Tenth International Conference on Computational Fluid Dynamics (ICCFD10), Barcelona,Spain, July 9-13, 2018

ICCFD10-140

A reconstructed Discontinuous Galerkin Method Based on a Variational Reconstruction for Compressible Flows

Lingquan Li, Jialin Lou and Hong Luo
Corresponding author: hong_luo@ncsu.edu

Department of Mechanical and Aerospace Engineering
North Carolina State University, Raleigh, NC 27695, USA.

Abstract: A reconstructed discontinuous Galerkin (rDG(P1P2)) method is developed for compressible flows on 3D arbitrary grids. In this method, a quadratic polynomial solution is reconstructed using a newly developed variational reconstruction based on the given linear discontinuous Galerkin solution. The second derivatives are obtained by minimizing the jump of the values of the reconstructed polynomial solutions and their spatial derivatives at cell interfaces, and therefore maximizes smoothness of the reconstructed polynomial solutions. Unlike the rDG(P1P2) method based on a least-squares reconstruction, the resulting rDG(P1P2) method is stable even on tetrahedral grids, since the stencils in the variational reconstruction are intrinsically the entire mesh. A variety of the benchmark test cases are presented to assess the accuracy, efficiency, robustness and flexibility of this rDG(P1P2) method. The numerical experiments demonstrate that the developed rDG(P1P2) method based on the variational reconstruction is able to maintain the linear stability, attain the designed third order of accuracy, and outperform its rDG(P1P2) method based on the least-squares reconstruction without a significant increase in computing costs and storage requirements.

Keywords: Computational Fluid Dynamics, reconstructed Discontinuous Galerkin, variational reconstruction.

1 Introduction

There are three main methods when it comes to Computational Fluid Dynamic (CFD), namely finite volume, finite element and discontinuous Galerkin (DG) method. The DG methods combine the advantages of both finite volume and finite element methods. Associated to finite volume and finite element methods, DG methods implement the high-order polynomials on each element and solving the Riemann problems that arise from the discontinuous representation of solution on each element interface. The cell-centered finite volume methods exactly correspond to the DG(P0) methods. The DG methods can be easily extended to high-order approximation (DG(Pn)) and are suited for complex geometries. They are compact so the coding can be structured and simplified. Since the elements are independent, and the inter-element communications are minimal (elements only communicate with von Neumann neighbors regardless of the order of accuracy), parallelization can be easily implemented. What's more, they have several useful mathematical properties with respect to conservation, stability, and convergence. Due to these attractive features, DG methods have been widely used for solving the systems of conservation laws.

However, the DG methods suffer from a number of their own weaknesses. For 3D applications, four variables need to be in storage for DG(P1) solution, while ten variables are needed for DG(P2) solution, and twenty variables are needed for DG(P3) solution for each variable on each element. The costs of computation and the storage requirement can be huge once it comes to system of equations and large number of grids. In order to reduce both computational costs and storage requirements, a new family of reconstructed Discon-

tinuous Galerkin (rDG) methods, termed PnPm methods, is introduced [1][2][3]. This method introduces the idea of reconstruction in FV methods to DG methods. Pn indicates that a piecewise polynomial of order of n is used to represent a DG solution, and Pm represents a reconstructed polynomial solution of degree of m (m>n) that is used to compute the fluxes. The PnPm schemes are for enhancing the order of accuracy by increasing the order of underlying polynomial solution. The PnPm schemes provide a unified formulation for both finite volume and DG methods. For example, when n=0, a piecewise constant polynomial is used to represent the numerical solution. The P0Pm schemes are classical high order finite volume methods where a polynomial of order of m is reconstructed from the piecewise constant solution. The PnPn schemes recover the standard DG(Pn) solution. This reconstructed DG method has been widely used for solving compressible flow problems[4][5][6][7][8][9][10] as well as turbulence problems[11][12].

The PnPm schemes require a reconstruction method which can extend the piecewise polynomial of order of n to order of m. Most of work [13][14][15][16] in this area is a continuation of the work of Barth amd Frederickson [17] based on a least-squares (LS) reconstruction. The resulting method is denoted as rDG(PnPm(LS)) method in this paper. However, the implementation of the least-squares method can suffer from two main disadvantages. First of all, the least-squares method may suffer from the linear instability on unstructured tetrahedral grids when the reconstruction stencils only involve von Neumann neighborhood, i.e., the adjacent face-neighboring cells [18]. Secondly, when it comes to high-order reconstruction, the stencils of the least-squares method must be extended. This extension of stencils may result in a more complicated data structure for unstructured grids. What's more, the elements which are adjacent to the boundaries must be distinguished for choosing the stencils when implementing high-order least squares reconstruction.

The objective of the presented work is to develop an accurate, efficient and robust rDG(P1P2) method for solving compressible flow problems on arbitrary grids. The novelty of this rDG method is to use a newly developed variational formulation [19] to reconstruct a high-order polynomial solution. This variational reconstruction (VR) can be regarded as an extension of the compact finite difference schemes [20] to the unstructured grids. The high-order terms are obtained by solving an extreme value problem, which minimizes the jump of the values of the reconstructured polynomial solutions and their spatial derivatives at cell interfaces, and therefore maximizes smoothness of the reconstructed polynomial solutions. It has been shown that the FV method based on VR can achieve 1-exactness [21]. This new reconstruction formulation has also been utilized with a hyperbolic method to solve diffusion model equation [22] and advection-diffusion equation [22][23]. In this paper, the presented VR method can obtain the property of 2-exactness just like the least-squares counterpart as they are applied to rDG(P1P2) formulation. The resulting method is denoted as rDG(P1P2(VR)) method in this paper. It is stable even on tetrahedral grids, since its stecils are intrinsically the entire mesh. However, the data structure required by VR is the same as the least-squares method and is thus compact and simple. A variety of the benchmark test cases are presented to assess the accuracy, efficiency, robustness and flexibility of this rDG method. The numerical experiments demonstrate that the VR method is more accurate than its LS counterpart and the developed rDG(P1P2(VR)) method is able to maintain the linear stability, attain the designed high-order of accuracy, and outperform the rDG(P1P2(LS)) method without a significant increase in computing costs and storage requirements [21][24]. The remainder of this paper is organized as follows. The governing equations are presented in Section 2. The developed variational reconstruction based DG method is described in Section 3. Extensive numerical experiments are reported in Section 4. Concluding remarks are given in Section 5.

2 Governing Equations

The Euler equations governing compressible inviscid flows can be expressed in conservative form as

$$\frac{\partial \mathbf{U}(\mathbf{x},t)}{\partial t} + \frac{\partial \mathbf{F}_j(\mathbf{U}(\mathbf{x},t))}{\partial x_j} = 0 \tag{1}$$

where the summation convention is used here. The conservative variable vector $\mathbf{U}$, and inviscid flux vector $\mathbf{F}$ are defined by

$$\mathbf{U} = \begin{pmatrix} \rho \\ \rho u_i \\ \rho e \end{pmatrix} \quad \mathbf{F}_j = \begin{pmatrix} \rho u_j \\ \rho u_i u_j + p\delta_{ij} \\ u_j(\rho e + p) \end{pmatrix} \tag{2}$$

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 x_i. The pressure is computed from the equation of state

$$p = (\gamma - 1)\rho(e - \frac{1}{2}u_j u_j) \tag{3}$$

which is valid for perfect gas, where γ is the ratio of the specific heats.

3 Numerical Method

3.1 Discontinuous Galerkin Spatial Discretization

To formulate the discontinuous Galerkin method, the weak formulation of (2.1) is introduced here by multiplying (2.1) by a test function $\mathbf{W}$, then integrating over the domaind Ω with boudary $\Gamma = \partial\Omega$, and performing an integration by parts:

$$\int_\Omega \frac{\partial \mathbf{U}}{\partial t}\mathbf{W}d\Omega + \int_\Gamma \mathbf{F}_j \mathbf{n}_j \mathbf{W}d\Gamma - \int_\Omega \mathbf{F}_j \frac{\partial \mathbf{W}}{\partial x_j}d\Omega = 0, \quad \forall \mathbf{W} \tag{4}$$

where $\mathbf{n}_j$ denotes the unit outward normal vector to the boundary. Assuming that Ω_h is a classical triangulation of Ω where the domain Ω is subdivided into a collection of non-overlapping elements Ω_e, triangles in two-dimension and tetrahedra in three-dimension, the semi-discrete form of (3.1) can be obtained by applying (3.1) on element Ω_e

$$\frac{d}{dt}\int_{\Omega_e} \mathbf{U}_h \mathbf{W}_h d\Omega + \int_{\Gamma_e} \mathbf{F}_j(\hat{\mathbf{U}}_h)\mathbf{n}_j \mathbf{W}_h d\Gamma - \int_{\Omega_e} \mathbf{F}_j(\hat{\mathbf{U}}_h)\frac{\partial \mathbf{W}_h}{\partial x_j}d\Omega = 0, \quad \forall \mathbf{W}_h \tag{5}$$

where Γ_e denotes the boundary of Ω_e. $\mathbf{U}_h$ and $\mathbf{W}_h$ represent the finite element approximations to the analytical solution $\mathbf{U}$ and the test function $\mathbf{W}$, respectively. $\hat{\mathbf{U}}_h$ denotes the approximation of the solution $\mathbf{U}$ at interface, including the reconstructed high-order terms. The approximate solution and test function can be expressed as piecewise polynomials in each element

$$\mathbf{U}_h(\mathbf{x}, t) = \sum_{k=1}^{N} \mathbf{U}_k(t) B_k^p(\mathbf{x}), \quad \hat{\mathbf{U}}_h(\mathbf{x}, t) = \sum_{k=1}^{M} \mathbf{U}_k(t) B_k^p(\mathbf{x}), \quad \mathbf{W}_h(\mathbf{x}) = \sum_{k=1}^{N} \mathbf{W}_k B_k^p(\mathbf{x}) \tag{6}$$

where $B_k^p(\mathbf{x})$, $1 \le k \le n$ is the basis function of the polynomials of degree p. The dimension of the polymonial space, $N = N(p, d)$ depends on the degree of the polynomials of the expansion p and the spatial dimensions d

$$N = \frac{(p+1)(p+2)...(p+d)}{d!}, \quad d = 1, 2, 3 \tag{7}$$

For the presented rDG(P1P2) method in three-dimension, $N = 4$ and $M = 10$. The basis functions are taken as

$$\begin{gathered}
B_1^p = 1, \quad B_2^p = \frac{x - x_c}{\Delta x}, \quad B_3^p = \frac{y - y_c}{\Delta y}, \quad B_4^p = \frac{z - z_c}{\Delta z}, \\
B_5^p = \frac{1}{2}\left((B_2^p)^2 - \frac{1}{\Omega_e}\int (B_2^p)^2 d\Omega\right), \quad B_6^p = \frac{1}{2}\left((B_3^p)^2 - \frac{1}{\Omega_e}\int (B_3^p)^2 d\Omega\right), \\
B_7^p = \frac{1}{2}\left((B_4^p)^2 - \frac{1}{\Omega_e}\int (B_4^p)^2 d\Omega\right), \quad B_8^p = B_2^p B_3^p - \frac{1}{\Omega_e}\int B_2^p B_3^p d\Omega, \\
B_9^p = B_2^p B_4^p - \frac{1}{\Omega_e}\int B_2^p B_4^p d\Omega, \quad B_{10}^p = B_3^p B_4^p - \frac{1}{\Omega_e}\int B_3^p B_4^p d\Omega,
\end{gathered} \tag{8}$$

where the subscript c denotes central values and Δx, Δy, Δz are lengh scales for each element in 3 directions. (3.2) must be satisfied for any test function $\mathbf{W}_h$. Since B_l^p is the basis for $\mathbf{W}_h$, (3.2) is equivalent to the following system of equations

$$\frac{d\mathbf{U}_k}{dt}\int_{\Omega_e} B_k^p B_l^p d\Omega + \int_{\Gamma_e} \mathbf{F}_j(\hat{\mathbf{U}}_h)\mathbf{n}_j B_l^p d\Gamma - \int_{\Omega_e} \mathbf{F}_j(\mathbf{U}_h)\frac{\partial B_l^p}{\partial x_j} d\Omega = 0, \quad 1 \le l \le n \tag{9}$$

Since the numerical solution $\mathbf{U}_h$ is discontinuous between element interfaces, the interface fluxes are not uniquely defined. The flux function $\mathbf{F}_j(\mathbf{U}_h)\mathbf{n}_j$ is replaced by a numerical Riemann flux function $\mathbf{H}(\hat{\mathbf{U}}_h^L, \hat{\mathbf{U}}_h^R, \mathbf{n})$, where $\hat{\mathbf{U}}_h^L$ and $\hat{\mathbf{U}}_h^R$ are the conservative state vector at the left and right side of the element boundary. In the present work, the Riemann flux function is approximated using the HLLC approximate Riemann solver [25][26][27].

Equation (3.6) represents a system of ordinary differential equation. A fully implicit time scheme is employed to integrate Equation (3.6) to reach steady-state solutions. An approximate Newton method is used to linearize the equations arising from the implicit discretization. GMRES+LU-SGS method [28] has been used to solve the system of linear equations resulted from a implicit BDF1 time marching method.

3.2 Reconstructin Schemes

A recently developed variational reconstruction can be used to obtain the solution gradients by solving an extreme value problem, which minimizes jumps of the values of the reconstructed polynomial solutions and their spatial derivatives at cell interfaces, and therefore maximizes smoothness of the reconstructed polynomial solutions. In this case, a cost function in the variational reconstruction can be defined as

$$\mathbf{I} = \sum_{iface=1}^{nface} \mathbf{I}_{iface} \tag{10}$$

where nface is the number of cell interfaces in a grid and $\mathbf{I}_{iface}$ denotes the interface jump integration function for a given interface. For an internal face Γ_{ij} separating the left element i and the right element j, $\mathbf{I}_{iface}$ is given by

$$\mathbf{I}_{iface} = \frac{1}{d_{ij}}\int_{\Gamma_{ij}} \sum_{p=0}^{m}\sum_{k=0}^{p}\sum_{l=0}^{k}\left[w_{p,k,l}\frac{\partial^p \mathbf{U}^-}{\partial x^l \partial y^k \partial z^{p-k-l}}(d_{ij})^p - w_{p,k,l}\frac{\partial^p \mathbf{U}^+}{\partial x^l \partial y^k \partial z^{p-k-l}}(d_{ij})^p\right]^2 d\Gamma \tag{11}$$

where d_{ij} is the distance between the centroids of the two cells, $w_{p,k,l}$ is the weight. In the presented work, where non-dimensional Taylor basis functions are used, the weights corresponding to $p = 0$, $p = 1$ and $p = 2$ are taken as 1.0, 0.5 and 0.25, respectively. $\mathbf{U}^-$ and $\mathbf{U}^+$ denote the values of reconstructed $\mathbf{U}$ on Γ_{ij} for the two elements i and j, respectively. The face integral can be computed exactly using Gaussian quadrature formulas with sufficient precision. The constitutive relations of the variational reconstruction are derived by minimizzing the total $\mathbf{I}$ with respect to the coefficients of the reconstruction polynomial. This leads to a

system of linear equations, which is then solved using the LU-SGS method. The resultant rDG(P1P2(VR)) method can achieve the linear stability, since the stencils of this variational reconstruction are intrinsically the entire mesh.

4 Numerical Examples

A few examples are presented in this section to demonstrate the high accuracy and robustness of our rDG(P1P2) method based on VR for compressible flow problems on arbitrary grids. For two-dimensional problems, the number of cells in the z-direction is simply set to be 1. The first test case is to assess the 2-exactness and the accuracy of the presented reconstruction method. In this test case, the error is measured as the L2 error of the reconstructed solution. Test case 2 and test case 3 are chosen to demonstrate that the developed rDG(P1P2) VR method is able to achieve the designed order of convergence for smooth flows. The length scale, characterizing the cell size of an unstructured grid, is defined as $1/\sqrt[2]{ncells}$ and $1/\sqrt[3]{ncells}$, for 2D and 3D problems, respectively, where $ncells$ is the number of cells. The L2-norm of the entropy production is used as the error measurement

$$\|\epsilon\|_{L_2(\Omega)} = \sqrt{\int_\Omega \epsilon^2 d\Omega} \tag{12}$$

where the entropy production ϵ is defined as

$$\epsilon = \frac{S - S_\infty}{s_\infty} = \frac{p}{p_\infty}(\frac{\rho_\infty}{\rho})^\gamma - 1. \tag{13}$$

Note that the entropy production, where the entropy is defined as $S = p/\rho^\gamma$, is a very good criterion to measure accuracy of the numerical solutions, when the flow under consideration is smooth and therefore isentropic.

Test case 1 Convergence study on reconstruction methods

This test case is chosen to assess the 2-exactness and the accuracy of the variational reconstruction method in comparison with its least-squares counterpart on both hexahedral and prismatic grids. The computational domain is a cube ($0 \leq x_i \leq 1$). The hexahedral mesh and prismatic mesh used are shown in figure 1. A fully quadratic polynomial function and a smooth function are used to assess the accuracy, the order of convergence, and the 2-exact property of the two reconstruction methods. Table 1 presents numerical results obtained by the least-squares reconstruction, VR reconstruction with 1 Gauss point for quardraliterals and VR reconstruction with 16 Gauss points for quadraliterals on hexahedral grids. Table 2 presents numerical results obtained by the VR reconstruction with 1 Gauss points for both triangles and quadraliterals and VR reconstruction with 4 Gauss points for triangles and 16 Gauss points for quadraliterals on prismatic grids. The order of convergence with respect to the smooth function is also illustrated in figure 2. As expected, both least-squares reconstruction and variational reconstruction methods have the 2-exact property. For a generally smooth function, both the least-squares reconstruction and the variational reconstruction can achieve the designed third order of convergence.

Table 1. Order of convergence of test case 1 on hexahedral grids

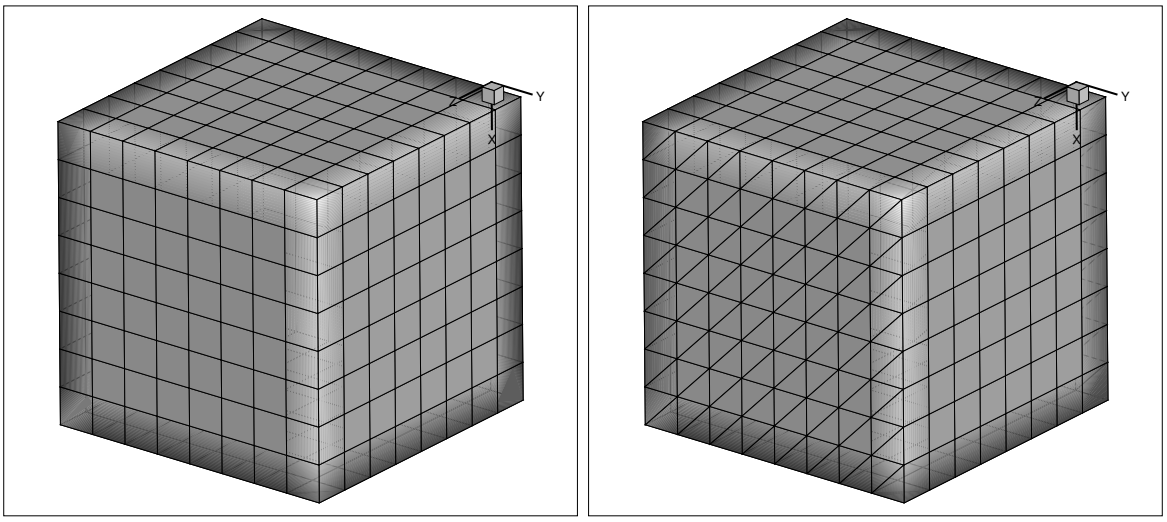


Figure 1: Hexahedral and prismatic mesh used for test case 1

	LS		VR(1 Gauss point)		VR(16 Gauss points)	
Element number	L2-error	Order	L2-error	Order	l2-error	Order
$f(x,y,z) = \frac{1}{2}(999x^2 - 888y^2 + 777z^2) - 666xy + 555xz - 444yz$						
$8 \times 8 \times 8$	1.0153e-12	-	1.0113e-12	-	1.0141e-12	-
$16 \times 16 \times 16$	1.4110e-12	-	1.4050e-12	-	1.4010e-12	-
$32 \times 32 \times 32$	1.2570e-12	-	1.2563e-12	-	1.2577e-12	-
$f(x,y,z) = sin(\pi x)sin(\pi y)sin(\pi z)$						
$8 \times 8 \times 8$	9.4959e-4	-	1.0980e-3	-	1.1708e-3	-
$16 \times 16 \times 16$	1.0687e-4	3.152	1.1620e-4	3.241	1.2178e-4	3.265
$32 \times 32 \times 32$	1.2832e-5	3.058	1.3360e-5	3.121	1.3711e-5	3.151

Table 2. Order of convergence of test case 1 on prismatic grids

	VR(1 and 1 Gauss point)		VR(4 and 16 Gauss points)	
Element number	L2-error	Order	l2-error	Order
$f(x,y,z) = \frac{1}{2}(999x^2 - 888y^2 + 777z^2) - 666xy + 555xz - 444yz$				
$2 \times 8 \times 8 \times 8$	7.2480e-13	-	7.2165e-13	-
$2 \times 16 \times 16 \times 16$	1.2532e-12	-	1.2509e-12	-
$2 \times 32 \times 32 \times 32$	1.7197e-12	-	1.7187e-12	-
$f(x,y,z) = sin(\pi x)sin(\pi y)sin(\pi z)$				
$2 \times 8 \times 8 \times 8$	5.8108e-4	-	5.6690e-4	-
$2 \times 16 \times 16 \times 16$	6.0981e-5	3.253	5.8323e-5	3.281
$2 \times 32 \times 32 \times 32$	7.0934e-5	3.104	6.7368e-6	3.114

Test case 2 Subsonic flow past a circular cylinder

An inviscid subsonic flow past a circular cylinder at a Mach number of $M_\infty = 0.38$ is considered in this test case to assess the order of accuracy and discretization error of the rDG(P1P2(VR)) and rDG(P1P2(LS)) methods for external flows. Computations are performed on two types of grids: one is consisted of hexahedral cells and the other is consisted of prismatic cells. Figure 3 and figure 5 show four successively refined o-type hexahedral and prismatic grids with 16×5, 32×9, 64×17 and 128×33 points, respectively. The first number refers to the number of points in the circular direction, and the second number refers to the number of concentric circles in the mesh. The radius of the cylinder is $r_1 = 0.5$. The domain is bounded by $r_{33} = 20$. Figure 4 and figure 6 show the computed Mach number contours with rDG(P1P2(VR)) method. Table 3 and table 4 show the order of convergence, which is also shown in figure 7. It can be observed that the presented

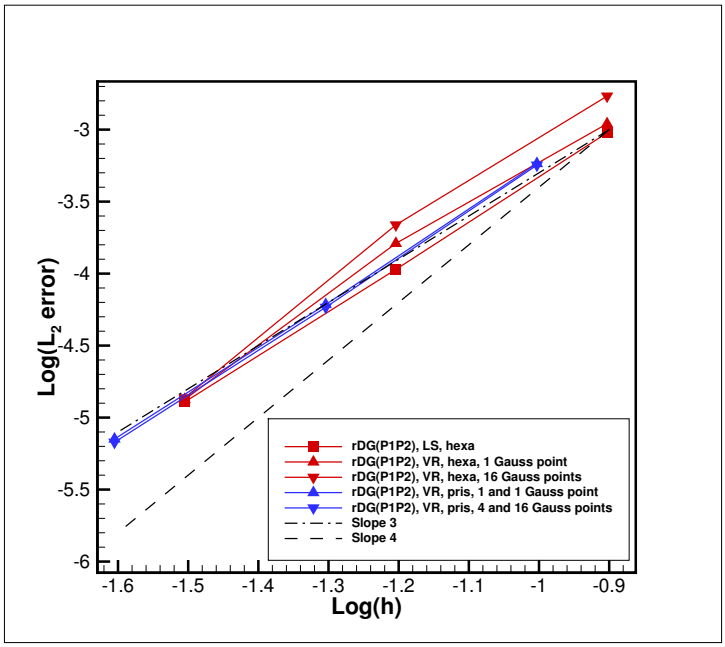


Figure 2: Order of convergence for test case 1

rDG(P1P2) method is able to attain an over-convergence of about fourth order, and the rDG(P1P2(VR)) method is able to outperform its counterpart LS method, especially on prismatic grids.

Table 3. Order of convergence of test case 2 on hexahedral grids

	rDG(P1P2(VR))		rDG(P1P2(LS))	
Log(Length Scale)	Log(L2-error)	Order	Log(L2-error)	Order
-0.903	-0.8594	-	-0.8865	-
-1.204	-1.7138	2.839	-1.8036	3.047
-1.505	-2.9497	4.106	-3.0157	4.027
-1.806	-4.1987	4.150	-4.0653	3.487

Table 4. Order of convergence of test case 2 on prismatic grids

	rDG(P1P2(VR))		rDG(P1P2(LS))	
Log(Length Scale)	Log(L2-error)	Order	Log(L2-error)	Order
-1.054	-1.4768	-	-1.3742	-
-1.355	-2.6658	3.950	-2.5413	3.877
-1.657	-3.9742	4.347	-3.7026	3.858
-1.957	-5.2016	4.078	-4.4100	2.350

Test case 3 Subsonic flow past a sphere

In this test case, a subsonic flow past a sphere at a Mach number of $M_\infty = 0.5$ is considered to assess if the rDG(P1P2(VR)) method can achieve a formal order of convergence rate on tetrahedral grids. The numbers of three successively refined tetrahedral elements, grid points and boundary faces for the four grids are (535, 167, 124), (2426, 598, 322) and (16467, 3425, 1188), respectively. Only a quarter of the configuration is modeled due to the symmetry of the problem. since the rDG(P1P2(LS)) method is unstable for this test case, the results obtained by a second-order DG(P1) method on the first three grids are presented for the purpose of comparison. Figure 8 illustrates the computed velocity contours with rDG(P1P2(VR)) method. Table 5 shows the order of convergence of both rDG(P1P2(VR)) method and DG(P1) method. This comparison is also shown in Figure 9. It can be observed that although the error magnitude is smaller with DG(P1)

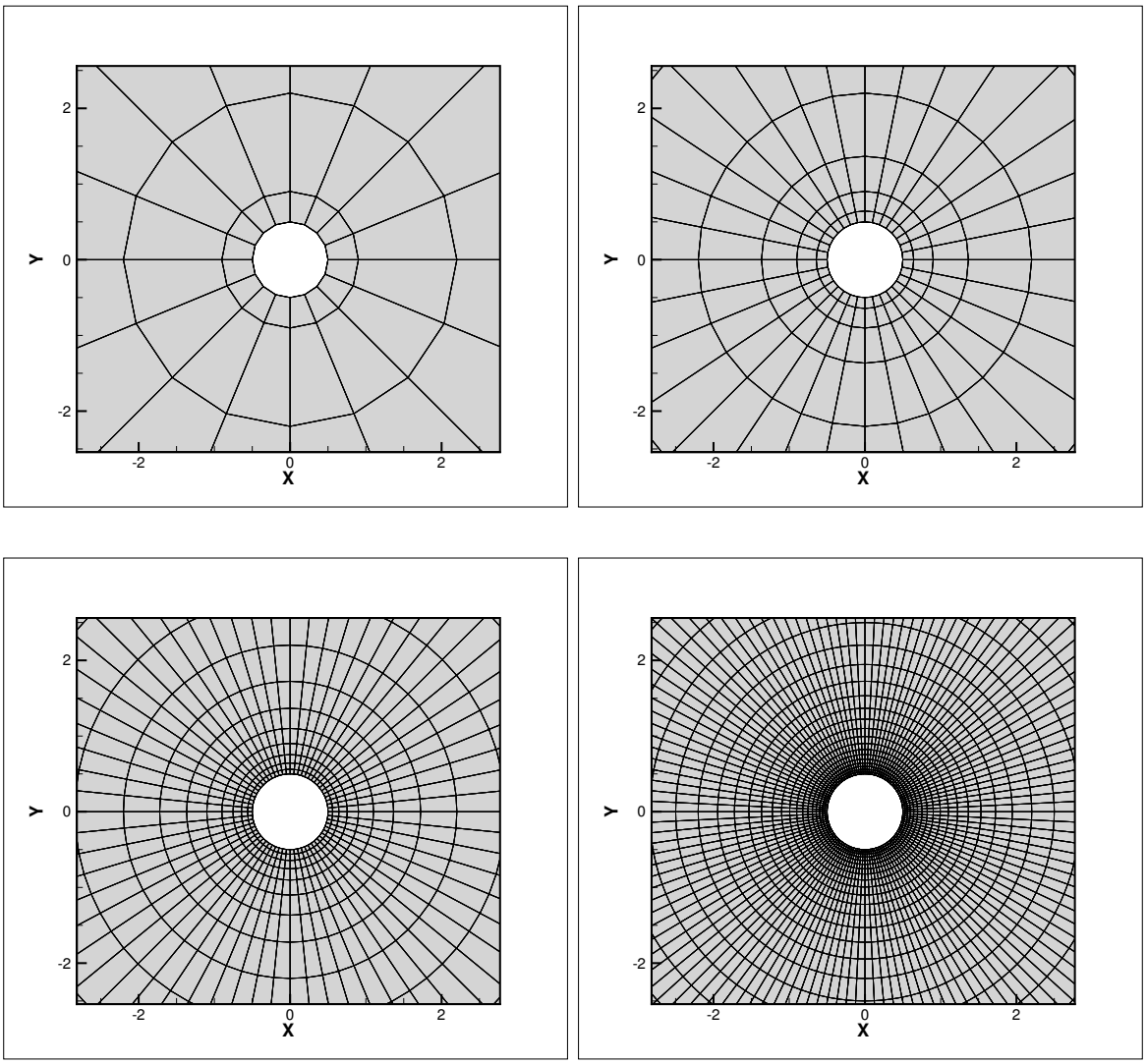
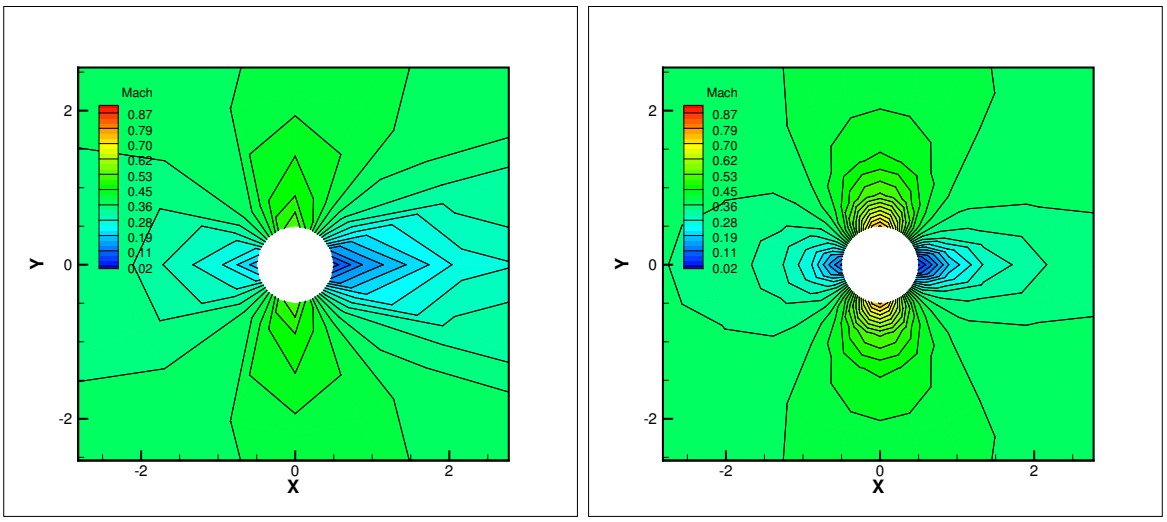


Figure 3: A series of four successively globally refined hexahedral meshes for test case 2



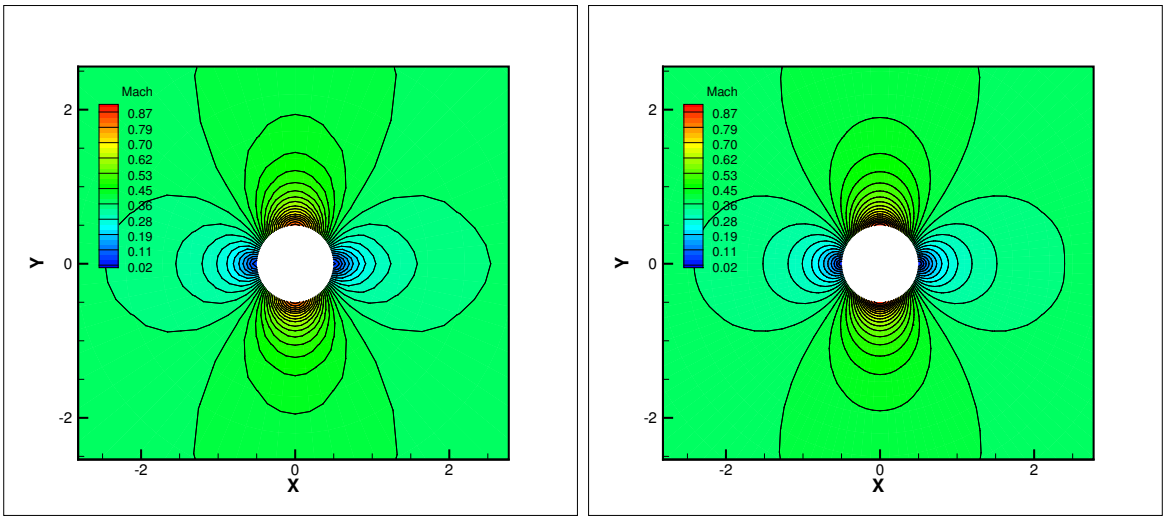


Figure 4: Computed Mach number contours on hexahedral meshes for test case 2

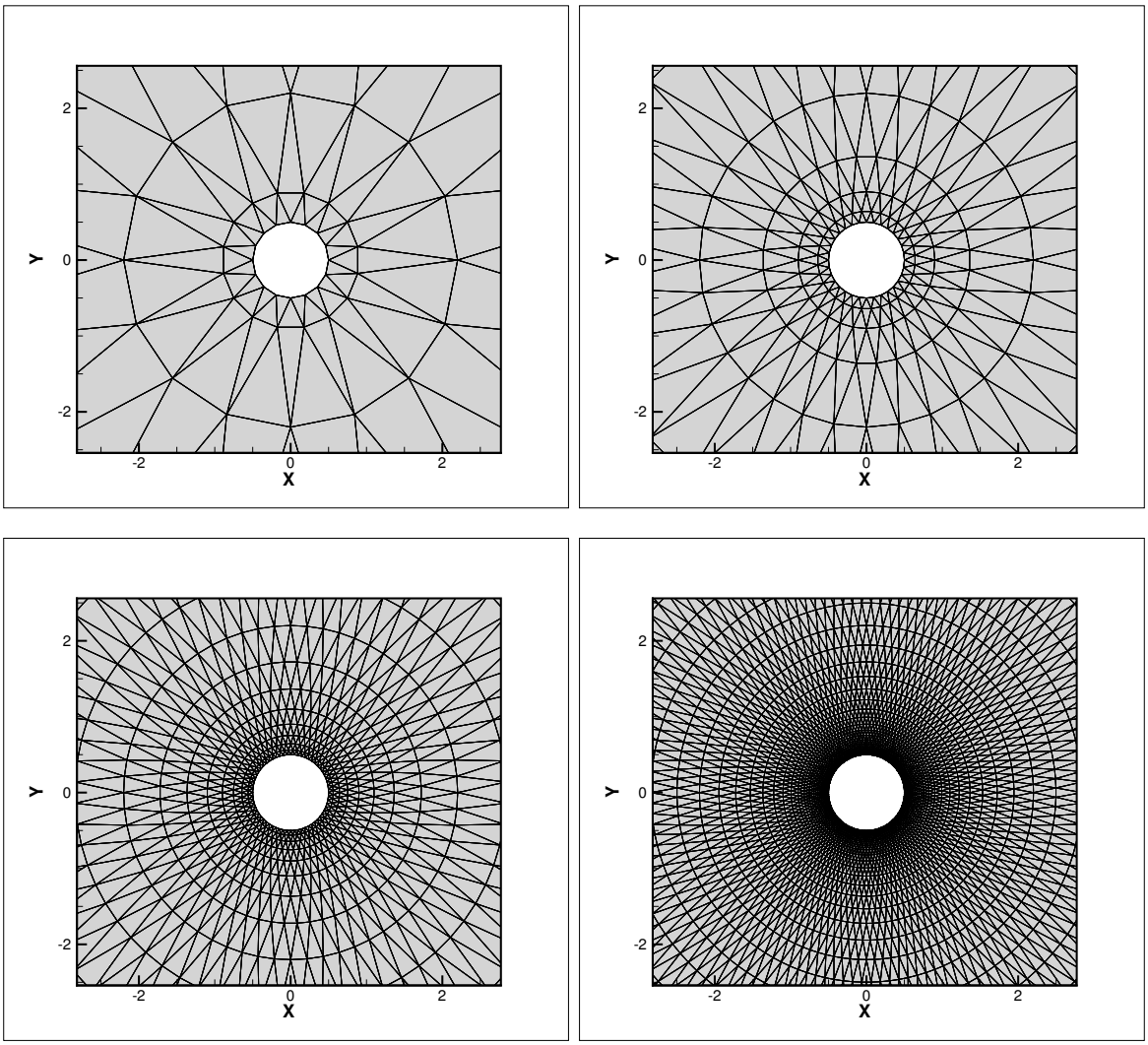


Figure 5: A series of four successively globally refined prismatic meshes for test case 2

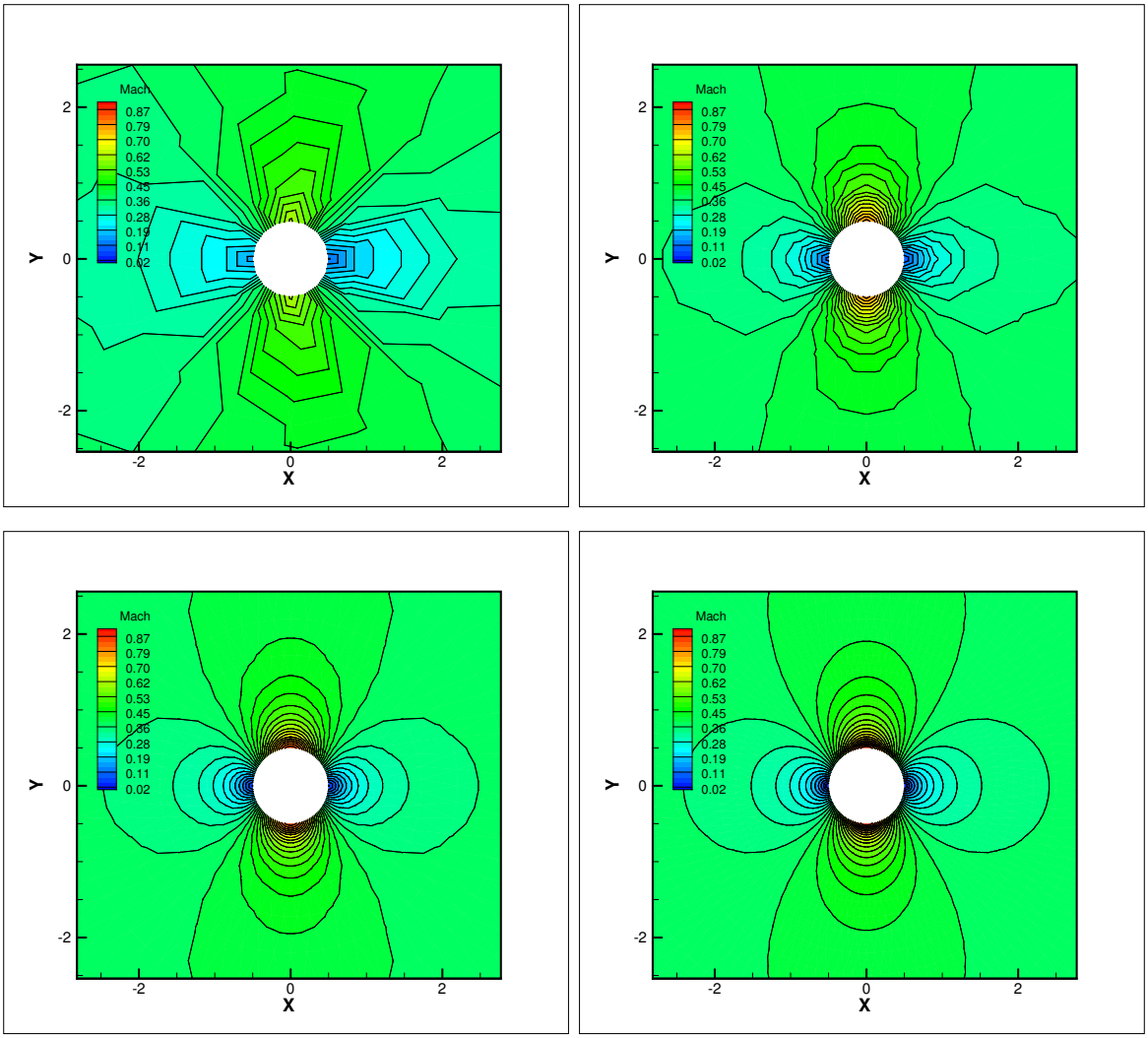


Figure 6: Computed Mach number contours on prismatic meshes for test case 2

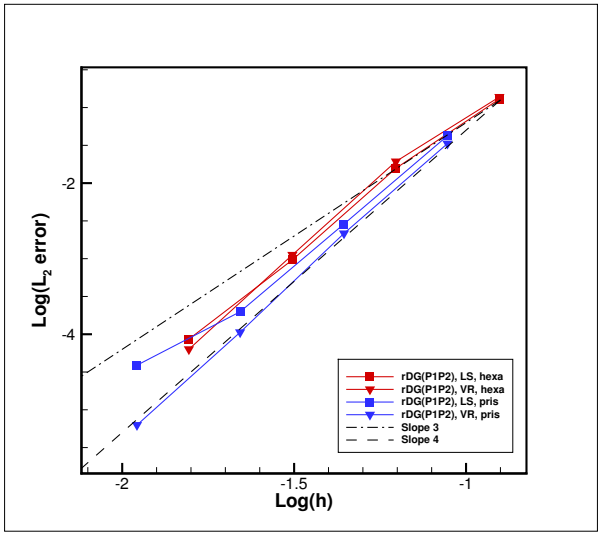


Figure 7: Order of convergence for test case 2

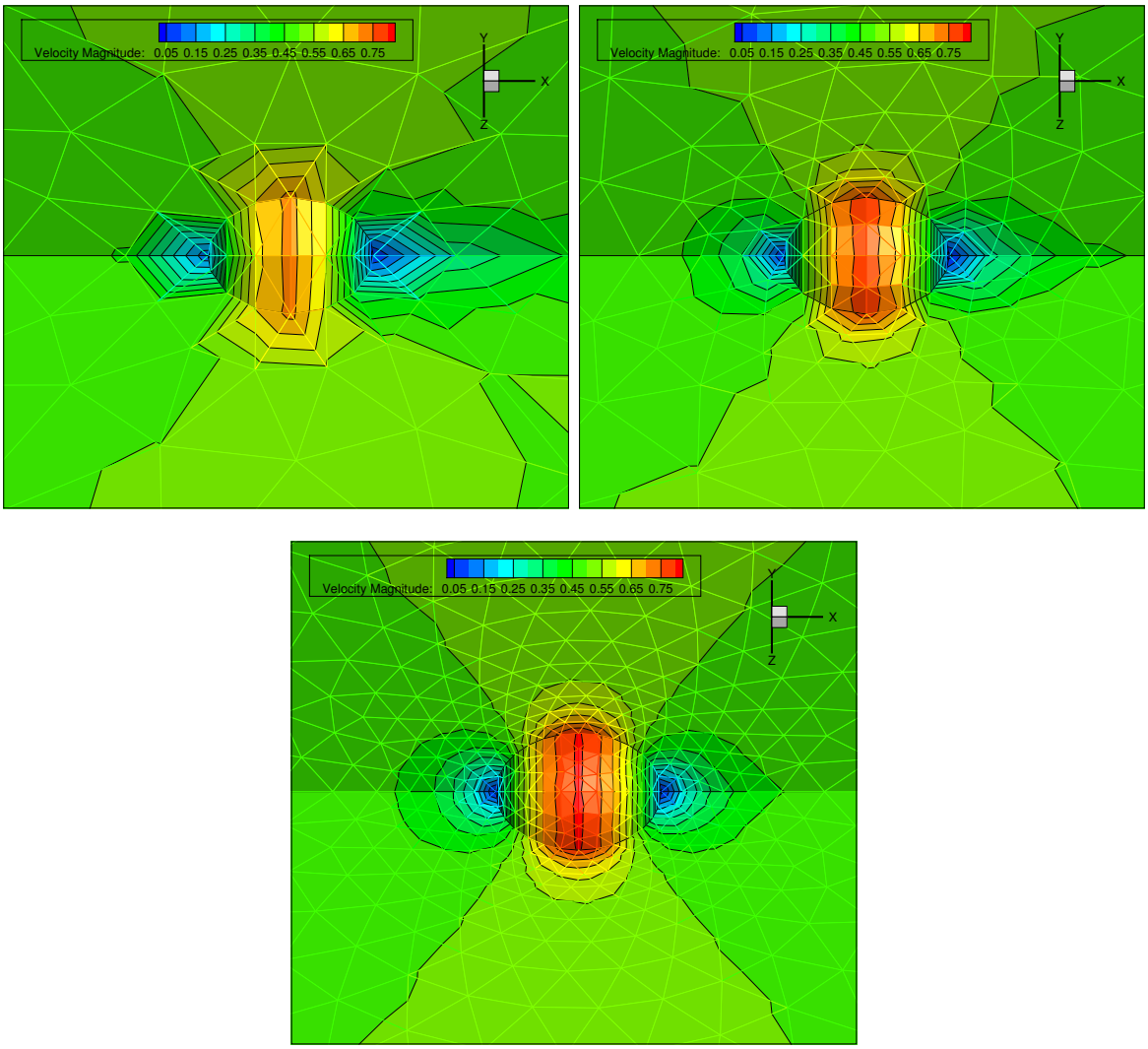


Figure 8: Mach contours on a serie of four successively globally refined tetrahedral meshes for subsonic flow past a sphere at $M_\infty = 0.5$

method, the rDG(P1P2(VR)) method is able to attain the designed third order of accuracy compared to the second order of DG(P1) method. If the mesh is further refined, it can be expected that the rDG(P1P2(VR)) method can deliver much better results.

Table 5. Order of convergence for test case 3

	rDG(P1P2(VR))	
Log(Length Scale)	Log(L2-error)	
-1.1283	-1.9646	-
-1.4055	-2.7361	2.783
-1.6986	-3.6257	3.035

5 Conclusion and Future Work

A rDG(P1P2) based on a variational reconstruction has been developed for solving the compressible Euler equations on 3D arbitrary grids. The variational reconstruction can attain the property of 2-exactness. The resulting rDG(P1P2(VR)) method is linealy stable, since its stencils are intrinsically the entire mesh.

However, the data structure required by the present method is compact and simple. A variety of the benchmark test cases are presented to assess the performance of this method. The numerical experiments indicate that the developed method is able to maintain the linear stability, attain the designed third-order of accuracy, and outperform the rDG(P1P2(LS)) method without a significant increase in computing costs and storage requirements, especially on tetrahedral grids where the LS method becomes unstable. Future development will be focused on the extension of the present method for solving Navier-Stokes equations.

References

[1] Michael Dumbser, Dinshaw S Balsara, Eleuterio F Toro, and Claus-Dieter Munz. A unified framework for the construction of one-step finite volume and discontinuous galerkin schemes on unstructured meshes. *Journal of Computational Physics*, 227(18):8209–8253, 2008.

[2] Michael Dumbser and Olindo Zanotti. Very high order pnpm schemes on unstructured meshes for the resistive relativistic mhd equations. *Journal of Computational Physics*, 228(18):6991–7006, 2009.

[3] Michael Dumbser. Arbitrary high order pnpm schemes on unstructured meshes for the compressible navier–stokes equations. *Computers & Fluids*, 39(1):60–76, 2010.

[4] Xiaodong Liu, Lijun Xuan, Yidong Xia, and Hong Luo. A reconstructed discontinuous galerkin method for the compressible navier-stokes equations on three-dimensional hybrid grids. *Computers & Fluids*, 152:217–230, 2017.

[5] Xiaodong Liu, Yidong Xia, Hong Luo, and Lijun Xuan. A comparative study of rosenbrock-type and implicit runge-kutta time integration for discontinuous galerkin method for unsteady 3d compressible navier-stokes equations. *Communications in Computational Physics*, 20(4):1016–1044, 2016.

[6] Yidong Xia, Xiaodong Liu, Hong Luo, and Robert Nourgaliev. A third-order implicit discontinuous galerkin method based on a hermite weno reconstruction for time-accurate solution of the compressible navier–stokes equations. *International Journal for Numerical Methods in Fluids*, 79(8):416–435, 2015.

[7] Jian Cheng, Tiegang Liu, and Hong Luo. A hybrid reconstructed discontinuous galerkin method for compressible flows on arbitrary grids. *Computers & Fluids*, 139:68–79, 2016.

[8] Yidong Xia, Jialin Lou, Hong Luo, Jack Edwards, and Frank Mueller. Openacc acceleration of an unstructured cfd solver based on a reconstructed discontinuous galerkin method for compressible flows. *International Journal for Numerical Methods in Fluids*, 78(3):123–139, 2015.

[9] Yidong Xia, Hong Luo, Megan Frisbey, and Robert Nourgaliev. A set of parallel, implicit methods for a reconstructed discontinuous galerkin method for compressible flows on 3d hybrid grids. *Computers & Fluids*, 98:134–151, 2014.

[10] Chuanjin Wang, Jian Cheng, Markus Berndt, Neil N Carlson, and Hong Luo. Application of nonlinear krylov acceleration to a reconstructed discontinuous galerkin method for compressible flows. *Computers & Fluids*, 2017.

[11] Xiaodong Liu and Hong Luo. Development and assessment of a reconstructed discontinuous galerkin method for the compressible turbulent flows on hybrid grids. In *54th AIAA Aerospace Sciences Meeting*, page 1359, 2016.

[12] Xiaodong Liu, Yidong Xia, and Hong Luo. A reconstructed discontinuous galerkin method for compressible turbulent flows on 3d curved grids. *Computers & Fluids*, 160:26–41, 2018.

[13] Carl Ollivier-Gooch, Amir Nejat, and Krzysztof Michalak. On obtaining high-order finite-volume solutions to the euler equations on unstructured meshes. In *18th AIAA Computational Fluid Dynamics Conference*, page 4464, 2007.

[14] Chris Michalak and Carl Ollivier-Gooch. Unstructured high-order accurate finite-volume solutions of the navier-stokes equations. In *47th AIAA Aerospace Sciences Meeting including The New Horizons Forum and Aerospace Exposition*, page 954, 2009.

[15] Boris Diskin and James L Thomas. Comparison of node-centered and cell-centered unstructured finite-volume discretizations: inviscid fluxes. *AIAA journal*, 49(4):836–854, 2011.

[16] Boris Diskin, James L Thomas, Eric J Nielsen, Hiroaki Nishikawa, and Jeffery A White. Comparison of node-centered and cell-centered unstructured finite-volume discretizations: viscous fluxes. *AIAA journal*, 48(7):1326–1338, 2010.

[17] Timothy Barth and PAUL FREDERICKSON. Higher order solution of the euler equations on unstructured grids using quadratic reconstruction. In *28th aerospace sciences meeting*, page 13, 1990.

[18] Florian Haider, J-P Croisille, and Bernard Courbet. Stability analysis of the cell centered finite-volume m uscl method on unstructured grids. *Numerische Mathematik*, 113(4):555–600, 2009.

[19] Qian Wang, Yu-Xin Ren, Jianhua Pan, and Wanai Li. Compact high order finite volume method on unstructured grids iii: Variational reconstruction. *Journal of Computational physics*, 337:1–26, 2017.

[20] Sanjiva K Lele. Compact finite difference schemes with spectral-like resolution. *Journal of computational physics*, 103(1):16–42, 1992.

[21] Xiaodong Liu, Jialin Lou, Lingquan Li, Hong Luo, Hiroaki Nishikawa, and Yuxin Ren. A compact high order finite volume method based on variational reconstruction for compressible flows on arbitrary grids. In *23rd AIAA Computational Fluid Dynamics Conference*, page 3097, 2017.

[22] Jialin Lou, Xiaodong Liu, Hong Luo, and Hiroaki Nishikawa. Reconstructed discontinuous galerkin methods for hyperbolic diffusion equations on unstructured grids. In *55th AIAA Aerospace Sciences Meeting, AIAA Paper*, volume 310, 2017.

[23] Jialin Lou, Lingquan Li, Hong Luo, and Hiroaki Nishikawa. Reconstructed discontinuous galerkin methods for linear advection–diffusion equations based on first-order hyperbolic system. *Journal of Computational Physics*, 369:103–124, 2018.

[24] Lingquan Li, Xiaodong Liu, Jialin Lou, Hong Luo, Hiroaki Nishikawa, and Yuxin Ren. A discontinuous galerkin method based on variational reconstruction for compressible flows on arbitrary grids. In *2018 AIAA Aerospace Sciences Meeting*, page 0831, 2018.

[25] Eleuterio F Toro, Michael Spruce, and William Speares. Restoration of the contact surface in the hll-riemann solver. *Shock waves*, 4(1):25–34, 1994.

[26] P Batten, MA Leschziner, and UC Goldberg. Average-state jacobians and implicit methods for compressible viscous and turbulent flows. *Journal of computational physics*, 137(1):38–78, 1997.

[27] Hong Luo, Joseph D Baum, and Rainald Lohner. High-reynolds number viscous flow computations using an unstructured-grid method. *Journal of Aircraft*, 42(2):483–492, 2005.

[28] Hong Luo, Joseph D Baum, and Rainald Löhner. A fast, matrix-free implicit method for compressible flows on unstructured grids. *Journal of Computational Physics*, 146(2):664–690, 1998.