



54th AIAA Aerospace Sciences Meeting, 4–8 January 2016, San Diego, CA

# Discontinuous Galerkin for Advection with Interface-Centered Reconstruction

Loc Khieu<sup>\*,†</sup>

*Vietnamese-German University—VGU, Binh Duong, Vietnam*

Krzysztof Fidkowski<sup>‡</sup> and Eric Johnsen<sup>§</sup>

*The University of Michigan, Ann Arbor, MI 48109-2140, USA*

## Abstract

The Recovery-based Discontinuous Galerkin (RDG) discretization has been shown to be the most accurate among contemporary DG schemes for diffusion on a Cartesian grid. It achieves the order of accuracy  $3p + 2$  for even  $p$  and  $3p + 1$  for odd  $p$ , where  $p$  is the order of the polynomial basis. However, the overall performance of a Navier–Stokes simulation, using RDG for viscous terms, is limited to  $2p + 1$  due to the DG discretization for advection. We describe two different approaches to improve the accuracy of the DG discretization for advection. The first option is able to reach a maximal order of  $4p + 3$ , albeit utilizing an enlarged computational stencil. The second one attains  $3p + 1$  without enlarging the stencil. It is also computationally cheaper owing to the lower-order reconstruction.

## I. Introduction

Discontinuous Galerkin (DG) methods combine advantages of finite volume (FV) and finite element (FE) approaches into a single framework. The Galerkin formulation, borrowed from the FE methodology, produces a high-order accurate and compact-stencil scheme with several advantages over high-order FV and finite difference (FD) methods, such as their portability to complex geometries, scalability in parallel architecture and relatively simple extension to high order. The discontinuous nature of the basis functions in each cell introduces means for adding numerical dissipation in the convective terms, such that, unlike traditional FE schemes, discontinuous solutions can be captured in a stable fashion. The

<sup>\*</sup>Lecturer, Computational Engineering, VGU. [loc.kh@vgu.edu.vn](mailto:loc.kh@vgu.edu.vn)

<sup>†</sup>Research Collaborator, Institute for Computational Science and Technology—ICST, Ho Chi Minh City, Vietnam.

<sup>‡</sup>Associate Professor, Aerospace Engineering.

<sup>§</sup>Assistant Professor, Mechanical Engineering.

Copyright © 2015 by Loc Khieu, Krzysztof Fidkowski, and Eric Johnsen. Published by the American Institute of Aeronautics and Astronautics, Inc. with permission.

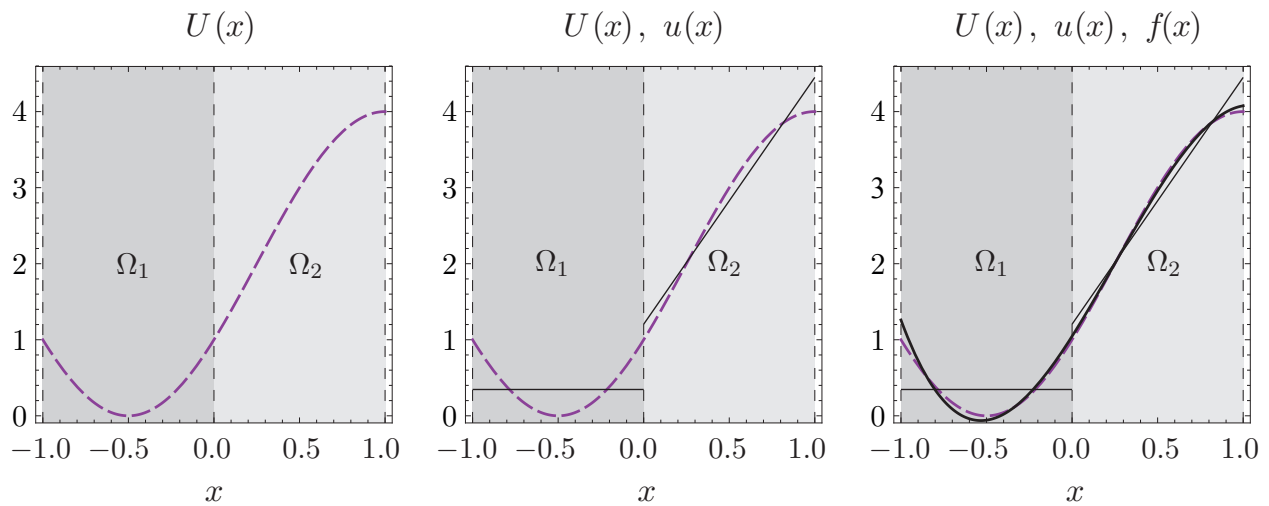


Figure 1: Recovery in one dimension for a piecewise-linear discretization ( $p = 1$ ) on two adjacent intervals  $(-1, 0)$  and  $(0, 1)$ . Shown are, from left to right, the original sin initial value  $U(x)$  (*dashed*), its piecewise-linear projection  $u(x)$  (*thin and solid*), together with  $U(x)$ , and the cubic recovered function  $f(x)$  (*thick and solid*) together with  $u(x)$  and  $U(x)$ . All three distributions yield the same value when taking their inner product with either test function on either interval, making them indistinguishable in the weak sense.

DG method was originally introduced for solving the steady-state neutron transport equation on triangular meshes,<sup>1</sup> and later analyzed.<sup>2–5</sup> The introduction of the Runge-Kutta DG (RKDG) method by Cockburn and Shu<sup>6–9</sup> made this approach appealing for time-dependent convection-dominated problems. The RKDG method was shown to be well-suited to handle shocks through the use of slope limiters in one and two dimensions, on rectangular and triangular meshes. Similar to FV schemes, DG methods use Riemann solvers to determine the inter-cell flux and thus introduce the appropriate amount of dissipation at the discontinuities.

However, the main difficulty preventing a simple extension to solving diffusive terms (i.e., second-order derivatives or differences), such as in the Navier-Stokes equations, is the fact that derivatives are undefined at cell interfaces due to the discontinuity in the basis functions between neighboring cells. Two main approaches have been followed in the past: one in which gradient information is provided and which is stabilized by interior penalty terms or artificial diffusion with adjustable parameters; and another more recent based on the idea of recovery of the underlying function over neighboring cells. With the former, interior penalty methods have been used for elliptic and parabolic problems.<sup>10–12</sup> Another class of methods treat the second-order partial differential equations as a system of first-order equations.<sup>13–16</sup> Such methods have been used to solve for diffusive terms, but there is no consistent methodology and adjustable parameters are required.

In recovery,<sup>17–21</sup> the discontinuity at the interface is removed in the weak sense by a local polynomial required to satisfy moments with the original solution in the elements that span the interface. A typical one-dimensional example is shown in Figure (1). The smooth recovered function is used to compute the necessary function values and derivatives at the interface. The resulting recovery-based discontinuous Galerkin method (RDG) is proved to be stable.<sup>17</sup> Utilizing a  $p$ -order elemental tensor-product polynomial basis, RDG has been demonstrated to achieve the order  $3p + 2$  or  $3p + 1$  for  $p$  even or odd, respectively,<sup>17</sup> on a Cartesian grid. Huynh<sup>22</sup> shows that RDG is the most accurate, and has the most favorable stability restriction, in a comparison study of all contemporary DG schemes for diffusion.

The result is robust: it holds in any number of dimensions, for linear as well as nonlinear equations, with or without mixed derivatives. Lo<sup>18,23</sup> was the first to show the optimal accuracy for a 2-D diffusion-shear operator and for the 2-D Navier-Stokes terms; Johnsen et al.<sup>24–26</sup> showed it for 3-D turbulence calculations.

We are interested in utilizing RDG in DNS simulations, e.g., compressible isotropic turbulence and the Taylor–Green vortex that were presented in 24. Owing to the simplicity in computational domains, a Cartesian grid is sufficient, and it brings out the best in RDG. However, the current overall accuracy of the simulations is seriously impaired by the DG discretization for the advection terms due to its lower order of accuracy ( $2p + 1$  comparing to  $3p + 2/3p + 1$  of RDG). To benefit fully from RDG, improvement to the advection discretization has to be made.

In our previous paper,<sup>27</sup> we have presented two approaches to improving the DG advection discretization. In the first approach named *cell-centered reconstruction*, an enhanced representation for the center cell is reconstructed using data also from the left and right neighbors, leading to an enlarged overall stencil of 5 cells in one dimension. One particular scheme of this family, the *ccf*, is identical to the  $P_p P_{3p+2}$  by Dumbser.<sup>28</sup> Schemes in this family can achieve the maximal order  $4p + 3$  as shown by Von Neumann (Fourier) analysis. To preserve the compactness, however, we develop a second approach denoted *interface-centered reconstruction*, in which the maximal order of accuracy is  $3p + 1$  and the overall stencil stays unchanged: 3 cells in one dimension and 5 cells in two dimensions.

The purpose of this paper is to present *i*) the numerical validation of the one-dimensional results and *ii*) the analysis in two dimensions. In Sections II and III, we present the concepts of improving advection discretization in one and two dimensions, respectively. Results obtained from Fourier analyses are also mentioned. The one-dimensional numerical results are shown in Section IV. Our conclusions are presented in Section V, together with ideas for further developments.

## II. Concepts of Improving DG Discretization for Advection in One Dimension

We start out by enhancing performance of DG discretization for the one-dimensional linear scalar advection equation

$$\partial_t u + \partial_x F(u) = 0, \quad (1)$$

where  $F(u) = au$  and the advection speed  $a$  is assumed to be positive without loss of generality. The flux at an arbitrary interface  $j+1/2$  is calculated by upwinding from the left and right values of  $u$  at that interface,  $u_L$  and  $u_R$  respectively. In the basic DG scheme for advection, they are calculated from the numerical representations  $u_j(x)$  and  $u_{j+1}(x)$  as shown schematically in Figure (2a). The resulted scheme achieves the order of accuracy  $2p + 1$ . The overall computational stencil to evolve solution in cell  $j$  contains three cells:  $j - 1$ ,  $j$  and  $j + 1$ .

To improve the accuracy *within* the upwinding framework, we have to raise the polynomial order of the numerical representations from which  $u_L$  and  $u_R$  are calculated. The higher-order representations  $\hat{u}_{j+1/2,L}(x)$  and  $\hat{u}_{j+1/2,R}(x)$  will be reconstructed from available data in the vicinity of the interface of interest. This is the standard technique employed in high-resolution FV methods, such as MUSCL.<sup>29</sup> This technique has been used successfully

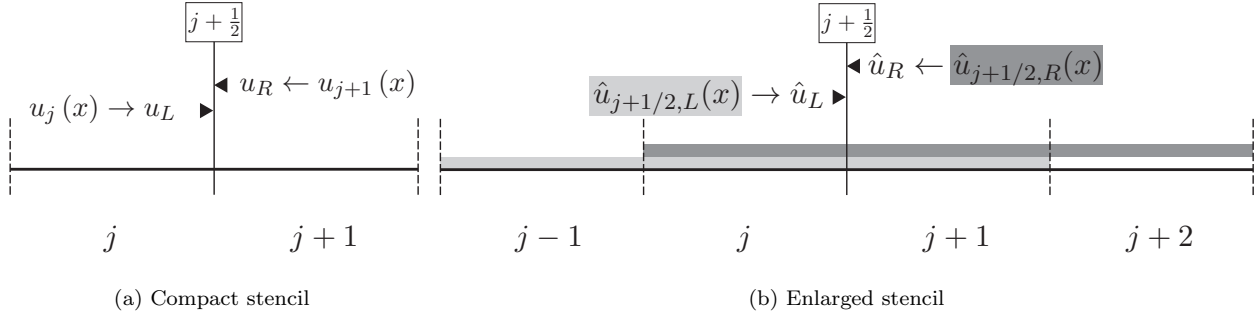


Figure 2: Stencils of flux computation at an arbitrary interface.

in DG+MUSCL<sup>26</sup> to gain one extra order of accuracy quickly and cheaply. Below, we will present two different *systematic* approaches to attain our goal of having comparable accuracy with that of RDG: *cell-centered* and *interface-centered* reconstructions.

## II.A. Cell-centered reconstruction

In this approach, the higher-order polynomial representation  $\hat{u}_{j+1/2,L}(x)$  is reconstructed using data from three cells  $j-1$ ,  $j$  and  $j+1$ . Similarly, the reconstruction of  $\hat{u}_{j+1/2,R}(x)$  uses data from three cells  $j$ ,  $j+1$  and  $j+2$ . Then the required interface values  $u_L$  and  $u_R$  are calculated from those as illustrated in Figure (2b). Repeating the same procedure at interface  $j-1/2$  will show that indeed  $\hat{u}_{j-1/2,R}(x)$  is identical to  $\hat{u}_{j+1/2,L}(x)$ . Thus, there is one enhanced representation  $\hat{u}_j$  that is *unique* to cell  $j$ ; the reconstruction is therefore called *cell-centered*.

The reconstructed  $\hat{u}$  is required to be indistinguishable from the original  $u_{j-1}$ ,  $u_j$  and  $u_{j+1}$  in the weak sense, that is,

$$\int_{x_{j-3/2}}^{x_{j-1/2}} (v_k)_{j-1} \hat{u}_j dx = \int_{x_{j-3/2}}^{x_{j-1/2}} (v_k)_{j-1} u_{j-1} dx, \quad k = 0, \dots, K_{j-1}, \quad (2)$$

$$\int_{x_{j-1/2}}^{x_{j+1/2}} (v_k)_j \hat{u}_j dx = \int_{x_{j-1/2}}^{x_{j+1/2}} (v_k)_j u_j dx, \quad k = 0, \dots, p, \quad (3)$$

$$\int_{x_{j+1/2}}^{x_{j+3/2}} (v_k)_{j+1} \hat{u}_j dx = \int_{x_{j+1/2}}^{x_{j+3/2}} (v_k)_{j+1} u_{j+1} dx, \quad k = 0, \dots, K_{j+1}. \quad (4)$$

To satisfy these equations,  $\hat{u}$  must contain  $K_{j-1} + p + K_{j+1} + 3$  degrees of freedom to form a polynomial of degree  $\hat{p} = K_{j-1} + p + K_{j+1} + 2$ , defined on the domain  $[x_{j-3/2}, x_{j+3/2}]$ . Basis functions  $(v_k)_{j-1}$ ,  $(v_k)_j$  and  $(v_k)_{j+1}$  define specific moments of the original solutions ( $u_{j-1}$ ,  $u_j$ ,  $u_{j+1}$ ) that  $\hat{u}$  has to preserve on their corresponding cells.

In theory, values of  $K_{j-1}$  and  $K_{j+1}$  can be different that leads to a *bias* reconstruction. In the context of linear scalar advection, one can argue for using upwind-bias reconstruction by utilizing more data from the left cell than from the right one ( $K_{j-1} > K_{j+1}$ ). For a system of advection equations, however, the wave structure at the interface is more complex with both left- and right-traveling waves. It is thus better not to favor any direction at this reconstruction step; from now on we take  $K_{j-1} = K_{j+1} = K$ . This approach is similar to the  $P_N P_M$  method developed by Dumbser.<sup>28,30,31</sup>

Based on the number of moments from cells  $j - 1$  and  $j + 1$  to be preserved in  $\hat{u}_j$ , we have considered two options for the reconstruction.

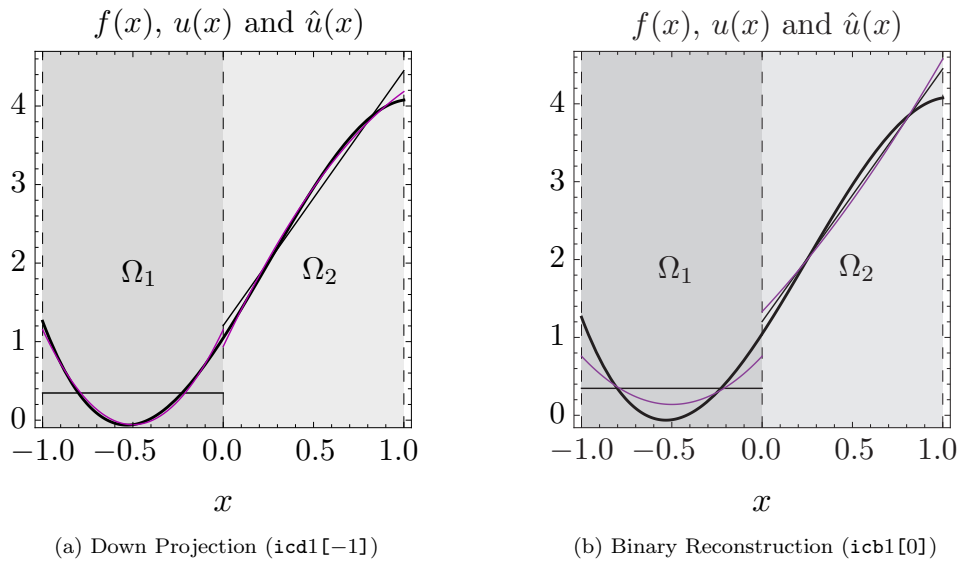
1. *Full reconstruction.* All  $p + 1$  moments from each neighbor are preserved, i.e.  $K = p$ , making  $\hat{u}_j$  a polynomial of degree  $3p + 2$ . We anticipate that the order of accuracy will increase to  $4p + 3$ , comparing to  $2p + 1$  of the basic scheme; the difference is precisely the amount of additional information taken from the neighbors. In the terminology of the  $P_N P_M$  method, this option corresponds to the  $P_p P_{3p+2}$  scheme. This option is called **ccf#** which denotes cell-centered full reconstruction at  $p = \#$ . Interchangeably, the term **ccf** might be used to indicate this scheme in a generic case.
2. *Partial reconstruction.* The previous option is an overkill as the accuracy of RDG for diffusion is only  $3p + 2/3p + 1$  for even/odd value of  $p$ . We expect to match the accuracy of RDG more closely by using only a subset of what is available. Specifically, we choose to preserve  $\frac{p+2}{2}$  or  $\frac{p+1}{2}$  moments (for  $p$  even or odd, respectively) from each neighbor cell to achieve the accuracy of  $3p + 3/3p + 2$ , making  $\hat{u}_j$  a polynomial of degree  $2p + 2/2p + 1$ . In the  $P_N P_M$  terminology, this option corresponds to  $N = p$  and  $M = 2p + 2/2p + 1$ . However, this is not strictly a  $P_N P_M$  as formulated by Dumbser et al.,<sup>28,30,31</sup> in which  $\hat{u}_j$  is required to always preserve all available moments of  $u_{j-1}$ ,  $u_j$ , and  $u_{j+1}$  on their respective domains. This generates  $3p + 3$  conditions while, by construction,  $\hat{u}_j$  has only  $2p + 3$  degrees-of-freedom for even  $p$  and  $2p + 2$  for odd  $p$ , leading to a overdetermined system that has to be solved by least-squares reconstruction. Our approach is therefore simpler and sufficiently accurate still. As before, this option is denoted **ccp#** to indicate cell-centered partial reconstruction at  $p = \#$ , or **ccp** for the generic case.

One apparent disadvantage of the cell-centered reconstruction is the enlarged five-cell overall stencil, even though the stencil stays unchanged regardless of  $p$  value. It is highly preferable to retain the original compact 3-cell stencil while improving the discretization. We will present our ideas for this matter in the next subsection.

## II.B. Interface-centered reconstruction

To keep the overall stencil unchanged from three cells, the calculation for flux at interface  $j+1/2$  should involve only data from its direct neighbors as shown in Figure (2a). The reconstruction of  $\hat{u}_{j+1/2,L}(x)$  and  $\hat{u}_{j+1/2,R}(x)$  therefore uses information only from cells  $j$  and  $j+1$ . Similarly, that of  $\hat{u}_{j-1/2,L}(x)$  and  $\hat{u}_{j-1/2,R}(x)$  contains information only from cells  $j-1$  and  $j$ . Now defined on the domain of cell  $j$ , there are two different enhanced representations,  $\hat{u}_{j-1/2,R}$  and  $\hat{u}_{j+1/2,L}$ , that strictly associate with interfaces  $j-1/2$  and  $j+1/2$ , respectively. In other words, they are *centered* at their corresponding interfaces, thus the name *interface-centered reconstruction* is used.

Our first idea is to reuse the recovered function  $f(x)$  that is already calculated for the diffusion discretization. As shown in Figure (1),  $f$  is centered at the interface of interest and its reconstruction involves only two direct neighbor cells. However, to directly use  $f_{j+2/2}$  for flux calculation at interface  $j+1/2$  is essentially similar to central discretization, which has undesired stability property for advection problem.<sup>32</sup> To create  $\hat{u}_{j+1/2,L}$  and  $\hat{u}_{j+1/2,R}$  that are discontinuous at  $x_{j+1/2}$ ,  $f$  is down projected to two lower-order polynomials defined on



**Figure 3:** Illustration of the down-projection and binary reconstructions for  $p = 1$ . The recovered function  $f$  (solid thick black line) and the piecewise-linear representations  $u$  (solid thin black line) are carried over from Figure (1). Note that  $\hat{u}(x)$  (solid thin magenta line) from both down-projection and binary reconstructions is discontinuous across  $x = 0$  interface as expected.

cells  $j$  and  $j + 1$ . Recall that  $f$  is a polynomial of degree  $2p + 1$ ; it can be down projected to polynomials of degrees  $\hat{p}$ , where  $p + 1 \leq \hat{p} \leq 2p$ .

$$\int_{x_{j-1/2}}^{x_{j+1/2}} (v_k)_j \hat{u}_{j+1/2,L} dx = \int_{x_{j-1/2}}^{x_{j+1/2}} (v_k)_j f_{j+1/2} dx, \quad k = 0, \dots, \hat{p}, \quad (5)$$

$$\int_{x_{j+1/2}}^{x_{j+3/2}} (v_k)_{j+1} \hat{u}_{j+1/2,R} dx = \int_{x_{j+1/2}}^{x_{j+3/2}} (v_k)_{j+1} f_{j+1/2} dx, \quad k = 0, \dots, \hat{p}. \quad (6)$$

The recovered function  $f$  is not down projected to polynomials of degree  $p$  because it results in  $u_j$  and  $u_{j+1}$ , which form the basic scheme.

Resulting schemes will be called  $\text{icd}\#_1[-\#_2]$  standing for interface-centered down projection at  $p = \#_1$ ; its generic name is  $\text{icd}$ . The number between the square brackets,  $\#_2$ , denotes the reduction amount in polynomial degree of  $u_{j+1/2,L/R}$  comparing to  $f$ . For example, scheme  $\text{icd}1[-1]$  has  $f$  down projected to polynomials of one-degree less, i.e.,  $u_{j+1/2,L/R}$  are  $2p^{\text{th}}$ -order polynomials. Figure (3a) illustrates two enhanced representations,  $\hat{u}_{0,L}$  and  $\hat{u}_{0,R}$ , associated with interface  $x = 0$  of scheme  $\text{icd}1[-1]$ , for  $p = 1$  discretization; the recovered function  $f$  and the piecewise-linear representations are carried over from example in Figure (1).

We expect that the central scheme resulted from direct usage of  $f$  achieves the order

$$(2p + 1) + (p + 1) = 3p + 2.$$

The order of accuracy of an arbitrary  $\text{icd}\#_1[-\#_2]$  scheme will then be  $3p + 2 - \#_2$ , and the only viable scheme is  $\text{icd}\#_1[-1]$ . It matches RDG scheme for odd  $p$  but achieves one-order lower when  $p$  is even.

The second idea of compact reconstruction is inspired by the recovery procedure in RDG, with a small modification. The reconstruction for  $\hat{u}_{j+1/2,L}$  will now bias towards the left

**Table 1: Down-projection and binary-reconstruction schemes.**

$p$	Down Projection		Binary Reconstruction		
	$\hat{p}$	Scheme names	$\hat{p}$	$\mathcal{K}$	Scheme names
1	2	icd1[-1]	2	{0}	icb1[0]
				{1}	icb1[1]
2	4	icd2[-1]	4	{0, 1}	icb2[0, 1]
				{0, 2}	icb2[0, 2]
				{1, 2}	icb2[1, 2]
	3	icd2[-2]	3	{0}	icb2[0]
				{1}	icb2[1]
				{2}	icb2[2]
3	6	icd3[-1]	6	{0, 1, 2}	icb3[0, 1, 2]
				{0, 1, 3}	icb3[0, 1, 3]
				{0, 2, 3}	icb3[0, 2, 3]
				{1, 2, 3}	icb3[1, 2, 3]
	5	icd3[-2]	5	{0, 1}	icb3[0, 1]
				{0, 2}	icb3[0, 2]
				{0, 3}	icb3[0, 3]
				{1, 2}	icb3[1, 2]
				{1, 3}	icb3[1, 3]
	4	icd3[-3]	4	{0}	icb3[0]
				{1}	icb3[1]
				{2}	icb3[2]
				{3}	icb3[3]

neighbor, cell  $j$ , indicating it might preserve more moments of  $u_j$  than those of  $u_{j+1}$ ,

$$\int_{x_{j-1/2}}^{x_{j+1/2}} (v_k)_j \hat{u}_{j+1/2,L} dx = \int_{x_{j-1/2}}^{x_{j+1/2}} (v_k)_j u_j dx, \quad k = 0, \dots, p, \quad (7)$$

$$\int_{x_{j+1/2}}^{x_{j+3/2}} (v_k)_{j+1} \hat{u}_{j+1/2,L} dx = \int_{x_{j+1/2}}^{x_{j+3/2}} (v_k)_{j+1} u_{j+1} dx, \quad k \in \mathcal{K}, \quad (8)$$

where  $\mathcal{K}$  is only a subset of  $\{0, \dots, p\}$ , which are indexes of all moments of the original numerical representations. Similarly, the reconstruction for  $\hat{u}_{j+1/2,R}$  biases towards  $u_{j+1}$  of

Table 2: Results from Fourier analysis for  $p=1$ 

Scheme	ooa <sup>a</sup>	Taylor-series expansion of $\lambda_{\text{con}}$
basic <sup>b</sup>	3	$-i\beta - \frac{\beta^4}{72} + O(\beta^5)$
ccf1	7	$-i\beta - \frac{\beta^8}{19\,600} + O(\beta^9)$
ccp1	5	$-i\beta - \frac{11\beta^6}{7\,200} + O(\beta^7)$
icd1[-1]	4	$-i\beta + \frac{i\beta^5}{180} + \frac{\beta^6}{108} + O(\beta^7)$
icb1[0]	4	$-i\beta + \frac{i\beta^5}{180} - \frac{\beta^6}{288} + O(\beta^7)$
icb1[1]	4	$-i\beta + \frac{i\beta^5}{180} - \frac{\beta^6}{432} + O(\beta^7)$

<sup>a</sup> Order of accuracy<sup>b</sup> Basic DG discretization for advection

the right neighbor,

$$\int_{x_{j-1/2}}^{x_{j+1/2}} (v_k)_j \hat{u}_{j+1/2,R} dx = \int_{x_{j-1/2}}^{x_{j+1/2}} (v_k)_j u_j dx, \quad k \in \mathcal{K}, \quad (9)$$

$$\int_{x_{j+1/2}}^{x_{j+3/2}} (v_k)_{j+1} \hat{u}_{j+1/2,R} dx = \int_{x_{j+1/2}}^{x_{j+3/2}} (v_k)_{j+1} u_{j+1} dx, \quad k = 0, \dots, p, \quad (10)$$

The same subset  $\mathcal{K}$  will be used in both Eqs. (8) and (9). There are at most  $p$  elements in  $\mathcal{K}$  and they are not required to be sequential. For example,  $\mathcal{K} = \{0, 2\}$  indicates that  $\hat{u}_{j+1/2,L}$  and  $\hat{u}_{j+1/2,R}$  will only preserve the 0<sup>th</sup>-order and 2<sup>nd</sup>-order moments of  $u_{j+1}$  and  $u_j$  respectively. All schemes of this family will be called **icb**<sub>#<sub>1</sub></sub>[#<sub>2</sub>] denoting interface-centered binary reconstruction at  $p = \#_1$ , and #<sub>2</sub> are all elements of subset  $\mathcal{K}$ ; again, **icb** is used for a generic case. The enhanced representations from scheme **icb1**[0] ( $p = 1$ ) is displayed in Figure (3b).

In terms of order of accuracy, we expect that the binary-reconstruction schemes will achieve a maximal order of  $3p + 1$  corresponding to  $\mathcal{K}$  has  $p$  members. There are  $p + 1$  possibilities to form distinct and orderless subsets of this type from the set  $\{0, 1, \dots, p\}$ , leading to  $p + 1$  flavors of a  $(3p + 1)$ -order binary-reconstruction scheme. When the number of members in  $\mathcal{K}$  is less than  $p$ , the order of accuracy reduces accordingly.

All possible down-projection and binary-reconstruction schemes for  $p \in [1, 3]$  are shown in Table 1. Tables 2 and 3 depict the results obtained by Fourier analysis for  $p = 1$  and  $p = 2$ ,



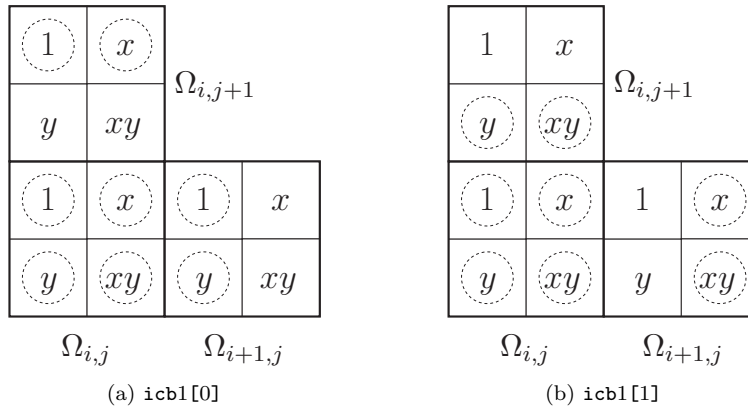
Table 3: Results from Fourier analysis for  $p=2$

Scheme	ooa	Taylor-series expansion of $\lambda_{\text{con}}$
basic	5	$-i\beta - \frac{\beta^6}{7200} + O(\beta^7)$
ccf2	11	$-i\beta - \frac{\beta^{12}}{426888000} + O(\beta^{13})$
ccp2	9	$-i\beta - \frac{103\beta^{10}}{457228800} + O(\beta^{11})$
icd2[-1]	7	$-i\beta + \frac{\beta^8}{793800} + O(\beta^9)$
icd2[-2]	6	$-i\beta - \frac{i\beta^7}{34650} + O(\beta^8)$
icb2[0,1]	7	$-i\beta + \frac{\beta^8}{120960} + O(\beta^9)$
icb2[0,2]	7	$-i\beta + \frac{13\beta^8}{1058400} + O(\beta^9)$
icb2[1,2]	7	$-i\beta + \frac{\beta^8}{66150} + O(\beta^9)$
icb2[0]	6	$-i\beta + \frac{17i\beta^7}{403200} + O(\beta^8)$
icb2[1]	6	$-i\beta + \frac{29i\beta^7}{604800} + O(\beta^8)$
icb2[2]	6	$-i\beta + \frac{i\beta^7}{16800} + O(\beta^8)$

respectively. It must be noted that the *ooa* obtained from Fourier analysis corresponds to that of the cell average.

### III. Interface-Centered Binary Reconstruction in Two Dimensions

Our previous results<sup>27</sup> from Von Neumann analysis showed that *icd* is not a good option due to its inconsistent behavior with respect to stability. In that same paper, the *icb* scheme was shown to behave much better. Thus, we expand only *icb* schemes into two dimensions.



**Figure 4: Two options to choose which original moments of cells  $\Omega_{i,j}$  and  $\Omega_{i+1,j}$  that the reconstructed  $\hat{u}_{i+1/2,L}$  will preserve ( $p = 1$ ). The two options for  $\hat{u}_{j+1/2,B}$  (between cells  $\Omega_{i,j}$  and  $\Omega_{i,j+1}$ ) are also shown.**

Without loss of generality, we consider the reconstructed polynomial  $\hat{u}_{i+1/2,L}$  that is centered at the interface between cell  $\Omega_{i,j}$  and cell  $\Omega_{i+1,j}$ , at  $p = 1$ . This enhanced polynomial biases to the left cell, hence it preserves all original moments of cell  $\Omega_{i,j}$  and only a few from the right one. The reconstruction equations are straightforward extensions of Eqs. (7) and (8):

$$\int_{\Omega_{i,j}} (v_{k,l})_{i,j} \hat{u}_{i+1/2,L} dx dy = \int_{\Omega_{i,j}} (v_{k,l})_{i,j} u_{i,j} dx dy, \quad \{k, l\} \in \{0, 1\}, \quad (11)$$

$$\int_{\Omega_{i+1,j}} (v_{k,l})_{i+1,j} \hat{u}_{i+1/2,L} dx dy = \int_{\Omega_{i+1,j}} (v_{k,l})_{i+1,j} u_{i+1,j} dx dy, \quad k \in \mathcal{K}, l \in \{0, 1\}, \quad (12)$$

where  $k$  and  $l$  are indications of the highest powers of  $x$  and  $y$  in the preserved moments, respectively. In **icb1[0]**,  $\mathcal{K} = \{0\}$  indicating that  $\hat{u}_{i+1/2,L}$  preserves moments that correspond to  $x^0 y^0 = 1$  and  $x^0 y^1 = y$  as shown in Figure (4a). Similarly, the reconstruction in **icb1[1]** ( $\mathcal{K} = \{1\}$ ) will preserve moments corresponding to  $x^1 y^0 = x$  and  $x^1 y^1 = xy$  as shown in Figure (4b). The highest order term in  $\hat{u}_{i+1/2,L}$  calculated from Eqs. (11) and (12), in both **icb1[0]** and **icb1[1]** schemes, is  $x^2 y^1$ .

The polynomial  $\hat{u}_{j+1/2,B}$  centered at the interface between cell  $\Omega_{i,j}$  and cell  $\Omega_{i,j+1}$  will satisfy the following similar equations:

$$\int_{\Omega_{i,j}} (v_{k,l})_{i,j} \hat{u}_{j+1/2,B} dx dy = \int_{\Omega_{i,j}} (v_{k,l})_{i,j} u_{i,j} dx dy, \quad \{k, l\} \in \{0, 1\}, \quad (13)$$

$$\int_{\Omega_{i,j+1}} (v_{k,l})_{i,j+1} \hat{u}_{j+1/2,B} dx dy = \int_{\Omega_{i,j+1}} (v_{k,l})_{i,j+1} u_{i,j+1} dx dy, \quad k \in \{0, 1\}, l \in \mathcal{K}. \quad (14)$$

By the same token,  $\mathcal{K} = \{0\}$  in **icb1[0]** and the reconstruction will preserve moments corresponding to  $x^0 y^0 = 1$  and  $x^1 y^0 = x$ . In **icb1[1]**, moments correspond to  $x^0 y^1 = y$  and  $x^1 y^1 = xy$  will be preserved.

We expected that the order of accuracy of **icb** schemes in two dimensions is still  $3p + 1$ , same as in one dimension. The increase is exactly the enhancement in polynomial order with respect to the normal direction of the interfaces. Preliminary results from Fourier analysis show that the order of accuracy is indeed  $3p + 1$ , at least for  $p = \{1, 2\}$  (see Table 4). The

Table 4: Results from Fourier analysis in two dimensions

Scheme	ooa	Taylor-series expansion of $\lambda_{\text{con}}$
basic( $p = 1$ )	3	$-2i\beta - \frac{\beta^4}{36} + O(\beta^5)$
icb1[0]	4	$-2i\beta + \frac{i\beta^5}{90} + O(\beta^6)$
basic( $p = 2$ )	5	$-2i\beta - \frac{\beta^6}{3600} + O(\beta^7)$
icb2[0, 1]	7	$-2i\beta + \frac{\beta^8}{60480} + O(\beta^{10})$

two-dimensional stencil for **icb** scheme remains unchanged from the five-cell stencil of the basic DG discretization for advection, which is *as compact as possible*.

#### IV. Numerical results for the linear scalar advection in one dimension

To validate the results of Fourier analysis shown in Section II, we solve the *scalar linear* one-dimensional advection equation numerically,

$$\partial_t u + \partial_x u = 0. \quad (15)$$

The initial data is a sine wave  $\sin(2\pi x)$  that is then advected in positive direction at the unit speed  $a = 1$ . The exact solution is  $u_{\text{exact}}(t, x) = \sin(2\pi(x - t))$ . The simulations are performed on a unit-length domain that is uniformly divided into  $N$  segments. The numerical approximation is a polynomial of degree  $p$ ,

$$u_j(x) = \sum_{\alpha=0}^p a_j^{(\alpha)} \phi_j^{(\alpha)}(x), \quad (16)$$

in which the values of the DG coefficients  $a^{(\alpha)}$  are the solutions of the numerical simulations.

In addition to the basic DG discretization, the advection term is discretized by the cell-centered *full* reconstruction (**ccf**), cell-centered *partial* reconstruction (**ccp**), and interface-centered binary reconstruction (**icb**). It is then evolved in time by the Runge–Kutta 4-stage (RK4) time-updating scheme until  $T = 1$ . The  $L_2$ -norms of errors in the numerical solutions, as well as the corresponding orders of accuracy, are shown in Tables 5 and 6. The *ooa* of the cell averages  $a^{(0)}$  matches the Fourier-analysis prediction presented in Tables 2 and 3.

It is exciting to note that the errors in  $a^{(0)}$  and  $a^{(1)}$  from the **icb2**[0, 1] ( $p = 2$ ) are smaller than those of **ccf1** ( $p = 1$ ), even though the overall stencil of the former has only 9 degrees of freedom while that of the latter contains 10. As explained in Section II, in one dimension, the stencil of **icb** comprises only three cells while that of **ccf** has five cells.

**Table 5: Errors and *ooa* of the numerical solutions ( $p = 1$ ) of the *scalar linear* advection equation in one dimension.**

$$\|a^{(0)}\|_2$$

$N$	basic		ccf1		ccp1		icb1[0]	
2	5.65E-1		1.82E-1		4.78E-1		4.89E-1	
4	1.63E-1	1.8	2.83E-3	6.0	4.26E-2	3.5	9.23E-2	2.4
8	2.72E-2	2.6	3.29E-5	6.4	1.81E-3	4.6	8.22E-3	3.5
16	3.67E-3	2.9	3.05E-7	6.8	6.18E-5	4.9	5.68E-4	3.9
32	4.68E-4	3	2.52E-9	6.9	1.98E-6	5	3.66E-5	4
64	5.89E-5	3	2.00E-11	7	6.24E-8	5	2.31E-6	4

$$\|a^{(1)}\|_2$$

$N$	basic		ccf1		ccp1		icb1[0]	
2	5.35E-2		4.79E-2		5.76E-2		2.59E-1	
4	1.48E-1	-1.5	2.73E-3	4.1	4.05E-2	0.5	9.11E-2	1.5
8	3.28E-2	2.2	4.06E-5	6.1	2.23E-3	4.2	1.02E-2	3.2
16	8.80E-3	1.9	7.35E-7	5.8	1.49E-4	3.9	1.37E-3	2.9
32	2.25E-3	2	1.22E-8	5.9	9.54E-6	4	1.77E-4	3
64	5.67E-4	2	1.93E-10	6	6.01E-7	4	2.22E-5	3

## V. Conclusions and future developments

We have presented two different approaches to improving accuracy of the DG method for advection using an upwind flux. Our goal is to match the accuracy level of the recovery-based DG method, which will be used to discretize the viscous terms of the Navier–Stokes equations.

In the first approach named *cell-centered reconstruction*, a unique enhanced representation is reconstructed for each computational cell. The computation uses data from its two left and right neighbor cells. Therefore, the stencil for flux calculation is enlarged from 2 cells of the basic DG discretization to 4 cells, leading to a unfavorable five-cell overall stencil. The order of accuracy for *full reconstruction* option (**ccf**) is as high as  $4p + 3$ , and that of the less-accurate version *partial reconstruction* (**ccp**) is  $3p + 3$  or  $3p + 2$  for  $p$  even or odd. The former version is the familiar  $P_N P_M$  method, but the latter is different. Von Neumann analyses indicate that both options have good stability properties.

Our desire for a compact overall stencil leads to the development of *interface-centered reconstruction* approach. The flux-calculation stencil remains two cells and the overall stencil contains 3 cells, as compact as possible. This approach leads to two distinct families of schemes: the *down-projection reconstruction* denoted by **icd**, and the *binary reconstruction* denoted by **icb**. The most accurate members in both families achieve the order  $3p + 1$ . This

order of accuracy is verified numerically for the *scalar linear* advection in one dimension. Further numerical experiments with the *nonlinear* advection equations are planned.

Both families have unfavorable stability properties; the situation of the *icd* family is more severe and we deem it is not worthwhile to explore it further. Further in the future, we will explore options to increase the stability domain of the *icb* family, e.g., via adding artificial dissipation, while the order  $3p + 1$  is preserved.

*Multidimensionality* does not cause any adverse effect to the *icb* scheme. The two-dimensional Fourier analysis shows that the order of accuracy remains unchanged,  $3p + 1$ .

## References

<sup>1</sup>Reed, W. and Hill, T., “Triangular mesh methods for the neutron transport equation,” Tech. Rep. LA-UR-73-479, Los Alamos Scientific Laboratory, 1973.

<sup>2</sup>Johnson, C. and Pitkäranta, J., “An analysis of the discontinuous Galerkin method for a scalar hyperbolic equation,” *Mathematics of Computation*, Vol. 46, No. 173, 1986, pp. 1–26.

<sup>3</sup>LeSaint, P. and Raviart, P.-A., “On a finite element method for solving the neutron transport equation,” *Mathematical aspects of finite elements in partial differential equations*, edited by C. De Boor, Academic Press, 1974, p. 89.

<sup>4</sup>Peterson, T. E., “A note on the convergence of the discontinuous Galerkin method for a scalar hyperbolic equation,” *SIAM Journal on Numerical Analysis*, Vol. 28, No. 1, 1991, pp. 133–140.

<sup>5</sup>Richter, G. R., “An optimal-order error estimate for the discontinuous Galerkin method,” *Mathematics of Computation*, Vol. 50, No. 181, 1988, pp. 75–88.

<sup>6</sup>Cockburn, B. and Shu, C.-W., “TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws. II. General framework,” *Mathematics of Computation*, Vol. 52, No. 186, 1989, pp. 411–435.

<sup>7</sup>Cockburn, B., Lin, S.-Y., and Shu, C.-W., “TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws III: one-dimensional systems,” *Journal of Computational Physics*, Vol. 84, No. 1, 1989, pp. 90–113.

<sup>8</sup>Cockburn, B., Hou, S., and Shu, C.-W., “The Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws. IV. The multidimensional case,” *Mathematics of Computation*, Vol. 54, No. 190, 1990, pp. 545–581.

<sup>9</sup>Cockburn, B. and Shu, C.-W., “The RungeKutta discontinuous Galerkin method for conservation laws V: multidimensional systems,” *Journal of Computational Physics*, Vol. 141, No. 2, 1998, pp. 199–224.

<sup>10</sup>Arnold, D. N., Brezzi, F., Cockburn, B., and Marini, L. D., “Unified analysis of discontinuous Galerkin methods for elliptic problems,” *SIAM Journal on Numerical Analysis*, Vol. 39, No. 5, 2002, pp. 1749–1779.

<sup>11</sup>Brezzi, F., Manzini, G., Marini, D., Pietra, P., and Russo, A., “Discontinuous Galerkin approximations for elliptic problems,” *Numerical Methods for Partial Differential Equations*, Vol. 16, No. 4, 2000, pp. 365–378.

<sup>12</sup>Douglas, J. and Dupont, T., “Interior penalty procedures for elliptic and parabolic Galerkin methods,” *Computing methods in applied sciences*, Springer, 1976, pp. 207–216.

<sup>13</sup>Bassi, F. and Rebay, S., “A high-order accurate discontinuous finite element method for the numerical solution of the compressible Navier-Stokes equations,” *Journal of Computational Physics*, Vol. 131, No. 2, 1997, pp. 267–279.

<sup>14</sup>Bassi, F., Rebay, S., Mariotti, G., Pedinotti, S., and Savini, M., “A high-order accurate discontinuous finite element method for inviscid and viscous turbomachinery flows,” *Proceedings of 2nd European Conference on Turbomachinery, Fluid Dynamics and Thermodynamics*, Technologisch Instituut, Antwerpen, Belgium, 1997, pp. 99–108.

<sup>15</sup>Cockburn, B. and Shu, C. W., “The local discontinuous Galerkin method for time-dependent convection-diffusion systems,” *SIAM Journal on Numerical Analysis*, Vol. 35, No. 6, 1998, pp. 2440–2463.

<sup>16</sup>Peraire, J. and Persson, P. O., “The compact discontinuous Galerkin (CDG) method for elliptic problems,” *Siam Journal on Scientific Computing*, Vol. 30, No. 4, 2008, pp. 1806–1824.

<sup>17</sup>Lo, M. and Van Leer, B., “Analysis and Implementation of Recovery-Based Discontinuous Galerkin for Diffusion,” 22-25 June 2009, AIAA 2009-3786.

<sup>18</sup>Lo, M. and Van Leer, B., “Recovery-Based Discontinuous Galerkin for Navier-Stokes Viscous Terms,” 27-30 June 2011 2011, AIAA 2011-3406.

<sup>19</sup>Van Leer, B. and Nomura, S., “Discontinuous Galerkin for Diffusion,” 2005, AIAA-2005-5108.

<sup>20</sup>Van Leer, B., Lo, M., and Van Raalte, M., “A Discontinuous Galerkin Method for Diffusion Based on Recovery,” 25–28 June 2007, AIAA 2007-4083.

<sup>21</sup>Van Leer, B. and Lo, M., “Unification of Discontinuous Galerkin Methods for Advection and Diffusion,” 5-8 January 2009 2009, AIAA 2009-400.

<sup>22</sup>Huynh, H. T., “A Reconstruction Approach to High-Order Schemes Including Discontinuous Galerkin for Diffusion,” *47th AIAA Aerospace Sciences Meeting Including The New Horizons Forum and Aerospace Exposition*, Orlando, Florida, US, 2009, AIAA 2009-0403.

<sup>23</sup>Lo, K. H. M., *A space-time discontinuous Galerkin method for Navier-Stokes with recovery*, Ph.D. thesis, 2011.

<sup>24</sup>Johnsen, E., Varadan, S., and Van Leer, B., “A Three-Dimensional Recovery-Based Discontinuous Galerkin Method for Turbulence Simulations,” *51st AIAA Aerospace Sciences Meeting*, Grapevine, Texas, 7–10 January 2013, AIAA 2013-0515.

<sup>25</sup>Johnsen, E., Varadan, S., and Nair, A., “Recovery Discontinuous Galerkin Method for Compressible Turbulence,” *21st AIAA Computational Fluid Dynamics Conference*, American Institute of Aeronautics and Astronautics, 24–27 June, AIAA 2013-3066.

<sup>26</sup>Johnsen, E., Khieu, L., and Nair, A., “A Simple Method to Improve the Accuracy of Advection in Discontinuous Galerkin Methods for Navier-Stokes Simulations,” *52nd Aerospace Science Meeting*, American Institute of Aeronautics and Astronautics, 13–17 January 2014, AIAA 2014-1276.

<sup>27</sup>Khieu, L. and Johnsen, E., “Analysis of Improved Advection Schemes for Discontinuous Galerkin Methods,” *44th AIAA Fluid Dynamics Conference*, American Institute of Aeronautics and Astronautics, 16–20 June, AIAA 2014-3221.

<sup>28</sup>Dumbser, M., Balsara, D. S., Toro, E. F., and Munz, C. D., “A unified framework for the construction of one-step finite volume and discontinuous Galerkin schemes on unstructured meshes,” *Journal of Computational Physics*, Vol. 227, No. 18, 2008, pp. 8209–8253.

<sup>29</sup>Van Leer, B., “Towards the ultimate conservative difference scheme. V. A second-order sequel to Godunov’s method,” *Journal of Computational Physics*, Vol. 32, No. 1, 1979, pp. 101–136.

<sup>30</sup>Dumbser, M., *Arbitrary high order schemes for the solution of hyperbolic conservation laws in complex domains*, Shaker, 2005.

<sup>31</sup>Dumbser, M. and Munz, C. D., “Arbitrary high order discontinuous Galerkin schemes,” *Numerical methods for hyperbolic and kinetic problems*, edited by S. Cordier, T. Goudon, M. Gutnic, and E. Sonnendrücker, Vol. 7 of *IRMA Lectures in Mathematics and Theoretical Physics*, EMS Publishing House, 2005, pp. 295–333.

<sup>32</sup>Hirsch, C., *Numerical Computation of Internal and External Flows*, Vol. 1: Fundamental of Computational Fluid Dynamics, Butterworth-Heinemann, 2nd ed., 2007.

**Table 6: Errors and  $ooa$  of the numerical solutions ( $p = 2$ ) of the *scalar linear* advection equation in one dimension.**

$$\|a^{(0)}\|_2$$

$N$	basic		ccf2		ccp2		icb2[0, 1]	
2	8.18E-2		1.20E-3		9.83E-3		2.54E-2	
4	4.73E-3	4.1	1.41E-6	9.7	4.42E-5	7.8	7.32E-4	5.1
8	1.80E-4	4.7	7.09E-10	11	1.10E-7	8.7	6.72E-6	6.8
16	5.93E-6	4.9	3.70E-13	10.9	2.28E-10	8.9	5.46E-8	6.9
32	1.88E-7	5	1.82E-16	11	4.53E-13	9	4.31E-10	7
64	5.89E-9	5	8.91E-20	11			3.38E-12	7

$$\|a^{(1)}\|_2$$

$N$	basic		ccf2		ccp2		icb2[0, 1]	
2	7.79E-2		7.02E-4		8.84E-3		7.66E-2	
4	4.58E-3	4.1	1.55E-6	8.8	5.07E-5	7.4	4.51E-4	7.4
8	2.03E-4	4.5	1.25E-9	10.3	1.23E-7	8.7	7.58E-6	5.9
16	1.35E-5	3.9	8.36E-13	10.5	5.20E-10	7.9	1.25E-7	5.9
32	8.67E-7	4	8.39E-16	10	2.09E-12	8	1.99E-9	6
64	5.46E-8	4	8.25E-19	10			3.13E-11	6

$$\|a^{(2)}\|_2$$

$N$	basic		ccf2		ccp2		icb2[0, 1]	
2	1.64E-1		2.78E-3		2.10E-2		6.07E-2	
4	2.16E-2	2.9	5.14E-6	9.1	1.89E-4	6.8	3.39E-3	4.2
8	2.84E-3	2.9	1.11E-8	8.9	1.73E-6	6.8	1.06E-4	5
16	3.57E-4	3	2.22E-11	9	1.37E-8	7	3.29E-6	5
32	4.46E-5	3	4.32E-14	9	1.07E-10	7	1.02E-7	5
64	5.58E-6	3	8.42E-17	9			3.20E-9	5

This article has been cited by:

1. Philip E. Johnson, Eric Johnsen. A New Family of Discontinuous Galerkin Schemes for Diffusion Problems . [[Citation](#)] [[PDF](#)] [[PDF Plus](#)]