Commun. Comput. Phys.
doi: 10.4208/cicp.210111.140711a

Vol. **12**, No. 1, pp. 284-314
July 2012

A Class of Hybrid DG/FV Methods for Conservation Laws III: Two-Dimensional Euler Equations

Laiping Zhang[1,2,*], Wei Liu[1], Lixin He[2] and Xiaogang Deng[1,2]

[1] *State Key Laboratory of Aerodynamics, China Aerodynamics Research and Development Center, Mianyang, Sichuan 621000, China.*
[2] *Computational Aerodynamics Institute, China Aerodynamics Research and Development Center, Mianyang, Sichuan 621000, China.*

Received 21 January 2011; Accepted (in revised version) 14 July 2011

Communicated by Chi-Wang Shu

Available online 30 January 2012

Abstract. A concept of ″static reconstruction″ and ″dynamic reconstruction″ was introduced for higher-order (third-order or more) numerical methods in our previous work. Based on this concept, a class of hybrid DG/FV methods had been developed for one-dimensional conservation law using a ″hybrid reconstruction″ approach, and extended to two-dimensional scalar equations on triangular and Cartesian/triangular hybrid grids. In the hybrid DG/FV schemes, the lower-order derivatives of the piecewise polynomial are computed locally in a cell by the traditional DG method (called as ″dynamic reconstruction″), while the higher-order derivatives are re-constructed by the ″static reconstruction″ of the FV method, using the known lower-order derivatives in the cell itself and in its adjacent neighboring cells. In this paper, the hybrid DG/FV schemes are extended to two-dimensional Euler equations on triangular and Cartesian/triangular hybrid grids. Some typical test cases are presented to demonstrate the performance of the hybrid DG/FV methods, including the standard vortex evolution problem with exact solution, isentropic vortex/weak shock wave interaction, subsonic flows past a circular cylinder and a three-element airfoil (30P30N), transonic flow past a NACA0012 airfoil. The accuracy study shows that the hybrid DG/FV method achieves the desired third-order accuracy, and the applications demonstrate that they can capture the flow structure accurately, and can reduce the CPU time and memory requirement greatly than the traditional DG method with the same order of accuracy.

AMS subject classifications: 76G25, 76H05, 76M10, 76M12, 76M25

Key words: Discontinuous Galerkin method, finite volume method, reconstruction, hybrid method.

[*] Corresponding author. *Email addresses:* zhanglp_cardc@126.com (L. P. Zhang), lw4992@gmail.com (W. Liu), hlx_cardc@126.com (L. X. He), xgdeng@skla.cardc.cn (X. G. Deng)

1 Introduction

While 2$^{\text{nd}}$ order methods are dominant in most compressible flow simulations, many types of problems, such as computational aeroacoustics (CAA), vortex-dominant flows and large eddy simulation (LES) of turbulent flows, call for higher order accuracy (third order and more). The main deficiency of widely available, second-order methods for the accurate simulations of the above-mentioned flows is the numerical diffusion and dissipation of vorticity to unacceptable level. Applications of high-order accurate, low-diffusion and low dissipation numerical methods can significantly alleviate this deficiency of the traditional second order methods, improve predictions of vortical and other complex, separated, unsteady flows. Therefore, various high-order methods have been developed in the last two decades, including the essentially non-oscillatory scheme (ENO) [1] and the weighted-ENO scheme (WENO) [2] on structured grids, the discontinuous Galerkin (DG) method [3–7], the ENO and WENO schemes [8–15], the spectral volume (SV) method [16–19], and the spectral difference (SD) method [20–22] on unstructured grids. Interested readers can refer to the comprehensive review articles for high-order methods by Ekaterinaris [23] on structured grids and by Wang [24] on unstructured grids. Because the structured/unstructured hybrid grid technique presents the trend of grid generation technique [25], due to the capability for complex geometries, the high-order methods on unstructured and hybrid (or mixed) grids are paid much more attention in recent years.

As the leader of high-order numerical methods for compressible flow computations in aerospace applications, the DG methods have recently become popular for problems with both complex physics and geometry. The DG method was originally developed by Reed and Hill to solve the neutron transport equation [3]. The development of high-order DG methods for hyperbolic conservation laws was pioneered by Cockburn, Shu and other collaborators in a series of papers on the Runge-Kutta DG (RKDG) method [4–7]. Many other researchers made significant contributions in the development. Refer to [26] for a comprehensive review on the DG method history and literature. The most distinguished feature of the DG methods is the ”compact” property on arbitrary grids.

However, the DG methods have a number of their own weaknesses, concentrating on the huge computational cost (memory requirement and CPU time). The block diagonal matrix requires a storage of $(ndof \times neqs) \times (ndof \times neqs) \times nelems$, where $ndof$ is the number of degrees of freedom (DOFs) for the polynomial, $neqs$ is the number of components in solution vector and $nelems$ is the total cell number of the grid. For example, the storage of this block diagonal matrix alone requires 10,000 words per element for a fourth-order DG scheme in 3D [27]! Indeed, the lack of efficient solver is one of the bottlenecks in the development of the DG methods for solving realistic problems.

Comparing with the traditional 2$^{\text{nd}}$ order DG method, the widely available 2$^{\text{nd}}$ order finite volume (FV) methods, as well as the finite difference (FD) methods, need smaller memory and computation cost, because they do not have to compute the volume integrals and the additional equations for the DOFs corresponding to the derivatives. In order to achieve high-order accuracy, Barth and Frederichson proposed the k-exact FV schemes [28] by increasing the grid stencil, which can be used for arbitrary high-order reconstruction on triangular or tetrahedral meshes. Following the ideas of high-order ENO/WENO FD schemes, the implementation of ENO for unstructured grids was developed by Abgrall [8], while WENO schemes for triangular meshes were developed by Friedrich [9], Hu and Shu [10], and others [11–15]. Similarly, large grid stencil is used here for higher-order reconstructions. Theoretically, arbitrary higher-order FV methods can be obtained with higher-order polynomial reconstruction following the idea of k-exact FV schemes on a larger grid stencil. However, some special non-linear WENO reconstruction approaches should be introduced. On the other hand, the large memory is required to store the reconstruction coefficients. Due to the large stencil, these high-order FV methods can not keep the "compact" feature of DG methods. In addition, the matrix computations increase the computational cost tremendously, which the computational cost may be one or more order times more than those of FD methods with same order on structured grids, especially for 3D cases.

Unlike the above high-order FV methods with increasing grid stencil and the high-order DG methods with Galerkin finite element (FE) formulations, Wang and Liu developed originally the SV method [16–19]. The SV method can be viewed as an extension of the Godunov method to higher order by adding more DOFs in the form of subcells in each cell (simplex). The simplex is referred to as a spectral volume (SV) and the subcells are referred to as control volumes (CV). As in the traditional FV method, the unknowns (or DOFs) are the subcell-averaged solutions. However, the stability hinges on how the element is partitioned into subcells. The partition of hybrid mesh cells in 3D to achieve stability for high-order elements is still unresolved. Recently, some researchers proposed the SD methods [20–22] in the differential framework. In the SD method, two sets of points, i.e., the solution points and flux points are defined in each element. The solution points are the locations where the nodal values of the state variable are specified. Flux points are the locations where the nodal values of fluxes are computed. The DOFs in the SD method are the conservative variables at the solution points. Similar to the SV method, more DOFs should be specified in each cell to increase the order of polynomial. Moreover, it appears that linear stability is difficult to achieve for high-order schemes on triangular and tetrahedral elements.

As discussed above, all the above-mentioned high-order methods have their distinguished features, and there is still room for improvement. So, a natural choice is to combine their advantageous features, especially on the optimal in terms of efficiency in both memory and CPU time for 3D realistic complex configurations. There are three kinds of hybrid approaches in references: 1) different schemes for inviscid and viscous flux discretization; 2) hybrid approach based on domain decomposition; 3) hybrid approach based on local polynomial reconstruction. Thareja et al. [29] and Luo et al. [30] adopted the first kind of hybrid approaches. They computed the inviscid flux with traditional FV methods, while the viscous flux with the standard DG methods. These methods improved the discretization accuracy for viscous term. However, they are still essentially

2nd-order FV methods. Based on the computational domain decomposition, He et al. [31] developed a hybrid FE/FV solver to simulate heat flux on 3D hybrid grids. They adopt the standard DG method nearby the wall, while the traditional 2nd-order FV method was implemented in far field. The numerical tests demonstrated that this hybrid approach improved the computational efficiency for 3D complex problems. However, it will suffer the same difficulty in both memory and CPU time for higher-order formulation. So only 2nd-order applications were carried out in their paper.

For the third kind of hybrid approach, very recently, Dumbser et al. [32–34] presented a unified framework for the construction of fully-discrete and high-order quadrature-free one step FV and DG schemes on unstructured meshes, and a class of P_NP_M schemes was developed. In the P_NP_M schemes, a conservative least-square polynomial reconstruction operator is applied to the standard DG method. In a first instance, piecewise polynomials of degree N are used as test functions as well as to represent the data in each element at the beginning of a time step. The time evolution of these data and the flux computation, however, are then done with a different set of piecewise polynomials of degree $M \geq N$, which are reconstructed from the underlying polynomials of degree N. This approach yields a general, unified framework that contains as two special cases classical high order FV schemes $(N=0)$, as well as the usual DG method $(N=M)$. The numerical tests demonstrated that it can save memory and CPU time by local polynomial reconstructions.

In fact, the Hermit WENO (HWENO) schemes proposed by Qiu and Shu [35,36] were the earlier experiment of hybrid reconstruction on structured grids. The construction of HWENO schemes was based on a finite volume formulation, Hermite interpolation, and nonlinearly stable Runge-Kutta methods. The idea of the reconstruction in the HWENO schemes comes from the original WENO schemes [2], however both the function and its first derivative values are evolved in time and used in the reconstruction, while only the function values are evolved and used in the original WENO schemes. One major advantage of HWENO schemes is its compactness in the reconstruction. For example in 1D case, five points are needed in the stencil for a fifth-order WENO reconstruction, while only three points are needed for a fifth-order HWENO reconstruction. For this reason, the HWENO finite volume methodology is more suitable to serve as limiters for the RKDG methods. Similarly, Luo et al. proposed a reconstruction-based DG method [37, 38] for the compressible Navier-Stokes equations on arbitrary grids.

In our previous work [39–41], by comparing the discontinuous Galerkin (DG) method, the k-exact finite volume (FV) method and the lift collocation penalty (LCP) method [42, 43], a concept of ”static reconstruction” and ”dynamic reconstruction” had been introduced for high-order numerical methods. Based on this concept, a class of higher-order hybrid DG/FV schemes had been developed for one-dimensional conservation law using a ”hybrid reconstruction” approach [39, 40], and extended for two-dimensional scalar equations on unstructured grids and hybrid grids [41]. In the hybrid DG/FV schemes, the lower-order derivatives of the piecewise polynomial are computed locally in a cell by the DG method based on Taylor basis functions [27, 44] (called as "dynamic reconstruction"), while the higher-order derivatives are re-constructed by the "static reconstruction" of the FV method, using the known lower-order derivatives in the cell itself and in its adjacent neighboring cells. The hybrid DG/FV schemes had been successfully applied to solving the 1D linear and non-linear scalar conservation law, 1D Euler equations and 2D scalar equations on hybrid grids. The numerical results demonstrated the accuracy, and the super-convergence property was shown for the third-order hybrid DG/FV schemes. In this follow-up paper, the hybrid DG/FV scheme is extended to solve two-dimensional Euler equations. Some typical test cases are presented to demonstrate the performance of the hybrid DG/FV schemes, including the standard vortex evolution problem with exact solution, isentropic vortex/weak shock wave interaction, subsonic flows past a circular cylinder and a three-element airfoil (30P30N), transonic flow past a NACA0012 airfoil. The numerical results show that the hybrid DG/FV schemes achieve the desired third-order accuracy, and more importantly as expected, they can reduce the CPU time and memory requirement greatly than the traditional DG method with the same order of accuracy on the same mesh.

The remainder of this paper is organized as follows. The governing equations are described in Section 2. In Sections 3 and 4, we discuss briefly the concepts of "static reconstruction", "dynamic reconstruction" and "hybrid reconstruction" firstly, and then the hybrid DG/FV methods for 2D Euler equations are proposed. In Section 5, the accuracy study and applications for the hybrid DG/FV schemes are carried out with typical cases to validate the performance. Finally, conclusions and some possibilities for future work are given in Section 6.

2 Governing equations

The two-dimensional Euler equations governing unsteady compressible inviscid flows can be expressed in following conservation forms as

$$\frac{\partial \boldsymbol{U}}{\partial t}+\frac{\partial \boldsymbol{F}_1(\boldsymbol{U})}{\partial x}+\frac{\partial \boldsymbol{F}_2(\boldsymbol{U})}{\partial y}=0, \tag{2.1}$$

where the conservation state vector $\boldsymbol{U}$ and inviscid vector $\boldsymbol{F}_1$, $\boldsymbol{F}_2$ are defined by

$$\boldsymbol{U}=(\rho,\rho u,\rho v,e)^\tau, \tag{2.2a}$$

$$\boldsymbol{F}_1(\boldsymbol{U})=\big(\rho u,\rho u^2+p,\rho uv,u(e+p)\big)^\tau,\quad \boldsymbol{F}_2(\boldsymbol{U})=\big(\rho v,\rho uv+p,\rho v^2,v(e+p)\big)^\tau, \tag{2.2b}$$

$$p=(\gamma-1)\Big[e-\frac{1}{2}\rho(u^2+v^2)\Big],\qquad S=\frac{p}{\rho^\gamma}, \tag{2.2c}$$

where ρ, p, e, S denote the density, the pressure, the specific total energy and the entropy of the fluid, respectively and u and v are the velocity components of the flow in the coordinate x and y. $\gamma=1.4$, is the ratio of the specific heat.

3 Static and dynamic reconstruction

3.1 Review of the DG method

The semi-discrete DG formulation of the Eq. (2.1) is obtained by multiplying by a test function W in a cell (Ω_e) and performing an integration by parts:

$$\frac{\partial}{\partial t}\int_{\Omega_e}\boldsymbol{U}_h^{(e)}Wd\Omega+\int_{\Gamma_e}\boldsymbol{F}\cdot\vec{\boldsymbol{n}}Wd\Gamma-\int_{\Omega_e}\boldsymbol{F}\cdot\nabla Wd\Omega=0, \tag{3.1}$$

where $\Gamma_e=\partial\Omega_e$ denotes the boundary of Ω_e, and $\vec{\boldsymbol{n}}$ the unit outward normal vector to boundary.

Let $\boldsymbol{U}_h^{(e)}\in\mathbb{V}_p^N$, where $\mathbb{V}_P^N$ is spanned by basis function $B_l(x,y)$

$$\mathbb{V}_P^N=span\{B_0,B_1\cdots,B_{N-1}\}. \tag{3.2}$$

Thus, $\boldsymbol{U}_h^{(e)}$can be expressed as

$$\boldsymbol{U}_h^{(e)}(t,x,y)=\sum_{l=0}^{N-1}\boldsymbol{U}_l^{(e)}B_l(x,y). \tag{3.3}$$

The coefficients $\boldsymbol{U}_l^{(e)}$ are the unknowns, generally, we call them degree of freedoms (DOFs). For Galerkin finite element method, the test function $W\in\mathbb{V}_p^N$, thus, Eq. (3.1) is equivalent to

$$\frac{\partial}{\partial t}\int_{\Omega_e}\boldsymbol{U}_h^{(e)}B_ld\Omega+\int_{\Gamma_e}\boldsymbol{F}\cdot\vec{\boldsymbol{n}}B_ld\Gamma-\int_{\Omega_e}\boldsymbol{F}\cdot\nabla B_ld\Omega=0. \tag{3.4}$$

And a common flux $\boldsymbol{H}$ is used to replace the normal flux $(\boldsymbol{F}\cdot\vec{\boldsymbol{n}})$ to provide element coupling. Substituting into Eq. (3.4), we obtain a system of

$$\sum_{l=0}^{N-1}\Big(\int_{\Omega_e}B_lB_k\frac{\partial\boldsymbol{U}_l^{(e)}}{\partial t}d\Omega\Big)+\int_{\Gamma_e}\boldsymbol{H}B_kd\Gamma-\int_{\Omega_e}\boldsymbol{F}\cdot\nabla B_kd\Omega=0,\quad k=0,1\cdots,N-1. \tag{3.5}$$

Or, the semi-discrete form is

$$\boldsymbol{M}\frac{d}{dt}(\boldsymbol{U}_l^{(e)})=(\boldsymbol{Rhs}_l), \tag{3.6}$$

in which $\boldsymbol{M}=(B_{kl})$ is called as the mass matrix.

3.2 Review of the LCP method

Huhyn [42] unified the DGM and SD/SV methods in 1D with the introduction of the flux reconstruction (FR) method. Wang and Gao [43] extended the idea to 2D triangular and mixed grids with the lifting collocation penalty (LCP) formulation, and applied this formulation to solve the 2D Euler equations.

Let W be an arbitrary weighting function, and $\boldsymbol{U}_h^{(e)}$ be an approximate solution of $\boldsymbol{U}$ at Ω_e. Generally, $\boldsymbol{U}_h^{(e)}$ does not satisfy Eq. (2.1) exactly, but

$$\frac{\partial \boldsymbol{U}_h^{(e)}}{\partial t}+\nabla\cdot\boldsymbol{F}(\boldsymbol{U}_h^{(e)})+\varepsilon^{(e)}=0, \tag{3.7}$$

where $\varepsilon^{(e)}$ is a residual, which should be zero as close as possible. The weighted residual method supposes that the integration over Ω_e derived by multiplying the residual with the weight function W is zero

$$\int_{\Omega_e}\varepsilon^{(e)}Wd\Omega=0, \tag{3.8}$$

which means that

$$\begin{aligned}&\int_{\Omega_e}\Big(\frac{\partial \boldsymbol{U}_h^{(e)}}{\partial t}+\nabla\cdot\boldsymbol{F}(\boldsymbol{U}_h^{(e)})\Big)Wd\Omega\\ =&\frac{\partial}{\partial t}\int_{\Omega_e}\boldsymbol{U}_h^{(e)}Wd\Omega+\int_{\Gamma_e}\boldsymbol{F}\cdot\vec{\boldsymbol{n}}Wd\Gamma-\int_{\Omega_e}\boldsymbol{F}(\boldsymbol{U}_h^{(e)})\cdot\nabla Wd\Omega=0.\end{aligned} \tag{3.9}$$

Since the solution is discontinuous across element interfaces, the surface integral in Eq. (3.9) is not well-defined. To remedy this problem, a common flux $\hat{\boldsymbol{F}}\equiv\hat{\boldsymbol{F}}(\boldsymbol{U}_h^{+},\boldsymbol{U}_h^{+},\vec{\boldsymbol{n}})$ is used to replace the normal flux to provide element coupling. Then Eq. (3.9) becomes

$$\frac{\partial}{\partial t}\int_{\Omega_e}\boldsymbol{U}_h^{(e)}Wd\Omega+\int_{\Gamma_e}\hat{\boldsymbol{F}}Wd\Gamma-\int_{\Omega_e}\boldsymbol{F}(\boldsymbol{U}_h^{(e)})\cdot\nabla Wd\Omega=0. \tag{3.10}$$

Applying integration by parts to the last term on the left-hand side of Eq. (3.10), we obtain

$$\int_{\Omega_e}\Big(\frac{\partial \boldsymbol{U}_h^{(e)}}{\partial t}+\nabla\cdot\boldsymbol{F}(\boldsymbol{U}_h^{(e)})\Big)Wd\Omega+\int_{\Gamma_e}[\hat{\boldsymbol{F}}-\boldsymbol{F}\cdot\vec{\boldsymbol{n}}]Wd\Gamma=0. \tag{3.11}$$

The last term in Eq. (3.11) can be viewed as a penalty term, i.e., penalizing the normal flux differences. Introduce a "correction field" belonging to the space of polynomials of degree k or less, $\delta^{(e)}\in P^k$, which is determined from a "lifting operator"

$$\int_{\Omega_e}\delta^{(e)}Wd\Omega=\int_{\Gamma_e}[\widetilde{\boldsymbol{F}}]Wd\Gamma, \tag{3.12}$$

where $[\widetilde{\boldsymbol{F}}]=\hat{\boldsymbol{F}}-\boldsymbol{F}\cdot\vec{\boldsymbol{n}}$ is the normal flux jump. Substituting Eq. (3.12) into Eq. (3.11), we obtain

$$\int_{\Omega_e}\Big(\frac{\partial \boldsymbol{U}_h^{(e)}}{\partial t}+\nabla\cdot\boldsymbol{F}(\boldsymbol{U}_h^{(e)})+\delta^{(e)}\Big)Wd\Omega=0. \tag{3.13}$$

Because W is arbitrary, Eq. (3.13) is equivalent to

$$\frac{\partial \boldsymbol{U}_h^{(e)}}{\partial t}+\nabla\cdot\boldsymbol{F}(\boldsymbol{U}_h^{(e)})+\delta^{(e)}=0. \tag{3.14}$$

Eq. (3.14) is satisfied everywhere in Ω_e. With the definition of a correction field $\delta^{(e)}$, the weighted residual formulation is reduced to an equivalent simple differential form successfully. Next let the DOFs be the solution at a set of points $\mathbf{x}_j^{(e)}$, named solution points (SPs). Then Eq. (3.14) must be true at the SPs, i.e.,

$$\frac{\partial \boldsymbol{U}_j^{(e)}}{\partial t}+\nabla\cdot\boldsymbol{F}(\boldsymbol{U}_j^{(e)})+\delta^{(e)}=0. \tag{3.15}$$

Using special correction field $\delta^{(e)}$, Wang successfully unified DGM, SVM, and SDM for Euler equations [43].

3.3 Review of the k-exact FV method

The high-order k-exact method was developed by Barth and Frederickson [28]. It belongs to a general class of Godunov-type FV methods. Applying the integral form of the Euler equations (2.1) to control volume Ω_e, and using Gauss formula, we obtain the following semi-discrete FV scheme

$$\frac{\partial \bar{\boldsymbol{U}}^{(e)}}{\partial t}+\frac{1}{|\Omega_e|}\sum_{f\in\partial\Omega_e}\int_f \boldsymbol{F}\cdot\vec{\boldsymbol{n}}dS=0. \tag{3.16}$$

The k-exactness property means that if the solution is a degree of k or lower polynomial, the solution is reconstructed exactly. If the piecewise polynomial in Ω_e is assumed as Eq. (3.3), we need at least $k-1$ neighboring cells in the local stencil to compute the coefficients since there is only one data in each target cell. To determine these coefficients, we again require that the polynomial is mean preserving in the local stencil

$$\frac{1}{|\Omega_j|}\int_{\Omega_j}\boldsymbol{U}_h^{(e)}d\Omega=\bar{\boldsymbol{U}}^{(j)},\quad \forall j\in S_e. \tag{3.17}$$

A compact form of Eq. (3.17) is

$$\boldsymbol{A}\boldsymbol{U}^{(e)}=\boldsymbol{b}. \tag{3.18}$$

From Eq. (3.18), we can see that there must be at least $k-1$ neighboring cells in S_e to obtain a solution. Generally speaking, the larger the stencil, the more stable the reconstruction is. In practice, in order to prevent singular stencils, more than $k-1$ neighboring cells are included in the stencil, therefore, Eq. (3.18) is solved in the least squares sense, which can be expressed using the pseudo-inverse matrix $\boldsymbol{A}$.

3.4 "Static" reconstruction and "dynamic" reconstruction

The above-mentioned high-order methods on unstructured grids, such as the LCP methods, the k-exact FV method and the DG methods, have a common feature: they achieve higher order accuracy by approximating the solutions as a higher order polynomial in each cell or element. Therefore the solution space is piecewise discontinuous polynomials of degree p, with a $(p+1)^{\text{th}}$ order of accuracy. Differently, the LCP method and the DG method approach the piece-wise polynomial locally, while k-exact FVM methods need the help of neighboring cells. Due to the local approximation only in each cell itself, LCP and DG methods are ″compact″, which means that each cell is de-coupled with its face adjacent neighbors. So these ″compact″ high-order methods are much more suitable for massively parallel computing because the communication of data between sub-blocks of grid is minimized, especially for implicit time evaluation. The difference between these methods lies in the reconstruction of DOFs and how the DOFs are updated. So, the reconstruction operator is crucial for the construction of numerical method.

Therefore, by comparing the DG method and the FV method, as well as the LCP method, we found that:

- In the DG method and the LCP method, all DOFs are time-dependent. They are computed via Galerkin finite element method, and ″extracted″ from governing equations. We think that it is one of the main reasons why the second order DG method are more accurate than the FV method with the same order, and why DG method need more memory and computation cost. So we call the DG method and the LCP method as a ″dynamic reconstruction″ method, or ″time-dependent″ reconstruction.

- In the k-exact FV method, the higher-order derivatives $(p>0)$ are re-constructed firstly using the cell-averaged value on a specified grid stencil. For example, in traditional second-order FV schemes, the first-order derivatives $(p=1)$ are usually solved by the well-known Green's formulation or the least-square approach. In other word, the reconstruction operator is applied at the final output time, and therefore we call this kind of reconstructions as a post-processing technique. So, for FV methods, the higher-order derivatives have nothing to do with the governing equations, and are computed with time-independent reconstruction operator. Therefore, we call this kind of approach as ″static reconstruction″ or ″grid-dependent″ reconstruction.

4 Hybrid reconstruction and DG/FV schemes

DG and LCP methods are ″compact″, since they reconstruct the high-order piece-wise polynomial locally, instead of extending grid stencil like k-exact FVM. As mentioned in Section 3, constructing piecewise polynomial in each cell is the key step for numerical method. In DGM, all DOFs are updated as time evolution (dynamic reconstruction), while in FVM, high-order derivatives are computed at the final output time (static reconstruction). Generally, ″dynamic reconstruction″ needs more CPU time and memory requirement to deal with all the DOFs. However, ″static reconstruction″ operator is unnecessary to update all DOFs at a time step, some of the DOFs can be computed by reconstruction with the help of neighboring cells, which results in a non-compact method. Naturally, a ″hybrid reconstruction″ approach can be adopted to achieve some balance between the ″compact″ property and the computational cost. Following this idea, the authors had successfully developed a class of hybrid DG/FV schemes for one-dimensional linear and non-linear scalar equation and Euler equations in [39,40], and for two-dimensional scalar equations on hybrid grids in [41]. Firstly, we calculate the lower-order derivatives using ″dynamic reconstruction″ of 2nd order DG formulation based on Taylor basis [27]. Then, following the idea of FV methods, higher-order derivatives are computed by ″static construction″ with the help of face neighboring cells. Once the higher-order derivatives are re-constructed, the higher-order polynomial can be obtained to achieve higher-order numerical solution. If we choose the set of derivatives, as well as the cell-averaged values, as the basis functions B_l like those in the DG method based on Taylor basis functions, we have,

$$\boldsymbol{U}_h=\sum_{l=1}^{M}\boldsymbol{U}_lB_l=\sum_{l=1}^{N}\boldsymbol{U}_lB_l+\sum_{l=N+1}^{M}\boldsymbol{U}_lB_l, \tag{4.1}$$

in which the first term in the right hand side of Eq. (4.1) represents the lower-order degrees $(\leq N)$ of the polynomial to be re-constructed by the DG method, while the second term represents the higher-order degrees $(>N)$ to be re-constructed directly by the FV method. Because of fewer DOFs of reconstruction polynomial are computed by the DG method, the hybrid schemes are expected to reduce the memory and computation cost, especially for 3D realistic applications.

4.1 Basis functions and distribution of DOFs

In the traditional DGM and the LCP methods, the numerical polynomial solution in each element are represented using standard Lagrange finite element or hierarchical node-based basis as shown in Fig. 1. As a result, the unknowns to be solved are the variables at the nodes, as illustrated in Fig. 1 for linear and quadratic polynomial approximations.

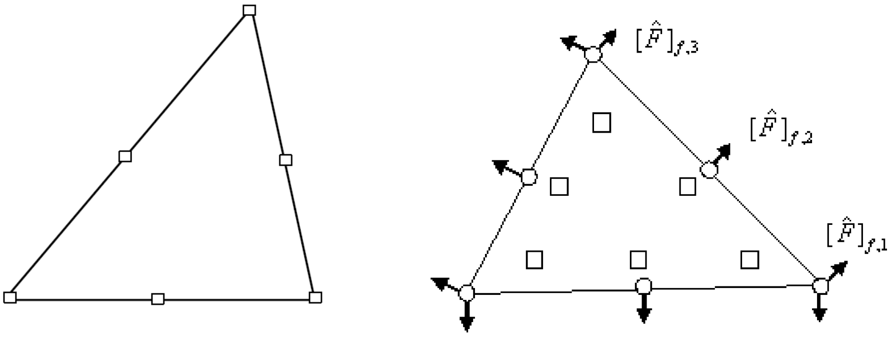


Figure 1: The DOFs distribution (squares) of DGM and the LCP method (refer to [43]).

In [27], Luo et al. developed a DG method based on Taylor basis functions. Similarly in [44], the Taylor expansion was adopted to develop a hierarchical reconstruction procedure and used it as a limiter for the central DG scheme. Unlike the traditional DG methods, in which either standard Lagrange finite element or hierarchical node-based basis functions are used to represent numerical polynomial solutions in each cell, this DG method represents the numerical polynomial solutions using a Taylor series expansion at the centroid of the target cell. Consequently, this formulation is able to provide a unified framework, where both cell-centered and vertex-centered finite volume schemes can be viewed as special cases of this DG method. Also, Taylor basis functions are hierarchical, which is convenient to handle *p*- or *hp*- multigrid strategy [45,46]. In present work, the numerical polynomial solutions are represented using a Taylor series expansion at the centroid. For sake of simplicity and easy presentation of the main ideas, let us consider P_2 approximation in 2D, where numerical solution in each cell is approximated using a quadratic polynomial. If we do a Taylor series expansion at the centroid of the element Ω_e, the quadratic polynomial solutions in $\boldsymbol{U}_h^{(e)}$ can be expressed as follows:

$$\begin{aligned}\boldsymbol{U}_h^{(e)}=\boldsymbol{U}_c^{(e)}&+\frac{\partial\boldsymbol{U}_c^{(e)}}{\partial x}(x-x_c^{(e)})+\frac{\partial\boldsymbol{U}_c^{(e)}}{\partial y}(y-y_c^{(e)})+\frac{1}{2}\frac{\partial^2\boldsymbol{U}_c^{(e)}}{\partial x^2}(x-x_c^{(e)})^2\\&+\frac{1}{2}\frac{\partial^2\boldsymbol{U}_c^{(e)}}{\partial y^2}(y-y_c^{(e)})^2+\frac{\partial^2\boldsymbol{U}_c^{(e)}}{\partial x\partial y}(x-x_c^{(e)})(y-y_c^{(e)}),\end{aligned}\tag{4.2}$$

where $(x_c^{(e)},y_c^{(e)})$ is the centroid of Ω_e. For convenience, we let the local coordinates

$$\xi^{(e)}=\frac{x-x_c^{(e)}}{h^{(e)}},\qquad \eta^{(e)}=\frac{y-y_c^{(e)}}{h^{(e)}},\tag{4.3}$$

in which $h^{(e)}$ is the characteristic scale of a cell for normalization, it can be defined as the radius of the circumcircle of the cell. Here, we let $h^{(e)}:=\sqrt{|\Omega_e|}$ for simplicity. In present paper, we sometimes write ξ and η as when there is no confusion. Furthermore, $\boldsymbol{U}_h^{(e)}$ can be expressed as cell-averaged value and its derivatives about local coordinate at the centroid of the cell

$$\begin{aligned}\boldsymbol{U}_h^{(e)}=\sum_{l=0}^{5}\boldsymbol{U}_l^{(e)}B_l=\overline{\boldsymbol{U}}^{(e)}&+\frac{\partial\boldsymbol{U}_c^{(e)}}{\partial\xi}\xi+\frac{\partial\boldsymbol{U}_c^{(e)}}{\partial\eta}\eta+\frac{1}{2}\frac{\partial^2\boldsymbol{U}_c^{(e)}}{\partial\xi^2}(\xi^2-M_{20}^{(e,e)})\\&+\frac{1}{2}\frac{\partial^2\boldsymbol{U}_c^{(e)}}{\partial\eta^2}(\eta^2-M_{02}^{(e,e)})+\frac{\partial^2\boldsymbol{U}_c^{(e)}}{\partial\xi\partial\eta}(\xi\eta-M_{11}^{(e,e)}),\end{aligned}\tag{4.4}$$

where, $M_{pl}^{(e,e)}$ is called as the centroid-moment of Ω_e, which is defined as:

$$M_{pl}^{(e,e)}=\frac{1}{|\Omega_e|}\int_{\Omega_e}(\xi^{(e)})^p(\eta^{(e)})^l d\Omega=\frac{1}{|\Omega_e|}\int_{\Omega_e}\frac{(x-x_c^{(e)})^p(y-y_c^{(e)})^l}{(h^{(e)})^{p+l}}d\Omega.\tag{4.5}$$

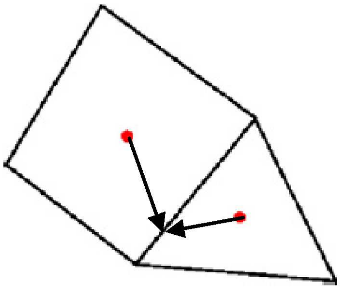


Figure 2: The distribution of DOFs of DGM based on Taylor basis functions.

Thus, the six basis functions of quadratic polynomial space for two dimensional cases are

$$B_0=1, \qquad B_1=\xi, \qquad B_2=\eta, \tag{4.6a}$$

$$B_3=\frac{1}{2}(\xi^2-M_{20}^{(e,e)}), \qquad B_4=\frac{1}{2}(\eta^2-M_{02}^{(e,e)}), \qquad B_5=\xi\eta-M_{11}^{(e,e)}. \tag{4.6b}$$

The unknowns (the DOFs) to be solved in DG formulation are the cell-averaged value and a set of derivatives

$$\boldsymbol{U}^{(e)}:=(\boldsymbol{U}_l^{(e)})=\Big(\overline{\boldsymbol{U}}^{(e)},\frac{\partial \boldsymbol{U}_c^{(e)}}{\partial\xi},\frac{\partial \boldsymbol{U}_c^{(e)}}{\partial\eta},\frac{\partial^2 \boldsymbol{U}_c^{(e)}}{\partial\xi^2},\frac{\partial^2 \boldsymbol{U}_c^{(e)}}{\partial\eta^2},\frac{\partial^2 \boldsymbol{U}_c^{(e)}}{\partial\xi\partial\eta}\Big), \tag{4.7}$$

which are unified for different shape of elements, as illustrated in Fig. 2.

4.2 Dynamic reconstruction of the lower-order DOFs

Similar to the one-dimensional hybrid DG/FV schemes in [39, 40] and the two-dimensional hybrid DG/FV schemes in [41], we update the DOFs using a two-step reconstruction approach. Firstly, adopting the DG method, the lower-order DOFs are computed. We call this operator as the "dynamic reconstruction". In the lower-order piecewise polynomial space $\mathbb{V}_P^N=span\{B_0,B_1,B_2\}$, the DGM formulation based on Taylor basis functions according to Eq. (3.6) can be expressed as:

$$\frac{d\bar{\boldsymbol{U}}^{(e)}}{dt}=-\frac{1}{|\Omega_e|}\sum_{f\in\partial\Omega_e}\int_f \boldsymbol{H}dS, \qquad \boldsymbol{M}\frac{d}{dt}\left(\begin{array}{c}\dfrac{\partial \boldsymbol{U}_c^{(e)}}{\partial\xi}\\ \dfrac{\partial \boldsymbol{U}_c^{(e)}}{\partial\eta}\end{array}\right)=\frac{1}{h^{(e)}}\left(\begin{array}{c}\boldsymbol{Rhs}_1\\ \boldsymbol{Rhs}_2\end{array}\right). \tag{4.8}$$

Note that in this formulation, the formulation for the cell-averaged variables is decoupled from the equations for the first-order derivatives. The right-hand sides $\boldsymbol{Rhs_1}$, $\boldsymbol{Rhs_2}$, are

$$\boldsymbol{Rhs}_1=\frac{1}{|\Omega_e|}\int_{\Omega_e}\boldsymbol{F}_1 d\Omega-\frac{1}{h^{(e)}}\int_{\Gamma_e}\boldsymbol{H}B_1 d\Gamma, \qquad \boldsymbol{Rhs}_2=\frac{1}{|\Omega_e|}\int_{\Omega_e}\boldsymbol{F}_2 d\Omega-\frac{1}{h^{(e)}}\int_{\Gamma_e}\boldsymbol{H}B_2 d\Gamma. \tag{4.9}$$

The mass matrix is

$$\boldsymbol{M}=\begin{pmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{pmatrix}=\begin{pmatrix} M_{20}^{(e,e)} & M_{11}^{(e,e)} \\ M_{11}^{(e,e)} & M_{02}^{(e,e)} \end{pmatrix}, \tag{4.10}$$

where $M_{pl}^{(e,e)}$ is the centroid-moment as Eq. (4.5). $M_{pl}^{(e,e)}$ can be exactly computed by proper Gauss integration.

Note that in Eq. (4.8), the lower-order DOFs (the cell-averaged value and the first-order derivatives) can be computed with time advancing in all cells. The higher-order DOFs are needed for high-order reconstruction polynomial. For example, the 2nd order derivatives are needed if we want to obtain a third-order scheme. In DGM, these higher-order DOFs are solved in the same manner as those of the mean variables and the first-order derivatives, therefore, more memory storage and CPU time are required to deal with them. Inspired by FVM, we can calculate the higher-order DOFs in an extending grid stencil (only include the face neighboring cells). This step is the "static reconstruction" as following.

4.3 Static reconstruction of the higher-order DOFs

After the "dynamic reconstruction" with the DGM based on Taylor basis functions, the lower-order derivatives (i.e., the cell-averaged values and the first-order derivatives) have been calculated. It means that there are more information than FVM in each cell so that we have more possible choice to reconstruct higher-order derivatives.

Similar to the traditional FVMs, if the first order derivatives are taken to construct the second order derivatives with Green's theorem, we can obtain

$$u_{xx}=\frac{1}{|\Omega_G|}\int_{\Omega_G}\frac{\partial^2 u}{\partial x^2}d\Omega=\frac{1}{|\Omega_G|}\sum_{\Gamma_i\in\partial\Omega}\int_{\Gamma_i}\frac{\partial u}{\partial x}n_x^{(i)}d\Gamma. \tag{4.11}$$

Similarly,

$$u_{yy}=\frac{1}{|\Omega_G|}\int_{\Omega_G}\frac{\partial^2 u}{\partial y^2}d\Omega=\frac{1}{|\Omega_G|}\sum_{\Gamma_i\in\partial\Omega}\int_{\Gamma_i}\frac{\partial u}{\partial y}n_y^{(i)}d\Gamma, \tag{4.12a}$$

$$u_{xy}=\frac{1}{|\Omega_G|}\int_{\Omega_G}\frac{\partial^2 u}{\partial x\partial y}d\Omega=\frac{1}{|\Omega_G|}\sum_{\Gamma_i\in\partial\Omega}\int_{\Gamma_i}\frac{\partial u}{\partial x}n_y^{(i)}d\Gamma, \tag{4.12b}$$

$$u_{yx}=\frac{1}{|\Omega_G|}\int_{\Omega_G}\frac{\partial^2 u}{\partial y\partial x}d\Omega=\frac{1}{|\Omega_G|}\sum_{\Gamma_i\in\partial\Omega}\int_{\Gamma_i}\frac{\partial u}{\partial y}n_x^{(i)}d\Gamma. \tag{4.12c}$$

The cross-derivatives can be calculated by Eq. (4.12b) or Eq. (4.12c), here we take an arithmetic mean as approximation in our code. The choice of the Gauss integral region Ω_G is the main issue of this static reconstruction. In this paper, we take the target cell as Ω_G, i.e., $\Omega_G=\Omega_e$, as shown in Fig. 3. The first-order derivatives at nodes of Ω_G

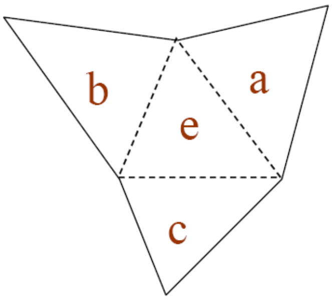


Figure 3: The selection of Gauss integral domain.

are calculated from the neighboring cells using a weighted-average approach. Here, the weight function is chosen as the reverse of the cell-center-to-node distance. For Cartesian grids with "hanging" node, same approach is adopted. This approach is only used to reconstruct the derivatives at nodes from the connecting cells, and do not coupled directly with the reconstruction of the high-order polynomial to compute the flux on the cell interface, so it can still keep the "compact" property. Meanwhile, the node-to-cell connections have been stored initially for post-process, so this approach will not increase the memory storage. We call this static reconstruction operator as the cell-vertex type (named as DG/FV-1) scheme. Anyway, other kinds of Gauss integral region can be adopted here, which had been shown in [41]. Moreover, the least-square approach can be adopted also to calculate the higher-order derivatives, as in [37,41].

4.4 Hybrid DG/FV scheme

Based on the hybrid reconstruction strategy above, we obtain the hybrid DG/FV schemes, the procedure is as follows

1. Partition the physical domain into a set of cells (grid generation). The initial condition is specified on the grid.
2. The cell-averaged values and the first derivatives are computed by the initial condition.
3. Reconstruct the higher-order DOFs (the second-order derivatives) by static reconstruction operator as discussed in Section 4.3.
4. Reconstruct the higher-order piecewise polynomial (quadratic polynomial) in each cell.
5. Calculate the residual items with appropriate order Gaussian quadrature, including the numerical flux and the domain integral, as discussed in Section 4.2.
6. By assembling together all the elemental contribution, a system of ordinary differential equations governing the evolution in time of the discretized solution can be written as

$$\boldsymbol{M}\frac{d}{dt}(\boldsymbol{U}_h)=\boldsymbol{Rhs}(\boldsymbol{U}_h). \tag{4.13}$$

7. The semi-discrete system can be integrated in time using the typical explicit three-stage third-order TVD Runge-Kutta scheme to update the cell-averaged values and the first derivatives. The procedure form Step 4 to Step 7 is the so-called "dynamic reconstruction".

$$\boldsymbol{U}_h^{(1)}=\boldsymbol{U}_h^n+\Delta t\boldsymbol{M}^{-1}\boldsymbol{Rhs}(\boldsymbol{U}_h^n), \tag{4.14a}$$

$$\boldsymbol{U}_h^{(2)}=\frac{3}{4}\boldsymbol{U}_h^n+\frac{1}{4}[\boldsymbol{U}_h^{(1)}+\Delta t\boldsymbol{M}^{-1}\boldsymbol{Rhs}(\boldsymbol{U}_h^{(1)})], \tag{4.14b}$$

$$\boldsymbol{U}_h^{n+1}=\frac{1}{3}\boldsymbol{U}_h^n+\frac{2}{3}[\boldsymbol{U}_h^{(2)}+\Delta t\boldsymbol{M}^{-1}\boldsymbol{Rhs}(\boldsymbol{U}_h^{(2)})]. \tag{4.14c}$$

8. Go to Step 3 to continue the cycle until time end.

Honestly, the idea of hybrid DG/FV schemes is very similar to that of the P_NP_M schemes proposed by Dumbser et al. [32–34]. In the P_NP_M schemes, piecewise polynomials of degree N are used as test (or basis) functions to represent the data in each element at the beginning of a time step. The time evolution of these data and the flux computation, however, are then done with a different set of piecewise polynomials of degree $(M \geq N)$. The lower-order reconstruction basis functions and the test functions representing the data are equal up to degree N, and the higher-order reconstruction $(> N)$ basis functions are chosen to be orthogonal to the lower-order test functions. For one-dimensional cases, if we choose the Legendre polynomials to form a local orthogonal basis, the present hybrid reconstructions are similar to those in the P_NP_M schemes. However, the present hybrid DG/FV schemes are different from the P_NP_M schemes in the following aspects. Firstly, the Taylor basis functions are adopted in this paper, while the traditional Lagrange finite element or hierarchical node-based functions are adopted in the P_NP_M schemes. Secondly, the higher-order derivatives (DOFs) are re-constructed directly by a polynomial as those in upwind FVMs, instead of using another set of orthogonal basis functions. Thirdly, the semi-discretization of the governing equations is considered only for discretization of the convection term, and a three-stage third order Runge-Kutta iteration method is adopted for time advancing, while a local continuous space-time Galerkin method is used in the P_NP_M schemes to construct a fully discrete scheme. Although the third-order scheme is presented only in this paper (corresponding to the cases of $N=1$ and $M=2$ in the P_NP_M schemes), the present hybrid reconstruction approach can be extended to construct higher-order schemes. For multi-dimensional cases, in order to extract the information of neighboring cells, a L_2-projection approach is adopted to re-construct the higher-order polynomial, so the least-square method is used to solve the higher-order DOFs in [37,38]. But in the present hybrid DG/FV schemes, the Green-Gauss formulation is used directly to re-construct the higher-order derivatives; this approach is relatively simpler to be implemented.

If we choose the traditional Lagrange finite element or hierarchical node-based functions, the unknowns to be solved are the variables at the nodes and the polynomial solutions are dependent on the shape of elements for multi-dimensional cases. For a linear polynomial approximation in 2D, a linear polynomial approximation is used for triangular elements and the unknowns to be solved are the variables at the three vertices and a bi-linear polynomial approximation is used for quadrilateral elements and the unknowns to be solved are the variables at the four vertices. Therefore, it is complicated to deal with the hybrid grids over complex geometries, especially in the cases with ″hanging″ nodes when using the well-known multi-level Cartesian grids or the adaptive grids, because the triangles and the quadrilateral elements in the physical space should be transformed into the canonical reference space with different mapping approaches. So only the results on ″pure″ triangles (2D) and tetrahedrons (3D) were shown in [32–34].

In the present work, the numerical polynomial solutions are represented using a Taylor series expansion at the centroid of the cell, which can be further expressed as a combination of cell-averaged values and their derivatives at the centroid of the cell. The unknowns to be solved in this formulation are the cell-averaged variables and their derivatives at the center of the cells. As mentioned in [27], this formulation has some distinguish advantages. Firstly, this formulation is suitable for any shapes of elements, which can be triangle, quadrilateral, and polygon in 2D, and tetrahedron, pyramid, prism, and hexahedron in 3D. Using this formulation, DG methods can be easily implemented on arbitrary meshes. Secondly, cell-averaged variables and their derivatives are handily available in this formulation. This makes implementation of a WENO limiter straightforward and efficient that is required to eliminate non-physical oscillations in the vicinity of discontinuities. Thirdly, the basis functions are hierarchic. This greatly facilitates implementation of p-multigrid methods and p-refinement [45,46]. Last, cell-averaged variable equations are decoupled from their derivatives equations in this formulation. This makes development of fast, low-storage implicit methods possible. That is the main reason why we choose this kind of basis functions.

4.5 Qualitative comparison of the computational cost

The computational cost is one of the main issues in this study. Before comparing the CPU time and memory for the test cases, we analysis qualitatively the computational cost for FVMs, DGMs, and the hybrid DG/FV schemes in two-dimensions (as shown in Table 1).

In FVMs, the cell-averaged values are defined only as the DOFs in each cell. In the case of linear (FVM2), quadratic (FVM3) polynomial, the boundary integrals are carried out using one or two Gauss points, respectively, and the domain integrals are unneces-

Table 1: Qualitative comparison of computational cost of FVM, DGM and hybrid DG/FV schemes in two-dimensional cases.

	FVM2	FVM3	DGM2	DGM3	DG/FV-1
nDOFs in a triangular cell	1	1	3	6	3
Accuracy order	$\mathcal{O}(h^2)$	$\mathcal{O}(h^3)$	$\mathcal{O}(h^2)$	$\mathcal{O}(h^3)$	$\mathcal{O}(h^3)$
Quadrature points for boundary integrals	1	2	2	3	3
Quadrature points for domain integrals	0	0	3	6	3
Storage for mass matrix	0×0	0×0	3×3	6×6	3×3
Storage for reconstruction matrix	2×2	5×5	0×0	0×0	0×0

sary. The FVMs employ only one DOF in each cell; as a result, the DOFs in the neighboring cell and the neighbors of neighboring cells from a local grid stencil are used to reconstruct a higher order polynomial in the cell. In DGMs, all the DOFs are defined within the target cell. This feature makes the DG method ideally suited for parallel computation. However, more DOFs should be calculated in each cell (as listed in Table 1), so the computational cost is more expensive. For the hybrid DG/FV schemes, the lower-order derivatives (lower-order DOFs) of the piecewise polynomial are computed locally in a cell by the DG method (DGM2), while the higher-order derivatives (higher-order DOFs) are re-constructed using the known lower-order DOFs in the cell itself and in its adjacent neighboring cells to ensure the compactness. It means that the solution of higher-order weak formulations in Eq. (3.5) or Eq. (3.6) corresponding to the higher-order DOFs is unnecessary. The weak formulation of the governing equations can be degenerated as those of DGM2, so less computational cost is required.

5 Accuracy study and applications

5.1 Vortex evolution problem

This is a standard test case for the Euler equations in 2D. The conditions of mean flow are $\{\rho,u,v,p\}=\{1,1,1,1\}$. An isotropic vortex is then added to the mean flow, i.e., with perturbations in u, v and the temperature $T=p/\rho$, and no perturbation in entropy S:

$$\delta u=\frac{\varepsilon}{2\pi}e^{(1-r^2)/2}(5-y), \qquad \delta v=\frac{\varepsilon}{2\pi}e^{(1-r^2)/2}(x-5), \tag{5.1a}$$

$$\delta T=\frac{(\gamma-1)\varepsilon^2}{8\gamma\pi^2}e^{1-r^2}, \qquad \delta S=0, \tag{5.1b}$$

where $r^2=(x-5)^2+(y-5)^2$, and the vortex strength ε=5.0. If the computational domain is large enough, the exact solution of the Euler equations with the above initial conditions is just the passive convection of the isotropic vortex with the mean velocity $(1,1)$. In this numerical simulation, the computational domain is taken to be $[0,10]\times[0,10]$, with exact inflow and outflow boundary condition imposed on the boundaries. The numerical simulation is carried out until $T=2$ on three kinds of different grids, a regular triangular mesh, a fully irregular mesh and a mixed mesh as shown in Fig. 4.

In Tables 2, 3 and 4, the cell-averaged density errors $(L_1,L_2$ and $L_\infty)$ are presented on the three kinds of grids for DGM2, DGM3, and the third-order hybrid DG/FV-1 schemes. The definitions of L_1, L_2 and L_∞ are listed as following:

$$L_1=\frac{1}{nTCell}\sum_{i=1}^{nTCell}(|\varepsilon_i|), \quad L_2=\Big(\frac{1}{nTCell}\sum_{i=1}^{nTCell}\varepsilon_i^2\Big)^{\frac{1}{2}}, \quad L_\infty=\mathrm{MAX}\{|\varepsilon_i|\}. \tag{5.2}$$

In which $nTCell$ is the total cell number, $\varepsilon_i=\bar{\rho}_i-\bar{\rho}_i^{ext}$ and the superscript $''ext''$ denotes the exact cell-average of density ρ. From Table 2, Table 3, and Table 4, we can see that

Table 2: Accuracy study of DGM2, DGM3, and DG/FV-1 for 2D Euler equations on regular mesh.

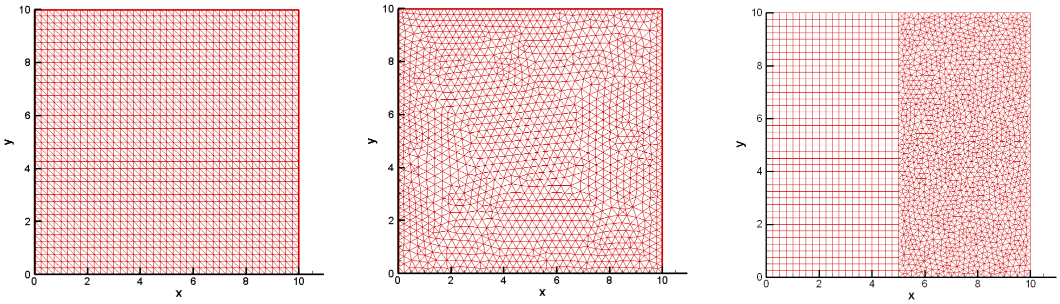
	Grid	L_1 error	Order	L_2 error	Order	L_∞ error	Order
2^{nd} order DGM	10×10	3.7384e-03		8.5678e-03		5.4759e-02	
	20×20	9.1448e-04	2.03	2.2040e-03	1.96	1.6644e-02	1.72
	40×40	1.8918e-04	2.27	4.6871e-04	2.23	4.3684e-03	1.93
	80×80	4.0898e-05	2.21	9.2790e-05	2.34	1.0014e-03	2.11
3^{rd} order DGM	10×10	3.8224e-04		7.6493e-04		3.7892e-03	
	20×20	3.6932e-05	3.37	8.5032e-05	3.17	6.4682e-04	2.55
	40×40	3.2582e-06	3.50	8.3437e-06	3.35	1.0414e-04	2.64
	80×80	2.6136e-07	3.64	8.3907e-07	3.32	1.2009e-05	3.12
DG/FV-1 scheme	10×10	2.6799e-03		6.2282e-03		3.9431e-02	
	20×20	3.1607e-04	3.08	9.4513e-04	2.72	7.8101e-03	2.34
	40×40	1.7906e-05	4.14	5.0985e-05	4.21	4.9749e-04	3.97
	80×80	1.0378e-06	4.11	2.7111e-06	4.23	2.1203e-05	4.55

Table 3: Accuracy study of DGM2, DGM3, and DG/FV-1 for 2D Euler equations on irregular mesh.

	Grid	L_1 error	Order	L_2 error	Order	L_∞ error	Order
2^{nd} order DGM	10×10	3.0709e-03		6.8638e-03		3.8634e-02	
	20×20	7.3911e-04	2.05	1.7337e-03	1.98	1.3825e-02	1.48
	40×40	1.4138e-04	2.38	3.1327e-04	2.47	2.6189e-03	2.40
	80×80	2.9902e-05	2.24	7.9528e-05	1.98	9.3041e-04	1.49
3^{rd} order DGM	10×10	3.0856e-04		6.2640e-04		3.4285e-03	
	20×20	5.0746e-05	2.60	1.1429e-04	2.45	6.5496e-04	2.39
	40×40	3.7432e-06	3.76	8.3926e-06	3.77	7.3331e-05	3.16
	80×80	5.3995e-07	2.79	1.4950e-06	2.49	3.0964e-05	1.24
DG/FV-1 scheme	10×10	1.4920e-03		3.3576e-03		2.1557e-02	
	20×20	2.9650e-04	2.33	8.0053e-04	2.07	9.0326e-03	1.25
	40×40	1.6883e-05	4.13	4.4650e-05	4.16	4.3925e-04	4.36
	80×80	1.8612e-06	3.18	4.2566e-06	3.39	4.8176e-05	3.12

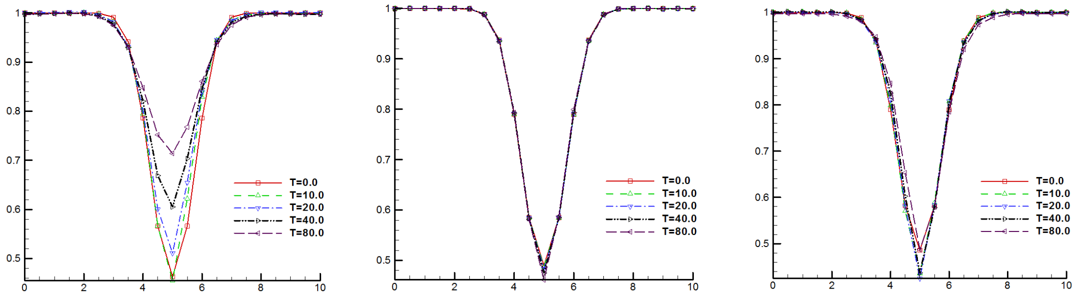
Table 4: Accuracy study of DGM2, DGM3, and DG/FV-1 for 2D Euler equations on the hybrid mesh.

	Grid	L_1 error	Order	L_2 error	Order	L_∞ error	Order
2^{nd} order DGM	10×10	4.0911e-03		8.4834e-03		6.2960e-02	
	20×20	9.4482e-04	2.11	2.0260e-03	2.07	1.3624e-02	2.21
	40×40	1.8333e-04	2.37	3.8350e-04	2.40	3.0649e-03	2.15
	80×80	3.4884e-05	2.39	7.3360e-05	2.38	6.1619e-04	2.31
3^{rd} order DGM	10×10	1.4068e-03		2.3522e-03		1.4995e-02	
	20×20	1.0212e-04	3.78	1.7711e-04	3.73	8.6630e-04	4.11
	40×40	8.6184e-06	3.57	1.6010e-05	3.46	1.1870e-04	2.87
	80×80	7.9091e-07	3.45	1.5806e-06	3.34	1.6993e-05	2.80
DG/FV-1 scheme	10×10	2.0911e-03		4.1537e-03		2.5111e-02	
	20×20	2.9828e-04	2.81	6.2987e-04	2.72	4.1649e-03	2.59
	40×40	2.1556e-05	3.79	4.5825e-05	3.78	3.8510e-04	3.43
	80×80	2.6244e-06	3.04	5.1905e-06	3.14	4.3163e-05	2.81



(a) regular grid (b) irregular grid (c) hybrid grid

Figure 4: Computational grids for isotropic vortex problem.



(a) DGM2 (b) DGM3 (c) DG/FV-1 scheme

Figure 5: Density profiles along $x=5$ at $t=0,10,20,80$.

the hybrid DG/FV scheme performs quite well on the three types of grids, achieving the desired third order of accuracy.

Next we test the DG/FV schemes for the long time evolution of the vortex. For this purpose, periodic boundary condition was employed at all the boundaries. The simulation was carried out on the coarse regular grid $(20\times20\times2)$. Fig. 5 shows the density profiles along $x=5$ at $t=0,10,20,80$ for the DGM2, DGM3 and the hybrid DG/FV-1 schemes. Note that the DGM2 displays significant dissipation errors, especially for the long time simulation. On the contrary, both the DGM3 and the hybrid DG/FV-1 scheme supply excellent results.

To demonstrate the advantage of the hybrid DG/FV-1 scheme, their computational costs are compared with those by the original DGM2 and DGM3, as listed in Table 5. The hybrid DG/FV-1 scheme, as well as DGM2 and DGM3 based on Taylor basis functions are programmed in a code, and run on a desktop computer running the *Linux* operator system. It can be seen that, comparing with DGM3, the memory requirement and CPU time are reduced by approximately 30% and 50%, respectively. We can expect that the hybrid DG/FV scheme be more efficient for the three-dimensional cases, which is very important for realistic applications.

Table 5: Comparison of the computational cost for 2D Euler equations.

	Grid	CPU Times (S)	Memory (M)
2nd order DGM	10×10	1.75E-02	0.304
	20×20	6.77E-02	0.778
	40×40	2.74E-01	2.700
	80×80	1.01E+00	8.300
3rd order DGM	10×10	4.50E-02	0.432
	20×20	1.73E-01	1.300
	40×40	7.22E-01	4.500
	80×80	2.95E+00	17.10
DG/FV-1 scheme	10×10	2.43E-02	0.352
	20×20	9.31E-02	0.988
	40×40	3.51E-01	3.400
	80×80	1.39E+00	11.20

5.2 Interaction of isentropic vortex and weak shock wave

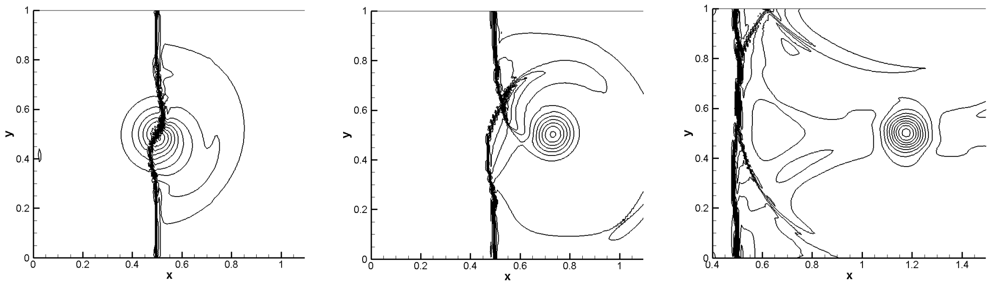
This problem describes the interaction between a stationary shock wave and a vortex, the flow conditions are the same as in [42]. The computational domain is set to be $[0,2]\times[0,1]$. A stationary shock with a shock Mach number of $Ms=1.1$ is positioned at $x=0.5$ and normal to the x-axis. Its left state is $(\rho,u,v,p)=(1,1.1,\sqrt{\gamma},0,1)$. An isentropic vortex is superposed to the flow left to the shock and centers at $(x_c,y_c)=(0.25,0.5)$, and the vortex conditions are

$$\delta u=\varepsilon\tau e^{\alpha(1-r^2)}\sin\theta, \qquad \delta v=-\varepsilon\tau e^{\alpha(1-r^2)}\cos\theta, \tag{5.3a}$$

$$\delta T=\frac{(\gamma-1)\varepsilon^2 e^{2\alpha(1-r^2)}}{4\alpha\gamma}, \qquad \delta S=0, \tag{5.3b}$$

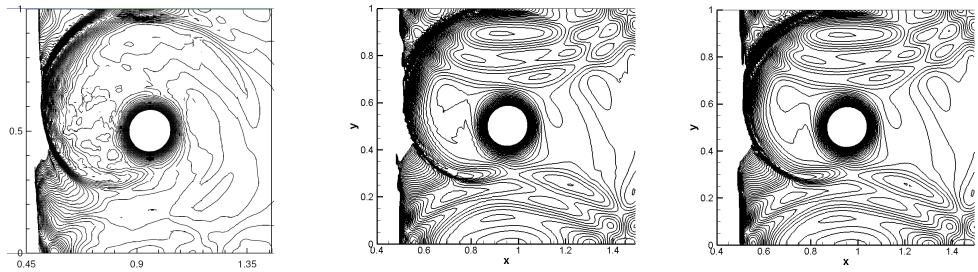
where $\tau=r/r_c$, $r=\sqrt{(x-x_c)^2+(y-y_c)^2}$. Here, ε denotes the strength of the vortex, α is the decay rate of the vortex, r_c is the critical radius for which the vortex has the maximum strength. They are set to be $\varepsilon=0.3$, $\alpha=0.204$, $r_c=0.05$.

The DGM2, the DGM3 and the hybrid DG/FV-1 scheme were employed in the simulation on a mesh of $dh=1/100$, the time step size used is $dt=0.0002$. The grids are uniform in y-direction and clustered properly near the shock in x-direction. The boundary conditions for top and bottom boundaries are set to be slip wall. We have not brought in any limiter operator since the shock is weak. The density contours computed with the present DG/FV-1 scheme at $t=0.2$,0.4, and 0.8 are shown in Fig. 6, respectively. It can be seen that the present simulation can capture the shock waves clearly and sharply. After the interaction between the shock and the vortex, the shape of the vortex is still hold perfectly, and we can see that the reflection boundary take effects at $t=0.8$, when one of the shock bifurcations reaches the top boundary and reflects. Fig. 7 shows the snapshots from DGM3, the hybrid DG/FV-1 scheme and the 3rd order SD by Wang [47] at $t=0.6$. It can be seen that the results by the hybrid DG/FV schemes are very similar to those by DGM3 and by the 3rd order SD method.



(a) $t=0.2s$ (b) $t=0.4s$ (c) $t=0.8s$

Figure 6: Density contours for the 2D shock-vortex interaction by the hybrid DG/FV-1 scheme (20 contours from minimum to maximum).



(a) 3rd SD scheme (from [47]) (b) DGM3 (c) DG/FV-1 scheme

Figure 7: Pressure contours for the 2D shock-vortex interaction (90 contours from 1.19∼1.37).

5.3 Subsonic flow past a circular cylinder

We consider a subsonic flow at Mach number $M_\infty = 0.38$ on four O-type grids having 16×4, 32×8, 64×16, and 128×32 points as shown in Fig. 8. The first number refers to the number of points in the circular direction, and the second describes the number of concentric circles in the mesh. The radius of the cylinder is $r_0 = 0.5$, the domain is bounded by $r_{32} = 20$, the radii of the concentric circles for 128×32 mesh are set up as those in [48].

$$r_j = r_0\Big(1+\frac{2\pi}{128}\sum_{k=0}^{j-1}\alpha^k\Big), \qquad j=1,2,\cdots,32, \tag{5.4}$$

with $\alpha = 1.1648336$. The coarse meshes are obtained by successively coarsening the finest mesh.

The solid wall boundary condition is identical to the reflecting boundary conditions, which state that no flow penetrates a solid wall, i.e., the normal velocity at the wall is zero. Depending on the numerical scheme, a ghost state created on the part of the numerical boundary $\partial\Omega$ corresponding to the solid wall. With the DGM and the hybrid DG/FV-1 scheme, a ghost solution is created at every integration point on $\partial\Omega$, where all components of the ghost solution are set equal to the corresponding interior values at the same point except for normal velocity, which is negated. Then, the interior and ghost

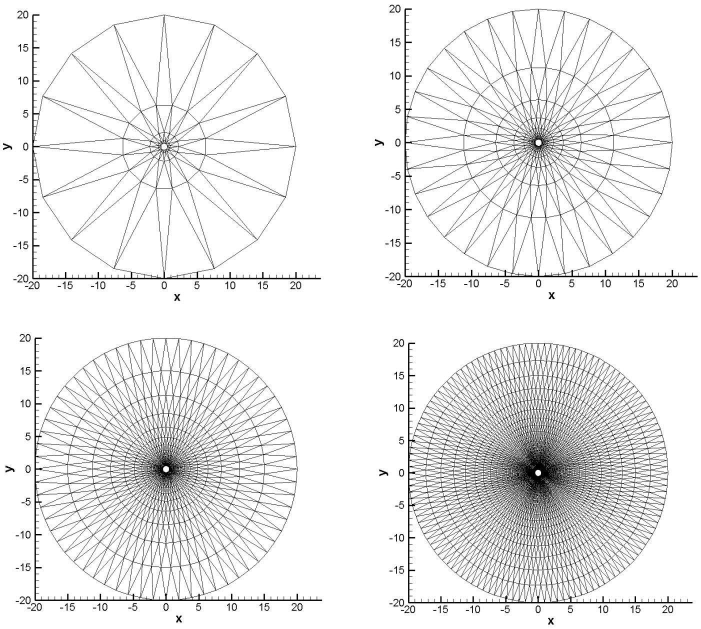


Figure 8: Computational grids (16×4, 32×8, 64×16, and 128×32) over a circular cylinder.

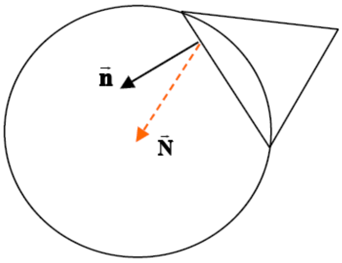


Figure 9: Curved boundary normal and straight boundary normal.

states are passed to a Riemann solver. Due to the symmetry of the reflection, the solution to the Riemann problem at integration point $\mathbf{x}_j \in \partial\Omega$ satisfies

$$\vec{u}(\mathbf{x}_j)\cdot\vec{\mathbf{n}}=0. \tag{5.5}$$

This approach works well for straight-sided bodies. However, results are inferior when a physical geometry is curving. In the present paper, the curved geometry is imposed using a novel approach by Lilia Krivodonova, et al. [48], where the curved elements are not required. Instead, an accurate representation of boundary normals is used in the quadrature points for imposed solid wall boundary conditions for curved geometries. In our implementation, the normal components (N_x,N_y) at the quadrature points are computed using the local true surface normal based on the analytically defined boundary geometries, as illustrated in Fig. 9.

Table 6: Entropy production of DGM2, DGM3, and DG/FV for cylinder flow.

	Grid	L_1 error	Order	L_2 error	Order	L_∞ error	Order
2nd order DGM	16×5	1.6432e-02		1.9588e-02		3.2709e-02	
	32×9	4.0481e-03	2.02	5.3508e-03	1.87	1.0248e-02	1.67
	64×17	7.4889e-04	2.43	1.0564e-03	2.34	2.1888e-03	2.23
	128×133	1.3586e-04	2.46	1.9779e-04	2.42	4.2632e-04	2.37
3rd order DGM	16×5	3.0532e-03		4.6557e-03		9.2237e-03	
	32×9	1.9026e-04	4.01	2.5584e-04	4.18	5.3884e-04	4.10
	64×17	1.5509e-05	3.62	1.8755e-05	3.77	2.7827e-05	4.28
	128×133	1.9011e-06	3.03	2.4119e-06	2.96	5.1777e-06	2.42
DG/FV-1 scheme	16×5	1.0038e-02		1.3306e-02		2.3564e-02	
	32×9	1.5588e-03	2.69	1.7695e-03	2.91	4.5893e-03	2.36
	64×17	2.0041e-04	2.95	2.5438e-04	2.80	7.9249e-04	2.53
	128×133	2.2406e-05	3.16	3.3568e-05	2.92	1.5107e-04	2.39

We compute ghost states at the boundary as follows: The density ρ_g and the pressure p_g are copied exactly from the interior values at the same point. The ghost velocity (u_g,v_g) are computed by exact wall normals.

$$\rho_g=\rho_b, \qquad p_g=p_b, \tag{5.6a}$$

$$u_g=u_b-2N_x(u_bN_x+v_bN_y), \qquad v_g=v_b-2N_y(u_bN_x+v_bN_y). \tag{5.6b}$$

Finally, the Riemann problem $\boldsymbol{H}(\boldsymbol{U}_b,\boldsymbol{U}_g,\vec{\mathbf{n}})$ is solved at solid boundaries.

First, we plot the Mach number contours obtained by the hybrid DG/FV-1 scheme on the 128×32 mesh in Fig. 10 with straight boundary condition and curved geometry boundary. Obviously, the solution obtained by curved solid wall is symmetric and does not have a visible wake, while a fearful, non-physical wake is formed at the rear with straight boundary.

Next, we perform DGM2, DGM3, and the hybrid DG/FV-1 scheme on the 64×17 mesh, and show the Mach number contours and the streamlines near the rear stagnation point in Fig. 11. The symmetry quality of the third-order solutions is much better than that by the 2nd order DGM.

To compare the accuracy of these methods, we measure the errors in entropy production ε on the cylinder. In this case, the following entropy production ε is defined as

$$\varepsilon=\frac{S_{wall}-S_\infty}{S_\infty}=\frac{p}{p_\infty}\Big(\frac{\rho}{\rho_\infty}\Big)^\gamma-1, \tag{5.7}$$

where p_∞ and ρ_∞ are the pressure and the density of the free flow, respectively. Note that the entropy production serves as a good criterion to measure accuracy of the numerical solution, since the flow under consideration is isentropic. Table 6 provides the details of the spatial accuracy of each method for this numerical experiment. The definitions of L_1, L_2 and L_∞ are similar to Eq. (5.2), but ε is the entropy production on the wall, and $nTCell$ is the boundary cell number. We can see that the hybrid DG/FV-1 scheme has

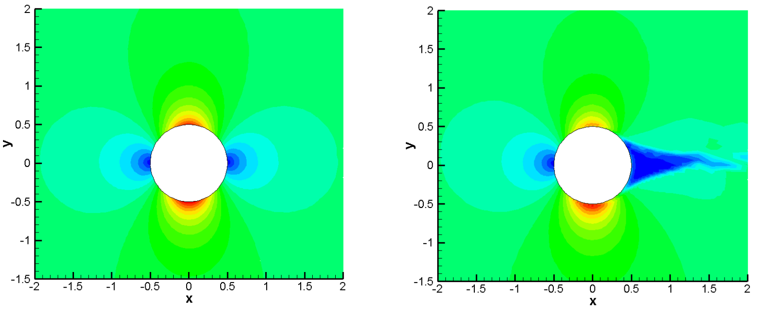


Figure 10: Mach contours (curved and straight boundary condition).

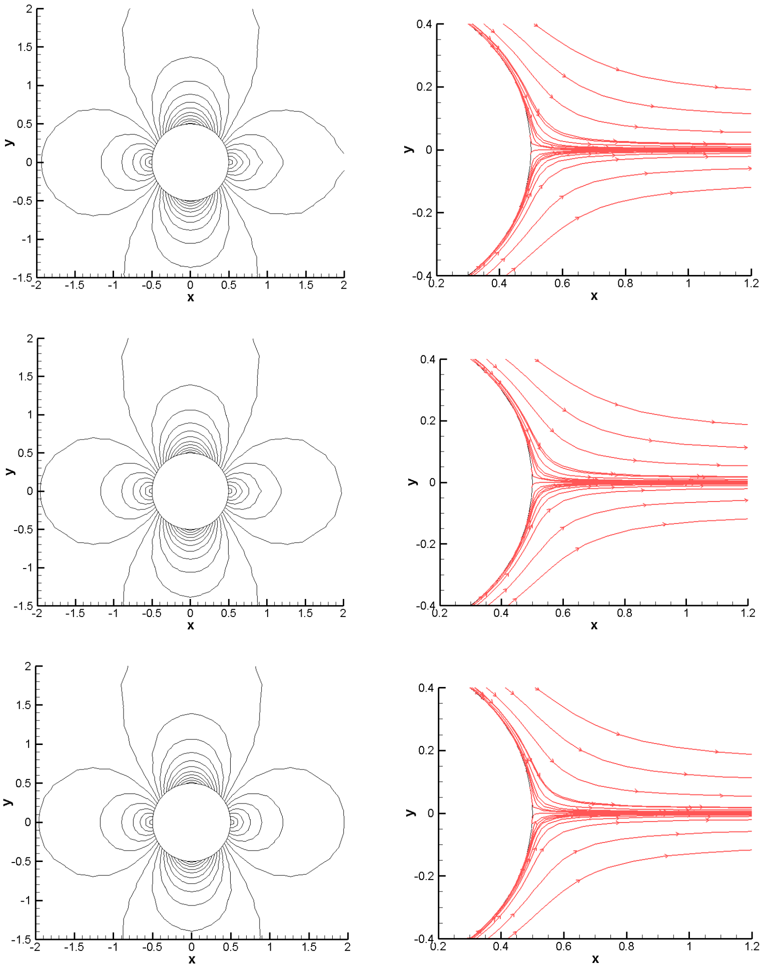


Figure 11: Mach contours (left), and streamlines near surface at the rear stagnation point (right) by DGM2 (top), DG/FV-1 scheme (middle), DGM3 (bottom).

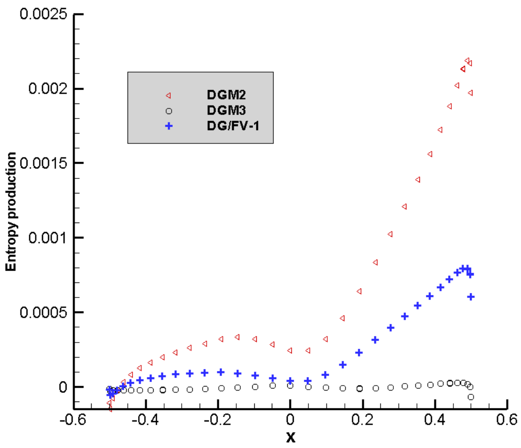


Figure 12: Comparison of computed entropy production on the surface of the cylinder by DGM2, DGM3 and DG/FV-1 scheme on 64×16 mesh.

achieved the desired accuracy. For this case, the solution obtained by DGM2 is much more dissipative, DGM3 shows the best performance, as the absolute error by DGM3 is almost one-order-of-magnitude smaller than the one by DG/FV-1, although they have the same 3$^{\text{rd}}$ accuracy, as illustrated in Fig. 12.

5.4 Transonic flow past a NACA0012 airfoil

A Cartesian/triangular hybrid grid is used in this example (as shown in Fig. 13) to demonstrate the capability of present DG/FV scheme for hybrid grids, which consists of 5004 elements. Two cases are considered. Case 1: Mach number $M=0.8$, and angle of attack $\alpha=1.25°$; Case 2: Mach number $M=0.85$, and angle of attack $\alpha=1.0°$. Barth-Jespersen limiter [49] is imposed to eliminate non-physical oscillation. Fig. 14 shows the computed Mach number contours in the flow field by the hybrid DG/FV-1 scheme. We can see that the existence of a strong shock on the upper surface and a weak shock on the lower surface. The pressure distributions for the two cases are shown in Fig. 15, and compared with the results by DGM2 and DGM3 and by Luo [50]. They agree well with each other.

5.5 Subsonic flow past a three-element airfoil (30P30N)

A subsonic flow past a 3-element airfoil (30P30N) is presented in this test case. A Cartesian/triangular hybrid (as shown in Fig. 16) is used also in this computation in order to demonstrate the capability of the hybrid DG/FV scheme for complex geometry. The mesh consists of 7781 triangles and 4953 Cartesian cells.

The computation is performed using DGM2, DGM3, and the hybrid DG/FV-1 scheme at a Mach number of 0.2, and an angle of attack 19°. Fig. 17 shows the computed pres-

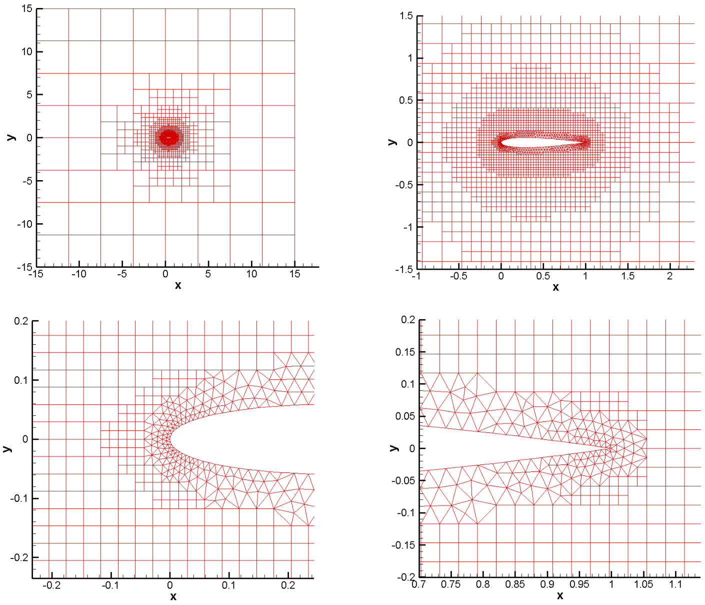


Figure 13: The hybrid unstructured Cartesian and triangle grid for NACA0012 airfoil.

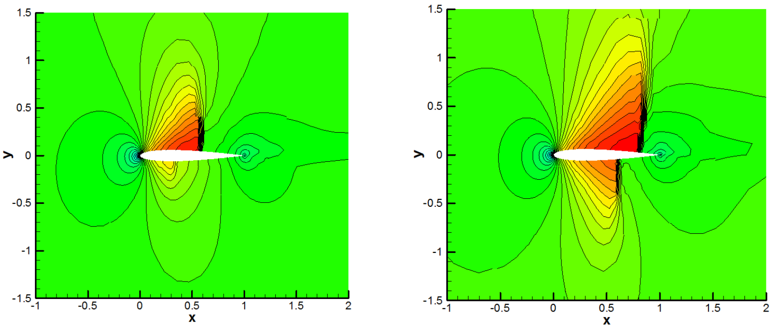


Figure 14: Computed Mach number contours by the DG/FV-1 scheme with Barth limiter for a NACA0012 airfoil. Thirty equally spaced Mach number contours from 0.172 to 1.325. Case 1 (left) and Case 2 (right).

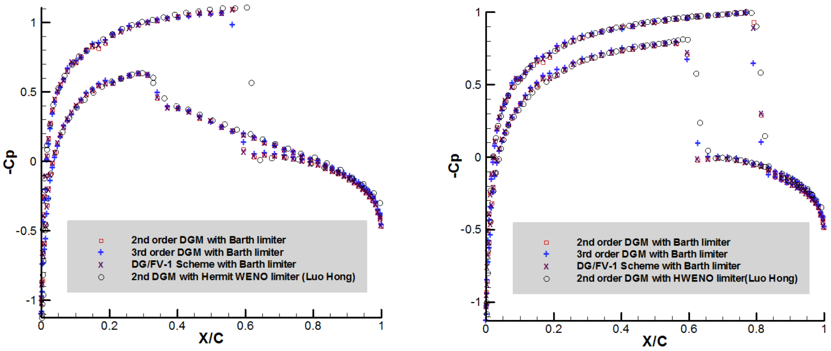


Figure 15: Pressure distribution on the surface of Case 1 (left) and Case 2 (right).

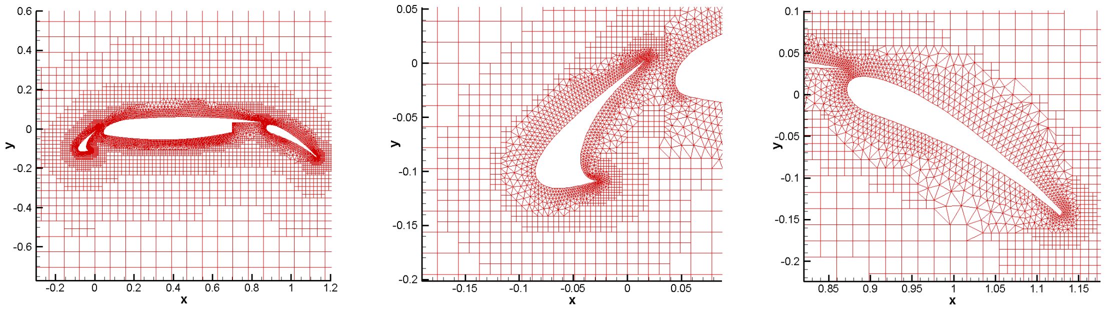


Figure 16: Cartesian/triangular hybrid grid for 3-element airfoil.

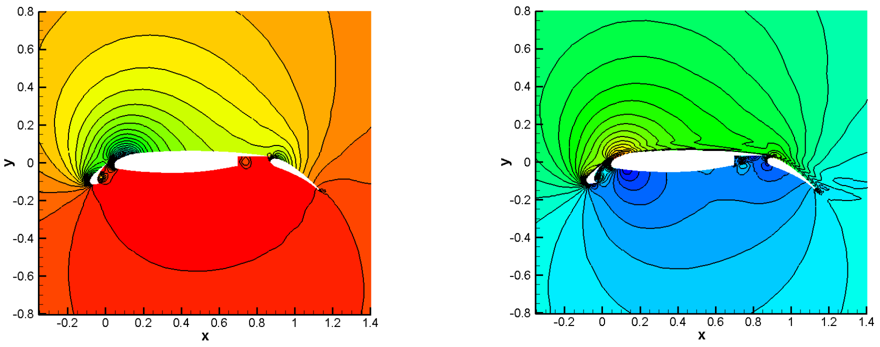


Figure 17: The computed pressure (left) and Mach number (right) contours for subsonic flow past a 3-element airfoil using DG/FV-1 scheme.

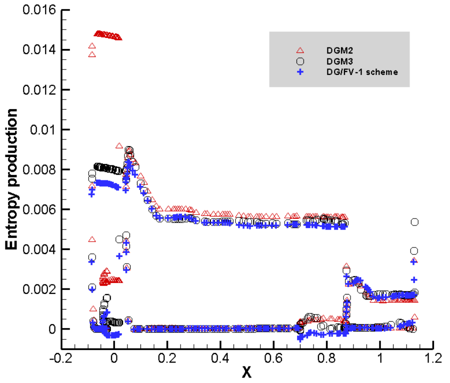


Figure 18: Entropy production on the surface.

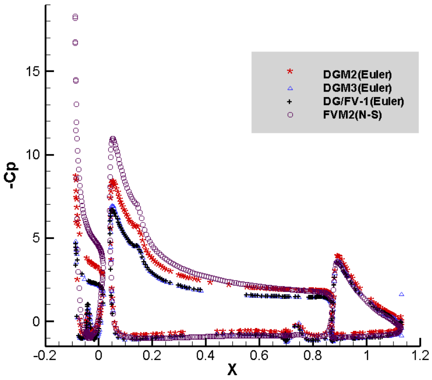


Figure 19: Pressure distribution on the surface.

sure and Mach number contours in the flow field, respectively. The results obtained by DG/FV-1 scheme appear to much better than DGM2, as shown in Fig. 18, where the entropy productions on the surface of the airfoil are compared. The pressure distributions are given in Fig. 19, in which Euler solutions by DGM and DG/FV-1 scheme are compared with N-S solution with a 2^{nd} order FV solver.

6 Conclusions

In this paper, the hybrid DG/FV schemes are extended to solve 2D Euler equations on unstructured and hybrid grids. The numerical results demonstrate that the present hybrid DG/FV scheme reaches the desired third order of accuracy. In addition, it can reduce greatly the CPU time and memory requirement, comparing with the same order DG schemes on the same mesh. It can be expected that it will be more efficient for three-dimensional realistic applications. Because we only consider the typical cases with weak shock when solving the 2D Euler equations, only a simple Barth limiter is adopted to suppress the non-physical oscillations near shock waves in the transonic cases. The applications for compressible flows with strong shock waves will be undertaken in the future work. A proper limiter for higher-order reconstruction should be studied systemically. Furthermore, the extension to Navier-Stokes equations will be undertaken also in the near future, the implicit method and the well-known *hp*-multigrid approach should be considered to accelerate the convergence history. For realistic three-dimensional problems, the parallel computation should be carried out firstly.

Acknowledgments

This work is supported partially by National Basic Research Program of China (Grant No. 2009CB723800) and by National Science Foundation of China (Grant No. 91016011, 91130029 and 11028205). The authors would like to appreciate Prof. Z. J. Wang in Iowa State University, Prof. H. Luo in North Carolina State University, and Prof. S. H. Zhang in State Key Laboratory of Aerodynamics (China Aerodynamics Research and Development Center) for the helpful discussion and suggestion.

References

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

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

[3] W. H. Reed and T. R. Hill, Triangular mesh methods for the neutron transport equation, Technical Report LA-UR-73-479, Los Alamos Scientific Laboratory, 1973.

[4] B. Cockburn and C. W. Shu, TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws II: general framework, Math. Comput., 52 (1989), 411–435.

[5] B. Cockburn, S. Y. Lin and C. W. Shu, TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws III: one-dimensional systems, J. Comput. Phys., 84 (1989), 90–113.

[6] B. Cockburn, S. Hou and C. W. Shu, TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws IV: the multidimensional case, Math. Comput., 54 (1990), 545–581.

[7] B. Cockburn and C. W. Shu, Runge-Kutta discontinuous Galerkin methods for convection-dominated problems, J. Sci. Comput., (2001), 173–261.

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

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

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

[11] M. Dumbser and M. Kaser, Arbitrary high order non-oscillatory finite volume schemes on unstructured meshes for linear hyperbolic systems, J. Comput. Phys., 221 (2007), 693–723.

[12] M. Dumbser, M. Kaser, V. Titarev and E. Toro, Quadrature-free non-oscillatory finite volume schemes on unstructured meshes for nonlinear hyperbolic systems, J. Comput. Phys., 226 (2007), 204–243.

[13] Y. Zhang and C. W. Shu, Third order WENO scheme on three dimensional tetrahedral meshes, Commun. Comput. Phys., 5 (2009), 836–848.

[14] V. A. Titarev, P. Tsoutsanis and D. Drikakis, WENO schemes for mixed-element unstructured meshes, Commun. Comput. Phys., 8(3) (2010), 585–609.

[15] P. Tsoutsanis, V. Titarev and D. Drikakis, WENO schemes on arbitrary mixed-element unstructured meshes in three space dimensions, J. Comput. Phys., 230 (2011), 1585–1601.

[16] Z. J. Wang, Spectral (finite) volume method for conservation laws on unstructured grids: basic formulation, J. Comput. Phys., 178 (2002), 210–251.

[17] Z. J. Wang and Y. Liu, Spectral (finite) volume method for conservation laws on unstructured grids ii: extension to two-dimensional scalar equation, J. Comput. Phys., 179 (2002), 665–697.

[18] Z. J. Wang and Y. Liu, Spectral (finite) volume method for conservation laws on unstructured grids III: one-dimensional systems and partition optimization, J. Sci. Comput., 20 (2004), 137–157.

[19] Z. J. Wang, L. P. Zhang and Y. Liu, Spectral (finite) volume method for conservation laws on unstructured grids IV: extension to two-dimensional Euler equations, J. Comput. Phys., 194 (2004), 716–741.

[20] Y. Liu, M. Vinokur and Z. J. Wang, Discontinuous spectral difference method for conservation laws on unstructured grids, J. Comput. Phys., 216 (2006), 780–801.

[21] V. Venkatakrishnan, S. R. Allmaras, D. S. Kamenetskii and F. T. Johnson, Higher order schemes for the compressible Navier-Stokes equations, AIAA-2003-3987, 2003.

[22] G. May and A. Jameson, A spectral difference method for the Euler and Navier-Stokes equations, AIAA-2006-304, 2006.

[23] J. A. Ekaterinaris, High-order accurate, low numerical diffusion methods for aerodynamics, Prog. Aero. Sci., 41 (2005), 192–300.

[24] Z. J. Wang, High-order methods for the Euler and Navier-Stokes equations on unstructured grids, Prog. Aero. Sci., 43 (2007), 1–41.

[25] T. J. Baker, Mesh generation: art or science?, Prog. Aero. Sci., 41 (2005), 29–63.

[26] B. Cockburn, G. E. Karniadakis and C. W. Shu, Discontinuous Galerkin Methods, Berlin, Springer, 2000.

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

[28] T. Barth and P. Frederickson, High order solution of the Euler equations on unstructured grids using quadratic reconstruction, AIAA Paper 90-0013, 1990.

[29] R. R. Thareja and J. R. Stewart, A point implicit unstructured grid solver for the Euler and

Navier-Stokes equations, Int. J. Num. Meth. Fluids, 9 (1989), 405–425.

[30] H. Luo, J. D. Baum and R. Lohner, High-Reynolds number viscous computations using an unstructured-grid method, J. Aircraft, 42 (2005), 483–492.

[31] L. X. He, L. P. Zhang and H. X. Zhang, A finite element/finite volume mixed solver on hybrid grids, Proceedings of the Fourth International Conference on Computational Fluid Dynamics, 10-14 July, 2006, Ghent, Belgium, edited by Herman Deconinck and Erik Dick, Springer Press, (2006), 695–700.

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

[33] M. Dumbser, Arbitrary high order PNPM schemes on unstructured meshes for the compressible Navier-Stokes equations, Comput. Fluids, 39 (2010), 60–76.

[34] M. Dumbser and O. Zanotti, Very high order PNPM schemes on unstructured meshes for the resistive relativistic MHD equations, J. Comput. Phys., 228 (2009), 6991–7006.

[35] J. X. Qiu and C. W. Shu, Hermite WENO schemes and their application as limiters for Runge-Kutta discontinuous Galerkin method: one-dimensional case, J. Comput. Phys., 193 (2003), 115–135.

[36] J. Qiu and C. W. Shu, Hermite WENO schemes and their application as limiters for Runge-Kutta discontinuous Galerkin method II: two-dimensional case, Comput. Fluids, 34 (2005), 642–663.

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

[38] H. Luo, L. P. Luo, A. Ali, R. Norgaliev and C. Cai, A parallel, reconstructed discontinuous Galerkin method for the compressible flows on arbitrary grids, Commun. Comput. Phys., 9(2) (2011), 363–389.

[39] L. P. Zhang, W. Liu, L. X. He and X. G. Deng, A new class of DG/FV hybrid schemes for one-dimensional conservation law, the 8th Asian Conference on Computational Fluid Dynamics, Hong Kong, 10-14 January, 2010.

[40] L. P. Zhang, W. Liu, L. X. He, X. G. Deng and H. X. Zhang, A class of hybrid DG/FV methods for conservation laws I: basic formulation and one-dimensional systems, J. Comput. Phys., 2011, in press.

[41] L. P. Zhang, W. Liu, L. X. He, X. G. Deng and H. X. Zhang, A class of hybrid DG/FV method for conservation laws II: two-dimensional cases, J. Comput. Phys., 2011, in press.

[42] H. T. Huynh, A flux reconstruction approach to high-order schemes including discontinuous Galerkin methods, AIAA-2007-4079, 2007.

[43] Z. J. Wang and H. Gao, A unifying lifting collocation penalty formulation for the Euler equations on mixed grids, AIAA-2009-0401, 2009.

[44] Y. J. Liu, C. W. Shu, E. Tadmor and M. P. Zhang, Central discontinuous Galerkin methods on overlapping cells with a non-oscillatory hierarchical reconstruction, SIAM J. Numer. Anal., 45 (2007), 2442–2467.

[45] H. Luo, J. D. Baum and R. Lhner, A p-multigrid discontinuous Galerkin method for the Euler equations on unstructured grids, J. Comput. Phys., 211 (2006), 767–783.

[46] C. R. Nastase and D. J. Mavriplis, High-order discontinuous Galerkin methods using an hp-multigrid approach, J. Comput. Phys., 213 (2006), 330–357.

[47] M. Yang and Z. J. Wang, A parameter-free generalized moment limiter for high-order methods on unstructured grids, AIAA-2009-605, 2009.

[48] L. Krivodonova and M. Berger, High-order accurate implementation of solid wall boundary condition in curved geometries, J. Comput. Phys., 211 (2006), 492–512.

[49] T. J. Barth and D. C. Jesperson, The design of application of upwind schemes on unstructured grids, AIAA-1989-0366, 1989.

[50] H. Luo, J. D. Baum and R. Lohner, A Hermite WENO-based limiter for discontinuous Galerkin method on unstructured grids, J. Comput. Phys., 225 (2007), 686–713.