

# A collocated-grid spectral difference method for compressible flows

Wenqian Chen<sup>a</sup>, Yaping Ju<sup>a</sup>, Chuhua Zhang<sup>a,\*</sup>

<sup>a</sup>*Department of Fluid Machinery and Engineering, School of Energy and Power Engineering,  
Xi'an Jiaotong University, Xi'an, Shaanxi, People's Republic of China*

---

## Abstract

Among various high-order numerical methods for compressible flows, spectral difference method is very promising and attractive due to its simple formulation, easy implementation and high efficiency, but most of the so far developed spectral difference methods are based on staggered grid and thus suffer from extra interpolation operations between solution points and flux points. In order to further improve computational efficiency, we propose a collocated-grid spectral difference method on unstructured hexahedral grids by removing the interior flux points. The proposed method holds all the advantages of spectral difference methods while involves the solution points and only the interface flux points at each element. In the proposed method, the spatial terms are calculated by a weight average of two proposed schemes: a spectral difference-like scheme and a discontinuous Galerkin spectral element-like scheme. The collocated-grid spectral difference method is firstly analysed with Fourier technique in terms of dissipation, dispersion and stability for the one-dimensional unsteady advection equation. Then, the proposed method is further

---

\*Corresponding author

Email addresses: [wenqianchen2016@gmail.com](mailto:wenqianchen2016@gmail.com) (Wenqian Chen), [yapingju@xjtu.edu.cn](mailto:yapingju@xjtu.edu.cn) (Yaping Ju)

URL: [chzhang@mail.xjtu.edu.cn](mailto:chzhang@mail.xjtu.edu.cn) (Chuhua Zhang)

verified with analytic solutions of Euler vortex propagation problem. The numerical accuracy of order  $P + 1$  can be hold for the reconstruction polynomial of order  $P$  and the best weights are obtained for the enhanced computational efficiency. Finally, the capability of the proposed method in handing curved boundary and turbulent flow is demonstrated with two typical problems: flow over NACA0012 airfoil and decay of Taylor Green vortex. The proposed method bridges the spectral difference and discontinuous Galerkin spectral element method in the framework of differential formulation, and promises a higher computational efficiency over them.

*Keywords:* Spectral difference, Discontinuous Galerkin, Fourier analysis, Unstructured hexahedral grid

---

## 1. Introduction

In recent years, high-order numerical methods have been attracting more and more attentions in computational fluid dynamics (CFD). Compared against low-order numerical methods, high-order numerical methods are capable of achieving higher accuracy with lower computational cost for many theoretical and practical problems, such as turbulence, acoustic generation and acoustic propagation [1]. Among various high-order numerical methods, spectral difference (SD) method is very promising due to its simple formulation, easy implementation and high efficiency. However, most of the so far developed SD methods are designed in the framework of staggered grid and thus suffer from massive interpolation operations between solution points and flux points. A more efficient SD method relieved from the interpolation operations and even the flux points will be of great significance for the development of high-order CFD methods and of potential application value for the practical problems with complex geometry.

The precursor of SD method, namely the conservative staggered-grid Chebyshev

multidomain method was proposed by Kopriva [2, 3] in 1996 to remedy the severe restrictions of Chebyshev spectral methods, such as non-conservation and poor geometric flexibility [4]. Later, Liu, Wang and their collaborators detailed a general formulation of SD method for unstructured grids [5], and then extended it to numerical solutions of Euler equations [6] and Naiver-Stokes equations [7]. The SD method possesses many desired features such as conservation, geometric flexibility, programming simplicity and scalable parallelization. More recently, the capability of SD method was assessed with several problems such as turbulence [8–10], aeroacoustics [11] and fluid-structure interaction [12, 13]. As a collocation point-based method, the above SD methods deployed two sets of points in each element, namely solution points and flux points, to store the solutions  $Q$  and fluxes  $\mathbf{F}$ , respectively. To update the solutions, the solutions were first interpolated from solution points to flux points; next, the fluxes at flux points were calculated with the solutions; then, the flux derivatives were interpolated from flux points to solution points; finally, the solutions were updated with the flux derivatives. Therefore, the above SD methods were essentially based on the staggered grid and thus involved massive interpolation operations between the solution points and the flux points, reducing considerably the computational efficiency.

The improvements with respect to mathematic formulation, computational accuracy and efficiency of SD method were intensively researched in last decade. Van den Abeele [14] utilized Fourier technique to analyse the mathematical properties of SD method. It was found that the accuracy as well as stability of SD method are determined by the locations of flux points rather than the locations of solution points. Based on this finding, Van den Abeele [14] designed a series of efficient SD schemes by setting solution points as a subset of flux points. Huynh [15] proposed a flux reconstruction (FR) method in a similar way with SD method. In FR method,

a flux correction function was designed to guarantee the continuity of flux at element interfaces. The flux correction function will approach zero in the case of continuity of flux at the interface. Given a proper definition of correction function, FR method can recover a more efficient version of SD method where the interior flux points are eliminated. Wang and Gao [16] extended FR method to general grids with a lifting collocation penalty (LCP) formulation. The LCP and FR methods can converge to the same discrete form, therefore Huynh and Wang renamed them as "correction procedure via reconstruction" (CPR) method. Jameson [17] proved the use of Legendre-Gauss points as interior flux points will always produce stable SD scheme for all order of accuracy for one-dimensional linear advection equation. May [18] proved that the quadrature-free discontinuous Galerkin method converges to SD method under three conditions: Lagrange basis, quadrature-free formulation and the use of the numerical flux in the volume integral. Liang et al. [19] conducted a comparison of computational efficiency between SD with CPR methods. Their results showed that CPR method is about 27% and 42% faster than SD method for inviscid and viscous flows, respectively. Yu et al. [20] analysed and compared the numerical accuracy and efficiency of several high-order methods including quadrature-based discontinuous Galerkin (QDG), nodal discontinuous Galerkin (NDG), SD and CPR. They stated that SD method is still a very promising efficient method in spite of its larger absolute error than CPR and QDG methods.

From the above reviews, SD method is a very promising high-order numerical method, but is essentially based on the staggered grid, which involves massive interpolation operations and hampers further improvement in computational efficiency. In this work, we propose a collocated-grid spectral difference (CGSD) method on unstructured hexahedral grids by removing the interior flux points. In the proposed method, only solution points and interface flux points are deployed, and thus the in-

terpolation operations are avoided between solution points and interior flux points, benefitting the improvement in efficiency. To compute spatial terms without the deployment of interior flux points, three schemes are proposed: a spectral difference-like scheme, a discontinuous Galerkin spectral element-like scheme and their hybrid. The mathematical and numerical properties of the CGSD method are verified with Fourier technique and the available analytic solutions. Its capability in handing curved boundary and turbulent flow is validated with two typical problems: flow over NACA0012 airfoil and decay of Taylor Green vortex. The CGSD method can provide a potential and feasible high-order numerical tool for large eddy simulation and even direct numerical simulation of turbulent flows in practical problems such as turbomachinery aerodynamics.

The remainder of the paper is organised as follows. In section 2, the mathematical models are described, involving governing equations and element mapping. Section 3 presents the formulation of the CGSD method. In section 4, the dissipation, dispersion and stability of the CGSD method are analysed with Fourier technique. Then its accuracy and efficiency are verified with the analytic solutions in section 5. In section 6, the CGSD method is further demonstrated with two typical flow problems. Finally, some conclusions are drawn in section 7.

## 2. Mathematical models

### 2.1. Governing equations

The governing equations for three-dimensional unsteady viscous compressible flows are written as Naiver-Stokes equations in conservative formulation in Cartesian coordinate system:

$$\frac{\partial Q}{\partial t} + \nabla_{\mathbf{x}} \cdot \mathbf{F} = 0 \quad (1)$$

where  $Q$  is the unknown vector of conserved variable,  $t$  the time, the gradient operator  $\nabla_{\mathbf{x}}$  in the physical space  $\mathbf{x} = (x_1, x_2, x_3)^T$ ,  $\mathbf{F} = \mathbf{F}^I(Q) - \mathbf{F}^V(Q, \nabla_{\mathbf{x}} Q)$  the flux vector composed of inviscid flux  $\mathbf{F}^I = (F_1^I, F_2^I, F_3^I)^T$  and viscous flux  $\mathbf{F}^V = (F_1^V, F_2^V, F_3^V)^T$ . They are given by

$$Q = \begin{pmatrix} \rho \\ \rho u_1 \\ \rho u_2 \\ \rho u_3 \\ \rho e \end{pmatrix} \quad F_l^I = \begin{pmatrix} \rho u_l \\ \rho u_l u_1 + \delta_{l1} p \\ \rho u_l u_2 + \delta_{l2} p \\ \rho u_l u_3 + \delta_{l3} p \\ u_l (\rho e + p) \end{pmatrix} \quad F_l^V = \begin{pmatrix} 0 \\ \tau_{l1} \\ \tau_{l2} \\ \tau_{l3} \\ \tau_{lj} u_j - q_l \end{pmatrix} \quad (2)$$

where  $l = 1, 2, 3$  is the free index,  $j = 1, 2, 3$  the summation index,  $\mathbf{u} = (u_1, u_2, u_3)^T$  the flow velocity in Cartesian coordinate,  $\rho$ ,  $p$  and  $e$  the density, pressure and specific total energy. The stress tensor  $\boldsymbol{\tau}$  and the heat flux  $\mathbf{q}$  can be calculated by

$$\boldsymbol{\tau} = \mu \left( \nabla_{\mathbf{x}} \mathbf{u} + (\nabla_{\mathbf{x}} \mathbf{u})^T - \frac{2}{3} (\nabla_{\mathbf{x}} \cdot \mathbf{u}) \mathbf{E} \right), \quad \mathbf{q} = -k \nabla_{\mathbf{x}} T \quad (3)$$

where  $\mu$  is the viscosity coefficient,  $k$  the thermal conductivity,  $T$  the temperature,  $\mathbf{E}$  the unit tensor of the second order. The governing equations (1) are closed with the equation of state for a perfect gas

$$p = (\kappa - 1) \rho \left( e - \frac{1}{2} \mathbf{u} \cdot \mathbf{u} \right) \quad (4)$$

where  $\kappa = c_p/c_v$  is the specific heat ratio,  $c_p$  and  $c_v$  the specific heat at constant pressure and constant density, respectively.

## 2.2. Element mapping

In order to tackle curved boundaries, the patched unstructured curved hexahedral elements are employed to discretize the solution domain. Considering the irregularity of physical elements, they are transformed from physical space  $\mathbf{x} = (x_1, x_2, x_3)^T$

to computational space  $\boldsymbol{\xi} = (\xi^1, \xi^2, \xi^3)^T \in [-1, 1]^3$ , as shown in Fig. 1. The corresponding mapping is written as

$$\mathbf{x} = \sum_{n=1}^{N_e} M_n(\boldsymbol{\xi}) \mathbf{x}_n \quad (5)$$

where  $N_e$  is the number of points defining a physical element,  $\mathbf{x}_n$  the corresponding coordinates of the  $n$ th point,  $M_n(\boldsymbol{\xi})$  the shape function with respect to the  $n$ th point.  $N_e$  is dependent on the curved element order  $P_e$ , and it follows  $N_e = (P_e + 1)^3$ . The non-singular Jacobian matrix of the mapping has the following form

$$\mathbf{J} = \frac{\partial \mathbf{x}}{\partial \boldsymbol{\xi}} = \left( \frac{\partial \mathbf{x}}{\partial \xi^1}, \frac{\partial \mathbf{x}}{\partial \xi^2}, \frac{\partial \mathbf{x}}{\partial \xi^3} \right) := (\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3) \quad (6)$$

Thus, the divergence in physical space can be transformed from computational space by

$$\nabla_{\mathbf{x}} \cdot \mathbf{F} = \left( (\mathbf{J}^T)^{-1} \nabla_{\boldsymbol{\xi}} \right) \cdot \mathbf{F} \quad (7)$$

Following Kopriva [21], this non-conservative divergence at the right hand side of Eq. (7) can be written as a conservative one

$$\nabla_{\mathbf{x}} \cdot \mathbf{F} = \frac{1}{J} \nabla_{\boldsymbol{\xi}} \cdot \tilde{\mathbf{F}} \quad (8)$$

where  $J = \mathbf{e}_1 \cdot (\mathbf{e}_2 \times \mathbf{e}_3)$  is the determinant of Jacobian matrix  $\mathbf{J}$ ,  $\tilde{\mathbf{F}} = (\tilde{F}^1, \tilde{F}^2, \tilde{F}^3)^T$  the contravariant fluxes in computational space defined by

$$\tilde{F}^l = (J \mathbf{e})^l \cdot \mathbf{F}, \quad l = 1, 2, 3 \quad (9)$$

with the metric terms specified by

$$(J \mathbf{e})^l = \mathbf{e}_k \times \mathbf{e}_m, \quad (l, k, m) \text{ cyclic} \quad (10)$$

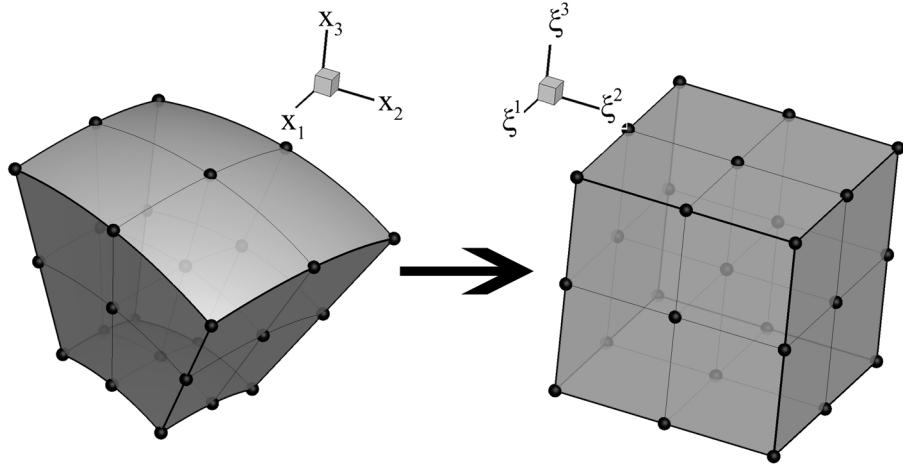


Figure 1: The transformation from (left) a physical element to (right) the computational element.

Following the idea of Kopriva [21], the evaluation of the metric terms are derived in invariant curl formulation to satisfy so-called free-stream preservation property:

$$\sum_{l=1}^3 \frac{\partial (Je)_d^l}{\partial \xi^l} = 0, \quad d = 1, 2, 3 \quad (11)$$

Substitution of Eqs. (8)-(10) into Eq. (1) leads to the following governing equations in computational space

$$\frac{\partial \tilde{Q}}{\partial t} + \nabla_{\xi} \cdot \tilde{\mathbf{F}} = 0 \quad (12)$$

where  $\tilde{Q} = JQ$  is the vector of conserved variables in computational space.

### 3. The collocated-grid spectral difference method (CGSD)

In the spectral difference (SD) method with staggered grid configuration [2, 3, 5–7], two nodal sets including solution points and flux points are defined for reconstruction of solution and flux, respectively. The spatial terms of governing equation (12) are computed mainly by three steps: interpolation of solutions from solution points to flux points, the calculation of fluxes at flux points, differentiation of fluxes

from flux points to solution points. According to the formulation of the SD method, the calculations of fluxes and the differentiation of fluxes are nearly unavoidable, while the interpolation computations can be reduced substantially by removing interior flux points. Based on this observation, we propose a collocated-grid spectral difference (CGSD) method, where only solution points and interface flux points are collocated in each element. Owing to this revision, the CGSD method will be as economic as the CPR method in terms of calculating spatial terms, and the CPR is about 27% and 42% faster than SD for two-dimensional inviscid and viscous compressible flows, respectively [19]. Fig. 2 illustrates the locations of the solution points and flux points in two-dimensional fourth order computational element for the CGSD method. Taking  $(P + 1)$ th order CGSD method as example, the implementation process is introduced in the following.

The formulation of the CGSD method, similar to that of the SD method, is essentially one dimensional by design. In each coordinate direction of computational element, the solution unknowns are conserved variables and discretized on  $P + 1$  solution points. The locations of these points are the roots of  $(P + 1)$ th Legendre polynomial  $\xi_n$ ,  $n = 0, 1, \dots, P$  and they are all located in the interior of element. Using the solutions at  $P + 1$  solution points, a  $P$ th order polynomial for solution is reconstructed with Lagrange basis

$$l_i(\xi) = \prod_{n=0, n \neq i}^P \frac{(\xi - \xi_n)}{(\xi_i - \xi_n)}, \quad 0 \leq i \leq P \quad (13)$$

The reconstruction of solution in computational element is simply the tensor product of three one-dimensional polynomials

$$Q(\xi) = \sum_{k=0}^P \sum_{j=0}^P \sum_{i=0}^P Q_{i,j,k} l_i(\xi^1) l_j(\xi^2) l_k(\xi^3) \quad (14)$$

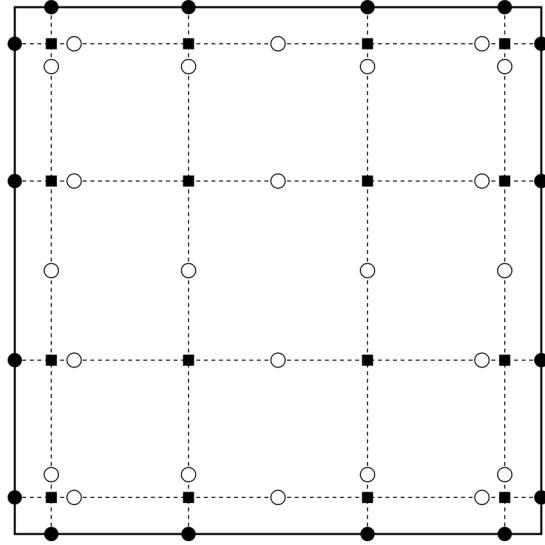


Figure 2: The locations of solution points (solid squares), interface points (solid circles) and interior ghost points (circles) in the fourth order element for the collocated-grid spectral difference method.

In order to discretize Eq. (12), flux derivatives are needed to calculate at solution points. In each direction, only fluxes at  $P + 1$  solution points are not adequate to reconstruct a suitable flux polynomial for the calculation of flux derivatives. The reasons are twofold: first,  $P+1$  points will at most produce a  $(P-1)$ th order polynomial for flux derivative, which is one order lower than solution polynomial; second, as solution points are all in the interior of element, the resultant flux derivatives cannot feel the impact of other elements or boundary. To remedy this problem, two interface flux points, i.e., two end points  $\pm 1$  in each coordinate direction, have to be included in the calculation of flux derivatives. Therefore, we have  $P + 3$  fluxes at  $P + 1$  solution points and two interface flux points. However, it remains a problem how to compute the flux derivatives at solution points, which will be discussed in details in the next subsections 3.1-3.3. Here, we use the differential matrix  $\mathbf{D}$  of

size  $(P + 1) \times (P + 3)$  to represent the operator of flux derivatives. For the sake of brevity, the two interface points in each coordinate direction are assigned to indexes  $-1$  and  $P + 1$ , respectively. The flux derivatives at solution points are calculated as

$$\left\{ \begin{array}{l} \frac{\partial \tilde{F}^1}{\partial \xi^1} \Big|_{i,j,k} = \sum_{n=-1}^{P+1} \tilde{F}_{n,j,k}^1 D_{i,n} \\ \frac{\partial \tilde{F}^2}{\partial \xi^2} \Big|_{i,j,k} = \sum_{n=-1}^{P+1} \tilde{F}_{i,n,k}^2 D_{j,n} \\ \frac{\partial \tilde{F}^3}{\partial \xi^3} \Big|_{i,j,k} = \sum_{n=-1}^{P+1} \tilde{F}_{i,j,n}^3 D_{k,n} \end{array} \right. \quad 0 \leq i, j, k \leq P \quad (15)$$

where  $D$  is an entry in the matrix  $\mathbf{D}$ , and its two subscripts indicate row number and column number, respectively. To compute flux derivatives at the two interface flux points  $\pm 1$ , we extrapolate conserved variables from solution points to interface flux points:

$$\left\{ \begin{array}{l} Q(\pm 1, \xi_j^2, \xi_k^3) = \sum_{n=0}^P Q_{n,j,k} l_n(\pm 1) \\ Q(\xi_i^1, \pm 1, \xi_k^3) = \sum_{n=0}^P Q_{i,n,k} l_n(\pm 1) \\ Q(\xi_i^1, \xi_j^2, \pm 1) = \sum_{n=0}^P Q_{i,j,n} l_n(\pm 1) \end{array} \right. \quad 0 \leq i, j, k \leq P \quad (16)$$

Due to the independent reconstruction in each element, solution is continuous in the interior of each element while discontinuous at the element interfaces, and so for flux. To guarantee the conservation and stability, a continuous flux is required to construct with discontinuous conserved variables ( $Q^L$  and  $Q^R$ ) at element interfaces. To this end, the Rusanov solver as a kind of Riemann solver is employed to compute inviscid numerical fluxes  $\tilde{\mathbf{F}}^I(Q^L, Q^R, \mathbf{n})$  where  $\mathbf{n}$  is the unit normal of element interface.

As for viscous flux, it is a function of conserved variable and its gradient, and thus viscous flux derivative contains second-order derivative term. However, as described above, the CGSD method are designed to use local reconstruction polynomials to discretize the first-order derivative terms in the interior of each element. Direct discretization of the second-order derivative terms will result in the problem of instability, which was addressed in Bassi and Rebay [22]. We here adopt the average approach [7] for the evaluation of viscous fluxes. The basic idea behind the approach is to compute gradients of conserved variables using additional auxiliary equations

$$(\nabla_{\mathbf{x}} Q)^d = \frac{\partial Q}{\partial x_d} \quad d = 1, 2, 3 \quad (17)$$

where the right hand sides of Eq. (17) are transformed from physical space to computational space

$$(\nabla_{\mathbf{x}} Q)^d = \frac{1}{J} \sum_{l=1}^3 \frac{\partial}{\partial \xi^l} \left( (Je)_d^l Q \right), \quad d = 1, 2, 3 \quad (18)$$

$$(\nabla_{\mathbf{x}} Q)^d = \frac{1}{J} \sum_{l=1}^3 \frac{\partial}{\partial \xi^l} (U^{d,l}), \quad U^{d,l} = (Je)_d^l Q \quad (19)$$

The gradients at solution points are calculated in the similar way with the calculation of flux derivatives in Eq. (15)

$$(\nabla_{\mathbf{x}} Q)_{i,j,k}^d = \frac{1}{J_{i,j,k}} \left[ \sum_{n=-1}^{P+1} U_{n,j,k}^{d,1} D_{i,n} + \sum_{n=-1}^{P+1} U_{i,n,k}^{d,2} D_{j,n} + \sum_{n=-1}^{P+1} U_{i,j,n}^{d,3} D_{k,n} \right] \quad (20)$$

As the conserved variable is discontinuous at interface flux point, the conserved variable is simply calculated as an average of the left and right solutions.

$$\hat{Q} = \frac{Q^L + Q^R}{2} \quad (21)$$

After the calculation of the gradients of conserved variables at solution points, the gradients are then extrapolated from solution points to interface flux points. Again, the gradient at interfaces is discontinuous, and the numerical viscous fluxes are

calculated with the average solution and the average gradient:

$$\tilde{\mathbf{F}}^V = \tilde{\mathbf{F}}^V \left( \frac{Q^L + Q^R}{2}, \frac{\nabla_{\mathbf{x}} Q^L + \nabla_{\mathbf{x}} Q^R}{2} \right) \quad (22)$$

As all the fluxes have been computed at solution points and interface flux points, flux derivatives at solution points are evaluated with Eq. (15) and thus we have the semi-discretized equations:

$$\frac{\partial \tilde{Q}_{i,j,k}}{\partial t} + \sum_{n=-1}^{P+1} \tilde{F}_{n,j,k}^1 D_{i,n} + \sum_{n=-1}^{P+1} \tilde{F}_{i,n,k}^2 D_{j,n} + \sum_{n=-1}^{P+1} \tilde{F}_{i,j,n}^3 D_{k,n} = 0 \quad (23)$$

or

$$\begin{aligned} \frac{\partial Q_{i,j,k}}{\partial t} &= -R_{i,j,k} \\ R_{i,j,k} &= \frac{1}{J_{i,j,k}} \left( \sum_{n=-1}^{P+1} \tilde{F}_{n,j,k}^1 D_{i,n} + \sum_{n=-1}^{P+1} \tilde{F}_{i,n,k}^2 D_{j,n} + \sum_{n=-1}^{P+1} \tilde{F}_{i,j,n}^3 D_{k,n} \right) \end{aligned} \quad (24)$$

In summary, the CGSD method is implemented with the following five steps:

- (1) Given conserved variables  $Q$  at solution points, extrapolate the conserved variables from solution points to interface flux points with Eq. (16).
- (2) Calculate the gradients of conserved variables at solution points with Eq. (20) and Eq. (21).
- (3) Extrapolate the gradients of conserved variables from solution points to interface flux points.
- (4) Calculate the three-dimensional fluxes  $\tilde{\mathbf{F}}$  at solution points, and the inviscid/viscous numerical fluxes at interface flux points.
- (5) March the semi-discretized Eq. (24) in time with third-order three-stage Runge-Kutta method [23], which is formulated as follows (the subscripts  $i$ ,  $j$  and  $k$  are omitted for brevity):

$$\begin{aligned}
G &= R(Q^n, t_n) \\
Q^{n+1/3} &= Q^n + \frac{1}{3}\Delta t G \\
G &= -\frac{5}{9}G + R\left(Q^{n+1/3}, t_n + \frac{1}{3}\Delta t\right) \\
Q^{n+2/3} &= Q^{n+1/3} + \frac{15}{16}\Delta t G \\
G &= -\frac{153}{128}G + R\left(Q^{n+2/3}, t_n + \frac{3}{4}\Delta t\right) \\
Q^{n+1} &= Q^{n+2/3} + \frac{8}{15}\Delta t G
\end{aligned} \tag{25}$$

where  $G$  is the temporary variable,  $n$  the time level and  $\Delta t$  the time step.

### 3.1. SD-like scheme (SD1)

To the authors' best knowledge, in all reported spectral difference methods, flux is reconstructed with a polynomial of order  $P + 1$ , so that flux derivative has an identical polynomial order with solution. Based on this observation,  $P + 3$  fluxes at solution points plus two interface points are utilized to construct flux derivative of polynomial order  $P$ . First, in each coordinate direction, fluxes at  $P + 1$  solution points are interpolated to  $P$  ghost points  $\xi_{n+1/2}, n = 0, 1, \dots, P - 1$ , which are roots of  $P$ th Legendre polynomial, as shown in Fig. 2.

$$\left\{
\begin{array}{ll}
\tilde{F}_{i+1/2,j,k}^1 = \sum_{n=0}^P \tilde{F}_{n,j,k}^1 l_n(\xi_{i+1/2}^1) & 0 \leq i \leq P - 1, \quad 0 \leq j, k \leq P \\
\tilde{F}_{i,j+1/2,k}^2 = \sum_{n=0}^P \tilde{F}_{i,n,k}^2 l_n(\xi_{j+1/2}^2) & 0 \leq j \leq P - 1, \quad 0 \leq i, k \leq P \\
\tilde{F}_{i,j,k+1/2}^3 = \sum_{n=0}^P \tilde{F}_{i,j,n}^3 l_n(\xi_{k+1/2}^3) & 0 \leq k \leq P - 1, \quad 0 \leq i, j \leq P
\end{array}
\right. \tag{26}$$

Then,  $P + 2$  fluxes at  $P$  ghost points and two interface flux points  $\pm 1$ , denoted as  $\xi_{n+1/2}, n = -1, 1, \dots, P$  for convenience, are employed to compute flux derivatives at

solution points by Lagrange differentiation.

$$\left\{ \begin{array}{l} \frac{\partial \tilde{F}^1}{\partial \xi^1} \Big|_{i,j,k} = \sum_{n=-1}^P \tilde{F}_{n+1/2,j,k}^1 h'_{n+1/2}(\xi_i^1) \\ \frac{\partial \tilde{F}^2}{\partial \xi^2} \Big|_{i,j,k} = \sum_{n=-1}^P \tilde{F}_{i,n+1/2,k}^2 h'_{n+1/2}(\xi_j^2) \\ \frac{\partial \tilde{F}^3}{\partial \xi^3} \Big|_{i,j,k} = \sum_{n=-1}^P \tilde{F}_{i,j,n+1/2}^3 h'_{n+1/2}(\xi_k^3) \end{array} \right. \quad 0 \leq i, j, k \leq P \quad (27)$$

where  $h_{n+1/2}(\xi)$  is Lagrange basis of polynomial  $P + 1$

$$h_{n+1/2}(\xi) = \prod_{s=-1, s \neq n}^{P-1} \frac{\xi - \xi_{s+1/2}}{\xi_{n+1/2} - \xi_{s+1/2}} \quad -1 \leq n \leq P \quad (28)$$

Substitution of Eqs. (26) and (27) into Eq. (15) leads to the difference matrix **D1** for scheme 1.

$$\left\{ \begin{array}{ll} D1_{m,-1} = h'_{-1/2}(\xi_m) & 0 \leq m \leq P \\ D1_{m,n} = \sum_{s=0}^{P-1} l_n(\xi_{s+1/2}) h'_{s+1/2}(\xi_m) & 0 \leq m, n \leq P \\ D1_{m,P+1} = h'_{P+1/2}(\xi_m) & 0 \leq m \leq P \end{array} \right. \quad (29)$$

### 3.2. DGSEM-like spectral difference scheme (SD2)

Given  $P + 3$  fluxes at solution points plus two interface points, we can derive directly a flux derivative of polynomial order  $P + 2$  through Lagrange differentiation. As this flux derivative preserves all the flux information of solution points and interface flux points, it will be more accurate. The calculations of flux derivatives are

written as follows

$$\left\{ \begin{array}{l} \frac{\partial \tilde{F}^1}{\partial \xi^1} \Big|_{i,j,k} = \sum_{n=-1}^{P+1} \tilde{F}_{n,j,k}^1 \hbar'_n(\xi_i^1) \\ \frac{\partial \tilde{F}^2}{\partial \xi^2} \Big|_{i,j,k} = \sum_{n=-1}^{P+1} \tilde{F}_{i,n,k}^2 \hbar'_n(\xi_j^2) \\ \frac{\partial \tilde{F}^3}{\partial \xi^3} \Big|_{i,j,k} = \sum_{n=-1}^{P+1} \tilde{F}_{i,j,n}^3 \hbar'_n(\xi_k^3) \end{array} \right. \quad 0 \leq i, j, k \leq P \quad (30)$$

where  $\hbar_n(\xi)$  is Lagrange basis of polynomial order  $P + 2$ .

$$\hbar_n(\xi) = \prod_{s=-1, s \neq n}^{P+1} \frac{\xi - \xi_s}{\xi_n - \xi_s} \quad -1 \leq n \leq P + 1 \quad (31)$$

By substituting Eq. (30) to Eq. (15), difference matrix **D2** for SD2 scheme is directly derived as follows

$$D2_{m,n} = \hbar'_n(\xi_m) \quad 0 \leq m \leq P, \quad -1 \leq n \leq P + 1 \quad (32)$$

It is remarkable that the CGSD method with SD2 scheme and discontinuous Galerkin spectral element method (DGSEM) [24] essentially converge to the same method, which is proved in the Appendix [Appendix A](#). Therefore, we denote scheme 2 as a DGSEM-like scheme.

### 3.3. Hybrid scheme (SD3)

Compared with SD2 scheme, SD1 scheme is less accurate for computing flux derivative, but to its advantage, the Courant-Friedrichs-Lowy (CFL) limit is nearly 3/2 times as large, which will be shown with Fourier analysis in section 4. Thus, the combination of SD1 and SD2 schemes will bring a balance for accuracy and stability. Given a weight  $w$ , the differentiation matrix **D3** of the hybrid scheme SD3 is defined by:

$$\mathbf{D3} = w\mathbf{D1} + (1-w)\mathbf{D2} \quad (33)$$

Scheme SD3 results in SD1 with  $w = 1$ , SD2 with  $w = 0$ . As scheme SD1 and SD2 are special cases of SD3, the subsequent discussions will be focused on CGSD method with SD3 scheme if not stated otherwise.

#### 4. Fourier analysis of the CGSD method

##### 4.1. Dissipation and dispersion

To study the dissipation and dispersion properties of the CGSD method, Fourier technique are utilized. Without the loss of generalization and for clarity, one-dimensional unsteady advection equation is considered.

$$\frac{\partial u}{\partial t} + \frac{\partial (au)}{\partial x} = 0, \quad a > 0 \quad (34)$$

where  $a$  is the constant wave speed. A uniform grid of mesh size is used to discretize computational domain. Discretization of Eq. (34) with the CGSD method yields the following semi-discretized equations on  $i$ th element.

$$\frac{\partial \mathbf{u}_i}{\partial t} + \frac{2a}{\Delta x} \mathbf{D} (\mathbf{M}^L \mathbf{u}_{i-1} + \mathbf{M} \mathbf{u}_i + \mathbf{M}^R \mathbf{u}_{i+1}) = 0 \quad (35)$$

where vector  $\mathbf{u}_i$  of length  $P + 1$  is composed of the solutions at solution points on the  $i$ th element,  $\mathbf{D}$  the aforementioned differentiation matrix. Matrixes  $\mathbf{M}^L$ ,  $\mathbf{M}$  and  $\mathbf{M}^R$  are all of size  $(P + 3) \times (P + 1)$  representing the interpolation of solution to solution points plus two interface flux points.

$$\mathbf{M} = \begin{bmatrix} (1 - \alpha) \mathbf{l}(-1) \\ \mathbf{E} \\ \alpha \mathbf{l}(1) \end{bmatrix}, \quad \mathbf{M}^L = \begin{bmatrix} \alpha \mathbf{l}(1) \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix}, \quad \mathbf{M}^R = \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ (1 - \alpha) \mathbf{l}(-1) \end{bmatrix} \quad (36)$$

where  $\mathbf{E}$  is the unit matrix of size  $P + 1$ ,  $\mathbf{l}(-1) = [l_0(-1), \dots, l_P(-1)]$  and  $\mathbf{l}(1) = [l_0(1), \dots, l_P(1)]$ ,  $\alpha$  the upwind coefficient for the calculation of interface flux in

upwind manner with  $\alpha = 1$  and central manner with  $\alpha = 0.5$ . Here,  $\alpha = 1$  is chosen if not stated otherwise. Any Fourier mode can be expressed as

$$\mathbf{u}_i(t) = \hat{\mathbf{u}}_i \exp(Iki\Delta x - I\omega t) \quad (37)$$

where  $\hat{\mathbf{u}}_i$  is the vector of modes containing the amplitude and phase at solution points,  $I = \sqrt{-1}$ ,  $k$  the wavenumber,  $\omega$  the frequency. For the sake of comparison, dimensionless wavenumber  $K = k\Delta x/(P + 1)$  and frequency  $\Omega = \omega\Delta x/[a(P + 1)]$  are introduced.  $\Delta x/(P + 1)$  denotes the average spacing of solution points. Substitution of Eq. (37) into Eq. (35) leads to

$$2\mathbf{D}(\mathbf{M}^L \exp(-IK(P + 1)) + \mathbf{M} + \mathbf{M}^R \exp(IK(P + 1))) := \mathbf{S} \quad (38)$$

$$\mathbf{S}\hat{\mathbf{u}}_i = I\Omega(P + 1)\hat{\mathbf{u}}_i$$

Given a real number  $K$ ,  $I\Omega(P + 1)$  is an eigenvalue of the matrix  $\mathbf{S}$  and is calculated with MATLAB® 2018a. The wavenumber  $K$  and the real part of  $\Omega$  are theoretically equal, and thus their discrepancy reflects numerical dispersion. The imaginary part of  $\Omega$  is theoretically equal to 0, and thus denotes numerical dissipation.

The dispersion and dissipation vs. scheme weight for the CGSD method are shown in Fig. 3. The modes are strongly dissipative at high wavenumbers and thus the calculation should be limited to low wavenumbers, i.e., small average spacing of solution points. Therefore, we only consider the effect of scheme weight on dispersion and dissipation at low wavenumbers. With the increase of scheme weight, the dispersion curve varies from overshoot to undershoot, and performs best at about  $w = 0.5$  for  $P = 2$ ,  $w = 0.75$  for  $P = 3$ ,  $w = 1.0$  for  $P = 4$ ; the maximum value of dispersion curve decreases monotonously, which implies an improvement in numerical stability. The dissipation curve shows a relatively simple tendency: the increase of scheme weight will lead to increase of numerical dissipation.

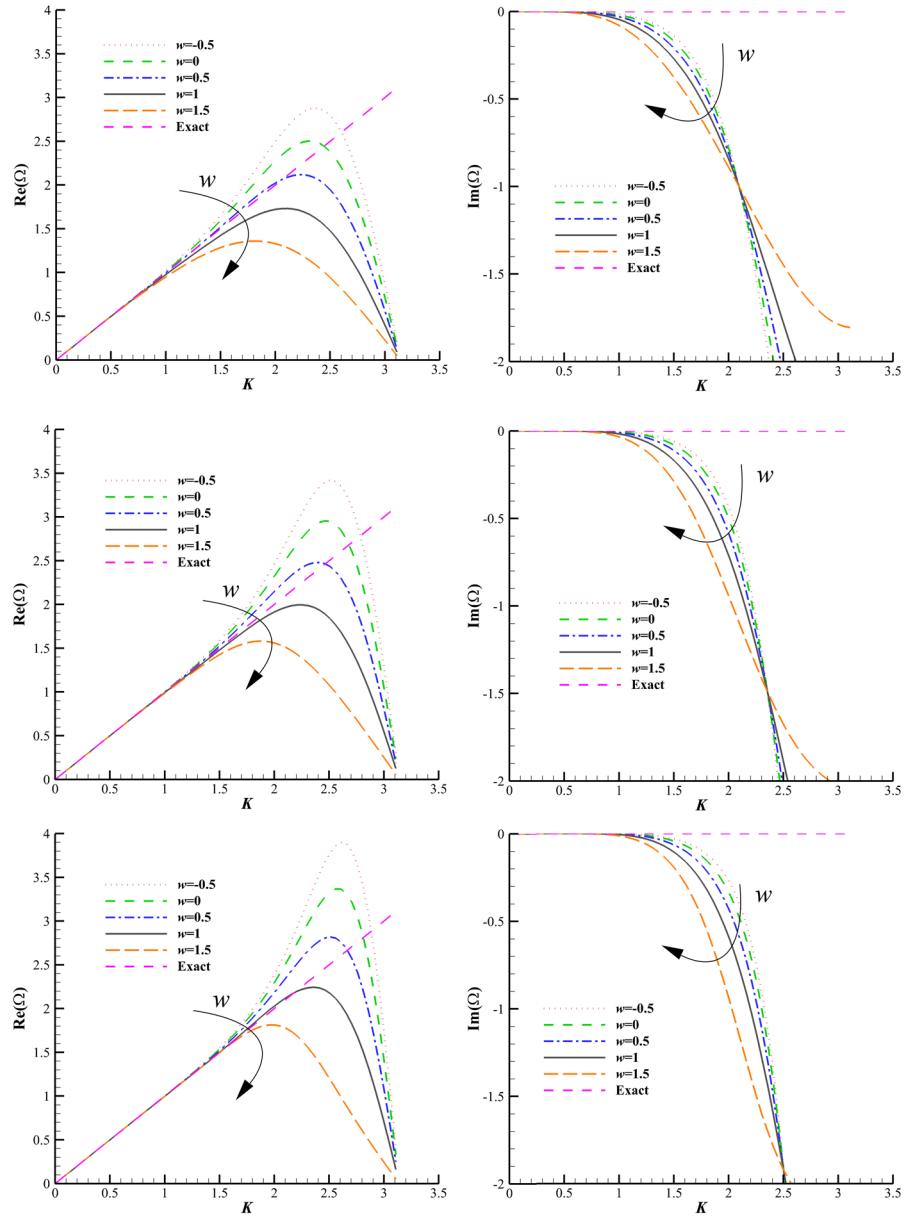


Figure 3: Dispersion (left) and dissipation (right) curves for the CGSD method of reconstruction order  $P = 2$  (top),  $P = 3$  (middle) and  $P = 4$  (bottom).

#### 4.2. CFL limit

CFL limit is analysed for the CGSD method. Marching the semi-discretization Eq. (35) in time with Eq. (25) leads to the fully-discretization equation as follows

$$\mathbf{u}^{t+\Delta t} = \left[ -\left( \mathbf{S} \frac{a\Delta t}{\Delta x} \right)^3 / 6 + \left( \mathbf{S} \frac{a\Delta t}{\Delta x} \right)^2 / 2 - \left( \mathbf{S} \frac{a\Delta t}{\Delta x} \right) + \mathbf{E} \right] \mathbf{u}^t := \mathbf{P} \mathbf{u}^t \quad (39)$$

The stability criterion of the linear system is expressed in terms of spectral radius  $\rho$  of matrix  $\mathbf{P}$

$$\rho(\mathbf{P}) \leq 1 \quad \forall K \in [0, \pi] \quad (40)$$

And thus CFL limit is determined by the allowable maximum time step

$$\text{CFL} = \frac{a\Delta t_{\max}}{\Delta x} \quad (41)$$

The CFL limit vs. scheme weight for the CGSD method with reconstruction order  $P=2, 3$  and  $4$  are shown in Fig. 4. It can be seen that CFL limit increases nonlinearly with weight  $w$  for a given  $P$ . For the same weight, CFL limit decreases quickly with  $P$ . Besides, CFL limit of SD1 ( $w = 1$ ) is about  $3/2$  times that of SD2 ( $w = 0$ ), which agrees with the work of Huynh [15] on the comparison between the spectral difference and discontinuous Galerkin method.

## 5. Accuracy verification and efficiency comparison of the CGSD method

In order to quantify the accuracy and efficiency of the CGSD method, Euler vortex propagation problem with an analytical solution is considered. The computational domain is set as  $\Omega = [-5, 5]^3$ . According to Shu [25], the analytical solution is composed of a uniform flow field  $[\rho, u, v, w, p]^T = [1, 1, 1, 0, 1]^T$  and a perturbation

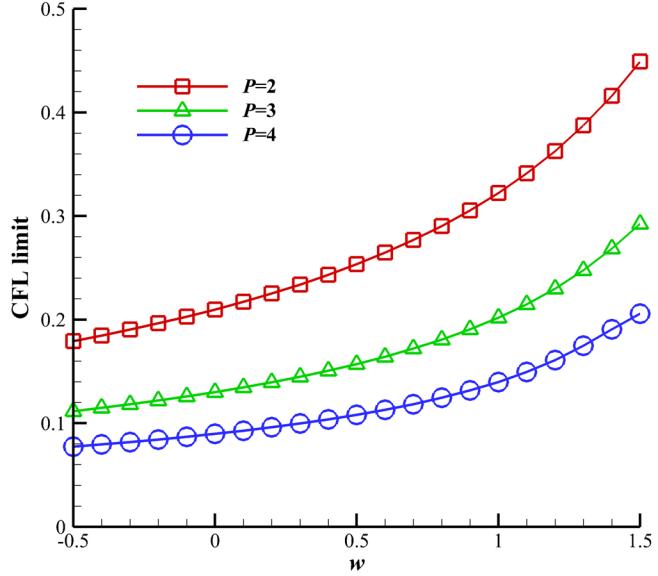


Figure 4: CFL limit of the CGSD method.

defined as

$$\left\{ \begin{array}{l} [\delta u, \delta v, \delta w]^T = \frac{\varepsilon}{2\pi} e^{0.5(1-r^2)} \mathbf{r} \\ \delta T = -\frac{\varepsilon^2 (\kappa - 1)}{8\kappa\pi^2} e^{1-r^2} \\ \delta S = \delta \left( \frac{p}{\rho^\kappa} \right) = 0 \end{array} \right. \quad (42)$$

where  $\mathbf{r} = [-(y - y_0), (x - x_0), 0]^T$  with  $x_0 = ut$ ,  $y_0 = vt$  and  $r = \|\mathbf{r}\|$ . The strength of vortex is chosen to be  $\varepsilon = 5$ .

The third, fourth and fifth order CGSD methods are employed for the simulation of the problem, which is governed by Eq. (1) with vanishing viscous terms. All the simulations start from  $t = 0$  to  $t = 1$ . Four sets of grids, namely  $4 \times 4 \times 1$ ,  $8 \times 8 \times 1$ ,  $16 \times 16 \times 1$  and  $40 \times 40 \times 1$  meshes, are employed to discretize the computational domain. Dirichlet boundary conditions with time varying analytical solutions are

enforced at surfaces  $x = \pm 5$  and  $y = \pm 5$ . Periodic boundary condition are applied to surfaces  $z = \pm 5$ . The  $L_2$  error of density in integral sense is defined to assess the simulation error of the CGSD method.

$$L_2(\rho) = \sqrt{\frac{\int_{\Omega} (\rho - \rho^{\text{exact}})^2 dV}{\int_{\Omega} dV}} \quad (43)$$

First, we consider the convergence order of the CGSD method with scheme weights  $w = -0.5, 0, 0.5, 1.0$  and  $1.5$ . A very small time step  $\Delta t = 10^{-4}$  is chosen so that temporal discretization error is negligibly small compared to spatial discretization error. The convergence order can be deduced from the slope of  $L_2$  error-element size curve as shown in Fig. 5 at double logarithmic coordinates. It is shown that  $(P + 1)$ th CGSD method with each scheme weight  $= -0.5, 0, 0.5, 1.0$  and  $1.5$  achieves approximately a convergence order  $P + 1$ . However, the absolute numerical error increases slightly with weight  $w$ , which coincides with the dissipation analysis as shown in Fig. 3.

Second, we consider the computational efficiency of the CGSD method. The previous analysis of numerical accuracy is conducted with a sufficiently small time step, which mainly reveals spatial term discretization accuracy but makes less sense for computational efficiency. Therefore, we use the maximum time step according to CFL limit. The CGSD method with  $w = -0.5, -0.4, \dots, 1.5$  is tested. Following Wang [26], the ratio of the simulation error to the consumed CPU time is employed to evaluate the computational efficiency, which follows

$$L_2 \propto (\text{CPU\_time})^{-B} \quad (44)$$

For each weight  $w$ , the estimation of  $B$  is obtained by solving the following linear regression model:

$$\log(L_2) = -B \times \log(\text{CPU\_time}) + C + \varepsilon \quad (45)$$

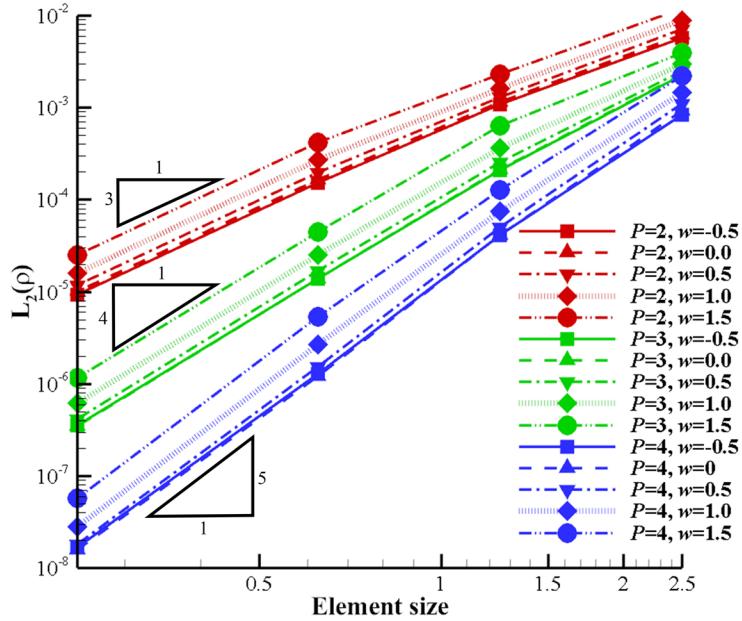


Figure 5: Convergence test of Euler vortex propagation problem for the CGSD method.

where  $B$  and  $C$  are undetermined coefficients,  $\varepsilon$  a disturbance term. A larger value of  $B$  and a smaller value of  $C$  imply a higher computational efficiency. As  $B$  and  $C$  are the slope and intercept of the fitted regression line respectively,  $B$  is the primary reference value for computational efficiency while  $C$  the secondary one. Fig. 6(a) displays the estimation of  $B$  and  $C$  vs. scheme weight  $w$  and reconstruction order  $P$ . A small variation of  $B$  is observed at different weights under consideration, implying that the CGSD method is almost of similar computational efficiency regardless of  $w$ . The maximum computational efficiency is achieved with the maximum  $B$  at about  $w = 1$  for  $P = 2$ ,  $w = 0.7$  for  $P = 3$ . For  $P = 4$ , the maximum  $B$  achieves at  $w = 1.5$ , but the corresponding  $C$  is too large, leading to a decrease in computational efficiency. Therefore, the maximum computational efficiency for  $P = 4$  is estimated at about  $w = 1.4$ . Fig. 6(b) presents the error vs. CPU time for the CGSD method

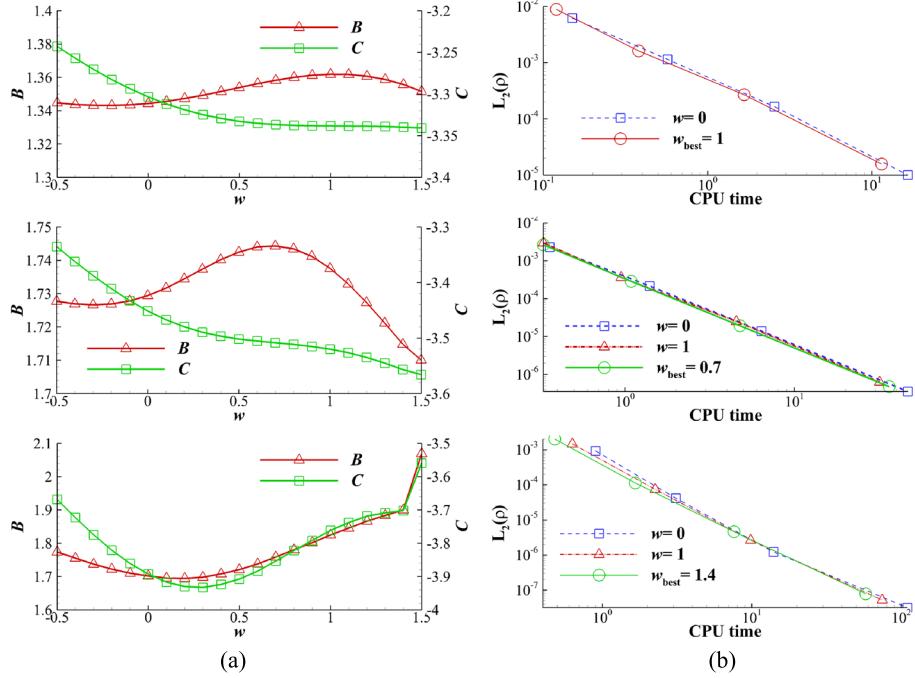


Figure 6: (a) Estimation of linear regression coefficients and (b) error vs. consumed CPU time for the CGSD method of reconstruction order  $P = 2$  (top),  $P = 3$  (middle) and  $P = 4$  (bottom).

with the best scheme weight, SD1 scheme ( $w = 1$ ) , and SD2 scheme ( $w = 0$ ) in double logarithmic coordinates. Compared with scheme SD1 and SD2, the CGSD method with best weight shows a slight improvement in computational efficiency.

## 6. Results and discussions

### 6.1. Flow over the NACA0012 airfoil

To demonstrate the capability of the CGSD method in handing curved boundary, the flow over NACA0012 airfoil is considered. The mesh is generated with an open source software HOPR [27]. As showed in Fig. 7, the grid in  $x - y$  plane contains 652 elements with 46 quadratic curved edges along the airfoil profile. In  $z$  direction,

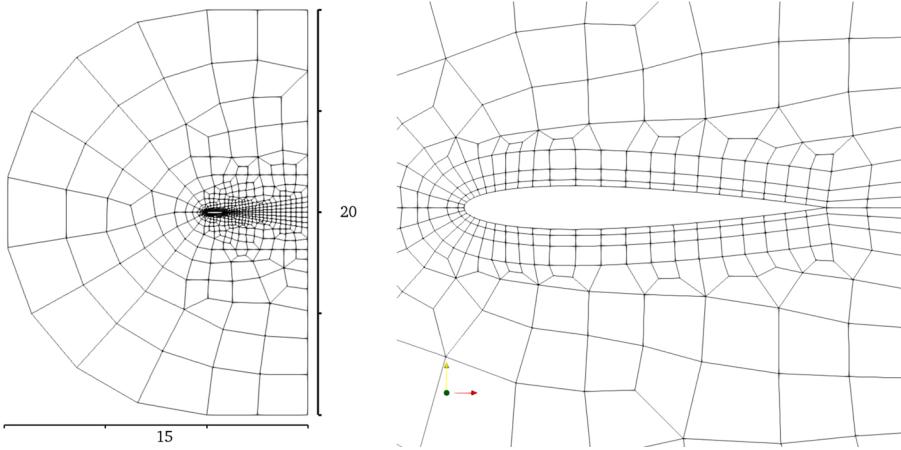


Figure 7: The global view (left) and local view (right) of mesh for flow over the NACA0012 airfoil. Given freestream as initial flow, the third, fourth and fifth order CGSD methods

one layer of grid is deployed. The Mach number of freestream is fixed at 0.2, and the Reynolds number is set as 500. Adiabatic and non-slip boundary condition is enforced at airfoil surface. Periodic boundary condition is specified in  $z$  direction. A weak imposition of far-field boundary condition [28] is applied to other surfaces.

Given freestream as initial flow, the third, fourth and fifth order CGSD methods with aforementioned best weights are utilized to simulate the flow. In order to assess the accuracy of the CGSD method, the drag coefficient  $C_d$  and lift coefficient  $C_l$  are calculated by the following definitions

$$C_d = \frac{F_d/A}{\frac{1}{2}\rho_\infty U_\infty^2}, \quad C_l = \frac{F_l/A}{\frac{1}{2}\rho_\infty U_\infty^2} \quad (46)$$

where  $F_d$  and  $F_l$  are surface integral of the drag force and lift force respectively,  $A = a \times c$  the platform area,  $c$  the chord length,  $a$  the length along  $z$  direction,  $\rho_\infty$  the freestream density,  $U_\infty$  the freestream speed.

The evolutions of the drag coefficient are presented in Fig. 8(a). The drag coefficient for reconstruction order  $P=2, 3$  and 4 reveals a similar convergent history.

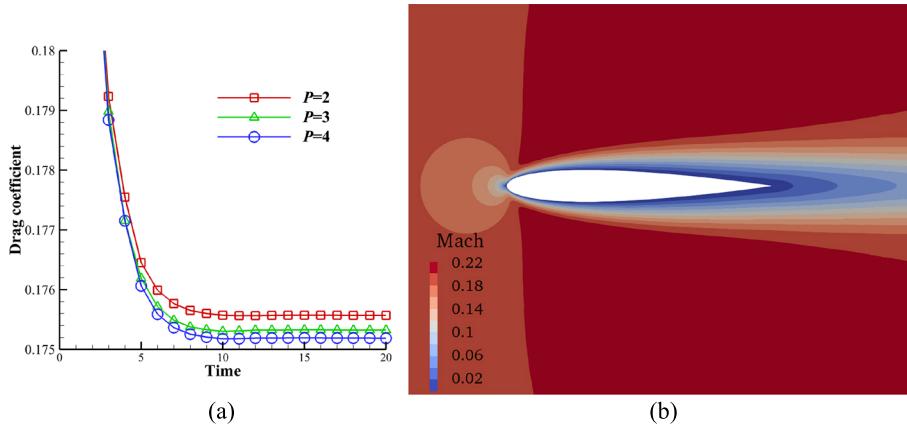


Figure 8: (a) Evolutions of drag coefficient and (b) Mach contours for the CGSD method of reconstruction order  $P = 4$ .

The convergent drag coefficient is predicted to be 0.1756 for  $P=2$ , 0.1753 for  $P = 3$  and 0.1751 for  $P = 4$ . The discrepancy among them is within 0.3%, which implies the order independence. The results agree well with the prediction 0.173 in Liang [19] using fourth order CPR and SD method. In addition, the lift coefficient for the three reconstruction orders are all predicted to be -6 orders of magnitude, agreeing well with the theoretical value 0. The corresponding Mach contours for the three reconstruction orders also show a good consistence, as shown in Fig. 8(b) for  $P = 4$ .

### 6.2. Decay of Taylor Green vortex

In order to further demonstrate the capability of the CGSD method in dealing with turbulence, the decay of Taylor Green vortex is considered. The problem concerns a three-dimensional periodic smooth initial flow transitioning to turbulence with continuous generation and decay of small-scale vortices. The initial flow speci-

fied within a cubic domain  $\Omega = [-\pi L, \pi L]^3$  is given as follows

$$\begin{aligned}\rho &= \rho_0 \\ u &= U_0 \sin\left(\frac{x}{L}\right) \cos\left(\frac{y}{L}\right) \cos\left(\frac{z}{L}\right) \\ v &= -U_0 \cos\left(\frac{x}{L}\right) \sin\left(\frac{y}{L}\right) \cos\left(\frac{z}{L}\right) \\ w &= 0 \\ p &= \frac{\rho_0 U_0^2}{\kappa Ma^2} + \frac{\rho_0 U_0^2}{16} \left[ 2 + \cos\left(\frac{2z}{L}\right) \right] \times \left[ \cos\left(\frac{2x}{L}\right) + \cos\left(\frac{2y}{L}\right) \right]\end{aligned}\quad (47)$$

where characteristic length  $L$  is set as 1, characteristic velocity  $U_0 = 1$  and constant density  $\rho_0 = 1$ . The viscosity coefficient  $\mu$  is specified as 0.000625, thus the Reynolds number  $\text{Re} = \rho_0 L U_0 / \mu$  of this problem is 1600. Mach number  $\text{Ma}$  is set as 0.1, for a comparison with incompressible flow simulation in Rees et al. [29].

To assess the accuracy of the CGSD method, the mean kinetic energy  $E_k$  and the mean enstrophy  $\varepsilon$  are compared with those obtained by a spectral method with an extremely high resolution  $512^3$  in Rees et al. [29].

$$E_k = \frac{\int_{\Omega} \frac{\mathbf{V} \cdot \mathbf{V}}{2} d\Omega}{\int_{\Omega} d\Omega}, \quad \varepsilon = \frac{\int_{\Omega} \frac{\boldsymbol{\omega} \cdot \boldsymbol{\omega}}{2} d\Omega}{\int_{\Omega} d\Omega} \quad (48)$$

The problem is simulated with uniform Cartesian grids of a fixed number of degrees of freedom (DOF)  $240^3$ , i.e.  $80^3$  elements for  $P = 2$ ,  $60^3$  elements for  $P = 3$  and  $48^3$  elements for  $P = 4$ . As shown in Fig. 9(a), all the simulations can obtain an accurate evolution of mean kinetic energy compared with that with spectral method, and a slight increase of accuracy with  $P$  is also observed. As shown in Fig. 9(b), the evolution of mean enstrophy is also well captured by all the simulations, with the exception of  $P = 2$ , which shows a remarkable deviation. This is resulted from that the CGSD method with  $P = 2$  is not accurate enough to resolve small-scale vortices, which play an important role in the evolution of enstrophy.

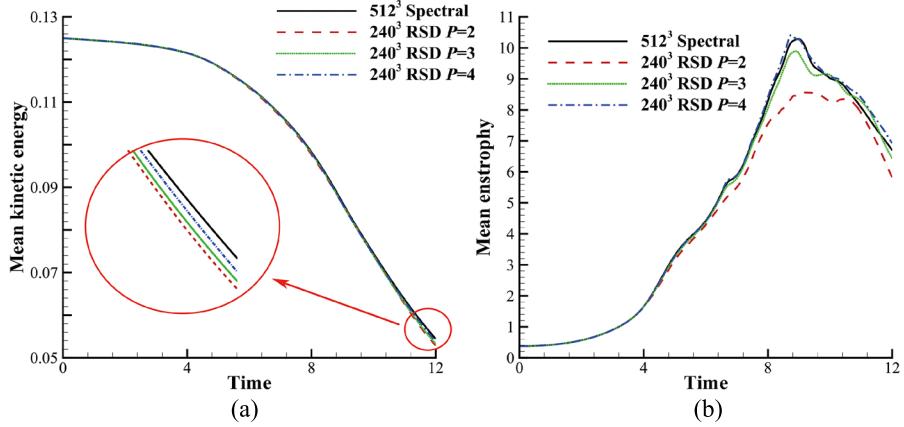


Figure 9: Evolutions of (a) the mean kinetic energy and (b) the mean enstrophy for Taylor Green vortex problem.

## 7. Conclusions

In this work, we propose a collocated-grid spectral difference (CGSD) method for three-dimensional viscous compressible flows on unstructured hexahedral grids. The CGSD method is formulated in the framework of spectral difference method but involves the solution points and only the interface flux points. In the CGSD method, flux derivatives are computed with a weight average of two proposed schemes: a spectral difference-like scheme (weight  $w = 1$ ) and a discontinuous Galerkin spectral element-like scheme (weight  $w = 0$ ). The numerical properties of the third, fourth and fifth order CGSD method are analysed with Fourier technique for one-dimensional unsteady advection problem. With the increase of scheme weight  $w$ , numerical dispersion undergoes from overshoot to undershot and performs best at about  $w = 0.5$  for  $P = 2$ ,  $w = 0.75$  for  $P = 3$ ,  $w = 1.0$  for  $P = 4$ ; numerical dissipation increases monotonously; stability range increases monotonously too. The accuracy and efficiency of the CGSD method are then verified with analytic solutions of Euler vortex propagation problem. By simulating the problem with a sufficiently

small time step, the  $P + 1$  order of convergence is verified independent of scheme weight  $w$  for reconstruction order  $P$ , while the absolute error increases with the scheme weight  $w$ . By simulating the problem with maximum time step according to CFL number, the computational efficiency varies slightly with scheme weight and the maximum efficiency is achieved with scheme weight about  $w = 1$  for  $P = 2$ ,  $w = 0.7$  for  $P = 3$ ,  $w = 1.4$  for  $P = 4$ . Finally, the capability of the CGSD method in handing curved boundary and turbulence is demonstrated with two typical problems: flow over NACA0012 airfoil and decay of Taylor Green vortex, and the numerical results agree well with the available data. The CGSD method is quite generic and has the potential in large eddy simulation and direct numerical simulation of fluid flow and heat transfer problems with complex geometry, such as compressible turbulent flow in turbomachinery.

## Appendix A. The proving of the equivalence between the CGSD method with SD2 scheme and DGSEM

Substituting Eq. (32) into Eq. (23), we have the semi-discretized governing equation of the CGSD method with SD2 scheme.

$$\frac{\partial \tilde{Q}_{i,j,k}}{\partial t} + \sum_{n=-1}^{P+1} \tilde{F}_{n,j,k}^1 \hbar'_n (\xi_i^1) + \sum_{n=-1}^{P+1} \tilde{F}_{i,n,k}^2 \hbar'_n (\xi_j^2) + \sum_{n=-1}^{P+1} \tilde{F}_{i,j,n}^3 \hbar'_n (\xi_k^3) = 0 \quad (\text{A.1})$$

Besides, the semi-discretized governing equation of discontinuous Galerkin spectral element method (DGSEM) [24] is formulated as follows.

$$\begin{aligned} \frac{\partial \tilde{Q}_{i,j,k}}{\partial t} + \sum_{n=0}^P & \left( -\frac{\omega_n}{\omega_i} l'_i(\xi_n^1) \tilde{F}_{n,j,k}^1 \right) + \left( -\frac{l_i(-1)}{\omega_i} \tilde{F}_{-1,j,k}^1 + \frac{l_i(1)}{\omega_i} \tilde{F}_{P+1,j,k}^1 \right) \\ & + \sum_{n=0}^P \left( -\frac{\omega_n}{\omega_j} l'_j(\xi_n^2) \tilde{F}_{i,n,k}^2 \right) + \left( -\frac{l_j(-1)}{\omega_j} \tilde{F}_{i,-1,k}^2 + \frac{l_j(1)}{\omega_j} \tilde{F}_{i,P+1,k}^2 \right) \\ & + \sum_{n=0}^P \left( -\frac{\omega_n}{\omega_k} l'_k(\xi_n^3) \tilde{F}_{i,j,n}^3 \right) + \left( -\frac{l_k(-1)}{\omega_k} \tilde{F}_{i,j,-1}^3 + \frac{l_k(1)}{\omega_k} \tilde{F}_{i,j,P+1}^3 \right) = 0 \end{aligned} \quad (\text{A.2})$$

where  $\omega_n$  is the weight of Gauss-Legendre quadrature at point  $\omega_n$ .

$$\omega_n = \frac{2}{(1 - \xi_n^2) [L'_{P+1}(\xi_n)]^2} \quad 0 \leq n \leq P \quad (\text{A.3})$$

In order to prove the equivalence between Eq. (A.1) and Eq.(A.2), we will prove the following equations.

$$\hbar'_n(\xi_i) = -\frac{\omega_n}{\omega_i} l'_i(\xi_n) \quad 0 \leq i, n \leq P \quad (\text{A.4})$$

$$\hbar'_{-1}(\xi_n) = -\frac{l_n(-1)}{\omega_n} \quad 0 \leq n \leq P \quad (\text{A.5})$$

$$\hbar'_{P+1}(\xi_n) = \frac{l_n(1)}{\omega_n} \quad 0 \leq n \leq P \quad (\text{A.6})$$

First, Lagrange basis  $l_n(\xi)$  and  $\hbar_n(\xi)$  are written in the alternative form.

$$l_n(\xi) = \frac{L_{P+1}(\xi)}{(\xi - \xi_n) L'_{P+1}(\xi_n)} \quad 0 \leq n \leq P \quad (\text{A.7})$$

$$\left\{ \begin{array}{ll} \hbar_n(\xi) = \frac{(\xi^2 - 1)}{(\xi_n^2 - 1)} l_n(\xi) & 0 \leq n \leq P \\ \hbar_{-1}(\xi) = \frac{(\xi - 1) L_{P+1}(\xi)}{-2 L_{P+1}(-1)} \\ \hbar_{P+1}(\xi) = \frac{(\xi + 1) L_{P+1}(\xi)}{2 L_{P+1}(1)} \end{array} \right. \quad (\text{A.8})$$

Next, their derivatives at solution points are derived.

$$\begin{cases} l'_n(\xi_i) = \frac{L'_{P+1}(\xi_i)}{(\xi_i - \xi_n)L'_{P+1}(\xi_n)} & 0 \leq i \neq n \leq P \\ l'_n(\xi_i) = \frac{L''_{P+1}(\xi_i)}{2L'_{P+1}(\xi_n)} & 0 \leq i = n \leq P \end{cases} \quad (\text{A.9})$$

$$\begin{cases} h'_n(\xi_i) = \frac{l'_n(\xi_i)(\xi_i^2 - 1) + 2\xi_i \delta_{i,n}}{\xi_n^2 - 1} & 0 \leq i, n \leq P \\ h'_{-1}(\xi_i) = \frac{(\xi_i - 1)L'_{P+1}(\xi_i)}{-2L_{P+1}(-1)} & 0 \leq i \leq P \\ h'_{P+1}(\xi_i) = \frac{(\xi + 1)L'_{P+1}(\xi_i)}{2L_{P+1}(1)} & 0 \leq i \leq P \end{cases} \quad (\text{A.10})$$

Then,

$$\frac{h'_n(\xi_i)}{l'_i(\xi_n)} = \frac{\xi_i^2 - 1}{\xi_n^2 - 1} \frac{l'_n(\xi_i)}{l'_i(\xi_n)} = -\frac{(\xi_i^2 - 1)[L'_{P+1}(\xi_i)]^2}{(\xi_n^2 - 1)[L'_{P+1}(\xi_n)]^2} = -\frac{\omega_n}{\omega_i} \quad 0 \leq i \neq n \leq P \quad (\text{A.11})$$

$$\frac{h'_n(\xi_n)}{l'_n(\xi_n)} = \frac{(\xi_n^2 - 1)L''_{P+1}(\xi_n) + 4\xi_n L'_{P+1}(\xi_n)}{(\xi_n^2 - 1)L''_{P+1}(\xi_n)} \quad 0 \leq n \leq P \quad (\text{A.12})$$

As  $L_{p+1}(\xi)$  satisfies Legendre's differential equation

$$[(1 - \xi^2)L'_{P+1}(\xi)]' + (P + 1)(P + 2)L_{P+1}(\xi) = 0 \quad (\text{A.13})$$

Therefore,

$$[(1 - \xi_n^2)L'_{P+1}(\xi_n)]' = (1 - \xi_n^2)L''_{P+1}(\xi_n) - 2\xi_n L'_{P+1}(\xi_n) = 0 \quad (\text{A.14})$$

Substitution of Eq. (A.14) into Eq. (A.12) leads to

$$\frac{h'_n(\xi_n)}{l'_n(\xi_n)} = -1 \quad (\text{A.15})$$

finally,

$$\frac{h'_{-1}(\xi_n)}{l_n(-1)} = -\frac{(1 - \xi_n^2)[L'_{P+1}(\xi_n)]^2}{2[L_{P+1}(-1)]^2} = -\frac{1}{\omega_n} \quad 0 \leq n \leq P \quad (\text{A.16})$$

$$\frac{h'_{P+1}(\xi_n)}{l_n(1)} = \frac{(1 - \xi_n^2) [L'_{P+1}(\xi_n)]^2}{2[L_{P+1}(-1)]^2} = \frac{1}{\omega_n} \quad 0 \leq n \leq P \quad (\text{A.17})$$

Thus, Eqs. (A.4)-(A.6) are true, proving that the CGSD method with SD2 scheme converges to the DGSEM.

## Acknowledgements

This work is financially supported by the National Key Research and Development Project of China [Grant number 2016YFB0200901], National Natural Science Foundation of China [Grant number 51776154] and Shaanxi Key Research and Development Project [Grant number 2018KWZ-01.]

## References

- [1] Z. J. Wang, High-order CFD tools for aircraft design, *Phil. Trans. R. Soc. A.* 372 (2022) (2014) 20130318.
- [2] D. A. Kopriva, J. H. Kolias, A conservative staggered-grid Chebyshev multidomain method for compressible flows, *J. Comput. Phys.* 125 (1) (1996) 244–261. [doi:10.1006/jcph.1996.0091](https://doi.org/10.1006/jcph.1996.0091).
- [3] D. A. Kopriva, A conservative staggered-grid chebyshev multidomain method for compressible flows II. A semi-structured method, *J. Comput. Phys.* 128 (2) (1996) 475–488. [doi:10.1006/jcph.1996.0225](https://doi.org/10.1006/jcph.1996.0225).
- [4] W. Q. Chen, Y. P. Ju, C. H. Zhang, A multidomain multigrid pseudospectral method for incompressible flows, *Numer. Heat Transf. Part B-Fundamentals.* 74 (1) (2018) 415–431. [doi:10.1080/10407790.2018.1490090](https://doi.org/10.1080/10407790.2018.1490090).

- [5] Y. Liu, M. Vinokur, Z. J. Wang, Spectral difference method for unstructured grids I: Basic formulation, *J. Comput. Phys.* 216 (2) (2006) 780–801. [doi:10.1016/j.jcp.2006.01.024](https://doi.org/10.1016/j.jcp.2006.01.024).
- [6] Z. J. Wang, Y. Liu, G. May, A. Jameson, Spectral difference method for unstructured grids II: extension to the Euler equations, *J. Sci. Comput.* 32 (1) (2007) 45–71. [doi:10.1007/s10915-006-9113-9](https://doi.org/10.1007/s10915-006-9113-9).
- [7] Y. Z. Sun, Z. J. Wang, Y. Liu, High-order multidomain spectral difference method for the Navier-Stokes equations on unstructured hexahedral grids, *Commun. Comput. Phys.* 2 (2) (2007) 310–333. [doi:10.2514/6.2004-2133](https://doi.org/10.2514/6.2004-2133).
- [8] J. H. Gao, X. D. Li, Implementation of delayed detached eddy simulation method to a high order spectral difference solver, *Comput. Fluids.* 154 (2017) 90–101. [doi:10.1016/j.compfluid.2017.05.035](https://doi.org/10.1016/j.compfluid.2017.05.035).
- [9] J. B. Chapelier, G. Lodato, Study of the spectral difference numerical sissipation for turbulent flows using unstructured grids, *Flow, Turbul. Combust.* 99 (3-4) (2017) 643–664. [doi:10.1007/s10494-017-9847-5](https://doi.org/10.1007/s10494-017-9847-5).
- [10] J. B. Chapelier, G. Lodato, A. Jameson, A study on the numerical dissipation of the spectral difference method for freely decaying and wall-bounded turbulence, *Comput. Fluids.* 139 (2016) 261–280. [doi:10.1016/j.compfluid.2016.03.006](https://doi.org/10.1016/j.compfluid.2016.03.006).
- [11] J. H. Gao, X. D. Li, Detached eddy simulation of a highlift airfoil noise with spectral difference method, *The Journal of the Acoustical Society of America* 140 (4) (2016) 3321.

- [12] B. Zhang, C. L. Liang, J. J. Yang, Y. W. Rong, A 2D parallel high-order sliding and deforming spectral difference method, *Comput. Fluids.* 139 (2016) 184–196.  
[doi:10.1016/j.compfluid.2016.06.019](https://doi.org/10.1016/j.compfluid.2016.06.019).
- [13] M. L. Yu, Z. J. Wang, H. Hu, A high-order spectral difference method for unstructured dynamic grids, *Comput. Fluids.* 48 (1) (2011) 84–97. [doi:10.1016/j.compfluid.2011.03.015](https://doi.org/10.1016/j.compfluid.2011.03.015).
- [14] K. Van Den Abeele, C. Lacor, Z. J. Wang, On the stability and accuracy of the spectral difference method, *J. Sci. Comput.* 37 (2) (2008) 162–188. [doi:10.1007/s10915-008-9201-0](https://doi.org/10.1007/s10915-008-9201-0).
- [15] H. T. Huynh, A Flux Reconstruction Approach to High-Order Schemes Including Discontinuous Galerkin Methods, 18th AIAA Comput. Fluid Dyn. Conf. (June) (2007) 1–42. [doi:10.2514/6.2007-4079](https://doi.org/10.2514/6.2007-4079).
- [16] Z. J. Wang, H. Y. Gao, A unifying lifting collocation penalty formulation including the discontinuous Galerkin, spectral volume/difference methods for conservation laws on mixed grids, *J. Comput. Phys.* 228 (21) (2009) 8161–8186. [doi:10.1016/j.jcp.2009.07.036](https://doi.org/10.1016/j.jcp.2009.07.036).
- [17] A. Jameson, A proof of the stability of the spectral difference method for all orders of accuracy, *J. Sci. Comput.* 45 (1-3) (2010) 348–358. [doi:10.1007/s10915-009-9339-4](https://doi.org/10.1007/s10915-009-9339-4).
- [18] G. May, On the connection between the spectral difference method and the discontinuous Galerkin method, *Commun. Comput. Phys.* 9 (4) (2011) 1071–1080. [doi:10.4208/cicp.090210.040610a](https://doi.org/10.4208/cicp.090210.040610a).

- [19] C. L. Liang, C. Cox, M. Plesniak, A comparison of computational efficiencies of spectral difference method and correction procedure via reconstruction, *J. Comput. Phys.* 239 (2013) 138–146. [doi:10.1016/j.jcp.2013.01.001](https://doi.org/10.1016/j.jcp.2013.01.001).
- [20] M. L. Yu, Z. J. Wang, Y. Liu, On the accuracy and efficiency of discontinuous Galerkin, spectral difference and correction procedure via reconstruction methods, *J. Comput. Phys.* 259 (2014) 70–95. [doi:10.1016/j.jcp.2013.11.023](https://doi.org/10.1016/j.jcp.2013.11.023).
- [21] D. A. Kopriva, Metric identities and the discontinuous spectral element method on curvilinear meshes, *J. Sci. Comput.* 26 (3) (2006) 301–327. [doi:10.1007/s10915-005-9070-8](https://doi.org/10.1007/s10915-005-9070-8).
- [22] F. Bassi, S. Rebay, A high-order accurate discontinuous finite element method for the numerical solution of the compressible Navier–Stokes equations, *J. Comput. Phys.* 131 (2) (1997) 267–279.
- [23] J. H. Williamson, Low-storage Runge-Kutta schemes, *J. Comput. Phys.* 35 (1) (1980) 48–56.
- [24] F. Hindenlang, G. J. Gassner, C. Altmann, A. Beck, M. Staudenmaier, C. D. Munz, Explicit discontinuous Galerkin methods for unsteady problems, *Comput. Fluids.* 61 (2012) 86–93.
- [25] C. W. Shu, Essentially non-oscillatory and weighted essentially non-oscillatory schemes for hyperbolic conservation laws, in: Advanced numerical approximation of nonlinear hyperbolic equations, Springer, 1998, pp. 325–432.
- [26] Z. J. Wang, High-order methods for the Euler and Navier-Stokes equations on unstructured grids, *Prog. Aerosp. Sci.* 43 (1-3) (2007) 1–41. [doi:10.1016/j.paerosci.2007.05.001](https://doi.org/10.1016/j.paerosci.2007.05.001).

- [27] F. Hindenlang, T. Boemann, C.-D. Munz, Mesh curving techniques for high order discontinuous Galerkin simulations, in: IDIHOM: Industrialization of High-Order Methods-A Top-Down Approach, Springer, 2015, pp. 133–152.
- [28] J. R. Carlson, Inflow/outflow boundary conditions with application to fun3d (2011).
- [29] W. M. van Rees, A. Leonard, D. I. Pullin, P. Koumoutsakos, A comparison of vortex and pseudo-spectral methods for the simulation of periodic vortical flows at high Reynolds numbers, *J. Comput. Phys.* 230 (8) (2011) 2794–2805.  
[doi:10.1016/j.jcp.2010.11.031](https://doi.org/10.1016/j.jcp.2010.11.031).