INL/JOU-15-36158
Accepted Manuscript

A Comparative Study of Rosenbrock-Type and Implicit Runge-Kutta Time Integration for Discontinuous Galerkin Method for Unsteady 3D Compressible Navier-Stokes equations

Xiaodong Liu, Yidong Xia, Hong Luo, Lijun Xuan

October 2016

Prepared for the U.S. Department of Energy
Office of Nuclear Energy
Under DOE Idaho Operations Office
Contract DE-AC07-05ID14517

Commun. Comput. Phys.
doi: 10.4208/cicp.300715.140316a

Vol. **xx**, No. x, pp. 1-29
xxx 201x

A Comparative Study of Rosenbrock-Type and Implicit Runge-Kutta Time Integration for Discontinuous Galerkin Method for Unsteady 3D Compressible Navier-Stokes equations

Xiaodong Liu[1], Yidong Xia[2], Hong Luo[1,*] and Lijun Xuan[1]

[1] *Department of Mechanical and Aerospace Engineering, North Carolina State University, Raleigh, NC 27695, USA.*
[2] *Department of Energy Resource Recovery & Sustainability, Idaho National Laboratory, Idaho Falls, ID 83415, USA.*

Received XXX; Accepted (in revised version) XXX

Abstract. A comparative study of two classes of third-order implicit time integration schemes is presented for a third-order hierarchical WENO reconstructed discontinuous Galerkin (rDG) method to solve the 3D unsteady compressible Navier-Stokes equations: — 1) the explicit first stage, single diagonally implicit Runge-Kutta (ESDIRK3) scheme, and 2) the Rosenbrock-Wanner (ROW) schemes based on the differential algebraic equations (DAEs) of Index-2. Compared with the ESDIRK3 scheme, a remarkable feature of the ROW schemes is that, they only require one approximate Jacobian matrix calculation every time step, thus considerably reducing the overall computational cost. A variety of test cases, ranging from inviscid flows to DNS of turbulent flows, are presented to assess the performance of these schemes. Numerical experiments demonstrate that the third-order ROW scheme for the DAEs of index-2 can not only achieve the designed formal order of temporal convergence accuracy in a benchmark test, but also require significantly less computing time than its ESDIRK3 counterpart to converge to the same level of discretization errors in all of the flow simulations in this study, indicating that the ROW methods provide an attractive alternative for the higher-order time-accurate integration of the unsteady compressible Navier-Stokes equations.

AMS subject classifications: 65M60, 76M10

Key words: Implicit time integration, Rosenbrock-Wanner, discontinuous Galerkin, WENO, Navier-Stokes.

*Corresponding author. *Email addresses:* xliu29@ncsu.edu (X. Liu), yidongxia@gmail.com (Y. Xia), hong_luo@ncsu.edu (H. Luo), lxuan@ncsu.edu (L. Xuan)

1 Introduction

The discontinuous Galerkin (DG) methods, originally introduced for solving the neutron transport by Reed and Hill [1], have become popular for the solution of systems of conservation laws in recent decades. Nowadays, they are widely used in computational fluid dynamics (CFD), computational acoustics, and computational magneto-hydrodynamics. A comprehensive overview of the DG methods can be found in [2]. The DG methods combine the attractive features of both the finite element (FE) and finite volume (FV) methods, and thus are especially advantageous in solving the hyperbolic-type system of equations in terms of solution accuracy [3–6], treatment of non-conforming meshes [7], and implementation of the hp-adaptivity [8]. However, the DG methods also have a number of weaknesses that have yet to be addressed, before they can be used for problems of practical interest. In particular, how to reduce the high computational costs and how to develop more efficient time integration methods are two of the most long-standing research challenges.

In order to reduce the high costs associated to the DG methods, Dumbser et al. [9–11] introduced a new family of so-called reconstructed DG, termed PnPm schemes and referred to as rDG (PnPm) in this paper. Pn indicates that a piecewise polynomial of degree of n is used to represent the underlying DG solution, and Pm represents a polynomial solution of degree of m ($m \geq n$) that is reconstructed from the underlying Pn polynomial and used to compute the fluxes. The PnPm schemes can be constructed based on a few different algorithms, e.g., the recovery approach [12], the reconstruction approach [13,14], and the Gauss-Green approach [15,16], all of which were proved to deliver the designed grid convergence of $\mathcal{O}(h^{m+1})$ [17]. Indeed, implicit methods can especially benefit from the use of rDG (PnPm) methods as the costs can be substantially reduced in two aspects [18, 19]. Firstly, fewer spatial integration points are required for evaluating the residual vector and Jacobian matrix. For instance, the third-order rDG (P1P2) only needs 4 points for triangular boundary integral whereas the equivalent DG (P2) requires 7. Secondly, the Jacobian matrix of rDG (PnPm) is based on the underlying DG (Pn), and thus requires much less storage than the equivalent DG (Pm). For example, the memory needed for the diagonal part of the Jacobian matrix of rDG (P1P2) is 400 word versus 2500 needed by DG (P2) for the 3D Navier-Stokes equations.

The spatial discretization of the compressible Navier-Stokes equations leads to a system of Ordinary Differential Equations (ODEs). Significant progress has also been made in developing efficient higher-order implicit time integration methods for such system in order to reduce the temporal discretization error incurred from the use of lower-order time integration methods. Bijl et al. [20] introduced ESDIRK schemes for the finite volume solutions to the Navier-Stokes equations. Later on, Wang and Mavriplis [21] extended the ESDIRK schemes to solve the compressible Euler equations using a high-order p-multigrid DG method. Xia et al. [22] also used ESDIRK for the time accurate solutions of the 3D compressible Navier-Stokes equations in the context of the reconstructed

P1P2 DG methods. They all have concluded that the higher-order ESDIRK schemes are more efficient than those second-order time integration methods to achieve the solutions for the same level of temporal accuracy. However, ESDIRK method has to solve nonlinear systems, increasing the computational cost. Therefore, recently, a family of Rosenbrock-type time integration schemes for the unsteady flow simulations was extensively explored, replacing non-linear systems with a sequence of linear systems. Some effective Rosenbrock-type methods were designed to solve the DAEs (Differential Algebraic Equations) of Index-1 and Index-2. The DAEs originate in the modeling of various physical or chemical phenomena. They are classified by their differential index, that is, the minimum number of times that a DAEs system must be differentiated in order to become an ODEs system. For instance, the discrete incompressible Stokes, Oseen, and Navier-Stokes equations belong to index-2 DAEs systems. Unlike the incompressible Navier-Stokes equations, there is no algebraic constraint for the compressible Navier-Stokes equations, indicating that the Rosenbrock-type methods, originally designed for the DAEs, could also be introduced to advance the unsteady compressible Navier-Stokes equations in time. However, a major disadvantage associated to the standard Rosenbrock method is that it requires the use of exact Jacobians in order to achieve formal order of temporal convergence, which is usually too demanding in practice. On the other hand, the Rosenbrock-Wanner method, abbreviated as ROW in this paper, does not have such prohibitive restriction, and would only need approximate Jacobians.

Bassi et al. [23] investigated the use of Rosenbrock-type schemes to integrate in time high order DG discretizations of Navier-Stokes equations. They applied the best performing Rosenbrock schemes to the Implicit Large Eddy Simulation of the transitional flow around SD7003 airfoil. Blom et al. [24] compared the time adaptive ESDIRK and ROW method in the context of finite volume discretizations of the compressible Navier-Stokes equations. They concluded that the ROW scheme, ROS34PW2 (34 stands for order 3 with 4 stages, W for W-method and 2 is an internal number), were suitable candidates for engineering accuracies. Birken et al. [25] also compared the performance of the adaptive time integration in the context of discontinuous Galerkin methods for the 3D unsteady compressible Navier-Stokes equations.

The objective of this paper is to present a set of efficient third-order ROW schemes for Index-2 DAEs [26, 27] to the solution of the ODEs in the context of a spatially third-order rDG (P1P2) method. And we also make comparative study between the ROW for index-1, ROW for index-2 and ESDIRK3 schemes in the context of rDG method through a variety of test problems.

The outline of the rest of this paper is organized as follows. The governing equations are described in Section 2. The developed reconstructed discontinuous Galerkin method is presented in Section 3. The Rosenbrock temporal discretization schemes are then described in detail in Section 4. The numerical examples are presented in Section 5. Finally, the concluding remarks are given in Section 6.

2 Governing equations

The Navier-Stokes equations governing unsteady compressible viscous flows can be expressed as

$$\frac{\partial \mathbf{U}(x,t)}{\partial t}+\frac{\partial \mathbf{F}_k(\mathbf{U}(x,t))}{\partial x_k}=\frac{\partial \mathbf{G}_k(\mathbf{U}(x,t))}{\partial x_k}, \tag{2.1}$$

where the summation convention has been used. The conservative variable vector **U**, advective (inviscid) flux vector **F**, and viscous flux vector **G** are defined by

$$\mathbf{U}=\begin{pmatrix}\rho\\ \rho u_i\\ \rho e\end{pmatrix},\quad \mathbf{F}_j=\begin{pmatrix}\rho u_j\\ \rho u_i u_j+p\delta_{ij}\\ u_j(\rho e+p)\end{pmatrix},\quad \mathbf{G}_j=\begin{pmatrix}0\\ \delta_{ij}\\ u_i(\delta_{ij}+q_j)\end{pmatrix}. \tag{2.2}$$

Here ρ, p, and e denote the density, pressure, and specific total energy of the fluid, respectively, and u_i is the velocity of the flow in the coordinate direction. The pressure can be computed from the equation of state

$$p=(\gamma-1)\rho\Big(e-\frac{1}{2}u_j u_j\Big), \tag{2.3}$$

which is valid for perfect gas, where γ is the ratio of the specific heats. The components of the viscous stress tensor σ_{ij} and the heat flux q_j are given by

$$\sigma_{ij}=\mu\Big(\frac{\partial u_i}{\partial x_j}+\frac{\partial u_j}{\partial x_i}\Big)-\frac{2}{3}\mu\frac{\partial u_k}{\partial x_k}\delta_{ij},\qquad q_j=\frac{1}{\gamma-1}\frac{\mu}{Pr}\frac{\partial T}{\partial x_j}. \tag{2.4}$$

In the above equations, T is the temperature of the fluid, Pr the laminar Prandtl number, which is taken as 0.7 for air. μ represents the molecular viscosity, which is determined through Sutherland's law

$$\frac{\mu}{\mu_0}=\Big(\frac{T}{T_0}\Big)^{\frac{3}{2}}\frac{T_0+S}{T+S}.$$

μ_0 denotes the viscosity at the reference temperature T_0, and S is a constant which for are assumes the value $S=110K$. The temperature of the fluid T is given by

$$T=\gamma\frac{p}{\rho}.$$

3 DG formulation for Navier-Stokes equations

The governing equation (2.1) is discretized using a discontinuous Galerkin finite element formulation. To formulate the discontinuous Galerkin method, we first introduce the following weak formulation, which is obtained by multiplying the above conservation law

by a test function W, integrating over the domain $\mathbf{\Omega}$, and then performing an integration by parts,

$$\int_\Omega \frac{\partial \mathbf{U}}{\partial t} W d\Omega + \int_\Gamma \mathbf{F}_k \mathbf{n}_k W d\Gamma - \int_\Omega \mathbf{F}_k \frac{\partial W}{\partial x_k} d\Omega = \int_\Gamma \mathbf{G}_k \mathbf{n}_k d\Gamma - \int_\Omega \mathbf{G}_k \frac{\partial W}{\partial x_k} d\Omega, \quad \forall W \in \mathbf{V}, \tag{3.1}$$

where $\Gamma\,(=\partial\Omega)$ denotes the boundary of Ω, and $\mathbf{n}_j$ the unit outward normal vector to the boundary. We assume that the domain Ω is subdivided into a collection of non-overlapping elements Ω_e. We introduce the following broken Sobolev space $\mathbf{V}_h^p$

$$\mathbf{V}_h^p = \{v_h \in [L_2(\Omega)]^m : v_h|_{\Omega_e} \in [V_p^m]\ \forall \Omega_e \in \Omega\}, \tag{3.2}$$

which consists of discontinuous vector-values polynomial functions of degree p, and where m is the dimension of the unknown vector and

$$\mathbf{V}_h^p = span\Big\{\prod x_i^{\alpha_i} : 0 \le \alpha \le p, 0 \le i \le d\Big\}, \tag{3.3}$$

where α denotes a multi-index and d is the dimension of space. Then, we can obtain the following semi-discrete form by applying weak formulation on each element Ω_e, find $\mathbf{U}_h \in \mathbf{V}_h^p$, such as

$$\begin{aligned} &\frac{d}{dt}\int_{\Omega_e} \mathbf{U}_h W_h d\Omega + \int_{\Gamma_e} \mathbf{F}_k(\mathbf{U}_h)\mathbf{n}_k W_h d\Gamma - \int_{\Omega_e} \mathbf{F}_k(\mathbf{U}_h)\frac{\partial W_h}{\partial x_k} d\Omega \\ =&\int_{\Gamma_e} \mathbf{G}_k(\mathbf{U}_h)\mathbf{n}_k W_h d\Gamma - \int_{\Omega_e} \mathbf{G}_k(\mathbf{U}_h)\frac{\partial W_h}{\partial x_k} d\Omega, \qquad \forall W_h \in \mathbf{V}_h^p, \end{aligned} \tag{3.4}$$

where $\mathbf{U}_h$ and W_h represent the finite element approximations to the analytical solution $\mathbf{U}$ and the test function W respectively, and they are approximated by a piecewise polynomial function of degrees p, which are discontinuous at the cell interfaces. Assume that B is the basis of polynomial function of degrees p, this is then equivalent to the following system of N equations,

$$\begin{aligned} &\frac{d}{dt}\int_{\Omega_e} \mathbf{U}_h B_i d\Omega + \int_{\Gamma_e} \mathbf{H}_k^{(inv)}(\mathbf{U}_h^L, \mathbf{U}_h^R, \mathbf{n}_k) B_i d\Gamma - \int_{\Omega_e} \mathbf{F}_k(\mathbf{U}_h)\frac{\partial B_i}{\partial x_k} d\Omega \\ =&\int_{\Gamma_e} \mathbf{H}_k^{(vis)}(\mathbf{U}_h^L, \nabla\mathbf{U}_h^L, \mathbf{U}_h^R, \nabla\mathbf{U}_h^R, \mathbf{n}_k) B_i d\Gamma - \int_{\Omega_e} \mathbf{G}_k(\mathbf{U}_h)\frac{\partial B_i}{\partial x_k} d\Omega, \qquad 1 \le i \le N, \end{aligned} \tag{3.5}$$

where N is the dimension of the polynomial space. The inviscid and viscous flux function appearing in Eq. (3.4) is replaced by numerical flux function $\mathbf{H}_k^{(inv)}$ and $\mathbf{H}_k^{(vis)}$, respectively, where $\mathbf{U}_h^L$ and $\mathbf{U}_h^R$ are the solution polynomial at the left and right states of the cell interface.

This scheme is called discontinuous Galerkin method of degree p, or in short notation DG(P) method. In the DG framework, numerical polynomial solutions $\mathbf{U}_h$ in each

element are expressed using either standard Lagrange finite element or hierarchical node-based basis as below

$$\mathbf{U}_h = \sum_i^N \mathbf{U}_i(t) B_i(x), \tag{3.6}$$

where B_i are the finite element basis functions. In the present work, the piecewise polynomial solutions are represented using a linear Taylor series expansion at the cell centroid, which can be expressed as a combination of cell-averaged variables and their gradients at the cell centers regardless of the element shapes. As a result, the very same numerical polynomial solutions are used for arbitrary shapes of elements, which can be triangle, quadrilateral, and polygon in 2D, and tetrahedron, pyramid, prism, and hexahedron in 3D [28]. This makes the implementation of the so-called WENO (P1P2) reconstruction straightforward [29, 30]. In this scheme, a quadratic polynomial P2 solution is first obtained via a least-squares curvature reconstruction from the underlying linear P1 solution in each cell and used for evaluating the viscous fluxes. The final P2 solution is then obtained through a WENO reconstruction on the curvatures in each cell, and used for evaluating the inviscid fluxes. Note that as previous investigation indicates in [29], although the least-squares reconstructed P1P2 method has been successfully used to solve the 2D compressible Euler equations on arbitrary grids [31], yet when used to solve the 3D compressible Euler equations on tetrahedral grids, it suffers from the so-called linear instability that is also observed in the second-order cell-centered finite volume methods [33]. Such linear instability can occur even when the 3D linear equations are being solved for smooth problems on tetrahedral grids. Indeed, this linear instability is attributed to the fact that the reconstruction stencils only involve von Neumann neighborhood, that is, adjacent face-neighboring cells [32]. The linear stability can be achieved using extended stencils, which will unfortunately sacrifice the compact- ness of the underlying DG methods. Alternatively, the non-linear ENO and WENO approaches can be used to achieve both the linear and non-linear stability. It has been demonstrated that this WENO (P1P2) scheme is able to deliver the designed third-order spatial accuracy for steady-state flow problems without much extra cost in computing time and storage than the underlying second-order DG (P1) method, and also eliminate linear instability on unstructured tetrahedral grids [29]. In addition, it is necessary to point out that although the standard P2 method is in general more accurate than the reconstruction based P2 method in terms of spatial discretization errors, both the methods are able to achieve a third-order grid convergence rate [33].

4 Rosenbrock-Wanner temporal discretization

The P1P2 reconstructed discontinuous Galerkin approximation to the Navier-Stokes equations will lead to the following semi-discrete system of non-linear equations:

$$\mathbf{M}\frac{\mathrm{d}\mathbf{U}}{\mathrm{d}t} = \mathbf{R}(\mathbf{U}), \tag{4.1}$$

where $\mathbf{M}$ denotes the mass matrix, $\mathbf{U}$ the global solution vector containing $N_e \times N_q \times N_d^{P1}$ degrees of freedom (DoFs) for the underlying discontinuous Galerkin P1 solution (N_e is the number of elements, N_q the number of equations, N_d^{P1} the number of degrees of freedom for DG (P1)), and $\mathbf{R}$ the residual vector. In the case of rDG (P1P2) for the 3D Navier-Stokes equations on tetrahedral elements, $N_q = 5$, $N_d^{P1} = 4$, and $N_d^{P2} = 10$. As we already stressed, one of the most advantageous features of rDG method is the low memory needed for storing the Jacobians when using implicit time integration methods, e.g., $(5\times 4)^2$ for rDG (P1P2) versus $(5\times 10)^2$ for DG (P2) for a single elemental diagonal matrix.

Rosenbrock-type schemes are part of a class of linearly implicit Runge-Kutta (IRK) time integration methods. In general, the formula of an s-stage Rosenbrock-type scheme to integrate Eq. (4.1) in time can be written as

$$(\mathbf{I}-\Delta t\gamma_{ii}\mathbf{J})\mathbf{U}^{(i)} = \Delta t\mathbf{R}(\mathbf{U}^n + \sum_{j=1}^{i-1}\alpha_{ij}\mathbf{U}^{(j)}) + \Delta t\mathbf{J}\sum_{j=1}^{i-1}\gamma_{ij}\mathbf{U}^{(j)}, \quad i=1,\cdots,s. \tag{4.2}$$

As shown in Eq. (4.2), the ROW methods are derived by linearizing a diagonally IRK (DIRK) scheme, and replace the non-linear systems with a sequence of linear systems, in which some attractive properties in terms of stability are lost. However, as a trade-off, the computational costs per time step are reduced in Eq. (4.2): s linear equation systems with a constant coefficient matrix and different right-hand-sides need to be solved, instead of s non-linear systems. They can be easily used along with the variable time step sizes and constructed to be A- and L-stable to obtain formal order of temporal accuracy [26,27].

In Eq. (4.2), the Jacobian matrix $\mathbf{J}$ is approximately computed using an automatic differentiation toolkit TAPENADE [34]. In many systems, the exact Jacobian matrix can be both costly and difficult to obtain, e.g., due to the size of the application and the use of complex spatial discretization schemes. The class of Rosenbrock-Wanner methods have been derived for such situations, which is aimed to not only reduce the computational costs within each time step, but also alleviate the impact from how the Jacobian matrix is formulated. They have the same form in Eq. (4.2), but the coefficients are selected such that the overall discretization order is preserved for approximate Jacobian matrix.

The solution at the next time step $\mathbf{U}^{n+1}$ is determined as below:

$$\mathbf{U}^{n+1} = \mathbf{U}^n + \sum_{j=1}^{s} b_j\mathbf{U}^{(j)}, \tag{4.3}$$

where the coefficients α_{ij}, γ_{ij} and b_j are generally shown in Refs. [26,27]. For implementation purposes, Eq. (4.2) and Eq. (4.3) can be rewritten by introducing a new variable vector $\mathbf{W}^{(i)}$:

$$\mathbf{W}^{(i)} = \sum_{j=1}^{i}\gamma_{ij}\mathbf{U}^{(j)}, \quad i=1,\cdots,s, \tag{4.4}$$

in order to avoid the matrix-vector multiplication $\mathbf{J}\sum\gamma_{ij}\mathbf{U}^{(j)}$, which could result from the direct implementation of Eqs. (4.2) and (4.3), as it requires the solution of a linear system

with the matrix $(\mathbf{I}-\Delta t\gamma_{ii}\mathbf{J})$ as well as the matrix-vector multiplication $\mathbf{J}\sum\gamma_{ij}\mathbf{U}^{(j)}$. Note that if $\gamma_{ij}\neq 0$ for $j\leq i$, then the matrix $\mathbf{\Gamma}=(\gamma_{ij})$ is invertible and $\mathbf{U}^{(i)}$ can be determined from $\mathbf{W}^{(i)}$ with

$$\mathbf{U}^{(i)}=\frac{1}{\gamma_{ii}}\mathbf{W}^{(i)}-\sum_{j=1}^{i-1}c_{ij}\mathbf{W}^{(j)}, \tag{4.5}$$

where $\mathbf{C}$ is given by $\mathbf{C}=diag(\gamma_{11}^{-1},\cdots,\gamma_{ss}^{-1})-\mathbf{\Gamma}^{-1}$. Thus the following formulation of the ROW method is found in practical implementations,

$$\mathbf{L}\mathbf{W}^{(i)}=\mathbf{F}\left(\mathbf{U}^{n}+\sum_{j=1}^{i-1}a_{ij}\mathbf{W}^{(j)}\right)+\frac{1}{\Delta t}\sum_{j=1}^{i-1}c_{ij}\mathbf{W}^{(j)},\quad i=1,\cdots,s, \tag{4.6}$$

with $\mathbf{L}=(\frac{1}{\gamma\Delta t}\mathbf{I}-\mathbf{J})$, and $\gamma=\gamma_{ii}$, thus $\mathbf{L}$ is constant for consecutive stages of the Rosenbrock scheme. The solution at the next time step $\mathbf{U}^{n+1}$ is given by

$$\mathbf{U}^{n+1}=\mathbf{U}^{n}+\sum_{j=1}^{s}m_j\mathbf{W}^{(j)}, \tag{4.7}$$

where the coefficients $(a_{ij})=(\alpha_{ij})\mathbf{\Gamma}^{-1}$, and $(m_1,\cdots,m_s)=(b_1,\cdots,b_s)\mathbf{\Gamma}^{-1}$.

In this paper, we mainly focus on the study of three third-order ROW methods: 1)ROSI2PW method (I2 for index 2 problems, P for semi-discretized PDE problems, W for W-method), 2) third-order ROS34PRW method (an extension of ROS34PW2 from index-1 DAEs to index-2 DAEs), and 3) ROS34PW2 method. The coefficients for these methods can be found in [26, 27], which also have been listed at the end of the paper. An efficient LU-SGS preconditioned GMRES solver [18], namely GMRES+LU-SGS, is then applied for solving the linearized system of equations in Eq. (4.6).

5 Numerical examples

A number of test cases are presented in order to assess the performance of the Rosenbrock schemes for computing the unsteady flows. In addition, comparative studies between the third-order ROW and ESDIRK3 schemes are conducted using the fixed time step sizes, where the related results of a four-stage, third-order ESDIRK3 scheme are referred to Xia et al. [22]. We also make comparison between ROW for index-1 and ROW for index-2 in case 5.2 and case 5.3. Since the relative error tolerances for the linear and non-linear systems solution have great influence on the efficiency and accuracy of a scheme, we investigate such influence by the case of convection of an isotropic vortex. The L_2-norm by Eq. (5.3), is used as the criterion to determine whether the relative tolerance is sufficient enough to get the converged solution. Here, the relative error tolerance 1E-n in the tables means that the residual of the iteration is dropped n orders of magnitude.

Based on the Tables 1 and 2, the relative error tolerances for the non-linear and linear iterations are determined for all the test cases in this paper, shown in Table 3.

Table 1: The investigation of relative error tolerances for the non-linear and linear iterations about the ESDIRK3.

Scheme	Non-linear	Linear	log(L2-norm)
ESDIRK3 × GMRES+LU-SGS	1E-3	1E-2	-4.8400
	1E-2	1E-2	-4.7073

Table 2: The investigation of relative error tolerances for the linear iterations about the ROW.

Scheme	Non-linear	Linear	log(L2-norm)
ROSI2PW × GMRES+LU-SGS	–	1E-4	-4.8190
	–	1E-3	-4.8182
	–	1E-2	-4.7970
ROS34PRW × GMRES+LU-SGS	–	1E-4	-4.8190
	–	1E-3	-4.8150
	–	1E-2	-4.38631

Table 3: Relative error tolerance for the non-linear and linear iterations.

Scheme	Non-linear	Linear
ESDIRK × GMRES+LU-SGS	1E-3	1E-2
Rosenbrock-Wanner × GMRES+LU-SGS	–	1E-3

5.1 Convection of an isentropic vortex in inviscid flows

In this test case,we consider the passive convection of a quasi-2D inviscid isentropic vortex [9, 35] in order to assess the temporal convergence for the Rosenbrock and ESDIRK methods. The spatial discretization is carried out using a third-order rDG(P1P2) method. The analytical solution to this problem at any time t is simply the passive advection of the initial solution at $t=0$, which provides a valuable reference for measuring the accuracy of a numerical solution. The initial condition is a linear superposition of a mean uniform flow with some perturbations. The free stream flow conditions are $(\rho_\infty,u_\infty,v_\infty,p_\infty)=(1,1,0,1)$. The perturbations of the velocity components u and v, entropy S, and temperature T for the vortex are given by

$$\delta T=-\frac{(\gamma-1)}{8\gamma\pi^2}e^{1-\gamma^2},\quad \begin{pmatrix}\delta u\\ \delta v\end{pmatrix}=\frac{\varepsilon}{2\pi}e^{\frac{1-r^2}{2}}\begin{pmatrix}-(y-y_0)\\ (x-x_0)\end{pmatrix},\quad \delta S=0, \tag{5.1}$$

where $r^2=(x-x_0)^2+(y-y_0)^2$, (x_0,y_0) is the coordinate of the vortex center, and ε is the vortex strength. From $\rho=\rho_\infty+\delta\rho$, $u=u_\infty+\delta u$, $v=v_\infty+\delta v$, $T=T_\infty+\delta T$, and the isentropic

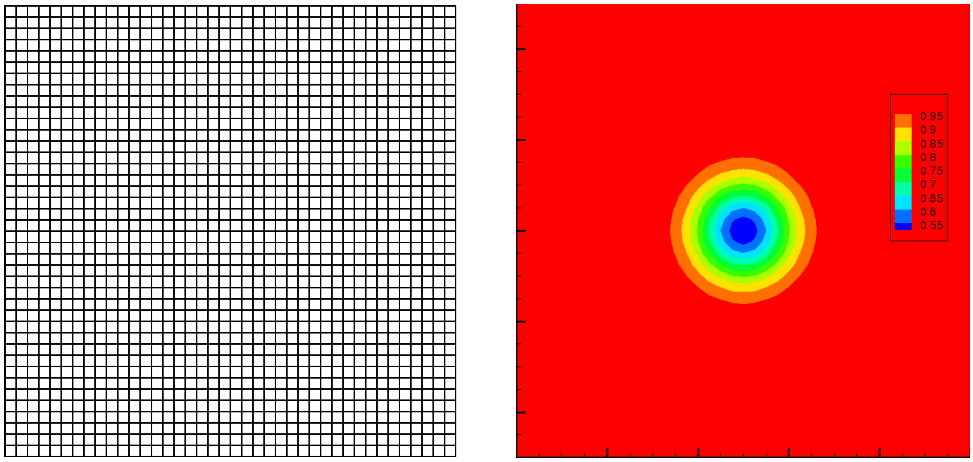
relation, other physical variables can be determined as follows:

$$\begin{aligned}
&\rho = T^{\frac{1}{(\gamma-1)}} = (T_\infty + \delta T)^{\frac{1}{\gamma-1}} = \left[T_\infty - \frac{(\gamma-1)}{8\gamma\pi^2} e^{1-\gamma^2}\right]^{\frac{1}{\gamma-1}},\\
&\rho u = \rho(u_\infty + \delta u) = \rho\left[u_\infty - \frac{\varepsilon}{2\pi} e^{\frac{1-r^2}{2}}(y-y_0)\right],\\
&\rho v = \rho(v_\infty + \delta v) = \rho\left[v_\infty + \frac{\varepsilon}{2\pi} e^{\frac{1-r^2}{2}}(x-x_0)\right],\\
&p = \rho^\gamma,\\
&e = \frac{p}{\rho(\gamma-1)} + \frac{1}{2}(u^2+v^2).
\end{aligned} \tag{5.2}$$

The vortex strength $\varepsilon = 5$, and the coordinate of the vortex center $(x_0, y_0) = (5,0)$. The computational domain Ω is $[0,10]\times[-5,5]$ and the periodic boundary conditions are imposed. The numerical solutions are obtained after one period, $t = 10$, and compared with the analytical solution simply given by the initial condition. The following L_2-norm

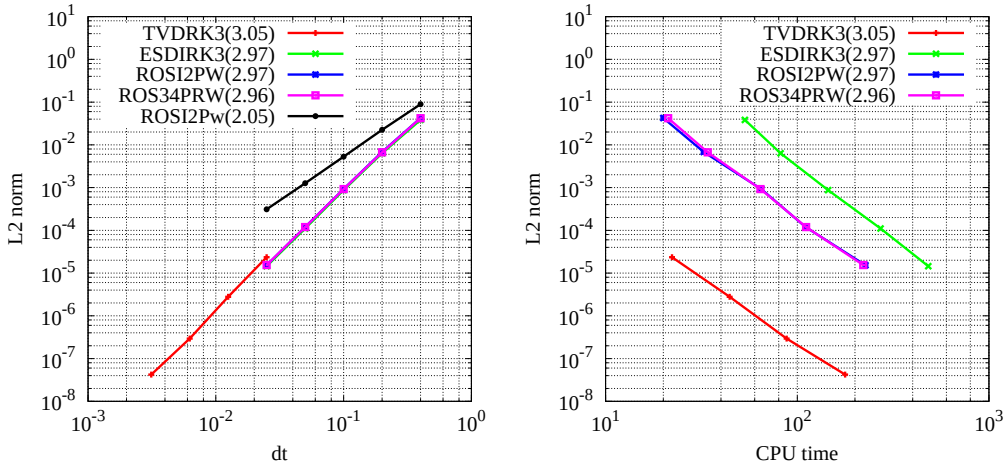
$$\| \rho - \rho^{\mathrm{r}} \|_{L^2(\Omega)} = \sqrt{\int_\Omega (\rho - \rho^{\mathrm{r}})^2 \, \mathrm{d}\Omega}, \tag{5.3}$$

is used to measure the error between the numerical and analytical solutions, where ρ is the numerical solution for the density and ρ^{r} is the reference one. Because the overall error is due to both spatial and temporal errors, a reference solution is obtained using a small time-step size at $\Delta t = 0.001$ in order to eliminate the effect of spatial error and in order to isolate the temporal error. Fig. 1(a) shows the hexahedral grid used in the computation, which consists of 1600 hexahedral cells. A series of successively refined time-step sizes $\Delta t = 0.4$, 0.2, 0.1, 0.05, and 0.025 are used in calculation for the temporal convergence study. Fig. 2(a) provides the details of the temporal convergence for this numerical experiment. The ESDIRK3, ROSI2PW, and ROS34PRW schemes exhibit a slope of 2.97, 2.97 and 2.96, respectively, indicating that these three methods are able to offer the nearly third-order accuracy of temporal convergence. However, the ROSI2Pw scheme only exhibits a slope of 2.0, which is consistent to the theory that this scheme would require the exact Jacobians in order to achieve the formal order of temporal accuracy. Fig. 2(b) indicates that third order ROW consumes much less computing time than ESDIRK3 to achieve the same level of temporal errors. For reference purpose, the results obtained by the explicit three-stage third-order TVD Runge-Kutta (TVDRK3) time integration scheme are also presented in Figs. 2(a) and 2(b). Note that the explicit methods are usually more efficient than their implicit counterparts for the type of problems like in this test case, where the physical time scales would not allow the implicit methods to use very large time step sizes while trying to retain the wave propagation accurately.



(a) (b)

Figure 1: (a) The computational grid used for convection of an isentropic vortex in inviscid flow. (b) Density distribution for ESDIRK3 at $t=10$.

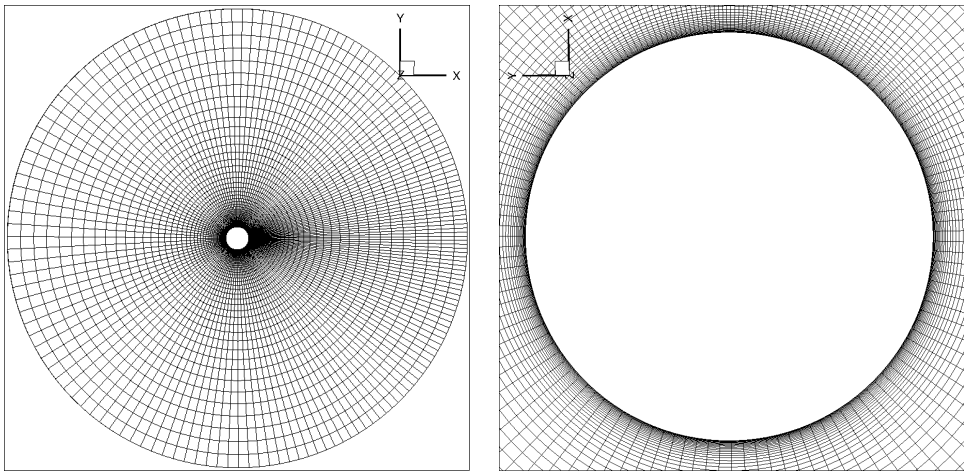


(a) L_2 norm vs. dt (b) L_2 norm vs. CPU time

Figure 2: Temporal convergence histories of WENO (P1P2) with different time integration schemes obtained for convection of an isentropic vortex. (a) L_2 norm versus dt and (b) L_2 norm versus CPU time.

5.2 von-Karman vortex street

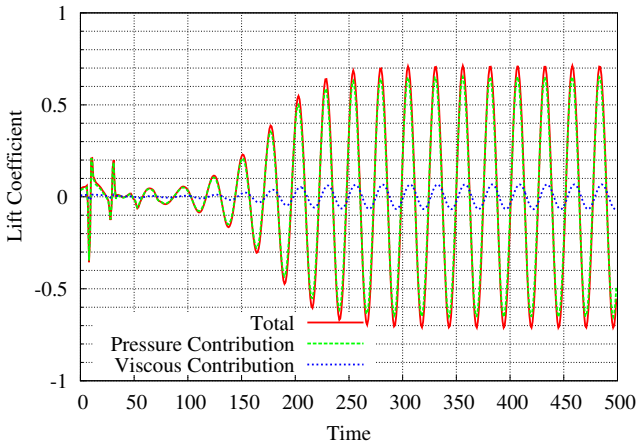
The von-Karman vortex street is probably one of the most extensively studied cases both experimentally and numerically in fluid dynamics. In this test case, the flow conditions are chosen (Reynolds number of $Re_\infty = 200$ based on the cylinder diameter of d=1.0) such that vortex shedding is expected to occur downstream of the cylinder. This simulation provides a geometrically simple, yet aerodynamically complex test for our solver. Fig. 3



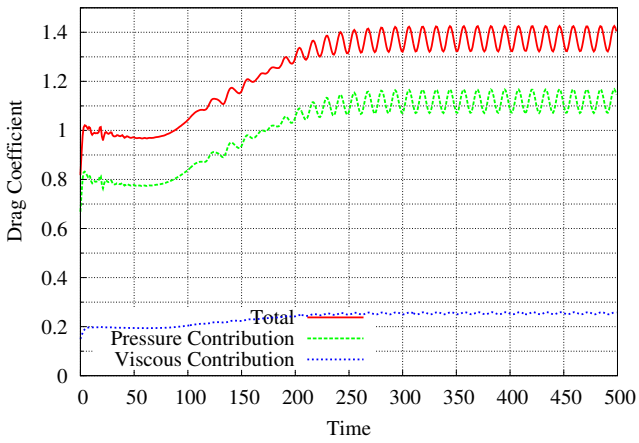
(a) Global view (b) Zoom-in view

Figure 3: Mesh used for computing a viscous flow past a cylinder (nelem=10204, npoin=0800, nboun=20800).

shows the grid used in the computations, which consists of 10,204 hexahedral elements, 20,800 grid points, and 20,800 boundary faces. Note that this 3D grid is extruded from a 2D grid used in [36]. The normal grid spacing near the cylinder surface is 0.001 (normalized by the cylinder diameter), as demonstrated in Fig. 3(b). The free-stream Mach number M_∞ is set as 0.2. We employ the steady-state solution obtained for $Re_\infty = 50$ (for which vortex shedding is not expected to occur) and an angle of attack $\alpha = 3^\circ$ as the initial solution for vortex shedding. To start the simulation of shedding, the flow conditions at the characteristic boundaries are set to $Re_\infty = 200$ and $\alpha = 0^\circ$ (the asymmetry of the initial flow, or sudden change of the angle of attack, leads to the vortex shedding behavior). The solutions were advanced in time with a fixed time-step size of $\Delta t = 0.5$ for ROS34PRW + rDG (P1P2), $\Delta t = 0.1$ for ROSI2PW + rDG (P1P2) and $\Delta t = 0.02$ for ROS34PW2 + rDG (P1P2), respectively, until 10 shedding cycles had passed to ensure that the solution was periodic in time. Note that in this test case, a much smaller allowable time-step size has to be used for ROSI2PW and ROS34PW2, which is determined considering the stability rather than the accuracy. Time histories of the computed surface drag and lift coefficients are presented in Fig. 4 obtained by ROS34PRW + rDG (P1P2), and in Fig. 5 obtained by ROSI2PW + rDG (P1P2). The Strouhal number is 1.923. These results agree well with experimental measurements and numerical results in the literature [36]. To assess the efficiency of the ROW methods, the computational costs for both ROW and ESDIRK3 solutions are listed in Table 4. Firstly, the comparison between ESDIRK3 and ROW methods is presented here. In comparison with ESDIRK3, ROS34PRW requires 45% CPU time less than ESDIRK3, using more strict criterion for GMRES. This is caused by that just linear systems need to be solved for ROW schemes, leading to that the computational costs per time step are reduced. Secondly, it could be observed that the ROW for index-2 DAEs



(a) Lift coefficient

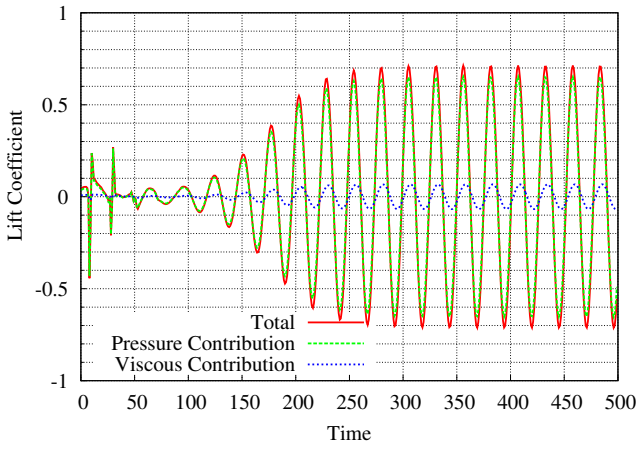


(b) Drag coefficient

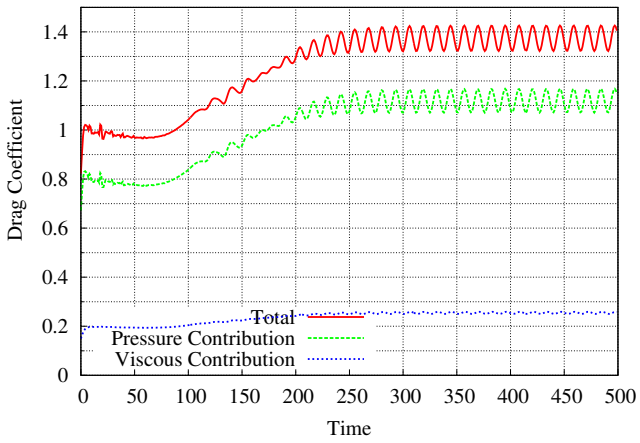
Figure 4: Time history of the computed lift and drag coefficients by ROS34PRW + rDG(P1P2) for flow past a cylinder at $M_\infty = 0.2$ and $Re_\infty = 200$.

Table 4: Comparison of the CPU time (evaluated by running on 256 cores) between the ROW and ESDIRK3 for computing a viscous flow past a cylinder.

Time integration method	Time-step size	Time steps	CPU time
ESDIRK3	0.50	1000	965.1
ROSI2PW	0.10	5000	1555.8
ROS34PRW	0.50	1000	531.64
ROS34PW2	0.02	25000	4356.0



(a) Lift coefficient



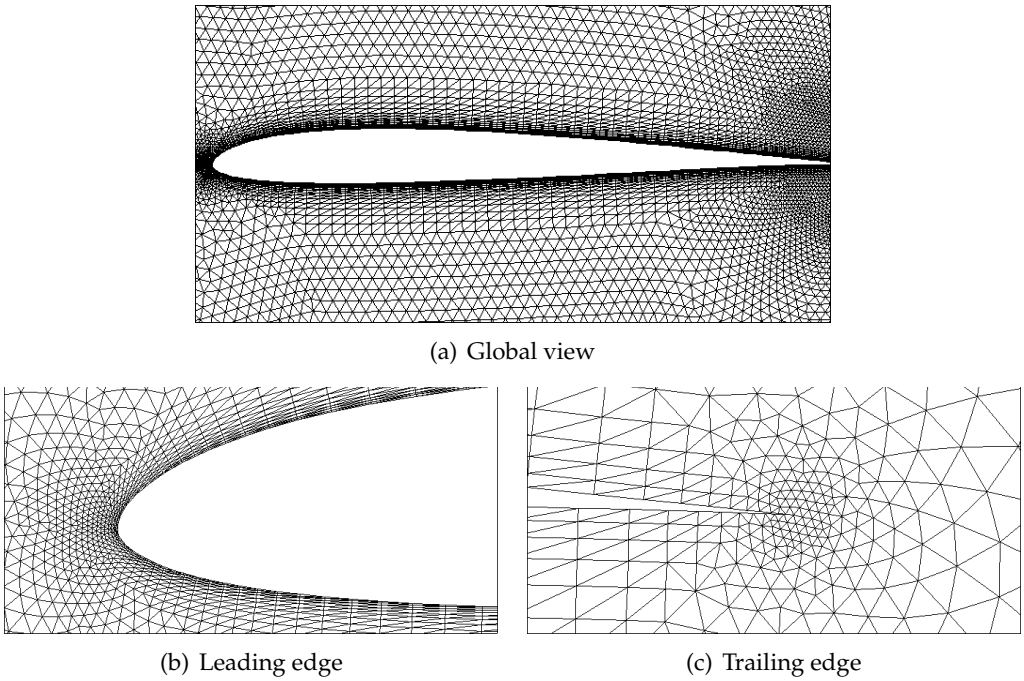
(b) Drag coefficient

Figure 5: Time history of the computed lift and drag coefficients by ROSI2PW + rDG(P1P2) for flow past a cylinder at $M_\infty = 0.2$ and $Re_\infty = 200$.

outperforms the one for index-1 in terms of the computation efficiency. ROS34PRW and ROSI2PW requires 88% and 64% CPU time less than ROS34PW2, respectively. This could be explained as follows, some ROW schemes have to use small time step to preserve their stability. Above all, ROS34PRW demonstrated the best efficiency in this test case.

5.3 Flow past a SD7003 airfoil

In this subsection, we consider a viscous flow past an SD7003 airfoil at $M_\infty = 0.2$, $\alpha = 4^\circ$, and $Re_\infty = 10{,}000$, as an abundance of numerical results can be found about this test case [35,37]. The computation is initialized with constant values in the entire domain with



(a) Global view

(b) Leading edge

(c) Trailing edge

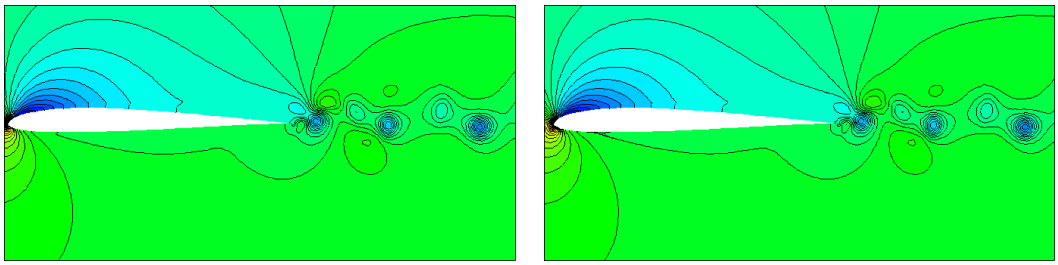
Figure 6: Grid used for computing the unsteady viscous flow past a SD7003 airfoil.

no-slip, adiabatic boundary conditions on the solid wall, and terminated at $t=125$. Fig. 6 shows the grid used in the computation, which consists of 50, 781 prismatic elements, 52, 176 grid point, 101, 562 triangular boundary faces, and 279 quadrilateral boundary faces.

Typical computed pressure contours in the flow field are displayed in Fig. 7. The instantaneous vorticity contours are displayed in Fig. 8, which capture the key flow features: separation of the flow on the upper surface of the airfoil and shedding of the tailing vortices.

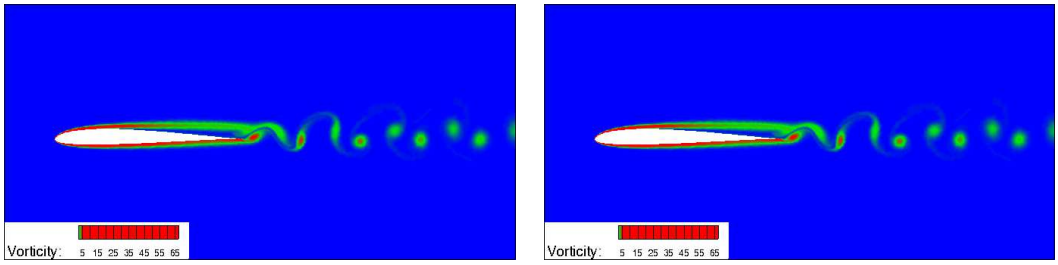
Fig. 9 shows the computed velocity vectors in the flow field, where the development of the boundary layers and flow separation on the upper surface of the airfoil are clearly visible.

From Ref. [37],the initial transient is over by $t=75$ as estimated from the forces on the wing. Therefore, unless otherwise specified, the instantaneous forces from $t=100$ to 125 are presented in Fig. 10, which could be comparable to Ref. [37]. The computation is performed using ROSI2PW, ROS34PRW, and ROS34PW2 with a fixed time-step size of $\Delta t=0.05$, 0.05, and 0.05 respectively. This time-step size is consistent with that in the reference, so that we could make some comparison directly. The instantaneous solutions are written every 0.25 second for time-averaging calculations, starting from $t=100$ when the flow separation on the upper surface of the airfoil is considered to have fully developed. It could be observed that, the lift and drag coefficient histories are very similar to that in Ref. [37].



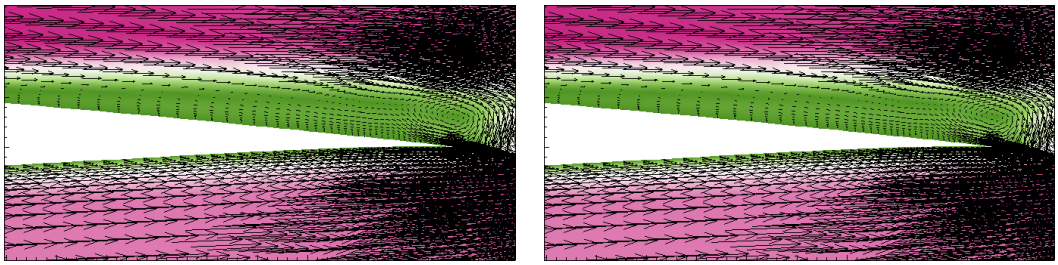
(a) ROSI2PW + rDG(P1P2), $\Delta t = 0.05$ (b) ROS34PRW + rDG(P1P2), $\Delta t = 0.05$

Figure 7: Computed pressure contours in the flow field for flow past a SD7003 airfoil at $M_\infty = 0.2$, $\alpha = 4^\circ$, and $Re_\infty = 10{,}000$.



(a) ROSI2PW+rDG(P1P2), $\Delta t = 0.05$ (b) ROS34PRW+rDG(P1P2), $\Delta t = 0.05$

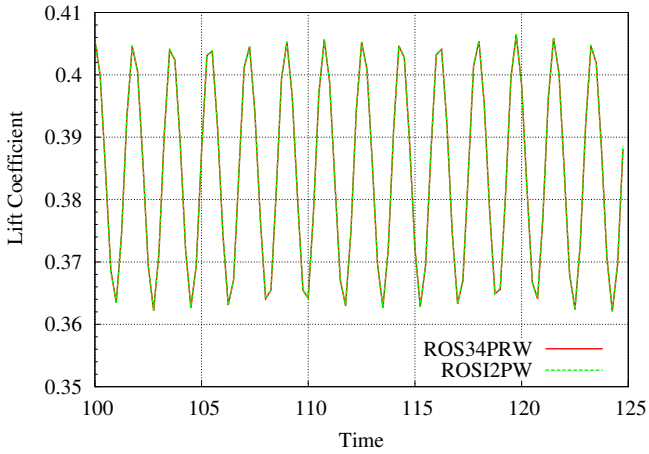
Figure 8: Computed vorticity contours in the flow field for flow past a SD7003 airfoil at $M_\infty = 0.2$, $\alpha = 4^\circ$, and $Re_\infty = 10{,}000$.



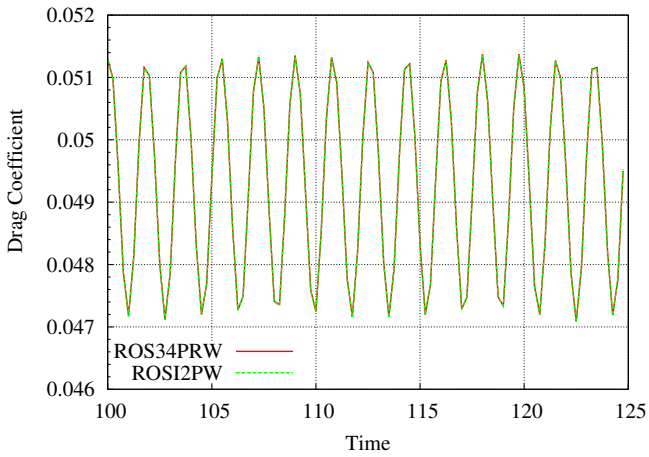
(a) ROSI2PW+rDG(P1P2), $\Delta t = 0.05$ (b) ROS34PRW+rDG(P1P2), $\Delta t = 0.05$

Figure 9: Computed velocity vector in the flow field for flow past a SD7003 airfoil at $M_\infty = 0.1$, $\alpha = 4^\circ$, and $Re_\infty = 10{,}000$.

Finally, in order to demonstrate the overall effectiveness of ROW + rDG (P1P2) on highly stretched grids, the computational costs for both the ROW and ESDIRK3 solutions are presented in Table 5. Compared to ESDIRK3, all ROW method require roughly 33% CPU time less; For ROW method, the CPU time for index-1 and index-2 is similar.



(a) Lift coefficient



(b) Drag coefficient

Figure 10: Time variation of lift (left) and drag (right) coefficients at $Re = 10{,}000$: comparison between ROSI2PW and ROS34PRW.

Table 5: Comparison of the CPU time (evaluated by running on 256 cores) between the ROW and ESDIRK3 for computing a viscous flow past an SD7003 airfoil.

Time integration method	Time-step size	Time steps	CPU time
ESDIRK3	0.05	500	711.9
ROSI2PW	0.05	500	474.04
ROS34PRW	0.05	500	474.93
ROS34PW2	0.05	500	470.0

5.4 Lid driven cavity in 3D

A transitional flow in a 3D lid driven cavity at $Re_\infty = 10{,}000$ is considered in this test problem, as the accuracy of the numerical results obtained can be assessed through comparison with both the experimental data by Prasad and Koseff [38] and the LES data by Zang et al. [39].

The cavity dimensions are 1 unit in the stream-wise x direction and vertical y direction, and 0.5 unit in the span-wise z direction. A hexahedral grid consisting of $64\times64\times32$ grid points is used in computation, as shown in Fig. 11. While being equally distributed in the z-direction, the grid points are clustered near the walls in the x-y plane, with the grid spacing geometrically stretched away from the wall with the first element thickness being 0.005 ($y^+ = 3.535$). On the bottom and side walls, the no-slip, adiabatic boundary conditions are prescribed. Along the top "lid", the no-slip, adiabatic boundary conditions with a lid velocity $v_b = (0.2,0,0)$ are prescribed as to ensure an essentially incompressible flow field. The computation is conducted in two stages. At stage I, the computation is started with a zero-velocity field, and sufficient steps are taken to evolve the field into a cyclically oscillating state by using BDF1 + rDG(P1P2) with $CFL=500$. We used the solution obtained at the end of stage I as the initial solution for stage II, and run 30,000 time steps with a fixed $\Delta t=0.1$, during which the instantaneous solutions are written every 300 time steps for time averaging calculations. For a comparative study, the following two options: 1) ROSI2PW + rDG (P1P2), 2) ROS34PRW + rDG (P1P2) are used respectively at stage II. The computed mean velocity and components of Reynolds stress along the center-lines on the span-wise mid-plane are presented in Figs. 12-14 respectively. Those profiles were obtained by using a linear polynomial interpolation of the elemental solu-

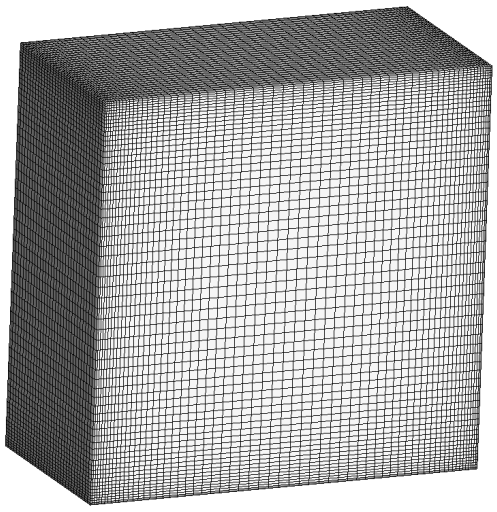


Figure 11: The hexahedral grid ($64\times64\times32$ points) for the LES of a lid driven cavity ($x:y:z=1:1:0.5$) at $Re=10{,}000$.

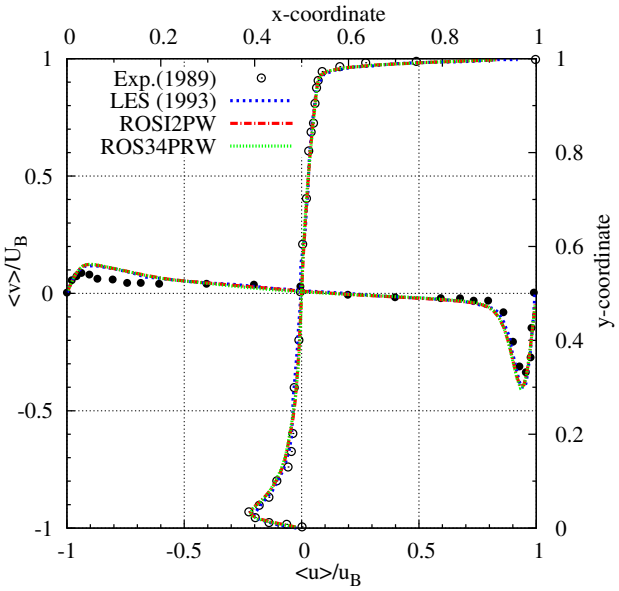


Figure 12: Comparison of the normalized mean velocity components $<u>/u_B$ and $<v>/u_B$ in the span-wise mid-plane with the classical experimental data by Prasad & Koseff (1989) and numerical data by Zang et al. (1993), for the LES of a lid driven cavity $(x:y:z=1:1:0.5)$ at $Re=10{,}000$.

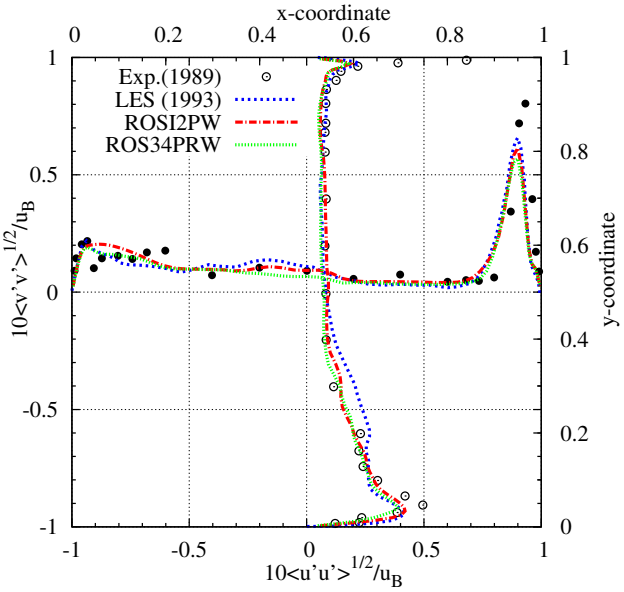


Figure 13: Comparison of the scaled RMS velocity components $10<u'u'>^{0.5}/u_B$ and $10<v'v'>^{0.5}/u_B$ in the span-wise midplane with the classical experimental data by Prasad & Koseff (1989) and numerical data by Zang et al. (1993), for the LES of a lid driven cavity $(x:y:z=1:1:0.5)$ at $Re=10{,}000$.

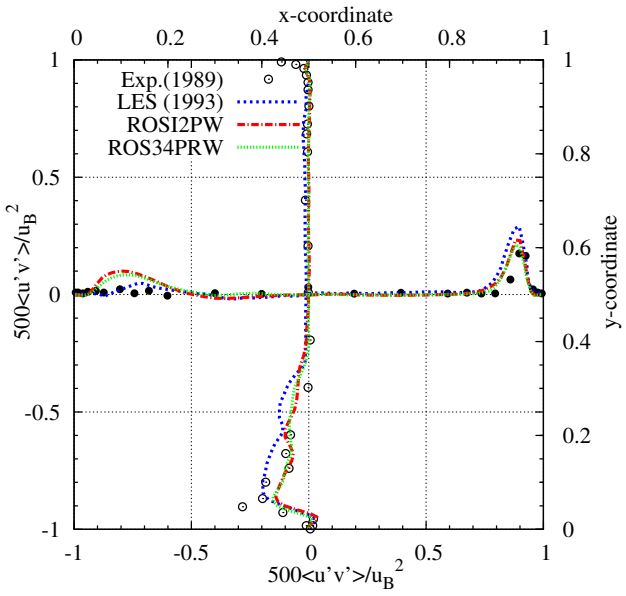


Figure 14: Comparison of the scaled mean Reynolds stress tensor components $500<u'v'>/u_B^2$ in the span-wise mid-plane with the classical experimental data by Prasad & Koseff (1989) and numerical data by Zang et al. (1993), for the LES of a lid driven cavity $(x:y:z=1:1:0.5)$ at $Re=10{,}000$.

tions at the intersected nodes cut through by the center-lines. For example, in Fig. 12, the mean x-velocity u is plotted along the vertical center-line (vs. y-coordinate), and the mean y-velocity v is plotted along the horizontal center-line (vs. x-coordinate). As clearly seen from Figs. 12 to 14, these results from the ROSI2PW and ROS34PRW agree well with experimental data and LES data. Furthermore, the high efficiency of the ROW methods is demonstrated in Table 6, where the costs for both the ROW and ESDIRK3 solutions are compared.

Table 6: Comparison of the CPU time (evaluated by running on 256 cores) between the ROW and ESDIRK3 for computing the LES of a lid driven cavity $(x:y:z=1:1:0.5)$ at $Re=10{,}000$.

Time integration method	Time-step size	Time steps	CPU time
ESDIRK3	0.1	30000	164988.0
ROSI2PW	0.1	30000	103478.0
ROS34PRW	0.1	30000	100998.6

Compared to ESDIRK3, ROW method require roughly 38% CPU time less. We also could observe that, with the identical time step, the computational cost for ROW methods ROSI2PW and ROS34PRW are not the same. This is caused by that, for different methods, different iteration numbers are required to satisfy the tolerance criterion for GMRES.

5.5 Direct numerical simulation of the Taylor-Green vortex

The Taylor-Green vortex flow problem, one of the benchmark cases in the 3^{rd} International Workshop on high order CFD methods, is chosen in this test case to assess the accuracy and performance of ROW for the Direct Numerical Simulation (DNS) of turbulent flows. This problem was originally designed to numerically study the dynamics of turbulence. The initial conditions are smooth, but the flow quickly transits to turbulence with the creation of small scales and begins to decay, mimicking homogeneous non-isotropic turbulence. The initial conditions are given by:

$$\begin{aligned} &u_1 = V_0 \sin\left(\frac{x}{L}\right)\cos\left(\frac{y}{L}\right)\cos\left(\frac{z}{L}\right), \\ &u_2 = -V_0 \cos\left(\frac{x}{L}\right)\sin\left(\frac{y}{L}\right)\cos\left(\frac{z}{L}\right), \\ &u_3 = 0, \\ &p = p_0 + \rho_0 {V_0}^2 \left[\frac{1}{16}\left(\cos\left(\frac{2x}{L}\right)+\cos\left(\frac{2y}{L}\right)\right)\left(\cos\left(\frac{2z}{L}\right)+2\right)\right], \end{aligned} \tag{5.4}$$

where $\rho_0 = 1$, $p_0 = 1/\gamma$, and u_1, u_2, and u_3 are the components of the velocity in the x-, y- and z-directions respectively, and p is the pressure. The flow is initialized to be isothermal ($p/\rho = p_0/\rho_0 = RT_0$). To minimize the effects of compressibility, the free-stream Mach number is set to 0.1. The Reynolds number in this case is 1,600, which corresponds to a peak Taylor microscale Reynolds number of about 22. The flow is computed in a periodic and square box, which spans $[-\pi L, \pi L]$ in each coordinate direction. The physical duration of the computation is 20 based on the characteristic convective time defined as $t_c = L/V_0$, i.e., $t_{\texttt{final}} = 20t_c$. From the numerical experiment [35], one hexahedral mesh, which have 204^3 elements with degrees of freedom 324^3, is used here, which are distributed equally in the three directions. The best way to visualize these structures is to plot positives values of Q. Fig. 15 shows the computed vortex detection criterion Q at $t = 8t_c$ using ROSI2PW method based on rDG(P1P2). One can observe that the vortex structure obtained by ROSI2PW method looks very similar to the one from Ref. [40]. Fig. 16 and Fig. 17 compare the time history of the kinetic energy and the kinetic energy dissipation rate computed from the data at the space-time quadrature points, respectively, with the result from an incompressible simulation using a spectral code on a mesh of 512^3 grid points [41]. Using ROW, the results from the rDG (P1P2) solution on this mesh agree very well with those from the spectral code solution. Fig. 18 shows the results for the enstrophy over simulated time. As the figure shows, the enstrophy is more difficult to resolve numerically. The individual terms in the kinetic energy evolution equation can be used to assess the accuracy of the numerical solutions. The kinetic energy dissipation rate in compressible flows is given by the sum of three contributions

$$\epsilon = \epsilon_1 + \epsilon_2 + \epsilon_3 = -\frac{dE_k}{dt},$$

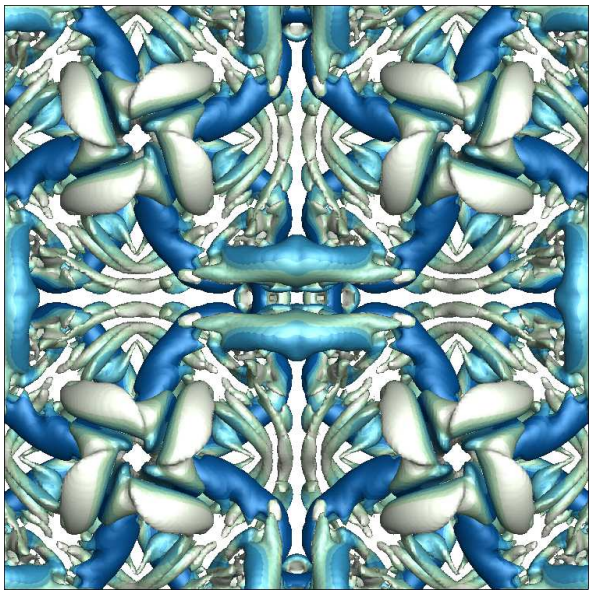


Figure 15: TGV solution using ROSI2PW method, showing isosurfaces of Q criterion colored by velocity magnitude at time $t=8t_c$.

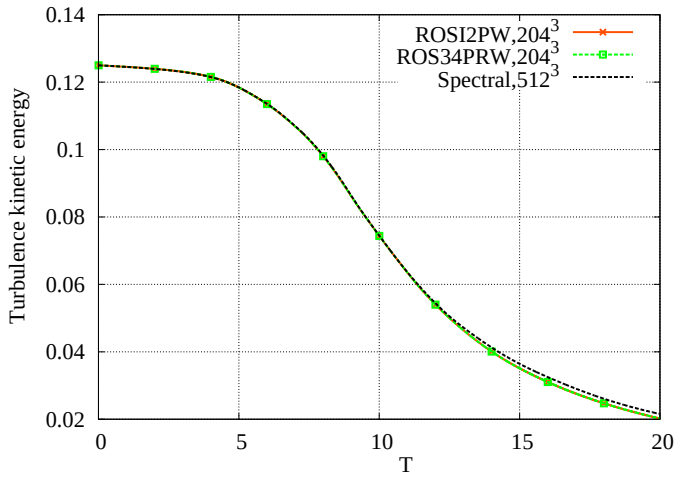


Figure 16: Evolution of the dimensionless kinetic energy as a function of the dimensionless time.

where

$$\epsilon_1=\frac{1}{\Omega}\int 2\mu s_{ij}s_{ij}d\Omega, \qquad \epsilon_2=\frac{1}{\Omega}\int 2\mu_v u_{kk}u_{kk}d\Omega, \qquad \epsilon_3=-\frac{1}{\Omega}\int pu_{kk}d\Omega,$$

where $s_{ij}=\frac{1}{2}(u_{ij}+u_{ji})$ is the strain-rate tensor. In this case, the gas is assumed to have zero bulk viscosity. Therefore, the dissipation due to the bulk viscosity is always equal to zero, meaning that $\epsilon_2=0$. Since the flow is nearly incompressible, the dissipation due

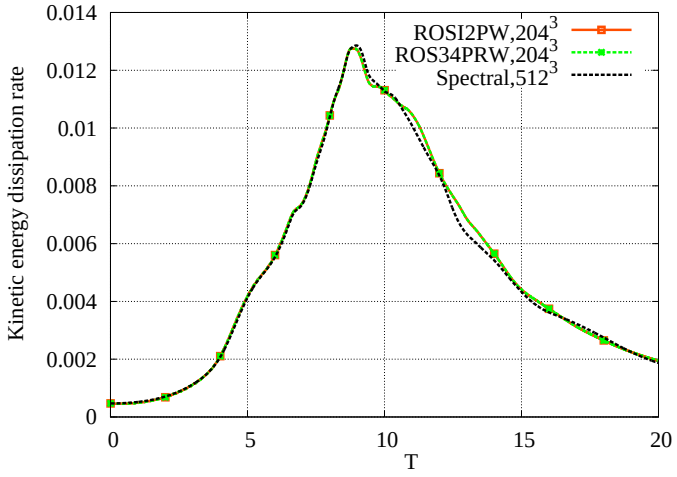


Figure 17: Evolution of the dimensionless kinetic energy dissipation rate as a function of the dimensionless time.

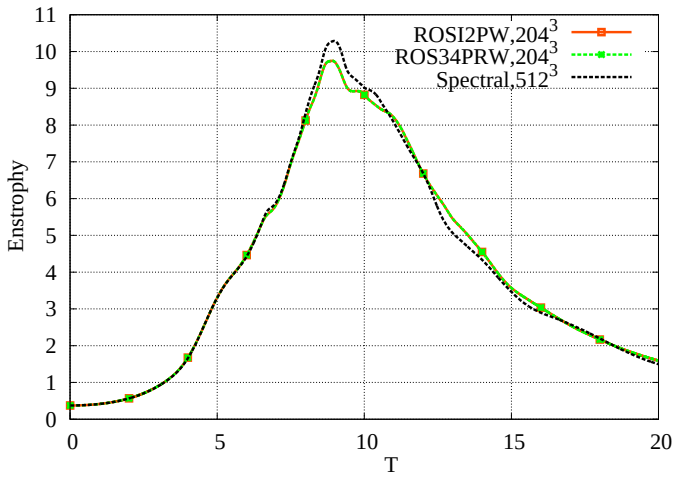


Figure 18: Evolution of the dimensionless enstrophy as a function of the dimensionless time.

to the pressure-dilatation term (ϵ_3) is expected to be tiny. The kinetic energy dissipation rate is then approximately equal to $\epsilon = \epsilon_1$. However, for the compressible simulation, this does not hold exactly. The pressure-dilatation, ϵ_3, has a significant bias, contributing net positive kinetic energy dissipation. Compressibility effects are evident in oscillations of the pressure dilatation term (ϵ_3). However, the biased pressure-dilatation term decreases toward zero as shown in Fig. 19. Time histories of the computed ϵ, ϵ_1, and ϵ_3 on the this mesh are presented in Fig. 20. The high effectiveness of the ROW methods is demonstrated in Table 7, where the costs for both the ROW and ESDIRK3 solutions are compared.

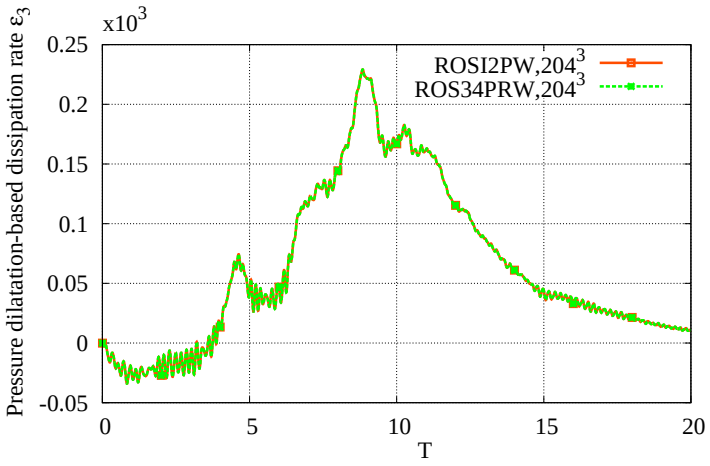


Figure 19: Biased pressure-dilatation term for the Taylor-Green vortex problem.

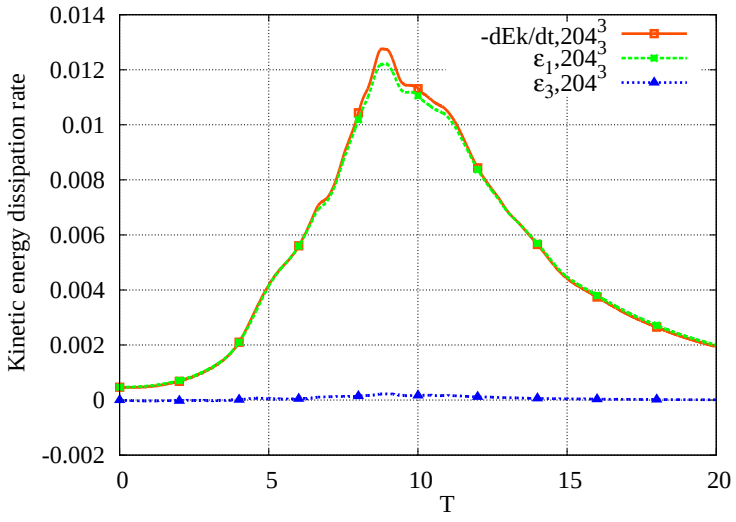
Table 7: Comparison of the CPU time (evaluated by running on 240 cores) between the ROW and ESDIRK3 for computing the Taylor-Green vortex at $Re=1600$.

Time integration method	Time-step size	Time steps	CPU time
ESDIRK3	0.08	2500	360250.0
ROSI2PW	0.04	5000	269998.7
ROS34PRW	0.04	5000	274921.4

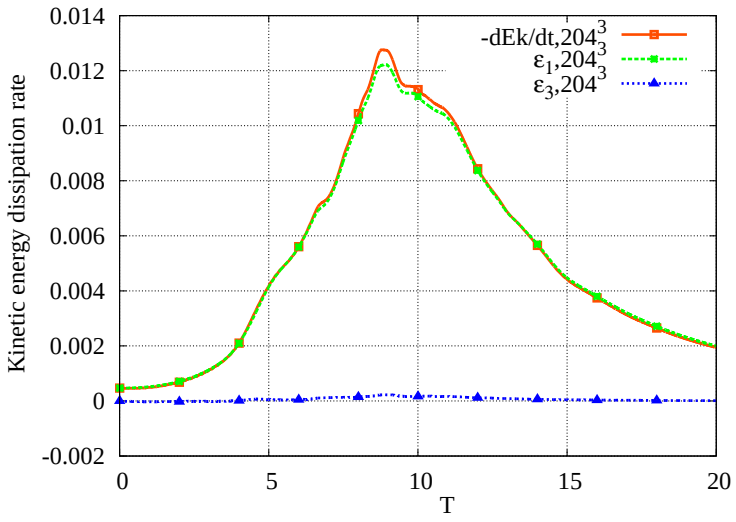
Compared to ESDIRK3, ROSI2PW and ROS34PRW require roughly 25% and 23% CPU time less respectively, although ESDIRK3 could use larger time step.

6 Conclusion and outlook

A comparative study has been performed on a set of third-order time integration schemes for the 3D unsteady compressible Navier-Stokes equations spatially discretized by a third-order hierarchical WENO reconstructed discontinuous Galerkin method. A variety of test cases have been conducted to assess the accuracy, efficiency, robustness, and versatility of those schemes. Numerical experiments demonstrated that both the third-order ESDIRK3 scheme, and the Rosenbrock-Wanner schemes based on the DAEs of Index-2, are able to achieve the designed order of temporal convergence, and the Rosenbrock-Wanner schemes are more efficient than the ESDIRK3 scheme in terms of computational cost. Future work will be focused on the comparative performance assessment between the ROK schemes based on the DAEs of Index-1, and those based on the DAEs of Index-2.



(a) ROSI2PW+rDG(P1P2)



(b) ROS34PRW+rDG(P1P2)

Figure 20: Kinetic energy dissipation balance for the Taylor-Green vortex problem.

Acknowledgments

The authors would like to acknowledge the support for this work provided by the Basic Research Initiative program of the Air Force Office of Scientific Research. Dr. F. Fariba and Dr. D. Smith serve as the technical monitors.

Appendix: Sets of coefficients of the Rosenbrock schemes considered in this paper

Table 8: Set of coefficients for ROSI2PW.

γ = 4.3586652150845900e-01	
α_{21} = 8.7173304301691801e-01	γ_{21} = -8.7173304301691801e-01
α_{31} = -7.9937335839852708e-01	γ_{31} = 3.0647867418622479e+00
α_{32} = -7.9937335839852708e-01	γ_{32} = 3.0647867418622479e+00
α_{41} = 7.0849664917601007e-01	γ_{41} = -1.0424832458800504e-01
α_{42} = 3.1746327955312481e-01	γ_{42} = -3.1746327955312481e-01
α_{43} = -2.5959928729134892e-02	γ_{43} = -1.4154917367329144e-02
b_1 = 6.0424832458800504e-01	
b_2 = -3.6210810811598324e-32	
b_3 = -4.0114846096464034e-02	
b_4 = 4.3586652150845900e-01	

Table 9: Set of coefficients for ROS34PRW.

γ = 4.3586652150845900e-01	
α_{21} = 8.7173304301691801e-01	γ_{21} = -8.7173304301691801e-01
α_{31} = 1.4722022879435914e+00	γ_{31} = -1.2855347382089872e+00
α_{32} = -3.1840250568090289e-01	γ_{32} = 5.0507005541550687e-01
α_{41} = 8.1505192016694938e-01	γ_{41} = -4.8201449182864348e-01
α_{42} = 5.0000000000000000e-01	γ_{42} = 2.1793326075422950e-01
α_{43} = -3.1505192016694938e-01	γ_{43} = -1.7178529043404503e-01
b_1 = 3.3303742833830591e-01	
b_2 = 7.1793326075422947e-01	
b_3 = -4.8683721060099439e 01	
b_4 = 4.3586652150845900e-01	

Table 10: Set of coefficients for ROS34PW2.

γ = 4.3586652150845900e-01	
α_{21} = 8.7173304301691801e-01	γ_{21} = -8.7173304301691801e-01
α_{31} = 8.4457060015369423e-01	γ_{31} = -9.0338057013044082e-01
α_{32} = -1.1299064236484185e-01	γ_{32} = 5.4180672388095326e-02
α_{41} = 0.0000000000000000e+00	γ_{41} = 2.4212380706095346e-01
α_{42} = 0.0000000000000000e+00	γ_{42} = -1.2232505839045147e+00
α_{43} = 1.0000000000000000e+00	γ_{43} = 5.4526025533510214e-01
b_1 = 2.4212380706095346e-01	
b_2 = -1.2232505839045147e+00	
b_3 = 1.5452602553351020e+00	
b_4 = 4.3586652150845900e-01	

References

[1] W. H. Reed and T. Hill, Triangular mesh methods for the neutron transport equation, Tech. Rep. Los Alamos Report LA-UR-73-479, 1973.

[2] B. Cockburn, G. E. Karniadakis, and C.W. Shu, Discontinuous Galerkin Methods: Theory, Computation and Applications. Lecture Notes in Computational Science and Engineering, Springer, 2000.

[3] F. Bassi and S. Rebay, High-order accurate discontinuous finite element solution of the 2D Euler equations, J. Comput. Phys., 138(2):251-285, 1997.

[4] F. Bassi and S. Rebay, A high-order accurate discontinuous finite element method for the numerical solution of the compressible NavierStokes equations, J. Comput. Phys., 131(2):267-279, 1997.

[5] T. Warburton and G. Karniadakis, A discontinuous Galerkin method for the viscous MHD equations, J. Comput. Phys., 152(2):608-641, 1999.

[6] P. Rasetarinera and M. Hussaini, An efficient implicit discontinuous spectral Galerkin method, J. Comput. Phys., 172(2):718-738, 2001.

[7] H. Luo, J. D. Baum, and R. Löhner, A discontinuous Galerkin method based on a Taylor basis for the compressible flows on arbitrary grids, J. Comput. Phys., 227(20):8875-8893, 2008.

[8] J. Sudirham, J. Van Der Vegt, and R. Van Damme, Spacetime discontinuous Galerkin method for advectiondiffusion problems on time-dependent domains, Applied Numerical Mathematics, 56(12):1491-1518, 2006.

[9] M. Dumbser, D. S. Balsara, E. F. Toro, and C. D. Munz, A unified framework for the construction of one-step finite volume and discontinuous Galerkin schemes on unstructured meshes, J. Comput. Phys., 227(18):8209-8253, 2008.

[10] M. Dumbser and O. Zanotti, Very high order P N P M schemes on unstructured meshes for the resistive relativistic MHD equations, J. Comput. Phys., 228(18):6991-7006, 2009.

[11] M. Dumbser, Arbitrary high order PNPM schemes on unstructured meshes for the compressible NavierStokes equations, Computer. Fluid, 39(1):60-76, 2010.

[12] B. van Leer, M. Lo, and M. van Raalte, A discontinuous Galerkin method for diffusion based on recovery, AIAA-2007-4083, 2007.

[13] H. Luo, L. Luo, R. Nourgaliev, V. A. Mousseau, and N. Dinh, A reconstructed discontinuous Galerkin method for the compressible NavierStokes equations on arbitrary grids, J. Comput. Phys., 229(19):6961-6978, 2010.

[14] H. Luo, Y. Xia, and R. Nourgaliev, A class of reconstructed discontinuous Galerkin meth- ods in computational fluid dynamics, in International Conference on Mathematics and Computational Methods Applied to Nuclear Science and Engineering (M&C2011), Brazil, 2011.

[15] L. Zhang, L. Wei, H. Lixin, D. Xiaogang, and Z. Hanxin, A class of hybrid dg/fv methods for conservation laws i: Basic formulation and one-dimensional systems, J. Comput. Phys., 231(4):1081-1103,2012.

[16] L. Zhang, L. Wei, H. Lixin, D. Xiaogang, and Z. Hanxin, A class of hybrid dg/fv methods for conservation laws ii: Two-dimensional cases, J. Comput. Phys., 231(4):1104-1120, 2012.

[17] H. Luo, H. Xiao, R. Nourgaliev, and C. Cai, A comparative study of different reconstruction schemes for a reconstructed discontinuous Galerkin method on arbitrary grids, AIAA-2011-3839, 2011.

[18] Y. Xia, H. Luo, and R. Nourgaliev, An implicit Hermite WENO reconstruction-based discontinuous Galerkin method on tetrahedral grids, Computer. Fluid, 96:406-421, 2014.

[19] Y. Xia, H. Luo, M. Frisbey, and R. Nourgaliev, A set of parallel, implicit methods for a recon-

structed discontinuous Galerkin method for compressible flows on 3D hybrid grids, Computer. Fluid, 98:134-151, 2014.
[20] H. Bijl, M. H. Carpenter, V. N. Vatsa, and C. A. Kennedy, Implicit time integration schemes for the unsteady compressible NavierStokes equations: laminar flow, J. Comput. Phys., 179(1):313-329, 2002.
[21] L. Wang and D. J. Mavriplis, Implicit solution of the unsteady Euler equations for high-order accurate discontinuous Galerkin discretizations, J. Comput. Phys., 225(2):1994-2015, 2007.
[22] Y. Xia, H. Luo, C. Wang, and R. Nourgaliev, An implicit, reconstructed discontinuous Galerkin method for the unsteady compressible Navier-Stokes equations on 3D hybrid grids, AIAA-2014-3220, 2014.
[23] F. Bassi, L. Botti, A. Colombo, A. Ghidoni, and F. Massa, Linearly implicit Rosenbrock-type Runge-Kutta schemes applied to the Discontinuous Galerkin solution of compressible and incompressible unsteady flows, Computer. Fluid, 118:305-320, 2015.
[24] D. S. Blom, H. Bijl, P. Birken, A. Meister, and A. H. Van Zuijlen, Rosenbrock time integration for unsteady flow simulations, in Coupled Problems 2013: Proceedings of the 5th International Conference on Computational Methods for Coupled Problems in Science and Engineering, Ibiza, Spain, 17-19 June 2013, CIMNE, 2013.
[25] P. Birken, G. Gassner, M. Haas, and C. Munz, Efficient Time Integration for Discontinuous Galerkin Methods for the Unsteady 3D Navier-Stokes Equations, in European Congress on Computational Methods and Applied Sciences and Engineering (ECCOMAS 2012), number Eccomas, 2012.
[26] J. Rang and L. Angermann, New Rosenbrock methods of order 3 for PDAEs of index 2. Comenius University Press, 2007.
[27] J. Rang, A new stiffly accurate rosenbrock-wanner method for solving the incompressible Navier-Stokes equations, in Recent Developments in the Numerics of Nonlinear Hyperbolic Conservation Laws, pp. 301-315, Springer, 2013.
[28] Y. Xia, M. Frisbey, H. Luo, and R. Nourgaliev, A WENO reconstruction-based discontinuous Galerkin method for compressible flows on hybrid grids, AIAA-2013-0516, 2013.
[29] H. Luo, Y. Xia, S. Li, and R. Nourgaliev, A Hermite WENO reconstruction-based discontinuous Galerkin method for the Euler equations on tetrahedral grids, J. Comput. Phys., 231(16):5489-5503, 2012.
[30] H. Luo, Y. Xia, S. Spiegel, R. Nourgaliev, and Z. Jiang, A reconstructed discontinuous Galerkin method based on a hierarchical WENO reconstruction for compressible flows on tetrahedral grids, J. Comput. Phys., 236:477-492, 2013.
[31] H. Luo, L. Luo, and R. Nourgaliev, A reconstructed discontinuous Galerkin method for the Euler equations on arbitrary grids, Commun. Comput Phys., 12(5):1495-1519, 2012.
[32] F. Haider, J. P. Croisille, and B. Courbet, Stability analysis of the cell centered finite-volume muscl method on unstructured grids, Numerische Mathematik, 113(4):555-600, 2009.
[33] H. Luo, L. Luo, A. Ali, R. Nourgaliev, and C. Cai, A parallel, reconstructed discontinuous Galerkin method for the compressible flows on arbitrary grids, Commun. Comput Phys., 9(2):363-389, 2011.
[34] Y. Xia, H. Luo, and R. Nourgaliev, An implicit reconstructed discontinuous Galerkin method based on automatic differentiation for the compressible flows on tetrahedral grids, AIAA-2013-0687, 2013.
[35] X. Liu, L. Xuan, H. Luo, and Y. Xia, A reconstructed discontinuous Galerkin method for the compressible Navier-Stokes equations on hybrid grids, AIAA-2015-0575, 2015.
[36] E. J. Hall, D. A. Topp, N. J. Heidegger, G. S. McNulty, K. F. Weber, and R. A. Delaney, Task

7: Endwall Treatment Inlet Flow Distortion Analysis, Tech. Rep. NASA Contractor Report 195-468, 1996.

[37] A. Uranga, P. O. Persson, M. Drela, and J. Peraire, Implicit large eddy simulation of transition to turbulence at low reynolds numbers using a discontinuous Galerkin method, Int. J. Numer. Meth. Fluids, 87(5):232-261, 2011.

[38] A. K. Prasad and J. R. Koseff, Reynolds number and end-wall effects on a lid-driven cavity flow, Physics of Fluids A: Fluid Dynamics (1989-1993), 1(2):208-218, 1989.

[39] Y. Zang, R. L. Street, and J. R. Koseff, A dynamic mixed subgrid-scale model and its ap plication to turbulent recirculating flows, Physics of Fluids A: Fluid Dynamics (1989-1993), 5(2):3186-3196, 1993.

[40] J. Bull and J. A., Simulation of the compressible Taylor Green Vortex using High-Order flux reconstruction schemes, AIAA-2014-3210, 2014.

[41] W. M. Van Rees, A. Leonard, D. Pullin, and P. Koumoutsakos, A comparison of vortex and pseudo-spectral methods for the simulation of periodic vortical flows at high reynolds numbers, J. Comput. Phys., 230(8):2794-2805, 2011.