Commun. Comput. Phys.
doi: 10.4208/cicp.020611.090312a

Vol. **13**, No. 4, pp. 1013-1044
April 2013

Differential Formulation of Discontinuous Galerkin and Related Methods for the Navier-Stokes Equations

Haiyang Gao[1,*], Z. J. Wang[1] and H. T. Huynh[2]

[1] *Department of Aerospace Engineering and CFD Center, Iowa State University, 2271 Howe Hall Ames, IA 50011, USA.*
[2] *NASA Glenn Research Center, MS 5-11, Cleveland, OH 44135, USA.*

Received 2 June 2011; Accepted (in revised version) 9 March 2012

Available online 21 September 2012

Abstract. A new approach to high-order accuracy for the numerical solution of conservation laws introduced by Huynh and extended to simplexes by Wang and Gao is renamed CPR (correction procedure or collocation penalty via reconstruction). The CPR approach employs the differential form of the equation and accounts for the jumps in flux values at the cell boundaries by a correction procedure. In addition to being simple and economical, it unifies several existing methods including discontinuous Galerkin, staggered grid, spectral volume, and spectral difference. To discretize the diffusion terms, we use the BR2 (Bassi and Rebay), interior penalty, compact DG (CDG), and I-continuous approaches. The first three of these approaches, originally derived using the integral formulation, were recast here in the CPR framework, whereas the I-continuous scheme, originally derived for a quadrilateral mesh, was extended to a triangular mesh. Fourier stability and accuracy analyses for these schemes on quadrilateral and triangular meshes are carried out. Finally, results for the Navier-Stokes equations are shown to compare the various schemes as well as to demonstrate the capability of the CPR approach.

AMS subject classifications: 76N15

Key words: Discontinuous Galerkin, lifting collocation penalty, flux reconstruction, Navier-Stokes equations, correction procedure via reconstruction, unstructured hybrid grids.

1 Introduction

Second-order methods are currently popular in fluid flow simulations. For many important problems such as computational aeroacoustics, vortex-dominant flows, and large

[*]Corresponding author. *Email addresses:* `hgao@iastate.edu` (H. Gao), `zjw@iastate.edu` (Z. J. Wang), `hung.t.huynh@nasa.gov` (H. T. Huynh)

©2013 Global-Science Press

eddy and direct numerical simulation of turbulent flows, the number of grid points required by a second-order scheme is often beyond the capacity of current computers. For these problems, high-order methods hold the promise of accurate solutions with a manageable number of grid points. Numerous high-order methods have been developed in the last two decades. Here, we focus only on those that employ a polynomial to approximate the solution in each cell or element, and the polynomials collectively form a function which is discontinuous across cell boundaries. Commonly used methods of this type include discontinuous Galerkin (DG) [2, 3, 5–7], staggered-grid (SG) [15], spectral volume (SV) [24–27], and spectral difference (SD) [17–19]. Among these, DG and SV are usually formulated via the integral form of the equation, whereas SG and SD, the differential one. From an algorithm perspective, the difference among these methods lies in the definition of the degrees of freedom (DOFs), which determine the polynomial in each cell, and how these DOFs are updated.

High-order methods for conservation laws discussed above deal with the first derivative. Diffusion problems (viscous flows) involve the second derivative. There are many ways to extend a method of estimating the first derivative to the second; Arnold et al. analyzed several of them in [1]. Here, we restrict ourselves to approaches of compact stencil: the second derivative estimate in an element involves data of only that element and the immediate face neighbors. Such approaches have several advantages: the associated boundary conditions are simpler, the coding is easier, and the implicit systems are smaller. The four schemes of compact stencil employed are BR2 (Bassi and Rebay) [4], compact DG or CDG [20], interior penalty [9,12], and I-continuous (the value and derivative are continuous across the interface) [14]. The BR2 scheme, an improvement of the non-compact BR1 [2], is the first successful approach of this type for the Navier-Stokes equations. The CDG scheme is a modification of the local DG or LDG [8] to obtain compactness for an unstructured mesh. The interior penalty scheme is employed here with a penalty coefficient using correction function [14]. The I-continuous approach is highly accurate for linear problems on a quadrilateral mesh. Nicknamed "poor man's recovery", it can be considered as an approximation to the recovery approach of Van Leer and Nomura [23]. (The recovery approach is beyond the scope of this paper since, although it is more accurate than the schemes discussed here based on Fourier analysis [14], is more complex and costly.)

For conservation laws, Huynh (2007) [13] introduced an approach to high-order accuracy called flux reconstruction (FR). The approach solves the equations in differential form. It evaluates the first derivative of a discontinuous piecewise polynomial function by employing the straightforward derivative estimate together with a correction which accounts for the jumps at the interfaces. The FR framework unifies several existing methods: with appropriate correction terms, it recovers DG, SG, SV, SD methods. This framework was extended to diffusion problems using quadrilateral meshes in [14], where several existing schemes for diffusion were recast and analyzed. Wang and Gao (2009) [28] extended the FR idea to 2D triangular and mixed meshes with the lifting collocation penalty (LCP) formulation. The LCP method was applied to solve the Euler and later Navier-Stokes equations in both two [10] and three dimensions [11]. Due to their tight connection, the FR and LCP methods are renamed *correction procedures via reconstruction* or CPR. The CPR formulation does not involve numerical integrations; the mass matrix inversion is built-in and therefore not needed; as a consequence, the approach is simpler and generally results in schemes more efficient than those by quadrature-based formulations.

In the present study, we demonstrate the capability of the CPR formulation for the numerical solutions of the Navier-Stokes equations as well as investigate the performance of several discretization approaches for diffusion. To accomplish these objectives and to prepare for complex geometries and viscous boundary layers, we first formulate the CPR framework on hybrid meshes of quadrilaterals and triangles. Next, we recast the four approaches for diffusion discussed above in this framework. Fourier stability and accuracy analyses of these schemes are then carried out on square and triangular meshes. Finally, results for several benchmark problems are shown.

The paper is organized as follows. The CPR formulation for hybrid meshes is presented in Section 2. Section 3 describes the discretization of diffusion/viscous terms. Fourier analyses of the schemes for diffusion on square and triangular meshes are carried out in Section 4. Numerical tests are shown in Section 5 for the Poisson and the Navier-Stokes equations. Conclusions are drawn in Section 6.

2 Governing equations and numerical formulations

2.1 Governing equations

For the scalar case, we consider the heat equation

$$u_t - \nabla^2 u = -f \tag{2.1}$$

with appropriate boundary conditions. Our goal is to obtain numerical solutions for the Navier-Stokes equations in conservative form

$$\frac{\partial Q}{\partial t} + \frac{\partial F}{\partial x} + \frac{\partial G}{\partial y} = 0, \tag{2.2}$$

where Q is the vector of conserved variables, and F and G are the flux vectors formed by the inviscid and viscous parts, $F = F^i - F^v$, $G = G^i - G^v$. More precisely,

$$Q = \begin{Bmatrix} \rho \\ \rho u \\ \rho v \\ E \end{Bmatrix}, \qquad F^i = \begin{Bmatrix} \rho u \\ p + \rho u^2 \\ \rho u v \\ u(E+p) \end{Bmatrix}, \qquad G^i = \begin{Bmatrix} \rho v \\ \rho u v \\ p + \rho v^2 \\ v(E+p) \end{Bmatrix}, \tag{2.3}$$

and

$$F^v=\mu\left\{\begin{array}{c}0\\2u_x+\lambda\left(u_x+v_y\right)\\v_x+u_y\\u\left[2u_x+\lambda\left(u_x+v_y\right)\right]+v\left(v_x+u_y\right)+\frac{C_p}{\mathrm{Pr}}T_x\end{array}\right\}, \tag{2.4a}$$

$$G^v=\mu\left\{\begin{array}{c}0\\v_x+u_y\\2v_y+\lambda\left(u_x+v_y\right)\\u\left(v_x+u_y\right)+v\left[2v_y+\lambda\left(u_x+v_y\right)\right]+\frac{C_p}{\mathrm{Pr}}T_y\end{array}\right\}. \tag{2.4b}$$

In (2.3)-(2.4), ρ is density, u and v velocity components in x and y directions, p pressure, E total energy, μ dynamic viscosity, C_p specific heat at constant pressure, Pr Prandtl number, and T temperature. For a perfect gas, pressure is related to total energy by

$$E=\frac{p}{\gamma-1}+\frac{1}{2}\rho\left(u^2+v^2\right). \tag{2.5}$$

The ratio of specific heats γ is assumed to be constant, and $\gamma=1.4$ for air. In addition, λ is set to $-2/3$ according to the Stokes hypothesis.

From (2.4), the viscous fluxes can be written as functions of both the conservative variables and their gradients,

$$F^v=F^v\left(Q,\vec{\nabla}Q\right),\qquad G^v=G^v\left(Q,\vec{\nabla}Q\right). \tag{2.6}$$

Note that the flux in the heat equation is a special case of the above: $F^v(Q,\vec{\nabla}Q)=Q_x$ and $G^v(Q,\vec{\nabla}Q)=Q_y$. Therefore, from here on unless otherwise stated, we employ only (2.6).

2.2 CPR (Correction Procedure via Reconstruction) formulation

General CPR formulation

The CPR formulation can be derived from a weighted residual method by transforming the integral formulation into a differential one. First, a hyperbolic conservation law can be written as

$$\frac{\partial Q}{\partial t}+\vec{\nabla}\bullet\vec{F}(Q)=0, \tag{2.7}$$

with proper initial and boundary conditions, where Q is the state vector and $\vec{F}=(F,G)$ is the flux vector. The computational domain is discretized into N non-overlapping elements $\{V_i\}$. Let W be an arbitrary weighting function. The weighted residual of Eq. (2.7) on element V_i can be written as

$$\int\limits_{V_i}\left(\frac{\partial Q}{\partial t}+\vec{\nabla}\bullet\vec{F}(Q)\right)WdV=\int\limits_{V_i}\frac{\partial Q}{\partial t}WdV+\int\limits_{\partial V_i}W\vec{F}(Q)\bullet\vec{n}dS-\int\limits_{V_i}\vec{\nabla}W\bullet\vec{F}(Q)dV=0. \tag{2.8}$$

Let Q_i be an approximate solution to Q at element i. We assume that in each element, the solution belongs to the space of polynomials of degree k or less, i.e., $Q_i \in P^k(V_i)$, (or P^k if there is no confusion) with no continuity requirement across element interfaces. The numerical solution Q_i should satisfy Eq. (2.8), i.e.,

$$\int_{V_i} \frac{\partial Q_i}{\partial t} W dV + \int_{\partial V_i} W \vec{F}(Q_i) \bullet \vec{n} dS - \int_{V_i} \vec{\nabla} W \bullet \vec{F}(Q_i) dV = 0. \tag{2.9}$$

Since the solution is discontinuous across element interfaces, the above surface integral is not well-defined. To remedy this problem, a common normal flux is employed:

$$\vec{F}(Q_i) \bullet \vec{n} \approx F^n_{\text{com}}(Q_i, Q_{i+}, \vec{n}), \tag{2.10}$$

where Q_{i+} is the solution onV_{i+}, which is outside V_i. Instead of (2.9), the solution is required to satisfy

$$\int_{V_i} \frac{\partial Q_i}{\partial t} W dV + \int_{\partial V_i} W F^n_{\text{com}} dS - \int_{V_i} \vec{\nabla} W \cdot \vec{F}(Q_i) dV = 0. \tag{2.11}$$

Applying integration by parts again to the last term of the above LHS, we obtain

$$\int_{V_i} \frac{\partial Q_i}{\partial t} W dV + \int_{V_i} W \vec{\nabla} \bullet \vec{F}(Q_i) dV + \int_{\partial V_i} W [F^n_{\text{com}} - F^n(Q_i)] dS = 0. \tag{2.12}$$

Here, we require test space to have the same dimension as the solution space. The test space is chosen in a manner to guarantee the existence and uniqueness of the numerical solution.

Note that the quantity $\vec{\nabla} \bullet \vec{F}(Q_i)$ involves no influence from the data in the neighboring cells; the influence of these data is represented by the above boundary integral, which is also called a "penalty term", penalizing the normal flux differences.

The next step is critical in the elimination of the test function. The boundary integral above is cast as a volume integral via the introduction of a "correction field" $\delta_i \in P^k$,

$$\int_{V_i} W \delta_i dV = \int_{\partial V_i} W [F^n] dS, \tag{2.13}$$

where $[F^n] = F^n_{\text{com}}(Q, Q^+, \vec{n}) - F^n(Q)$ is the normal flux difference. The above equation is sometimes referred to as the "lifting operator", which has the normal flux differences on the boundary as input and an element of $P^k(V_i)$ as output. Substituting Eq. (2.13) into Eq. (2.12), we obtain

$$\int_{V_i} \left[\frac{\partial Q_i}{\partial t} + \vec{\nabla} \bullet \vec{F}(Q_i) + \delta_i \right] W dV = 0. \tag{2.14}$$

If the flux vector is a linear function of the state variable, then $\vec{\nabla}\bullet\vec{F}(Q_i)\in P^{k-1}$. In this case, the terms inside the square bracket are all elements of P^k. Because the test space is selected to ensure a unique solution, Eq. (2.14) is equivalent to

$$\frac{\partial Q_i}{\partial t}+\vec{\nabla}\bullet\vec{F}(Q_i)+\delta_i=0. \tag{2.15}$$

For non-linear conservation laws, $\vec{\nabla}\bullet\vec{F}(Q_i)$ is usually not an element of $P^k(V_i)$. As a result, we approximate it by its projection onto $P^k(V_i)$, denoted by $\Pi(\vec{\nabla}\bullet\vec{F}(Q_i))$,

$$\int_{V_i}\Pi\Big(\vec{\nabla}\bullet\vec{F}(Q_i)\Big)W\mathrm{d}V=\int_{V_i}\vec{\nabla}\bullet\vec{F}(Q_i)W\mathrm{d}V. \tag{2.16}$$

We will discuss how to approximate this projection later. Eq. (2.14) then reduces to

$$\frac{\partial Q_i}{\partial t}+\Pi\Big(\vec{\nabla}\bullet\vec{F}(Q_i)\Big)+\delta_i=0. \tag{2.17}$$

With the introduction of the correction field δ_i, and a projection of $\vec{\nabla}\bullet\vec{F}(Q_i)$ for non-linear conservation laws, we have reduced the weighted residual formulation to a differential formulation, which involves no integrals.

Note that for δ_i defined by (2.13), if $W\in P^k$, Eq. (2.17) is equivalent to the DG formulation; if W varies on another space, the resulting δ_i is different, and we obtain a formulation corresponding to a different method [28].

Let the solution points (SPs) be denoted by $\{\vec{r}_{i,j}\}$ (j varies from 1 to K), as shown in Fig. 1. Let $Q_{i,j}$ be an approximation to the solution at these points. Then we require Eq. (2.17) to hold at the SPs, i.e.,

$$\frac{\partial Q_{i,j}}{\partial t}+\Pi\Big(\vec{\nabla}\bullet\vec{F}(Q_{i,j})\Big)+\delta_{i,j}=0, \tag{2.18}$$

where $\vec{\nabla}\bullet\vec{F}(Q_{i,j})=[\vec{\nabla}\bullet\vec{F}(Q)]_{\vec{r}_{i,j}}$.

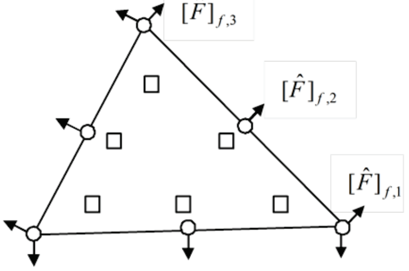


Figure 1: Solution points (squares) and flux points (circles) for $k=2$.

Along each edge, we also need to define $k+1$ points where the common normal fluxes are calculated; these points are called flux points (Fig. 1). Once the solution points and flux points are chosen, the correction at the SPs can be written as

$$\delta_{i,j}=\frac{1}{|V_i|}\sum_{f\in\partial V_i}\sum_{l}\alpha_{j,f,l}[F^n]_{f,l}S_f, \tag{2.19}$$

where $\alpha_{j,f,l}$ are constants independent of the solution. Here, again, j is the index for solution points on each cell, i the cell index, f the face index, l the index of flux points on face f, and $S_f$the area.

Note that the correction for each solution point, namely, $\delta_{i,j}$ is a linear combination of all the normal flux differences on all the faces of the cell. Conversely, a normal flux difference at a flux point on a face, say (f,l) results in a correction at all solution points j of an amount $\alpha_{j,f,l}[F^n]_{f,l}S_f/|V_i|$.

Along each edge f, the flux difference values $[F^n]_{f,l}$ at the flux points defines a 1-D polynomial of degree k denoted by $[F^n]_f$. Eq. (2.13) can then be written as

$$\int_{V_i}W_k\sum_j L_j\delta_{i,j}dV=\sum_{f}^{f\in\partial V_i}\int_f W_k[F^n]_f dS. \tag{2.20}$$

Here L_j is the Lagrange polynomials, and W_k the test functions. This equation yields a linear system as k varies; the unknowns are $\delta_{i,j}$; both k and j vary from 1 to K. By setting $W_k=L_k$, the unknowns can be solved and the coefficients can be determined. Note that after the approximation using the Lagrange polynomials (1D and 2D), the volume and surface integrals are carried out exactly in (2.20) for the solution of the linear system.

To approximate the term $\Pi(\vec{\nabla}\bullet\vec{F}(Q_i))$, either the Lagrange polynomial (LP) or the chain-rule (CR) approach can be used.

For the LP approach, the flux function is first approximated by a Lagrange polynomial; then the divergence operator is applied (the following right hand side),

$$\Pi\left(\vec{\nabla}\bullet\vec{F}(Q_i)\right)=\nabla\bullet\sum_{j\in SP}L_j\vec{F}_j. \tag{2.21}$$

As for the CR approach, it was mentioned in [13] as an alternative. But numerical tests later in [28] show that it in fact yields more accurate solutions for non-linear cases. The CR approach can be written as

$$\Pi\left(\vec{\nabla}\bullet\vec{F}(Q_i)\right)=\nabla\bullet\left(\sum_{j\in SP}L_j\frac{\partial F}{\partial Q}\nabla Q_j\right). \tag{2.22}$$

The idea of the CR approach is to obtain a best divergence quantity of the fluxes at solution points in the following sense: if the exact solution is identical to a solution polynomial (of degree k), then this approach yields the exact divergence values at the solution points. Note that the LP approach does not have this property. Comparisons of numerical tests by the two approaches for the case of non-linear fluxes can be found in [28].

Substituting (2.19) into (2.18) we obtain the following correction procedure via reconstruction (CPR) formulation

$$\frac{\partial Q_{i,j}}{\partial t}+\Pi\left(\vec{\nabla}\bullet\vec{F}(Q_{i,j})\right)+\frac{1}{|V_i|}\sum_{f\in\partial V_i}\sum_{l}\alpha_{j,f,l}[F^n]_{f,l}S_f=0. \tag{2.23}$$

It was shown that the location of SPs does not affect the numerical scheme for linear conservation laws [13,22]. For efficiency, therefore, the solution points and flux points are always chosen to include corners of the cell; in addition, the solution points are chosen to coincide with the flux points along cell faces. Furthermore, in a computation with hybrid mesh, the flux points are always of the same distribution for different cell types for ease of interface treatment (Fig. 2). For the 2D cases presented here, the Legendre Lobatto points along the edges are used as the flux points and also (part of) the solution points for both triangular and quadrilateral cells.

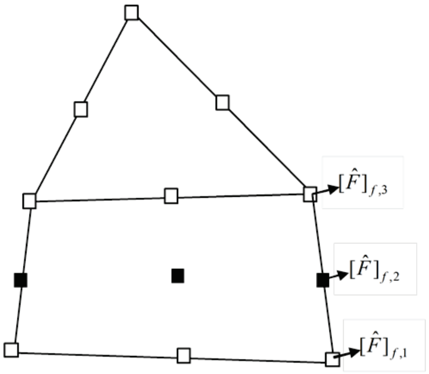


Figure 2: Solution points for the 3rd order LCP scheme on hybrid meshes.

CPR for 1D

For the one-dimensional (1D) case, consider the element $V_i=[x_{i-1/2},x_{i+1/2}]$; here, the flux $\vec{F}$ is F. Assuming a linear flux for derivation purpose, Eq. (2.22) takes the form

$$\frac{\partial Q_{i,j}}{\partial t}+\frac{\partial\left[F(Q_{i,j})\right]}{\partial x}+\alpha_{j,L}[F^n]_L+\alpha_{j,R}[F^n]_R=0, \tag{2.24}$$

where $[F^n]_L$ and $[F^n]_R$ are the flux differences on the left and right interfaces of the cell V_i. The quantities $\alpha_{j,L}$ and $\alpha_{j,R}$ are the correction coefficients. Recall that for the 1D case, the number of solution points is $K=k+1$; as a result, j varies from 1 to K, and $\alpha_{j,L}$ define a polynomial of degree $k=K-1$ as j varies. This polynomial is the derivative of a correction function, which approximates the zero function as explained below.

In 1D, the CPR method is equivalent to the flux reconstruction approach. The FR approach is a more general and flexible framework, making it very easy to obtain various sets of the coefficients $\alpha_{j,L}$. The FR approach for both convection and diffusion equations is documented by Huynh in [13] and [14]. Therefore, we now discuss (2.24) via the FR approach. Due to the discontinuity at the interfaces of the piecewise solution polynomial, the flux is also a piecewise discontinuous function. The idea of the FR method is to construct a new flux function $\hat{F}_i(x)$, which satisfies the following three criteria:

1. $\hat{F}_i(x)$ is a degree $k+1$ polynomial, i.e., one degree higher than the solution polynomial Q_i;
2. $\hat{F}_i(x)$ approximates $F_i(x)$ in some sense. In other words, some norm of the difference $\|\hat{F}_i(x)-F_i(x)\|$ is minimized;
3. At both ends of the element, the flux takes on the value of the common fluxes, i.e.,
$$\begin{aligned}\hat{F}_i(x_{i-1/2})&=F_{\rm com}(Q_{i-1}(x_{i-1/2}),Q_i(x_{i-1/2}))\equiv F_{{\rm com},i-1/2},\\ \hat{F}_i(x_{i+1/2})&=F_{\rm com}(Q_i(x_{i+1/2}),Q_{i+1}(x_{i+1/2}))\equiv F_{{\rm com},i+1/2},\end{aligned}$$
where, at each interface, $F_{\rm com}(Q^-,Q^+)$ is a common interface flux obtained from the two values Q^- and Q^+ to the left and right of that interface.

For the inviscid flux, the Riemann or upwind flux is employed. For the diffusion or viscous flux, the common quantities will be discussed in Section 4. Once this flux function $\hat{F}_i(x)$ is found, the DOFs are updated using the following differential equation

$$\frac{\partial Q_{i,j}}{\partial t}+\frac{\partial \hat{F}_i(x_{i,j})}{\partial x}=0. \tag{2.25}$$

Next, the reconstructed flux function is re-written as

$$\hat{F}_i(x)=F_i(x)+\sigma_i(x), \tag{2.26}$$

where $\sigma_i(x)$ approximates the zero function. The function $\sigma_i(x)$ is further expressed as

$$\sigma_i(x)=[F_{{\rm com},i-1/2}-F_i(x_{i-1/2})]g_L(x)+[F_{{\rm com},i+1/2}-F_i(x_{i+1/2})]g_R(x), \tag{2.27}$$

where $g_L(x)$ and $g_R(x)$ are called correction functions. They are of degree $k+1$, approximate the zero function, and satisfy

$$g_L(x_{i-1/2})=1, \qquad g_L(x_{i+1/2})=0 \tag{2.28a}$$
$$g_R(x_{i-1/2})=0, \qquad g_R(x_{i+1/2})=1. \tag{2.28b}$$

For g_L, the condition $g_L(x_{i-1/2})=1$ deals with the jump at the left interface, and the condition $g_L(x_{i+1/2})=0$ leaves the right interface value unchanged. Eq. (2.24) then becomes

$$\begin{aligned}\frac{\partial Q_{i,j}}{\partial t}+\frac{\partial F_i(x_{i,j})}{\partial x}&+[F_{{\rm com},i-1/2}-F_i(x_{i-1/2})]g_L'(x_{i,j})\\ &+[F_{{\rm com},i+1/2}-F_i(x_{i+1/2})]g_R'(x_{i,j})=0.\end{aligned} \tag{2.29}$$

Due to symmetry, we only need to consider $g_L(x)$, or simply $g(x)$. Using a linear transformation, it suffices to consider the correction function g on the standard element $[-1,1]$. Since g is a polynomial of degree $k+1$ where $k=K-1$, it is defined by $k+2$ conditions. Two conditions are given by the first half of (2.28), namely $g(-1)=1$ and $g(1)=0$. Thus, k conditions remain. The requirement of approximating the zero function can be satisfied by the condition that g is orthogonal to P^{k-1}, i.e.,

$$\int_{-1}^{1} g(\xi)\xi^m d\xi = 0, \qquad m=0,1,\cdots,k-1. \tag{2.30}$$

Such a correction function is the Radau polynomial, and (2.24) results in a differential formulation of the DG method. For other correction functions, see [13,14].

CPR for quadrilateral elements

The 1D CPR formulation can be extended to quadrilateral cells by tensor products. First, the mapping from the standard square $[-1,1]\times[-1,1]$ to a quadrilateral element (Fig. 3) is given by

$$\begin{pmatrix} x \\ y \end{pmatrix} = \sum_{j=1}^{N_n} M_j(\xi,\eta) \begin{pmatrix} x_j \\ y_j \end{pmatrix}, \tag{2.31}$$

where x and y are the physical coordinates, ξ and η the local coordinates, (x_j, y_j) are the control points of the mapping (the four corners for straight-edge elements). N_n is the number of nodes used to define the physical element, $N_n=4$ for elements with straight edges, $N_n>4$ for elements with one or more curved edges. For each j, $M_j(\xi,\eta)$ is the shape function, which takes on the value 1 at node j and 0 at all other nodes. The Jacobian of the formulation is

$$J = \left| \frac{\partial(x,y)}{\partial(\xi,\eta)} \right| = \begin{vmatrix} x_\xi & x_\eta \\ y_\xi & y_\eta \end{vmatrix}. \tag{2.32}$$

Assuming that the transformation is nonsingular, the metrics can be computed by

$$\xi_x = y_\eta / J, \qquad \xi_y = -x_\eta / J, \qquad \eta_x = -y_\xi / J, \qquad \eta_y = x_\xi / J. \tag{2.33}$$

On the computational domain, the governing equation takes the form

$$\frac{\partial \tilde{Q}}{\partial t} + \frac{\partial \tilde{F}}{\partial \xi} + \frac{\partial \tilde{G}}{\partial \eta} = 0, \tag{2.34}$$

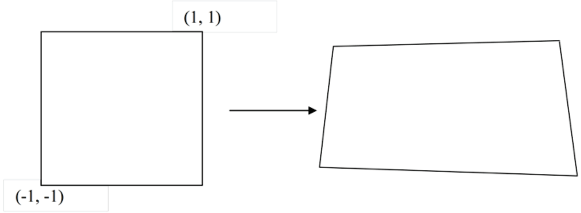


Figure 3: Transformation of a quadrilateral element to a standard element.

where

$$\tilde{Q}=J\cdot Q, \tag{2.35a}$$
$$\tilde{F}=J(\xi_x F+\xi_y G), \tag{2.35b}$$
$$\tilde{G}=J(\eta_x F+\eta_y G). \tag{2.35c}$$

Next, set $\vec{S}_\xi=J(\xi_x,\xi_y)$, $\vec{S}_\eta=J(\eta_x,\eta_y)$. Then we have $\tilde{F}=\vec{F}\bullet\vec{S}_\xi$, $\tilde{G}=\vec{F}\bullet\vec{S}_\eta$. In our implementation, J and $\vec{S}_\xi$, $\vec{S}_\eta$ are stored at the solution points. Within the i-th element, the solution polynomial is a tensor product of 1D Lagrange polynomials, i.e.,

$$Q_i(\xi,\eta)=\sum_{m=1}^{k+1}\sum_{j=1}^{k+1}Q_{i;j,m}L_j(\xi)L_m(\eta). \tag{2.36}$$

There are $K=(k+1)^2$ solution points inside each element. Here $Q_{i;j,m}$ are the state variables at the solution point (j,m) of cell i, for clarity we use ";" to separate the cell index i from the two indices "j,m" of the solution point, with j the index in ξ direction, and m, η direction; $L_j(\xi)$ and $L_m(\eta)$ are 1D Lagrange polynomials in ξ and η directions, respectively. The fluxes $\tilde{F}_{i;j,m}$ and $\tilde{G}_{i;j,m}$ can be calculated by using $\tilde{F}(Q_{i;j,m})$, $\tilde{G}(Q_{i;j,m})$. The fluxes can be represented with Lagrange interpolation polynomials in the following form:

$$\tilde{F}_i(\xi,\eta)=\sum_{m=1}^{k+1}\sum_{j=1}^{k+1}\tilde{F}_{i;j,m}L_j(\xi)L_m(\eta), \tag{2.37a}$$
$$\tilde{G}_i(\xi,\eta)=\sum_{m=1}^{k+1}\sum_{j=1}^{k+1}\tilde{G}_{i;j,m}L_j(\xi)L_m(\eta). \tag{2.37b}$$

Common normal fluxes at the interfaces are computed at all four element edges as follows:

$$\tilde{F}_{\rm com}(-1,\eta)=-F^n_{\rm com}(Q_i(-1,\eta),Q_{i+}(-1,\eta),\vec{n})\left|\vec{S}_\xi\right|, \tag{2.38a}$$
$$\tilde{F}_{\rm com}(1,\eta)=F^n_{\rm com}(Q_i(1,\eta),Q_{i+}(1,\eta),\vec{n})\left|\vec{S}_\xi\right|. \tag{2.38b}$$

The equations for $\tilde{G}$ is similar. Finally the DOFs are updated using the following equation

$$\begin{aligned}&\frac{\partial\tilde{Q}_{i;j,m}}{\partial t}+\frac{\partial\tilde{F}_i(\xi_{j,m},\eta_{j,m})}{\partial\xi}+\frac{\partial\tilde{G}_i(\xi_{j,m},\eta_{j,m})}{\partial\eta}\\&\quad+[\tilde{F}_{\rm com}(-1,\eta_{j,m})-\tilde{F}_i(-1,\eta_{j,m})]g'_L(\xi_{j,m})+[\tilde{F}_{\rm com}(1,\eta_{j,m})-\tilde{F}_i(1,\eta_{j,m})]g'_R(\xi_{j,m})\\&\quad+[\tilde{G}_{\rm com}(\xi_{j,m},-1)-\tilde{G}_i(\xi_{j,m},-1)]g'_L(\eta_{j,m})+[\tilde{G}_{\rm com}(\xi_{j,m},1)-\tilde{G}_i(\xi_{j,m},1)]g'_R(\eta_{j,m})\\&=0.\end{aligned} \tag{2.39}$$

Note that the correction is done in a 1D manner.

CPR for triangular elements

The CPR method for triangular elements is computationally efficient: the coefficients $\alpha_{j,f,l}$ are not only independent of the solution, but also independent of the shape of the element. This means once the distribution of the solution points is chosen, the coefficients $\alpha_{j,f,l}$ can be computed analytically on a standard triangle by Eq. (2.20), and then applied to an arbitrary triangular element. The same comment holds for tetrahedral elements in 3D. The value of coefficients $\alpha_{j,f,l}$ for specific sets of solution points can be found in [28].

For the case of elements with curved boundaries, they are first mapped to a standard triangle, and then CPR coefficients are applied to the transformed governing equation in conservative form, i.e., Eq. (2.34). Therefore, the same set of α coefficients can be used.

3 Compact schemes for discretization of the diffusion/viscous term

3.1 Basic framework

We now use the CPR formulation discussed above to discretize the diffusion term. This discretization includes two parts: calculate the divergence of flux $\vec{\nabla}\bullet\vec{F}$ on the interior of the element and evaluate the common flux at the interface.

First, following [2], we introduce a new variable $\vec{R}$:

$$\vec{R}=\vec{\nabla}Q. \tag{3.1}$$

Eq. (3.1) is solved using the weak formulation. In the CPR framework, the result is a collocation formulation:

$$\vec{R}_{i,j}=\left(\vec{\nabla}Q_i\right)_j+\frac{1}{|V_i|}\sum_{f\in\partial V_i}\sum_l \alpha_{j,f,l}[Q^{\text{com}}-Q_i]_{f,l}\vec{n}_f S_f, \tag{3.2}$$

where $Q^{\text{com}}_{f,l}$ is the common solution on interface f, and $Q_{i;f,l}$ is the solution within cell i on face f. The definition of $Q^{\text{com}}=Q^{\text{com}}(Q^-,Q^+,\vec{n})$ varies with different approaches to discretize the diffusion terms. Here, with uniquely defined left and right side for each interface, Q^- and $Q^+$are the left and right solutions, and $\vec{n}$ the unit normal from left to right.

Next, the viscous fluxes at solution points are evaluated by

$$F^v_{i,j}=F^v(Q_{i,j},\vec{R}_{i,j}),\qquad G^v_{i,j}=G^v(Q_{i,j},\vec{R}_{i,j}). \tag{3.3}$$

Then, $\partial F^v/\partial x$ and $\partial G^v/\partial y$ can be obtained (using Lagrange polynomial). The divergence of the flux namely $\vec{\nabla}\bullet\vec{F}$ on the interior of the element can be calculated.

Besides those interior derivatives, common diffusion/viscous fluxes at the interfaces are also needed for the CPR formulation

$$F^{\text{com}}_{f,l}=F^v\left(Q^{\text{com}}_{f,l},\vec{\nabla}Q^{\text{com}}_{f,l}\right),\qquad G^{\text{com}}_{f,l}=G^v\left(Q^{\text{com}}_{f,l},\vec{\nabla}Q^{\text{com}}_{f,l}\right). \tag{3.4}$$

This means we also need to define a common gradient

$$\vec{\nabla}Q^{\text{com}}=\vec{\nabla}Q^{\text{com}}\left(\vec{\nabla}Q^{-},\vec{\nabla}Q^{+},Q^{-},Q^{+},\vec{n}\right) \tag{3.5}$$

at the cell interfaces.

In the following subsections, we define Q^{com} and $\vec{\nabla}Q^{\text{com}}$ by the BR2, I-continuous, interior penalty and CDG schemes. Let the face f be fixed. We focus our attention on this face and the cells on both sides.

3.2 BR2

The common solution in BR2 is simply the average of solutions at two sides of the face

$$Q_{f,l}^{\text{com}}=\frac{Q_{f,l}^{-}+Q_{f,l}^{+}}{2}. \tag{3.6}$$

For the common gradient,

$$\vec{\nabla}Q_{f,l}^{\text{com}}=\frac{1}{2}\left(\vec{\nabla}Q_{f,l}^{-}+\vec{r}_{f,l}^{-}+\vec{\nabla}Q_{f,l}^{+}+\vec{r}_{f,l}^{+}\right) \tag{3.7}$$

where $\vec{\nabla}Q_{f,l}^{-}$ and $\vec{\nabla}Q_{f,l}^{+}$ are the gradients of the solution of the left and right cells with no correction, while $\vec{r}_{f,l}^{-}$ and $\vec{r}_{f,l}^{+}$ are the corrections to the gradients due to the common the solution on only the face f. More precisely,

$$\vec{r}_{f,l}^{-}=\frac{1}{|V^{-}|}\sum_{m=1}^{N_{\text{fp}}}\beta_{l,m}^{-}[Q^{\text{com}}-Q^{-}]_{f,m}\vec{n}_{f}S_{f}, \tag{3.8a}$$

$$\vec{r}_{f,l}^{+}=\frac{1}{|V^{+}|}\sum_{m=1}^{N_{\text{fp}}}\beta_{l,m}^{+}[Q^{\text{com}}-Q^{+}]_{f,m}\left(-\vec{n}_{f}\right)S_{f}, \tag{3.8b}$$

where N_{fp} is the number of solution points on face f, $\beta_{l,m}$ is the coefficient of correction due to face f. Note that the indices l and m vary on the face f and, for our choice of solution points, $\beta_{l,m}=\alpha_{j,f,m}$, where the index j for the solution points corresponds to the index l for the flux point on the face f. For triangular elements, $\beta_{l,m}$ are identical for any face f with a fixed distribution of flux points.

For quadrilateral elements, because the tensor product is used, $\vec{r}_{f,l}^{-}$ and $\vec{r}_{f,l}^{+}$are thus computed in a 1D manner, which means the penalty matrix $\beta_{l,m}$ reduces to a diagonal matrix. Therefore, for a quadrilateral element, we only need to consider the coefficient $\gamma_l=\beta_{l,l}S_f/V$. Next, denote by $\tilde{\gamma}_l$ the quantity γ_l on the standard element $[-1,1]\times[1,1]$. Then $\tilde{\gamma}_l=-g'(-1)$. This is also of the same form as the diffusion discretization of the FR method [14]. However, for a general quadrilateral element, $\tilde{\gamma}_l$ needs to be transformed back to γ_l in the physical domain (x,y) before being inserted to Eq. (3.8). Depending on the direction (ξ or η) we are dealing with, $\tilde{\gamma}_l$ and γ_l can be related by

$$\gamma_l = \sqrt{\xi_x^2+\xi_y^2}\tilde{\gamma}_l = -\sqrt{\xi_x^2+\xi_y^2} g'(-1) \tag{3.9a}$$

or

$$\gamma_l = \sqrt{\eta_x^2+\eta_y^2}\tilde{\gamma}_l = -\sqrt{\eta_x^2+\eta_y^2} g'(-1). \tag{3.9b}$$

According to the Fourier analysis by Huynh [14], the derivative of Legendre polynomial $g_{LE}'(-1)$ offers the best accuracy for the linear diffusion equation, and therefore $g_{LE}'(-1)$ (the value is $-k(k+1)/2$ for degree k polynomial) is used in all the linear diffusion numerical tests presented later. For the N-S equations, however, a slightly larger coefficient $-(k+1)(k+2)/2$ is used to stabilize the scheme. The same coefficient is also used in the other 3 diffusion schemes.

3.3 I-continuous

The I-continuous approach in 1D was proposed by Huynh (2009) [14]. Its basic idea is: instead of prescribing a common solution Q^{com} at the interfaces, Q^{com} is an unknown to be solved by the condition that the corrected derivatives $\partial Q^C/\partial x$ is continuous across the interface f. Once we obtain Q^{com}, the common derivative $\partial Q^{\text{com}}/\partial x$ is also defined, since there is only one value at the interface.

In the 2D case, requiring the corrected gradient $\vec{\nabla} Q^C$ to be continuous at the interfaces would give us two conditions-continuity in x and y directions. Since we are only solving one variable Q^{com}, we require continuity only in the normal direction. The corrected gradients on the left $(-)$ and right $(+)$ can be expressed as

$$\vec{\nabla} Q_{f,l}^{C-} = \vec{\nabla} Q_{f,l}^{-} + \frac{S_f^-}{|V^-|}\sum_{m=1}^{N_{\text{fp}}} [Q^{\text{com}} - Q^-]_{f,m}\beta_{l,m}^-\vec{n}, \tag{3.10a}$$

$$\vec{\nabla} Q_{f,l}^{C+} = \vec{\nabla} Q_{f,l}^{+} + \frac{S_f^+}{|V^+|}\sum_{m=1}^{N_{\text{fp}}} [Q^{\text{com}} - Q^+]_{f,m}\beta_{l,m}^+(-\vec{n}). \tag{3.10b}$$

Then we require the gradient to be continuous in the normal direction

$$\nabla Q^{C-}\cdot\vec{n} = \nabla Q^{C+}\cdot\vec{n}. \tag{3.11}$$

Substituting in (3.10), we have

$$\begin{aligned}&\sum_{m=1}^{N_{\text{fp}}}\left(\frac{\beta_{l,m}^- S_f^-}{|V^-|}+\frac{\beta_{l,m}^+ S_f^+}{|V^+|}\right)Q_{f,m}^{\text{com}}\\ =&\vec{\nabla} Q_{f,l}^{+}\cdot\vec{n} - \vec{\nabla} Q_{f,l}^{-}\cdot\vec{n} + \sum_{m=1}^{N_{\text{fp}}}\left(\frac{Q_{f,m}^-\beta_{l,m}^- S_f^-}{|V^-|}+\frac{Q_{f,m}^+\beta_{l,m}^+ S_f^+}{|V^+|}\right).\end{aligned} \tag{3.12}$$

Eq. (3.12) represents a linear system, from which $Q_{f,l}^{\text{com}}$ can be easily solved. Then, the common viscous flux $\vec{\nabla}Q^{\text{com}}$ is obtained by

$$\vec{\nabla}Q_{f,l}^{\text{com}}\cdot\vec{n}=\vec{\nabla}Q^{C-}\cdot\vec{n} \qquad \text{or} \qquad \vec{\nabla}Q_{f,l}^{\text{com}}\cdot\vec{n}=\vec{\nabla}Q^{C+}\cdot\vec{n}, \tag{3.13a}$$

$$\vec{\nabla}Q_{f,l}^{\text{com}}\cdot\vec{t}=\frac{\vec{\nabla}Q_{f,l}^{-}\cdot\vec{t}+\vec{\nabla}Q_{f,r}^{+}\cdot\vec{t}}{2}. \tag{3.13b}$$

Here $\vec{t}$ is the unit vector tangential to the face. The direction of $\vec{t}$ makes no difference provided that same $\vec{t}$ is used for both sides. In the case that one or both sides of the face is a quadrilateral cell, we can use $\gamma_l=\beta_{l,l}S_f/V$ in (3.12).

While the I-continuous approach is more involved than the other three schemes, it was shown in [14] via Fourier analysis that if g_{Le} is employed as the correction function, there is a significant gain in accuracy. Since such a Legendre-polynomial correction function is not yet known on triangles, we use the equivalence of the Radau polynomial.

Note that we need to solve a $k+1$ linear system for each face. The cost of this step is minimal since the matrices are independent of the solution. They only need to be inverted once during initialization. Therefore, the I-continuous approach is only slightly costlier than the BR2 approach.

3.4 Interior penalty

Interior penalty is a simplified version of BR2 for triangular meshes, and is identical to BR2 for quadrilateral meshes. In BR2 the correction (or penalty) $\vec{r}_{f,l}^{+}$ and $\vec{r}_{f,l}^{-}$ at one face flux point is a linear combination of the solution differences of all points on the face. In interior penalty method, the penalty is only dependent on the solution difference at that point, i.e. the penalty is computed in a 1-D manner. Therefore, Eqs. (3.6) and (3.7) still holds for interior penalty method, the only difference lies in the computation of the corrections due to the common solution,

$$\vec{r}_{f,l}^{-}=-\left[Q^{\text{com}}-Q^{-}\right]_{f,l}g'(-1)\left(\frac{S_f}{V}\right)^{-}\vec{n}, \tag{3.14a}$$

$$\vec{r}_{f,l}^{+}=\left[Q^{\text{com}}-Q^{+}\right]_{f,l}g'(-1)\left(\frac{S_f}{V}\right)^{+}\vec{n}, \tag{3.14b}$$

$\frac{S_f}{V}\vec{n}$ here is in the place of terms like (ξ_x,ξ_y) for the quadrilateral mesh, which represents the length scale factor.

3.5 CDG

The idea of CDG is to use one side of the face for the common solution and the other side for the common gradient.

For example, if we use the right $(+)$ side for common solution and the left $(-)$ side for common gradient, we obtain,

$$Q^{\text{com}}_{f,l} = Q^{+}_{f,l}, \tag{3.15a}$$

$$\vec{\nabla} Q^{\text{com}}_{f,l} = \vec{\nabla} Q^{-}_{f,l} + \vec{r}^{\,-}_{f,l}, \tag{3.15b}$$

where

$$\vec{r}^{\,-}_{f,l} = \frac{1}{|V^-|} \sum_{m=1}^{N_{\text{fp}}} \beta^{-}_{l,m} \left[Q^{\text{com}} - Q^{-} \right]_{f,m} \vec{n}_f S_f. \tag{3.16}$$

Alternatively, we can also use the opposite sides for common solution and common gradient. Note that the difference between the CDG and LDG methods is in the definition of the common gradient. At a fixed face, for CDG, the correction term $\vec{r}^{\,-}_{f,l}$ in (3.16) involves only the jumps at that face, whereas, for LDG, it involves the jumps at all the faces of the corresponding cell.

3.6 Boundary conditions

In this subsection, we discuss the boundary conditions for the compact diffusion/viscous schemes. Let's assume a face f is on the boundary of the computational domain and the left side of f is the computational domain. Only the solutions and gradients on the left $(-)$ side are available, and the final objective is to obtain Q^{com} and $\vec{\nabla} Q^{\text{com}}$ for face f.

In the linear diffusion cases presented in Section 5, two types of boundary conditions are used. For Dirichlet boundary condition, Q^+ is set to the boundary value while the gradient is extrapolated, i.e., $\nabla Q^+ = \nabla Q^-$. Then the common quantities can be computed according to Sections 3.2-3.5. For Neumann boundary conditions, the common solution is extrapolated $Q^{\text{com}} = Q^-$, while ∇Q^{com} is set to the boundary value.

In the case of Navier-Stokes equations, the boundary condition is similar to the Dirichlet boundary condition for the linear case. Q^+ can be simply fixed to the exact boundary values, or obtained through characteristic boundary conditions. For solid non-slipping walls, Q^+ is set to Q^- with reversed velocity, resulting zero velocity for Q^{com}. The gradients are always extrapolated from the interior domain, i.e., $\nabla Q^+ = \nabla Q^-$. For adiabatic walls, the normal temperature gradients are set to 0 when computing the fluxes by Eq. (3.4).

4 Two-dimensional Fourier (Von Neumann) analysis

We now carry out the Fourier (Von Neumann) stability and accuracy analysis of the schemes discussed above on a square and a triangular mesh. It turns out that, for the diffusion equation (below), all discussed schemes have real and nonpositive eigenvalues. This fact implies that these schemes are stable. In addition, the spectral radius is identical to the magnitude of the minimum eigenvalue. In this section, for convenience, we use the term order of accuracy of a scheme to refer to its order of accuracy by Fourier analysis. All schemes discussed have the property of super-accuracy or super-convergence for $k \geq 2$, i.e., the order of accuracy of each scheme by Fourier analysis (but not in general) is higher than the expected order of $k+1$ where k is the degree of the polynomials approximating the solution. In practice, when the mesh is nonuniform and/or the equations are nonlinear, the super-accuracy property no longer holds.

On the domain $(-\infty,\infty)\times(-\infty,\infty)$, consider the diffusion equation

$$u_t = u_{xx} + u_{yy}. \tag{4.1}$$

With initial condition $u_{\text{init}}(x) = e^{I(w_x x + w_y y)}$, where $I = \sqrt{-1}$ (we use i for the cell index), and the wave numbers w_x and w_y lie between $-\pi$ and π. Low frequency data corresponds to wave numbers of small magnitude; high frequency, to those near $\pm\pi$. The exact solution is $u_{\text{exact}}(x) = e^{-(w_x{}^2 + w_y{}^2)t} e^{I(w_x x + w_y y)}$. At $(x,y,t) = (0,0,0)$,

$$(u_{\text{exact}})_t(0,0,0) = -(w_x^2 + w_y^2). \tag{4.2}$$

Square-Mesh Case. The cells are $E_{i,j} = [i,i+1]\times[j,j+1]$. Assume the solution is represented by degree k polynomial. The solution points on each square are formed by using tensor product; thus, there are $K = (k+1)^2$ of them. Denote by $\mathbf{u}_{i,j}$ the column vector of K solution values

$$\mathbf{u}_{i,j} = \left(u_{i,j,1,1}, u_{i,j,1,2}, \cdots, u_{i,j,1,k+1}, u_{i,j,2,1}, \cdots, u_{i,j,2,k+1}, \cdots, u_{i,j,k+1,1}, \cdots, u_{i,j,k+1,k+1}\right)^T, \tag{4.3}$$

where the superscript T denotes the transpose. The following property of the data plays a key role in the calculation of eigenvalues,

$$\mathbf{u}_{i-1,j} = e^{-Iw_x}\mathbf{u}_{i,j} \qquad \text{and} \qquad \mathbf{u}_{i,j-1} = e^{-Iw_y}\mathbf{u}_{i,j}. \tag{4.4}$$

For the schemes of compact stencil discussed here, the solution can be expressed as

$$\frac{d\mathbf{u}_{i,j}}{dt} = \mathbf{C}_{0,0}\mathbf{u}_{i,j} + \mathbf{C}_{-1,0}\mathbf{u}_{i-1,j} + \mathbf{C}_{1,0}\mathbf{u}_{i+1,j} + \mathbf{C}_{0,-1}\mathbf{u}_{i,j-1} + \mathbf{C}_{0,1}\mathbf{u}_{i,j+1}, \tag{4.5}$$

where all $\mathbf{C}_{\alpha,\beta}$ are $K\times K$ matrices. Using (4.4), we replace $\mathbf{u}_{i-1,j}$ by $e^{-Iw_x}\mathbf{u}_{i,j}$, and similarly for all terms of the right hand side above. The result is

$$\frac{d\mathbf{u}_{i,j}}{dt} = \mathbf{S}\mathbf{u}_{i,j}, \tag{4.6}$$

where

$$\mathbf{S} = \mathbf{C}_{0,0} + e^{-\mathbf{I}w_x}\mathbf{C}_{-1,0} + e^{\mathbf{I}w_x}\mathbf{C}_{1,0} + e^{-\mathbf{I}w_y}\mathbf{C}_{0,-1} + e^{\mathbf{I}w_y}\mathbf{C}_{0,1}. \tag{4.7}$$

Here, **S**, which stands for 'space' or 'semidiscrete', is a $K\times K$ matrix with K eigenvalues. The one approximating $-({w_x}^2+{w_y}^2)$ is called the *principal eigenvalue* and is denoted by $S(w_x,w_y)$:

$$S(w_x,w_y)\approx -({w_x}^2+{w_y}^2). \tag{4.8}$$

All other eigenvalues are spurious. For stability, all eigenvalues must have nonpositive real parts.

It turns out that for all methods discussed, via Fourier analysis, the scheme on a square mesh and the corresponding 1D version have the same order of accuracy.

Concerning stability (time-stepping) limits, the method on a square mesh has a limit of 1/2 that of the 1D version. That is, the largest magnitude of all eigenvalues in the case of a square mesh is twice that of the corresponding 1D method. This reduction by a factor of 2 in the stability limit for the 2D diffusion equation is consistent with the reduction by a factor of $\sqrt{2}$ for advection equation in [13,14]: each derivative corresponds to a reduction by $\sqrt{2}$.

The minimum eigenvalues, orders of accuracy, and errors of the schemes on a square mesh are shown in Table 1. In the calculations of order of accuracy, the coarse mesh corresponds to $w_x=\pi/8$ and $w_y=\pi/10$, and the fine mesh, $w_x=\pi/16$ and $w_y=\pi/20$. Next, the principle eigenvalue $S(w_x,w_y)$ is compared to the exact eigenvalue $-(w_x^2+w_y^2)$ to determine the error of the scheme. The errors $E(w)$ and $E(w/2)$ are then evaluated for the wave numbers of the coarse and fine meshes respectively. The order of accuracy is calculated by $\mathrm{Log}(E(w)/E(w/2))/\mathrm{Log}(2)-2$. See, e.g., [14].

Table 1: Minimum eigenvalues, orders of accuracy, and errors for schemes on a square mesh.

Square-mesh case Polynomial degree $k=1$	Min. Eigval.	Ord. Acc.	Coarse mesh error, $w_x=\pi/8$, $w_y=\pi/10$	Fine mesh error, $w_x=\pi/16$, $w_y=\pi/20$
1. BR2	-26.7	1.96	-2.68e-03	1.73e-04
2. I-Continuous	-24.0	2.00	1.39e-03	8.72e-05
3. IP = BR2				
4. CDG	-72.	4.01	8.64e-06	1.34e-07
Polynomial degree $k=2$				
1. BR2	-120.	3.98	2.72e-06	4.29e-08
2. I-Continuous	-84.	4.00	2.76e-06	4.31e-08
3. IP = BR2				
4. CDG	-297.	6.00	5.25e-09	2.06e-11
Polynomial degree $k=3$				
1. BR2	-340.	5.89	-5.94e-09	-2.50e-11
2. I-Continuous	-244.	6.00	2.62e-09	1.02e-11
3. IP = BR2				
4. CDG	-878.	8.00	2.17e-12	2.12e-15

Note that all schemes are of order at least $2k$, i.e., they are super-convergent for $k\geq 2$. The advantage of the CDG scheme is that its accuracy order is $2(k+1)$; its disadvantage is that the minimum eigenvalues have rather large magnitudes (nearly three times) compared to those of the other schemes. Also note that for the square mesh case, the interior penalty (IP) scheme reduces to the BR2 scheme. As for the I-continuous scheme, the one employed here is the member whose triangular-mesh version is readily available, i.e., the member using the correction function $g_{\rm DG}$ not the most accurate member, which employs the Legendre polynomial as correction function. Compared to BR2, the I-continuous scheme has a smaller error as well as a minimum eigenvalue with a smaller magnitude.

Triangular-Mesh Case. Each square $E_{i,j}=[i,i+1]\times[j,j+1]$ is cut into two triangles by the diagonal from the northwest to the southeast corners. To carry out the Fourier analysis for the triangular-mesh case, we must pair up these two triangles so that the solution vector on each square has a repeatable pattern explained below. Recall that if k is the degree of polynomial representing the solution, then the corresponding number of solution points on each triangle is $K=(k+2)(k+1)/2$. Combining the data on two triangles, denote the vector of $2K$ entries on the square E_i,j by $\mathbf{u}_{i,j}$. The data $\mathbf{u}_{i,j}$ is repeatable in the sense that (4.4) holds. The semidiscrete equation is similar to (4.5) and, the corresponding $(2K)^2$ matrix $\mathbf{S}$, (4.7).

The minimum eigenvalues, orders of accuracy, and errors of the schemes on a triangular mesh are shown in Table 2. In the order of accuracy calculations, again, the coarse mesh corresponds to $w_x=\pi/8$ and $w_y=\pi/10$, and the fine mesh, $w_x=\pi/16$ and $w_y=\pi/20$.

Table 2: Minimum eigenvalues, orders of accuracy, and errors for schemes on a triangular mesh.

Triangular-mesh case Polynomial degree $k=1$	Min. Eigval.	Ord. Acc.	Coarse mesh error, $w_x=\pi/8$, $w_y=\pi/10$	Fine mesh error, $w_x=\pi/16$, $w_y=\pi/20$
1. BR2	-66.7	1.97	-5.83e-04	-3.72e-05
2. I-Continuous	- 56.8	2.00	8.90e-04	5.55e-05
3. IP	-60.	2.00	8.45e-04	5.27e-05
4. CDG	-160.	1.84	-1.14e-04	-1.63e-05
Polynomial degree $k=2$				
1. BR2	- 204.	3.99	1.00e-06	1.57e-08
2. I-Continuous	- 203.	4.00	1.64e-06	2.57e-08
3. IP	- 207.	4.01	3.42e-06	5.30e-08
4. CDG	- 565.	3.96	-4.19e-08	-6.74e-10
Polynomial degree $k=3$				
1. BR2	- 522.	5.95	-1.32e-09	-5.33e-12
2. I-Continuous	- 508.	6.00	1.42e-09	5.54e-12
3. IP	- 516.	5.82	-1.47e-09	-6.51e-12
4. CDG	- 1458.	5.98	-4.10e-11	-1.62e-13

Note that all schemes are of order $2k$. For a triangular mesh, the CDG scheme is no longer of order $2(k+1)$. It has the same order of accuracy as the other schemes, but its error is considerably smaller than those of the others. The disadvantage of the CDG scheme is that the minimum eigenvalues have rather large magnitudes (roughly three times) compared to those of the other schemes. Compared to BR2, the I-continuous scheme is slightly less accurate in this case; however, it has the advantage that its minimum eigenvalue has a slightly smaller magnitude.

Comparing Table 2 to 1, i.e., the schemes on a triangular mesh versus those on a square mesh, accuracy on a triangular mesh is better, but stability limit is worse. The reason is perhaps that corresponding to each k, there are more solution points on (the two triangles of) the square in the former case compared to the latter (i.e., $(k+2)(k+1)$ versus $(k+1)^2$).

As noted earlier, in practice, when the mesh is non-uniform and/or the equations are nonlinear, the super-accuracy (or super-convergence) property no longer holds. In fact, even for the case of a uniform mesh and linear equations, since the data is approximated by a polynomial of degree k in each cell, if we simply compare the numerical solution with the exact one, the degree of accuracy is only $k+1$; to observe super-accuracy, we may need to compare special quantities such as the average of the solutions on each square (even when the mesh is triangular). In short, super accuracy results typically do not hold in practice.

5 Numerical results

5.1 Poisson's equation

The following Poisson's equation

$$u_{xx}+u_{yy}=-\frac{\pi^2}{2}\sin\Big(\frac{\pi x}{2}\Big)\sin\Big(\frac{\pi y}{2}\Big), \tag{5.1}$$

with the exact solution $u=\sin(\pi x/2)\sin(\pi y/2)$ is solved on the domain $[0,1]\times[0,1]$ with both regular quadrilateral meshes and irregular triangular meshes (Fig. 4). The solution

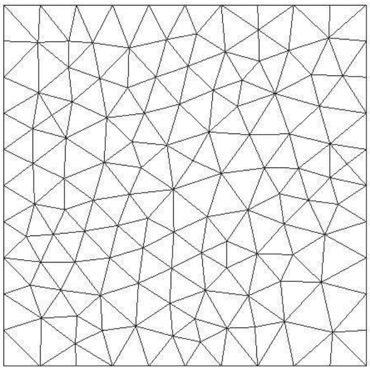


Figure 4: Irregular triangular mesh for Poisson's equation on $[0,1]\times[0,1]$.

variable u is fixed to the exact solution at the boundary faces. The CPR solver starts with the exact solution. The steady state is reached by solving the unsteady equation $u_t = u_{xx}+u_{yy}+\pi^2/2\sin(\pi x/2)\sin(\pi y/2)$. For time marching, we use the 1st order backward Euler scheme with an LU-SGS algorithm [21].

The root mean square of the errors at the solution points (the term "RMS error" is used from here on) is evaluated and the order of accuracy is calculated via an h-refinement by

$$p=\frac{\ln(\mathrm{Er}_1/\mathrm{Er}_2)}{\ln\left(\sqrt{N_2/N_1}\right)}, \tag{5.2}$$

where Er_1 and Er_2 are the RMS errors for mesh 1 and mesh 2, with number of cells N_1 and N_2 respectively. (The error is always calculated in this manner in the rest of the paper).

Table 3 shows the accuracy results of the BR2, I-continuous and CDG approaches on quadrilateral meshes. All schemes produce comparable errors and orders of accuracy, with CDG bested the other two by a small margin.

Table 3: Accuracy results of Poisson's equation with regular quadrilateral mesh (Interior Penalty is identical to BR2).

Polynomial		BR2		I-Continuous		CDG	
Degree	Mesh	RMS error	order	RMS error	Order	RMS error	Order
$k=2$	5×5	1.949e-04	-	1.915e-04	-	1.329e-04	-
	10×10	2.360e-05	3.05	2.337e-05	3.03	1.718e-05	2.95
	20×20	2.891e-06	3.03	2.876e-06	3.02	2.176e-06	2.98
	40×40	3.573e-07	3.02	3.564e-07	3.01	2.735e-07	2.99
$k=3$	5×5	4.396e-06	-	5.179e-06	-	2.738e-06	-
	10×10	2.438e-07	4.17	3.298e-07	3.97	1.777e-07	3.95
	20×20	1.409e-08	4.11	2.080e-08	3.99	1.132e-08	3.97
	40×40	8.408e-10	4.07	1.306e-09	3.99	7.140e-10	3.99
$k=4$	5×5	6.501e-08	-	6.385e-08	-	4.398e-08	-
	10×10	1.981e-09	5.04	1.961e-09	5.03	1.398e-09	4.98
	20×20	6.108e-11	5.02	6.076e-11	5.01	4.398e-11	4.99

Table 4 shows accuracy results for irregular triangular meshes. Here the meshes are generated independently, and for each refinement, the number of elements roughly quadruples. Again, the errors by BR2, I-continuous and interior penalty schemes are comparable, and the CDG error is the best by a small margin.

5.2 The potential flow over a cylinder

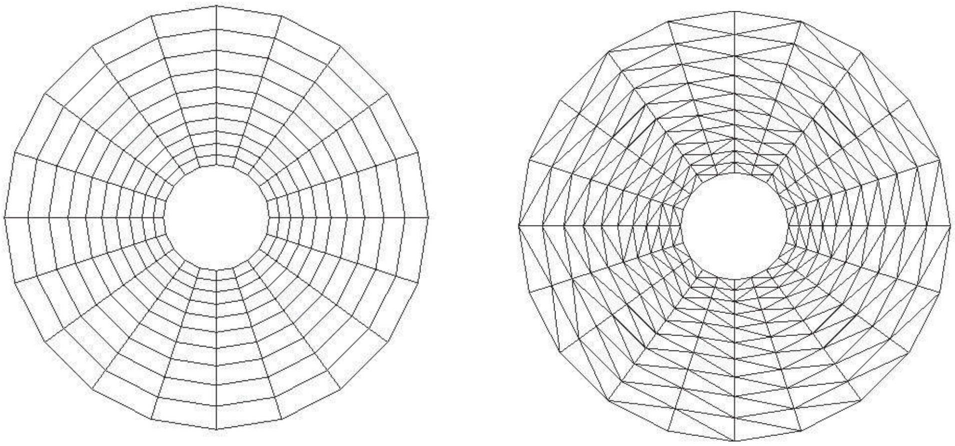
The potential for an incompressible, irrotational and inviscid flow satisfies the Laplace equation (2.1). To test the CPR diffusion solver on curved boundaries, the potential flow over a cylinder with radius $r=1$ is computed. The exact solution for this case is $\varphi_{exact}=x(\frac{1}{x^2+y^2}+1)$.

Table 4: Accuracy results of Poisson's equation with irregular triangular mesh.

Polynomial Degree	Mesh (No. of cells)	BR2		I-Continuous	
		RMS error	order	RMS error	Order
$k=2$	58	2.993e-04	-	2.995e-04	-
	230	4.188e-05	2.85	4.130e-05	2.88
	936	4.590e-06	3.15	4.624e-06	3.12
$k=3$	58	1.196e-05	-	1.238e-05	-
	230	5.887e-07	4.37	6.476e-07	4.28
	936	3.382e-08	4.07	3.722e-08	4.07
Polynomial Degree	Mesh (No. of cells)	Interior Penalty		CDG	
		RMS error	order	RMS error	Order
k=2	58	4.026e-04	-	2.308e-04	-
	230	5.590e-05	2.87	3.343e-05	2.80
	936	6.226e-06	3.13	3.832e-06	3.09
k=3	58	1.509e-05	-	9.185e-06	-
	230	7.428e-07	4.37	4.578e-07	4.35
	936	4.225e-08	4.09	2.812e-08	3.98

The solution variable at the outer boundary faces is fixed to the exact solution. For the inner boundary, the Neumann condition $\vec{\nabla}\varphi\cdot\vec{n}=0$ is imposed. Each curved wall segment is represented by a polynomial of the same degree as that of the solution polynomials. This approximation also leads to a non-linear transformation of the corresponding boundary element. The initial condition is set to the exact solution and an LU-SGS algorithm is used with a convergence criteria of machine zero for all cases.

Fig. 5(a) shows the 10×10 structured quadrilateral O-mesh used in this test case.



(a) Quadrilateral mesh

(b) Triangular mesh

Figure 5: Meshes for potential flow over a cylinder.

Table 5: Accuracy results for a potential flow over a cylinder with quadrilateral meshes (Interior Penalty is identical to BR2).

Polynomial		BR2		I-Continuous		CDG	
Degree	Mesh	RMS error	order	RMS error	Order	RMS error	Order
$k=2$	5×5	3.610e-03	-	3.759e-03	-	3.103e-03	-
	10×10	4.653e-04	2.96	4.721e-04	2.99	4.125e-04	2.91
	20×20	6.061e-05	2.94	6.013e-05	2.97	5.331e-05	2.95
$k=3$	5×5	1.166e-04	-	6.231e-04	-	4.459e-04	-
	10×10	4.131e-06	4.82	3.750e-05	4.05	3.148e-05	3.82
	20×	4.246e-07	3.28	2.394e-06	3.97	2.114e-06	3.90
$k=4$	5×5	8.845e-05	-	9.185e-05	-	7.535e-05	-
	10×10	2.847e-06	4.96	2.873e-06	5.00	2.605e-06	4.85
	20×20	9.264e-08	4.94	9.264e-08	4.95	8.627e-08	4.92

Table 6: Accuracy results of Poisson's equation with irregular triangular mesh.

Polynomial		BR2		I-Continuous	
Degree	Mesh	RMS error	order	RMS error	Order
$k=2$	10×10×2	9.485e-04	-	7.935e-04	-
	20×20×2	9.779e-05	3.28	7.776e-05	3.35
	40×40×2	1.179e-05	3.05	9.404e-06	3.05
$k=3$	10×10×2	6.881e-05	-	7.764e-05	-
	20×20×2	3.944e-06	4.12	3.651e-06	4.41
	40×40×2	2.440e-07	4.01	2.219e-07	4.04
Polynomial		Interior Penalty		CDG	
Degree	Mesh	RMS error	order	RMS error	Order
$k=2$	10×10×2	1.976e-03	-	7.564e-04	-
	20×20×2	1.301e-04	3.92	7.863e-05	3.27
	40×40×2	1.477e-05	3.14	9.229e-06	3.09
$k=3$	10×10×2	9.140e-05	-	5.199e-05	-
	20×20×2	4.746e-06	4.27	3.022e-06	4.10
	40×40×2	2.956e-07	4.00	1.918e-07	3.98

Fig. 5(b) shows the $10\times10\times2$ triangular mesh, which is obtained by cutting each quadrilateral cell into two triangles.

Fig. 6 shows the solution φ by the BR2 method with $k=2$.

Tables 5 and 6 respectively show the numerical results for quadrilateral and triangular meshes. The RMS errors produced by all the schemes seem comparable, and so is the order of accuracy.

Fig. 7 shows the convergence history of the four schemes for $k=3$ on both quadrilateral and triangular meshes. The residuals are plotted against the iterations. Since the cost per iteration of the four approaches is about the same, these figures also represent the CPU time used. For both quadrilateral and triangular meshes, BR2 and interior penalty converge fastest and CDG, slowest.

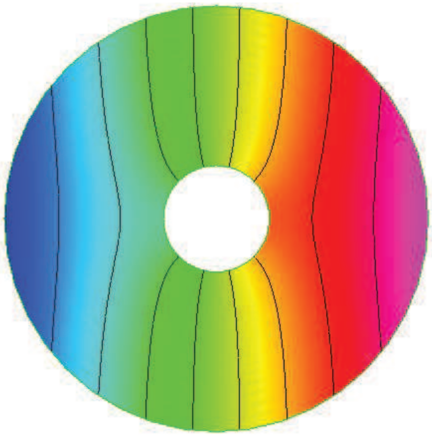
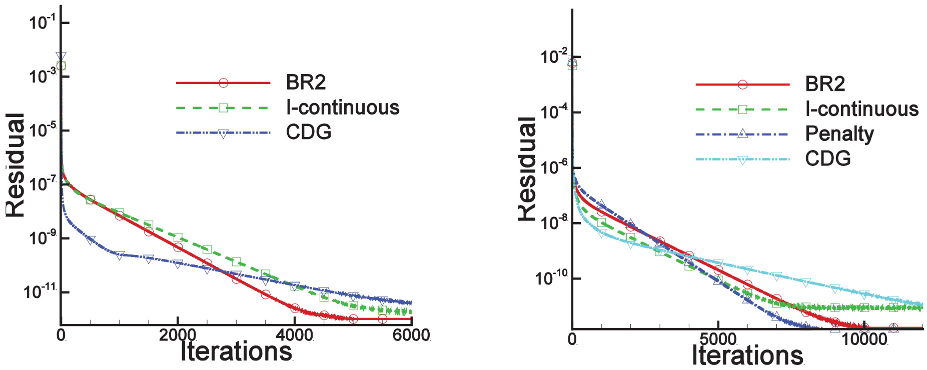


Figure 6: Solution contours for potential flow over a cylinder (BR2, 3rd order, 20×20 mesh).



(a) (b)

Figure 7: Convergence history of potential flow over a cylinder (4th order, 40×40 meshes) (a) quadrilateral mesh (b) triangular mesh.

5.3 The Couette flow

Compressible Couette flow between two parallel walls is used to evaluate the accuracy of the method on irregular mixed grids. The exact solution for this case is

$$u=\frac{U}{H}y, \qquad v=0, \tag{5.3a}$$

$$p=Const, \qquad \rho=\frac{p}{RT}, \tag{5.3b}$$

$$T=T_0+\frac{y}{H}(T_1-T_0)+\frac{\mu U^2}{2k}\frac{y}{H}\left(1-\frac{y}{H}\right). \tag{5.3c}$$

The following parameters are chosen: the speed of the upper wall $U=0.3$, the temperature of the lower wall $T_0=0.8$, the temperature of the upper wall $T_1=0.85$, viscosity

$\mu = 0.01$, domain height (in y direction) $H = 2$, and the computational domain is a 4×2 rectangle.

The flow variables at boundary faces are fixed to the exact solution. All the tests cases presented below are obtained with an LU-SGS algorithm and implicit time integration approach [21], and all cases converge to machine zero. Density errors are used for accuracy evaluation.

Fig. 8 shows the three meshes used in the mesh refinement study. The meshes are generated independently, rather than in an h-refinement manner. The schemes employed are 2$^{\text{nd}}$-4$^{\text{th}}$ order accurate. The order of accuracy is also calculated by again Eq. (5.2).

Table 7 shows accuracy results for the four schemes. Again, all the schemes are of comparable accuracy except for the case $k=2$ for CDG, where the order of accuracy is not quite as high as the others. The accuracy of CDG demonstrated in Fourier analysis and linear diffusion case is not carried over to this Navier-Stokes case. The BR2 and I-continuous approaches perform consistently well, whereas the interior penalty approach produces slightly larger errors for the case $k=1$. Note that although the orders of accuracy of all the schemes are slightly less than optimal, these results are obtained with highly irregular and mixed meshes of poor quality.

Table 7: Accuracy results of Couette flow with hybrid mesh.

Polynomial	Mesh	BR2		I-Continuous	
Degree	(No. of cells)	RMS error	order	RMS error	Order
$k=1$	61	7.261e-05	-	7.112e-05	-
	250	2.312e-05	1.62	2.306e-05	1.60
	919	7.151e-06	1.80	7.162e-06	1.80
$k=2$	61	8.914e-07	-	8.795e-07	-
	250	1.309e-07	2.72	1.400e-07	2.61
	919	2.091e-08	2.82	2.456e-08	2.67
$k=3$	61	8.525e-09	-	8.190e-09	-
	250	6.525e-10	3.64	6.157e-10	3.67
	919	5.785e-11	3.72	5.186e-11	3.80
Polynomial	Mesh	Interior Penalty		CDG	
Degree	(No. of cells)	RMS error	order	RMS error	Order
$k=1$	61	1.007e-04	-	1.019e-04	-
	250	3.421e-05	1.53	3.363e-05	1.57
	919	1.013e-05	1.87	9.714e-06	1.91
$k=2$	61	8.499e-07	-	8.766e-07	-
	250	1.278e-07	2.69	1.696e-07	2.33
	919	2.045e-08	2.82	3.641e-08	2.36
$k=3$	61	9.444e-09	-	8.834e-09	-
	250	7.044e-10	3.68	7.555e-10	3.49
	919	6.158e-11	3.74	7.228e-11	3.61

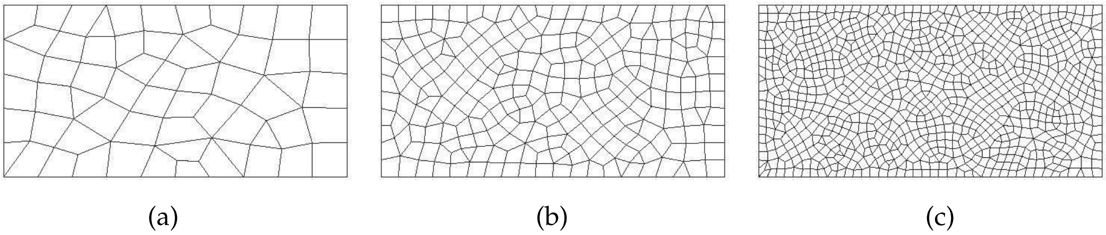


Figure 8: Mixed mesh used for Couette flow (a) 10×5 with 61 cells (b) 20×10 with 250 cells (c) 40×20 with 919 cells.

5.4 Laminar flow around a NACA0012 airfoil

Viscous laminar flow around a NACA 0012 airfoil is simulated with the CPR method, using the BR2, I-continuous, interior penalty and CDG schemes for the viscous flux. 2nd-5th order schemes are tested.

The flow conditions are Mach $=0.5$ and Re $=5000$, with an angle of attack of 1 degree. Under such conditions, steady laminar separations are expected for both upper and lower surfaces of the airfoil. Adiabatic no-slip wall condition is prescribed at the airfoil surface. Subsonic characteristic far field condition is used at the outer surface of the computational domain. The curved wall boundary is represented by the same degree polynomial as the solution. The computational domain extends 20 chord lengths away from the center of the airfoil.

Fig. 9 shows the mesh of 2692 cells, which is composed of regular quadrilateral elements near the airfoil and irregular mixed elements elsewhere, with some refinement at the trailing edge. An LU-SGS algorithm [21] is used for time implicit integration and all cases converge to machine zero.

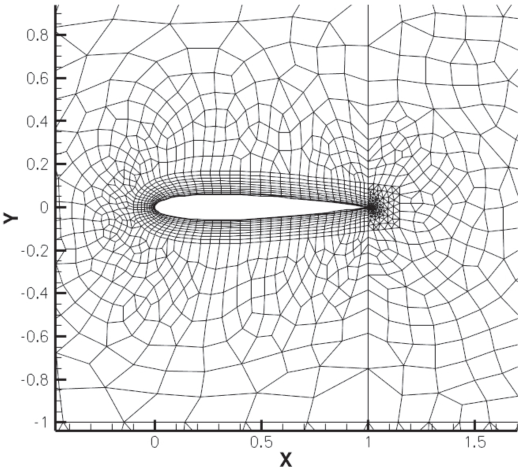


Figure 9: Mixed mesh around an NACA0012 airfoil.

Figs. 10(a-c) shows the computed Mach number contours of 2nd-4th order schemes. Only the BR2 results are shown, since the results of other schemes are very similar. Due to the coarse mesh, the 2nd order results are not smooth, especially at the wake. Note that for the 3rd and 4th order cases, the contour lines are smooth across the interfaces between regular cells and irregular ones and also between triangular cells and quadrilateral ones. Visually, 3rd and 4th order results are nearly identical.

The skin friction coefficient C_f is defined as

$$C_f = \frac{\tau_w}{\frac{1}{2}\rho U_\infty^2}, \tag{5.4}$$

where the shear stress at wall is computed by $|\tau_w| = \mu \partial V / \partial n$, with its sign chosen to be the same as $\partial u / \partial n$.

Fig. 11 shows the C_f distribution near the separation point. For the four diffusion schemes, the 3rd and 4th order results are very close; this fact indicates convergence with p-refinement. Except for the 2nd order case, the four schemes produce almost identical distributions of C_f.

Tables 8-10 show the computed pressure drag coefficients $C_{D,p}$, friction drag coefficients $C_{D,f}$, and the separation points on the upper wall. The data converge as the polynomial degree increases and, again, there is no significant difference among the four schemes considered.

Table 8: Flow over a NACA0012 airfoil — pressure drag coefficients $C_{D,p}$.

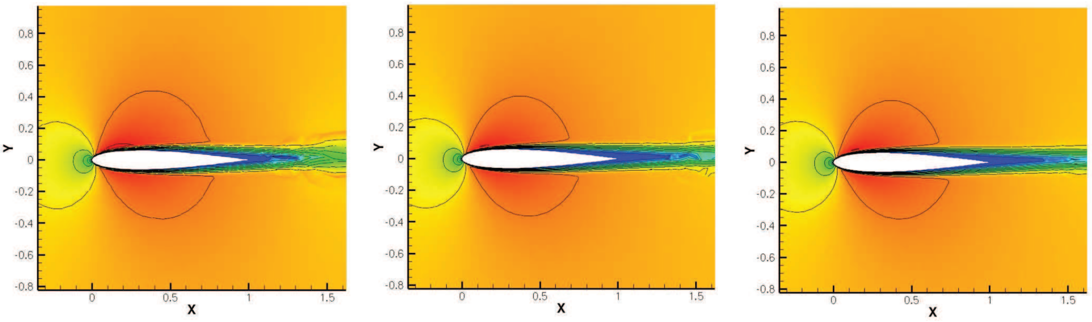
	BR2	I-Continuous	Interior Penalty	CDG
$k=1$	0.02252	0.02233	0.02252	0.02173
$k=2$	0.02275	0.02272	0.02276	0.02259
$k=3$	0.02295	0.02294	0.02295	0.02294
$k=4$	0.02296	0.02295	0.02296	0.02295

Table 9: Flow over a NACA0012 airfoil — friction drag coefficients $C_{D,f}$.

	BR2	I-Continuous	Interior Penalty	CDG
$k=1$	0.03248	0.03289	0.03248	0.03287
$k=2$	0.03271	0.03272	0.03271	0.03277
$k=3$	0.03250	0.03250	0.03251	0.03250
$k=4$	0.03248	0.03250	0.03249	0.03248

Table 10: Flow over a NACA0012 airfoil — separation point on the upper surface.

	BR2	I-Continuous	Interior Penalty	CDG
$k=1$	79.84%	78.30%	79.84%	80.39%
$k=2$	68.67%	68.65%	68.67%	68.67%
$k=3$	67.84%	67.81%	67.84%	67.82%
$k=4$	67.91%	67.90%	67.91%	67.91%



(a) 2nd Order (b) 3rd Order (c) 4th Order

Figure 10: Mach number contours of flow around an NACA 0012 airfoil.

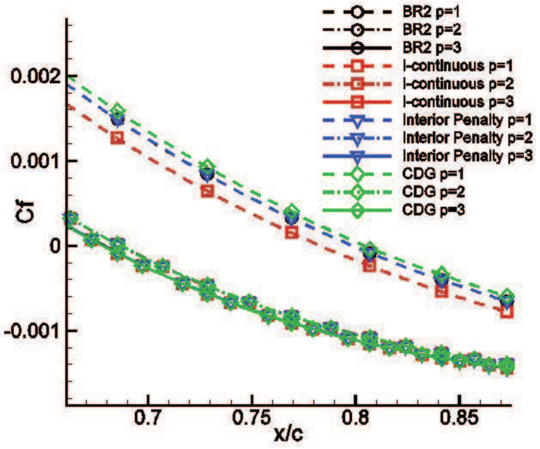
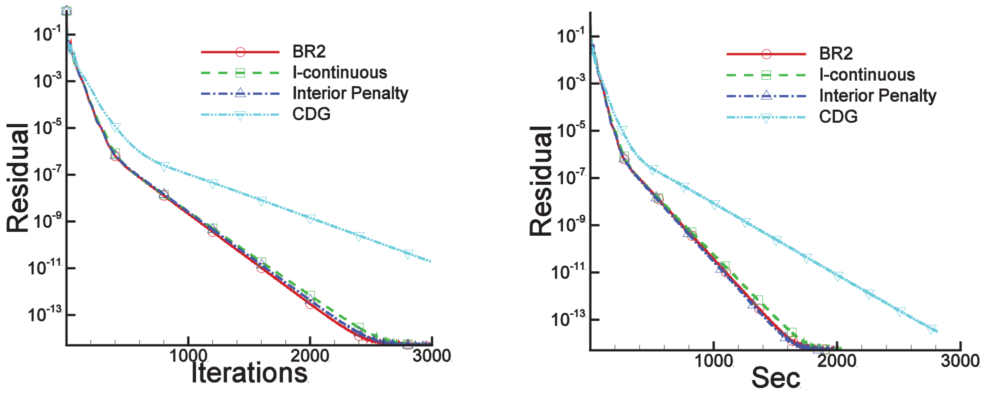


Figure 11: C_f on the upper surface of NACA0012 airfoil for BR2, I-continuous, interior penalty and CDG.



(a) (b)

Figure 12: 4th order convergence history of laminar flow over NACA0012 airfoil in terms of (a) iterations (b) CPU time.

Figs. 12(a, b) show the convergence history of the 4th order case, which provides an estimate of the efficiency of the four diffusion schemes. The same time step, which is also the largest allowable by stability, is used for all schemes. In terms of both iteration and CPU time, CDG is least efficient, while the other three approaches show no significant difference.

5.5 Unsteady flow around a circular cylinder

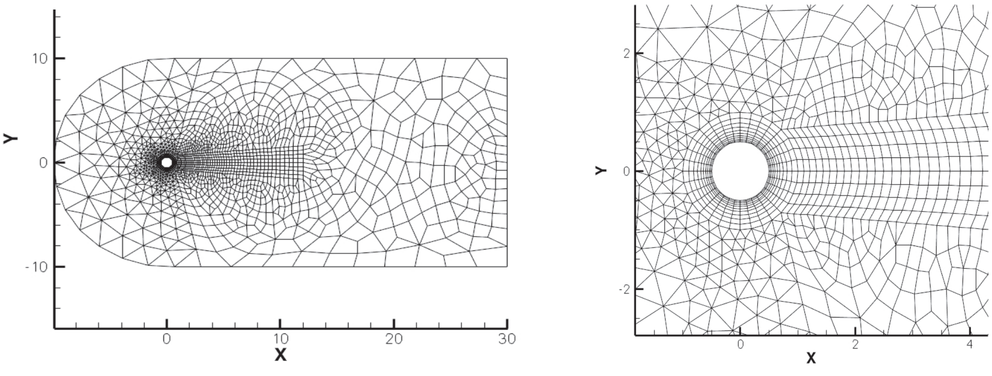
The unsteady laminar flow around a cylinder is simulated with the 2nd-4th order CPR methods using the BR2 diffusion scheme. For time integration, the 2nd order trapezoidal rule is used with an LU-SGS algorithm [21]. The Reynolds number $\mathrm{Re}=75$, the free stream Mach number $\mathrm{M}=0.2$. A vortex street is expected to form in the wake of the cylinder. The frequency of the vortex shedding is often called the Strouhal number defined by

$$St=\frac{f_sL_c}{U_\infty}. \tag{5.5}$$

The length scale L_c here is the diameter of the cylinder. In an experimental study by Williamson [29], St was found to be 0.148.

Subsonic far field boundary condition is used at the outer boundary of the domain. Adiabatic wall condition is used for the cylinder surface.

Fig. 13 shows the mesh, which contains 2,028 cells, with regular quadrilateral cells near the cylinder and in the wake region, and irregular mixed cells elsewhere. The effects of domain size, time step and convergence criteria of inner iterations are carefully studied to make sure the correct numerical solution is obtained.

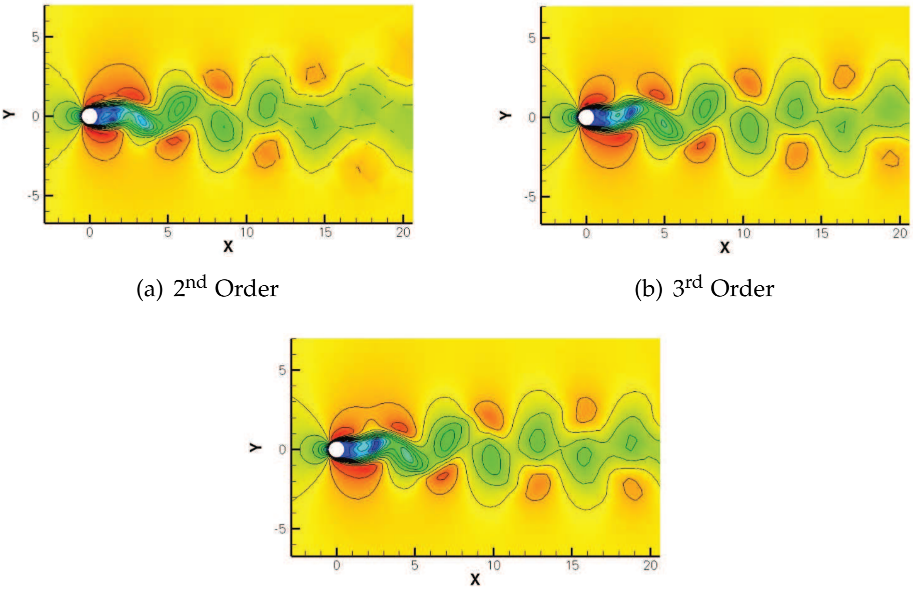


(a) The whole domain

(b) Near the cylinder

Figure 13: Mixed mesh for unsteady flow around a cylinder.

Figs. 14(a-c) shows the instantaneous Mach number contours. Note that the 2nd order contours in (a) are not smooth. The 3rd and 4th order results are smooth and similar, with 4th order contours smoother at 20 diameters downstream.



(a) 2nd Order

(b) 3rd Order

(c) 4th Order

Figure 14: Mach number contours for unsteady flow around a cylinder.

Table 11 shows the average drag coefficient $\overline{C_D}$ and computed St for $k=1,2,3$. Both converge as p increases, and the computed St is comparable to the experimental data.

Table 11: P-refinement study for unsteady flow around a cylinder.

Viscous Method – Polynomial Degree	$\overline{C_D}$	St
BR2–1	1.4299	0.153
BR2–2	1.4365	0.153
BR2–3	1.4364	0.153

6 Conclusion

The CPR formulation, which has the advantage of simplicity and economy, is extended to the diffusion and Navier-stokes equations on hybrid meshes of triangles and quadrilaterals. The four schemes of compact stencil employed to discretize the diffusion terms are BR2, interior penalty, compact DG (CDG), and I-continuous. The first three of these schemes, originally derived using the integral formulation, were recast here in the CPR framework, whereas the I-continuous scheme, originally derived for a quadrilateral mesh, was extended to a triangular mesh. Fourier stability and accuracy analyses on quadrilateral and triangular meshes were carried out. Numerical tests varying from linear diffusion to steady and unsteady laminar flows were conducted.

The results show that the CPR approach produces highly accurate schemes of expected order of accuracy, i.e., schemes using polynomial of degree k is accurate to order $k+1$. Concerning methods for diffusion, the four compact schemes are comparable in terms of accuracy. In terms of convergence rate, however, the CDG scheme is slowest: this finding by numerical experiments is consistent with the finding via Fourier analysis. Among the four schemes, due to its simplicity, BR2 is an optimal choice; while I-continuous approach is promising, a more accurate triangular-mesh version must be found for the scheme to be competitive; the interior penalty approach is the simplest, but is also slightly less accurate. Finally, the CPR formulation presented here can be extended to 3D and, due to its hybrid-mesh capability, the resulting methods can be applied to practical problems involving complex geometries.

Acknowledgments

The first two authors were funded by AFOSR grant FA9550-06-1-0146. The third author is supported by NASA's Fundamental Aeronautics Program. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of AFOSR, NASA or the U.S. Government.

References

[1] D.N. Arnold, F. Brezzi, B. Cockburn and L.D. Marini, Unified analysis of discontinuous Galerkin methods for elliptic problems, SIAM J. Numer. Anal., 19(4), 742-760 (2002).

[2] F. Bassi and S. Rebay, A high-order accurate discontinuous finite element method for the numerical solution of the compressible Navier-Stokes equations, J. Comput. Phys., 131(1), 267-79 (1997).

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

[4] F. Bassi, S. Rebay, GMRES discontinuous Galerkin solution of the compressible Navier-Stokes equations, in: B. Cockburn, G.E. Karniadakis and C.-W. Shu (Eds.), Discontinuous Galerkin Methods: Theory, Computation and Applications, Springer, Berlin, 2000, pp. 197-208.

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

[6] 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, 90-113 (1989).

[7] B. Cockburn and C.-W. Shu, The Runge-Kutta discontinuous Garlerkin method for conservation laws V: Multidimensional systems, J. Comput. Phys., 141, 199-224 (1998).

[8] B. Cockburn and C.-W. Shu, The local discontinuous Galerkin methods for time-dependent convection diffusion systems, SIAM J. Numer. Anal., 35, 2440-2463 (1998).

[9] V. Dolejší, On the discontinuous Galerkin method for numerical solution of the Navier-Stokes equations, Int. J. Numer. Methods Fluids, 45, 1083-1106 (2004).

[10] H. Gao and Z.J. Wang, A high-order lifting collocation penalty formulation for the Navier-Stokes equations on 2-D mixed grids, AIAA Paper 2009-3784.

[11] T. Haga, H. Gao and Z.J. Wang, A high-order unifying discontinuous formulation for 3-D mixed grids, AIAA Paper 2010-540.

[12] R. Hartmann and P. Houston, Symmetric interior penalty DG methods for the incompressible Navier-Stokes Equations I: Method formulation, Int. J. Numer. Anal. Model., 3(1), 1-20 (2006).

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

[14] H.T. Huynh, A reconstruction approach to high-order schemes including discontinuous Galerkin for diffusion, AIAA Paper 2009-403.

[15] D.A. Kopriva and J.H. Kolias, A conservative staggered-grid Chebyshev multidomain method for compressible flows, J. Comput. Phys., 125, 244-261 (1996).

[16] C. Liang, A. Jameson and Z.J. Wang, Spectral difference method for two-dimensional compressible flow on unstructured grids with mixed elements, J. Comput. Phys., 228, 2847-2858 (2009).

[17] Y. Liu, M. Vinokur and Z.J. Wang, Discontinuous spectral difference method for conservation laws on unstructured grids, in Proceedings of the 3rd International Conference on Computational Fluid Dynamics, Toronto, Canada, July 12-16 2004.

[18] Y. Liu, M. Vinokur and Z.J. Wang, Spectral difference method for unstructured grids I: Basic formulation, J. Comput. Phys., 216, 780-801 (2006).

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

[20] J. Peraire and P.-O. Persson, The compact discontinuous Galerkin (CDG) method for elliptic problems, SIAM J. Sci. Comput., 30(4), 1806-1824 (2008).

[21] Y. Sun, Z.J. Wang and Y. Liu, Efficient implicit non-linear LU-SGS approach for compressible flow computation using high-order spectral difference method, Commun. Comput. Phys., 5(2-4), 760-778 (2009).

[22] K. Van den Abeele, C. Lacor and Z.J. Wang, On the stability and accuracy of the spectral difference method, J. Sci. Comput., 37(2), 162-188 (2008).

[23] B. van Leer and S. Nomura, Discontinuous Galerkin for diffusion, AIAA Paper 2005-5108.

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

[25] Z.J. Wang, High-order methods for the Euler and Navier-Stokes equations on unstructured grids, J. Progress in Aerospace Sciences, 43, 1-47 (2007).

[26] 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(1), 137-157 (2004).

[27] Z.J. Wang, L. 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(2), 716-741 (2004).

[28] Z.J. Wang and H. 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, 8161-8186 (2009).

[29] C.H.K. Williamson, Oblique and parallel modes of vortex shedding in the wake of a cylinder at low Reynolds number, J. Fluid Mech., 206, 579-627 (1989).