WILEY

High-order implicit time-stepping with high-order central essentially-non-oscillatory methods for unsteady three-dimensional computational fluid dynamics simulations

T. Binh Nguyen[1] | Hans De Sterck[2] | Lucie Freret[3] | Clinton P. T. Groth[3]

[1]Computational Engineering Study Program, Vietnamese–German University, Thu Dau Mot City, Vietnam

[2]Department of Applied Mathematics, University of Waterloo, Waterloo, Ontario, Canada

[3]Institute for Aerospace Studies, University of Toronto, Toronto, Ontario, Canada

Correspondence
T. Binh Nguyen, Computational Engineering Study Program, Vietnamese–German University, Le Lai Street, Hoa Phu Ward, Thu Dau Mot City, Binh Duong, Vietnam.
Email: ngthienbinh@gmail.com

Abstract

This article develops high-order implicit time-stepping methods combined with the fourth-order central essentially-non-oscillatory (CENO) scheme for stiff three-dimensional computational fluid dynamics problems having disparate characteristic time scales. Both aerodynamic and magnetohydrodynamic problems are considered on three-dimensional multiblock body-fitted grids with hexahedral cells. Several implicit time integration methods of third- and fourth-order accuracy are considered, including the multistep backward differentiation formulas (BDF4), multistage explicitly singly diagonally implicit Runge-Kutta (ESDIRK4), and Rosenbrock-type methods (ROS34POW2). The resulting nonlinear algebraic system of equations is solved via a preconditioned Jacobian-free inexact Newton–Krylov method with additive Schwarz preconditioning using block-based incomplete LU decomposition. The performance of the high-order implicit time-stepping methods on smooth and stiff problems is compared with a standard fourth-order explicit Runge-Kutta (RK4) method. It is shown that the Rosenbrock methods, despite their advantage of only requiring the solution of linear systems, have significant drawbacks in terms of robustness issues for highly nonlinear compressible flows. The implicit BDF4 and ESDIRK4 methods are found to be much more efficient than the explicit fourth-order RK4 method for a stiff resistive magnetohydrodynamic (MHD) problem discretized with the fourth-order CENO method. When applied to the problem of vortex shedding governed by the Navier–Stokes equations, an A-stable ESDIRK4 scheme proved to be the more robust and accurate implicit time-marching scheme and was able to offer significant speedup compared with the RK4 method. Initial results are also shown for high-order implicit time integration applied to two problems with discontinuities. The current study represents the first to achieve high-order implicit time integration for MHD, enabling large time steps and substantial speedups for stiff MHD problems with high-order

accuracy, and it also represents the first to establish high-order implicit time integration for high-order CENO in space.

1 | INTRODUCTION

Obtaining efficient and stable high-order implicit time integration schemes for high-order spatial discretizations remains a significant challenge for large-scale parallel compressible computational fluid dynamics (CFD). It is well known that CFD problems can have disparate characteristic time scales that are associated with different characteristic waves.[1-5] This is the case for example in solutions of the ideal magnetohydrodynamics (MHD) equations applied to a tokamak geometry.[2,6] By geometric design, the slow magnetosonic and Alfvén waves only propagate in the direction parallel to the background magnetic field, whereas the fast magnetosonic wave propagates in a nearly perpendicular direction. The magnetic field configuration is such that $c_s < c_A \ll c_f$, where c_s, c_A, and c_f are the speeds of the slow, Alfvén, and fast waves, respectively. In MHD problems the dynamics that need to be resolved by simulations often play out primarily on the slow and Alfvén timescales and not on the timescales of the fast MHD waves, thus rendering the numerical simulation problem stiff. Another source of stiffness may arise from the underlying physics of the problem when there exists multiple physical timescales. This often occurs when diffusivity or resistivity terms are present in the governing equations, both for cases with low and high diffusivity.

The stiffness of such CFD problems constrains the choice of the time-stepping methods. Indeed explicit, conditionally stable, time-stepping methods are widely used for time-accurate solutions of unsteady flow problems due to their easy implementation, low cost per time step, and moderate storage requirements, but unfortunately their time step, Δt, is controlled by a so-called Courant-Friedrichs-Lewy (CFL) constraint which must be satisfied to retain stability. For a stiff problem, this condition is determined by the fastest time scale, which may not be of primary of physical interest, and thus computational time and costs can be prohibitive. This renders standard conditionally stable explicit schemes, such as the family of explicit Runge-Kutta (RK) schemes, as being very expensive, or in some cases not practical, for stiff problems in terms of computational resources. Conversely, unconditionally stable, implicit time-stepping methods may provide better overall performance as the time step can be chosen to be much larger while still retaining the desired accuracy for the solution of the slow physical time scales of interest. However, one issue associated with implicit time integration methods is the requirement to solve a large algebraic system, which is either nonlinear or linear, at each time step. For the nonlinear case, Newton's method may be invoked to convert the problem into one of performing a sequence of linear iterations. Direct linear solvers are often infeasible to handle the large dimension of the system.[7] As an alternative, Krylov-based iterative methods, such as the generalized minimal residual (GMRES) method, coupled with Newton's method have proved to be an efficient approach to large, sparse, and nonsymmetric systems.[8,9] Inexact Newton and Jacobian-free approaches[10] have made the implementation of implicit time-stepping schemes far more feasible. The inexact nature of the scheme relates to the early termination of the inner solver loop, for example, the GMRES iteration, reducing computational effort without affecting the overall accuracy. The Jacobian-free feature takes into consideration the fact that Krylov iterative methods require only the results of matrix–vector multiplications and an explicit form of the full Jacobian is not required.

While stiff MHD applications may clearly benefit from efficient implicit solution methods, high-order implicit time integration methods with order of accuracy greater than two have, to the author's knowledge, not been previously described for MHD in the literature. Similarly, high-order implicit time integration has not been considered extensively in conjunction with the high-order CENO spatial discretization. Charest et al.[11] previously applied a standard fourth-order backward differentiation formula (BDF) implicit scheme in conjunction with a fourth-order CENO finite-volume scheme to three-dimensional low-speed viscous flows on unstructured meshes; however, a pseudo-compressibility approach was used to reduce the numerical stiffness for the variable density low-speed flows of interest. Therefore, the focus and main contribution of this study is the development of a fourth-order accurate implicit time integration schemes for compressible MHD combined with a fourth-order-accurate CENO scheme in space, enabling large time steps for stiff MHD problems with high-order accuracy on parallel block-structured grids.

There exist a variety of implicit time-stepping methods, for example the aforementioned multi-step BDF[12,13] have been applied in many engineering problems, especially the second-order BDF2 scheme due to its robustness and stability.[13-15] Unfortunately, higher-order BDF schemes do not fully maintain the desirable stability properties of BDF2.[13-15] Furthermore, depending on the accuracy order, several previous time step solutions are required to determine the solution at the next time step. This may cause difficulties in initializing the time stepping and constant time steps are required to maintain high order of accuracy. As an alternate approach, multi-stage diagonally implicit Runge-Kutta (IRK) and related methods[13,16] can be employed. Thanks to their rational stability functions, IRK methods can be devised that are A-stable for any order (see, e.g., References 13,16 and references therein). In addition, initialization of IRK methods is simple since only one initial condition is required. However, a general IRK method of s stages and m unknowns requires solving a nonlinear system of size $sm \times sm$, which is impractical for large m. Further improvements of the IRK class includes diagonally IRK (DIRK), singly DIRK (SDIRK), or explicit SDIRK (ESDIRK) methods. A detailed discussion of these methods is given in Section 4 below. Besides the well-known BDF and diagonally implicit RK methods, Rosenbrock-type methods,[12,16] which can be seen as a linearization of the diagonally implicit RK methods have also been studied. For classical Rosenbrock schemes, an exact Jacobian is needed for accuracy and stability.[17,18] This may lead to a high cost due to the computation and storage of the Jacobian matrix. To overcome this shortcoming, Rosenbrock-W schemes[17,18] propose an approach to better handle approximate Jacobians, thus facilitating the matrix-free approach and the use of Krylov based methods. In addition, the class of Rosenbrock-K schemes[19] minimize the dimension of the Krylov subspaces coupled with the Rosenbrock schemes. In Reference 20, the authors thoroughly investigate different types of Rosenbrock schemes for the unsteady compressible Navier–Stokes equations. Liu et al.[21] present a comparative study of Rosenbrock and implicit Runge-Kutta methods used in concert with discontinuous Galerkin (DG) spatial discretization schemes, and apply the combined approaches to a variety of numerical simulations for the three-dimensional (3D) unsteady compressible Navier–Stokes equations. They present results for the Taylor–Green vortex problem which is challenging because it incurs disparate length and time scales. It is useful to mention other time-stepping schemes for stiff problems, for example, exponential-type time integrator methods[22-25] which multiply an integrating factor into the original time-dependent system of ordinary differential equations (ODEs) before applying a corresponding time discretization method, or implicit–explicit methods in which an implicit time-stepping method is applied to approximate the fast timescales, whereas the slow scales are numerically treated with a proper explicit time-stepping method.[26,27]

In the present study, high-order implicit time-stepping is pursued for the high-order central essentially-non-oscillatory (CENO) scheme of Ivan et al.[28] applied to stiff CFD problems. The CENO spatial discretization method was introduced by Ivan and Groth[28] for two-dimensional compressible flows,[29] and later extended to three-dimensional nonlinear hyperbolic conservation laws with cubed-sphere grids.[30] The method was successfully applied to inviscid and viscous flows,[11,28,31] the ideal MHD equations[28,30,32] and also applied to reactive turbulent flows.[33] Both isotropic and anisotropic adaptive mesh refinement (AMR) techniques have also been successfully used with the CENO method.[31,32,34-36] Belonging to the essentially-non-oscillatory (ENO) category of schemes, the CENO method adaptively switches between a high-order linear K-exact polynomial reconstruction in regions where the solution is sufficiently smooth, and limited linear least-squares reconstruction in the vicinity of shocks and high gradients.[37] This switching mechanism depends on a so-called smoothness indicator which measures how smooth the solution is over the reconstruction stencil. Note that for CENO, the solution is reconstructed based on a fixed central stencil. This avoids the complexities associated with multiple stencils of other ENO and weighted essentially-non-oscillatory (WENO) schemes, such that the high-order reconstruction is relatively less computational expensive and well suited for extension to unstructured grids. The CENO method has been proven to be suitable for general polygonal grids, it can be implemented with arbitrary order, and it combines well with dynamically adaptive mesh refinements. In addition, high-order accuracy is maintained at curved domain boundaries of body-fitted mesh thanks to accurate spline approximation of the boundaries.

The implementation and performance of several high-order implicit time integrators, in terms of accuracy and efficiency, coupled with the high-order CENO spatial discretization method for several stiff numerical problems are considered herein. In particular, the effectiveness of the high-order BDF4, ESDIRK4, and ROS34POW2 time-implicit methods are examined and compared with the standard conditionally stable, explicit, fourth-order-accurate RK scheme (RK4) in terms of computational cost to achieve a given level of accuracy. While the primary focus of the current study is on problems involving smooth or regular solutions, a standard Sod shock tube problem for the Euler equations and a more challenging MHD Shu-Osher-type problem are also considered herein. The numerical results for these cases confirm that the fourth-order accurate implicit approach works well for shocks, maintaining positivity using the fourth-order accurate CENO spatial discretization for moderate CFL numbers. As is well known, extending strong-stability preserving time integration (SSP) schemes[38,39] to implicit high-order integration methods with larger CFL numbers in an efficient manner

is an unsolved research problem, even for equation systems like compressible Euler that are simpler than MHD, and this remains a topic of future research. Note that the high-order time integration techniques used here as building blocks were developed previously in their basic form, but this work is the first that combines and integrates these building blocks to achieve high-order implicit time integration for high-order MHD and CENO discretizations in space, and the first to thoroughly analyze the potential gains from high-order implicit time integration for MHD and CENO, demonstrating substantial potential gains for stiff MHD problems.

The remainder of the article is organized as follows. The governing equations are described in Section 2. In Section 3 the fourth-order CENO method is briefly summarized. Various implicit time-stepping methods are discussed in detail in Section 4. In Section 5, six numerical simulations are presented to illustrate the benefits of the aforementioned schemes with a focus on stiff problems. In particular, numerical simulations of inviscid flow (Section 5.1), magnetohydrodynamic flow (Sections 5.2 and 5.3), and viscous flow (Section 5.4) are all presented. In Sections 5.5 and 5.6 numerical simulations of inviscid and magnetohydrodynamic flow for problems with shocks are proposed.

2 | GOVERNING EQUATIONS

The governing equations for both nonconducting viscous hydrodynamic and conducting resistive MHD flows can be summarized as

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{v}) = 0, \tag{1}$$

$$\frac{\partial (\rho \mathbf{v})}{\partial t} + \nabla \cdot \left(\rho \mathbf{v} \mathbf{v} + \mathbf{I} \left(p + \frac{\mathbf{B} \cdot \mathbf{B}}{2} \right) - \mathbf{B} \mathbf{B} \right) = \nabla \cdot \tau + \mathbf{S}_m, \tag{2}$$

$$\frac{\partial \mathbf{B}}{\partial t} + \nabla \cdot (\mathbf{v} \mathbf{B} - \mathbf{B} \mathbf{v}) + \nabla \times (\eta \mathbf{J}) = \mathbf{S}_b, \tag{3}$$

$$\frac{\partial e}{\partial t} + \nabla \cdot \left(\left(e + p + \frac{\mathbf{B} \cdot \mathbf{B}}{2} \right) \mathbf{v} - (\mathbf{v} \cdot \mathbf{B}) \mathbf{B} + \eta \mathbf{J} \times \mathbf{B} \right) = \nabla \cdot (\mathbf{v} \cdot \tau) - \nabla \cdot \mathbf{q} + S_e, \tag{4}$$

$$\nabla \cdot \mathbf{B} = 0. \tag{5}$$

Equations (1)–(4) represent the conservation equations for the mass, momentum, induction, and energy respectively. Equation (5) is called the solenoidality condition for the magnetic field. Here, ρ, p, and e are the gas density, pressure, and total energy, respectively. The vectors $\mathbf{v}$, $\mathbf{B}$, $\mathbf{J}$, and $\mathbf{q}$ represent the velocity field, magnetic field, current density, and heat flux. $\mathbf{S}_m$, $\mathbf{S}_b$, and S_e are the source terms in the momentum, induction, and energy equations. The resistivity coefficient, η, is taken here to be constant. Finally, τ is the fluid stress tensor which is explicitly given as[28]

$$\tau = \tau(\mu, \nabla \mathbf{v}) = 2\mu \left(\varepsilon - \frac{1}{3} (\nabla \cdot \mathbf{v}) \mathbf{I} \right), \tag{6}$$

where μ is the dynamic viscosity, $\mathbf{I}$ is the identity tensor, and

$$\varepsilon = \frac{1}{2} \left(\nabla \mathbf{v} + (\nabla \mathbf{v})^T \right), \tag{7}$$

is the fluid strain rate tensor. The relation between pressure and energy is given through the following equation of state for a perfect gas

$$p = (\gamma - 1) \left(e - \frac{1}{2} \rho \|\mathbf{v}\|^2 - \frac{1}{2} \|\mathbf{B}\|^2 \right), \tag{8}$$

where γ is the adiabatic index or specific heat ratio. The square length of a vector is denoted by $\| \cdot \|^2 = (\cdot)_x^2 + (\cdot)_y^2 + (\cdot)_z^2$. The heat flux follows from Fourier's law of thermal conduction given by

$$\mathbf{q} = -\kappa \nabla T, \tag{9}$$

where κ is the thermal conductivity, and T is the temperature. Applying Ampere's law, and ignoring the displacement current, the current density can be expressed in terms of the curl of the magnetic field

$$\mathbf{J} = \nabla \times \mathbf{B} = \left[\frac{\partial B_z}{\partial y} - \frac{\partial B_y}{\partial z}, \frac{\partial B_x}{\partial z} - \frac{\partial B_z}{\partial x}, \frac{\partial B_y}{\partial x} - \frac{\partial B_x}{\partial y} \right]^T. \tag{10}$$

For clarity, these terms are explicitly expanded here in terms $\mathbf{J}$ and $\mathbf{B}$ and can be expressed as

$$\nabla \times \mathbf{J} = \nabla \cdot \begin{bmatrix} 0 & \frac{\partial B_y}{\partial x} - \frac{\partial B_x}{\partial y} & \frac{\partial B_z}{\partial x} - \frac{\partial B_x}{\partial z} \\ \frac{\partial B_x}{\partial y} - \frac{\partial B_y}{\partial x} & 0 & \frac{\partial B_z}{\partial y} - \frac{\partial B_y}{\partial z} \\ \frac{\partial B_x}{\partial z} - \frac{\partial B_z}{\partial x} & \frac{\partial B_y}{\partial z} - \frac{\partial B_z}{\partial y} & 0 \end{bmatrix}, \tag{11}$$

and

$$\nabla \cdot (\mathbf{J} \times \mathbf{B}) = \frac{\partial \left(B_z J_y - B_y J_z \right)}{\partial x} + \frac{\partial \left(B_x J_z - B_z J_x \right)}{\partial y} + \frac{\partial \left(B_y J_x - B_x J_y \right)}{\partial z}. \tag{12}$$

For nonconducting hydrodynamic flows, where there are no magnetic fields ($\mathbf{B} = \mathbf{0}$), Equations (1)–(5) reduce to the Navier–Stokes equations describing compressible laminar viscous flows, for which $\mu \neq 0$ and $\kappa \neq 0$, and further reduce to the Euler equations if the viscosity and thermal conduction vanish ($\mu = \kappa = 0$). The latter represents compressible inviscid flows. When a magnetic field is present ($\mathbf{B} \neq 0$) and resistivity is negligible such that $\eta = 0$, the equations revert to the well-known ideal MHD equations governing perfectly conducting fully ionized plasma flows. Finally, for $\mathbf{B} \neq 0$ and $\eta \neq 0$, Equations (1)–(5) are the so-called resistive MHD equations.

The characteristic MHD wavespeeds for Equations (1)–(5) above are given by, for example, in the x-direction,

$$c_{fx}^2 = \frac{1}{2} \left(\frac{\gamma p + \|\mathbf{B}\|^2}{\rho} + \sqrt{\left(\frac{\gamma p + \|\mathbf{B}\|^2}{\rho} \right)^2 - 4 \frac{\gamma p B_x^2}{\rho^2}} \right), \tag{13}$$

$$c_{sx}^2 = \frac{1}{2} \left(\frac{\gamma p + \|\mathbf{B}\|^2}{\rho} - \sqrt{\left(\frac{\gamma p + \|\mathbf{B}\|^2}{\rho} \right)^2 - 4 \frac{\gamma p B_x^2}{\rho^2}} \right), \tag{14}$$

$$c_{Ax}^2 = \frac{B_x^2}{\rho}, \tag{15}$$

where c_{fx}, c_{sx}, and c_{Ax} are the fast, slow, and Alfvén speeds, respectively. The eigenvalues, λ_i, of the flux Jacobian for the MHD equation are as follow

$$\begin{aligned} \lambda_{1,2} &= v_x \pm c_{fx}, & \lambda_{3,4} &= v_x \pm c_{Ax}, \\ \lambda_{5,6} &= v_x \pm c_{sx}, & \lambda_7 &= v_x \pm c_{Ax}, & \lambda_8 &= 0. \end{aligned} \tag{16}$$

The corresponding eigenvectors can also be found. See, for example, the article by Susanto et al.[40]

3 | FINITE-VOLUME CENO SPATIAL DISCRETIZATION

3.1 | High-order finite-volume CENO scheme

As discussed in the introduction, spatial discretization of the preceding conservation equations is considered herein using a high-order finite-volume CENO scheme.[28,30,34,40,41] The CENO schemes have been successfully applied to both inviscid and viscous compressible flows[28,34] as well as to ideal MHD flows.[30,32] This family of schemes is also readily applicable to

high-order solutions on hexahedral cells with nonplanar cell faces and, in particular, those associated with cubed-sphere grids.[30] Moreover it has been developed and applied to problems involving disparate spatial scales when used in conjunction with both isotropic and anisotropic block-based AMR strategies.[31,32,34-36] Detailed descriptions of the CENO method can be found in References 29,42 and references therein.

A fourth-order CENO finite-volume scheme is applied here to the weak conservation form of the governing equations. Equations (1)–(4) can be rewritten in weak-conservation vector form as

$$\frac{\partial \mathbf{U}}{\partial t} + \nabla \cdot \mathbf{F} = \mathbf{S}, \tag{17}$$

where $\mathbf{U}$ is the vector of conservative variables and given by

$$\mathbf{U} = [\rho, \rho\mathbf{v}, e, \mathbf{B}]^{\mathrm{T}}, \tag{18}$$

$\mathbf{F}$ is the corresponding flux dyad, and $\mathbf{S}$ is the vector source term. The flux dyad comprises both the inviscid (hyperbolic), $\mathbf{F}^H$, and viscous (elliptic), $\mathbf{F}^E$, components where the latter depends on both the solution states and their gradients and can be written as

$$\nabla \cdot \mathbf{F} = \nabla \cdot \mathbf{F}^H(\mathbf{U}) + \nabla \cdot \mathbf{F}^E(\mathbf{U}, \nabla\mathbf{U}). \tag{19}$$

Integrating Equation (17) over a hexahedral cell, $\mathcal{I}_{ijk}$, applying the divergence theorem to the flux dyad term, and subsequently dividing by the cell volume, one can obtain the following integral form:

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{1}{V_{ijk}}\iiint_{V_{ijk}}\mathbf{U}(\mathbf{X})dV\right) = \frac{1}{V_{ijk}}\left(-\oiint_{\partial V_{ijk}}\mathbf{F}\cdot\mathbf{n}dA + \iiint_{V_{ijk}}\mathbf{S}(\mathbf{X})dV\right), \tag{20}$$

where V_{ijk} and ∂V_{ijk} are the volume and surface of cell of interest, $\mathcal{I}_{ijk}$, and $\mathbf{n}$ is the unit outward normal vector for the surface ∂V_{ijk}. Introducing numerical Gaussian quadrature to approximate both the surface and volume integrals appearing in the equation above, the so-called semidiscrete form of the governing equations for cell, $\mathcal{I}_{ijk}$, can be obtained and written as

$$\frac{\mathrm{d}\overline{\mathbf{U}}_{ijk}}{\mathrm{d}t} = -\frac{1}{V_{ijk}}\sum_{l=1}^{N_f}\sum_{m=1}^{N_g}(\omega\mathbf{F}_{\mathrm{num}}\cdot\mathbf{n}\Delta A)_{ijk,l,m} + \overline{\mathbf{S}}_{ijk} \equiv \mathbf{R}_{ijk}(\overline{\mathbf{U}}), \tag{21}$$

where

$$\mathbf{F}_{\mathrm{num}}\cdot\mathbf{n} = \left(\mathbf{F}_{\mathrm{num}}^H + \mathbf{F}_{\mathrm{num}}^E\right)\cdot\mathbf{n} = \mathcal{F}(\mathbf{U}_L, \mathbf{U}_R, \mathbf{n}) + \mathbf{F}_{\mathrm{num}}^E\cdot\mathbf{n}, \tag{22}$$

and where $\overline{\mathbf{U}}_{ijk}$ and $\overline{\mathbf{S}}_{ijk}$ are the volume-averaged values of the solution and source vectors, respectively, for cell $\mathcal{I}_{ijk}$ defined by

$$\overline{\mathbf{U}}_{ijk} \equiv \frac{1}{V_{ijk}}\iiint_{V_{ijk}}\mathbf{U}(\mathbf{X})dV, \quad \overline{\mathbf{S}}_{ijk} \equiv \frac{1}{V_{ijk}}\iiint_{V_{ijk}}\mathbf{S}(\mathbf{X})dV. \tag{23}$$

In the above, $\mathbf{X} = (x, y, z)$ is the coordinate vector $\mathbf{X} = (x, y, z)$, $\mathbf{F}_{\mathrm{num}}$ is the so-called numerical flux defined by the numerical flux function, and ΔA the surface area. Here, $N_f = 6$ is the number of faces in each hexahedral cell, N_g is the number of Gaussian quadrature points, and ω is the weight associated with each of the Gauss points. Additionally, $\mathbf{R}_{ijk}(\overline{\mathbf{U}})$ is defined to be the residual vector for cell $\mathcal{I}_{ijk}$.

Standard Riemann-solver-based numerical flux functions such as Lax-Friedrichs, Roe, or HLLE[43,44] are used to compute the hyperbolic numerical flux, $\mathbf{F}_{\mathrm{num}}^H\cdot\mathbf{n} = \mathcal{F}(\mathbf{U}_L, \mathbf{U}_R, \mathbf{n})$. A central-type approximation of the solution gradients $\nabla\mathbf{U}_{i+\frac{1}{2},j,k}$ at cell interfaces is used in the evaluation of the numerical viscous or elliptic flux, $\mathbf{F}_{\mathrm{num}}^E$, which are given by

$$\mathbf{F}_{\mathrm{num}}^E\left(\mathbf{U}_{i+\frac{1}{2},j,k}, \nabla\mathbf{U}_{i+\frac{1}{2},j,k}\right) = \mathbf{F}^E\left(\frac{1}{2}\left(\mathbf{U}_L^K + \mathbf{U}_R^K\right), \frac{1}{2}\left(\nabla\mathbf{U}_L^K + \nabla\mathbf{U}_R^K\right)\right), \tag{24}$$

where $\mathbf{U}_L^K$ and $\mathbf{U}_R^K$ are the high-order K-exact left and right solution reconstructions provided by the CENO method. The reconstructed gradient is given by

$$\nabla \mathbf{U}^K(\mathbf{X}) = \frac{\partial \mathbf{U}^K}{\partial x}\mathbf{i} + \frac{\partial \mathbf{U}^K}{\partial y}\mathbf{j} + \frac{\partial \mathbf{U}^K}{\partial z}\mathbf{k}, \tag{25}$$

where the spatial derivatives are obtained by differentiating directly the reconstruction polynomial (26). Additional details of the procedures used in the flux evaluation are given by Ivan and Groth.[28]

Evaluation of the numerical flux at faces of the hexahedral cell, $\mathcal{I}_{ijk}$, requires reconstruction of the solution within each cell based on cell-averaged solutions quantities associated with a stencil of cells having compact support. Unlike other ENO schemes in which a hierarchy of multiple upwinding stencils are used in the solution reconstruction,[45,46] or WENO schemes with a high-order reconstructed solution based on nonlinear convex combinations of several low-order reconstructions,[47,48] the high-order, linear (optimal), solution reconstruction of the CENO method is based on a fixed, central, and overdetermined reconstruction stencil using Barth's K-exact polynomial reconstruction.[28,30,40] The choice of overdetermined stencils is particularly useful for complicated grid topology, for example, unstructured grids or grid degeneration near root-block edges of a cubed-sphere grid,[30,40] where the number of available cells is fewer than that of regular Cartesian grid topology. Following Ivan and Groth,[30] a K-exact polynomial reconstruction for each solution variable in cell $\mathcal{I}_{ijk}$ is used of the form

$$u_{ijk}^K(\mathbf{X}) = \sum_{\substack{p_1=0 \\ p_1+p_2+p_3 \leq K}}^{K} \sum_{p_2=0}^{K} \sum_{p_3=0}^{K} (x - \bar{x}_{ijk})^{p_1}(y - \bar{y}_{ijk})^{p_2}(z - \bar{z}_{ijk})^{p_3} D_{p_1 p_2 p_3}^K, \tag{26}$$

where K is the polynomial degree, $\overline{\mathbf{X}} = (\bar{x}_{ijk}, \bar{y}_{ijk}, \bar{z}_{ijk})$ is the cell center, and $D_{p_1 p_2 p_3}^K$ are the polynomial coefficients to be determined. For the purely hyperbolic equations of interest here (i.e., the Euler and ideal MHD equations), a $K = 3$ cubic polynomial reconstruction is chosen here so that the resulting CENO scheme is fourth-order accurate. For the governing equations involving both hyperbolic and elliptic fluxes (i.e., the Navier–Stokes and resistive MHD equations), a $K = 4$ quartic polynomial reconstruction is adopted to ensure consistent fourth-order accuracy of the scheme.[11,28] The total number, N_D, of unknown coefficients, $D_{p_1 p_2 p_3}^K$, for the fourth-order reconstruction scheme is

$$N_D = \frac{(K + 1)(K + 2)(K + 3)}{6} = 20 \text{ and } 35, \tag{27}$$

for $K = 3$ and $K = 4$, respectively, and they are computed such that the following conditions are satisfied:[30,37,45]

- *Conservation of the mean*:

$$\bar{u}_{ijk} = \frac{1}{V_{ijk}} \iiint_{V_{ijk}} u_{ijk}^K(\mathbf{X})dV, \tag{28}$$

and

$$\left(\frac{1}{V_{\gamma\delta\zeta}} \iiint_{V_{\gamma\delta\zeta}} u_{ijk}^K(\mathbf{X})dV \right) - \bar{u}_{\gamma\delta\zeta} = 0, \tag{29}$$

for all conserved solution variables, u, and for all cells $\mathcal{I}_{\gamma\delta\zeta} \neq \mathcal{I}_{ijk}$ in the reconstruction stencil of $\mathcal{I}_{ijk}$.

- *K-exactness*:

$$u_{ijk}^K(\mathbf{X}) - u_{\text{exact}}(\mathbf{X}) = \mathcal{O}\left(\Delta x^{K+1}\right). \tag{30}$$

Here, $u_{\text{exact}}(\mathbf{X})$ is the exact solution, and Δx is the grid size.

- *Compactness*: the polynomial function is reconstructed over an overdetermined reconstruction stencil with the number of cells that is at least equal to the number of unknown coefficients.

The unknown coefficients, $D_{p_1 p_2 p_3}^K$, are determined by first representing D_{000}^K in terms of the other coefficients, substituting Equation (26) into Equation (28). By enforcing the condition of Equation (29), one solves an overdetermined linear system for the unknowns using either the normal equation, QR factorization, or multiplication with a pseudo-inverse matrix.[49] A combination of inverse-distance weighting and column-scaling are used here to deal with conditioning and/or invertibility issues of the least-squares problem that arise as the stencil size and hence the size of the least-squares problem grow with the order of the K-exact resconstruction procedure.[28] The validity of Equation (30) can be readily shown using standard Taylor approximation theory. For a more detailed discussion of these properties, the reader is referred to the original article by Ivan and Groth.[28]

The main concept behind all ENO/WENO schemes is a switching mechanism between high-order reconstruction in smooth regions and low-order reconstruction in areas where the solution is deemed nonsmooth or underresolved, in order to suppress $\mathcal{O}(1)$, nonphysical, Gibbs-like oscillations.[45,47] The CENO method uses a limited linear least-squares reconstruction polynomial of degree one ($K = 1$), for example, a MUSCL method[43] as the low-order scheme. A variety of slope limiters can be used within CENO such as Barth-Jespersen limiter[37] or Venkatakrishnan limiter.[50] Additionally, a smoothness indicator determines the switching between the high-order and low-order reconstruction. Various proposed smoothness indicators for ENO/WENO schemes are available, see, for example, References 47,48,51. Following Ivan et al.,[30] the CENO smoothness indicator, S, is evaluated as

$$S = \frac{\alpha c_s}{\max(1 - \alpha, \varepsilon)}, \tag{31}$$

where

$$\alpha = 1 - \frac{\sum_\gamma \sum_\delta \sum_\zeta \left(u_{\gamma \delta \zeta}^K \left(\vec{X}_{\gamma \delta \zeta} \right) - u_{ijk}^K \left(\vec{X}_{\gamma \delta \zeta} \right) \right)^2}{\sum_\gamma \sum_\delta \sum_\zeta \left(u_{\gamma \delta \zeta}^K \left(\vec{X}_{\gamma \delta \zeta} \right) - \bar{u}_{ijk}^K \right)^2}, \tag{32}$$

which measures how accurately centroidal solution values of neighboring cells can be reproduced using the reconstruction for cell $\mathcal{I}_{ijk}$. Here, $\varepsilon = 10^{-8}$ and is introduced to prevent division by zero, and

$$c_s = \frac{N_{\text{SOS}} - N_D}{N_D - 1}. \tag{33}$$

For the purely hyperbolic conservation equations and $K = 3$ cubic reconstruction of interest here, $N_D = 20$ as given by Equation (27) above, and $N_{\text{SOS}} = 32$ is the actual number of cells used in the overdetermined reconstruction stencil based on an extended nearest neighbor reconstruction stencil. In the case of conservation equations also involving elliptic numerical fluxes and $K = 4$ quartic reconstruction, $N_D = 35$ and $N_{\text{SOS}} = 56$ or $N_{\text{SOS}} = 125$ based on stencils involving next to nearest neighbors.[30] Note that in both cases $N_{\text{SOS}} > N_D$ as required for an overdetermined stencil.

It is observed that for smooth flow, the smoothness parameter, α, tends to 1 and it follows that $S \to c_s / \varepsilon$ whereas α tends to $-\infty$ such that $S \to -c_s$ for nonsmooth or underresolved features. The switching between third-order and first-order reconstructions is then triggered as follows:

$$\begin{cases} S > S_c & \text{smooth case} \Rightarrow \text{Barth's optimal third-order reconstruction with } K = 3, \\ S \leq S_c & \text{nonsmooth case} \Rightarrow \text{limited first-order reconstruction with } K = 1. \end{cases} \tag{34}$$

Here, S_c is a user-defined cutoff parameter chosen as $S_c = 1500$ for all the simulations considered in Section 5 of this study. In practice, in order to prevent the 0/0 indefinite form in the second term of α, the computation of the smoothness indicator, S, is carried out in cell, $\mathcal{I}_{ijk}$, only when a gauge parameter ξ_{ijk} exceeds some threshold value, that is,[30]

$$\xi_{ijk} > \varepsilon_A + \varepsilon_R \bar{u}_{ijk}, \tag{35}$$

where

$$\xi_{ijk} = \sqrt{\frac{1}{N_D - 1} \sum_{\substack{p_1 = 0 \\ 0 < p_1 + p_2 + p_3 \leq K}}^K \sum_{p_2 = 0}^K \sum_{p_3 = 0}^K (D_{p_1 p_2 p_3}^K)^2 (V_{ijk}^{2/3})^{p_1 + p_2 + p_3}}, \tag{36}$$

which measures the variability of the solution u in cell $\mathcal{I}_{ijk}$. Here, ε_A and ε_R represent the absolute and relative variability thresholds, respectively. For the numerical simulations considered herein, $\varepsilon_A = \varepsilon_R = 10^{-5}$ as prescribed by Ivan et al.[30]

3.2 | GLM control of the $\nabla \cdot \mathbf{B} = 0$ constraint

For the purpose of the numerical treatment of the solenoidality condition on the magnetic field, $\vec{B}$, following the work of Susanto et al.,[40] the generalized Lagrangian multiplier (GLM) divergence cleaning originally proposed by Dedner et al.[52] is used here. For this, Equations (3) and (5) are replaced by

$$\frac{\partial \vec{B}}{\partial t} + \nabla \cdot \left(\vec{v}\vec{B} - \vec{B}\vec{v} \right) + \nabla \psi = 0, \tag{37}$$

$$\frac{\partial \psi}{\partial t} + c_h^2 \nabla \cdot \vec{B} = -\frac{c_h^2}{c_p^2} \psi. \tag{38}$$

Here, ψ is an additional nonphysical potential variable which is added to the solution vector of unknowns, and the coefficients c_h and c_p are appropriately selected to control the diffusion in ψ. By introducing ψ, the errors in the divergence of the magnetic field are propagated out of the computational domain, whereas other physical conservative variables retain their hyperbolicity. It is also advantageous that no staggered grids are needed with the GLM-MHD formulation, and the method is readily applicable to problems with more general grid topologies. As suggested by Susanto et al.,[40] c_h is chosen as follows,

$$c_h = \max \left(|v_n| + c_{f_n} \right), \tag{39}$$

where v_n and c_{f_n} are the velocity component and fast MHD wave in a direction normal to cell interfaces, and the maximum value is taken over all cells in the reconstructed stencil. This choice allows the error introduced by ψ to be propagated out of the computational domain as rapidly as possible, yet does not add additional constraints on the time step, Δt, as determined by the CFL condition. It is recommended that c_p satisfy the relation $c_r = c_p^2/c_h = 0.18$.[40]

Note that, besides the GLM-MHD method described above, a number of other methods have been proposed for handling the difficulty of the solenoidality condition, without resorting to staggered mesh approaches. For example, the so-called "Hodge Projection" scheme,[53,54] which projects a vector field onto its solenoidal part, or the Powell method,[55] where an additional source term proportional to the divergence of the magnetic field is introduced into the ideal MHD equations to maintain strict hyperbolicity and Galilean invariance are possible alternatives. Please refer to Reference 55 and references given therein.

4 | HIGH-ORDER IMPLICIT TIME-STEPPING METHODS

Application of the fourth-order CENO finite-volume spatial discretization scheme described in the previous section results in the semidiscrete form of the governing equations given by Equation (21). The latter is a coupled system of nonlinear ODEs for the cell-averaged values of conserved solution vector, $\overline{\mathbf{U}}_{ijk}$ for computational cell, $\mathcal{I}_{ijk}$, and can be reexpressed as

$$\frac{d\overline{\mathbf{U}}}{dt} = \mathbf{R}(\overline{\mathbf{U}}), \quad \overline{\mathbf{U}} \in \mathbb{R}^m, \tag{40}$$

where $\overline{\mathbf{U}}$ is taken to be the solution vector containing all volume-averaged solution quantities from all cells in the entire computational domain. Given initial data for $\overline{\mathbf{U}}$, this coupled system of nonlinear ODEs can be evolved forward in time using a variety of time-marching schemes. For robust implicit time-stepping, desirable methods are A-stable, that is, the stability region contains the entire left half of the complex plane.

4.1 | Backward differentiation formulas (BDF)

BDF time-marching schemes are linear multistep methods that approximate the solution, $\overline{\mathbf{U}}^{n+1}$, at time step t_{n+1} by a pth degree polynomial that interpolates $\overline{\mathbf{U}}^{n+1-k}$ at t_{n+1-k}, $k = 0, \ldots, p$. This results in a nonlinear equation for $\overline{\mathbf{U}}^{n+1}$

$$\sum_{k=0}^{p} \alpha_k \overline{\mathbf{U}}^{n+1-k} = \Delta t \mathbf{R} \left(\overline{\mathbf{U}}^{n+1} \right), \tag{41}$$

where the coefficients, α_k, are identified from the interpolation process. Due to the second Dahlquist barrier,[12,14] linear multistep methods with an order of accuracy greater than two cannot be A-stable and BDF schemes of order $p > 2$ can be at most A(α)-stable with the value of α decreasing as p increases. For comparison with other high-order implicit time-stepping methods of interest here, the fourth-order BDF4 method applied to the ODE system of Equation (40) can be written as

$$\frac{25}{12}\overline{\mathbf{U}}^{n+1} - \frac{48}{12}\overline{\mathbf{U}}^{n} + \frac{36}{12}\overline{\mathbf{U}}^{n-1} - \frac{16}{12}\overline{\mathbf{U}}^{n-2} + \frac{3}{12}\overline{\mathbf{U}}^{n-3} - \Delta t \mathbf{R} \left(\overline{\mathbf{U}}^{n+1} \right) = 0. \tag{42}$$

4.2 | Diagonally implicit Runge-Kutta (DIRK) methods

An alternative approach for implicit time integrators is to apply multistage IRK methods. The s-stage IRK method applied to Equation (40) is given by

$$\mathbf{k}_i = \mathbf{R} \left(\overline{\mathbf{U}}^{n} + \Delta t \sum_{j=1}^{s} a_{ij} \mathbf{k}_j \right), \quad \text{for } i = 1, \ldots, s, \tag{43}$$

$$\overline{\mathbf{U}}^{n+1} = \overline{\mathbf{U}}^{n} + \Delta t \sum_{i=1}^{s} b_i \mathbf{k}_i, \tag{44}$$

where $\mathbf{k}_i$ are stage tangent vectors of the same dimension as $\overline{\mathbf{U}}$, and the coefficients a_{ij}, b_i are specified to satisfy the desired accuracy and stability properties. For IRK methods, a full nonlinear system of size $sm \times sm$ must be solved at every time step. This is often infeasible in practice for $s \geq 2$ and large m, as for example in the system arising from Equation (21). Diagonally IRK (DIRK) methods[16,56] circumvent this difficulty. By choosing $a_{ij} = 0$ for all $j > i$, only a sequence of s nonlinear systems of smaller size needs solving. A DIRK scheme is formulated as

$$\mathbf{s}_i = \overline{\mathbf{U}}^{n} + \Delta t \sum_{j=1}^{i-1} a_{ij} \mathbf{k}_j, \tag{45}$$

$$\mathbf{k}_i = \mathbf{R} \left(\mathbf{s}_i + \Delta t a_{ii} \mathbf{k}_i \right), \tag{46}$$

for $i = 1, \ldots, s$, and the solution is updated following Equation (44). If all values of a_{ii} are identical, the method is called singly DIRK (SDIRK).[16] A further relaxation is to have the first stage solved explicitly results in ESDIRK methods.[16]

Having a rational stability function,[16] all IRK methods are A-stable. This allows a choice of high-order DIRK schemes where the costly computation of solving s nonlinear systems at each time step is compensated by choosing a time step size, Δt, as large as possible while achieving a desired solution accuracy. Further restriction on stability allows the design of stiffly accurate DIRK methods, or L-stable schemes which are desirable for especially stiff problems. L-stability implies an A-stable scheme with an additional condition

$$\lim_{z \to \infty} \phi(z) = 0, \tag{47}$$

where $\phi(z)$ is the stability function. It is recalled that L-stable methods correctly represent the solution quantitatively, even for large time steps. This is due to the fact that $\phi(z) = \phi(\lambda \Delta t) \approx 0$ even when Δt is large, which resembles the analytic

TABLE 1 Coefficients for ESDIRK4 scheme[20]

$a_{ii} = \frac{1}{4}$					
$a_{21} = \frac{1}{4}$					
$a_{31} = \frac{8611}{62500}$	$a_{32} = -\frac{1743}{31250}$				
$a_{41} = \frac{5012029}{34652500}$	$a_{42} = -\frac{654441}{2922500}$	$a_{43} = \frac{174375}{388108}$			
$a_{51} = \frac{15267082809}{155376265600}$	$a_{52} = -\frac{71443401}{120774400}$	$a_{53} = \frac{730878875}{902184768}$	$a_{54} = \frac{2285395}{8070912}$		
$a_{61} = \frac{82889}{524892}$	$a_{62} = 0$	$a_{63} = \frac{15625}{83664}$	$a_{64} = \frac{69875}{10672}$	$a_{65} = -\frac{2260}{8211}$	
$b_1 = \frac{82889}{524892}$	$b_2 = 0$	$b_3 = \frac{15625}{83664}$	$b_4 = \frac{69875}{102672}$	$b_5 = -\frac{2260}{8211}$	$b_6 = \frac{1}{4}$

solution $e^{\lambda \Delta t} \approx 0$. In this study, the L-stable, six-stage, fourth-order, ESDIRK4 method,[57] which has been previously applied to the solution of CFD problems,[20] is considered. The coefficients of this scheme are listed in Table 1.

4.3 | Rosenbrock-type methods

The idea of Rosenbrock methods is to linearize the nonlinear systems arising in the formulas of SDIRK schemes. Thus, nonlinear iteration techniques (i.e., Newton's method) are not required. Instead, one only needs to solve a sequence of s linear systems in each time step. This strategy may considerably reduce the computational cost of each time step. The tradeoff for this reduction in computational effort is a loss of some stability and accuracy. Rosenbrock methods can be thought of as applying only one Newton iteration to the nonlinear system of each DIRK stage, but the coefficients are chosen such that the desired accuracy and stability can be achieved.

Linearizing Equation (45) about $\mathbf{s}_i$ yields

$$\mathbf{k}_i = \mathbf{R}(\mathbf{s}_i) + \Delta t a_{ii} \mathbf{J} \mathbf{k}_i, \tag{48}$$

where $\frac{\partial \mathbf{R}(\overline{\mathbf{U}})}{\partial \overline{\mathbf{U}}}\big|_{\mathbf{s}_i}$ at each stage has been replaced by $\mathbf{J} \equiv \frac{\partial \mathbf{R}(\overline{\mathbf{U}})}{\partial \overline{\mathbf{U}}}\big|_{\overline{\mathbf{U}}^n}$. Replacing a_{ii} by $\gamma_{ii} = \gamma$, and adding unknowns γ_{ij}, for $j < i$, to create more degrees of freedom in the method, the general form of Rosenbrock-type methods is given by

$$\begin{cases} (\mathbf{I} - \gamma_{ii} \Delta t \mathbf{J}) \mathbf{k}_i = \mathbf{R}(\mathbf{s}_i) + \Delta t \mathbf{J} \sum_{j=1}^{i-1} \gamma_{ij} \mathbf{k}_j, \\ \mathbf{s}_i = \overline{\mathbf{U}}^n + \Delta t \sum_{j=1}^{i-1} a_{ij} \mathbf{k}_j, \end{cases} \quad i = 1, \dots, s, \tag{49}$$

$$\overline{\mathbf{U}}^{n+1} = \overline{\mathbf{U}}^n + \Delta t \sum_{i=1}^{s} b_i \mathbf{k}_i. \tag{50}$$

In practice, the multiple matrix–vector multiplications on the right-hand side of Equation (49) can be avoided by an efficient implementation. Introducing the auxiliary variables

$$\begin{cases} \mathbf{V}_i = \Delta t \sum_{j=1}^{i} \gamma_{ij} \mathbf{k}_j, \\ \mathbf{k}_i = \frac{1}{\Delta t} \left(\frac{1}{\gamma_{ii}} \mathbf{V}_i - \sum_{j=1}^{i-1} c_{ij} \mathbf{V}_j \right), \end{cases} \quad i = 1, \dots, s. \tag{51}$$

TABLE 2 Coefficients for ROS34PW2 scheme[16]

$\gamma = 0.435866521508459$	
$a_{21} = 2$	$\gamma_{21} = -0.87173304301691801$
$a_{31} = 1.419217317455765$	$\gamma_{31} = -0.90338057013044082$
$a_{32} = -0.259232211672970$	$\gamma_{32} = 0.054180672388095326$
$a_{41} = 4.184760482319161$	$\gamma_{41} = 0.24212380706095346$
$a_{42} = -0.285192017355496$	$\gamma_{42} = -1.2232505839045147$
$a_{43} = 2.294280360279042$	$\gamma_{43} = 0.54526025533510214$
$b_1 = 0.24212380706095346$	$b_2 = -1.2232505839045147$
$b_3 = 1.545260255335102$	$b_4 = 0.435866521508459$

Equations (49)–(50) can be rearranged as

$$
\begin{cases}
(\mathbf{I} - \gamma_{ii}\Delta t\mathbf{J})\mathbf{V}_i = \Delta t\gamma_{ii}\mathbf{R}(\hat{\mathbf{s}}_i) + \gamma_{ii}\displaystyle\sum_{j=1}^{i-1} c_{ij}\mathbf{V}_j, \\[2mm]
\hat{\mathbf{s}}_i = \overline{\mathbf{U}}^n + \displaystyle\sum_{j=1}^{i-1} \tilde{a}_{ij}\mathbf{V}_j,
\end{cases}
\qquad i = 1, \ldots, s,
\tag{52}
$$

$$
\overline{\mathbf{U}}^{n+1} = \overline{\mathbf{U}}^n + \sum_{i=1}^{s} m_i\mathbf{V}_i,
\tag{53}
$$

where, with $\mathbf{\Gamma} = (\gamma_{ij})$ and $\mathbf{A} = (a_{ij})$, the c_{ij}, $\tilde{a}_{ij}$, and m_i coefficients are given by $\mathbf{C} = (c_{ij}) = \mathrm{diag}(\gamma_{11}^{-1}, \ldots, \gamma_{ss}^{-1}) - \mathbf{\Gamma}^{-1}$, $\tilde{\mathbf{A}} = (\tilde{a}_{ij}) = \mathbf{A}\mathbf{\Gamma}^{-1}$, and $\mathbf{m}^t = \mathbf{b}^t\mathbf{\Gamma}^{-1}$.

Obtaining an exact form of the Jacobian, $\mathbf{J}$, is not an easy task, especially for large systems which can arise from a method of lines approach applied to the semidiscrete form of the PDEs. Instead, one may seek an approximation of the Jacobian, for example, using a low-order finite-difference approximation. Such a strategy is adopted in the class of Rosenbrock-W methods[18] which considered as part of this study. The coefficients in this case are computed in a way that attempts to retain the desired order of accuracy when a matrix, $\mathbf{W}$, is used to approximate the Jacobian, $\mathbf{J}$. It is also observed that Rosenbrock-W methods may suffer from an order reduction when applied to the semidiscrete forms of certain types of PDEs.[58,59] Examples of high-order Rosenbrock-W methods include the third- and fourth-order schemes proposed by Rang and Angermann.[18] In this study, the third-order, four-stage Rosenbrock-W ROS34PW2,[18] which supports the Jacobian-free approach is applied for the numerical simulations and comparisons presented in Section 5. The coefficients of this scheme are given in Table 2, with $\gamma_{ii} = \gamma \ \forall i$.

4.4 | Solution of algebraic systems

Application of the implicit time-marching methods discussed above results in algebraic systems of equations that must be solved at each time step. The systems are generally sparse, nonsymmetric, and often ill-conditioned. Depending on which temporal integration method is used, the resulting algebraic system is either nonlinear, as for the BDF and DIRK methods, or linear, as for the Rosenbrock-type schemes. Newton's method coupled with the GMRES method is invoked here for iterative solution of the nonlinear equations, and in the case of the Rosenbrock schemes, the GMRES method alone is used for solving the resulting linear system. Since the algebraic systems are ill-conditioned, preconditioning is important in both instances. The additive global Schwarz preconditioner with incomplete LU decomposition in each block is applied here for all the numerical simulations considered in Section 5. For Rosenbrock schemes, as the matrix on the left-hand side is the same for all stages, the same preconditioner can be applied for all stages of each time step. Furthermore, as is possible with the GMRES approach, a Jacobian-free implementation of the iterative method is adopted as the Krylov subspace construction requires only Jacobian-vector multiplications.

All of the implicit time-marching methods of interest here applied to the coupled ODEs of Equation (40) can be written in the following form:

$$\mathbf{G}(\mathbf{U}) \equiv \mathbf{U} - \tilde{\mathbf{U}} - \alpha \Delta t \mathbf{R}(\mathbf{U}) = \mathbf{0}. \tag{54}$$

Starting from an initial estimate, $\mathbf{U}^{(0)}$, Newton's method seeks the root of Equation (54) by the iteration

$$\begin{cases} \dfrac{\partial \mathbf{G}(\mathbf{U})}{\partial \mathbf{U}}\Big|_{\mathbf{U}^{(k)}} \Delta \mathbf{U} = -\mathbf{G}\left(\mathbf{U}^{(k)}\right), \\ \mathbf{U}^{(k+1)} = \mathbf{U}^{(k)} + \Delta \mathbf{U}, \end{cases} \quad k = 0, 1, 2, \dots, \tag{55}$$

where the Jacobian is

$$\frac{\partial \mathbf{G}(\mathbf{U})}{\partial \mathbf{U}} = \mathbf{I} - \alpha \Delta t \frac{\partial \mathbf{R}(\mathbf{U})}{\partial \mathbf{U}}. \tag{56}$$

In practice, the Jacobian matrix is often ill-conditioned, thus a preconditioning technique is required before applying the GMRES inner iteration. A right preconditioning is considered here that can be expressed as

$$\left(\frac{\partial \mathbf{G}(\mathbf{U})}{\partial \mathbf{U}}\Big|_{\mathbf{U}^{(k)}} \mathbf{M}^{-1} \right) (\mathbf{M}\Delta \mathbf{U}) = -\mathbf{G}\left(\mathbf{U}^{(k)}\right), \tag{57}$$

where here $\mathbf{M}$ is a global additive Schwarz preconditioning matrix which has the form

$$\mathbf{M}^{-1} = \sum_{k=1}^{N_b} \mathbf{B}_k^t \mathbf{M}_k^{-1} \mathbf{B}_k, \tag{58}$$

and where $\mathbf{B}_k$ is the operator for the kth block which gathers the solution unknowns for the block from the global solution vector and N_b is the number of grid blocks. The local preconditioning matrix in each block is obtained via an incomplete lower-upper (ILU) factorization, that is,

$$\mathbf{M}_k = \mathbf{L}_k \mathbf{U}_k, \tag{59}$$

which approximates the Jacobian in each block.

As Krylov methods only require Jacobian-vector multiplications, no storage or explicit form of the Jacobian is necessary. Hence, a Jacobian-free approach can be applied. One can invoke the directional derivative such that, for any vector $\mathbf{v}$,

$$\left(\frac{\partial \mathbf{G}(\mathbf{U})}{\partial \mathbf{U}} \mathbf{M}^{-1} \right) \mathbf{v} \approx \mathbf{M}^{-1}\mathbf{v} - \alpha \Delta t \frac{\mathbf{R}(\mathbf{U} + \varepsilon \mathbf{M}^{-1}\mathbf{v}) - \mathbf{R}(\mathbf{U})}{\varepsilon}. \tag{60}$$

The choice of ε follows Northrup[41] and is given by

$$\varepsilon = \frac{\varepsilon_0}{\sqrt{\|\mathbf{v}\|_2}}, \tag{61}$$

where $\varepsilon_0 = 10^{-7}\text{--}10^{-8}$. The tolerances for the Newton and GMRES methods, and row scaling of the linear systems are discussed in detail in the works of Northrup et al.[41,60]

4.5 │ Block ILU preconditioner

In Equation (59), the local ILU preconditioner M_l factorizes an approximate Jacobian of block k, that is,

$$\mathbf{M}_k = \mathbf{L}_k \mathbf{U}_k \approx \tilde{\mathbf{J}}_k \approx \left(\mathbf{I} - \alpha \Delta t \frac{\partial \mathbf{R}}{\partial \mathbf{U}} \right)_k. \tag{62}$$

The Jacobian has both hyperbolic and elliptic components. For the former part, the computation is carried out in a fixed reference frame and then projected back to the original cell face orientation through a rotation matrix $\mathbf{A}$, that is,

$$\frac{\partial \mathcal{F}(\mathbf{U}_L, \mathbf{U}_R, \mathbf{n})}{\partial \mathbf{U}} = \frac{\partial \left(\mathbf{A}^{-1} \mathcal{F}(\mathbf{A}\mathbf{U}_L, \mathbf{A}\mathbf{U}_R)\right)}{\partial \mathbf{U}} = \mathbf{A}^{-1} \frac{\partial \left(\mathcal{F}(\mathbf{A}\mathbf{U}_L, \mathbf{A}\mathbf{U}_R)\right)}{\partial \mathbf{U}}. \tag{63}$$

Then, an approximate Jacobian of the hyperbolic flux can be obtained by invoking the chain rule

$$\mathbf{A}^{-1} \frac{\partial \left(\mathcal{F}(\mathbf{A}\mathbf{U}_L, \mathbf{A}\mathbf{U}_R)\right)}{\partial \mathbf{U}} \approx \mathbf{A}^{-1} \frac{\partial \left(\mathcal{F}(\mathbf{A}\mathbf{U}_L, \mathbf{A}\mathbf{U}_R)\right)}{\partial \mathbf{U}_L} = \mathbf{A}^{-1} \frac{\partial \left(\mathcal{F}(\mathbf{A}\mathbf{U}_L, \mathbf{A}\mathbf{U}_R)\right)}{\partial (\mathbf{A}\mathbf{U}_L)} \mathbf{A}. \tag{64}$$

The derivative here can be computed by a direct differentiation with respect to the first argument of the numerical flux function, for example, the Lax–Friedrichs flux function. More details can be found in References 41,60. The preconditioner terms for the viscous flux can be obtained via differentiation and the chain rule applied to the flux at the cell interfaces as described by Northrup.[41]

5 | NUMERICAL RESULTS

In order to assess the relative performance and efficiencies of the preceding high-order implicit BDF4, ESDIRK4, and ROS34PW2 time-marching schemes used in conjunction with the high-order CENO finite-volume scheme, the application of the combined approaches to several representative CFD-type problems are now examined. Four smooth flow cases, including several stiff problems, are first examined: the inviscid convection of a density pulse; the transport of an isodensity vortex governed by the ideal MHD equations; the transport of stiff resistive MHD waves; and an unsteady viscous flow vortex shedding problem. Two additional nonsmooth flow problems involving shocks and discontinuous solutions are also considered here: the so-called Sod shock tube problem[61] and an ideal MHD version of the Shu–Osher problem.[40,62] For each of these cases, comparisons are also made to predictions obtained using the explicit RK4 method, which serves as a reference. Additionally, the numerical solution errors of the various schemes were evaluated in terms of the exact solution (when available and appropriate) using the following L^1, L^2, and L^∞ norms:

$$\|u^{\text{exact}} - u^{\text{num}}\|_{L^1} = \sum_{i=1}^{N_x} \sum_{j=1}^{N_y} \sum_{k=1}^{N_z} \left| u_{ijk}^{\text{exact}} - u_{ijk}^{\text{num}} \right| \Delta V_{ijk}, \tag{65}$$

$$\|u^{\text{exact}} - u^{\text{num}}\|_{L^2} = \left(\sum_{i=1}^{N_x} \sum_{j=1}^{N_y} \sum_{k=1}^{N_z} \left(u_{ijk}^{\text{exact}} - u_{ijk}^{\text{num}} \right)^2 \Delta V_{ijk} \right)^{1/2}, \tag{66}$$

$$\|u^{\text{exact}} - u^{\text{num}}\|_{L^\infty} = \max_{i,j,k} \left| u_{ijk}^{\text{exact}} - u_{ijk}^{\text{num}} \right|, \tag{67}$$

where u^{exact} and u^{num} are the exact solution and its numerical approximation of one component of the solution vector $\mathbf{U}$, and $\Delta V_{ijk} = (\Delta x \Delta y \Delta z)_{ijk}$ is the volume of cell $\mathcal{I}_{ijk}$, and N_x, N_y, and N_z are the numbers of cells in the x, y, and z directions, respectively. It is also noted that as the BDF4 scheme is not self-starting, the exact solution (again when available) was imposed for the first four time steps in Equation (42) in the numerical validation tests which now follow.

5.1 | Numerical validation 1: Inviscid convection of a density pulse

The first problem considered in the evaluation of the proposed combined time-marching, finite-volume methods is associated with the convection of a prescribed density pulse or variation across a two-dimensional (2D) domain in a background constant pressure inviscid flow by a uniform and constant velocity field. There is no magnetic field in this case and the exact solution of the Euler equations is given by

$$\begin{cases} \rho(\mathbf{x}, t) = \frac{1}{10} \left(1 - \tanh^2(|\bar{\mathbf{x}} - (\mathbf{x} - t\mathbf{v})|)\right), \\ \mathbf{v}(\mathbf{x}, t) = \left(\frac{1}{10}, 1\right), \\ p(\mathbf{x}, t) = 1, \end{cases} \tag{68}$$

where the constant velocity is taken to be $\mathbf{v} = (\frac{1}{10}, 1)$, $\mathbf{x} = (x, y)$, and $\bar{\mathbf{x}} = (-\frac{1}{10}, 0)$ is the initial position of the center of the density pulse. The 2D computational domain is defined by $(x, y) = [-1, 1] \times [-\frac{1}{2}, \frac{1}{2}]$ such that the density pulse is entirely located within the domain. Constant extrapolation of the solution quantities is applied to all domain boundaries and simulations of this problem were run until a maximum time, $T = 0.1$.

To study the convergence order of the various time-stepping schemes, the simulations for the density pulse were performed on a sequence of uniform Cartesian grids with resolutions $N_x = [32, 64, 128, 256, 512, 1024]$, $N_y = [16, 32, 64, 128, 256, 512]$, where N_x, N_y are the numbers of grid cells in the x- and y-direction, respectively. The z-direction was discretized by a single cell so as to make the problem two-dimensional. The time step, Δt, was specified via the CFL condition with a Courant number of $\nu_{\mathrm{CFL}} = 0.50$ for the explicit RK4 method. This value was also adopted in all of the simulations with the implicit time-stepping methods.

The convergence of the predicted numerical errors in terms of L^1, L^2, and L^∞ norms as a function of mesh spacing for each of the time-stepping schemes considered are depicted in Figure 1 and also shown quantitatively in Table 3. It can be seen that all time-stepping schemes, except for the Rosenbrock ROS34PW2 scheme, achieve the desired fourth-order accuracy on the grids with $N_x = 512$ and $N_x = 1024$ cells (shown in bold face text in table). It is interesting to observe that the Rosenbrock scheme achieves third-order accuracy on the grids with $N_x = 256$ and $N_x = 512$ but the accuracy deteriorates for $N_x = 1024$ (shown in italic text in table). Note that reducing the CFL leads to improved convergence (i.e., for $\nu_{\mathrm{CFL}} = 0.125$ the final slope is 2.5), but the time step size for fully accurate results may be overly restrictive. It is also worth mentioning that the approximate Jacobian of Equation (60) is used in place of the exact Jacobian. For the BDF4 and ESDIRK4 schemes, the finite-difference error introduced by the Jacobian-free approach does not affect the scheme's accuracy because the accuracy of the solution from Newton's method is fully determined by the accuracy of the right-hand side terms in Equation (55), and is not influenced by the accuracy of the Jacobian, given the fixed-point nature of Newton's method. Rosenbrock methods are generally sensitive to the accuracy of the Jacobian since they are no longer fixed-point methods, due to the linearization process, but Rosenbrock-Wanner-type schemes such as the ROS34PW2 method of Equation (52) were especially derived to be less sensitive to the accuracy of the Jacobian, so the deterioration of the ROS34PW2 error in Figure 1 for $N_x = 1024$ is not clear.

5.2 | Numerical validation 2: Isodensity vortex problem for ideal MHD

The next problem considered is the 3D isodensity vortex governed by the ideal MHD equations as previously studied by Ivan et al.[30] Numerical results for the various time-marching schemes combined with the CENO method are compared with the exact solution for this case, which is given by

$$
\begin{cases}
\rho(\mathbf{x}, t) = 1, \\
\mathbf{v}(\mathbf{x}, t) = 1 + (-y, x, 0)\kappa \exp[q(1 - r^2)], \\
\mathbf{B}(\mathbf{x}, t) = (-y, x, 0)\mu \exp[q(1 - r^2)], \\
p(\mathbf{x}, t) = 1 + \frac{1}{4q}(\mu^2(1 - 2qr^2) - \kappa^2\rho) \exp[2q(1 - r^2)],
\end{cases}
\tag{69}
$$

where $\kappa = \mu = \frac{1}{2\pi}$, $q = \frac{1}{10}$, and $r^2 = x^2 + y^2$ and $\mathbf{x} = (x, y, z)$ is defined for the computational domain $\Omega = [-5, 5]^3$. Uniform Cartesian meshes were used in all of the simulations of this case and periodic conditions were applied to all boundaries. Predicted solutions were obtained to a maximum time of $T = 1$.

An illustration of the numerical solution of the magnetic field in the x-direction, B_x, at $T = 1$ is provided in Figure 2 for the isodensity ideal MHD problem obtained using the RK4 scheme on a $64 \times 64 \times 64$ grid. Furthermore, the errors in the predicted solution on 3D Cartesian meshes with uniform spacing grids and $N_x = N_y = N_z \equiv N$ cells in each coordinate direction are depicted in Figure 3 as well as listed in Table 4 for $N = [16, 32, 64, 128]$ as a function of N_x. It is evident that the desired fourth-order accuracy is achieved for RK4, BDF4, and ESDIRK4 from grid size $N = 64$ cells. However, the ROS34PW2 scheme suffered from robustness issues with negative densities and pressures for this nonlinear compressible MHD problem and, as such, results for this approach have been omitted. For this reason, as well as the accuracy issues identified in the first validation problem, the ROS34PW2 scheme was not assessed further as part of the current study.
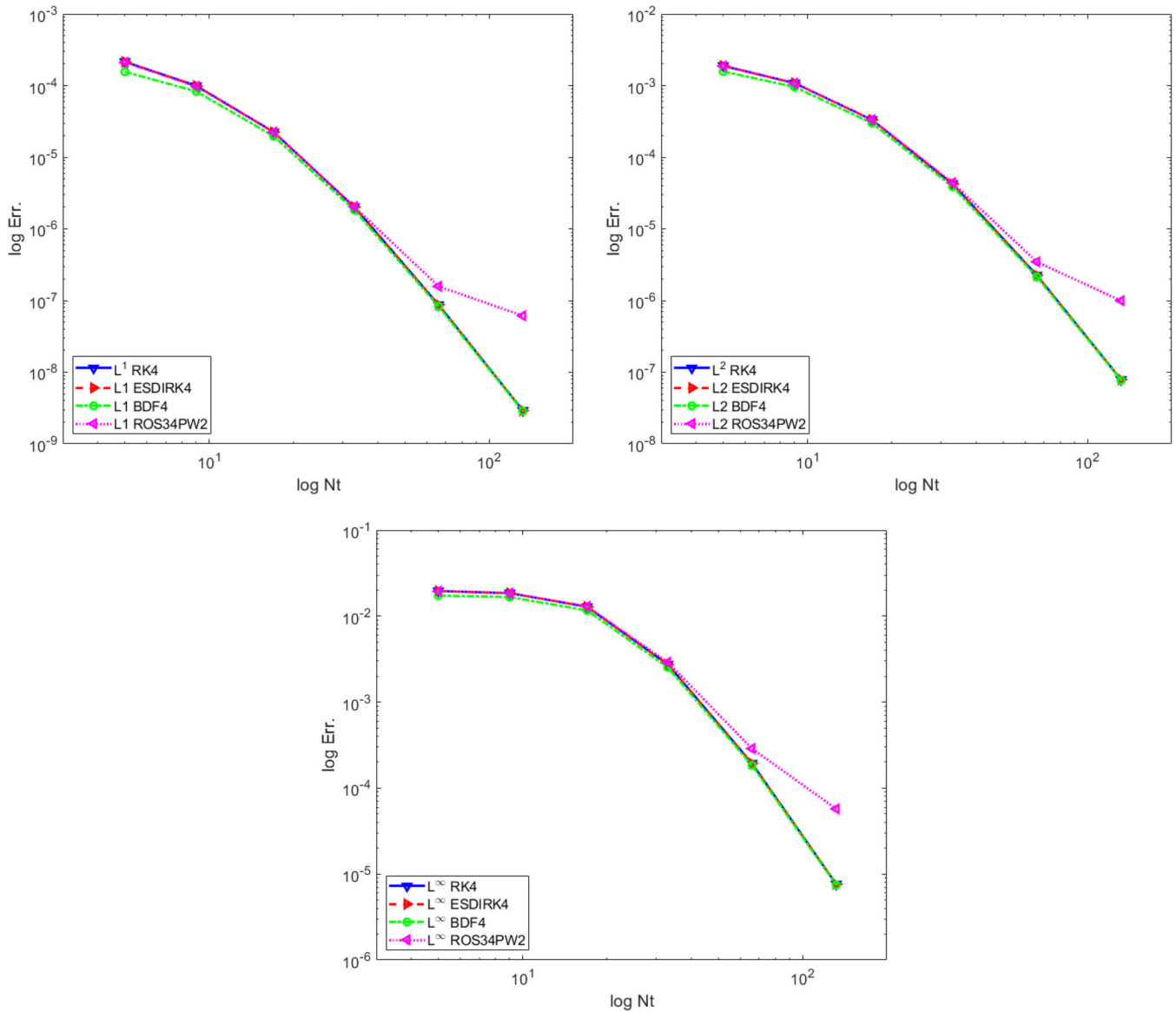
FIGURE 1 Numerical validation 1: Comparison of the errors in the predicted solution at final time $T = 0.1$ as a function of mesh size for inviscid convection of a density pulse on a 2D domain governed by the Euler equations. The errors are measured in (left) L^1 norm, (right) L^2 norm, and (bottom) L^∞ norm [Colour figure can be viewed at wileyonlinelibrary.com]

5.3 | Numerical validation 3: Transport of 1D stiff resistive MHD wave

The efficiencies of the explicit and implicit time-stepping methods coupled to the fourth-order CENO finite-volume scheme are now investigated when applied to the solution of a one-dimensional (1D) manufactured resistive MHD problem governed by the resistive MHD equations and having the following solution:

$$\begin{cases} \rho(\mathbf{x}, t) = p(\mathbf{x}, t) = 100, \\ \mathbf{v}(\mathbf{x}, t) = 0, \\ \mathbf{B}(\mathbf{x}, t) = (Bx, By, Bz) = (\cos(2\pi(z - t)), 0, 0), \end{cases} \tag{70}$$

where $(x, y, z) = [-4\Delta z, 4\Delta z] \times [-4\Delta z, 4\Delta z] \times [-\frac{1}{2}, \frac{1}{2}]$ and $\Delta z = 1/N_z$ where N_z defines the extent of the computational domain for the problem and represents the number of computational cells in z direction. The solution given in Equation (70) is obtained by adding a source term to the resistive MHD equations (1)–(4). Thanks to this manufactured

TABLE 3 Numerical validation 1: Convergence of errors in the predicted solution at final time $T = 0.1$ for inviscid convection of a density pulse on a 2D domain governed by the Euler equations.

N_x	L^1 err.	L^1 order	L^2 err.	L^2 Order	L^∞ err.	L^∞ order
RK4						
32	$2.1492 \, (10)^{-4}$	–	$1.8878 \, (10)^{-3}$	–	$1.9584 \, (10)^{-2}$	–
64	$9.9073 \, (10)^{-5}$	1.1172	$1.0816 \, (10)^{-3}$	0.8352	$1.8502 \, (10)^{-2}$	0.0820
128	$2.2459 \, (10)^{-5}$	2.1412	$3.3313 \, (10)^{-4}$	1.6991	$1.2837 \, (10)^{-2}$	0.5274
256	$2.0229 \, (10)^{-6}$	3.4728	$4.2379 \, (10)^{-5}$	2.9747	$2.7498 \, (10)^{-3}$	2.2229
512	$\mathbf{8.7109 \, (10)^{-8}}$	**4.5375**	$\mathbf{2.2453 \, (10)^{-6}}$	**4.2384**	$\mathbf{1.9337 \, (10)^{-4}}$	**3.8299**
1024	$\mathbf{2.9235 \, (10)^{-9}}$	**4.8970**	$\mathbf{7.8904 \, (10)^{-8}}$	**4.8307**	$\mathbf{7.6329 \, (10)^{-6}}$	**4.6630**
BDF4						
32	$1.5550 \, (10)^{-4}$	–	$1.5754 \, (10)^{-3}$	–	$1.7341 \, (10)^{-2}$	–
64	$8.2853 \, (10)^{-5}$	0.9083	$9.5447 \, (10)^{-4}$	0.7230	$1.6650 \, (10)^{-2}$	0.0586
128	$1.9658 \, (10)^{-5}$	2.0754	$2.9927 \, (10)^{-4}$	1.6732	$1.1653 \, (10)^{-2}$	0.5148
256	$1.8367 \, (10)^{-6}$	3.4200	$3.8869 \, (10)^{-5}$	2.9448	$2.5354 \, (10)^{-3}$	2.2005
512	$\mathbf{8.2934 \, (10)^{-8}}$	**4.4690**	$\mathbf{2.1401 \, (10)^{-6}}$	**4.1829**	$\mathbf{1.8425 \, (10)^{-4}}$	**3.7824**
1024	$\mathbf{2.8601 \, (10)^{-9}}$	**4.8578**	$\mathbf{7.7189 \, (10)^{-8}}$	**4.7931**	$\mathbf{7.4641 \, (10)^{-6}}$	**4.6256**
ESDIRK4						
32	$2.1494 \, (10)^{-4}$	–	$1.8881 \, (10)^{-3}$	–	$1.9587 \, (10)^{-2}$	–
64	$9.9073 \, (10)^{-5}$	1.1173	$1.0816 \, (10)^{-3}$	0.8037	$1.8502 \, (10)^{-2}$	0.0822
128	$2.2459 \, (10)^{-5}$	2.1412	$3.3313 \, (10)^{-4}$	1.6991	$1.2837 \, (10)^{-2}$	0.5274
256	$2.0229 \, (10)^{-6}$	3.4728	$4.2379 \, (10)^{-5}$	2.9747	$2.7498 \, (10)^{-3}$	2.2229
512	$\mathbf{8.7109 \, (10)^{-8}}$	**4.5375**	$\mathbf{2.2453 \, (10)^{-6}}$	**4.2384**	$\mathbf{1.9337 \, (10)^{-4}}$	**3.8299**
1024	$\mathbf{2.9235 \, (10)^{-9}}$	**4.8970**	$\mathbf{7.8904 \, (10)^{-8}}$	**4.8307**	$\mathbf{7.6329 \, (10)^{-6}}$	**4.6630**
ROS34PW2						
32	$2.1023 \, (10)^{-4}$	–	$1.8750 \, (10)^{-3}$	–	$1.9492 \, (10)^{-2}$	–
64	$9.7440 \, (10)^{-5}$	1.1094	$1.0738 \, (10)^{-3}$	0.8041	$1.8459 \, (10)^{-2}$	0.0785
128	$2.2315 \, (10)^{-5}$	2.1265	$3.3340 \, (10)^{-4}$	1.6875	$1.2956 \, (10)^{-2}$	0.5107
256	$\mathbf{2.0397 \, (10)^{-6}}$	**3.4516**	$\mathbf{4.3980 \, (10)^{-5}}$	**2.9224**	$\mathbf{2.9044 \, (10)^{-3}}$	**2.1574**
512	$\mathbf{1.5804 \, (10)^{-7}}$	**3.6900**	$\mathbf{3.4648 \, (10)^{-6}}$	**3.6660**	$\mathbf{2.8809 \, (10)^{-4}}$	**3.3336**
1024	$\mathit{6.1272 \, (10)^{-8}}$	*1.3670*	$\mathit{9.9394 \, (10)^{-7}}$	*1.8015*	$\mathit{5.7283 \, (10)^{-5}}$	*2.3304*

source term, the sinusoidal varying MHD solution is not damped out by the diffusivity, but rather propagates with a slow speed. There exists two widely different time scales for the problem: a fast diffusion scale dictating the numerical stability when conditionally stable time-marching schemes are used to obtain solutions, and a slow wave propagation scale which determines the accuracy of the numerical simulation. As such, this manufactured solution is a stiff problem. In order to examine the stiffness, the fast MHD wavespeed can be approximated as

$$c_f \approx \sqrt{\gamma} = 1.2923, \tag{71}$$

where the adiabatic constant $\gamma = 1.67$ for the monotonic charged molecules of the plasma. As noted above, this problem exhibits two distinct time scales. The slow time scale corresponds to the MHD wavespeed $c_f = \mathcal{O}(1)$, whose required time step, Δt_S, determines the simulation accuracy of the wave in Equation (70). On the other hand, there is a fast diffusion (resistivity) time scale in which the time step, Δt_F, depends on the resistivity, η, and this fast time scale determines the numerical stability. To make the problem stiff, the resistivity is taken to be large with a value of $\eta = 100$. For $N_z = 32$,
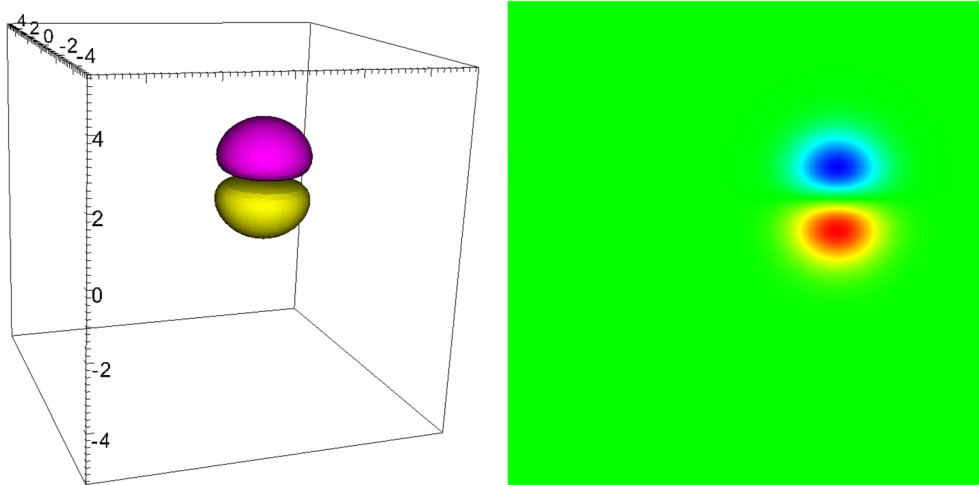
FIGURE 2 Numerical validation 2: Predictions of (left) the 3D isosurfaces of x-component of the magnetic field, B_x; and (right) 2D contour plot of x-component of the magnetic field, B_x, across the vortex of the isodensity vortex problem of ideal MHD for a grid size $N = 64$ and final time $T = 1$ [Colour figure can be viewed at wileyonlinelibrary.com]

$\Delta z = 1/32$ and the required time steps for the two disparate time scales can then be estimated as follows:

$$
\begin{cases}
\Delta t_S \approx \dfrac{\Delta z}{c_f} = 2.4206 \ (10)^{-2}, \\
\Delta t_F \approx \dfrac{\Delta z^2}{C_{\text{diff}} \eta} = 1.2207 \ (10)^{-6},
\end{cases}
\tag{72}
$$

with

$$
\varepsilon_{\Delta t} = \frac{\Delta t_S}{\Delta t_F} = 1.9830 \ (10)^4,
\tag{73}
$$

where $\varepsilon_{\Delta t}$ is the time scale disparity ratio. The coefficient, C_{diff}, depends on the problem dimension. Here, for a 3D simulation, $C_{\text{diff}} = 8$. For numerical stability, the CFL time step restriction for the explicit RK4 scheme depends on the fast time scale, that is, $\Delta t_{\text{EX}} = \mathcal{O}(\Delta t_F) = \mathcal{O}(10^{-6})$, whereas a much larger time step $\Delta t_{\text{IM}} = \mathcal{O}(\Delta t_S) = \mathcal{O}(10^{-2})$ can be used for the implicit time-stepping methods.

The resistive MHD problem settings for the simulations performed here are summarized in Table 5. It is noted that the number of time steps for all implicit schemes was fixed to be 20 for all grid resolutions considered. In order to arrive at the appropriate choices for the convergence tolerances using in the Newton and GMRES iterative schemes of the implicit time-stepping solution methods, parametric studies were performed at each grid resolution. Optimal tolerances for each iterative scheme were selected in terms of producing reducing the simulation time while preserving solution accuracy. This provides a fair comparison between the best case for each algorithm, but may, of course, not be practical in real application settings. For the current stiff resistive MHD problem, the optimal values for the Newton and GMRES tolerances are given in Table 6. It is worth mentioning that since an inexact Newton's method is used here, the GMRES tolerance can be chosen to be much less stringent than the Newton tolerance.

The numerical distribution of the x-component of the magnetic field, B_x, is provided in Figure 4 as obtained using the explicit RK4 scheme on the regular Cartesian mesh with $N_z = 32$. The sinusoidal nature of the resistive MHD solutions is clearly shown. Moreover, in Figure 5, the convergence of the predicted solution errors for the explicit RK4 as well as implicit BDF4 and ESDIRK4 schemes is depicted as a function of N_z. The L^1, L^2, and L^∞ norms of the solution error for the RK4, ESDIRK, and BDF4 time-marching schemes are all shown and one can see that all of the methods provide virtually equivalent accuracy and converge at the same rate. As $\Delta t_{\text{EX}} \ll \Delta z$, the solution error is indeed determined by the spatial error associated with each grid resolution and not the time marching errors.

The efficiency of the various time-marching schemes applied to the resistive MHD problem, in terms of total computation time, is summarized in Figure 6 (left panel), where the computational time is given as a function of the mesh size
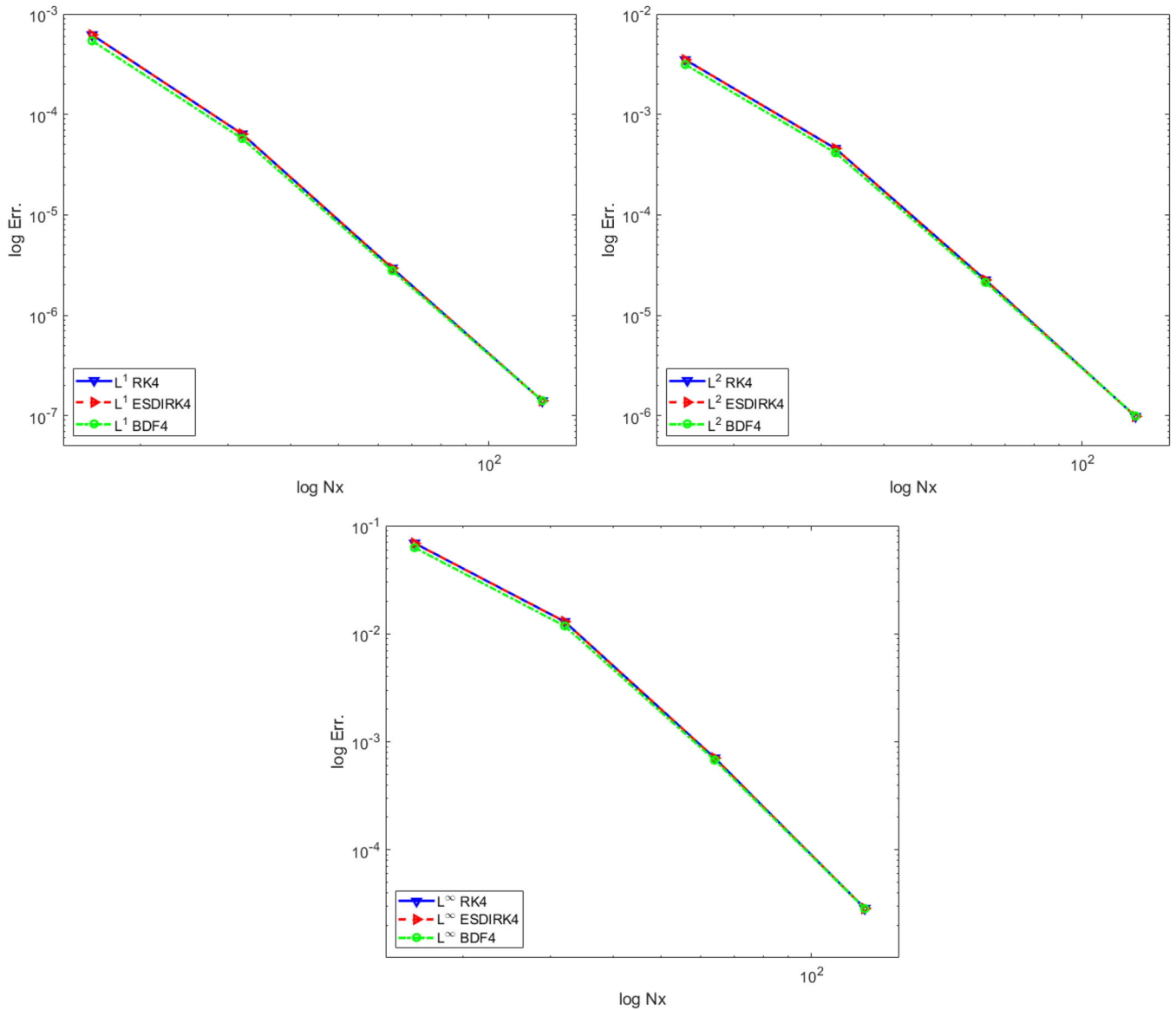
FIGURE 3 Numerical validation 2: Comparison of the errors in the predicted solution for B_x at final time $T = 1$ for the isodensity vortex problem of ideal MHD. The errors are measured in (left) L^1 norm, (right) L^2 norm, and (bottom) L^∞ norm [Colour figure can be viewed at wileyonlinelibrary.com]

parameter N_z. The results of this figure clearly show that the implicit ESDIRK4 and BDF4 schemes require much less computational effort than the explicit RK4 method in order to achieve the same level of accuracy. Defining the speedup factor, S, of the implicit schemes as compared with the explicit scheme RK4 for each grid resolution as

$$S = \frac{\mathcal{T}_{EX}}{\mathcal{T}_{IM}}, \tag{74}$$

where $\mathcal{T}_{EX}$ is the computational (CPU) time or cost for the explicit RK4 method and $\mathcal{T}_{EX}$ is the corresponding computational time for the implicit method, the speedup factors of the ESDIRK4 and BDF4 schemes are then shown in Figure 6 (right panel). From the latter, it can be observed that the maximal speedup for the ESDIRK4 scheme is obtained at grid size $N_z = 32$ where $S \approx 166$ and the minimal speedup occurs at grid size $N_z = 64$ with $S \approx 48$, whereas for the BDF4 scheme, the maximal speedup is achieved at grid size $N_z = 16$ with $S \approx 148$ and it is minimal at grid size $N_z = 64$ with $S \approx 83$. It is also noted that, although the ESDIRK4 scheme involves the solution of a greater number of nonlinear systems than the BDF4 scheme at each time step, overall the former is more efficient, except for grid size $N_z = 64$.

TABLE 4 Numerical validation 2: Convergence of errors in the predicted solution for B_x at final time $T = 1$ for the isodensity vortex problem of ideal MHD

N	L^1 err.	L^1 order	L^2 err.	L^2 order	L^∞ err.	L^∞ order
RK4						
16	$6.1969 (10)^{-4}$	–	$3.4995 (10)^{-3}$	–	$6.9021 (10)^{-2}$	–
32	$6.3773 (10)^{-5}$	3.2789	$4.5803 (10)^{-4}$	2.9322	$1.2959 (10)^{-2}$	2.4119
64	**$2.9413 (10)^{-6}$**	**4.4378**	**$2.2458 (10)^{-5}$**	**4.3495**	**$7.0752 (10)^{-4}$**	**4.1944**
128	**$1.4075 (10)^{-7}$**	**4.3850**	**$9.8378 (10)^{-7}$**	**4.5126**	**$2.8628 (10)^{-5}$**	**4.6271**
BDF4						
16	$5.3993 (10)^{-4}$	–	$3.1345 (10)^{-3}$	–	$6.2740 (10)^{-2}$	–
32	$5.7601 (10)^{-5}$	3.2270	$4.1528 (10)^{-4}$	2.9147	$1.1812 (10)^{-2}$	2.4080
64	**$2.7855 (10)^{-6}$**	**4.3695**	**$2.1302 (10)^{-5}$**	**4.2844**	**$6.7687 (10)^{-4}$**	**4.1246**
128	**$1.4105 (10)^{-7}$**	**4.3035**	**$9.9312 (10)^{-7}$**	**4.4227**	**$2.8508 (10)^{-5}$**	**4.5693**
ESDIRK4						
16	$6.1980 (10)^{-4}$	–	$3.4998 (10)^{-3}$	–	$6.9037 (10)^{-2}$	–
32	$6.3775 (10)^{-5}$	3.2792	$4.5806 (10)^{-4}$	2.9323	$1.2962 (10)^{-2}$	2.4119
64	**$2.9419 (10)^{-6}$**	**4.4375**	**$2.2466 (10)^{-5}$**	**4.3491**	**$7.0806 (10)^{-4}$**	**4.1937**
128	**$1.4083 (10)^{-7}$**	**4.3846**	**$9.8476 (10)^{-7}$**	**4.5117**	**$2.8660 (10)^{-5}$**	**4.6266**

TABLE 5 Numerical validation 3: Simulation settings for stiff resistive MHD wave transport problem: T, final time; Δt_{EX}, CFL stable time step for the explicit RK4 scheme; Δt_{IM}, time step for the implicit schemes; and N_t, total number of time steps

N_z	T	Δt_{EX}	$N_{t,EX}$	Δt_{IM}	$N_{t,IM}$
8	0.8	$9.77 (10)^{-6}$	81,921	$4.00 (10)^{-2}$	20
16	0.4	$2.44 (10)^{-6}$	163,841	$2.00 (10)^{-2}$	20
32	0.2	$6.10 (10)^{-7}$	327,680	$1.00 (10)^{-2}$	20
64	0.1	$1.53 (10)^{-7}$	655,360	$5.00 (10)^{-3}$	20

TABLE 6 Numerical validation 3: Optimal Newton and GMRES convergence tolerances used in simulation of stiff resistive MHD wave transport problem

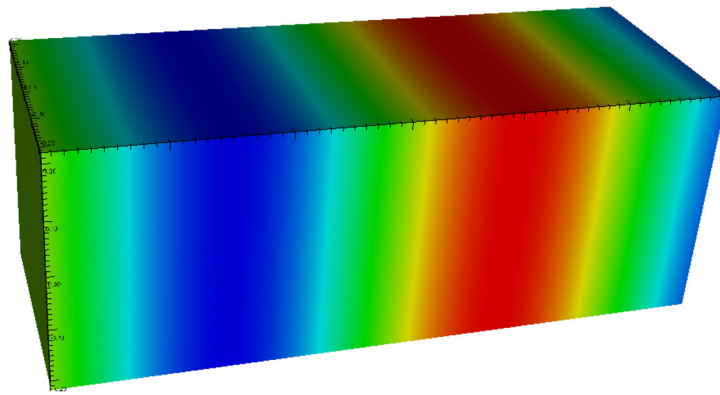| N_z	ESDIRK4 Newton	ESDIRK4 GMRES	BDF4 Newton	BDF4 GMRES
8	10^{-3}	10^{-1}	10^{-4}	10^{-2}
16	10^{-4}	10^{-1}	10^{-5}	10^{-2}
32	10^{-5}	10^{-1}	10^{-6}	10^{-2}
64	10^{-6}	10^{-2}	10^{-7}	10^{-2}

FIGURE 4 Numerical validation 3: Predicted distribution of the *x*-component of the magnetic field, B_x for stiff resistive MHD wave transport problem obtained using the explicit RK4 scheme on a Cartesian mesh with $N_z = 32$ [Colour figure can be viewed at wileyonlinelibrary.com]
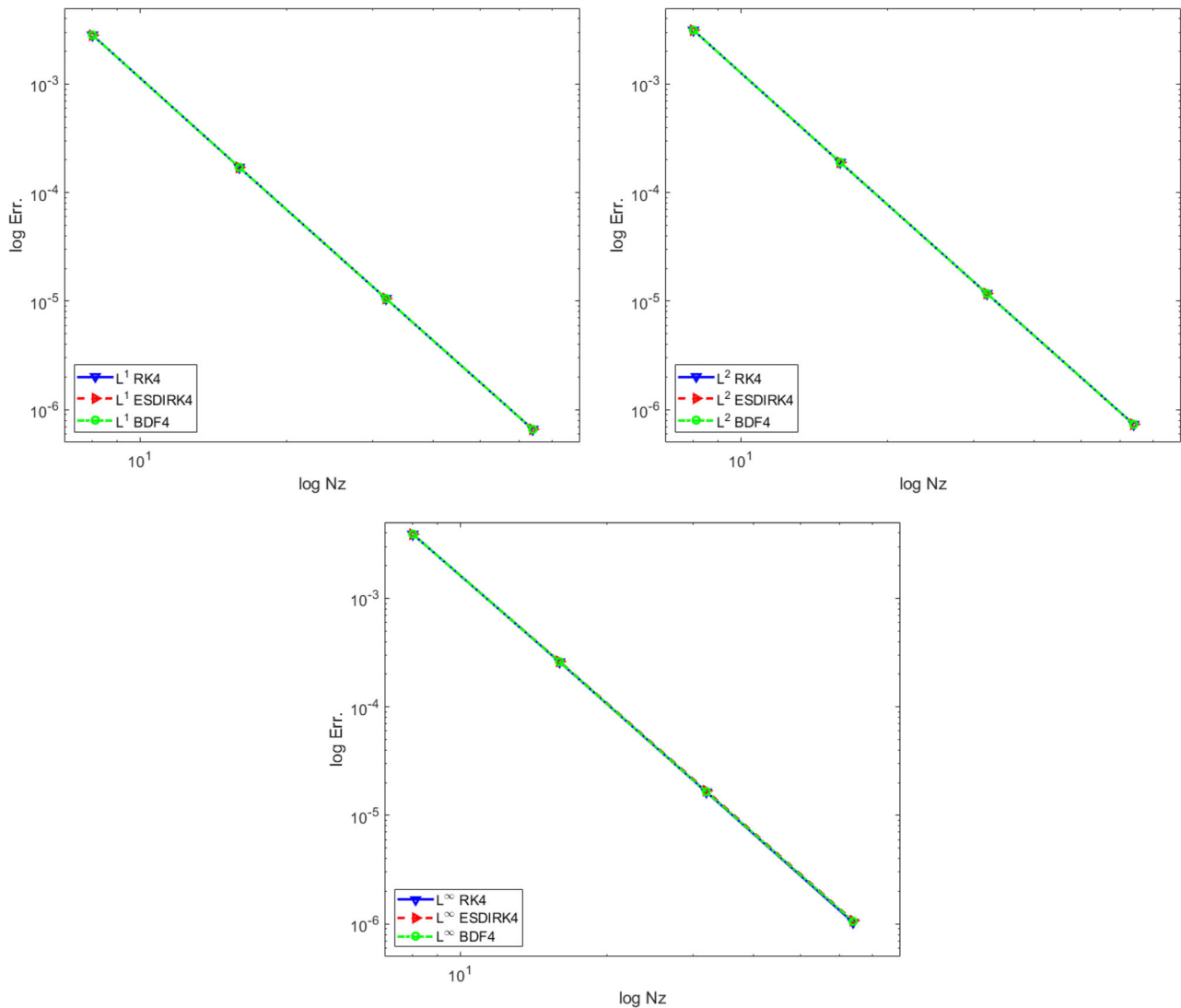

FIGURE 5 Numerical validation 3: Comparison of the errors in the predicted solution as a function of the mesh size parameter, N_z, for stiff resistive MHD wave transport problem. The errors are measured in (left) L^1 norm, (right) L^2 norm, and (bottom) L^∞ norm [Colour figure can be viewed at wileyonlinelibrary.com]
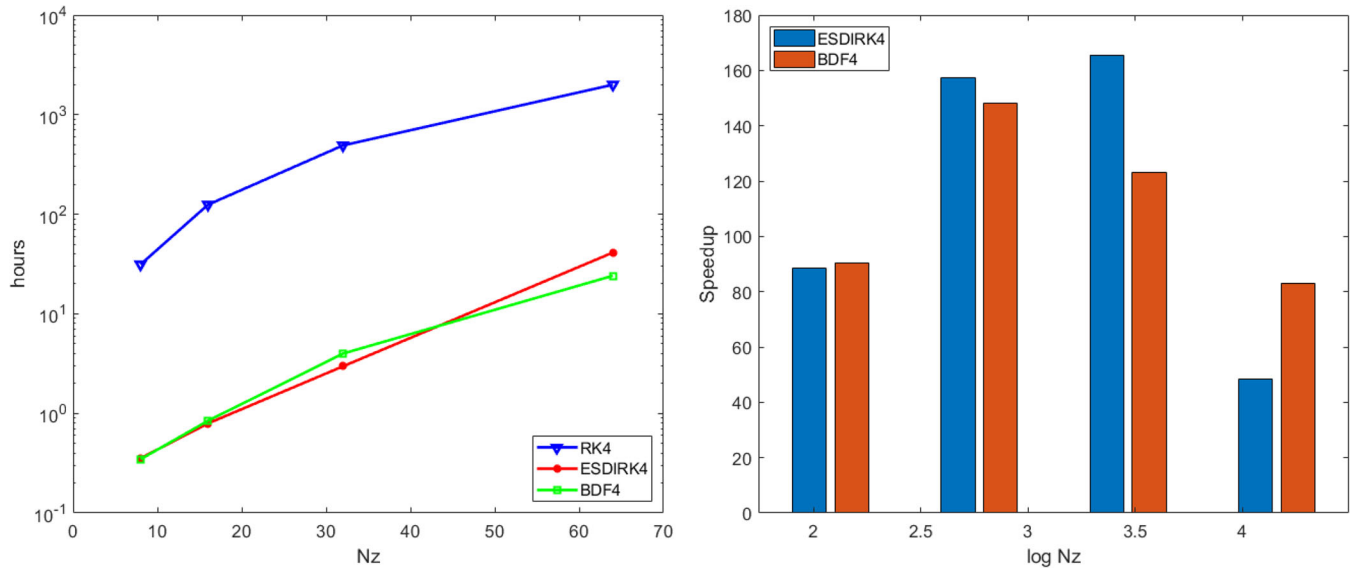
FIGURE 6 Numerical validation 3: Computational cost (left) and speedup factor, S, (right) as a function of the mesh size parameter N_z for stiff resistive MHD wave transport problem [Colour figure can be viewed at wileyonlinelibrary.com]
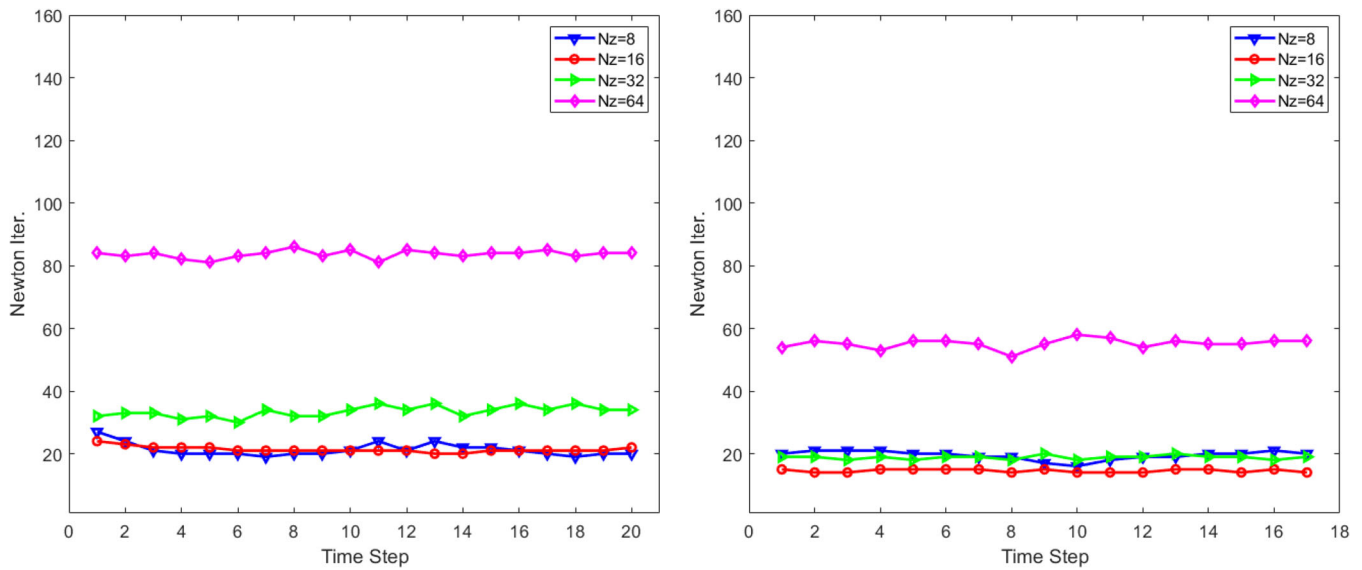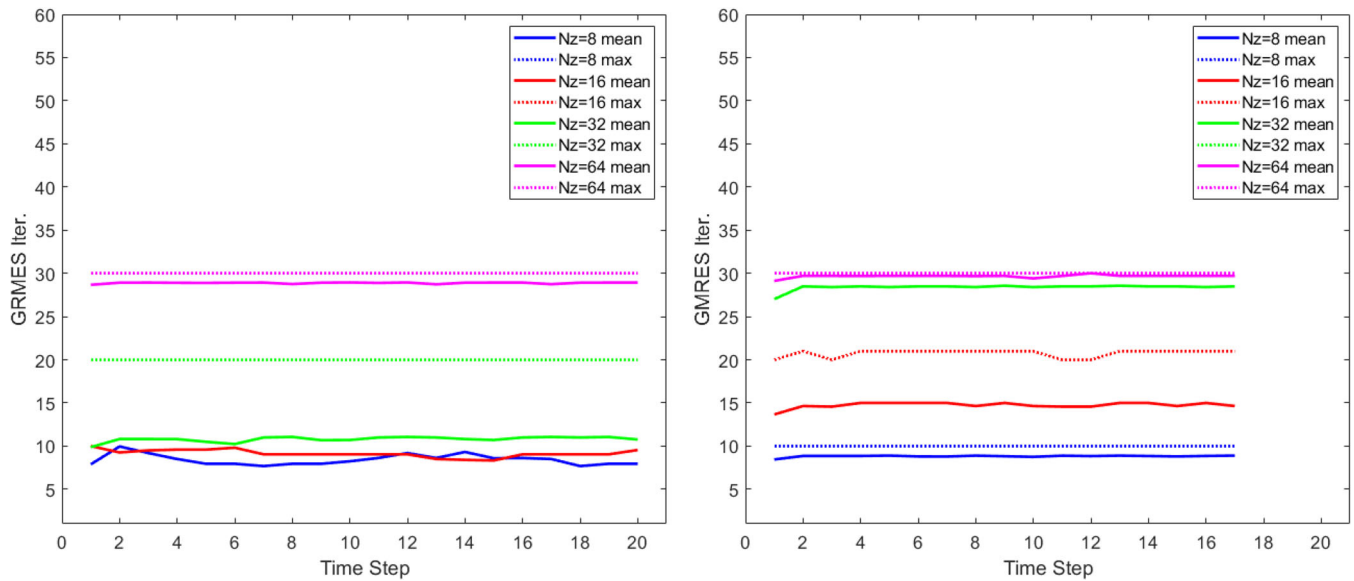

FIGURE 7 Numerical validation 3: Total number of Newton iterations per time step required by ESDIRK4 scheme (left) and BDF4 scheme (right) in the solution of the stiff resistive MHD wave transport problem [Colour figure can be viewed at wileyonlinelibrary.com]

To understand the performance of the nonlinear Newton iterative solution scheme, the total number of Newton and GMRES iterations at each time step for each implicit scheme has been investigated and the findings summarized in Figures 7 and 8, respectively. Recall that at each time step, the fourth-order ESDIRK4 scheme solves a sequence of five nonlinear systems of size $m \times m$ where m is the number of unknown cell centers, whereas the fourth-order BDF4 solves only one such nonlinear system. This explains why ESDIRK4 requires more Newton iterations than BDF4. In exchange, thanks to a better initial guess at each stage, ESDIRK4 requires far fewer GMRES iterations than BDF4 for the inner linear solves as shown in Figure 8. This results in the overall cost of the ESDIRK4 scheme being less than or comparable to that of the BDF4 method, except for the case $N_z = 64$, where the number of GMRES iterations for both schemes is similar.

FIGURE 8 Numerical validation 3: Total number of GMRES iterations per time step required by ESDIRK4 scheme (left) and BDF4 scheme (right) in the solution of the stiff resistive MHD wave transport problem [Colour figure can be viewed at wileyonlinelibrary.com]

5.4 | Numerical validation 4: Unsteady viscous vortex shedding

Further assessment of the predictive capabilities of the proposed high-order implicit schemes used in conjunction with the CENO finite-volume scheme has been carried here by considering the prediction of the unsteady, viscous, vortex shedding associated with subsonic flow past a circular cylinder. This vortex shedding problem was previously examined by Ivan and Groth[28] and is governed by the Navier–Stokes equations. For the particular case of interest, air is the working gas and the Reynolds number, Re, and Mach number, M_∞, based on free-stream conditions are $Re = 110$ and $M_\infty = 0.1$, respectively. The Strouhal number, St, based on experimental data is $St = 0.1711$ and the mean value of the drag coefficient, C_D, is expected to be close to 1.34.[63] The radius of the cylinder is $d_i = 0.0001$ m. For the simulations, a cylindrical computational domain was used which is comprised of the cylindrical-shaped volume between two cylinders of diameter $d_i = 0.0001$ m and $d_o = 0.004$ m and extruded in the third direction over a length of $L = 0.002$ m. The latter places the outer boundary about 40 diameters away from the inner boundary of the cylinder. The initial and boundary conditions used here are similar to those described by Ivan and Groth.[28]

A quasi-steady periodic solution with periodic vortex shedding leading to the formation of a so-called a Kármán vortex street was first obtained using the explicit RK4 scheme on a body-fitted O-grid mesh consisting of 1024 self-similar grid blocks, each containing $8 \times 16 \times 4$ cells, having a total of 524,288 computational cells. Figure 9 depicts the predicted density distribution, showing the periodic shedding of the vortices, obtained at the end of this initial simulation period at $t = 6\ (10)^{-5}$ s. All subsequent simulations were then restarted from this initial physical time of $t_0 = 6\ (10)^{-5}$ s and the computations were continued for a total computational time of 24 h on 1024 processors with a number of different configurations for the restarted simulations. For one set of simulations, the ESDIRK4 scheme was used with four different values for the size of the time step: $\Delta t = 1\ (10)^{-7}$, $\Delta t = 1\ (10)^{-8}$, $\Delta t = 1\ (10)^{-9}$, and $\Delta t = 1\ (10)^{-10}$. In the fifth simulation, the explicit RK4 scheme was used with the time step satisfying both the CFL condition and Neumann conditions for the viscous terms and having a value of $1\ (10)^{-10}$. Finally, the BDF4 time marching scheme was subsequently used in simulations with time steps of $\Delta t = 5\ (10)^{-9}$, $\Delta t = 1\ (10)^{-9}$, $\Delta t = 5\ (10)^{-10}$, and $\Delta t = 1\ (10)^{-10}$.

The predicted unsteady drag coefficient, C_D, as a function of time obtained using the reference explicit RK4 scheme is depicted in Figure 10 (top panel, orange colored curve). The numerical mean drag is 1.346 which is in very good agreement with References 28,63. The period or time length between any two peaks is around $1.64\ (10)^{-5}$ s. This ultimately constrains the size of the time step that may be used with the implicit time-marching schemes for accurate predictions. The largest time step for ESDIRK4, $\Delta t = 1\ (10)^{-7}$, provides about 86 time steps to describe the period of oscillations for the drag, which of course is associated with the vortex shedding. The predicted temporal variations of C_D for the ESDIRK4 and RK4 simulations are also given in Figure 10 (top panel, blue, green, red, black and orange colored curves, respectively). The
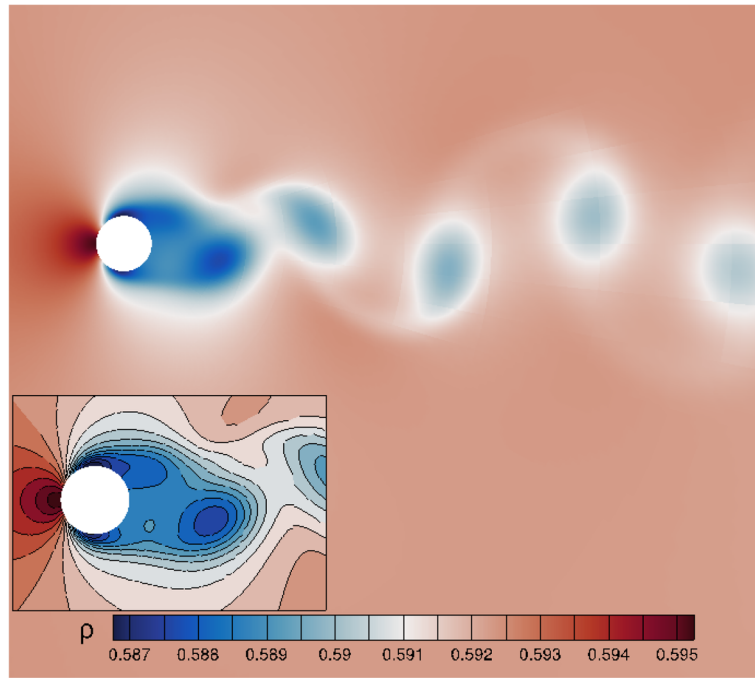
FIGURE 9 Numerical validation 4: Predicted instantaneous distribution of the density, ρ, at $t = 1.20(10)^{-4}$ s for unsteady, subsonic, laminar flow past a circular cylinder with a free-stream Mach number of $M_\infty = 0.1$ and Reynolds number of $Re = 110$ obtained using the fourth-order, CENO, finite-volume method combined with the explicit RK4 time marching scheme [Colour figure can be viewed at wileyonlinelibrary.com]
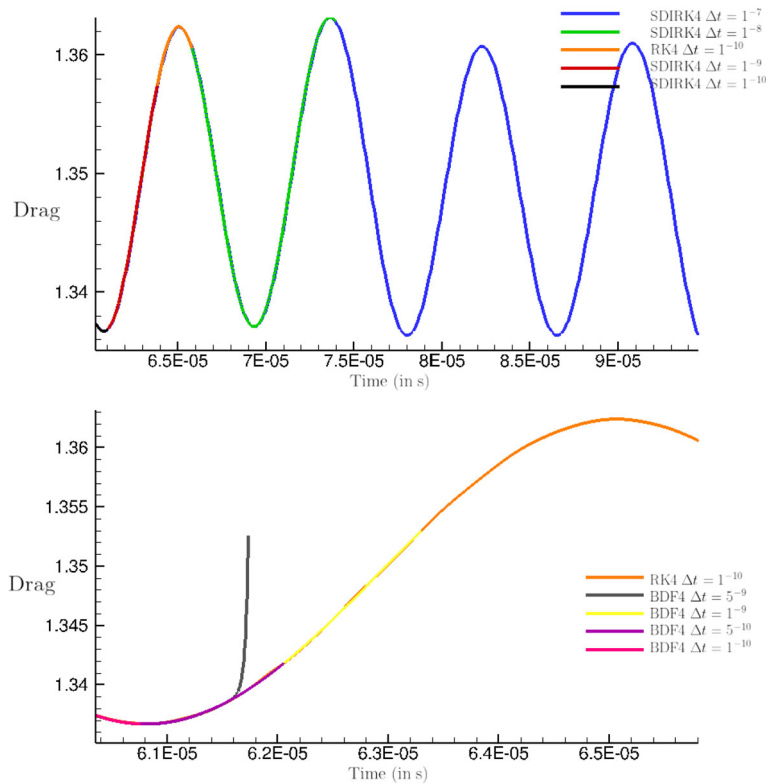

FIGURE 10 Numerical validation 4: (top) Predicted time histories of the unsteady drag coefficient, C_D, obtained using the ESDIRK4 scheme for the time step $\Delta t = 1 \ (10)^{-7}$ (blue), $\Delta t = 1 \ (10)^{-8}$ (green), $\Delta t = 1 \ (10)^{-9}$ (red), and $\Delta t = 1 \ (10)^{-10}$ (black) compared with the reference solution of the RK4 method (orange); (bottom) predicted time histories of the unsteady drag coefficient, C_D, obtained using the BDF4 scheme for $\Delta t = 5 \ (10)^{-9}$ (gray), $\Delta t = 1 \ (10)^{-9}$ (blue), $\Delta t = 5 \ (10)^{-10}$ (green), and $\Delta t = 1 \ (10)^{-10}$ (red) compared with the reference solution of the RK4 method (orange) [Colour figure can be viewed at wileyonlinelibrary.com]

time histories of this figure clearly demonstrate that the implicit ESDIRK4 scheme outperforms the explicit RK4 method, in terms of the total length of simulated time for a fixed computational cost (i.e., $24 \times 1024 = 24,576$ CPU core hours), for the two larger time steps $\Delta t = 1 \ (10)^{-8}$ (green) and $\Delta t = 1 \ (10)^{-7}$ (blue). Additional predicted time histories of the drag coefficient for the BDF4 simulations are shown in Figure 10 (bottom panel, gray, yellow, purple, and pink colored curves). It is evident that, for time steps larger than $1 \ (10)^{-9}$, the BDF4 scheme proved to be unstable and the drag coefficient diverges from the expected periodic behavior. The divergent results correspond to the gray curve of BDF4 - $\Delta t = 5 \ (10)^{-9}$ shown in Figure 10 (bottom). For a time step equal to that used in the RK4 simulation, $\Delta t = 1 \ (10)^{-10}$, and up to a time step of $\Delta t = 1 \ (10)^{-9}$, the BDF4 scheme was found to be stable for cylinder flow problem.

The efficiency, $\mathcal{E}$, of the ESDIRK4 and BDF4 implicit time-marching schemes, relative to the explicit RK4 method, is estimated herein using

$$\mathcal{E} = \frac{t_{\text{end}}^S - t_0}{t_{\text{end}}^{\text{RK4}} - t_0}, \tag{75}$$

where t_{end}^S is the final physical time obtained for scheme S at the end of the 24-h simulation. The resulting efficiencies for the ESDIRK4 simulations are $\mathcal{E}_{\text{ESDIRK4}-\Delta t=1(10)^{-7}} = 6.82$, $\mathcal{E}_{\text{ESDIRK4}-\Delta t=1(10)^{-8}} = 2.49$, $\mathcal{E}_{\text{ESDIRK4}-\Delta t=1(10)^{-9}} = 0.63$, and $\mathcal{E}_{\text{ESDIRK4}-\Delta t=1(10)^{-10}} = 0.12$, respectively. Conversely, the accuracy, $\mathcal{A}$, of the implicit time-marching schemes for the vortex shedding simulations, relative to the RK4 method, can also be estimate in term of the computed unsteady drag coefficient for the cylinder

$$\mathcal{A} = \frac{\left\| C_D^S - C_D^{\text{RK4}} \right\|_1}{\left\| C_D^{\text{RK4}} \right\|_1}, \tag{76}$$

where $||C_D^S - C_D^{\text{RK4}}||_1$ is the L_1-norm such that

$$\left\| C_D^S - C_D^{\text{RK4}} \right\|_1 = \frac{(t_1 - t_0)}{N} \sum_{i=0}^{N} \left| C_D^{S^i} - C_D^{\text{RK4}^i} \right|, \tag{77}$$

where $t_1 = \min(t_{\text{end}}^S, t_{\text{end}}^{\text{RK4}})$, N is the number of points between t_0 and t_1 in the RK4 temporal grid and C_D^S is the drag of the implicit scheme, S, interpolated on to the RK4 temporal grid. The accuracy estimates obtained for the ESDIRK4 simulations are $\mathcal{A}_{\text{ESDIRK4}-\Delta t=1(10)^{-7}} = 2.12 \ (10)^{-6}$, $\mathcal{A}_{\text{ESDIRK4}-\Delta t=1(10)^{-8}} = 3.42 \ (10)^{-9}$, $\mathcal{A}_{\text{ESDIRK4}-\Delta t=1(10)^{-9}} = 9.67 \ (10)^{-12}$, and lastly $\mathcal{A}_{\text{ESDIRK4}-\Delta t=1(10)^{-10}} = 4.1 \ (10)^{-14}$, respectively. These values for $\mathcal{E}$ and $\mathcal{A}$ indicate that while there are some sacrifices in terms of solution accuracy in going to the larger time steps with the ESDIRK4 scheme, these errors may be acceptable depending on the application, and for stiff applications speedup in terms of reduced computational times can be realized. Note also that the test problem of Figure 10 is only moderately stiff, and increasing the grid resolution would increase the stiffness.

Finally, for the BDF4 scheme, the following measures of the efficiency and accuracy measures were obtained: $\mathcal{E}_{\text{BDF4}-\Delta t=1(10)^{-9}} = 0.53$ and $\mathcal{A}_{\text{BDF4}-\Delta t=1(10)^{-9}} = 5.43 \ (10)^{-12}$ for $\Delta t = 1 \ (10)^{-9}$; $\mathcal{E}_{\text{BDF4}-\Delta t=5(10)^{-10}} = 0.31$ and $\mathcal{A}_{\text{BDF4}-\Delta t=5(10)^{-10}} = 2.45 \ (10)^{-13}$ for $\Delta t = 5 \ (10)^{-10}$ and $\mathcal{E}_{\text{BDF4}-\Delta t=1(10)^{-10}} = 0.07$ and $\mathcal{A}_{\text{BDF4}-\Delta t=1(10)^{-10}} = 2.16 \ (10)^{-14}$ for $\Delta t = 1 \ (10)^{-10}$ indicating very good accuracy but rather poor computational efficiency due to the rather restrictive upper bound on the time step for the vortex shedding simulation.

The computational savings provided by employing the ESDIRK4 scheme over the conditionally stable RK4 method are summarized in Figure 11. For these purposes, the accuracy of the solutions has been estimated by comparing the predicted value of the x-direction component of the flow velocity over one full oscillation of the vortex shedding process. As there is no analytical solution for this case, the solution error is calculated with reference to a computed solution obtained using a very small time step (i.e., $\Delta t = 5 \ (10)^{-12}$). Figure 11 depicts this estimated error in the predicted temporal variation of the x-direction component of the flow velocity as a function of the total computational CPU time (left) and the size of the physical time step (right) used to obtain the unsteady vortex-shedding solution. It can be seen that rather significant computational savings are provided by the ESDIRK4 scheme without significantly compromising the global accuracy of the solution. Overall, the slight loss of accuracy incurred in using ESDIRK4 scheme is thought to be outweighed by the computational savings afforded by the use of a larger time step. The numerical results for the vortex shedding problem show that the implicit, A-stable, ESDIRK4, time-marching scheme is both an efficient and stable
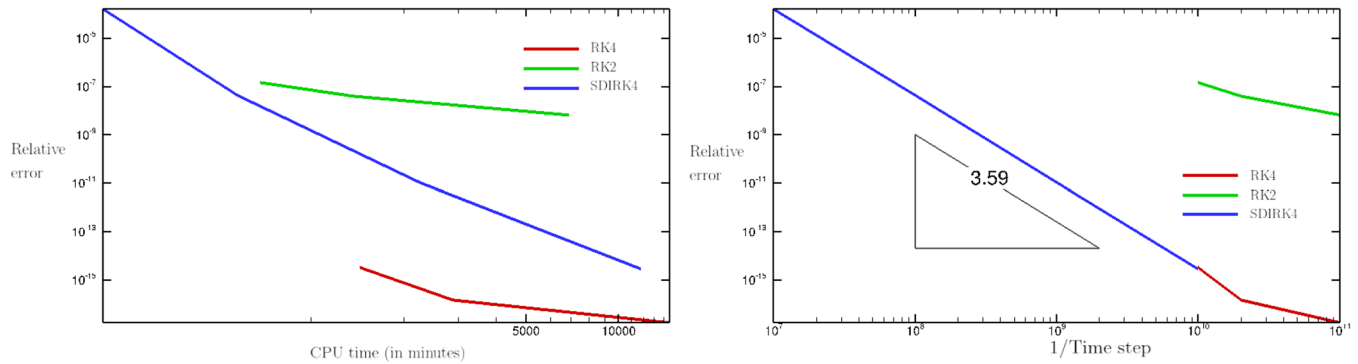
FIGURE 11 Numerical validation 4: Estimated error in the x-direction component of the flow velocity as a function of the total CPU time for the simulation (left) and as a function of the size of the time step, Δt, (right) obtained using the RK2, RK4, and ESDIRK4 time marching schemes when simulating the unsteady viscous vortex shedding flow problem [Colour figure can be viewed at wileyonlinelibrary.com]

method. When combined with CENO finite-volume scheme, the ESDIRK scheme is capable of outperforming standard explicit time stepping schemes in terms of computational work or effort for relatively stiff problems. On the other hand, the BDF4 scheme, which is not A-stable, was found to be unstable even for time steps that are not large compared with stable time step sizes of comparable explicit time stepping methods. For example, a value of $\Delta t = 5 \, (10)^{-9}$ proved to be unstable while stable RK4 results could be obtained for $\Delta t = 1 \, (10)^{-10}$.

5.5 | Numerical validation 5: 1D Sod shock tube problem

While the focus of the present study is the formulation and assessment of high-order implicit time-stepping schemes in combination with the CENO finite-volume method for application to smooth unsteady flows having regular solutions and the formulation and application to problems with nonsmooth solution will be considered in future follow-on studies, the capabilities of the proposed high-order implicit time-stepping schemes for several nonsmooth problems, for example, with shocks and/or discontinuities, is now investigated thus providing a starting point for the follow-on research. As a first nonsmooth flow case, the well-known 1D Sod shock tube problem[61] for the inviscid Euler equations of gas dynamics is examined here. For the purpose of this validation study, the fourth-order CENO method for spatial discretization coupled with the fourth-order RK4, ESDIRK4, and BDF4 time integrators was applied to the solution of the Sod problem using various 1D grid resolutions. The HLLE flux function was used in this case. The initial data for the Sod shock tube problem associated with the one-dimensional coordinate, x, is given by[61]

$$(\rho, \, \mathbf{v}, \, p) = \begin{cases} (\rho_0, \, \mathbf{0}, \, p_0), & \text{for } x < 0, \\ (8\rho_0, \, \mathbf{0}, \, 10p_0), & \text{for } x > 0, \end{cases} \tag{78}$$

where here ρ_0 and p_0 are the standard atmospheric density of dry air and standard atmospheric pressure at sea level, respectively.

Figure 12 depicts the numerical approximations obtained using both the explicit and implicit time-stepping schemes on the increasingly finer meshes corresponding to $N_x = 64$, 128, 256, 512 uniformly spaced cells in the x-direction, respectively, on $x \in [-1/2, 1/2]$ for the 1D Sod shock tube problem. The step size, Δt, for the RK4 scheme was restricted to a relatively small value corresponding to a Courant number of $\nu_{\text{CFL}} = 0.25$. For comparison purposes, numerical results are also given in the figure for the related second-order limited least-squares spatial discretization scheme with Venkatakrishnan limiter[50] coupled with the RK2 time marching method with a Courant number of $\nu_{\text{CFL}} = 0.25$. For this value of the CFL or Courant number, the RK2 time marching scheme applied to the semidiscrete form of the ODEs resulting from the limited linear least squares spatial discretization procedure can be shown to satisfy the SSP property as originally defined by Shu and Osher[38,39] and hence this reference second-order numerical solution is expected to be total-variation-diminishing (TVD) (i.e., monotone, at least for a nonlinear scalar PDE). The exact solution of Sod problem is also provided in Figure 12 for reference purposes. The latter was determined via solution of the corresponding Riemann
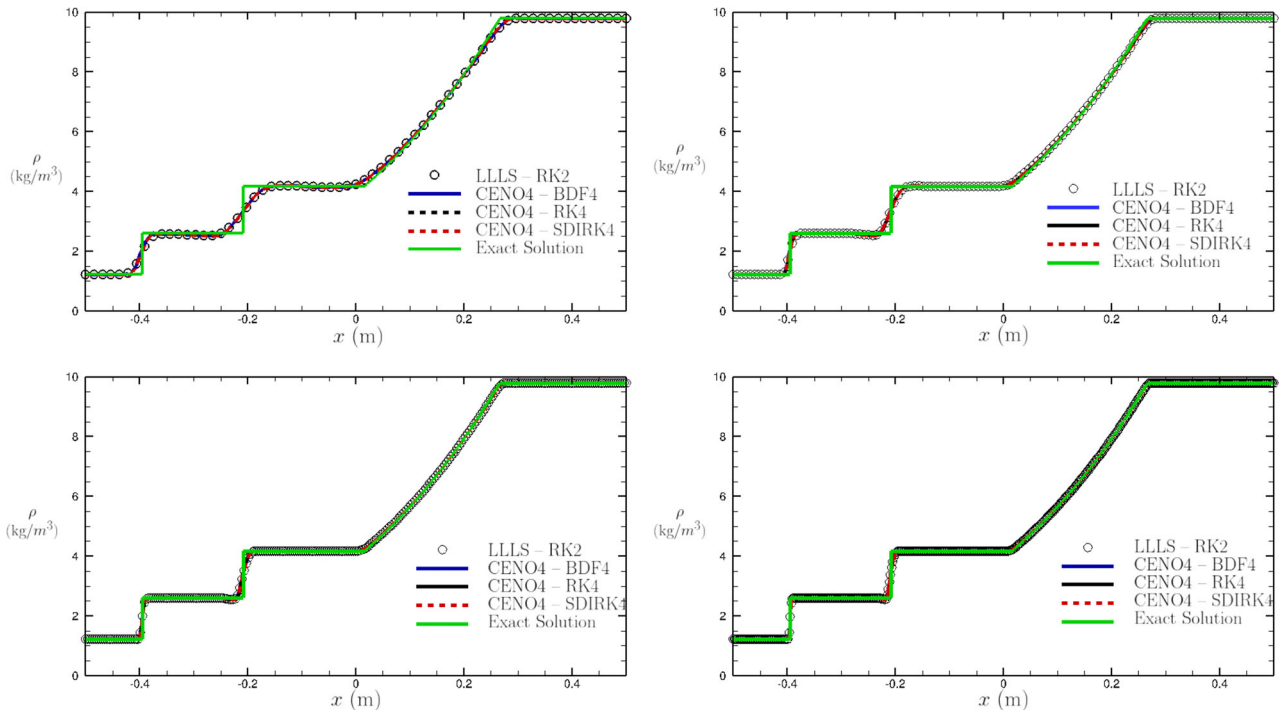
FIGURE 12 Numerical validation 5: Predicted density profiles for the 1D Sod tube problem at simulation time $t = 0.7$ obtained for 1D mesh sizes of 64 cells (top left), 128 cells (top right), 256 cells (bottom left), 512 cells (bottom right) in the x-coordinate direction using the second-order limited linear least-squares scheme in space with the explicit RK2 time integrator (LLLS-RK2); the fourth-order CENO scheme in space with the implicit BDF4 time integrator (CENO4-BDF4); the fourth-order CENO scheme in space with the explicit RK4 time integrator (CENO4-RK4); and the fourth-order CENO scheme in space with the implicit ESDIKR4 time integrator (CENO4-SDIRK4). The predicted solutions are also compared with the exact solution of the Sod problem [Colour figure can be viewed at wileyonlinelibrary.com]

problem using the exact Riemann solver of Gottlieb and Groth.[64] For the high-order implicit time marching methods, CFL or Courant numbers of $\nu_{CFL} = 0.5$ and $\nu_{CFL} = 1$ were used for the BDF4 and ESDIRK4 time marching schemes, respectively. It is noted that for higher values of ν_{CFL}, that is, for $\nu_{CFL} > 0.5$ and $\nu_{CFL} > 1$ respectively, the BDF4 and ESDIRK4 fail to provide solutions due to loss of solution monotonicity and the occurrence of nonphysical solution values (i.e., negative densities and/or energies).

As can be observed from Figure 12, the explicit RK4 and implicit BDF4 and ESDIRK4 methods coupled with the fourth-order CENO scheme provide comparable predictions for the Sod problem and the latter are virtually equivalent to those of the TVD limited least-squares spatial scheme coupled with the SSP RK2 time marching method with respect to their shock-capturing capability. In particular, the predicted density profiles captured by the high-order schemes are almost indistinguishable on all meshes considered and are very similar to those of the second-order SSP RK2 results. It should be noted that the Sod problem largely involves a sequence of shocks or discontinuities separated by regions with constant solution values and hence the improvements offered by the high-order spatial accuracy of the CENO finite-volume scheme and BDF4 and ESDIRK4 time marching schemes are not expected to be significant as the smoothness indicator and limiter will be active near the shocks and discontinuities to enforce solution monotonicity and the constant flow regions are of course well captured by standard lower order scheme. Nevertheless, the numerical results for this case provide evidence for the reliable shock-capturing properties of the proposed high-order implicit schemes, at least for modest time steps, and the feasibility of the inner nonlinear solvers similar to those of the ESDIRK4 method coupled with the fourth-order CENO spatial discretization scheme for handling problems with shocks and contact discontinuities.

5.6 | Numerical validation 6: Ideal MHD version of 1D Shu–Osher problem

To further investigate the application of the proposed high-order implicit methods for problems involving shocks, we also consider the application of the implicit schemes to an extended version of the 1D initial value problem proposed by Shu
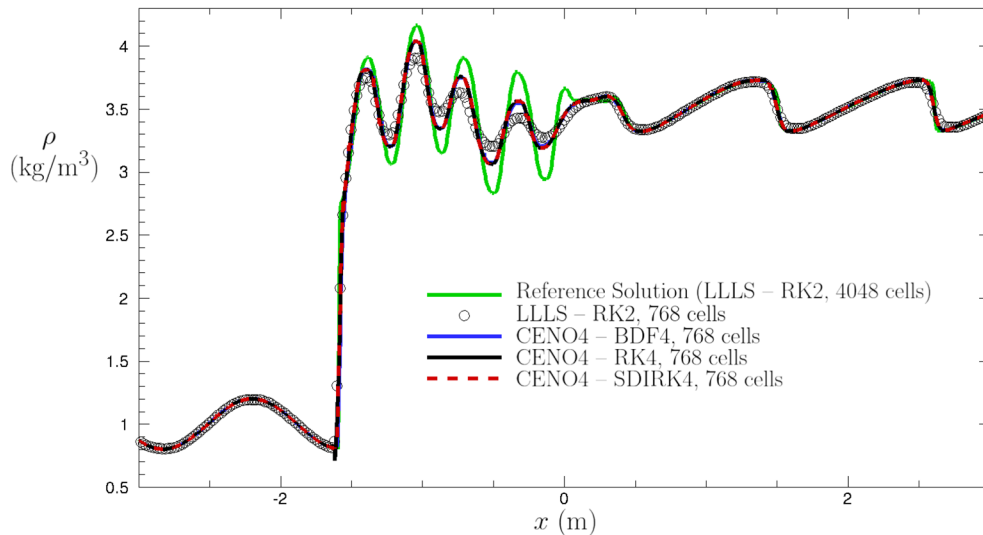
FIGURE 13   Numerical validation 6: Predicted density profiles for the 1D Shu–Osher ideal MHD problem at simulation time $t = 679.74$ obtained on a 1D mesh size of 768 cells in the x-coordinate direction using the second-order limited linear least-squares scheme in space with the explicit RK2 time integrator (LLLS-RK2); the fourth-order CENO scheme in space with the implicit BDF4 time integrator (CENO4-BDF4); the fourth-order CENO scheme in space with the explicit RK4 time integrator (CENO4-RK4); and the fourth-order CENO scheme in space with the implicit ESDIKR4 time integrator (CENO4-SDIRK4). The predicted solutions are also compared with a reference solution for the Shu–Osher problem obtained using the combined second-order limited linear least-squares finite-volume and RK2 time marching schemes (LLLS-RK2) using 4048 computational cells [Colour figure can be viewed at wileyonlinelibrary.com]

and Osher[62] to the ideal MHD equations as considered previously by Susanto et al.[40] For this case, a sinusoidal density perturbation is added downstream of a purely advecting superfast shock wave. The interaction of the shock wave with the sinusoidal varying density field gives rise to fast oscillations and complex flow features downstream of the shock. For $\gamma = 5/3$, the unperturbed initial conditions (in terms of vector field components along and perpendicular to the x coordinate direction of the one-dimensional initial value problem) are given by:[40]

$$(\rho, \mathbf{v}, \mathbf{B}, p) = \begin{cases} (1, \mathbf{0}, (1, 1, 0), 1) & \text{for } x < 0, \\ (3.5, (-5.8846, 1.1198, 0), (1, 3.6359, 0), 42.0267) & \text{for } x > 0, \end{cases} \tag{79}$$

and a sinusoidal density perturbation, $\Delta \rho$, is added to the upstream portion of the density field solution according to

$$\Delta \rho = 0.2 \sin(5x), \text{ for } x < 0. \tag{80}$$

The fourth-order CENO method for spatial discretization coupled with the fourth-order RK4, ESDIRK4, and BDF4 time integrators was again applied to the solution of this problem using a one-dimensional mesh consisting of $N_x = 768$ uniformly spaced computational cells on $x \in [-7.5, 7.5]$ and for a maximum simulated time of $t = 679.74$. A CFL number of $\nu_{CFL} = 0.25$ and local Lax–Friedrichs flux function were used in each case. For comparison, numerical predictions were also obtained using the second-order limited least-squares spatial discretization scheme with Venkatakrishnan limiter coupled with the SSP RK2 time marching method with a Courant number of $\nu_{CFL} = 0.25$ for mesh sizes consisting of 768 and 4048 cells. The coarse mesh second-order solution provides a means of measuring of the relative improvement offered by the high-order discretization methods and, as an analytical solution is not available for this case, the fine mesh result serves as a reference "exact" solution.

A comparison of the predicted density profiles of the various high-order time marchings coupled to the fourth-order CENO finite-volume scheme is given in Figure 13. Again, it is clear that for this relatively modest CFL number of $\nu_{CFL} = 0.25$, the high-order time marching schemes combined with the CENO approach are stable at the shock and do not produce spurious oscillations. Furthermore, unlike in the Sod problem for the same number of computational cells, the fourth-order methods captures here the small-scale flow features following the shock much better than the reference SSP second-order method. It should be noted that, as for the Sod problem, as the Courant number, ν_{CFL}, was increased for the

Shu–Osher problem and made to approach unity the nonmonotone behavior of the BDF4 and ESDIRK4 time marching scheme solutions becomes significant and stable physically realizable solutions could not be obtained.

6 | CONCLUSION

A numerical investigation of different high-order implicit time-stepping methods, namely, the BDF4, ESDIRK4, and Rosenbrock-type ROS34PW2 methods, coupled with the high-order, CENO, finite-volume, spatial discretization scheme in a method of lines framework has been presented. These schemes have been applied to the simulation of a number of unsteady CFD flow problems associated with the Euler, ideal and resistive MHD, and Navier–Stokes equations of compressible gases and plasmas. It has been shown that, in case of smooth stiff problems, the implicit ESDIRK4 time integrator is both robust and accurate and can be considerably more efficient in terms of computational costs for a comparable accuracy compared with the standard explicit RK4 scheme. Larger cost savings can be achieved thanks to the much larger choice of time step sizes possible with this implicit method, which compensates for the higher costs per time step. It was found that the Rosenbrock-type ROS34PW2 scheme appears to suffer from accuracy issues on fine grids, and it suffers from robustness issues for stiff problems. Furthermore, while the BDF4 scheme was successfully applied for simulations for the Euler and MHD equations, it suffered from stability issues for the Navier–Stokes simulations. Two additional nonsmooth problems involving shocks were also considered, showing that the proposed fourth-order accurate implicit approach maintains positivity via the fourth-order accurate CENO spatial discretization for moderate CFL numbers. The application of implicit time stepping methods for problems with shocks while using large time steps will be the subject of further follow-on research.

DATA AVAILABILITY STATEMENT
The data that support the findings of this study are available from the corresponding author, T. Binh Nguyen, upon reasonable request.

ORCID
Lucie Freret https://orcid.org/0000-0002-8499-6962

REFERENCES
1. Brackbill JU, Cohen BI. *Multiple Time Scales*. Academic Press, Inc; 1985.
2. Jardin SC. Review of implicit methods for the magnetohydrodynamic description of magnetically confined plasmas. *J Comput Phys*. 2012;231:822-838.
3. Reynolds DR, Samtaney R, Woodward CS. A fully implicit numerical method for single-fluid resistive magnetohydrodynamics. *J Comput Phys*. 2006;219:144-162.
4. Bao W, Jin S. The random projection method for hyperbolic conservation laws with stiff reaction terms. *J Comput Phys*. 2000;163:216-248.
5. Rossow CC. Efficient computation of compressible and incompressible flows. *J Comput Phys*. 2007;220:879-899.
6. Zhang M, Feng X. Implicit dual-time stepping method for a solar wind model in spherical coordinates. *Comput Fluids*. 2015;115:115-123.
7. Trefethen LN, Bau D III. *Numerical Linear Algebra*. SIAM; 1997.
8. Saad Y. *Iterative Methods for Sparse Linear Systems*. PWS Publishing Company; 1996.
9. Saad Y, Schultz MH. GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear equations. *SIAM J Sci Stat Comput*. 1986;7:856-869.
10. Knoll DA, Keyes DE. Jacobian-free Newton-Krylov methods: a survey of approaches and applications. *J Comput Phys*. 2004;193:357-397.
11. Charest MRJ, Groth CPT, Gauthier PQ. A high-order central ENO finite-volume scheme for three-dimensional low-speed viscous flows on unstructured mesh. *Commun Comput Phys*. 2015;17(3):615-656.
12. Hairer E, Wanner G. *Solving Ordinary Differential Equations II, Series in Edn*. Springer; 2004.
13. Hairer E, Wanner G. *Solving Ordinary Differential Equations II: Stiff and Differential-Algebraic Problems*. Springer; 1980.
14. Dahlquist GG. A special stability problem for linear multistep methods. *BIT*. 1963;3:27-43.

15. Wanner G. Dahlquist's classical papers on stability theory. *BIT Numer Math*. 2006;46:671-683.

16. Birken P. *Numerical Methods for the Unsteady Compressible Navier-Stokes Equations*. PhD thesis. Habilitation thesis. University of Kassel; 2012.

17. Weiner R, Schmitt BA, Podhaisky H. ROWMAP - a ROW-code with Krylov techniques for large stiff ODEs. *Appl Numer Math*. 1997;25:303-319.

18. Rang J, Angermann L. New Rosenbrock W-methods of order 3 for partial differential algebraic equations of index 1. *BIT Numer Math*. 2005;45:761-787.

19. Tranquilli P, Sandu A. Rosenbrock-Krylov methods for large systems of differential equations. *SIAM J Sci Comput*. 2014;36:A1313-A1338.

20. Blom DS, Birken P, Bijl H, Kessels F, Meister A, van Zuijlen AH. A comparison of Rosenbrock and ESDIRK methods combined with iterative solvers for unsteady compressible flows. *Adv Comput Math*. 2016;42:1401-1426. https://doi.org/10.1007/s10444-016-9468-x

21. Liu X, Xia Y, Luo H, Xuan L. A comparative study of Rosenbrock-type and implicit Runge-Kutta time integration for discontinuous Galerkin method for unsteady 3D compressible Navier-Stokes equations. *Commun Comput Phys*. 2016;20(4):1016-1044.

22. Cox SM, Matthews PC. Exponential time differencing for stiff problems. *J Comput Phys*. 2002;176:430-455.

23. Jung CY, Nguyen TB. Semi-analytical time differencing methods for stiff problems. *J Sci Comput*. 2015;63(2):355-373.

24. Jung CY, Nguyen TB. New time differencing methods for spectral methods. *J Sci Comput*. 2016;66(2):650-671.

25. Hochbruck M, Ostermann A, Schweitzer J. Exponential Rosenbrock-type methods. *SIAM J Numer Anal*. 2009;47(1):786-803.

26. Pareschi L, Russo G. Implicit-explicit Runger-Kutta schemes and applications to hyperbolic systems with relaxation. *J Sci Comput*. 2005;25(1):129-155.

27. Vermeire BC, Nadarajah S. Adaptive IMEX schemes for high-order unstructured methods. *J Comput Phys*. 2015;280:261-286.

28. Ivan L, Groth CPT. High-order solution adaptive central essentially non-oscillatory (CENO) method for viscous flows. *J Comput Phys*. 2014;257:830-862.

29. Ivan L. *Development of High-Order CENO Finite-Volume Schemes with Block-Based Adaptive Mesh Refinement*. PhD thesis. University of Toronto; 2011.

30. Ivan L, De Sterck H, Susanto A, Groth CPT. High-order central ENO finite-volume scheme for hyperbolic conservation laws on three-dimensional cubed-sphere grids. *J Comput Phys*. 2015;282:157-182.

31. Freret L, Groth CPT. Anisotropic non-uniform block-based adaptive mesh refinement for three-dimensional inviscid and viscous flows. Proceedings of the 22nd AIAA Computational Fluid Dynamics Conference; 2015:2613.

32. Freret L, Ivan L, De Sterck H, Groth CPT. High-order finite-volume method with block-based AMR for magnetohydrodynamics flows. *J Sci Comput*. 2019;79:176-208.

33. Neto LT, Groth CPT. A high-order finite-volume scheme for large-eddy simulation of turbulent premixed flames. Proceedings of the 52nd AIAA Aerospace Sciences Meeting; 2014:1024; American Institute of Aeronautics and Astronautics.

34. Ivan L, Groth CPT. High-order central ENO finite-volume scheme with adaptive mesh refinement. Proceedings of the 18th AIAA Computational Fluid Dynamics Conference; 2007:4323.

35. Freret L, Groth CPT, Nguyen TB, De Sterck H. High-order finite-volume scheme with anisotropic adaptive mesh refinement: efficient inexact Newton method for steady three-dimensional flows. Proceedings of the 23rd AIAA Computational Fluid Dynamics Conference; 2017:3297.

36. Freret L, Groth CPT, De Sterck H. A parallel high-order CENO finite-volume scheme with AMR for three-dimensional ideal MHD flows. *Spectral and High Order Methods for Partial Differential Equations ICOSAHOM*. Springer; 2017:343-355.

37. Barth TJ. Recent developments in high order K-exact reconstruction on unstructured meshes. Proceedings of the 31st Aerospace Sciences Meeting; 1993:0668.

38. Shu CW, Osher S. Efficient implementation of essentially non-oscillatory shock-capturing schemes. *J Comput Phys*. 1988;77:439-471.

39. Gottlieb S, Shu CW, Tadmor E. Strong stability-preserving high-order time discretization methods. *SIAM Rev*. 2001;43(1):89-112.

40. Susanto A, Ivan L, De Sterck H, Groth CPT. High-order central ENO finite-volume scheme for ideal MHD. *J Comput Phys*. 2013;250:141-164.

41. Northrup SA. *A Parallel Implicit Adaptive Mesh Refinement Algorithm for Predicting Unsteady Fully-Compressible Reactive Flows*. PhD thesis. University of Toronto; 2014.

42. Susanto A. *High-Order Finite-Volume Schemes for Magnetohydrodynamics*. PhD thesis. University of Waterloo; 2014.

43. Toro EF. *Riemann Solvers and Numerical Methods for Fluid Dynamics: A Practical Introduction*. 3rd ed. Springer; 2009.

44. Roe PL. Approximate Riemann solvers, parameter vectors, and difference schemes. *J Comput Phys*. 1981;43:357-372.

45. Harten A. High resolution schemes for hyperbolic conservation laws. *J Comput Phys*. 1983;49:357-393.

46. Harten A, Chakravarthy SR. Multi-dimensional ENO schemes for general geometries. ICASE report; 1991:76.

47. Jiang GS, Shu CW. Efficient implementation of weighted ENO schemes. *J Comput Phys*. 1996;126:202-228.

48. Jung CY, Nguyen TB. A new adaptive weighted essentially non-oscillatory WENO-θ scheme for hyperbolic conservation laws. *J Comput Appl Math*. 2018;328:314-339.

49. Lawson C, Hanson R. *Solving Least Squares Problems*. Prentice-Hall, Inc; 1974.

50. Venkatakrishnan V. On the accuracy of limiters and convergence to steady state solutions. Proceedings of 31st Aerospace Sciences Meeting:1993; 90-0880.

51. Borges R, Carmona M, Costa B, Don WS. An improved weighted essentially non-oscillatory scheme for hyperbolic conservation laws. *J Comput Phys*. 2008;227:3191-3211.

52. Dedner A, Kemm F, Köner D, Munz CD, Schnitzer T, Wesenberg M. Hyperbolic divergence cleaning for the MHD equations. *J Comput Phys*. 2002;175:645-673.

53. Brackbill JU, Barnes DC. Note: the effect of nonzero $\nabla \cdot \mathbf{B}$ on the numerical solution of the magnetohydrodynamic equations. *J Comput Phys*. 1980;35:426.

54. Chorin AJ. A numerical method for solving incompressible viscous flow problems. *J Comput Phys*. 1967;2:12-26.

55. Powell KG. An approximate Riemann solver for magnetohydrodynamics. Technical report. The University of Michigan; 1994.

56. Alexander R. Design and implementation of DIRK integrators for stiff systems. *Appl Numer Math*. 2003;46:1-17.

57. Kværnø A. Singly diagonally implicit Runge-Kutta methods with an explicit first stage. *BIT Numer Math*. 2004;44:489-502.

58. Ostermann A, Roche M. Rosenbrock methods for partial differential equations and fractional orders of convergence. *SIAM J Numer Anal*. 1998;30:1084-1098.

59. Lang J, Verwer J. ROS3P - an accurate third-order Rosenbrock solver designed for parabolic problems. *BIT*. 2001;41:731-738.

60. Northrup SA, Groth CPT. Parallel implicit adaptive mesh refinement scheme for unsteady fully-compressible reactive flows. Proceedings of the 17th AIAA Computational Fluid Dynamics Conference; 2013.

61. Sod GA. A survey of several finite difference methods for systems of nonlinear hyperbolic conservation laws. *J Comput Phys*. 1978;27:1-31.

62. Shu CW, Osher S. Efficient implementation of essentially non-oscillatory shock-capturing schemes: II. *J Comput Phys*. 1989;83:32-78.

63. Henderson RD. Details of the drag curve near the onset of vortex shedding. *Phys Fluids*. 1995;7(9):2102-2104.

64. Gottlieb JJ, Groth CPT. Assessment of Riemann solvers for unsteady one-dimensional inviscid flows of perfect gases. *J Comput Phys*. 1988;78:437-458.