54:545, 1990.
- [40] R. Hartmann and P. Houston. *J. Comput. Phys.*, 183:508–532, 2002.
- [41] H. Luo, J.D. Baum, and R. Löhner. *J. Comput. Phys.*, 225:686–713, 2007.
- [42] G. Gassner, F. Lörcher, and C.D. Munz. *J. Comput. Phys.*, 224(2):1049–1063, 2007.
- [43] Z.J. Wang. *J. Comput. Phys.*, 178:210–251, 2002.
- [44] Z.J. Wang and Y. Liu. *J. Comput. Phys.*, 179:665–697, 2002.
- [45] Z.J. Wang, L. Zhang, and Y. Liu. High-order spectral volume method for 2d euler equations. Paper 2003–3534, AIAA, June 2003.
- [46] Z.J. Wang and Y. Liu. *Journal of Scientific Computing*, 20(1):137–157, 2004.
- [47] Y. Sun, Z.J. Wang, and Y. Liu. *J. Comput. Phys.*, 215(1):41–58, 2006.
- [48] A. Haselbacher. AIAA paper 2005-0879, 2005.
- [49] L. Ivan and C.P.T. Groth. AIAA paper 2007-4323, 2007.
- [50] L. Ivan and C.P.T. Groth. AIAA paper 2011-0367, 2011.
- [51] S.D. McDonald, M.R.J. Charest, and C.P.T. Groth. High-order CENO finite-volume schemes for multi-block unstructured mesh. In *20th AIAA Computational Fluid Dynamics Conference*, Honolulu, Hawaii, June 27–30 2011. AIAA-2011-3854.

- [52] S.E. Rogers and D. Kwak. *Appl. Numer. Math.*, 8(1):43–64, 1991.
- [53] Y.N. Chen, S.C. Yang, and J.Y. Yang. *Int. J. Numer. Meth. Fluids*, 31(4):747–765, 1999.
- [54] S.E. Rogers, J.L.C. Chang, and D. Kwak. *J. Comput. Phys.*, 73(2):364–379, 1987.
- [55] S. Yoon and D. Kwak. AIAA Paper 89-1964, 1989.
- [56] R.S. Dembo, S.C. Eisenstat, and T. Steihaug. *SIAM J. Numer. Anal.*, 19(2):400–408, 1982.
- [57] T.F. Chan and K.R. Jackson. *SIAM J. Sci. Stat. Comput.*, 5(3):533–542, 1984.
- [58] P.N. Brown and Y. Saad. *SIAM J. Sci. Stat. Comput.*, 11(3):450–481, 1990.
- [59] D.A. Knoll and D.E. Keyes. *J. Comput. Phys.*, 193(2):357–397, 2004.
- [60] D.A. Knoll, V.A. Mousseau, L. Chacón, and J. Reisner. *J. Comput. Phys.*, 25(1):213–230, 2005.
- [61] D. Pan and C. Chang. *Int. J. Numer. Meth. Fluids*, 33(2):203–222, 2000.
- [62] L. Qian, D.M. Causon, D.M. Ingram, and C.G. Mingham. *J. Hydraul. Eng.*, 129(9):688–696, 2003.
- [63] U. Riedel. *Combust. Sci. Tech.*, 135(1-6):99–116, 1998.
- [64] E. Shapiro and D. Drikakis. *J. Comput. Phys.*, 210(2):584–607, 2005.
- [65] M. Azhdarzadeh and S.E. Razavi. *J. Appl. Sci.*, 8(18):3183–3190, 2008.
- [66] S.L. Chang and K.T. Rhee. *Int. Commun. Heat Mass Transfer*, 11(5):451–455, 1984.
- [67] A. Jameson. AIAA paper 1991-1596, 1991.
- [68] C.L. Merkle and M. Athavale. AIAA paper 87-1137, 1987.
- [69] W.Y. Soh and J.W. Goodrich. *J. Comput. Phys.*, 79(1):113–134, 1988.
- [70] Z. Qian, J. Zhang, and C. Li. *Science China: Physics, Mechanics and Astronomy*, 53(11):2090–2102, 2010.
- [71] H. Lee and S. Lee. *Int. J. Aeronaut. Space Sci.*, 12(4):318–330, 2011.
- [72] A.G. Malan, R.W. Lewis, and P. Nithiarasu. *Int. J. Numer. Meth. Engin.*, 54(5):695–714, 2002.
- [73] A.G. Malan, R.W. Lewis, and P. Nithiarasu. *Int. J. Numer. Meth. Engin.*, 54(5):715–729, 2002.
- [74] R.L. Naff, T.F. Russell, and J.D. Wilson. *Shape functions for three-dimensional control-volume mixed finite-element methods on irregular grids*, volume 47 of *Developments in Water Science*, pages 359–366. Elsevier, 2002.
- [75] R. Naff, T. Russell, and J. Wilson. *Computat. Geosci.*, 6:285–314, 2002.
- [76] C.A. Felippa. *Eng. Computation.*, 21(8):867–890, 2004.
- [77] L. Ivan. *Development of high-order CENO finite-volume schemes with block-based adaptive mesh refinement*. PhD thesis, University of Toronto, 2011.
- [78] C.L. Lawson and R.J. Hanson. *Solving least squares problems*. Prentice-Hall, 1974.
- [79] D.J. Mavriplis. AIAA paper 2003-3986, 2003.
- [80] J.S. Park, S.H. Yoon, and C. Kim. *J. Comput. Phys.*, 229(3):788–812, 2010.
- [81] V. Venkatakrishnan. AIAA Paper 93-0880, 1993.
- [82] P.L. Roe. *J. Comput. Phys.*, 43:357–372, 1981.

- [83] P.L. Roe and J. Pike. Efficient construction and utilisation of approximate Riemann solutions. In R. Glowinski and J.L. Lions, editors, *Computing Methods in Applied Science and Engineering*, volume VI, pages 499–518, Amsterdam, 1984. North-Holland.
- [84] S.R. Mathur and J.Y. Murthy. *Numer. Heat Transfer, Part B*, 31(2):195–215, 1997.
- [85] G. Karypis and K. Schloegel. <http://www.cs.umn.edu/~metis>, 2011.
- [86] W. Gropp, E. Lusk, and A. Skjellum. *Using MPI*. MIT Press, Cambridge, Massachussets, 1999.
- [87] C.P.T. Groth and S.A. Northrup. Parallel implicit adaptive mesh refinement scheme for body-fitted multi-block mesh. In *17th AIAA Computational Fluid Dynamics Conference*, Toronto, Ontario, Canada, 6-9 June 2005. AIAA paper 2005-5333.
- [88] M.R.J. Charest, C.P.T. Groth, and Ö.L. Gülder. *Combust. Theor. Modelling*, 14(6):793–825, 2010.
- [89] M.R.J. Charest, C.P.T. Groth, and Ö.L. Gülder. *J. Comput. Phys.*, 231(8):3023–3040, 2012.
- [90] Y. Saad and M.H. Schultz. *SIAM J. Sci. Stat. Comput.*, 7(3):856–869, 1986.
- [91] Y. Saad. *SIAM J. Sci. Stat. Comput.*, 10(6):1200–1232, 1989.
- [92] Y. Saad. *Iterative Methods for Sparse Linear Systems*. PWS Publishing Company, Boston, 1996.
- [93] E. Cuthill and J. McKee. Reducing the bandwidth of sparse symmetric matrices. In *Proceedings of the 1969 24th National Conference*, ACM '69, pages 157–172, New York, NY, USA, 1969. ACM.
- [94] W.A. Mulder and B. Van Leer. *J. Comput. Phys.*, 59:232–246, 1985.
- [95] M. Tabesh and D.W. Zingg. Efficient implicit time-marching methods using a Newton-Krylov algorithm. In *47th AIAA Aerospace Sciences Meeting and Exhibit*, Orlando, Florida, 5-8 January 2009. AIAA paper 2009-0164.
- [96] A. Genz and R. Cools. *ACM Trans. Math. Softw.*, 29(3):297–308, 2003.
- [97] R. Cools and A. Haegemans. *ACM Trans. Math. Softw.*, 29(3):287–296, 2003.
- [98] G.I. Taylor and A.E. Green. *Proc. Royal Soc. London A*, 158(895):499–521, 1937.
- [99] J.S. Sachdev, C.P.T. Groth, and J.J. Gottlieb. *Int. J. Comput. Fluid Dyn.*, 19(2):159–177, 2005.
- [100] X. Gao and C.P.T. Groth. *Int. J. Comput. Fluid Dyn.*, 20(5):349–357, 2006.
- [101] X. Gao and C.P.T. Groth. *J. Comput. Phys.*, 229(9):3250–3275, 2010.
- [102] X. Gao, S. Northrup, and C.P.T. Groth. *Prog. Comput. Fluid. Dy.*, 11(2):76–95, 2011.