Discontinuous Galerkin Methods for Convection Diffusion Equations: Positivity Preserving and Multi-scale Resolution

by

Yifan Zhang

B.Sc., Nanjing University; Nanjing, China, 2009

A dissertation submitted in partial fulfillment of the
requirements for the degree of Doctor of Philosophy
in The Division of Applied Mathematics at Brown University

PROVIDENCE, RHODE ISLAND

May 2013

This dissertation by Yifan Zhang is accepted in its present form
by The Division of Applied Mathematics as satisfying the
dissertation requirement for the degree of Doctor of Philosophy.

Date_____ _____

Chi-Wang Shu, Ph.D., Advisor

Recommended to the Graduate Council

Date_____ _____

Jan S. Hesthaven, Ph.D., Reader

Date_____ _____

Johnny Guzmán, Ph.D., Reader

Approved by the Graduate Council

Date_____ _____

Peter M. Weber, Dean of the Graduate School

Vitae

Biographical Information

Born on October 25th, 1987 in Yangzhou, Jiangsu, China

Education

Brown University, Ph.D. in Applied Mathematics, May 2013 (expected)

Nanjing Unviersity, B.Sc in Statistics, June 2009

Publication

- Y. Zhang, X. Zhang and C.-W. Shu, "Maximum-principle-satisfying second order discontinuous Galerkin schemes for convection-diffusion equations on triangular meshes ", *Journal of Computational Physics*, **234**, 2013, 295–316
- Y. Zhang and R.-D. Nair, "A Non-Oscillatory discontinuous Galerkin transport scheme on the Cubed-Sphere", *Monthly Weather Review*, **140**, 2012, 3106–3126
- Y. Zhang, W. Wang, J. Guzmán and C.-W. Shu, "Multi-scale discontinuous

Galerkin method for solving elliptic problems with curvilinear unidirectional rough coefficients", *In preparation*, 2013

Experience

09/2010-05/2011 **Teaching Assistant**, Brown University

05/2011-08/2011 **Research Intern**, National Center for Atmospheric Research

Acknowledgements

These four years at Brown, 2009-2013 have definitely been the most intense yet exciting period of time in my life so far. I am truly fortunate to have been granted the luxury of focusing on various challenging and interesting research projects, working hard and pushing my intellectual limit. For all of these wonderful and unique experiences in life, I have to thank Prof. Chi-Wang Shu, my thesis advisor, and also the person who encouraged me to apply for graduate school at Brown in the first place. The academic training and excellent guidance I received from him made the pursuit of a doctoral's degree a much smoother journey than I had anticipated. Beyond research, I would always be indebted to the tremendous freedom and unconditional support that he gave me to explore what I really want to choose as a life long career. The high standards which he holds towards research are such inspirations that will always carry on in my later life and career.

Along with my advisor, I want to thank Prof. Jan Hesthaven and Prof. Johnny Guzmán who have kindly agreed to serve on my dissertation committee, for spending time reading my work and provide valuable feedback.

Over the years, I have had the honer and pleasure to work on various interesting projects with many extraordinary collaborators, Prof. Johnny Guzmán at Brown University, Prof. Wei Wang at Florida International University, Dr. Xiangxiong Zhang at MIT, Dr. Ram Nair at NCAR. Their help, insights and intelligent work

have benefited me in many ways. Parts of this dissertation are based on two publications, Y. Zhang et al.[82], Y. Zhang et al.[80]. My coauthors have contributed considerably to both articles through their ideas, suggestions and feedbacks.

I consider myself very lucky to have grown up academically within the graduate student community at Brown University, Division of Applied Mathematics. I have to admit that surrounding myself with exceptionally smart and hard working peers motivated me greatly during all these four years. Thus, I am thankful to the high-achieving peer stimulation. In particular, I want to thank a former group member, Dr. Robert GS Jiang for mentoring me. His suggestions and encouragement were and will continue to be a great asset in my future career.

Last, but by no means least, I would like to thank all my family members, my mother and father for understanding and respecting every decision I made in my life, for sheltering me from being worried about external pressures and allowing me to pursue what I was truly interested in right after college; my cousin Ying Dai for cheering me up during all those tough times. And finally, my deepest gratitude goes to Yuan, the special one in my life, who always embraces me as who I am.

Abstract of " Discontinuous Galerkin Methods for Convection Diffusion Equations: Positivity Preserving and Multi-scale Resolution " by Yifan Zhang, Ph.D., Brown University, May 2013

This dissertation focuses on studies of two different discontinuous Galerkin (DG) methods for general convection-diffusion equations. One preserves the strict maximum principle for general nonlinear convection-diffusion equations on unstructured triangular meshes, and the other is designed specifically for a class of second order elliptic problems with rough coefficients, where the local oscillating directions vary smoothly in the computational domain.

It is a highly desirable property that a numerical scheme is maximum-principle-satisfying when solving general nonlinear convection-diffusion equations, not only because violating it may produce physically meaningless numerical solutions, but could also lead to numerical instability or ill-posedness. Similar to the pure convection case [75] we propose, under suitable time step constraints, by using Strong Stability Preserving (SSP) time discretizations, and coupling with a simple scaling limiter, DG methods preserve the physical bounds indicated by the initial condition while maintaining uniform second order accuracy. Extensions to two-dimensional convection-diffusion equations on triangular meshes are derived without imposing any geometric constraints on the mesh such as angle acuteness.

It is well known that when designing finite element schemes for elliptic equations with rough coefficients, computational efficiency can be improved by properly incorporating local oscillating features into the approximation space. Thus, by choosing a special non-polynomial approximation space, we propose special DG schemes which capture the multi-scale solution without having to resolve the finest scales. Detailed error estimates for two dimensional second order DG methods are derived, and a general guidance on how to construct such non-polynomial basis is discussed.

This methodology can be applied to various problems and applications including porous medium equations, incompressible Navier-Stokes equation, as well as composite material problems. In addition, the many good properties that DG scheme has make it an ideal method for applications in atmospheric research. After combining it with the H-WENO (Hermite Weighted Essentially Non-Oscillatory) limiter and a Bound-Preserving (BP) filter, we designed a transport equation DG solver for large scale atmospheric modeling on a cubed-sphere geometry, guaranteed to provide non-oscillatory, positivity-preserving solutions.

Contents

List of Tables

List of Figures

Chapter One

Introduction

The discontinuous Galerkin (DG) method, which falls into the category of non-conventional finite-element methods, was first introduced by Reed and Hill in 1973 [54], for time-independent linear hyperbolic equations. Later in the 1990's, in a series of paper [13, 12, 11], Cockburn et al. combined the DG spatial discretization, using Riemann solver for numerical fluxes at element interfaces, with high order stable explicit Runge-Kutta (RK) time integration [25, 26] to develop a general framework for the construction of robust algorithms, known as the RKDG method, for solving nonlinear time-dependent problems. Moreover, in the presence of strong shocks or contact discontinuities, a non-linear slope limiter [11], already studied extensively in the context of low-order finite-volume methods, is employed to suppress oscillations to maintain non-oscillatory properties. Due to the various desired properties of DG schemes such as geometric flexibility, mass conservations, h-p adaptivity and high parallel efficiency, the method has found rapid application in various areas, e.g. gas dynamics, semiconductor simulations, atmospheric modeling, etc. To deal with higher order spatial derivatives, in particular, general nonlinear convection-diffusion equations, a direct generalization of the previous DG methodology does not apply, and may lead to inconsistent solutions [15, 72]. By carefully designing the numerical fluxes, various DG formulations have been proposed to cope with second order derivatives, e.g, [2, 5, 10, 66, 1, 55, 14]. Successful applications include porous medium equations, composite material problems, incompressible Navier-Stokes equations etc.

The DG method may be viewed as a hybrid approach combining the good features of two classical numerical methods, the finite-element and finite-volume methods, exploiting the merits of both. By allowing completely discontinuous basis functions and foregoing the continuity constraints across element interfaces, required by the conventional finite element methods, the DG methods have a flexibility that is not shared with the traditional finite volume such as the ability to increase the order by

choosing the local approximation spaces. Furthermore, the local compact data structure of DG schemes typically ensures a better parallel efficiency compared with most finite element methods. In this dissertation, we utilize the intrinsic flexibility of the DG scheme to design numerical techniques and special algorithms for general nonlinear convection-diffusion equations, which not only improve the quality and resolution of the numerical approximations, but also enhance the computational efficiency. In particular, we propose two DG methods, one being maximum-principle-satisfying for nonlinear convection-diffusion equations, and the other capturing the multi-scale behavior for elliptic equations with curvilinear unidirectional rough coefficients.

In the first part of the dissertation, we are interested in constructing maximum-principle-satisfying DG schemes for nonlinear convection-diffusion equations:

$$u_t + f(u)_x = a(u)_{xx}, \quad u(x,0) = u_0(x), \tag{1.0.1}$$

An important property of the exact solution is the strict maximum principle, i.e., if $m = \min_x u_0(x)$, $M = \max_x u_0(x)$, then for any (x,t), $u(x,t) \in [m,M]$. In particular, the positivity-preserving property is a special case where the lower bound $m = 0$. Numerical algorithms that fail to meet this requirement will not only produce physically meaningless results, but may also lead to ill-posedness and numerical instability.

The discrete maximum principle (DMP) for numerical schemes solving elliptic type equations have been studied since the 1960s. Even though DMP is more difficult to achieve for parabolic equations, maximum-principle-satisfying finite element methods were studied in a similar fashion. The earliest work about DMP for parabolic equations is [21] in which Fujii investigated DMP for the heat equation using piecewise linear finite elements. For the most recent results on nonlinear parabolic

equations, see [19] and the references therein. The main techniques to achieve DMP for parabolic equations appear to be algebraic, e.g. the lumped mass method. Such approaches tend to pose certain restrictions on the spatial mesh such as angle acuteness [60] and also the rather unnatural time step restriction that the time step must be larger than a threshold, in order to use the larger numerical viscosity associated with an implicit time discretization with a large time step.

Motivated by [48], Zhang et al. recently developed an arbitrarily high order maximum-principle-satisfying DG scheme for scalar conservation laws on unstructured triangular meshes [75, 79]. They proved a sufficient condition for the cell averages of the high order DG scheme to stay bounded in $[m, M]$. A simple scaling limiter based on a one dimension limiter in [40] is then utilized to enforce this sufficient condition. It is appealing to extend the results in [75, 79] to convection-diffusion equations since the method is easy to implement and does not add constraint on the spatial mesh. Unfortunately, it proves difficult to do so, at least for arbitrarily high order schemes. The main difficulty lies in the step to obtain a sufficient condition. To achieve an arbitrarily high order approximation, a non-conventional maximum-principle-satisfying finite volume scheme for convection-diffusion equations was developed in [74]. This was achieved by defining a quantity called a double cell average over the interval I_j, utilizing the cell averaging operator twice on the original solution. It is then straightforward to apply the methodology in [75], including the scaling limiter, to ensure a strict maximum principle for the numerical solutions while keeping high order accuracy. Even though this finite volume method suggests a possible alternative for high order maximum-principle -satisfying numerical schemes for convection-diffusion equations, it is an unconventional finite volume method requiring new implementations. It is also not obvious how to generalize this technique to finite volume methods on arbitrary unstructured meshes, or to DG methods.

In [82], we propose a generalization of the framework in [75, 79] for piecewise linear DG methods for solving one- and two-dimensional convection-diffusion equations on arbitrary triangular meshes. We also show that the second order accurate maximum-principle-satisfying DG schemes maintain all the good properties of those in [75, 79] including mass conservation and ease of implementation. In other words, we propose a DG scheme for convection-diffusion equations satisfying the maximum principle without imposing any geometric constraints on the computational mesh or restrictive time step constraints.

By further exploring the flexibility of the DG scheme, a multi-scale DG method is proposed in the second part of the dissertation. The scheme is restricted to a special class of two dimensional second order elliptic boundary value problems with highly oscillatory coefficients locally varying sharply in one direction:

$$-\nabla \cdot (A(\mathbf{x})\nabla u) = f(\mathbf{x}) \quad \text{in } \Omega \tag{1.0.2}$$

It is well-known result that if the elliptic coefficients $A(\mathbf{x})$ are highly oscillatory functions involving a small scale ϵ, the true solution u will also be oscillating rapidly. In general, we will not have enough regularity in u with respect to ϵ. As a result, it requires a computational mesh size h at least comparable to ϵ to obtain reasonable resolutions and to observe any convergence rate. This intrinsic difficulty prohibits the application of traditional numerical methods like finite-element, finite-difference, due to the large computational cost.

There have been a lot of efforts to overcome or bypass this numerical difficulty. As early as in the 60s, Tikhonov and Samarskii [61] (see also [24]) designed a simple 3-point finite difference scheme which can solve the one-dimensional version of (1.0.2) exactly at the grid points by utilizing harmonic averages and the special so-

lution structure. However, this method is difficult to extend to multi-dimensions. Later, various multi-scale finite element methods [3, 17, 29, 27, 28, 16, 8, 71] have been developed and extensively studied in the literature. The universal idea therein is to have the local properties of the differential operators built in to the scheme and, as a result, allowing adequate resolutions on coarser meshes. Among them, the special finite element methods proposed by Babuška et al. in [29], exploring the special feature of unidirectional or curvilinear unidirectional elliptic coefficients, provide an explicit and efficient framework for constructing local trial spaces for this specific subclass of two dimensional elliptic problem. The key ingredient is to prove a generalized Berstein Theorem for u in a transformed domain. For more general cases of the oscillatory coefficients $A(\mathbf{x})$, though the general idea of incorporating the oscillatory basis into the approximation spaces still applies, such regularity result no longer holds. Local problems have to be first solved on a finer grid in order to obtain an approximation for the oscillatory basis. More details regarding a general framework for error analysis and related applications can be found in a series of work [27, 28, 16, 8] by Hou et al.

Motivated by [3, 17, 29], Yuan and Shu proposed a multiscale Babuška-Zlámal DG method with a non-polynomial approximation DG space [70] for the one dimensional version of Eq. (1.0.2), and they proved arbitrary high order error estimates. Later, in [64], Wang improved the one-dimensional proof of the multiscale Babuška-Zlámal DG method by only assuming that the solution is uniformly bounded with respect to ϵ in $H^1(\Omega)$. More regularity was assumed in [71] while proving estimates for the multiscale Babuška-Zlámal DG method. Followed by previous work, in [65], Wang et al. proposed a multi-scale symmetric interior penalty DG (IP-DG ([31, 66, 1])) method. This is proved to be uniformly high order accurate, with respect to the small scale ϵ in one space dimension, even in the case where there is no separation

of scales (the problem could have a continuum of scales from $O(\epsilon)$ to $O(1)$) and the mesh size h is much larger than ϵ. Optimal order convergence for second order scheme in two space dimension given unidirectional oscillations was also proven.

As for the curvilinear unidirectional rough coefficients, in [81] we extend the idea in [29] to a multi-scale DG [65] framework, which automatically overcome the difficulties of requiring the elements to be conforming for multi-dimensions of the continuous Galerkin method and the practical inconvenience of using curvilinear elements. In addition, we prove uniform optimal accuracy for a second order approximation space in the two-dimensional case, without assuming any properties such as periodicity or scale separation in the rough coefficients.

As mentioned previously, the various desirable features of the DG method have made it a popular choice for many real world applications and large scale simulations. In the last part of this dissertation, we present a non-oscillatory DG transport equation solver on the cubed-sphere geometry in the context of atmospheric research. By coupling the DG scheme with the Hermite-Weighted Essentially Non-oscillatory (H-WENO) type limiter developed by Qiu et al. in [51] and the Bound Preserving (BP) filter proposed by Zhang et al. in [75], we obtain an efficient algorithm [80] which is non-oscillatory and positivity preserving, and which can be easily extended to a parallel environment, thus makes it an ideal method for atmospheric modeling.

The dissertation is organized as follows: in Chapter 2, we first propose a maximum-principle satisfying DG scheme for general nonlinear convection-diffusion equations and prove it to be uniformly second order accurate with generalizations to unstructured meshes and incompressible Navier-Stokes equations. In Chapter 3, a multi-scale DG method is proposed for a class of second order two dimensional multi-scale elliptic equations with coefficients that are curvilinear unidirectional oscillatory. Op-

timal error estimates are proven for a second order approximation space regardless of the fine scales. Applications of non-oscillatory and positivity preserving DG scheme to large scale computation in atmospheric research are discussed in Chapter 4. Concluding remarks are given in Chapter 5.

Chapter Two

Maximum-Principle-Satisfying Second Order Discontinuous Galerkin Schemes for Convection-Diffusion Equations on Triangular Meshes

2.1 Introduction

In this chapter, we are interested in constructing maximum-principle-satisfying discontinuous Galerkin (DG) schemes for nonlinear convection-diffusion equations:

$$u_t + f(u)_x = a(u)_{xx}, \quad u(x,0) = u_0(x), \tag{2.1.1}$$

or equivalently,

$$u_t + f(u)_x = (b(u)u_x)_x, \quad u(x,0) = u_0(x), \tag{2.1.2}$$

where $b(u) = a'(u) \geq 0$, and their two-dimensional versions. An important property satisfied by the exact solution is the strict maximum principle, i.e., if $\max u_0(x) = M$, $\min u_0(x) = m$, then

$$u(x,t) \in [m, M], \quad \forall x \in \mathbb{R}, \quad \forall t \geq 0. \tag{2.1.3}$$

In particular, for the case $m = 0$, it is the positivity-preserving property, which is desired for numerical schemes in many applications. For physical quantities like densities and probability distributions, negative values are non-physical in the numerical simulations, and may lead to ill-posedness and instability for certain nonlinear equations such as chemotaxis problems [59].

To have a maximum-principle-satisfying scheme for a convection-diffusion problem (2.1.1) with a significant or dominant convection part, one may prefer an explicit time discretization and will first need to construct such schemes for the purely convection case.

$$u_t + f(u)_x = 0, \quad u(x,0) = u_0(x). \tag{2.1.4}$$

Motivated by [48], it turns out that one can construct arbitrarily high order maximum-

principle-satisfying DG schemes for scalar conservation laws on unstructured triangular meshes [75, 79]. It was the first time that genuinely high order maximum-principle-satisfying finite volume and DG schemes for multidimensional nonlinear scalar conservation laws on arbitrary triangular meshes became available. This methodology has been successfully applied to various problems [69, 77, 63, 78, 68, 73, 53, 56, 9] for positivity-preservation of physically relevant properties such as density, pressure and water height. For a survey of the recent developments of such schemes and applications, see [76]. The most interesting observations in [75, 79] consist of two parts. First, if an intuitively reasonable sufficient condition is satisfied, i.e., the values of the DG polynomials at the current time level and at certain quadrature points are in the desired range $[m, M]$, then the cell averages of the numerical solutions at the next time step will also be in this range under suitable time step restrictions with an explicit Euler forward time stepping. Second, a simple scaling limiter can enforce the sufficient condition of keeping the DG polynomials at these quadrature points in the desired bounds, once we know their cell averages are already in these bounds, without destroying accuracy and conservation. High order accuracy in time can be achieved by strong stability preserving Runge-Kutta or multi-step methods, which are convex combinations of Euler forward steps. Thus DG methods with this simple scaling limiter will satisfy the maximum principle for scalar conservation laws.

However, it proves difficult to extend the results in [75, 79] to convection-diffusion equations in a straightforward way, at least for arbitrarily high order schemes. The main difficulty lies in the first step above, namely to write the formula for obtaining the cell average at the next time step by forward Euler time stepping, in terms of certain point values while ensuring monotonicity with respect to these point values, which is a key ingredient in [75]. We explain in more detail this difficulty in next subsection. The main content of this chapter is to the extend the frame-

work of [75, 79] to piecewise linear DG methods in solving one- and two-dimensional convection-diffusion equations on arbitrary triangular meshes. The second order accurate maximum-principle-satisfying DG schemes share all the good properties as those in [75, 79] including mass conservation and ease of implementation. We also discuss applications of this scheme to nonlinear degenerate parabolic equations and the incompressible Navier-Stokes equations.

We remark that most numerical schemes satisfying a maximum principle for parabolic equations in the literature use implicit time discretizations. However the small time step $\Delta t = \mathcal{O}(h^2)$ turns out to be necessary even for implicit time discretizations to have the contraction property, see [18]. We use explicit time discretizations which are advantageous for solving nonlinear equations, especially for convection dominated problems. Moreover, our method is straightforward to implement. The limiter is a local operator on each cell. In practice, for each time step in a multistep method or each time stage in a Runge-Kutta method, one only needs to add the limiter to the DG scheme as a pre-processing step. See Section 2.3.3 for a discussion of implementation.

2.2 Maximum-principle-satisfying DG schemes in one dimension

2.2.1 Preliminaries

We establish the maximum-principle-satisfying DG schemes for convection-diffusion equations in one space dimension in this section. We first review briefly the mono-

tonicity results for the conservation laws (2.1.4). For simplicity, we will assume periodic boundary conditions from now on. All results regarding maximum principle can be easily incorporated into any initial-boundary value problems.

On the mesh $x_{\frac{1}{2}} < x_{\frac{3}{2}} < \cdots < x_{N-\frac{1}{2}} < x_{N+\frac{1}{2}}$, the equation satisfied by the cell averages of a DG method with forward Euler time discretization can be written as

$$\overline{u}_j^{n+1} = \overline{u}_j^n - \frac{\Delta t}{h}\left[\widehat{f}\left(u^-_{j+\frac{1}{2}}, u^+_{j+\frac{1}{2}}\right) - \widehat{f}\left(u^-_{j-\frac{1}{2}}, u^+_{j-\frac{1}{2}}\right)\right] \tag{2.2.1}$$

where $h = x_{j+\frac{1}{2}} - x_{j-\frac{1}{2}}$ is the mesh size, assumed to be uniform for simplicity, n refers to the time level and j denotes the spatial cell, and $\overline{u}_j$ is the numerical approximation to the cell average obtained by the DG scheme. $u^{\pm}_{j+\frac{1}{2}}$ are the values of the numerical solution at the point $x_{j+\frac{1}{2}}$ evaluated from the cell I_{j+1} and I_j respectively at time level n. $\widehat{f}(\cdot,\cdot)$ is a monotone flux, which is an increasing function of the first argument and a decreasing function of the second argument, e.g, the global Lax-Friedrichs flux,

$$\widehat{f}(u,v) = \frac{1}{2}\left(f(u) + f(v) - a(v-u)\right), \quad a = \max_u |f'(u)|. \tag{2.2.2}$$

It is well-known that the first order scheme

$$H_\lambda(u,v,w) = v - \lambda(\widehat{f}(v,w) - \widehat{f}(u,v)) \tag{2.2.3}$$

is a monotonically increasing function of all three arguments u, v and w provided a CFL condition is satisfied. For the Lax-Friedrichs flux (2.2.2) the CFL condition is

$$\lambda a \leq 1, \tag{2.2.4}$$

where $\lambda = \frac{\Delta t}{h}$.

Thus, a first order monotone scheme must satisfy the strict maximum principle. Motivated by [48], in [75, 79], a general framework was proposed to construct a genuinely high order maximum principle satisfying DG scheme for (2.1.4). Let the piecewise polynomials $p_j(x)$ of degree k denote the DG polynomials at time level n and $\overline{u}_j^n$ denote the cell averages of $p_j(x)$. $u^{\pm}_{j+\frac{1}{2}}$ are the values of the numerical solution at $x_{j+\frac{1}{2}}$ evaluated from the cell I_{j+1} and I_j respectively at time level n. The new strategy is to first consider an N-point Legendre Gauss-Lobatto quadrature rule which is exact for the integral of polynomials of degree up to $2N-3$ on the interval $I_j = [x_{j-\frac{1}{2}}, x_{j+\frac{1}{2}}]$. Denote these quadrature points on I_j as

$$S_j = \{x_{j-\frac{1}{2}} = \widehat{x}_j^1, \widehat{x}_j^2, \cdots, \widehat{x}_j^{N-1}, \widehat{x}_j^N = x_{j+\frac{1}{2}}\}. \tag{2.2.5}$$

Let $\widehat{\omega}_\beta$ be the quadrature weights for the interval $[-\frac{1}{2}, \frac{1}{2}]$ such that $\sum_{\beta=1}^{N} \widehat{\omega}_\beta = 1$. Choose N to be the smallest integer satisfying $2N-3 \geq k$, then

$$\overline{u}_j^n = \frac{1}{h}\int_{I_j} p_j(x)\, dx = \sum_{\beta=1}^{N} \widehat{\omega}_\beta p_j(\widehat{x}_j^\beta) = \sum_{\beta=2}^{N-1} \widehat{\omega}_\beta p_j(\widehat{x}_j^\beta) + \widehat{\omega}_1 u^+_{j-\frac{1}{2}} + \widehat{\omega}_N u^-_{j+\frac{1}{2}}. \tag{2.2.6}$$

With (2.2.6), by adding and subtracting $\widehat{f}\left(u^+_{j-\frac{1}{2}}, u^-_{j+\frac{1}{2}}\right)$, (2.2.1) can be rewritten as

$$\begin{aligned}
\overline{u}_j^{n+1} &= \sum_{\beta=2}^{N-1} \widehat{\omega}_\beta p_j(\widehat{x}_j^\beta) + \widehat{\omega}_N \left(u^-_{j+\frac{1}{2}} - \frac{\lambda}{\widehat{\omega}_N}\left[\widehat{f}\left(u^-_{j+\frac{1}{2}}, u^+_{j+\frac{1}{2}}\right) - \widehat{f}\left(u^+_{j-\frac{1}{2}}, u^-_{j+\frac{1}{2}}\right)\right]\right) \\
&\quad + \widehat{\omega}_1 \left(u^+_{j-\frac{1}{2}} - \frac{\lambda}{\widehat{\omega}_1}\left[\widehat{f}\left(u^+_{j-\frac{1}{2}}, u^-_{j+\frac{1}{2}}\right) - \widehat{f}\left(u^-_{j-\frac{1}{2}}, u^+_{j-\frac{1}{2}}\right)\right]\right) \\
&= \sum_{\beta=2}^{N-1} \widehat{\omega}_\beta p_j(\widehat{x}_j^\beta) + \widehat{\omega}_N H_{\lambda/\widehat{\omega}_N}(u^+_{j-\frac{1}{2}}, u^-_{j+\frac{1}{2}}, u^+_{j+\frac{1}{2}}) + \widehat{\omega}_1 H_{\lambda/\widehat{\omega}_1}(u^-_{j-\frac{1}{2}}, u^+_{j-\frac{1}{2}}, u^-_{j+\frac{1}{2}}).
\end{aligned} \tag{2.2.7}$$

At this point, recalling the property of the first order monotone operator (2.2.3) under condition (2.2.4) and that $\widehat{\omega}_1 = \widehat{\omega}_N$, it is clear that under the CFL condition $\lambda a \leq \widehat{\omega}_1$, $\overline{u}_j^{n+1}$ is a monotonically increasing function of all the arguments involved, namely $u^-_{j-\frac{1}{2}}$, $u^+_{j+\frac{1}{2}}$ and $p_j(\widehat{x}_j^\beta)$ for $1 \leq j \leq N$. Therefore, if all these point values are in $[m, M]$, then $\overline{u}_j^{n+1} \in [m, M]$ as well. A sufficient condition to make $\overline{u}_j^{n+1} \in [m, M]$ is then $p_j(\widehat{x}_j^\beta) \in [m, M]$ for all j and β. This can be achieved by a scaling limiter which will be described later.

To summarize, the key ingredients in constructing a high order maximum-principle-satisfying DG scheme for (2.1.4) in [75] are:

- The cell average at the next time step by forward Euler time stepping, i.e., (2.2.1), is a monotone function with respect to certain point values (Gauss-Lobatto points for the one-dimensional case), under a suitable CFL condition.
- A simple limiter modifies the DG polynomial $p_j(x)$ on cell I_j into $\widetilde{p}_j(x)$ such that $\widetilde{p}_j(x) \in [m, M]$ at these special points without changing its cell average. Moreover, it can be proven that the modified polynomial $\widetilde{p}_j(x)$ is also a high order approximation just as $p_j(x)$. Thus we have $\overline{u}_j^{n+1} \in [m, M]$ if all the degree of freedoms at time level n on the right hand side of (2.2.1) are replaced by those of modified polynomials $\widetilde{p}_j(x)$.
- Replace the forward Euler by the strong stability preserving (SSP) high order time discretizations which, as convex combinations of forward Euler steps, will maintain the bounds. It suffices to prove the maximum principle for (2.2.1).

2.2.2 Maximum-principle-satisfying DG schemes for convection - diffusion equations: one-dimensional case

There exist several different types of DG formulations to treat the second order viscous terms. In this section we focus on three different formulations: the DG formulation of Cheng and Shu for higher order derivatives [10], the interior penalty DG (IPDG) scheme [66, 1, 55] and the local DG (LDG) scheme [14].

We start with the Cheng-Shu DG scheme to take care of the second order derivatives [10]. For (2.1.1), their scheme can be described as: given a computational mesh of the domain $[a,b]$: $a = x_{\frac{1}{2}} < x_{\frac{3}{2}} < \cdots < x_{N-\frac{1}{2}} < x_{N+\frac{1}{2}} = b$, where the mesh size $h = x_{i+\frac{1}{2}} - x_{i-\frac{1}{2}}$ is assumed to be uniform, define the approximation space $V_h^k = \{v : v|_{I_j} \in P^k(I_j), \forall j\}$, where $P^k(I_j)$ denotes the set of all polynomials up to degree k defined on the cell I_j. For any test function $v \in V_h^k$, the DG formulation is to find $u \in V_h^k$ such that,

$$\begin{aligned} & \int_{I_j} u_t v dx - \int_{I_j} f(u) v_x dx - \int_{I_j} a(u) v_{xx} dx + \widehat{f}_{j+\frac{1}{2}} v^-_{j+\frac{1}{2}} - \widehat{f}_{j-\frac{1}{2}} v^+_{j-\frac{1}{2}} \\ - \quad & \widehat{a_x}_{j+\frac{1}{2}} v^-_{j+\frac{1}{2}} + \widehat{a_x}_{j-\frac{1}{2}} v^+_{j-\frac{1}{2}} + \widehat{a}_{j+\frac{1}{2}} (v_x)^-_{j+\frac{1}{2}} - \widehat{a}_{j-\frac{1}{2}} (v_x)^+_{j-\frac{1}{2}} = 0, \end{aligned} \tag{2.2.8}$$

where the convection flux $\widehat{f}_{j+\frac{1}{2}} = \widehat{f}(u^-_{j+\frac{1}{2}}, u^+_{j+\frac{1}{2}})$ is chosen as described previously, and the diffusion fluxes are chosen in an alternating way with an additional penalty on the second order derivative term:

$$\widehat{a_x}(u) = \frac{[a(u)]}{[u]} \left(u_x^- + \frac{\alpha}{h}[u] \right), \qquad \widehat{a}(u) = a(u^+). \tag{2.2.9}$$

Here α is a sufficiently large positive coefficient, and $[u] = u^+ - u^-$ denotes the jump across the cell boundaries. It is proven in [10] that the DG scheme (2.2.8) is stable with a sub-optimal error estimate. Numerical experiments indicate optimal $(k+1)$-th

order convergence in L^2.

By taking the test function $v = 1$, we obtain the evolutionary equation for the cell averages of the numerical scheme:

$$\frac{d}{dt}\overline{u}_j = -\frac{1}{h}\left((\widehat{f}_{j+\frac{1}{2}} - \widehat{f}_{j-\frac{1}{2}}) - (\widehat{a_x}_{j+\frac{1}{2}} - \widehat{a_x}_{j-\frac{1}{2}})\right). \tag{2.2.10}$$

We consider the first order forward Euler time discretization for (2.2.10); higher order time discretization will be discussed later. Our first result is:

Theorem 2.2.1. *For the P^1-DG scheme (2.2.8) with forward Euler time stepping, the maximum principle holds for the cell averages $\overline{u}_j^{n+1} \in [m, M]$ for all j, if $u^{\pm}_{j+\frac{1}{2}} \in [m, M]$ for all j and the following CFL conditions hold*

$$\alpha \geq 1, \quad \lambda \max_u |f'(u)| \leq \frac{1}{4}, \quad \mu(\alpha+1)\max_u(a'(u)) \leq \frac{1}{4} \tag{2.2.11}$$

where $\mu = \frac{\lambda}{h} = \frac{\Delta t}{h^2}$.

Proof. Applying forward Euler time stepping on (2.2.10), we obtain the formula for the cell averages of the numerical solution at the next time step

$$\overline{u}_j^{n+1} = \overline{u}_j^n - \lambda(\widehat{f}_{j+\frac{1}{2}} - \widehat{f}_{j-\frac{1}{2}}) + \lambda\left((\widehat{a_x})_{j+\frac{1}{2}} - (\widehat{a_x})_{j-\frac{1}{2}}\right). \tag{2.2.12}$$

To utilize the result in the purely convection case [75], we divide the cell average $\overline{u}_j^n$ into two halves:

$$\overline{u}_j^{n+1} = \left(\frac{1}{2}\overline{u}_j^n - \lambda(\widehat{f}_{j+\frac{1}{2}} - \widehat{f}_{j-\frac{1}{2}})\right) + \left(\frac{1}{2}\overline{u}_j^n + \lambda((\widehat{a_x})_{j+\frac{1}{2}} - (\widehat{a_x})_{j-\frac{1}{2}})\right). \tag{2.2.13}$$

This decomposition may not lead to the optimal CFL restriction for the proof but

is adopted for easy presentation.

The diffusion term:

$$D_j = \frac{1}{2}\overline{u}_j^n + \lambda((\widehat{a_x})_{j+\frac{1}{2}} - (\widehat{a_x})_{j-\frac{1}{2}}). \tag{2.2.14}$$

By the linearity of the approximation space with $k = 1$:

$$\overline{u}_j = \frac{u^-_{j+\frac{1}{2}} + u^+_{j-\frac{1}{2}}}{2}, \quad (u_x^-)_{j+\frac{1}{2}} = \frac{u^-_{j+\frac{1}{2}} - u^+_{j-\frac{1}{2}}}{h}$$

$$(\widehat{a_x})_{j+\frac{1}{2}} = \xi_{j+\frac{1}{2}} \frac{\alpha u^+_{j+\frac{1}{2}} - (\alpha-1)u^-_{j+\frac{1}{2}} - u^+_{j-\frac{1}{2}}}{h}, \quad \xi_{j+\frac{1}{2}} = \frac{[a]_{j+\frac{1}{2}}}{[u]_{j+\frac{1}{2}}}.$$

By the mean value theorem, there exists ξ_0 such that $\xi_{j+\frac{1}{2}} = \frac{[a]_{j+\frac{1}{2}}}{[u]_{j+\frac{1}{2}}} = a'(\xi_0) \geq 0$. Rewriting (2.2.14), omitting the superscript denoting time level n, we have

$$\begin{aligned} D_j &= \frac{1}{4}u^-_{j+\frac{1}{2}} + \frac{1}{4}u^+_{j-\frac{1}{2}} + \mu\left(\xi_{j+\frac{1}{2}}\alpha u^+_{j+\frac{1}{2}} - \xi_{j+\frac{1}{2}}(\alpha-1)u^-_{j+\frac{1}{2}} - (\xi_{j+\frac{1}{2}} + \alpha\xi_{j-\frac{1}{2}})u^+_{j-\frac{1}{2}}\right. \\ &\quad \left. +\xi_{j-\frac{1}{2}}(\alpha-1)u^-_{j-\frac{1}{2}} + \xi_{j-\frac{1}{2}}u^+_{j-\frac{3}{2}}\right) \\ &= \mu\xi_{j+\frac{1}{2}}\alpha u^+_{j+\frac{1}{2}} + \left(\frac{1}{4} - \mu\xi_{j+\frac{1}{2}}(\alpha-1)\right)u^-_{j+\frac{1}{2}} + \left(\frac{1}{4} - \mu(\xi_{j+\frac{1}{2}} + \alpha\xi_{j-\frac{1}{2}})\right)u^+_{j-\frac{1}{2}} \\ &\quad +\mu(\alpha-1)\xi_{j-\frac{1}{2}}u^-_{j-\frac{1}{2}} + \mu\xi_{j-\frac{1}{2}}u^+_{j-\frac{3}{2}}. \end{aligned}$$

The convection term:

$$C_j = \frac{1}{2}\overline{u}_j^n - \lambda(\widehat{f}_{j+\frac{1}{2}} - \widehat{f}_{j-\frac{1}{2}}). \tag{2.2.15}$$

According to the discussion in the preliminaries, in the P^1-DG case, we can choose

the 2-point Gauss-Lobatto quadrature with the weights $\widehat{\omega}_1 = \widehat{\omega}_2 = \frac{1}{2}$. Then we have

$$C_j = \frac{1}{4}H_1 + \frac{1}{4}H_2 \tag{2.2.16}$$

where

$$H_1 = u^-_{j+\frac{1}{2}} - 4\lambda\left(\widehat{f}(u^-_{j+\frac{1}{2}}, u^+_{j+\frac{1}{2}}) - \widehat{f}(u^+_{j-\frac{1}{2}}, u^-_{j+\frac{1}{2}})\right) = H_{4\lambda}\left(u^+_{j-\frac{1}{2}}, u^-_{j+\frac{1}{2}}, u^+_{j+\frac{1}{2}}\right),$$

$$H_2 = u^+_{j-\frac{1}{2}} - 4\lambda\left(\widehat{f}(u^+_{j-\frac{1}{2}}, u^-_{j+\frac{1}{2}}) - \widehat{f}(u^-_{j-\frac{1}{2}}, u^+_{j-\frac{1}{2}})\right) = H_{4\lambda}\left(u^-_{j-\frac{1}{2}}, u^+_{j-\frac{1}{2}}, u^-_{j+\frac{1}{2}}\right).$$

Combining the results for the convection and diffusion terms together, under the CFL conditions (2.2.11), $\overline{u}_j^{n+1}$ becomes a convex combination of point values in $[m, M]$, thus belonging to $[m, M]$. □

Remark 2.2.2. *The key ingredient in the proof above is to write D_j in (2.2.14) as a convex combination of point values in the current time step under suitable time step restriction. If we attempt to follow the same route for the piecewise quadratic case $k = 2$, we encounter difficulties even for the linear case $a(u) = u$. We can see from the above derivation that we only need to pay attention to $(u_x)^-_{j-1/2}$. Rewriting the DG polynomial in cell I_{j-1} as an Lagrangian interpolation of three point values, one being $u^-_{j-\frac{1}{2}}$ and the other two point values inside I_{j-1} being denoted as u_a and u_b. We then have $(u_x)^-_{j-1/2} = au_a + bu_b + cu^-_{j-\frac{1}{2}}$ for three constants a, b and c, and simple algebra indicates that, in order to ensure D_j is written as a convex combination of point values in the current time step, we would need the coefficients a and b to be both non-positive. It can be easily checked that this is not possible. Thus our current approach does not work for DG methods of order higher than 2.*

Remark 2.2.3. *We have only proved the result for the first order forward Euler time stepping. The result also holds for high order SSP Runge-Kutta or multistep methods [25] as they are convex combinations of forward Euler steps. For instance,*

the second order SSP Runge-Kutta method (with the CFL coefficient $c = 1$) is

$$\begin{aligned} u^{(1)} &= u^n + \Delta t F(u^n) \\ u^{n+1} &= \frac{1}{2}u^n + \frac{1}{2}(u^{(1)} + \Delta t F(u^{(1)})) \end{aligned} \tag{2.2.17}$$

where $F(u)$ is the spatial operator, the CFL coefficient c for a SSP time discretization refers to the fact that, if we assume the forward Euler time discretization for solving the equation $u_t = F(u)$ is stable in a norm or a semi-norm under a time step restriction $\Delta t \leq \Delta t_0$, then the high order SSP time discretization is also stable in the same norm or semi-norm under the time step restriction $\Delta t \leq c\Delta t_0$.

Remark 2.2.4. *This P^1-DG result can also be extended to other penalty type DG formulations for second order derivatives, for example the interior penalty Galerkin methods [66, 1, 55], with slightly different CFL conditions and stability coefficient restrictions. If we consider the interior penalty DG formulation for (2.1.2): for any test function $v \in V_h^k$, the IPDG scheme is to find $u \in V_h^k$, such that*

$$\begin{aligned} \int_\Omega u_t v dx &= \sum_{j=1}^{N} \left(\int_{I_j} f(u) v_x dx - \widehat{f}_{j+\frac{1}{2}} v^-_{j+\frac{1}{2}} + \widehat{f}_{j-\frac{1}{2}} v^+_{j-\frac{1}{2}} \right) \\ &- \sum_{j=1}^{N} \left(\int_{I_j} b(u) u_x v_x dx + \{bu_x\}_{j+\frac{1}{2}} [v]_{j+\frac{1}{2}} \pm \{bv_x\}_{j+\frac{1}{2}} [u]_{j+\frac{1}{2}} \right. \\ &\left. + \frac{\alpha}{h} [u]_{j+\frac{1}{2}} [v]_{j+\frac{1}{2}} \right) \end{aligned}$$

where $\{bu_x\} = \frac{1}{2}((bu_x)^+ + (bu_x)^-)$, and α is the penalty coefficient. The plus and minus signs above in front of the $\{bv_x\}_{j+\frac{1}{2}}[u]_{j+\frac{1}{2}}$ term refer to symmetric [66, 1] and nonsymmetric [55] penalty methods respectively. The diffusion term (2.2.14) then becomes:

$$D_j = \frac{1}{2}\overline{u}_j^n + \lambda \left(\{bu_x\}_{j+\frac{1}{2}} - \{bu_x\}_{j-\frac{1}{2}} + \frac{\alpha}{h}[u]_{j+\frac{1}{2}} - \frac{\alpha}{h}[u]_{j-\frac{1}{2}} \right) \tag{2.2.18}$$

which can be written out as (omitting the superscript denoting the time level n):

$$\begin{aligned} D_j &= \frac{1}{4}u^-_{j+\frac{1}{2}} + \frac{1}{4}u^+_{j-\frac{1}{2}} + \mu\left(\frac{1}{2}b^+_{j+\frac{1}{2}}u^-_{j+\frac{3}{2}} + (\alpha - \frac{1}{2}b^+_{j+\frac{1}{2}})u^+_{j+\frac{1}{2}}\right. \\ &\quad -(\alpha + \frac{1}{2}b^+_{j-\frac{1}{2}} - \frac{1}{2}b^-_{j+\frac{1}{2}})u^-_{j+\frac{1}{2}} - (\alpha + \frac{1}{2}b^-_{j+\frac{1}{2}} - \frac{1}{2}b^+_{j-\frac{1}{2}})u^+_{j-\frac{1}{2}} \\ &\quad \left. + (\alpha - \frac{1}{2}b^-_{j-\frac{1}{2}})u^-_{j-\frac{1}{2}} + \frac{1}{2}b^-_{j-\frac{1}{2}}u^+_{j-\frac{3}{2}}\right). \end{aligned}$$

Collecting the terms, we obtain:

$$\begin{aligned} D_j &= \frac{1}{2}\mu b^+_{j+\frac{1}{2}}u^-_{j+\frac{3}{2}} + \mu(\alpha - \frac{1}{2}b^+_{j+\frac{1}{2}})u^+_{j+\frac{1}{2}} + (\frac{1}{4} - \mu(\alpha + \frac{1}{2}b^+_{j-\frac{1}{2}} - \frac{1}{2}b^-_{j+\frac{1}{2}}))u^-_{j+\frac{1}{2}} \\ &\quad +(\frac{1}{4} - \mu(\alpha + \frac{1}{2}b^-_{j+\frac{1}{2}} - \frac{1}{2}b^+_{j-\frac{1}{2}}))u^+_{j-\frac{1}{2}} + \mu(\alpha - \frac{1}{2}b^-_{j-\frac{1}{2}})u^-_{j-\frac{1}{2}} + \frac{1}{2}\mu b^-_{j-\frac{1}{2}}u^+_{j-\frac{3}{2}} \end{aligned}$$

here $b^\pm_{j+\frac{1}{2}} = b(u^\pm_{j+\frac{1}{2}}) \geq 0$. *Collecting terms and using simple algebra, we can easily show that, if* $u^\pm_{j+\frac{1}{2}} \in [m, M]$ *for all* j, *then under the CFL condition specified in Table 2.2.1, the conclusion of Theorem (2.2.1) holds.*

Remark 2.2.5. *This* P^1*-DG result can also be extended to the LDG method for second order derivatives. To obtain an LDG formulation for (2.1.2), we first rewrite it as:*

$$u_t + f(u)_x = (b^*(u)q)_x, \quad q - B(u)_x = 0 \tag{2.2.19}$$

where $b^*(u) = \sqrt{b(u)}$, $B(u) = \int^u b^*(s)ds$. *We then seek* $u, q \in V^k_h$, *such that for test functions* $v, p \in V^k_h$*:*

$$\int_{I_j} u_t v dx - \int_{I_j} (f(u) - b^*(u)q)v_x dx + (\widehat{f} - \widehat{b^*}\widehat{q})_{j+\frac{1}{2}}v^-_{j+\frac{1}{2}} - (\widehat{f} - \widehat{b^*}\widehat{q})_{j-\frac{1}{2}}v^+_{j-\frac{1}{2}} = 0 \tag{2.2.20}$$

$$\int_{I_j} qp dx + \int_{I_j} B(u)p_x dx - \widehat{B}_{j+\frac{1}{2}}p^-_{j+\frac{1}{2}} + \widehat{B}_{j-\frac{1}{2}}p^+_{j-\frac{1}{2}} = 0 \tag{2.2.21}$$

where the diffusion fluxes are:

$$\widehat{b}^* = \frac{B(u^+) - B(u^-)}{u^+ - u^-}, \quad \widehat{q} = q^-, \quad , \widehat{B} = B(u^+).$$

As before, the diffusion term for (2.2.20) is:

$$D_j = \frac{1}{2}\overline{u}_j^n + \lambda(\widehat{b}^*_{j+\frac{1}{2}}\widehat{q}_{j+\frac{1}{2}} - \widehat{b}^*_{j-\frac{1}{2}}\widehat{q}_{j-\frac{1}{2}}) \tag{2.2.22}$$

where the $\widehat{q}$ is based on q, solved locally from (2.2.21). If we replace the volume integral by a quadrature rule with sufficient accuracy (see [12]), we have

$$q_j(x) = \frac{(B^+_{j+\frac{1}{2}} - B^+_{j-\frac{1}{2}})}{h} + 6\frac{(B^+_{j+\frac{1}{2}} - B^-_{j+\frac{1}{2}})}{h}\left(\frac{x - x_j}{h}\right)$$

Inserting it into (2.2.22), omitting the superscript n, we obtain:

$$\begin{aligned} D_j &= \frac{1}{4}u^-_{j+\frac{1}{2}} + \frac{1}{4}u^+_{j-\frac{1}{2}} + \mu\left[\xi_{j+\frac{1}{2}}(4B^+_{j+\frac{1}{2}} - 3B^-_{j+\frac{1}{2}} - B^+_{j-\frac{1}{2}})\right. \\ &\quad \left. -\xi_{j-\frac{1}{2}}(4B^+_{j-\frac{1}{2}} - 3B^-_{j-\frac{1}{2}} - B^+_{j-\frac{3}{2}})\right]. \end{aligned}$$

Here $\xi = \frac{[B]}{[u]} \geq 0$. If we further rearrange the terms of D_j, we have

$$\begin{aligned} D_j &= \frac{1}{4}u^-_{j+\frac{1}{2}} + \frac{1}{4}u^+_{j-\frac{1}{2}} + 3\mu\xi^2_{j+\frac{1}{2}}[u]_{j+\frac{1}{2}} - 3\mu\xi^2_{j-\frac{1}{2}}[u]_{j-\frac{1}{2}} \\ &\quad +\mu\xi_{j+\frac{1}{2}}(B^+_{j+\frac{1}{2}} - B^+_{j-\frac{1}{2}}) - \mu\xi_{j-\frac{1}{2}}(B^+_{j-\frac{1}{2}} - B^+_{j-\frac{3}{2}}) \end{aligned}$$

Notice that by definition, $B^+_{j+\frac{1}{2}} - B^+_{j-\frac{1}{2}} = \int_{u^+_{j-\frac{1}{2}}}^{u^+_{j+\frac{1}{2}}} b^*(s)ds$, $B^+_{j-\frac{1}{2}} - B^+_{j-\frac{3}{2}} = \int_{u^+_{j-\frac{3}{2}}}^{u^+_{j-\frac{1}{2}}} b^*(s)ds$, $b^* \geq 0$. By mean value thereom, we can find $u_1 \in \left(\min(u^+_{j+\frac{1}{2}}, u^+_{j-\frac{1}{2}}), \max(u^+_{j+\frac{1}{2}}, u^+_{j-\frac{1}{2}})\right)$, $u_2 \in \left(\min(u^+_{j-\frac{1}{2}}, u^+_{j-\frac{3}{2}}), \max(u^+_{j-\frac{1}{2}}, u^+_{j-\frac{3}{2}})\right)$, such that $B^+_{j+\frac{1}{2}} - B^+_{j-\frac{1}{2}} = b^*(u_1)(u^+_{j+\frac{1}{2}} -$

$u^+_{j-\frac{1}{2}}$) *and* $B^+_{j-\frac{1}{2}} - B^+_{j-\frac{3}{2}} = b^*(u_2)(u^+_{j-\frac{1}{2}} - u^+_{j-\frac{3}{2}})$. *Thus, by simple algebra we have,*

$$\begin{aligned} D_j &= \frac{1}{4}u^-_{j+\frac{1}{2}} + \frac{1}{4}u^+_{j-\frac{1}{2}} + 3\mu\xi^2_{j+\frac{1}{2}}[u]_{j+\frac{1}{2}} - 3\mu\xi^2_{j-\frac{1}{2}}[u]_{j-\frac{1}{2}} \\ &\quad +\mu\xi_{j+\frac{1}{2}}b^*(u_1)(u^+_{j+\frac{1}{2}} - u^+_{j-\frac{1}{2}}) - \mu\xi_{j-\frac{1}{2}}b^*(u_2)(u^+_{j-\frac{1}{2}} - u^+_{j-\frac{3}{2}}) \\ &= (3\mu\xi^2_{j+\frac{1}{2}} + \mu\xi_{j+\frac{1}{2}}b^*(u_1))u^+_{j+\frac{1}{2}} + (\frac{1}{4} - 3\mu\xi^2_{j+\frac{1}{2}})u^-_{j+\frac{1}{2}} \\ &\quad +(\frac{1}{4} - 3\mu\xi^2_{j-\frac{1}{2}} - \mu\xi_{j+\frac{1}{2}}b^*(u_1) - \mu\xi_{j-\frac{1}{2}}b^*(u_2))u^+_{j-\frac{1}{2}} \\ &\quad +3\mu\xi^2_{j-\frac{1}{2}}u^-_{j-\frac{1}{2}} + \mu\xi_{j-\frac{1}{2}}b^*(u_2)u^+_{j-\frac{3}{2}} \end{aligned}$$

Now if $u^\pm_{j+\frac{1}{2}} \in [m, M]$ *for all* j*, then under the CFL condition shown in Table 2.2.1, the conclusion for Theorem (2.2.1) holds.*

Table 2.2.1: The CFL conditions for (2.1.1) or (2.1.2) and restrictions for penalty coefficients in the one dimensional case.

Scheme	convection CFL	diffusion CFL	penalty coefficient
Cheng-Shu DG	$\lambda \max \lvert f'(u)\rvert \le \frac{1}{4}$	$\mu \le \frac{1}{4\max(a'(u))(\alpha+1)}$	$\alpha \ge 1$
IPDG	$\lambda \max \lvert f'(u)\rvert \le \frac{1}{4}$	$\mu \le \frac{1}{4\alpha+2\max(a'(u))}$	$\alpha \ge \frac{1}{2}\max(a'(u))$
LDG	$\lambda \max \lvert f'(u)\rvert \le \frac{1}{4}$	$\mu \le \frac{1}{20\max(a'(u))}$	–

Remark 2.2.6. *From Table 2.2.1, we note that the CFL conditions for maximum-principle-preserving DG schemes are comparable with or just slightly more restrictive than the standard time step restrictions for linear stability of DG methods for diffusion terms. For example, for the LDG method using piecewise linear polynomials and second order Runge-Kutta time stepping, the CFL number for linear stability is* $\mu \max(a'(u)) \le 0.055$ *and, according to Table 2.2.1, a sufficient condition is* $\mu \max(a'(u)) \le 0.05$ *for maintaining the maximum-principle-preserving property.*

2.2.3 Implementation

To enforce the sufficient conditions in Theorem 2.2.1, i.e., $u^\pm_{j\pm\frac{1}{2}}$ at time level n, we can use the same limiter as in [75]. Given approximation polynomials $p_j(x)$ evolved

from a P^1-DG scheme, with its cell averages $\overline{u}_j^n \in [m, M]$, we seek to modify $p_j(x)$ into $\tilde{p}_j(x)$ by

$$\tilde{p}_j(x) = \theta(p_j(x) - \overline{u}_j^n) + \overline{u}_j^n, \quad \theta = \min\left\{1, \left|\frac{M - \overline{u}_j^n}{M_j - \overline{u}_j^n}\right|, \left|\frac{m - \overline{u}_j^n}{m_j - \overline{u}_j^n}\right|\right\}, \tag{2.2.23}$$

where

$$M_j = \max\{p_j(x_{j+\frac{1}{2}}), p_j(x_{j-\frac{1}{2}})\}, \qquad m_j = \min\{p_j(x_{j+\frac{1}{2}}), p_j(x_{j-\frac{1}{2}})\}.$$

It is clear that $\tilde{p}_j(x_{j\pm\frac{1}{2}}) \in [m, M]$ and $\tilde{p}_j(x)$ has the same cell average $\overline{u}_j^n$. Moreover, we have $||\tilde{p}_j - p_j||_\infty = \mathcal{O}(h^2)$ if the exact solution is smooth, [75].

For forward Euler time discretization, at time level n, assume the P^1-DG polynomial in cell I_j is $p_j(x)$, and the cell average of $p_j(x)$ is $\overline{u}_j^n \in [m, M]$, then

- In each cell, monitor the point values of $p_j(x)$ at the two end points $x_{j\pm\frac{1}{2}}$ by computing θ in (2.2.23).
- If $\theta = 1$, leave the DG solution $p_j(x)$ unchanged.
- If $\theta < 1$, use $\tilde{p}_j(x)$ instead of $p_j(x)$ in the DG scheme with forward Euler in time under the CFL condition in Table 2.2.1.

For SSP high order time discretizations, we need to use the limiter in each stage of a Runge-Kutta method or in each step of a multistep method. Note that the CFL conditions in Table 2.2.1 are sufficient but not necessary to ensure the maximum principle. A more efficient implementation would be to enforce the more restrictive CFL conditions in Table 2.2.1 only when a preliminary calculation to next time step with a normal time step violates the maximum principle. More details will be offered in Section 2.3.3 regarding this.

2.3 Maximum-principle-satisfying DG schemes in two dimensions

2.3.1 Preliminaries

In this section, we extend our results to P^1-DG schemes on triangular meshes for solving the initial value problem for two-dimensional nonlinear convection-diffusion equations with a general form

$$\frac{\partial u}{\partial t} + \nabla \cdot \mathbf{F}(u) = \nabla \cdot (\mathbf{A}\nabla u), \quad \mathbf{F}(u) = \langle f(u), g(u) \rangle, \tag{2.3.1}$$

where $\mathbf{A} = \mathbf{A}(u, x)$ is a 2×2 symmetric semi-positive-definite matrix. Let $[m, M]$ denote the desired range indicated by the initial condition. Again, we will assume periodic boundary condition for simplicity and all the results regarding the maximum principle can be easily incorporated into any initial-boundary value problems.

We first review how to construct maximum-principle-satisfying high order DG schemes for hyperbolic conservation laws

$$u_t + \nabla \cdot \mathbf{F}(u) = 0, \quad \mathbf{F}(u) = \langle f(u), g(u) \rangle, \tag{2.3.2}$$

on triangular meshes, as explained by Zhang et al. in details in [79]. Given a triangulation T_h, for each triangle K, denote by $|K|$ the area of the triangle and l_K^i $(i = 1, 2, 3)$ the length of the corresponding edges e_K^i $(i = 1, 2, 3)$, with outward unit normal vectors being $\vec{\nu}_i$ $(i = 1, 2, 3)$. Denote also the neighboring triangle along e_K^i as K_i. Any one-dimensional monotone flux can be used, for example the global

Lax-Friedrichs flux

$$\widehat{f}(u,v,\vec{\nu}) = \frac{1}{2}(\mathbf{F}(u)\cdot\vec{\nu} + \mathbf{F}(v)\cdot\vec{\nu} - a(v-u)), \quad a = \max_u |\mathbf{F}'(u)\cdot\vec{\nu}|. \tag{2.3.3}$$

The first order Lax-Friedrichs scheme for (2.3.2) on triangular meshes is defined as:

$$u_K^{n+1} = u_K^n - \frac{\Delta t}{|K|}\sum_{i=1}^{3}\widehat{f}(u_K^n, u_{K_i}^n, \vec{\nu}_i)l_K^i$$

By the same token, the scheme for the cell averages of the DG schemes for (2.3.2) can be written as:

$$\overline{u}_K^{n+1} = \overline{u}_K^n - \frac{\Delta t}{|K|}\sum_{i=1}^{3}\int_{e_K^i}\widehat{f}(u_i^{in}, u_i^{out}, \vec{\nu}_i)ds$$

where $\overline{u}_K$ is the cell average of the numerical solution over the triangle K, u_i^{in} and u_i^{out} are the values of the numerical solution on the edge e_K^i evaluated from the DG polynomials defined in K and in K_i respectively. Given the approximation space $\mathrm{P}^k(K)$, with sufficient accuracy, the line integrals along the edges can be approximated by the $(k+1)$-point Gauss quadrature rules:

$$\overline{u}_K^{n+1} = \overline{u}_K^n - \frac{\Delta t}{|K|}\sum_{i=1}^{3}\sum_{\beta=1}^{k+1}\widehat{f}(u_K^{i,\beta}, u_{K_i}^{i,\beta}, \vec{\nu}_i)\omega_\beta l_K^i \tag{2.3.4}$$

where ω_β denotes the quadrature weights for the $(k+1)$-point Gauss quadrature rule on $[-\frac{1}{2}, \frac{1}{2}]$, and $u_K^{i,\beta}$ denotes the value of the DG polynomial defined on K, evaluated at the β-th Gauss quadrature point on the edge e_K^i.

The strategy in [79] is to express the right hand side of (2.3.4) as a monotonically increasing function with respect to certain point values of the DG polynomials, i.e., the maximum principle can be enforced by the methodology for the one-dimensional

case. These points, denoted by S_K^k on each triangle K, contain the Gauss quadrature points on the three edges. For the DG method with polynomials of arbitrary degree k, S_K^k can be constructed by taking the Dubiner transformation of the tensor product of Gauss-Lobatto and Gauss quadratures, which is then summarized (see [79] for details). In particular, for the P^1-DG method under consideration in this section, the set S_K^1 is defined as the six points shown in Figure 2.3.1 where the symbols denote the 2-point Gaussian quadrature points for each edge.

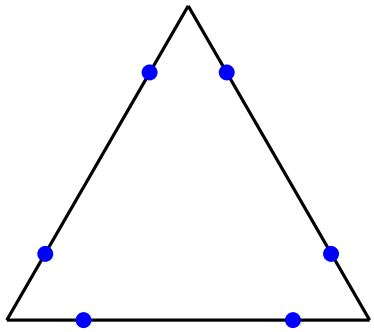


Figure 2.3.1: The local stencil S_K^1 for P^1-DG on a triangle K

Let $u_K(\mathbf{x})$ denote the DG polynomial on the cell K. If the point values of $u_K(\mathbf{x})$ on the three vertices are in the range $[m, M]$ for each K at time level n, then the point values of $u_K(\mathbf{x})$ on any points inside K including on the stencil S_K^1 are in the range $[m, M]$ due to the linearity. This follow the result in [79] and the solutions of (2.3.4) satisfy $\overline{u}_K^{n+1} \in [m, M]$ under the CFL condition

$$a\frac{\Delta t}{|K|}\sum_{i=1}^{3} l_K^i \leq \frac{1}{3}. \tag{2.3.5}$$

Therefore, to seek a sufficient condition to ensure the maximum principle for the convection-diffusion equations, it suffices to focus only on the diffusion part. Straightforward extension to convection-diffusion equations can then be obtained using a similar cell average splitting approach as for the one-dimensional case. In the following subsection, we will first consider the two-dimensional diffusion equation, i.e.,

(2.3.1) with $\mathbf{F} \equiv \mathbf{0}$.

2.3.2 Monotonicity of the scheme for the diffusion part

We consider the Cheng-Shu DG scheme in [10] for (2.3.1) with $\mathbf{F} \equiv \mathbf{0}$ as an example. Similar analysis can be performed for the IPDG scheme and the LDG scheme as in the one dimensional case. Given a triangulation T_h, define the approximation space as: $V_h^k = \{v : v|_K \in P^k(K)\}$. The DG scheme is to seek $u \in V_h^k$, such that for any test function $\varphi \in V_h^k$, the following equality holds:

$$\frac{\partial}{\partial t}\iint_K u\varphi d\mathbf{x} = \int_{\partial K} \widehat{(\mathbf{A}\nabla u \cdot \vec{\nu})}\varphi ds - \int_{\partial K} \left(\widetilde{u\mathbf{A}}\nabla\varphi\right)\cdot\vec{\nu} ds + \iint_K u\nabla\cdot(\mathbf{A}\nabla\varphi)d\mathbf{x}. \tag{2.3.6}$$

The numerical fluxes on each edge e_K^i are:

$$\widehat{(\mathbf{A}\nabla u \cdot \vec{\nu})} = \mathbf{A}(u^-)\nabla u^- \cdot \vec{\nu} + \frac{\alpha_{\mathbf{A}}}{l_K^i}(u^{out} - u^{in}), \quad \widetilde{u\mathbf{A}} = u^+\mathbf{A}(u^+), \tag{2.3.7}$$

$$\alpha_{\mathbf{A}} = \lambda_{\mathbf{A}}\alpha, \quad \lambda_{\mathbf{A}} = \max_{u,x} \parallel \mathbf{A}(u,x) \parallel,$$

where the maximum can be taken either locally on each edge or globally over the whole domain, $\vec{\nu}$ is the outward unit normal vector on the edges, $||\mathbf{A}||$ is the matrix 2-norm or the spectral radius (maximum eigenvalue in absolute value) for the symmetric matrix $\mathbf{A}$, α is a sufficiently large positive constant to ensure stability. $u^{\pm}$ denotes the numerical solution on the edges, evaluated from K or K_i. The '$\pm$' for each edge e_K^i is determined by the inner product of $\vec{\nu}_i$ and a predetermined constant vector $\vec{\nu}_0$ which is not parallel to any edge in the mesh: for each edge e_K^i in the cell K,

$$u^- = u_K, \quad u^+ = u_{K_i} \quad if\ \vec{\nu}_0 \cdot \vec{\nu}_i < 0;$$

$$u^+ = u_K, \quad u^- = u_{K_i} \quad if\ \vec{\nu}_0 \cdot \vec{\nu}_i > 0.$$

Taking $\varphi = 1$ on K and $\varphi = 0$ everywhere else,

$$\frac{\partial}{\partial t}\overline{u}_K = \frac{1}{|K|}\int_{\partial K} (\widehat{\mathbf{A}\nabla u \cdot \vec{\nu}})ds.$$

Therefore, the scheme satisfied by the cell averages of the DG method with forward Euler time integration on a triangular mesh is

$$\overline{u}_K^{n+1} = \overline{u}_K^n + \frac{\Delta t}{|K|}\int_{\partial K} (\widehat{\mathbf{A}\nabla u^n \cdot \vec{\nu}})ds.$$

The integral can be approximated by a 2-point Gauss quadrature with sufficient accuracy for P^1-DG. We use $\mathbf{x}_{i,\beta}$ $(\beta = 1, 2)$ to denote the local Gauss quadrature points for each edge e_K^i. Let ω_β $(\beta = 1, 2)$ denote the corresponding quadrature weights for the 2-point Gauss quadrature on the interval $[-\frac{1}{2}, \frac{1}{2}]$. The scheme then becomes (omitting the superscripts for time level n on the right hand side):

$$\overline{u}_K^{n+1} = \overline{u}_K^n + \frac{\Delta t}{|K|}\sum_{i=1}^{3} l_K^i \sum_{\beta=1}^{2} \omega_\beta (\widehat{\mathbf{A}\nabla u \cdot \vec{\nu}_i})\Big|_{\mathbf{x}=\mathbf{x}_{i,\beta}}. \tag{2.3.8}$$

Now the idea is to rewrite the right hand side above as a convex combination of certain point values. To find an explicit expression for the flux $(\widehat{\mathbf{A}\nabla u \cdot \vec{\nu}_i})$, we consider two different cases.

- **Case A:** $u^- = u^{out}$, $u^+ = u^{in}$. For this case, we have $u_{i,\beta}^- = u_{K_i}(\mathbf{x}_{i,\beta})$. The subscript or superscript (i, β, K_i) for $\mathbf{A}$ and ∇u will denote the value evaluated at $u_{K_i}(\mathbf{x}_{i,\beta})$. Then we have

$$(\widehat{\mathbf{A}\nabla u \cdot \vec{\nu}_i})\Big|_{\mathbf{x}=\mathbf{x}_{i,\beta}} = \mathbf{A}_{K_i}^{i,\beta}\nabla u_{K_i}^{i,\beta} \cdot \vec{\nu}_i + \frac{\alpha_{\mathbf{A}}}{l_K^i}(u_{K_i}^{i,\beta} - u_K^{i,\beta}) \tag{2.3.9}$$

Let $\vec{v}_{i,\beta} = \mathbf{A}_{K_i}^{i,\beta}\vec{\nu}_i$, then $\vec{v}_{i,\beta} \cdot \vec{\nu}_i = \mathbf{A}_{K_i}^{i,\beta}\vec{\nu}_i \cdot \vec{\nu}_i \geq 0$ because $\mathbf{A}$ is semi-positive definite. Namely, $\vec{v}_{i,\beta}$ is a direction pointing into K_i or along the edge e_K^i. Since $\mathbf{A}$ is symmetric, we have a directional derivative

$$\mathbf{A}_{K_i}^{i,\beta}\nabla u_{K_i}^{i,\beta} \cdot \vec{\nu}_i = \nabla u_{K_i}^{i,\beta} \cdot \left(\mathbf{A}_{K_i}^{i,\beta}\vec{\nu}_i\right) = \left.\frac{\partial u_{K_i}(\mathbf{x})}{\partial \vec{v}_{i,\beta}}\right|_{\mathbf{x}=\mathbf{x}_{i,\beta}}.$$

For each point $\mathbf{x}_{i,\beta}$ on the edge e_K^i, consider the straight line given by the parametric equation $\mathbf{r}(t) = t\vec{v}_{i,\beta} + \mathbf{x}_{i,\beta}, t \in \mathbb{R}$, which intersects with the other two edges of K_i at one point denoted by $\mathbf{x}_{i,\beta}^*$. See Figure 2.3.2 (a) for an illustration.

By linearity of $u_{K_i}(\mathbf{x})$, the directional derivative can be written as:

$$\left.\frac{\partial u_{K_i}(\mathbf{x})}{\partial \vec{v}_{i,\beta}}\right|_{\mathbf{x}=\mathbf{x}_{i,\beta}} = \frac{u_{K_i}(\mathbf{x}_{i,\beta}^*) - u_{K_i}^{i,\beta}}{||\mathbf{x}_{i,\beta} - \mathbf{x}_{i,\beta}^*||}. \tag{2.3.10}$$

Inserting (2.3.10) into (2.3.9), we obtain:

$$\left.\widehat{(\mathbf{A}\nabla u \cdot \vec{\nu}_i)}\right|_{\mathbf{x}=\mathbf{x}_{i,\beta}} = \frac{1}{||\mathbf{x}_{i,\beta} - \mathbf{x}_{i,\beta}^*||}u_{K_i}(\mathbf{x}_{i,\beta}^*) + \left(\frac{\alpha_{\mathbf{A}}}{l_K^i} - \frac{1}{||\mathbf{x}_{i,\beta} - \mathbf{x}_{i,\beta}^*||}\right)u_{K_i}^{i,\beta} - \frac{\alpha_{\mathbf{A}}}{l_K^i}u_K^{i,\beta}. \tag{2.3.11}$$

- **Case B:** $u^+ = u^{out}$, $u^- = u^{in}$. For this case, we have $u_{i,\beta}^- = u_K(\mathbf{x}_{i,\beta})$. Then

$$\left.\widehat{(\mathbf{A}\nabla u \cdot \vec{\nu}_i)}\right|_{\mathbf{x}=\mathbf{x}_{i,\beta}} = \mathbf{A}_K^{i,\beta}\nabla u_K^{i,\beta} \cdot \vec{\nu}_i + \frac{\alpha_{\mathbf{A}}}{l_K^i}(u_{K_i}^{i,\beta} - u_K^{i,\beta})$$

Let $\vec{v}_{i,\beta} = -\mathbf{A}_K^{i,\beta}\vec{\nu}_i$, then $\vec{v}_{i,\beta} \cdot \vec{\nu}_i = -\mathbf{A}_K^{i,\beta}\vec{\nu}_i \cdot \vec{\nu}_i \leq 0$. Here $\vec{v}_{i,\beta}$ is a direction pointing into K or along the edge e_K^i, and the directional derivative is

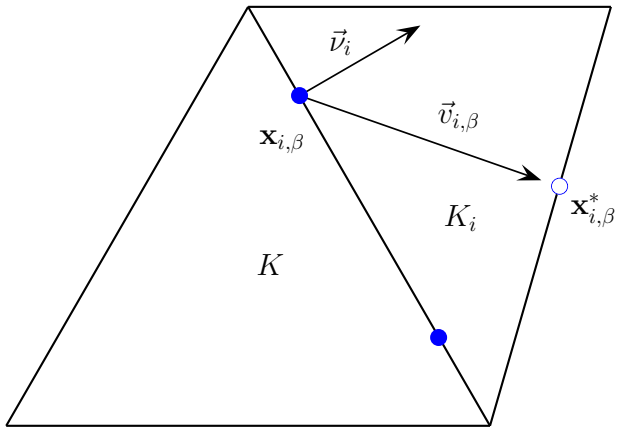
$$\mathbf{A}_K^{i,\beta}\nabla u_K^{i,\beta} \cdot \vec{\nu}_i = \nabla u_K^{i,\beta} \cdot \left(\mathbf{A}_K^{i,\beta}\vec{\nu}_i\right) = \nabla u_K^{i,\beta} \cdot (-\vec{v}_{i,\beta}) = -\left.\frac{\partial u_K(\mathbf{x})}{\partial \vec{v}_{i,\beta}}\right|_{\mathbf{x}=\mathbf{x}_{i,\beta}}.$$

For each point $\mathbf{x}_{i,\beta}$ along the edge e_K^i, consider the straight line given by the

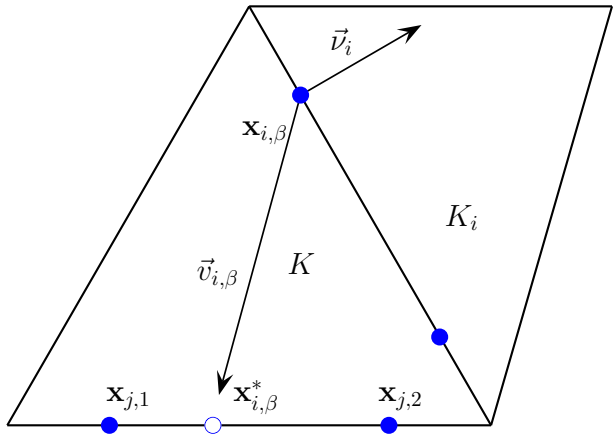
parametric equation $\mathbf{r}(t) = t\vec{v}_{i,\beta} + \mathbf{x}_{i,\beta}, t \in \mathbb{R}$, which intersects with the other two edges of K at one point denoted by $\mathbf{x}^*_{i,\beta}$. See Figure 2.3.2 (b) for an illustration. Assume $\mathbf{x}^*_{i,\beta}$ lies on the edge e^j_K $(j \neq i)$. Since $u_K(\mathbf{x})$ is a linear function, $u_K(\mathbf{x}^*_{i,\beta})$ can be represented as a linear combination of point values at the two Gauss quadrature points $u_K^{j(i,\beta),1}, u_K^{j(i,\beta),2}$ on the edge e^j_K where j depends on i and β:

$$u_K(\mathbf{x}^*_{i,\beta}) = c^1_{i,\beta} u_K^{j(i,\beta),1} + c^2_{i,\beta} u_K^{j(i,\beta),2} \tag{2.3.12}$$

with $c^1_{i,\beta}, c^2_{i,\beta} \in [-\frac{\sqrt{3}-1}{2}, \frac{\sqrt{3}+1}{2}]$ and $c^1_{i,\beta} + c^2_{i,\beta} = 1$.



(a) Case A



(b) Case B

Figure 2.3.2: Two cases for the numerical flux on triangular meshes

By linearity of $u_K(\mathbf{x})$, the directional derivative can be written as:

$$\left.\frac{\partial u_K(\mathbf{x})}{\partial \vec{v}_{i,\beta}}\right|_{\mathbf{x}=\mathbf{x}_{i,\beta}} = \frac{u_K(\mathbf{x}^*_{i,\beta}) - u_K^{i,\beta}}{||\mathbf{x}_{i,\beta} - \mathbf{x}^*_{i,\beta}||}.$$

Thus we obtain:

$$\left.(\widehat{\mathbf{A}\nabla u \cdot \vec{\nu}_i})\right|_{\mathbf{x}=\mathbf{x}_{i,\beta}} = -\frac{1}{||\mathbf{x}_{i,\beta} - \mathbf{x}^*_{i,\beta}||}u_K(\mathbf{x}^*_{i,\beta}) - \left(\frac{\alpha_{\mathbf{A}}}{l_K^i} - \frac{1}{||\mathbf{x}_{i,\beta} - \mathbf{x}^*_{i,\beta}||}\right) u_K^{i,\beta} + \frac{\alpha_{\mathbf{A}}}{l_K^i}u_{K_i}^{i,\beta}. \tag{2.3.13}$$

By the exactness of the quadrature in Figure 2.3.1 for the linear polynomial $u_K(\mathbf{x})$ on the triangle K (see [79] for construction of this quadrature), we can decompose the cell average $\overline{u}_K^n$ as a convex combination of the point values on the stencil S_K^1,

$$\overline{u}_K^n = \frac{1}{|K|}\iint_K u_K(\mathbf{x})d\mathbf{x} = \sum_{i=1}^{3}\sum_{\beta=1}^{2}\frac{1}{3}\omega_\beta u_K^{i,\beta}. \tag{2.3.14}$$

Let $h_{i,\beta}$ denote $||\mathbf{x}_{i,\beta} - \mathbf{x}^*_{i,\beta}||$. Combining (2.3.11), (2.3.12), (2.3.13) and (2.3.14), we can rewrite (2.3.8) as:

$$\begin{aligned}
\overline{u}_K^{n+1} &= \overline{u}_K^n + \frac{\Delta t}{|K|}\sum_{i=1}^{3} l_K^i \sum_{\beta=1}^{2}\omega_\beta \left.(\widehat{\mathbf{A}\nabla u \cdot \vec{\nu}_i})\right|_{\mathbf{x}=\mathbf{x}_{i,\beta}} \\
&= \frac{1}{2}\overline{u}_K^n + \frac{1}{2}\overline{u}_K^n + \frac{\Delta t}{|K|}\sum_{i=1}^{3} l_K^i \sum_{\beta=1}^{2}\omega_\beta \left.(\widehat{\mathbf{A}\nabla u \cdot \vec{\nu}_i})\right|_{\mathbf{x}=\mathbf{x}_{i,\beta}} \\
&= \sum_{i=1}^{3}\sum_{\beta=1}^{2}\omega_\beta\left(\frac{1}{6}u_K^{i,\beta} + \Delta t\frac{l_K^i}{|K|}\left.(\widehat{\mathbf{A}\nabla u \cdot \vec{\nu}_i})\right|_{\mathbf{x}=\mathbf{x}_{i,\beta}}\right) + \frac{1}{2}\overline{u}_K^n \\
&= \sum_{i\in CaseA}\sum_{\beta=1}^{2}\omega_\beta\left[\left(\frac{1}{6} - \Delta t\frac{\alpha_{\mathbf{A}}}{|K|}\right)u_K^{i,\beta} + \frac{\Delta t}{|K|}\frac{l_K^i}{h_{i,\beta}}u_{K_i}(\mathbf{x}^*_{i,\beta})\right. \\
&\quad \left. + \frac{\Delta t}{|K|}\left(\alpha_{\mathbf{A}} - \frac{l_K^i}{h_{i,\beta}}\right)u_{K_i}^{i,\beta}\right] \\
&+ \sum_{i\in CaseB}\sum_{\beta=1}^{2}\omega_\beta\left[\left(\frac{1}{6} - \Delta t\frac{\alpha_{\mathbf{A}}}{|K|} + \frac{\Delta t}{|K|}\frac{l_K^i}{h_{i,\beta}}\right)u_K^{i,\beta} + \Delta t\frac{\alpha_{\mathbf{A}}}{|K|}u_{K_i}^{i,\beta}\right]
\end{aligned}$$

$$+ \sum_{i\in CaseB}\sum_{\beta=1}^{2}\omega_\beta\left(-\frac{\Delta t}{|K|}\frac{l_K^i}{h_{i,\beta}}u_K(\mathbf{x}_{i,\beta}^*)\right)+\frac{1}{2}\overline{u}_K^n. \tag{2.3.15}$$

Note that we can express the cell average as

$$\overline{u}_K^n=\frac{1}{3}\sum_{i=1}^{3}\overline{u}_K^n=\frac{1}{3}\left(\sum_{i\in CaseA}\overline{u}_K^n+\sum_{i\in CaseB}\overline{u}_K^n\right)=\frac{1}{3}\left(\sum_{i\in CaseA}\overline{u}_K^n+\sum_{i\in CaseB}\sum_{\beta=1}^{2}\omega_\beta\overline{u}_K^n\right),$$

so the third term in (2.3.15) can be further expanded:

$$\begin{aligned}
&\sum_{i\in CaseB}\sum_{\beta=1}^{2}\omega_\beta\left(-\frac{\Delta t}{|K|}\frac{l_K^i}{h_{i,\beta}}u_K(\mathbf{x}_{i,\beta}^*)\right)+\frac{1}{2}\overline{u}_K^n\\
=&\sum_{i\in CaseB}\sum_{\beta=1}^{2}\omega_\beta\left(-\frac{\Delta t}{|K|}\frac{l_K^i}{h_{i,\beta}}u_K(\mathbf{x}_{i,\beta}^*)\right)+\frac{1}{6}\left(\sum_{i\in CaseA}\overline{u}_K^n+\sum_{i\in CaseB}\sum_{\beta=1}^{2}\omega_\beta\overline{u}_K^n\right)\\
=&\sum_{i\in CaseB}\sum_{\beta=1}^{2}\omega_\beta\left(-\frac{\Delta t}{|K|}\frac{l_K^i}{h_{i,\beta}}u_K(\mathbf{x}_{i,\beta}^*)\right)+\frac{1}{6}\sum_{i\in CaseB}\sum_{\beta=1}^{2}\omega_\beta\sum_{k=1}^{3}\sum_{\gamma=1}^{2}\frac{1}{3}\omega_\gamma u_K^{k,\gamma}\\
&+\frac{1}{6}\sum_{i\in CaseA}\overline{u}_K^n\\
=&\sum_{i\in CaseB}\sum_{\beta=1}^{2}\omega_\beta\left[\frac{1}{18}\sum_{k=1}^{3}\sum_{\gamma=1}^{2}\omega_\gamma u_K^{k,\gamma}-\frac{\Delta t}{|K|}\frac{l_K^i}{h_{i,\beta}}\left(c_{i,\beta}^1u_K^{j(i,\beta),1}+c_{i,\beta}^2u_K^{j(i,\beta),2}\right)\right]\\
&+\frac{1}{6}\sum_{i\in CaseA}\overline{u}_K^n\\
=&\sum_{i\in CaseB}\sum_{\beta=1}^{2}\omega_\beta\left[\frac{1}{18}\sum_{k\neq j}\sum_{\gamma=1}^{2}\omega_\gamma u_K^{k,\gamma}+\left(\frac{1}{36}-\frac{\Delta t}{|K|}\frac{l_K^i}{h_{i,\beta}}c_{i,\beta}^1\right)u_K^{j,1}\right.\\
&\left.+\left(\frac{1}{36}-\frac{\Delta t}{|K|}\frac{l_K^i}{h_{i,\beta}}c_{i,\beta}^2\right)u_K^{j,2}\right]+\frac{1}{6}\sum_{i\in CaseA}\overline{u}_K^n,
\end{aligned}$$

where in the last step we use the fact that $\omega_1=\omega_2=\frac{1}{2}$. Finally, we have an equivalent formulation of (2.3.8):

$$\begin{aligned}
\overline{u}_K^{n+1}=&\sum_{i\in CaseA}\sum_{\beta=1}^{2}\omega_\beta\left[\left(\frac{1}{6}-\Delta t\frac{\alpha_{\mathbf{A}}}{|K|}\right)u_K^{i,\beta}+\frac{\Delta t}{|K|}\frac{l_K^i}{h_{i,\beta}}u_{K_i}(\mathbf{x}_{i,\beta}^*)\right.\\
&\left.+\frac{\Delta t}{|K|}\left(\alpha_{\mathbf{A}}-\frac{l_K^i}{h_{i,\beta}}\right)u_{K_i}^{i,\beta}\right]
\end{aligned}$$

$$
\begin{aligned}
&+ \sum_{i\in CaseB}\sum_{\beta=1}^{2}\omega_\beta\left[\left(\frac{1}{6}-\Delta t\frac{\alpha_{\mathbf{A}}}{|K|}+\frac{\Delta t}{|K|}\frac{l_K^i}{h_{i,\beta}}\right)u_K^{i,\beta}+\Delta t\frac{\alpha_{\mathbf{A}}}{|K|}u_{K_i}^{i,\beta}\right]+\frac{1}{6}\sum_{i\in CaseA}\overline{u}_K^n\\
&+ \sum_{i\in CaseB}\sum_{\beta=1}^{2}\omega_\beta\left[\frac{1}{18}\sum_{k\neq j}\sum_{\gamma=1}^{2}\omega_\gamma u_K^{k,\gamma}+\left(\frac{1}{36}-\frac{\Delta t}{|K|}\frac{l_K^i}{h_{i,\beta}}c_{i,\beta}^1\right)u_K^{j,1}\right.\\
&\quad\left.+\left(\frac{1}{36}-\frac{\Delta t}{|K|}\frac{l_K^i}{h_{i,\beta}}c_{i,\beta}^2\right)u_K^{j,2}\right].
\end{aligned}
\tag{2.3.16}
$$

To have a convex combination in (2.3.16), it suffices to require that following terms

$$
\frac{1}{6}-\Delta t\frac{\alpha_{\mathbf{A}}}{|K|},\quad \alpha_{\mathbf{A}}-\frac{l_K^i}{h_{i,\beta}},\quad \frac{1}{36}-\frac{\Delta t}{|K|}\frac{l_K^i}{h_{i,\beta}}c_{i,\beta}^{1,2},
\tag{2.3.17}
$$

are non-negative, which imply a time step restriction. Next, we derive an explicit CFL condition for a regular triangulation, i.e., a triangulation with the minimum angle bounded uniformly away from zero. Recall that for each triangle K, $\mathbf{x}_{i,\beta}$ ($\beta = 1, 2$) are the two Gaussian quadrature points on the edge e_K^i. By a simple geometric relationship, we have,

$$
h_{i,\beta}\geq\tilde{C}\sin(\tilde{\theta})l_K^i,\qquad \frac{l_K^i}{h_{i,\beta}}\leq\frac{1}{\tilde{C}\sin(\tilde{\theta})}
\tag{2.3.18}
$$

Here $\tilde{C}=\frac{D(\mathbf{x}_{i,\beta})}{l_K^i}$, with $D(\mathbf{x}_{i,\beta})$ denotes the distance from the point $\mathbf{x}_{i,\beta}$ to vertices of the edge e_K^i. Notice that there is a uniform lower bound enforcing $\tilde{C}\geq\frac{3-\sqrt{3}}{6}$, since $\mathbf{x}_{i,\beta}$ are 2-point Gauss quadrature points on e_K^i. $\tilde{\theta}$ here denotes the angle between the edge e_K^i and e_K^j or $e_{K_i}^j$ ($j\neq i$) on which the point $\mathbf{x}_{i,\beta}^*$ lies on. Thus, we let $\theta_K^{i,j}$, $i,j\in 1,2,3$, $i\neq j$ denote the angles between edges e_K^i and e_K^j. Notice that $c_{i,\beta}^{1,2}\in[-\frac{\sqrt{3}-1}{2},\frac{\sqrt{3}+1}{2}]$, it suffices to have the following conditions for the terms in (2.3.17) to be non-negative,

$$
\alpha_{\mathbf{A}}\geq\frac{1}{\tilde{C}\min_{K,i,j}\left(\sin(\theta_K^{i,j})\right)},\qquad \Delta t\frac{\alpha_{\mathbf{A}}}{|K|}\leq\frac{1}{6},
$$

$$\Delta t \frac{1}{\tilde{C} \min_{K,i,j} \left(\sin(\theta_K^{i,j})\right)} \frac{1}{|K|} \leq \frac{\sqrt{3}-1}{36},$$

which can be simplified as

$$\alpha_{\mathbf{A}} \geq (3+\sqrt{3}) \frac{1}{\min_{K,i,j} \left(\sin(\theta_K^{i,j})\right)}, \qquad \Delta t \frac{\alpha_{\mathbf{A}}}{|K|} \leq \frac{\sqrt{3}-1}{36}. \tag{2.3.19}$$

2.3.3 A second order maximum principle satisfying DG algorithm

For a given triangulation T_h, define the approximation space as: $V_h^k = \{v : v|_K \in P^k(K)\}$. The DG scheme for a convection-diffusion problem (2.3.1) is to seek $u \in V_h^k$, such that for any test function $\varphi \in V_h^k$, the following equality holds:

$$\begin{aligned} \frac{\partial}{\partial t} \iint_K u\varphi d\mathbf{x} &= \iint_K \mathbf{F} \cdot \nabla\varphi d\mathbf{x} - \int_{\partial K} \widehat{\mathbf{F} \cdot \vec{\nu}} \varphi ds \\ &+ \int_{\partial K} (\widehat{\mathbf{A}\nabla u \cdot \vec{\nu}}) \varphi ds - \int_{\partial K} \left(\widetilde{u\mathbf{A}} \nabla\varphi\right) \cdot \vec{\nu} ds + \iint_K u \nabla \cdot (\mathbf{A}\nabla\varphi) d\mathbf{x}, \end{aligned} \tag{2.3.20}$$

with the numerical fluxes:

$$\begin{aligned} \widehat{\mathbf{F} \cdot \vec{\nu}} &= \frac{1}{2}\left(\mathbf{F}(u^{in}) \cdot \vec{\nu} + \mathbf{F}(u^{out}) \cdot \vec{\nu} - a(u^{out} - u^{in})\right), \quad a = \max_{u,\vec{\nu}} |\mathbf{F}'(u) \cdot \vec{\nu}|, \\ (\widehat{\mathbf{A}\nabla u \cdot \vec{\nu}}) &= \mathbf{A}(u^-)\nabla u^- \cdot \vec{\nu} + \frac{\alpha_{\mathbf{A}}}{l_K^i}(u^{out} - u^{in}), \quad \widetilde{u\mathbf{A}} = u^+ \mathbf{A}(u^+). \end{aligned}$$

For forward Euler time stepping, take the test function as the indicator function on the cell K and replace the line integral with 2-point Gauss quadrature. This result

in the cell average scheme for P^1-DG method:

$$\overline{u}_K^{n+1} = \overline{u}_K^n - \frac{\Delta t}{|K|}\sum_{i=1}^{3} l_K^i \sum_{\beta=1}^{2} \omega_\beta (\widehat{\mathbf{F}\cdot\vec{\nu}})_{i,\beta} + \frac{\Delta t}{|K|}\sum_{i=1}^{3} l_K^i \sum_{\beta=1}^{2} \omega_\beta (\widehat{\mathbf{A}\nabla u\cdot\vec{\nu}_i})\Big|_{\mathbf{x}=\mathbf{x}_{i,\beta}}. \tag{2.3.21}$$

We can decompose (2.3.21) as $\overline{u}_K^{n+1} = \mathbf{C}_K + \mathbf{D}_K$ with

$$\mathbf{C}_K = \frac{1}{2}\overline{u}_K^n - \frac{\Delta t}{|K|}\sum_{i=1}^{3} l_K^i \sum_{\beta=1}^{2} \omega_\beta (\widehat{\mathbf{F}\cdot\vec{\nu}})_{i,\beta} \tag{2.3.22}$$

$$\mathbf{D}_K = \frac{1}{2}\overline{u}_K^n + \frac{\Delta t}{|K|}\sum_{i=1}^{3} l_K^i \sum_{\beta=1}^{2} \omega_\beta (\widehat{\mathbf{A}\nabla u\cdot\vec{\nu}_i})\Big|_{\mathbf{x}=\mathbf{x}_{i,\beta}} \tag{2.3.23}$$

By the results in [79], the convection part (2.3.22) is a monotonically increasing function of point values of DG polynomials under the CFL condition $a\frac{\Delta t}{|K|}\sum_{i=1}^{3} l_K^i \le \frac{1}{6}$. By the discussion in the previous subsection, the diffusion part (2.3.23) is a convex combination of point values of DG polynomials under the constraints $\alpha_{\mathbf{A}} \ge (3+\sqrt{3})\frac{1}{\min_{K,i,j}\left(\sin(\theta_K^{i,j})\right)}$, $\Delta t\frac{\alpha_{\mathbf{A}}}{|K|} \le \frac{\sqrt{3}-1}{72}$, with one half smaller time step than (2.3.19) since the cell average $\overline{u}_K^n$ is halved in (2.3.23).

Notice that at time level n, for the linear DG polynomials $u_K(\mathbf{x})$, if $u_K(\mathbf{x}) \in [m, M]$ at three vertices of the triangle K, then $u_K(\mathbf{x}) \in [m, M]$ for any $\mathbf{x} \in K$. Thus, we have,

Theorem 2.3.1. *Consider the P^1-DG scheme (2.3.20). If the DG polynomial at time level n on each triangle K satisfies $u_K(\mathbf{x}) \in [m, M]$ on all three vertices of the triangle K, then the cell average at next time step satisfies the maximum principle, i.e., $\overline{u}_K^{n+1} \in [m, M]$, under the CFL conditions:*

$$a\frac{\Delta t}{|K|}\sum_{i=1}^{3} l_K^i \le \frac{1}{6}, \quad \Delta t\frac{\alpha_{\mathbf{A}}}{|K|} \le \frac{\sqrt{3}-1}{72}, \quad \alpha_{\mathbf{A}} \ge (3+\sqrt{3})\frac{1}{\min_{K,i,j}\left(\sin(\theta_K^{i,j})\right)}, \tag{2.3.24}$$

where $\theta_K^{i,j}$ *is defined as in previous subsection, and* $\alpha_{\mathbf{A}} = \lambda_{\mathbf{A}}\alpha, \quad \lambda_{\mathbf{A}} = \max_{u\in[m,M],x\in T_h} \| \mathbf{A}(u,x) \|$.

Remark 2.3.2. *The CFL conditions (2.3.24) is only a sufficient condition rather than a necessary condition for maximum principle to hold. Larger time steps are possible by trying different expansions in (2.3.15).*

Given linear polynomials $p_K(\mathbf{x})$ as the DG polynomial defined on triangle K with $\overline{u}_K^n$ as its cell average over K at time level n. Then if $\overline{u}_K^n \in [m, M]$, the following limiter ensures $\tilde{p}_K(\mathbf{x}) \in [m, M]$ for any $\mathbf{x} \in K$:

$$\tilde{p}_K(\mathbf{x}) = \theta(p_K(\mathbf{x}) - \overline{u}_K^n) + \overline{u}_K^n, \quad \theta = \min\left\{1, \left|\frac{M - \overline{u}_K^n}{M_K - \overline{u}_K^n}\right|, \left|\frac{m - \overline{u}_K^n}{m_K - \overline{u}_K^n}\right|\right\} \quad (2.3.25)$$

where M_k and m_k are the maximum and minimum of $p_K(\mathbf{x})$ on the triangle K. Notice that the limiter (2.3.25) does not change the cell average. Moreover, we have $||\tilde{p}_K - p_K||_\infty = \mathcal{O}(h^2)$ if the exact solution is smooth, see [75] for the proof.

An efficient implementation of our maximum-principle-satisfying P^1-DG method (2.3.20) is:

1. Find the desired bound $[m, M]$ by evaluating the minimum and maximum of the initial condition. Given the triangular mesh, find the mesh constant $\min_{K,i,j}\left(\sin(\theta_K^{i,j})\right)$ and a direction $\vec{\nu}_0$ which is not parallel to any edge in the mesh. Calculate $\lambda_{\mathbf{A}} = \max_{u\in[m,M],x\in T_h} ||\mathbf{A}(u,x)||$. Take the penalty constant as $\alpha = (3+\sqrt{3})\frac{1}{\lambda_{\mathbf{A}}}\frac{1}{\min_{K,i,j}\left(\sin(\theta_K^{i,j})\right)}$ as suggested by (2.3.24).

2. Choose a discretization of the initial data such that the cell averages of the numerical initial data are in $[m, M]$. For instance, the standard L^2-projection.

3. Use a second order SSP time discretizations, for instance, the second order Runge-Kutta method (2.2.17).

4. For each forward Euler time stepping in (2.2.17), use the limiter (2.3.25) to pre-process the DG polynomials.

5. Consider the DG scheme with the limiter as an operator. Given the numerical solution u^n at time level n as an input, let $\widetilde{u}^{n+1}$ denote the output with a normal time step,

   - if the cell averages of $\widetilde{u}^{n+1}$ violates the maximum principle, then go back to time level n and evolve u^n with a smaller time step suggested by (2.3.24).
   - otherwise use $\widetilde{u}^{n+1}$ as the solution at the next time level, i.e., set $u^{n+1} = \widetilde{u}^{n+1}$ and proceed.

This algorithm is guaranteed to produce numerical solutions within the range $[m, M]$ with uniform second order accuracy for smooth exact solutions.

Remark 2.3.3. *The methodology discussed above can also be extended to the LDG or IPDG spatial discretizations on 2D triangular meshes. In this case all desirable numerical properties such as strict maximum principle satisfying, conservation of mass and second order of accuracy are still preserved as long as the proper time step restrictions are respected. Detailed algebra and theoretical proofs are omitted to keep this section reasonably concise. From a practical point of view, one can simply follow the efficient implementation approach proposed above.*

2.4 One dimensional numerical tests

In this section, we give numerical tests for the Cheng-Shu DG formulation [10]. The same set of numerical experiments have also been carried out using the IPDG and LDG schemes. The results are similar, thus we will not list them here to save space.

2.4.1 Accuracy test

We first consider the linear equation

$$u_t + u_x = \varepsilon u_{xx} \tag{2.4.1}$$

with initial data

$$u_0(x) = \sin(x)$$

on $[0, 2\pi]$ and periodic boundary conditions. The exact solution is:

$$u(x,t) = \exp(-\varepsilon t)\sin(x-t)$$

We take $\varepsilon = 0.001$ and the final time is $t = 1$. The time step is taken as suggested in (2.2.11). From Table 2.4.1, we observe the expected second order accuracy for the maximum-principle-satisfying DG scheme.

Table 2.4.1: Accuracy test ($\alpha = 10$)

	DG without Limiter				DG with Limiter			
N	L^1 error	order	L^∞ error	order	L^1 error	order	L^∞ error	order
20	1.83E-02	–	1.70E-01	–	2.82E-02	–	2.38E-01	–
40	4.55E-03	2.01	4.33E-02	1.97	6.18E-03	2.19	7.72E-02	1.62
80	1.13E-03	2.00	1.09E-03	1.99	1.38E-03	2.16	2.09E-03	1.89
160	2.82E-04	2.01	2.73E-04	2.00	3.25E-04	2.09	5.43E-04	1.94
320	7.03E-05	2.01	6.82E-05	2.00	7.69E-05	2.08	1.35E-04	2.01
640	1.75E-05	2.01	1.70E-05	2.01	1.82E-05	2.08	2.95E-05	2.19

2.4.2 Porous medium equations

A typical example of degenerate parabolic equations is the porous medium equation. In one space dimension, the porous medium equation can be written as:

$$u_t = (u^m)_{xx}, \quad m > 1. \tag{2.4.2}$$

We test the proposed scheme using the Barenblatt solution:

$$B_m(x,t) = t^{-k}[(1 - \frac{k(m-1)}{2m}\frac{|x|^2}{t^{2k}})_+]^{\frac{1}{m-1}} \tag{2.4.3}$$

which is an exact solution to (2.4.2) with compact support. The initial condition is taken to be $B_m(x,1)$, and the numerical solution is computed to $t = 2$, with zero boundary conditions.

We compare the numerical solutions obtained from the maximum principle satisfying DG scheme to the analytical solutions, see Figure 2.4.1 (a). The maximum-principle-satisfying DG scheme resolves the discontinuities in the solutions quite well, and keeps the solution strictly within the initial bounds everywhere for all time. If the DG schemes are applied without the limiter, there are significant undershoots near the foot of the numerical solution. See Figure 2.4.1 (b) and more clearly in the

zoomed version Figure 2.4.1 (c).

2.4.3 The Buckley-Leverett equation

Now we consider the convection-diffusion Buckley-Leverett equation, which is a model often used in reservoir simulations and a standard numerical test, see [34].

$$u_t + f(u)_x = \varepsilon(\nu(u)u_x)_x, \tag{2.4.4}$$

The numerical solution is computed at $t = 0.2$ with $\varepsilon = 0.01$ and boundary condition is $u(0,t) = 1$. The $\nu(u)$ and the initial condition are given as:

$$\nu(u) = \begin{cases} 4u(1-u) & 0 \le u \le 1 \\ 0 & \text{otherwise} \end{cases}, \quad u(x,0) = \begin{cases} 1-3x & 0 \le x \le \frac{1}{3} \\ 0 & \frac{1}{3} \le x \le 1 \end{cases} \tag{2.4.5}$$

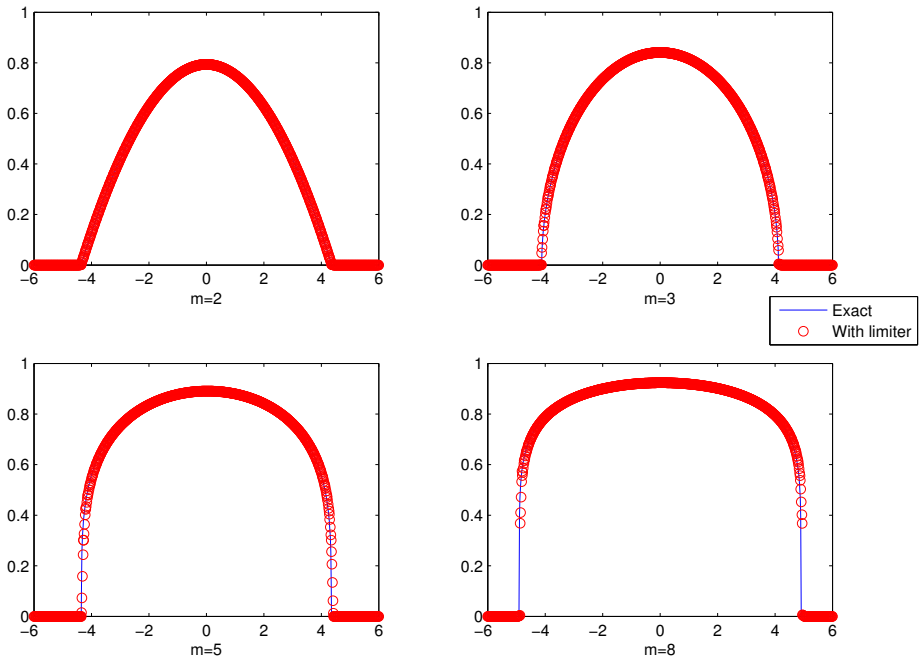
$f(u)$ has an s-shape:

$$f(u) = \frac{u^2}{u^2 + (1-u)^2}$$

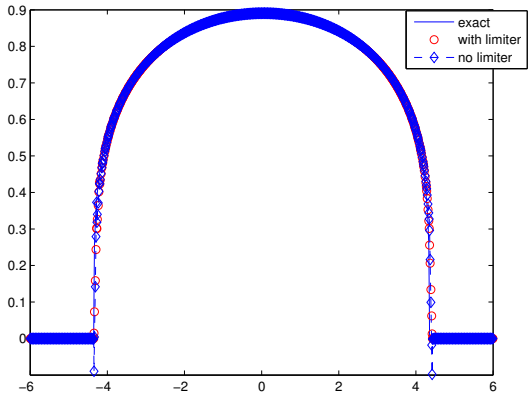
As shown in Figure 2.4.2, the maximum-principle-satisfying DG scheme provides good approximation and at the same time eliminates all the negative values.

2.5 Two-dimensional numerical tests

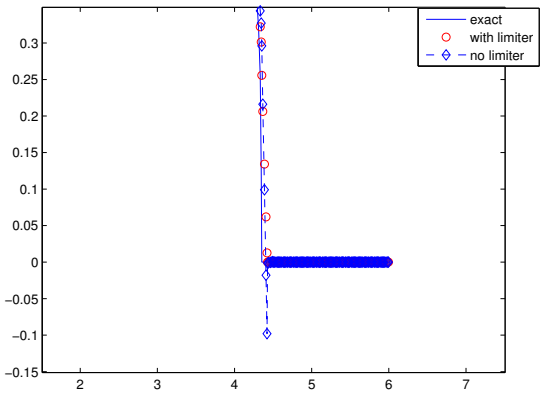
In this section, we list numerical results obtained from the Cheng-Shu DG formulation on triangular meshes. Again the numerical results for 2D LDG/IPDG formulations are similar to those obtained from the Cheng-Shu DG formulation,



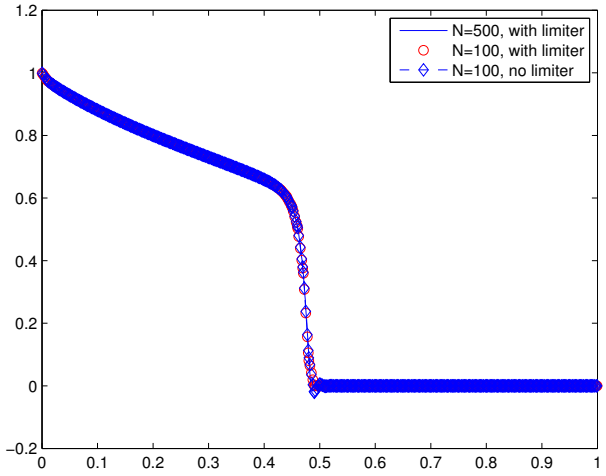
(a) Dashed lines: The numerical solutions computed by maximum-principle-satisfying DG schemes for $m = 2, 3, 5, 8$ on a uniform mesh with mesh size $h = 0.075$. Solid lines: The exact solutions.



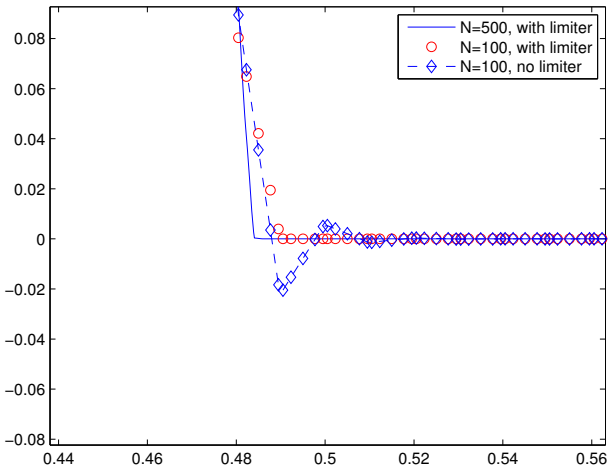
(b) Comparison of the numerical solutions ($m = 5$).



(c) Zoomed-in numerical solutions ($m = 5$).



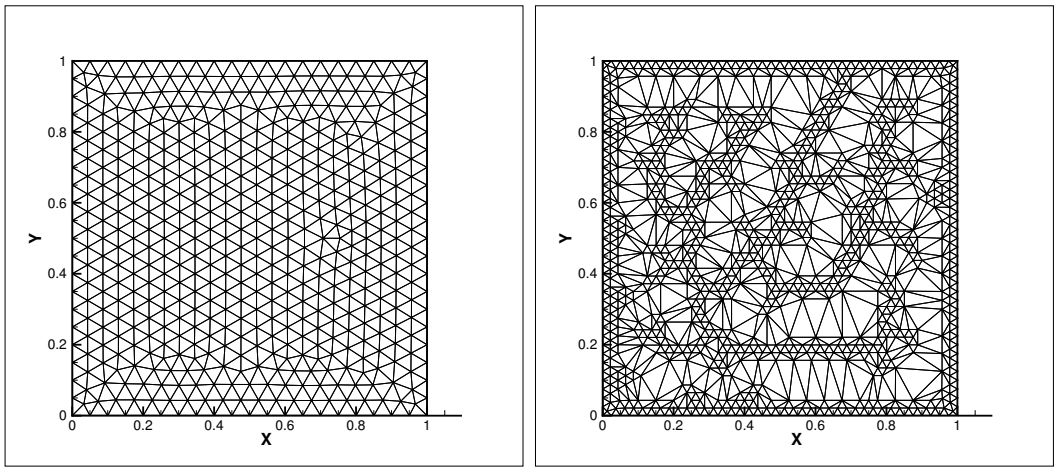
(a) Numerical solutions computed by DG with and without limiter.



(b) Zoomed in results near the corner of the numerical solutions.

Figure 2.4.2: The numerical results for one-dimensional Buckley-Leverett equation.

thus only one example will be shown. All tables and figures in this section refer to the results obtained from the Cheng-Shu DG formulation unless otherwise stated. We first give examples of the meshes that we use in the computation, see Figure 2.5.1. As shown in Section 2.3, our maximum-principle-satisfying DG schemes put no additional constraints on the shape of the elements. To be more specific, the schemes preserve the physical bounds strictly even if there are obtuse elements in the triangulation. All the meshes in this section are unstructured, generated by EashMesh [47]. Unless specified, the unstructured meshes used in the computation are refined independently, which means for each refinement we generate a new mesh with half of the previous mesh size.



(a) Triangular mesh. (b) Mesh containing obtuse elements.

Figure 2.5.1: An illustration of meshes used in Section 2.5.1.

2.5.1 Accuracy test

Similar to the one-dimensional case, we test the accuracy of the maximum-principle-satisfying DG scheme by solving:

$$u_t + \nabla \cdot u = \varepsilon \Delta u, \quad u_0(x) = \sin(2\pi(x+y)) \tag{2.5.1}$$

on $(x, y) \in [0, 1] \times [0, 1]$ with periodic boundary conditions. The exact solution is

$$u(x, y, t) = \exp(-8\pi^2 \varepsilon t) \sin(2\pi(x + y - 2t)).$$

We take $\varepsilon = 0.0001$ and the final time to be $t = 0.1$. The schemes are implemented as discussed in Section 2.3. Again, we observe the expected second order convergence rates for all three types of DG spatial discretizations (Cheng-Shu DG, LDG and IPDG) on meshes similar to those in Figure 2.5.1 (a), see Table 2.5.1 and Table 2.5.3. We also compute the numerical solutions on the obtuse mesh shown in Figure 2.5.1 (b), with the mesh size $h = 0.025$, and the largest angle among all the elements is about $\frac{2\pi}{3}$. For the DG scheme without limiter, we obtain the maximum and minimum of the numerical solutions to be 1.0958 and -1.0783 respectively; while by the DG scheme with limiter, the maximum and minimum of the numerical solutions are exactly 1.0000 and -1.0000, as expected by the initial conditions. If we refine the unstructured obtuse mesh like Figure 2.5.1 (b) in a structured way, namely every triangle is divided into four similar subtriangles for each refinement, we still observe uniform second order accuracy for the DG scheme with or without the limiter, see Table 2.5.2, as expected from the conclusions in Section 2.3.

Table 2.5.1: Two dimensional accuracy test ($\alpha = 10$).

	DG without Limiter				DG with Limiter			
h	L^1	rate	L^∞	rate	L^1	rate	L^∞	rate
0.1	1.71E-02	–	1.72E-01	–	2.59E-02	–	1.57E-01	–
0.05	4.27E-03	2.00	4.90E-02	1.81	6.00E-03	2.11	5.26E-02	1.58
0.025	1.05E-03	2.02	1.23E-02	2.00	1.33E-03	2.17	1.73E-02	1.61
0.0125	2.56E-04	2.04	3.61E-03	1.77	2.85E-04	2.23	3.90E-03	2.15
0.00625	6.12E-05	2.06	8.37E-04	2.11	6.19E-05	2.21	8.41E-04	2.21

Table 2.5.2: Two dimensional accuracy test on obtuse meshes.

	DG without Limiter				DG with Limiter			
h	L^1	rate	L^∞	rate	L^1	rate	L^∞	rate
0.025	4.21E-03	–	1.46E-01	–	7.29E-03	–	1.34E-01	–
0.0125	8.92E-04	2.24	4.58E-02	1.67	1.51E-03	2.27	4.57E-02	1.56
0.00625	1.92E-04	2.22	1.16E-02	1.98	2.85E-04	2.41	1.47E-02	1.64
0.003125	4.21E-05	2.19	2.76E-03	2.07	5.12E-05	2.48	3.73E-03	2.02

Table 2.5.3: Two dimensional accuracy test for LDG and IPDG.

	LDG accuracy results				IPDG accuracy results			
h	L^1	rate	L^1-limiter	rate	L^1	rate	L^1-limiter	rate
0.1	1.70E-02	–	2.59E-02	–	1.71E-02	–	2.60E-02	–
0.05	4.26E-03	2.00	5.98E-03	2.12	4.28E-03	2.00	6.00E-03	2.11
0.025	1.04E-03	2.03	1.32E-03	2.18	1.06E-03	2.02	1.34E-03	2.17
0.0125	2.52E-04	2.05	2.80E-04	2.24	2.58E-04	2.03	2.87E-04	2.22
0.00625	6.03E-05	2.06	6.31E-05	2.15	6.28E-05	2.04	6.35E-05	2.18

2.5.2 Porous medium equations

To test the proposed scheme with the two-dimensional porous medium equation:

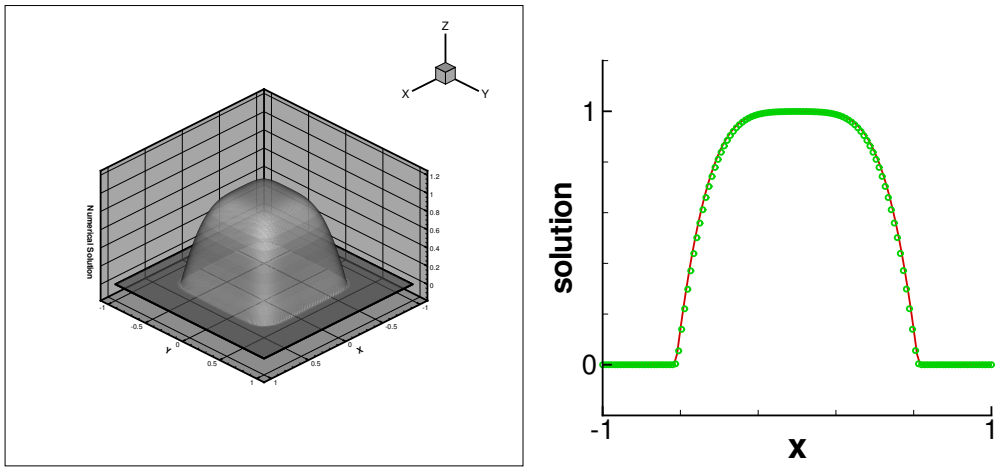
$$u_t = \Delta(u^2) \tag{2.5.2}$$

with a periodic boundary condition and the initial condition

$$u_0(x,y) = \begin{cases} 1, & \text{if} \quad (x,y) \in [-\frac{1}{2},\frac{1}{2}] \times [-\frac{1}{2},\frac{1}{2}] \\ 0, & \text{if otherwise on} \quad (x,y) \in [-1,1] \times [-1,1], \end{cases} \tag{2.5.3}$$

we compute our numerical solution up to time $t = 0.005$. For this degenerate parabolic equation, if we solve with the DG scheme only, the non-physical negative values will grow with time, while the maximum-principle-satisfying DG scheme will keep the numerical solution nonnegative, thus also ensuring the well-posedness of (2.5.2) and stability of the numerical scheme. In Figure 2.5.2 (b), we compare our numerical results with the non-conventional finite volume scheme proposed in [74].

We also compare our maximum-principle-satisfying DG scheme with the unlimited DG scheme, see Figure 2.5.3. Here we only compute our numerical solutions up to a relatively shorter time, since the negative values in the unlimited DG scheme will grow with time, and eventually lead to blowing up of the numerical solution. In contrast, our limited DG solution can be used for long time. The minimum of our numerical solutions on different meshes is reported in Table 2.5.4, turning out to be identically zero.



(a) Surface of the DG solution

(b) Cut along $y = 0$

Figure 2.5.2: The figure on the left and the dotted curve on the right are the numerical solutions at $T = 0.005$ of the maximum-principle-satisfying DG scheme on a $h = 0.0125$ mesh. The figure on the right is the cut of the numerical solution computed by maximum-principle-satisfying DG compared with the reference solution computed by the maximum-principle-satisfying finite volume WENO scheme [74] on a 128×128 mesh.

Mesh size	0.05	0.025	0.0125	0.00625
DG with limiter	0	0	0	0

Table 2.5.4: The minimum of the numerical solutions at $t = 0.005$.

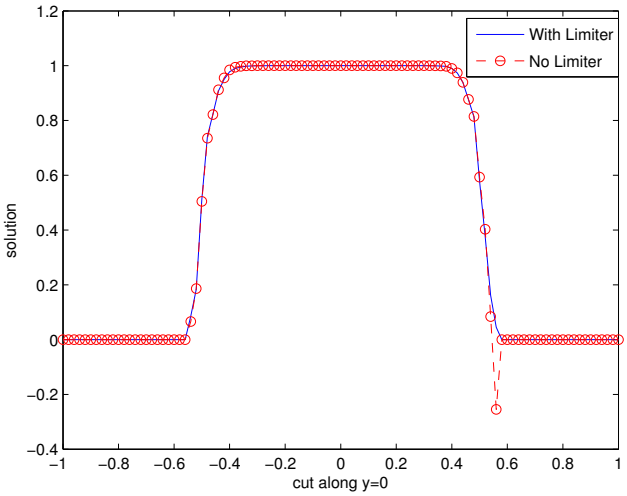


Figure 2.5.3: The numerical solutions computed by DG with and without limiter at $t = 0.0005$. The mesh size $h = 0.05$

2.6 Applications to incompressible Navier-Stokes equations

2.6.1 Preliminaries

To extend our maximum-principle-satisfying DG scheme to incompressible Navier-Stokes equations, we consider the following vorticity-stream function formulation:

$$w_t + (uw)_x + (vw)_y = \frac{1}{Re}\Delta w \tag{2.6.1}$$

$$\Delta\phi = w, \qquad \langle u, v\rangle = \langle -\phi_y, \phi_x\rangle \tag{2.6.2}$$

$$w(x, y, 0) = w_0(x, y), \qquad \langle u, v\rangle \cdot \vec{\nu} = given\ on\ \partial\Omega.$$

The exact solution of (2.6.1)-(2.6.2) satisfies the maximum principle, i.e.,

$$w(x,y,t) \in [m,M], \quad \forall (x,y) \in \Omega, \quad \forall t \geq 0$$

where $m = \min w_0(x,y)$, $M = \max w_0(x,y)$. In [39], Liu and Shu developed a high order DG method for solving (2.6.1). The flowchart of the scheme is:

- Solve (2.6.2) by a standard continuous finite element Poisson solver for the stream function ϕ.
- Take $\mathbf{u} = \langle u, v \rangle = \langle -\phi_y, \phi_x \rangle$, notice here $\langle u, v \rangle \cdot \vec{\nu}_i$ is continuous across the edges e_i since ϕ is continuous.
- Finally solve (2.6.1) by a DG scheme defined as follows: for a given triangulation T_h of the domain Ω, define the approximation spaces as:

$$V_h^k = \{v : v|_K \in P^k(K), \forall K \in T_h\}$$

For given $\mathbf{u} = \langle u, v \rangle$, which is obtained from the previous step, and test function $\varphi \in V_h^k$, the DG scheme is to find $w \in V_h^k$ such that

$$\begin{aligned} &\iint_K (\partial_t w)\varphi d\mathbf{x} - \iint_K w\mathbf{u} \cdot \nabla\varphi d\mathbf{x} + \sum_{e \in \partial K} \int_e \mathbf{u} \cdot \vec{\nu}\widehat{w}\varphi ds \\ &= \frac{1}{Re}\left(\sum_{e \in \partial K} \int_e (\widetilde{\nabla w \cdot \vec{\nu}})\varphi ds - \sum_{e \in \partial K} \int_e \widetilde{w}(\nabla\varphi \cdot \vec{\nu}) ds + \iint_K w\Delta\varphi d\mathbf{x} \right) \end{aligned} \tag{2.6.3}$$

Since $\mathbf{u} \cdot \vec{\nu}$ is continuous across the element edges, $\mathbf{u} \cdot \vec{\nu}\widehat{w}$ can be defined as the Lax-Friedrichs upwind biased flux, [79]:

$$\mathbf{u} \cdot \vec{\nu}\widehat{w} = \frac{1}{2}[\mathbf{u} \cdot \vec{\nu}(w^{in} + w^{out}) - a(w^{in} - w^{out})], \tag{2.6.4}$$

where a is taken to be the maximum of $|\mathbf{u}\cdot\vec{\nu}|$, either globally or locally. The diffusion fluxes $\widetilde{\nabla w\cdot\vec{\nu}}$, $\widetilde{w}$ are chosen as described in Section 2.3, (2.3.7):

$$\widetilde{\nabla w\cdot\vec{\nu}} = \nabla w^- + \frac{\alpha}{l_K^i}(w^{out} - w^{in}), \quad \widetilde{w} = w^+ \tag{2.6.5}$$

We use the same triangular mesh as in Section 2.3 , and both the Poisson solver and the DG method (2.6.3) use P^1 elements.

In [79], it is shown that for incompressible Euler equations on triangular meshes, the high order DG scheme, which is the same as in (2.6.3), omitting the diffusion terms, coupled with a linear scaling limiter, satisfies the maximum principle under the CFL condition

$$a\frac{\Delta t}{|K|}\sum_{i=1}^{3} l_K^i \le \frac{2}{3}\widehat{w}_1, \tag{2.6.6}$$

where $\widehat{w}_1$ is the one-dimensional Gauss-Lobatto quadrature weight for two cell ends. In particular, for the P^1 case, we have the following CFL condition for purely convection case:

$$a\frac{\Delta t}{|K|}\sum_{i=1}^{3} l_K^i \le \frac{1}{3} \tag{2.6.7}$$

Following the discussion in Section 2.3 and the result in [79], Theorem 2.3.1 also holds for (2.6.1).

2.6.2 Accuracy test

Firstly, we solve (2.6.1) with the initial condition: $w(x,y,0) = -2\sin(x)\sin(y)$ on $[0,2\pi]\times[0,2\pi]$ and periodic boundary conditions. The exact solution is available:

$$w(x,y,t) = -2\sin(x)\sin(y)\exp(-\frac{2t}{Re}).$$

In the computation, the Reynolds number is $Re = 100$ and the final time is $T = 0.1$. As shown in Table 2.6.1, the maximum-principle-satisfying DG scheme is second order accurate.

Table 2.6.1: Accuracy test.

	DG without Limiter				DG with Limiter			
h	L^1 error	order	L^∞ error	order	L^1 error	order	L^∞ error	order
0.8	3.36E-02	–	2.22E-01	–	3.85E-02	–	2.50E-01	–
0.4	8.71E-03	1.94	5.84E-02	1.93	8.73E-03	2.14	5.83E-02	2.10
0.2	2.18E-03	2.00	1.57E-02	1.90	2.18E-03	2.00	1.57E-02	1.90
0.1	5.78E-04	1.92	3.66E-03	2.10	5.78E-04	1.92	3.72E-03	2.07
0.05	1.50E-04	1.95	1.00E-03	1.87	1.50E-04	1.92	1.00E-03	1.90

2.6.3 The vortex patch problem

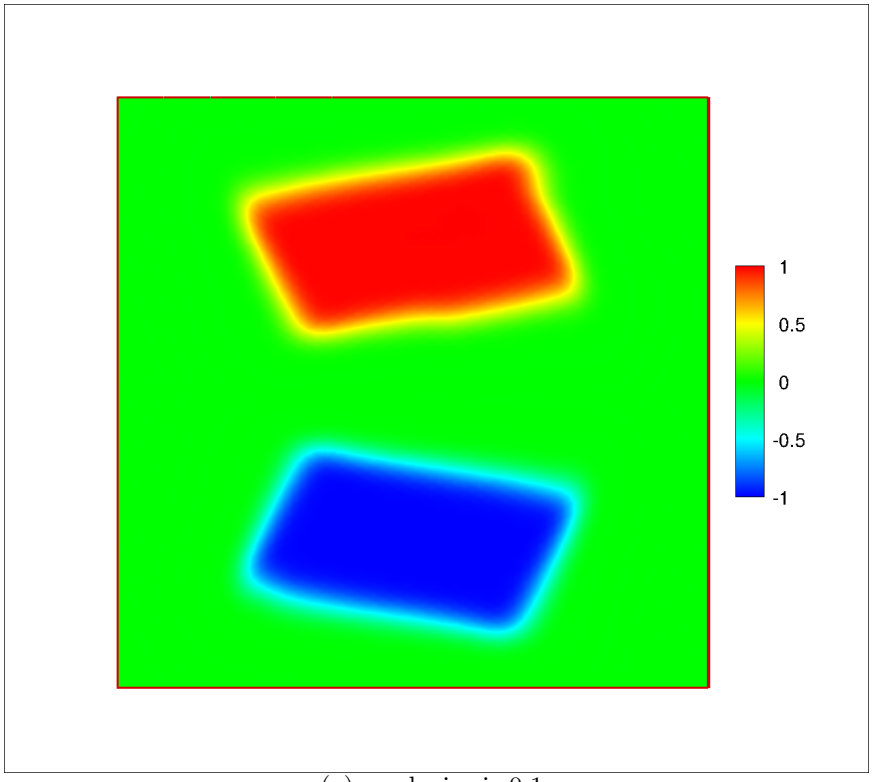
We test our maximum-principle-satisfying DG scheme by solving Navier-Stokes equations with the initial condition:

$$w_0(x,y,0) = \begin{cases} -1, & \text{if} \quad (x,y) \in [-\frac{\pi}{2}, \frac{3\pi}{2}] \times [-\frac{\pi}{4}, \frac{3\pi}{4}] \\ 1, & \text{if} \quad (x,y) \in [-\frac{\pi}{2}, \frac{3\pi}{2}] \times [-\frac{5\pi}{4}, \frac{7\pi}{4}] \\ 0, & \text{if otherwise on} \quad [0,2\pi] \times [0,2\pi] \end{cases} \tag{2.6.8}$$

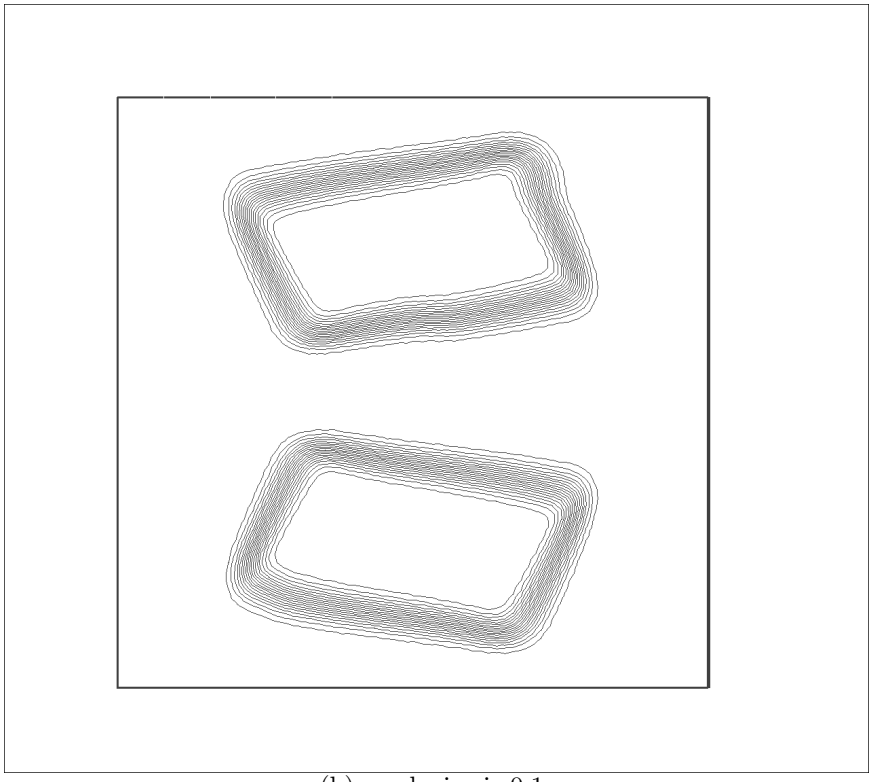
and periodic boundary conditions. The Reynolds number is specified to be $Re = 100$. We compare our numerical scheme with the unlimited DG scheme at $t = 0.1$ (see Table 2.6.2) and $t = 1$ (see Figure 2.6.1).

Table 2.6.2: Maximum and minimum of the numerical solutions at $T = 0.1$.

	DG without Limiter		DG with Limiter	
h	Min	Max	Min	Max
0.8	-1.002485131209995	1.008209167432207	-0.996833462986856	1.000000000000000
0.4	-1.012248342818721	1.029509746350882	-1.000000000000000	1.000000000000000
0.2	-1.063529424090108	1.048046525326357	-1.000000000000003	1.000000000000016
0.1	-1.028127448686370	1.037932424206786	-1.000000000000000	1.000000000000000



(a) mesh size is 0.1



(b) mesh size is 0.1

Figure 2.6.1: The contour of vorticity at time $t = 1$.

2.7 Concluding remarks

In this chapter, we present second order accurate maximum-principle-satisfying discontinuous Galerkin schemes for convection-diffusion equations. Through careful analysis and extensive numerical tests, we show that under suitable CFL conditions, with a simple scaling limiter involving little additional computational cost, the numerical schemes satisfy the strict maximum principle while maintaining uniform second order accuracy. The methodology works on both structured and unstructured triangular meshes, and also for two dimensional incompressible Navier-Stokes equations in the vorticity-streamfunction formulation. An important application is the positivity-preserving property for nonlinear degenerate parabolic equations, which may become ill-posed for negative values.

The effectiveness of the maximum-principle-satisfying DG schemes has been demonstrated through extensive numerical examples including two dimensional porous medium equations and the vortex patch problems. For maximum principle preserving, the proposed schemes require only slightly more restrictive time step restrictions than that required for linear stability. Generalizations to implicit time marching schemes constitute our ongoing work. It is also interesting to generalize the result to higher order discontinuous Galerkin schemes, although this appears difficult and suggests that there might be a need to modify the numerical fluxes used in the DG formulation to achieve this.

Chapter Three

Multi-scale Discontinuous Galerkin Schemes for Elliptic Equations with Curvilinear Unidirectional Rough Coefficients

3.1 Introduction

In chapter, we consider the solution of a special class of two dimensional second order elliptic boundary value problems with highly oscillatory coefficients

$$-\nabla \cdot (A(\mathbf{x})\nabla u) = f(\mathbf{x}) \quad \text{in } \Omega \tag{3.1.1}$$

and Dirichlet boundary condition

$$u = 0 \quad \text{on } \partial\Omega,$$

where Ω is a bounded computational domain, f is a function in $L^2(\Omega)$ and $A(\mathbf{x})$ is a special diagonal coefficient matrix containing small scales ϵ

$$A(\mathbf{x}) = \begin{pmatrix} a^\epsilon(x,y) & 0 \\ 0 & a^\epsilon(x,y) \end{pmatrix},$$

or equivalently,

$$-(a^\epsilon(x,y)u_x)_x - (a^\epsilon(x,y)u_y)_y = f(x,y). \tag{3.1.2}$$

As the typical situation in multi-scale modeling, we assume the elliptic coefficients $a^\epsilon(x,y)$ are highly oscillatory functions involving a small scale ϵ. Furthermore, $a^\epsilon(x,y)$ belongs to $L^\infty(\Omega)$ and satisfies

$$0 < \alpha \leq a^\epsilon(x,y) \leq \beta < \infty \tag{3.1.3}$$

for any $(x,y) \in \Omega$, where α and β are constants independent of ϵ. In particular, we are interested in a special subclass of rough coefficients, in which a^ϵ locally varies very rapidly in one direction, as defined originally by Babuška et al. in [29], and to

be specified more precisely in Subsection 3.2.1.

Applications of this type of problems arise in modeling two phase flow in porous media (see [35, 27, 8]), where Eq. (3.1.1) is the pressure equation, with u and $A(\mathbf{x})$ representing the pressure and the relative permeability tensor, respectively. In particular, $A(\mathbf{x})$ would be a diagonal tensor if the stochastic permeabilities are upscaled. Eq. (3.1.1) also captures the mechanics of steady state heat (electrical) conduction through a composite material (e.g. [27]), where $A(\mathbf{x})$ and u describe the thermal (electronic) conductivity and temperature (electronic potential).

It is a well known fact that if the coefficient $a^\epsilon(x, y)$ is rough, the solution to (3.1.2) will also be rough; to be more specific, we will in general have

$$||a^\epsilon||_{H^1(\Omega)} \to \infty, \qquad ||u||_{H^2(\Omega)} \to \infty, \quad \text{as } \epsilon \to 0.$$

Thus, u is not bounded uniformly with respect to ϵ in $H^2(\Omega)$ or in $H^{1+\delta}(\Omega)$ for any $\delta > 0$. As a result, numerical simulations often require highly refined computational meshes to ensure any reasonable accuracy. This intrinsic difficulty prohibits the application of traditional numerical methods for solving such problems due to the limitations of available computational resources, e.g. memory and CPU time.

In this chapter, we focus on the special class where the oscillation coefficient $a^\epsilon(x, y)$ is curvilinear unidirectional oscillatory. In [29], Babuška et al. proposed a special finite element method, which provides an efficient and convenient framework for constructing local trial spaces for this specific subclass of problem, i.e. Eq. 3.1.2. The key ingredients of the special finite element methods in [3, 17, 29] can be summarized as:

- A regularity result for the rough solution in a transformed space with respect to the right hand side function, $f \in L^2(\Omega)$.
- Construction of trial spaces with good approximation properties. In particular, they utilized special shape functions for which the explicit construction of a local multi-scale approximation basis is available, thus making the algorithm efficient and easy to implement.
- Construction of reasonable test spaces to ensure stability.

By exploiting the special features of the oscillatory coefficients, the special finite element method is proved to yield uniform second order convergence with respect to ϵ, even if the coefficient a^ϵ is rough. Moreover, for this particular subclass of two dimensional rough elliptic equations, the coefficients and the solution u can have a continuum scale spectrum from $O(\epsilon)$ to $O(1)$.

It is appealing to extend the results in [3, 29] to discontinuous Galerkin (DG) methods which do not enforce continuity constraints across the element interfaces, thus providing an easier way to construct a multiscale basis in higher dimensions with high-order elements. Motivated by special finite element method in [3, 17, 29], and the multi-scale DG method for unidirectional two-dimensional oscillatory coefficients proposed by Wang et al. in [65], we extend the idea in [29] for curvilinear unidirectional two-dimensional elliptic coefficients to multi-scale IP-DG [65] methods in two space dimension. Furthermore, in this chapter, we propose new ways of constructing local oscillatory approximation spaces, ensuring that the features of local preferred oscillating directions are automatically incorporated. In particular, we prove optimal error estimates for the second order scheme in two space dimension. The local mapping may put some constraints on the computational geometry, although this turns out to be non-essential restrictions required only for the error analysis. An

easy implementation framework is also suggested for the construction of a local basis, assuming that the leading oscillatory direction does not change abruptly within each element.

We remark that one of the practical difficulties for multi-dimensions special finite element method is that the method is designed for curvilinear elements. It is ideal for the purpose of error estimates, but could cause complications on the implementation side. For the multi-scale IP-DG method, we could use regular triangulations. As the trade-off, we must analyze the error associated with these discontinuities across element interfaces to obtain optimal error estimates.

3.2 Multi-scale DG method: the methodology

In this section, we focus on solving the elliptic multi-scale problem (3.1.2) on a bounded domain Ω, with additional features to be specified in Subsection 3.2.1. Let $\mathcal{T}_h$ be a collection of *quasi-uniform rectangular* partitions of Ω and $\mathcal{E}_h$ be the collection of edges of the $\mathcal{T}_h$. Define the following inner products

$$(v, w)_{\mathcal{T}_h} = \sum_{K \in \mathcal{T}_h} \int_K v(x, y) \cdot w(x, y) dx\, dy,$$
$$\langle v, w \rangle_{\mathcal{E}_h} = \sum_{e \in \mathcal{E}_h} \int_e v(s) \cdot w(s)\, ds.$$

where K denotes a given element in the triangulation, and e represents the edges associated with it.

There are many different DG formulations for such elliptic equations. In this chapter, we focus on the IP-DG formulation for (3.1.2): for any test function $v_h \in V_h$,

the IP-DG scheme is to find $u_h \in V_h$, such that

$$B_h(u_h, v_h) = (f, v_h)_{\mathcal{T}_h} \quad \forall v \in V_h, \tag{3.2.1}$$

where the bilinear form $B_h(u_h, v_h)$ is defined as

$$B_h(u, v) := (A\nabla u, \nabla v)_{\mathcal{T}_h} - \langle \{\!\{A\nabla u\}\!\}, [\![v]\!] \rangle_{\mathcal{E}_h} - \langle \{\!\{A\nabla v\}\!\}, [\![u]\!] \rangle_{\mathcal{E}_h} + \frac{\eta}{h} \langle [\![u]\!], [\![v]\!] \rangle_{\mathcal{E}_h}$$

where η is a sufficiently large penalty coefficient to ensure stability. For the scalar valued function u, $\{\!\{u\}\!\}$, $[\![u]\!]$ represent the average and jump across the element interface, respectively. To be more precise, let e be an interior edge shared by elements K_1 and K_2, define the unit normal vectors $\boldsymbol{n}_1$ and $\boldsymbol{n}_2$ on e pointing exterior to K_1 and K_2, respectively. We set

$$\{\!\{u\}\!\} = \frac{1}{2}(u_1 + u_2), \quad [\![u]\!] = u_1\boldsymbol{n}_1 + u_2\boldsymbol{n}_2 \quad \text{on } e \in \mathcal{E}_h^o, \tag{3.2.2}$$

with $u_i := u|_{\partial K_i}$, and $\mathcal{E}_h^o$ is the set of interior edges e. In a similar fashion, for a vector-valued function $\boldsymbol{q}$, we define $\boldsymbol{q}_1$ and $\boldsymbol{q}_2$ analogously and set

$$\{\!\{\boldsymbol{q}\}\!\} = \frac{1}{2}(\boldsymbol{q}_1 + \boldsymbol{q}_2), \quad [\![\boldsymbol{q}]\!] = \boldsymbol{q}_1 \cdot \boldsymbol{n}_1 + \boldsymbol{q}_2 \cdot \boldsymbol{n}_2 \quad \text{on } e \in \mathcal{E}_h^o. \tag{3.2.3}$$

For $e \in \mathcal{E}_h^\partial$, the set of boundary edges, we set

$$[\![u]\!] = u\boldsymbol{n}, \quad \{\!\{\boldsymbol{q}\}\!\} = \boldsymbol{q} \quad \text{on } e \in \mathcal{E}_h^\partial, \tag{3.2.4}$$

where $\boldsymbol{n}$ is the outward unit normal. We do not require either of the quantities $\{\!\{u\}\!\}$ or $[\![\boldsymbol{q}]\!]$ on boundary edges, and leave them undefined.

3.2.1 Curvilinear unidirectional oscillating coefficients

As Babuska et al. suggested in [29], the strict definition of "locally varying sharply in at most one direction" is outlined as follows. For the two dimensional elliptic equation (3.1.2) on the domain Ω, assume the existence of the following local change of variables, such that

- There exists an open cover $\{\Omega_i, i = 1, \ldots, n\}$ of Ω, with $\cup_{i=1}^{n} \Omega_i = \Omega$, and $\cup_{i=1}^{n} (\bar{\Omega}_i \cap \partial\Omega) = \partial\Omega$.
- For each Ω_i, there exists a local mapping: (ξ_i, η_i): $\Omega_i \to \tilde{\Omega}_i$ such that,
  - The mapping is one-to-one and onto, with $|\frac{\partial(\xi_i,\eta_i)}{\partial(x,y)}| \geq \gamma_i > 0$ on Ω_i,
  - $\nabla\xi_i \cdot \nabla\eta_i = 0$, and $\tilde{\Omega}_i$ is rectangular,
  - The boundary of each open set Ω_i is constraint by the boundaries of the domain Ω:

$$\begin{aligned} \bar{\Omega}_i \cap \partial\Omega &= \emptyset, \quad \text{if all the edges of } \Omega_i \text{ are interior edges} \\ &= \text{boundary edges of } \Omega_i, \text{ and remaining are interior edges} \end{aligned}$$

- Under the mapping, $a^\epsilon(x, y) = \tilde{a}_i^\epsilon(\xi_i)$, $\forall (x, y) \in \Omega_i$.

Under such conditions, the elliptic problem (3.1.2) is said to have curvilinear unidirectional coefficients. In other words, the local oscillation direction changes smoothly within the computational domain Ω. Furthermore, in [29], the authors proved a generalized version of the Berstein Theorem for this local change of variables, which we quoted here:

Theorem 3.2.1. *([29]) Let u be the solution in H_0^1 of (3.1.2), where we assume $f \in L^2(\Omega)$ and a^ϵ is locally varying sharply in at most one direction. Let $O_i \subset \Omega_i$ be open and satisfying $O_i \subset\subset \Omega_i$ if $\bar{O}_i \cap \partial\Omega = \emptyset$, and $(\partial O_i \cap \partial\Omega_i) \subset \partial\Omega$ if $\bar{O}_i \cap \partial\Omega \neq \emptyset$. Let $\tilde{O}_i$ be the image of O_i under the mapping (ξ_i, η_i). Then there is a constant C which does not depend on ϵ, such that*

$$\int_{\tilde{O}_i} \{\tilde{a}_i^\epsilon \mid \frac{\partial}{\partial\xi_i}(\tilde{a}_i^\epsilon \frac{\partial\tilde{u}}{\partial\xi_i}) \mid^2 + \tilde{a}_i^\epsilon \mid \frac{\partial^2\tilde{u}}{\partial\xi_i\partial\eta_i} \mid^2 + \frac{1}{\tilde{a}_i^\epsilon} \mid \frac{\partial^2\tilde{u}}{\partial\eta_i^2} \mid^2\} d\xi_i d\eta_i \leq C||f||_{L^2(\Omega)} \tag{3.2.5a}$$

Remark 3.2.2. *As suggested in [29], the result can be generalized to coefficients which could locally be transformed into the form $\tilde{a}^\epsilon(\frac{\xi}{\epsilon}, \eta)$.*

Remark 3.2.3. *One may notice that due to the definition of curvilinear unidirectional coefficients and properties of the associated local change of coordinates specified above, some constraints may apply to the computational domain that would enable the multi-scale DG method to deal with. For example, if we consider coefficients of the specific form $a^\epsilon(x, y) = \frac{1}{4+cos(\frac{x^2+y^2}{\epsilon})}$, the convenient coordinates mapping would be the polar one:*

$$\xi = \sqrt{x^2 + y^2}, \quad \eta = \tan^{-1}(\frac{y}{x})$$

Thus a quarter torus domain such as

$$1 \leq x^2 + y^2 \leq 4, \quad 0 \leq \eta \leq \frac{\pi}{2}$$

would satisfy the requirement that the transformed domain $\tilde{\Omega}$ is rectangular in (ξ, η). However, the definition suggests that a square domain like $[0, 1] \times [0, 1]$ will be ruled out. We notice that this restriction is only required for the regularity result above, which simplifies the error analysis of the multi-scale DG scheme. In Section 3.4, extensive numerical results suggest that this requirement is not essential. We still observe an optimal convergence rate for the second order multi-scale DG scheme,

even if the constraints associated with the local mapping are not strictly satisfied.

3.2.2 Approximation spaces

In this part, we first propose a way to construct the approximation space motivated by [29]. For each element $K_j \in \mathcal{T}_h$, the local oscillating approximation space is obtained as follows:

- Find Ω_i such that $K_j \subset \Omega_i$. and the local mapping (ξ_i, η_i) associated with Ω_i,
- Define the second order local approximation space on K_j as:

$$V_j^1 = \left\{ v : v|_{K_j} \in \text{span} \left\{ 1, \int_{\xi_j^*}^{\xi_i} \frac{1}{\tilde{a}_i^\epsilon(s)} ds, \eta_i - \eta_j^* \right\} \right\}.$$

- The multi-scale DG approximation space on Ω is $S_h^1 = \bigcup_{j=1}^N V_j^1$.
- Likewise, a higher order approximation space can be defined by including more basis function in, e.g. $S_h^2 = \bigcup_{j=1}^N V_j^2$, where

$$\begin{aligned} V_j^2 \qquad = \left\{ v : v|_{K_j} \in \text{span} \left\{ 1, \int_{\xi_j^*}^{\xi_i} \frac{1}{\tilde{a}_i^\epsilon(s)} ds, \eta_i - \eta_j^*, \int_{\xi_j^*}^{\xi_i} \frac{s}{\tilde{a}_i^\epsilon(s)} ds, \right. \right. \\ \left. \left. (\eta_i - \eta_j^*) \int_{\xi_j^*}^{\xi_i} \frac{1}{\tilde{a}_i^\epsilon(s)} ds, (\eta_i - \eta_j^*)^2 \right\} \right\}. \end{aligned}$$

This construction is vital to the success of the multi-scale DG method, which can be expected to yield the same convergence rate as the usual polynomial based DG methods when the coefficient a is smooth, while achieves greatly improved accuracy on a coarse mesh when a is rough. An easy implementation of the basis construction is proposed in Subsection 3.4.3, to address the cases where the local mapping is not

obvious or directly available in an analytical form. Numerical experiments show that this practical implementation achieves optimal order of convergence with a second order approximation space.

3.3 Multi-scale DG method: the error estimates

Combining all the pieces together, we have completely defined the multi-scale DG method for solving two dimensional elliptic equations (3.1.2) with curvilinear unidirectional coefficients. In this section, we proceed to analyze the theoretical aspects of the multi-scale DG methods. The first result to prove is the interpolation error associated with the special approximation space S_h^1 defined in the previous section.

3.3.1 Error estimates of the interpolation error

We first define the following straightforward energy norm

$$|||v|||^2 := (A\nabla v, \nabla v)_{\mathcal{T}_h} + \frac{1}{h}\langle [\![v]\!], [\![v]\!]\rangle_{\mathcal{E}_h} + h\langle \{\!\{A\nabla v\}\!\}, \{\!\{A\nabla v\}\!\}\rangle_{\mathcal{E}_h}.$$

Meanwhile, given $K_j \in \mathcal{T}_h$, find the containing partition Ω_i, such that $K_j \subset \Omega_i$. Suppose the corresponding local mapping (ξ_i, η_i) satisfies all the conditions specified in Subsection 3.2.1. Denote the images of K_j, Ω_i with such mapping as $\tilde{K}_j$, $\tilde{\Omega}_i$ respectively. Then define a further mapping as follows:

$$\hat{\xi}_i := \int^{\xi_i} \frac{1}{\tilde{a}_i^{\epsilon}(s)}\, ds$$

$$\hat{\eta}_i := \eta_i.$$

which maps the element $\tilde{K}_j \in \tilde{\mathcal{T}}_h$ to $\hat{K}_j \in \hat{\mathcal{T}}_h$. Accordingly, for any function u defined on K_j, we could define functions $\tilde{u}$ on $\tilde{K}_j$ and $\hat{u}$ on $\hat{K}_j$ separately as

$$\tilde{u}(\xi_i, \eta_i) = \hat{u}(\hat{\xi}_i, \hat{\eta}_i) = u(x, y).$$

Before preceding to a proof of the interpolation error in the energy norm, we state and prove the following two lemmas, one certifying the equivalence of L^2-estimates after changing coordinates, the other assuring the trace inequality estimates, both regarding the transformed "hat" domain.

Lemma 3.3.1. *Given K_j and the local mapping $(\hat{\xi}_i, \hat{\eta}_i)$ defined above, then for any function $w \in H^1(K_j)$, the following inequality holds*

$$\|w\|^2_{L^2(K_j)} \leq C' \|\hat{w}\|^2_{L^2(\hat{K}_j)} \tag{3.3.1}$$

$$\int_{K_j} A\nabla w \cdot \nabla w dx\, dy \leq C \|\hat{\nabla}\hat{w}\|^2_{L^2(\hat{K}_j)} \tag{3.3.2}$$

where C' and C are constants depending only on $\alpha, \beta, \xi_i, \eta_i$.

Proof. (3.3.1) can be proved by straightforward calculus:

$$\|w\|^2_{L^2(K_j)} = \int_{\hat{K}_j} \hat{w}^2 \left| \frac{\partial(x, y)}{\partial(\hat{\xi}_i, \hat{\eta}_i)} \right| d\hat{\xi}_i\, d\hat{\eta}_i \leq C' \|\hat{w}\|^2_{L^2(\hat{K}_j)}$$

Similar for (3.3.2)

$$\begin{aligned}\int_{K_j} A\nabla w \cdot \nabla w dx\, dy =& \int_{\tilde{K}_j} \left[\tilde{a}^\epsilon_i (\frac{\partial}{\partial \xi_i}\tilde{w})^2 |\nabla \xi_i|^2 + \tilde{a}^\epsilon_i (\frac{\partial}{\partial \eta_i}\tilde{w})^2 |\nabla \eta_i|^2 \right] \frac{\partial(x, y)}{\partial(\xi_i, \eta_i)} d\xi_i\, d\eta_i \\ \leq& C \int_{\tilde{K}_j} \left[\tilde{a}^\epsilon_i ((\frac{\partial}{\partial \xi_i}\tilde{w})^2 + \tilde{a}^\epsilon_i ((\frac{\partial}{\partial \eta_i}\tilde{w})^2 \right] d\xi_i\, d\eta_i\end{aligned}$$

$$\begin{aligned}
&\leq C \int_{\hat{K}_j} \left[(\frac{\partial}{\partial \hat{\xi}_i} \hat{w})^2 + (\tilde{a}_i^{\epsilon} \frac{\partial}{\partial \hat{\eta}_i} \hat{w})^2 \right] d\hat{\xi}_i \, d\hat{\eta}_i \\
&\leq C \|\hat{\nabla} \hat{w}\|^2_{L^2(\hat{K}_j)}
\end{aligned}$$

□

Lemma 3.3.2. *For each element K_j and the associated mapping, given any function $w \in C^1(K_j)$, the following trace inequality holds:*

$$\|\hat{w}\|_{L^2(\partial \hat{K}_j)} \leq C_1 h_{\hat{K}_j}^{-\frac{1}{2}} \|\hat{w}\|_{L^2(\hat{K}_j)} + C_2 h_{\hat{K}_j}^{\frac{1}{2}} \|\hat{\nabla} \hat{w}\|_{L^2(\hat{K}_j)}.$$

where $h_{\hat{K}_j} = diam(\hat{K}_j)$. Note that both constants C_1 and C_2 do not depend on the small scale ϵ.

Proof. Denote M as the mapping defined on $\mathbf{x} \in \partial K_j$: $M(\mathbf{x}) = \hat{\mathbf{x}} \in \partial \hat{K}_j$. As a result,

$$\hat{w}(\hat{\mathbf{x}}) = (\hat{w} \circ M)(\mathbf{x}) = w(\mathbf{x})$$

Moreover, by variable substitution,

$$\int_{\partial \hat{K}_j} \hat{w}(\hat{\mathbf{x}}) d\hat{\mathbf{x}} = \int_{\partial K_j} (\hat{w} \circ M)(\mathbf{x}) |det(J_M)(\mathbf{x})| d\mathbf{x}$$

where J_M is the Jacobian of the mapping M. It is obvious that there exist constants independent of ϵ, $0 < m_1(\alpha, \beta, \xi_i, \eta_i) \leq m_2(\alpha, \beta, \xi_i, \eta_i)$, such that $m_1 \leq |det(J_M)| \leq m_2$. Therefore, by the trace inequality on K_j and Lemma 3.3.1,

$$\begin{aligned}
\|\hat{w}\|_{L^2(\partial \hat{K}_j)} &\leq C \|w\|_{L^2(\partial K_j)} \leq C_1 h^{-\frac{1}{2}} \|w\|_{L^2(K_j)} + C_2 h^{\frac{1}{2}} \|\nabla w\|_{L^2(K_j)} \\
&\leq C_1 h^{-\frac{1}{2}} \|\hat{w}\|_{L^2(\hat{K}_j)} + C_2 h^{\frac{1}{2}} \|\hat{\nabla} \hat{w}\|_{L^2(\hat{K}_j)} \\
&\leq C_1 h_{\hat{K}_j}^{-\frac{1}{2}} \|\hat{w}\|_{L^2(\hat{K}_j)} + C_2 h_{\hat{K}_j}^{\frac{1}{2}} \|\hat{\nabla} \hat{w}\|_{L^2(\hat{K}_j)}
\end{aligned}$$

We have used here that $M_1 h \leq h_{\hat{K}_j} \leq M_2 h$. □

We then proceed to prove the following approximation result.

Lemma 3.3.3. *Let u be the exact solution of (3.1.2). Then, there exists $v \in S_h^1$ so that*

$$|||u - v||| \leq Ch\|f\|_{L^2(\Omega)}.$$

Proof. Define $\hat{v}$ to be the piecewise linear function defined on $\hat{K}_j$ such that $\hat{v}|_{\hat{K}_j} \in P^1(\hat{K}_j)$ satisfies

$$h_{\hat{K}_j}\|\hat{\nabla}(\hat{u} - \hat{v})\|_{L^2(\hat{K}_j)} + \|\hat{u} - \hat{v}\|_{L^2(\hat{K}_j)} \leq Ch_{\hat{K}_j}^2 |\hat{D}^2\hat{u}|_{L^2(\hat{K}_j)}, \tag{3.3.3}$$

From the definition of $|||\cdot|||$ we have

$$\begin{aligned} |||u - v|||^2 =& (A\nabla(u - v), \nabla(u - v))_{\mathcal{T}_h} + \frac{1}{h}\langle [\![(u - v)]\!], [\![(u - v)]\!]\rangle_{\mathcal{E}_h} \\ &+ h\langle \{\!\{A\nabla(u - v)\}\!\}, \{\!\{A\nabla(u - v)\}\!\}\rangle_{\mathcal{E}_h}. \end{aligned} \tag{3.3.4}$$

for which each term will be bounded individually.

As for the first term in the energy norm (3.3.4), by Lemma 3.3.1 and the linear approximation results (3.3.3)

$$\int_{K_j} A\nabla(u - v) \cdot \nabla(u - v)dx\, dy \leq C\|\hat{\nabla}(\hat{u} - \hat{v})\|_{L^2(\hat{K}_j)}^2 \leq Ch_{\hat{K}_j}^2 |\hat{D}^2\hat{u}|_{L^2(\hat{K}_j)}^2.$$

Hence, if we sum over $K_j \in \Omega_i$ we recover that there exists some $O_i \subset \Omega_i$

$$\sum_{K_j \in \Omega_i} (A\nabla(u-v), \nabla(u-v))_{K_j} \leq (A\nabla(u-v), \nabla(u-v))_{O_i} \leq Ch^2 |\hat{D}^2\hat{u}|_{L^2(\hat{O}_i)}^2. \tag{3.3.5}$$

$$(A\nabla(u-v),\nabla(u-v))_{\mathcal{T}_h} \leq \sum_i Ch^2|\hat{D}^2\hat{u}|^2_{L^2(\hat{O}_i)}. \tag{3.3.6}$$

For the second term in the energy norm (3.3.4), we use a trace inequality to get

$$\frac{1}{h}\langle[\![(u-v)]\!],[\![(u-v)]\!]\rangle_{\mathcal{E}_h} \leq \frac{C}{h^2}\|u-v\|^2_{L^2(\Omega)} + C\|\nabla(u-v)\|^2_{L^2(\Omega)} \tag{3.3.7}$$

By Lemma 3.3.1 , the first term in (3.3.7) satisfies

$$\|u-v\|^2_{L^2(K_j)} \leq C\|\hat{u}-\hat{v}\|^2_{L^2(\hat{K}_j)} \leq Ch^4_{\hat{K}_j}|\hat{D}^2\hat{u}|^2_{L^2(\hat{K}_j)},$$

Taking the sum over $K_j \in \Omega_i$ we get

$$\sum \|u-v\|^2_{L^2(K_j)} \leq Ch^4|\hat{D}^2\hat{u}|^2_{L^2(\hat{O}_i)}, \quad \frac{1}{h^2}\|u-v\|^2_{L^2(\Omega)} \leq \sum_i Ch^2|\hat{D}^2\hat{u}|^2_{L^2(\hat{O}_i)} \tag{3.3.8}$$

As for the second term in (3.3.7), it can be bounded by scaling argument,

$$\|\nabla(u-v)\|^2_{L^2(\Omega)} \leq C\frac{1}{\alpha}(A\nabla(u-v),\nabla(u-v))_{\mathcal{T}_h} \leq \sum_i Ch^2|\hat{D}^2\hat{u}|^2_{L^2(\hat{O}_i)}, \tag{3.3.9}$$

where in the last inequality we use (3.3.6). Therefore for the middle term in the energy norm (3.3.4),

$$\frac{1}{h}\langle[\![(u-v)]\!],[\![(u-v)]\!]\rangle_{\mathcal{E}_h} \leq \sum_i Ch^2|\hat{D}^2\hat{u}|^2_{L^2(\hat{O}_i)} \tag{3.3.10}$$

For the last term in the energy norm (3.3.4), following the mapping argument carried out in Lemma 3.3.1 and Lemma 3.3.2, we have

$$h\langle\{\!\{A\nabla(u-v)\}\!\},\{\!\{A\nabla(u-v)\}\!\}\rangle_{\mathcal{E}_h} \leq Ch\sum\|\hat{\nabla}(\hat{u}-\hat{v})\|^2_{L^2(\partial\hat{K}_j)} \tag{3.3.11}$$

By the trace inequality in Lemma 3.3.2, it is clear that

$$\begin{aligned} h\sum \|\hat{\nabla}(\hat{u}-\hat{v})\|^2_{L^2(\partial\hat{K}_j)} \leq & C\sum_{\hat{K}_j}\|\hat{\nabla}(\hat{u}-\hat{v})\|^2_{L^2(\hat{K}_j)} + C\,h^2\sum_{\hat{K}_j}\|\hat{D}^2\hat{u}\|^2_{L^2(\hat{K}_j)} \\ \leq & C\,h^2\sum_{\hat{K}_j}\|\hat{D}^2\hat{u}\|^2_{L^2(\hat{K}_j)}. \end{aligned}$$

Thus we bound the third term in the (3.3.4) as:

$$h\langle\{\!\{A\nabla(u-v)\}\!\},\{\!\{A\nabla(u-v)\}\!\}\rangle_{\mathcal{E}_h} \leq C\,h^2\sum_i \|\hat{D}^2\hat{u}\|^2_{L^2(\hat{O}_i)} \tag{3.3.12}$$

Combine (3.3.6), (3.3.10), (3.3.12) to obtain

$$|||u-v|||^2 \leq \sum_i Ch^2|\hat{D}^2\hat{u}|^2_{L^2(\hat{O}_i)}.$$

To complete the proof we argue that

$$\sum_i |\hat{D}^2\hat{u}|_{L^2(\hat{O}_i)} \leq C\|f\|_{L^2(\Omega)}. \tag{3.3.13}$$

As indicated by Theorem 3.2.1, (3.3.13) is immediate and this completes the proof.

□

3.3.2 Coercivity result

Next, we are going to prove the coercivity of the bilinear form, again by referring to Lemma 3.3.2.

Lemma 3.3.4. *The following coercivity result holds*

$$|||v|||^2 \leq CB_h(v,v) \quad \forall v \in S_h^1. \tag{3.3.14}$$

Proof. Let $v \in S_h^1$. Then by the definition of $B_h(\cdot,\cdot)$ we get

$$B_h(v,v) = (A\nabla v, \nabla v)_{\mathcal{T}_h} - 2\langle \{\!\{A\nabla v\}\!\}, [\![v]\!]\rangle_{\mathcal{E}_h} + \frac{\eta}{h}\langle [\![v]\!], [\![v]\!]\rangle_{\mathcal{E}_h}. \tag{3.3.15}$$

By the arithmetic-geometric mean inequality we get that

$$2\langle \{\!\{A\nabla v\}\!\}, [\![v]\!]\rangle_{\mathcal{E}_h} \leq \frac{1}{\delta h}\langle [\![v]\!], [\![v]\!]\rangle_{\mathcal{E}_h} + h\delta\langle \{\!\{A\nabla v\}\!\}, \{\!\{A\nabla v\}\!\}\rangle_{\mathcal{E}_h}, \tag{3.3.16}$$

for any $\delta > 0$. Next, we bound $h\delta\langle \{\!\{A\nabla v\}\!\}, \{\!\{A\nabla v\}\!\}\rangle_{\mathcal{E}_h}$. By (3.3.11), it is immediate that

$$\langle \{\!\{A\nabla v\}\!\}, \{\!\{A\nabla v\}\!\}\rangle_{\mathcal{E}_h} \leq C \sum \|\hat{\nabla}\hat{v}\|^2_{L^2(\partial \hat{K}_j)} \tag{3.3.17}$$

Invoking Lemma 3.3.2, the inverse inequality for the transformed domain,

$$\langle \{\!\{A\nabla v\}\!\}, \{\!\{A\nabla v\}\!\}\rangle_{\mathcal{E}_h} \leq C \sum \|\hat{\nabla}\hat{v}\|^2_{L^2(\partial \hat{K}_j)} \leq Ch^{-1} \sum_{\hat{K}_j} \|\hat{\nabla}\hat{v}\|^2_{L^2(\hat{K}_j)} \tag{3.3.18}$$

In the last inequality, the higher order derivative term drops out since $\hat{v} \in P^1(\hat{K}_j)$.

Combining this with Lemma 3.3.1,

$$h\delta\langle \{\!\{A\nabla v\}\!\}, \{\!\{A\nabla v\}\!\}\rangle_{\mathcal{E}_h} \leq C\delta \sum_{\hat{K}_j} \|\hat{\nabla}\hat{v}\|^2_{L^2(\hat{K}_j)} \leq C\delta(A\nabla v, \nabla v)_{\mathcal{T}_h}. \tag{3.3.19}$$

If we insert this result into (3.3.16) we get

$$2\langle \{\!\{A\nabla v\}\!\}, [\![v]\!]\rangle_{\mathcal{E}_h} \leq \frac{1}{\delta h}\langle [\![v]\!], [\![v]\!]\rangle_{\mathcal{E}_h} + C\delta(A\nabla v, \nabla v)_{\mathcal{T}_h}. \tag{3.3.20}$$

Using (3.3.15) to follow that

$$\begin{aligned} B_h(v,v) \geq & (A\nabla v, \nabla v)_{\mathcal{T}_h} + \frac{\eta}{h}\langle [\![v]\!], [\![v]\!]\rangle_{\mathcal{E}_h} - \frac{1}{\delta h}\langle [\![v]\!], [\![v]\!]\rangle_{\mathcal{E}_h} \\ & - C\delta (A\nabla v, \nabla v)_{\mathcal{T}_h} \\ = & (1 - C\delta)(A\nabla v, \nabla v)_{\mathcal{T}_h} + (\frac{\eta}{h} - \frac{1}{\delta h})\langle [\![v]\!], [\![v]\!]\rangle_{\mathcal{E}_h}. \end{aligned}$$

If we choose δ such that $(1 - C\delta) \leq \frac{1}{2}$ and assume that η is sufficiently large so that $\eta \geq (\frac{1}{2} + \frac{1}{\delta})$, we recover

$$(A\nabla v, \nabla v)_{\mathcal{T}_h} + \frac{1}{h}\langle [\![v]\!], [\![v]\!]\rangle_{\mathcal{E}_h} \leq C\, B(v,v).$$

The proof is complete by using (3.3.19). □

3.3.3 Main result

Combining the interpolation error analysis and the coercivity result, we now state our main error estimate as follows:

Theorem 3.3.1. *Let u be the solution of* (3.1.2) *and let $u_h \in V_h$ be the IP-DG approximation, then*

$$|||u - u_h||| \leq C\, h \|f\|_{L^2(\Omega)}, \tag{3.3.21a}$$

$$\|u - u_h\|_{L^2(\Omega)} \leq C\, h^2 \|f\|_{L^2(\Omega)}. \tag{3.3.21b}$$

Proof. We first prove (3.3.21a). To this end, by Lemma 3.3.4 we have, for any $v \in S_h^1$,

$$|||v - u_h|||^2 \leq C\, B_h(v - u_h, v - u_h),$$

By Galerkin orthogonality of the IP-DG method we have

$$|||v - u_h|||^2 \leq C\, B_h(v - u, v - u_h).$$

Clearly, $B_h(\cdot, \cdot)$ is a bounded bilinear form. That is,

$$B_h(v - u, v - u_h) \leq C \; ||| \, v - u \, ||| \; |||v - u_h \, ||| \, .$$

Therefore,

$$|||v - u_h||| \leq C \; ||| \, v - u \, ||| \, .$$

The triangle inequality gives

$$|||u - u_h||| \leq C \; ||| \, v - u \, ||| \, .$$

Since this holds for any $v \in S_h^1$, Lemma 3.3.3 yields (3.3.21a).

To prove (3.3.21b), we will use a duality argument. Define the problem

$$-(a(x,y)\phi_x(x,y))_x - (a(x,y)\phi_y(x,y))_y = (u - u_h)(x,y), \quad (x,y) \in \Omega, \tag{3.3.22a}$$

$$\phi(x,y) = 0, \quad (x,y) \in \partial\Omega. \tag{3.3.22b}$$

By adjoint consistency of the IP-DG method we have

$$\|u - u_h\|^2_{L^2(\Omega)} = B_h(u - u_h, \phi) = B_h(u - u_h, \phi - v),$$

for any $v \in S_h^1$. Here we have used Galerkin orthogonality. Hence,

$$\|u - u_h\|^2_{L^2(\Omega)} \leq C \; ||| \, u - u_h \, ||| \; |||\phi - v||| \leq C\, h \; ||| \, u - u_h \, ||| \; \|u - u_h\|_{L^2(\Omega)},$$

where we have used Lemma 3.3.3. Inequality (3.3.21b) now follows from (3.3.21a). □

3.4 Multi-scale DG method: the numerical results

In this section, various two-dimensional numerical examples are presented to demonstrate that the proposed multi-scale DG method is able to capture oscillatory solutions without having to completely resolve the finest scales therein. This is in contrast to the traditional polynomial a DG method based on approximation spaces.

We consider two two-dimensional elliptic multiscale examples on the domain $[-1,1]^2$. In the first example we do not enforce Dirichlet boundary condition, thus an exact solution can be obtained. The second example is a real two-dimensional Dirichlet problem, in which case explicit formulas for the exact solutions are not available. Thus we compute the reference solutions by a spectral Chebyshev collocation method with a 512×512 mesh to check the convergence rates of the DG methods.

We note that in the second example, due to the difficulty of computing a high resolution reference solution, we are unable to test for very small values of ϵ (we tested $\epsilon = 0.01$ and $\epsilon = 0.005$). Indeed, the smaller ϵ is, the more significant advantage of the multiscale IP-DG method is expect to be. Since the error of the multiscale IP-DG method does not depend on ϵ, whereas the traditional IP-DG method only has convergence when the mesh size is comparable to ϵ.

Finally, an easy implementation is proposed. Though no theoretical proof for the error estimates has been obtained, we demonstrate the effectiveness of this new way of implementation through the same set of numerical tests.

3.4.1 Two-dimensional example I: Accuracy test

In the first example, we consider $a^\epsilon(x,y) = \frac{1}{4+\cos(\frac{x^2+y^2}{\epsilon})}$. By not imposing the Dirichlet boundary condition, we can assume an exact solution and smooth right hand side function, i.e.,

$$u = (x^2+y^2)^2 + \frac{1}{2}\epsilon(x^2+y^2)\sin(\frac{x^2+y^2}{\epsilon}) + \frac{1}{2}\epsilon^2\cos(\frac{x^2+y^2}{\epsilon}), \quad f = -4(x^2+y^2). \tag{3.4.1}$$

Note that throughout the domain, the oscillations only appear in the radical direction. Thus we define the corresponding mapping $(\xi,\eta) : \xi(x,y) = \sqrt{x^2+y^2}$, $\eta(x,y) = \tan^{-1}(\frac{y}{x})$. The second order local approximations space S_h^1 is just

$$\text{span}\left\{1, \int_{\xi^*_{K_j}}^{\xi} \frac{1}{\tilde{a}^\epsilon(s)}ds, \eta - \eta^*_{K_j}\right\},$$

where $\tilde{a}(\xi) = \frac{1}{4+\cos(\frac{\xi^2}{\epsilon})}$. The third order local approximation space S_h^2 is

$$\text{span}\left\{1, \int_{\xi_j^*}^{\xi_i} \frac{1}{\tilde{a}^\epsilon(s)}ds, \eta_i - \eta_j^*, \int_{\xi_j^*}^{\xi_i} \frac{s}{\tilde{a}^\epsilon(s)}ds, (\eta_i-\eta_j^*)\int_{\xi_j^*}^{\xi_i} \frac{1}{\tilde{a}^\epsilon(s)}ds, (\eta_i-\eta_j^*)^2\right\}.$$

The integration in the construction of basis above is done numerically with a sufficient number of quadrature points to completely resolve $\epsilon = 0.01$. In Table 3.4.1, the L^2-errors and convergence rates of the multi-scale DG methods are listed, showing an almost second order convergence rate and a clear third order of convergence rate for the S_h^1 and S_h^2 spaces, respectively. The results for smaller ϵ, e.g., $\epsilon = 0.005$ are

similar, thus we do not list the results here.

Table 3.4.1: L^2-errors and orders of accuracy by the multi-scale DG method: two-dimensional accuracy test when $\epsilon = 0.01$.

	S_h^1		S_h^2	
N	error	order	error	order
10	5.59E-01	–	3.50E-03	–
20	1.77E-01	1.66	3.21E-04	3.45
40	4.84E-02	1.87	3.35E-05	3.26
80	1.22E-02	1.99	3.93E-06	3.09

3.4.2 Two-dimensional example II: Dirichlet problem

In this example, the multi-scale DG method is tested with the coefficients with no separation of scales:

$$a^\epsilon(x,y) = \frac{1}{4+\sin(\frac{\cos(x^2+y^2)}{\epsilon})}, \quad f = -2. \tag{3.4.2}$$

The same local mapping is used as in the first example, i.e. the polar one (ξ, η). In this case, an explicit formula for the exact solution is no longer available, thus the numerical solutions are compared with reference solutions as mentioned above.

Table 3.4.2 shows the L^2-errors and orders of convergence for the cases with $\epsilon = 0.01$ and $\epsilon = 0.005$. As expected, it shows a nearly second order of convergence rates for the S_h^1 space. Although an clear improvement in the errors could be observed, for the S_h^2 space, it is not clear what the convergence rate is, since the error decreases faster after only a few refinements, the reference solution is no longer reliable.

Table 3.4.2: L^2-errors and orders of accuracy by the multi-scale DG method: two-dimensional Dirichlet boundary condition.

	$\varepsilon = 0.01$		$\varepsilon = 0.005$	
S_h^1				
N	error	order	error	order
10	3.53E-01	–	3.58E-01	–
20	1.25E-01	1.50	1.25E-01	1.52
40	3.72E-02	1.75	3.65E-02	1.78
80	1.03E-02	1.85	9.91E-03	1.88
160	2.68E-03	1.95	2.58E-03	1.94
S_h^2				
10	8.50E-03	–	7.06E-03	–
20	1.53E-03	2.48	6.94E-04	3.55
40	9.08E-04	0.75	1.98E-04	1.81
80	2.89E-04	1.65	1.88E-04	0.08

3.4.3 Two-dimensional example III: Easy implementations

In this subsection, an easy implementation is proposed for the multi-scale DG methods. From the assumption on a^ϵ, it is clear that within each local element, the oscillation direction of the coefficients changes smoothly. Intuitively the oscillation direction at each cell center should be able to capture and explain the majority part of the local oscillations. This approach would be especially helpful for practical purposes, when the local mapping (ξ, η), is not directly available, or can not be written out in an analytical form. A new approximation space could be defined as:

$$ES_h^1 = \bigcup_j V_j^h = \left\{ v : v|_{K_j} \in \text{span} \left\{ 1, \int_{\xi^*_{K_j}}^{\xi_j} \frac{1}{\tilde{a}_j^\epsilon(s, \eta^*_{K_j})} ds, \eta_j - \eta^*_{K_j} \right\} \right\},$$

$$ES_h^2 = \bigcup_j V_j^h = \left\{ v : v|_{K_j} \in \text{span} \left\{ 1, \int_{\xi^*_{K_j}}^{\xi_j} \frac{1}{\tilde{a}_j^\epsilon(s, \eta^*_{K_j})} ds, \eta_j - \eta^*_{K_j}, \right.\right.$$
$$\left.\left. (\eta_j - \eta^*_{K_j}) \int_{\xi^*_{K_j}}^{\xi_j} \frac{1}{\tilde{a}_j^\epsilon(s, \eta^*_{K_j})} ds, \int_{\xi^*_{K_j}}^{\xi_j} \frac{s}{\tilde{a}_j^\epsilon(s, \eta^*_{K_j})} ds, (\eta_j - \eta^*_{K_j})^2 \right\} \right\}.$$

Where ξ_j denotes the oscillation direction of $a^\epsilon(x,y)$ at cell center of K_j, and η_j is the direction perpendicular to ξ_j. So (ξ_j, η_j) is now just a rotation of the original coordinates (x,y). $(\xi^*_{K_j}, \eta^*_{K_j})$ denotes the element center of K_j under the new mapping (ξ_j, η_j).

Consider the two dimensional accuracy test case for example, for each cell K_j with cell center's coordinates denoted as $(x^*_{K_j}, y^*_{K_j})$. We know that the oscillation direction at the cell center is $\xi_j = \cos(\theta^*_{K_j})x + \sin(\theta^*_{K_j})y$, and in turn the orthogonal direction to ξ_j is $\eta_j = -\sin(\theta^*_{K_j})x + \cos(\theta^*_{K_j})y$, where $\theta^*_{K_j} = \tan^{-1}(\frac{y^*_{K_j}}{x^*_{K_j}})$. Under the mapping $\xi_j(x,y), \eta_j(x,y)$ defined as above, the element center (x_j, y_j) can now be represented as $\xi^*_{K_j} = \cos(\theta^*_{K_j})x^*_{K_j} + \sin(\theta^*_{K_j})y^*_{K_j}$, $\eta^*_{K_j} = -\sin(\theta^*_{K_j})x^*_{K_j} + \cos(\theta^*_{K_j})y^*_{K_j}$. As a result, the new local oscillatory basis defined on K_j now is:

$$\int_{\xi^*_{K_j}}^{\xi_j} \frac{1}{\tilde{a}_j(s, \eta^*_{K_j})} ds = \int_{\xi^*_{K_j}}^{\xi_j} \left(4 + \cos(\frac{s^2 + (\eta^*_{K_j})^2}{\epsilon})\right) ds$$

which can be calculated through numerical integration as in the previous session.

We study the L^2-projection error of the oscillating solution (3.4.1) onto the new approximation space. As shown in Table 3.4.3, we observe second order of convergence for both the ES^1_h and ES^2_h spaces, indicating that for this easy implementation approach, enriching the basis does not help improve the convergence rates. As a result, we just focus on ES^1_h and omit the results obtained from ES^2_h.

Table 3.4.3: L^2 projection error of the easy implementation basis.

	ES^1		ES^2	
N	error	order	error	order
10	4.48E-01	–	4.59E-02	–
20	1.27E-01	1.82	1.06E-02	2.11
40	2.98E-02	2.09	2.50E-03	2.08
80	7.28E-03	2.03	5.98E-04	2.06

The same set of numerical tests as in the previous subsection are performed to validate and to show the effectiveness of this method based on an easy implementation. From Table 3.4.4, it is clear that the multi-scale DG with the approximation space ES_h^1 also provide a second order of convergence rate.

Table 3.4.4: L^2 errors and orders of accuracy obtained by the easy implementation.

Accuracy Test			Dirichlet B.C.			
ES^1	$\epsilon = 0.01$		$\epsilon = 0.01$		$\epsilon = 0.005$	
N	error	order	error	order	error	order
10	4.84E-01	–	4.71E-01	–	5.41E-01	–
20	1.50E-01	1.69	1.72E-01	1.45	1.83E-01	1.56
40	4.32E-02	1.80	5.12E-02	1.75	5.12E-02	1.84
80	1.13E-02	1.93	1.39E-02	1.88	1.36E-02	1.91
160	2.88E-03	1.97	3.60E-03	1.95	3.66E-03	1.93

3.5 Concluding remarks

In this chapter, we propose a multi-scale DG method for solving a class of two dimensional second order elliptic equations with rough coefficients which locally vary sharply in at most one direction, or in other words, are curvilinear unidirectional. Solving this type of problems is computationally expensive for most traditional numerical methods, which motivates us to exploit the flexibility of DG methods in choosing a special local approximation basis that allows a much more efficient algorithm.

Motivated by previous work in [29, 65], we build our local approximation space by utilizing special oscillatory basis functions indicated by the differential operators. As a result, we prove that the scheme yields optimal accuracy for the second order approximation on coarse meshes. The convergence rate is uniform and independent

of the small scale ϵ. Furthermore, the constraints on the computational geometry, indicated in the error estimates appeal artificial, in the sense of being assumed only for the convenience of analysis rather than enforcing restrictions on real application.

Various numerical tests have been performed to demonstrate the effectiveness of the proposed schemes, in particularly, on coarse meshes. In addition, an easy implementation is suggested and validated through numerical examples. The technique is helpful especially when it is hard to find the explicit local mapping. In future work, we plan to apply this methodology to real applications that might arise in engineering problems.

Chapter Four

Application: Non-Oscillatory Discontinuous Galerkin Schemes on the Cubed-Sphere

4.1 Introduction

In this chapter, we are interested in applying the discontinuous Galerkin (DG) scheme to real world, large scale simulations. Because of its computationally attractive features such as local and global conservation, high-order accuracy, high parallel efficiency (petascale capability) and geometric flexibility, the DG method is becoming increasingly popular in atmospheric modeling (for the spherical geometry application, see for example; Giraldo et al. [23], Nair et al. [44] and Läuter et al. [36]). A recent review by Nair et al. [46] presents various DG applications in atmospheric science with an extensive list of references.

Meanwhile, in the context of atmospheric research, a non-oscillatory, positivity preserving numerical solution is often a basic requirement. Even though the RK-DG scheme has many desirable properties, a non-linear TVB type limiter must be employed, for transport equations with strong shocks or contact discontinuities, to suppress oscillations. The slope limiters used for finite volume schemes can be easily extended to the *modal* form of DG methods [11]. However, it drastically reduces the high-order accuracy of the DG scheme [30, 33].

To maintain the properties of the DG scheme, the limiter should not reduce the order of accuracy in region where the solution is smooth. To address this issue, Qiu et al. [52] have developed high-order limiters based on the WENO (weighted essentially non-oscillatory) schemes, where the WENO type non-oscillatory reconstruction technique serves as a limiter for the RKDG method. A major disadvantage of this type of limiter is its requirement for a wider halo region (stencil) which potentially impedes parallel efficiency. For example, a third-order DG scheme requires a 5×5 stencils (i.e., halo size of width 2 in each direction) to apply a consistent WENO

limiter. Subsequently, Qiu et al. [51] improved this deficiency by developing the Hermite-WENO (or H-WENO) limiter, for which a more compact stencil is used. For a third-order DG scheme, the H-WENO limiter only requires a 3×3 stencil. The DG scheme combined with an H-WENO limiter has recently been used for a system of conservation laws in several applications [41, 4]. Note that WENO or H-WENO limiters are only "essentially" non-oscillatory by design, which implies the minor wiggles may appear in the solution even after the limiter is applied to the DG scheme. Therefore the terminology "non-oscillatory DG scheme" is used throughout in the text in a weak sense.

Although WENO type limiters can remove spurious oscillations, there is no guarantee that it always keeps the numerical solution within physical bounds. The numerical solution may still have small amplitude oscillations even after the limiter is applied. In other words, these schemes are not strictly positivity preserving. For many atmospheric tracers such as humidity and mixing ratios, the global maximum and minimum values are known in advance, and such tracers have "zero tolerance" for negative values. Therefore, for tracer transport models, positivity preservation is often required to obtain a physically meaningful numerical approximation. Very recently, Zhang et al. [75] developed a genuinely high-order bound-preserving (BP) filter for multi-dimensional RKDG methods, based on the one-dimensional limiter proposed in [40]. The BP filter clips extrema of the solution which exceeds physically legitimate bounds without violating the conservation property. A nice feature of the BP filter is that it is local, and can be turned to be a positivity-preserving filter when the lower bound is specified as zero. Zhang et al. [76], further extended this maximum-principle-satisfying filter to a variety of problems including systems of equations such as the Euler and shallow-water equations. The non-oscillatory and positivity-preservation properties for the proposed DG transport scheme are achieved

by applying the H-WENO limiter and BP filter, respectively.

In this chapter, we first introduce the basic DG scheme, with emphasis on a P^2 modal (third-order) version, and the implementation of an H-WENO limiter with a BP filter option. The basic ideas are developed in 2D Cartesian geometry and then extended to the cubed-sphere (2D) with several benchmark tests.

4.2 Non-Oscillatory DG Transport Scheme

We first consider the two-dimensional (2D) conservative transport equation:

$$\frac{\partial U}{\partial t} + \nabla \cdot \mathbf{F}(U) = S(U), \quad \text{in} \quad \mathcal{D} \times (0, T], \tag{4.2.1}$$

where $U = U(x, y, t)$ is a conserved quantity with given initial value $U_0(x, y) = U(x, y, 0)$, $\mathbf{F}$ is a flux function, $S(U)$ is a source term. The DG spatial discretization procedure consists of partitioning the domain $\mathcal{D}$ into non-overlapping elements (cells) $I_{i,j}$, and seeking an approximate solution $U_h \approx U$ on each element. We assume the elements to be rectangular such that $I_{i,j} = [x_{i-1/2}, x_{i+1/2}] \otimes [y_{j-1/2}, y_{j+1/2}]$. The approximate solution is $U_h \in V_h^k(I_{i,j}), \forall I_{i,j} \in \mathcal{D}$, such that $U_h|_{I_{i,j}} \in P^k(I_{i,j})$, where V_h^k is the vector space of polynomials P^k up to degree k, defined over $I_{i,j}$.

A weak Galerkin formulation of the problem is obtained by multiplying (4.2.1) by a test function $\varphi_h \in V_h^k$, and invoking the divergence theorem over $I_{i,j}$ leading to:

$$\int_{I_{i,j}} \frac{\partial U_h}{\partial t} \varphi_h \, dx \, dy = \int_{I_{i,j}} \mathbf{F}(U_h) \cdot \nabla \varphi_h \, dx \, dy - \oint_{\partial I_{i,j}} \varphi_h \, \mathbf{F}(U_h) \cdot \vec{n} \, ds + \int_{I_{i,j}} S(U_h) \, \varphi_h \, dx \, dy, \tag{4.2.2}$$

where $\partial I_{i,j}$ is the boundary corresponding to element $I_{i,j}$ and $\vec{n}$ is the outward point-

ing normal vector of the boundary. The discontinuity at the element boundaries (interfaces) within $\mathcal{D}$ is resolved by applying suitable numerical fluxes (or approximate Riemann solvers).

4.2.1 A Third-Order Modal Formulation

Equation (4.2.2) is the crux of the DG algorithm; the accuracy and efficiency of which is determined by the particular choice of P^k and the quadrature rules for the integrals. An arbitrary high-order modal DG discretization on the cubed-sphere is given in [45]. Nevertheless, our focus here is the development of a third-order modal non-oscillatory DG scheme. Details of the 'modal' and 'nodal' variants of DG scheme and their relative merits can be found in [46].

A major limitation of the DG scheme is the stringent CFL stability constraint associated with explicit time stepping. For high-order DG schemes employing polynomials of degree $k > 1$, an approximate CFL limit estimate is $1/(2k + 1)$, [11]. Reducing the order-accuracy has some benefits. It significantly improves CFL stability and allows one to implement limiting algorithms based on those designed for FV methods. A *moderate* order DG scheme, such as a third-order one ($k = 2$), has a CFL limit (approximately 0.2) comparable to some high-order FV schemes [7]. In certain cases, a third-order DG scheme provides a solution which is qualitatively comparable to that of a fourth-order or fifth-order WENO scheme. Therefore, in this chapter our main focus is the development of a *practical* third-order DG scheme, which we will refer to as the "P^2-DG" scheme. For that we utilize the *modal* version of DG method which is amenable to limiting methods used for FV schemes.

In order to simplify the integrals in (4.2.2), we introduce new independent vari-

ables (ξ, η), and every element is mapped to the standard element $[-1, 1]^2$ such that

$$\xi = \frac{2(x - x_i)}{\Delta x_i}, \quad \eta = \frac{2(y - y_j)}{\Delta y_j}; \quad \xi, \eta \in [-1, 1], \tag{4.2.3}$$

where $x_i = (x_{i-1/2} + x_{i+1/2})/2$ and $y_j = (y_{j-1/2} + y_{j+1/2})/2$; $\Delta x_i = x_{i+1/2} - x_{i-1/2}$ and $\Delta y_j = y_{j+1/2} - y_{j-1/2}$. The approximate solution $U_h|_{I_{i,j}} = U_{i,j}$ for an element $I_{i,j}$ can be represented in terms of modal basis functions from the set $\mathcal{B} = \{1, \xi, \eta, \xi\eta, (3\xi^2 - 1)/2, (3\eta^2 - 1)/2\}$, composed of the Legendre (orthogonal) polynomials.

$$\begin{aligned} U_h(\xi, \eta, t)|_{I_{i,j}} &= U_{i,j}^{0,0}(t) + U_{i,j}^{1,0}(t)\,\xi + U_{i,j}^{0,1}(t)\,\eta + U_{i,j}^{1,1}(t)\,\xi\,\eta + \\ &\quad U_{i,j}^{2,0}(t)\,\frac{3\xi^2 - 1}{2} + U_{i,j}^{0,2}(t)\,\frac{3\eta^2 - 1}{2}, \end{aligned} \tag{4.2.4}$$

where $U_{i,j}^{l,m}(t)$, $0 \leq l + m \leq 2$, are the six *dofs* (or moments) associated with the P^2 DG scheme. The moments are defined as follows

$$U_{i,j}^{l,m}(t) = \frac{\Delta x\,\Delta y}{4} \int_{-1}^{1} \int_{-1}^{1} U_{i,j}(\xi, \eta, t)\,\varphi^{l,m}(\xi, \eta)\,d\xi\,d\eta, \tag{4.2.5}$$

where the test functions $\varphi^{l,m} \in \mathcal{B}$. From (4.2.5) it is clear that the first moment $U_{i,j}^{0,0}$ is the "cell-average," in a FV sense. The integrals in (4.2.2) associated with the P^2-DG scheme are evaluated either by using the Gauss-Lobatto-Legendre (GLL) or the Gauss-Legendre (GL) grid consistent with the order of accuracy of the scheme, as shown in Fig. (4.2.1).

For the flux term in (4.2.2), we use the local Lax-Friedrichs flux given by

$$\hat{F}(U_h^-, U_h^+) = \frac{1}{2}[(\mathbf{F}(U_h^+) + \mathbf{F}(U_h^-)) \cdot \vec{n} - \bar{\alpha}(U_h^+ - U_h^-)],$$

where $\bar{\alpha}$ is the maximum value of the flux Jacobian (which is the local maximum wind speed for an advection problem) and U_h^-, U_h^+ denote the left (bottom) and

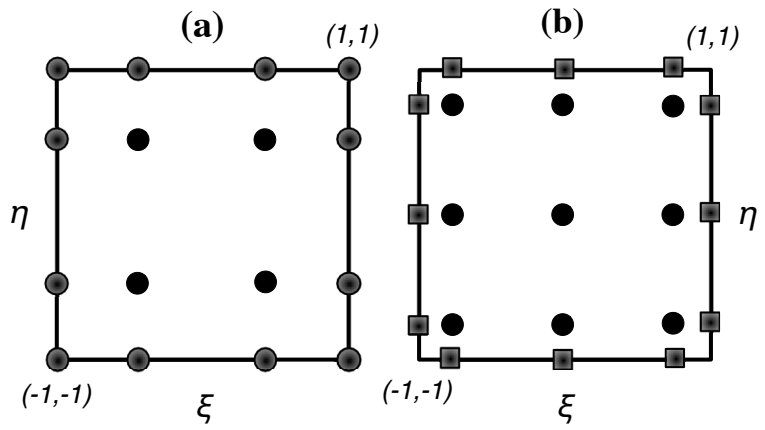


Figure 4.2.1: For the DG discretization, every rectangular element is mapped onto a standard element $[-1,1]^2$ with the local (ξ,η)-coordinates. Panel (a) shows the Gauss-Lobatto-Legendre (GLL) quadrature grid with 4×4 points, where on the boundary the solution points and flux points coincide. Panel (b) shows the Gauss-Legendre (GL) quadrature grid with 3×3 internal points (filled circles) and 3 flux points on each side (filled squares). The values at the flux points are interpolated using the modal basis functions.

right (upper) values of the numerical solution as suggested in [11]. This resolves the discontinuity at the element edges. Further simplification of the weak form (4.2.2) leads to the semi-discrete form which is a system of ordinary differential equations (ODE) in time for each moment. More details of the discretization can be found in [46]. In short, the six modes $U_h^{l,m}$ of the DG solution on any element are described by the ODEs

$$\frac{d}{dt}U_h^{l,m} = L(U_h^{l,m}), \tag{4.2.6}$$

where L represents the DG spatial discretization resulting from (4.2.2). The approximate solution U_h on each element can be computed by solving (4.2.6) and using (4.2.4). In this chapter, we employ the third-order Strong Stability Preserving (SSP) Runge-Kutta time integration scheme as in [26] to solve (4.2.6).

4.2.2 The H-WENO Limiter

To suppress the oscillations in the numerical solution in the presence of a shock or discontinuity, we employ the Hermite-WENO (H-WENO) limiting strategy. The H-WENO limiting strategy for DG scheme was first introduced by Qiu et al in [51]. It is a variant of the traditional WENO type limiter [52], in the sense that they share the same methodology. Once a cell is identified as an oscillatory (or "troubled") cell, using a shock detection technique such as the total variation bounded (TVB) type limiter [12], then all the higher-order moments are modified except for the first moment (i.e., cell average). Preservation of the cell-average is required to guarantee conservation. We briefly outline the procedure as follows, the details can be found in [58].

The first step is to reconstruct several polynomials P_n using the information from the neighboring cells, for which a family of candidate stencils are required. Each big stencil contains several small stencils (or sub-stencils) S_n consisting of the cell $I_{i,j}$ and its neighbors, as depicted schematically in Fig. 4.2.2. The WENO scheme uses a convex combination of nonlinear weights w_n from each stencil which depends on the local smoothness of the solution, and ultimately creates the non-oscillatory solution. The *smoothness indicators* β_n, which are a measure of smoothness of the solution, are computed for each stencil; note that a smaller value of β_n indicates a smoother function P_n in S_n. The smoothness indicators are then used to convert pre-computed linear weights (γ_n) to nonlinear weights (w_n). As a limiter, WENO modifies the high-order moments of the DG solution using a combination of nonlinear weights [52].

The major difference between the H-WENO and the WENO scheme is the more compact stencil of H-WENO for a given order of accuracy, making the H-WENO algorithm more desirable for a parallel computing environment. Unlike the WENO

method, which employs only the cell-averages from the neighboring cells, the H-WENO scheme relies on cell-averages as well as derivative information (or "Hermite" information). For the DG discretization, derivatives are readily available, making the H-WENO limiter more computationally attractive. The H-WENO limiter retains all the nice qualities of the WENO limiter such as conservation, and the non-oscillatory property, and does not reduce the order of accuracy of the underlying scheme in the smooth region. For instance, the H-WENO stencil adopted for our P^2-DG scheme uses a 3×3 stencil, which is labeled in Fig. 4.2.2, as compared to the 5×5 stencil required by the WENO limiter.

To better illustrate the reconstruction process of the H-WENO limiter, we require that the reconstructed polynomial retains the cell-averages of all the cells contained in each stencil. Our focus is on the implementation of the 2D H-WENO limiter for the P^2-DG scheme. This requires a fourth-order accurate H-WENO reconstruction employing a stencil with 3×3 cells, and a set of 8 small stencils $\{S_n\}_{n=1}^{8}$, as shown in Fig. (4.2.2). For convenience, we denote the cells with a single index I_ℓ, $\ell = \ell(i,j)$ for the reconstruction procedure. The H-WENO reconstruction involves the reconstruction of polynomials P_n in each stencil S_n subject to some integral constraints. For example, on S_1, we seek $P_1(x,y)$ satisfying the following constraints:

$$\begin{aligned} \int_{I_\ell} P_1(x,y)\,dxdy &= a_0\,U_\ell^{0,0}, \quad \ell = 1,2,4,5 \\ \int_{I_4} P_1(x,y)\varphi_1^{1,0}\,dxdy &= a_1\,U_4^{1,0}, \\ \int_{I_2} P_1(x,y)\varphi_2^{0,1}\,dxdy &= a_2\,U_2^{0,1}, \end{aligned} \tag{4.2.7}$$

where a_0, a_1 and a_2 are known coefficients. Note that the above constraints not only use the cell-average $(U_\ell^{0,0})$ but also exploit the derivative information $(U_\ell^{0,1}, U_\ell^{1,0})$. Without indicating the cell dependence, a general from of $P_n(x,y)$ in terms of the

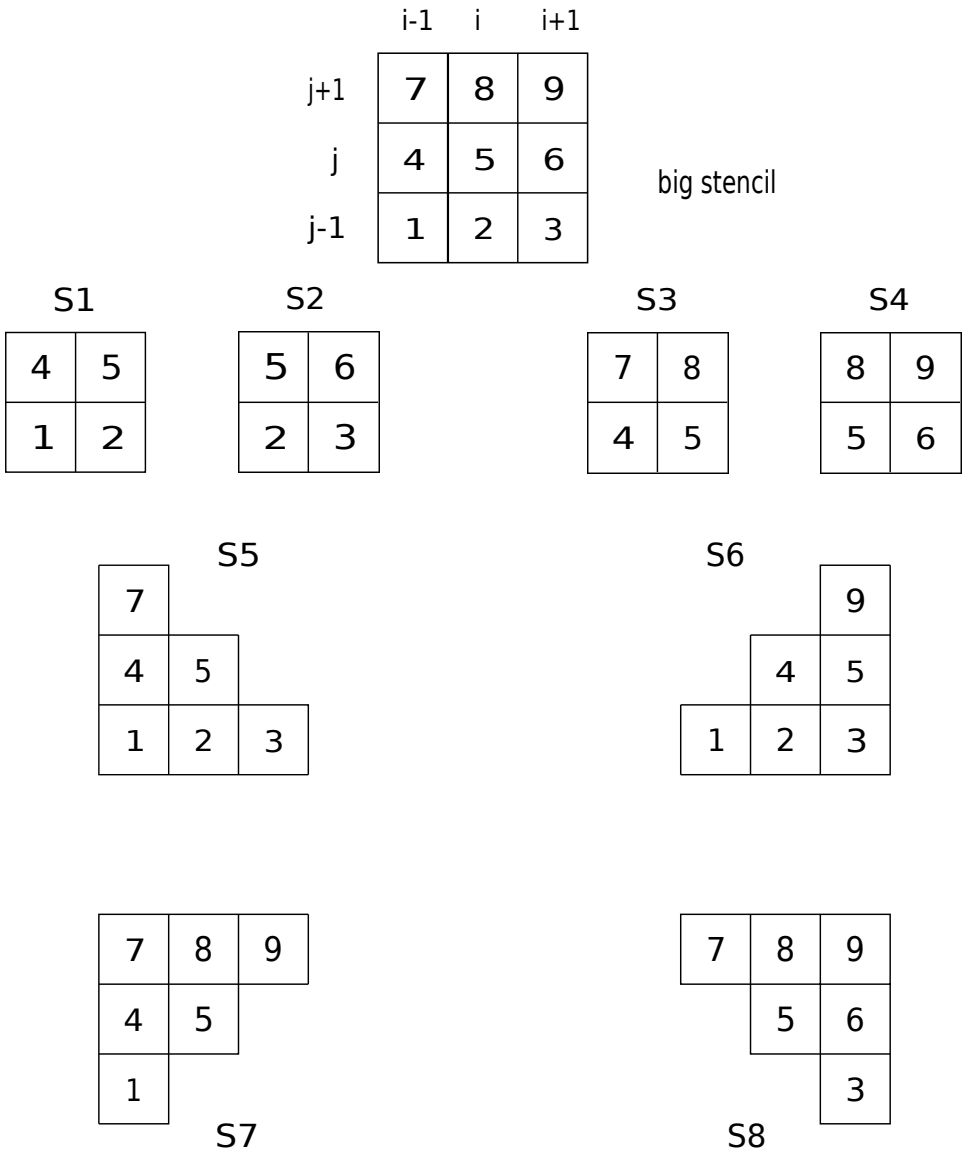


Figure 4.2.2: The stencils used for the H-WENO reconstruction to limit a cell $I_{i,j}$, are shown for a single index $I_{\ell=5}$. For the P^2-DG scheme, the H-WENO limiter requires a stencil with 3×3 cells, and 8 small stencils $S_1, S_2, \ldots, S_8$. Each small stencil contains a combination of cells from the larger stencil including the cell to be limited (I_5).

local coordinates (ξ, η) can be expressed as,

$$P_n(\xi,\eta) = U_n^{0,0} + U_n^{1,0}\xi + U_n^{0,1}\eta + U_n^{1,1}\xi\,\eta + U_n^{2,0}\,\frac{3\xi^2-1}{2} + U_n^{0,2}\,\frac{3\eta^2-1}{2}. \quad (4.2.8)$$

Using the integral constraints, (4.2.8) can be further modified for each stencil S_n. The details for reconstruction process are given in the Appendix-A.

We use a TVB type limiter to identify an oscillatory cell $I_{i,j}$, while keeping the cell-average $(U_{i,j}^{0,0})$ unchanged. The modification of higher order moments involves several steps. First we compute the optimal linear weights γ_n assigned for each reconstructed polynomial $P_n(x, y)$. The details of this can be found in [51]; we list only the key results here. For the first-order moments $U_{i,j}^{1,0}$ and $U_{i,j}^{0,1}$ the linear weights are:

$$\gamma_1 = \gamma_2 = \gamma_3 = \gamma_4 = \frac{11}{76}; \quad \gamma_5 = \gamma_6 = \gamma_7 = \gamma_8 = \frac{2}{19}$$

For second-order moments $U_{i,j}^{1,1}$, $U_{i,j}^{2,0}$ and $U_{i,j}^{0,2}$, $\gamma_n = 1/8$, $n = 1, \ldots, 8$. The smoothness indicators β_n associated with each $P_n(x, y)$ for the first-order moments $U_{i,j}^{1,0}$ and $U_{i,j}^{0,1}$ are, respectively,

$$\beta_n = \sum_{m=1}^{2} |I_{i,j}|^{m-1} \int_{I_{i,j}} (\frac{\partial^m}{\partial x^m} P_n(x,y))^2 dx\,dy,\ \beta_n = \sum_{m=1}^{2} |I_{i,j}|^{m-1} \int_{I_{i,j}} (\frac{\partial^m}{\partial y^m} P_n(x,y))^2 dx\,dy. \quad (4.2.9)$$

For the second-order moments $U_{i,j}^{1,1}$, $U_{i,j}^{2,0}$ and $U_{i,j}^{0,2}$ the smoothness indicators are computed as follows:

$$\beta_n = \sum_{|m|=2}^{2} |I_{i,j}|^{m-1} \int_{I_{i,j}} (\frac{\partial^{|m|}}{\partial x^{m_1} \partial y^{m_2}} P_n(x,y))^2 dx\,dy. \quad (4.2.10)$$

See Appendix-B for the details of the smoothness indicator computation. The next step is to convert the linear weights γ_n into normalized nonlinear weights w_n using

smooth indicators β_n:

$$w_n = \frac{\tilde{w}_n}{\sum_n \tilde{w}_n}, \quad \tilde{w}_n = \frac{\gamma_n}{(\varepsilon + \beta_n)^2},$$

where $\varepsilon = \mathcal{O}(10^{-6})$, to prevent a zero denominator. Finally, we replace all higher order moments $U_{i,j}^{l,m}$, $1 \leq (l+m) \leq 2$ by modified moments $\tilde{U}_{i,j}^{l,m}$ using:

$$\tilde{U}_{i,j}^{l,m} = \sum_{n=1}^{8} w_n U_n^{l,m} \tag{4.2.11}$$

The limited approximate solution corresponding to (4.2.4) is obtained by replacing the unlimited coefficients with the limited coefficients (4.2.11), hence completing the H-WENO limiting process for the P^2-DG scheme.

Qiu et al. [51] have shown that the H-WENO limiter coupled with a DG scheme is indeed third-order accurate for smooth problems. However, just like the WENO limiter, the H-WENO limiter is also only "essentially" non-oscillatory, which means it may not eliminate all small oscillations near the physical bounds. This is the motivation for us to further implement a bound-preserving filter for the P^2-DG scheme combined with the H-WENO limiter.

4.2.3 The Bound-Preserving (BP) Filter

To preserve the initial bounds of the numerical solutions and eliminate negative densities (if positivity is a requirement), we apply a Bound-Preserving (BP) filter as an additional option. The BP filter has several attractive features. It is local, conservative, computationally cheap and easy to implement [75]. For P^2-DG, the moments are evolving with respect to time. However, the grid-point values of the

approximate solution U_h on $I_{i,j}$ at any instant can be computed from the polynomial representation (4.2.4). Let $p_{i,j}(x,y)$ be the modal DG polynomial on the cell $I_{i,j}$ with cell-average $\bar{u}_{i,j}$, and let $S_{i,j}$ be the local computational stencil in the grid-point (physical) space corresponding to $I_{i,j}$.

The BP filter replaces $p_{i,j}(x,y)$ with a modified polynomial $\tilde{p}_{i,j}(x,y)$ such that

$$\tilde{p}_{i,j}(x,y) = \hat{\theta}\, p_{i,j}(x,y) + (1-\hat{\theta})\, \bar{u}_{i,j}, \quad \hat{\theta} = \min\left\{ |\frac{M-\bar{u}_{i,j}}{M_{i,j}-\bar{u}_{i,j}}|, |\frac{m^*-\bar{u}_{i,j}}{m^*_{i,j}-\bar{u}_{i,j}}|, 1 \right\}, \tag{4.2.12}$$

where the local extrema are $M_{i,j} = \max_{S_{i,j}} p_{i,j}(x,y)$ and $m^*_{i,j} = \min_{S_{i,j}} p_{i,j}(x,y)$. In (4.2.12) M and m^* are the global maximum and minimum values of the initial condition, typically known in the context of certain atmospheric tracer transport. From (4.2.12) it is clear that $\tilde{p}_{i,j}(x,y)$ preserves the cell-average $\bar{u}_{i,j}$, which is a basic requirement for local conservation, for $\hat{\theta} \in [0,1]$. More details of the filter function (4.2.12) and other applications can be found in a recent review by Zhang et al. [76]. The positivity preserving option is a special case of the BP filter, and can be achieved my setting $m^* = 0$. The local grid-point stencil $S_{i,j}$ in the P^2-DG context has a tensor product of 4 GLL quadrature points in each dimension or the tensor product of 3 GL points plus the Gauss points on the four boundaries for flux interpolation as shown in Fig. 4.2.1.

The filter (4.2.12) is in fact based on the 1D filter developed by Liu et al. [40] for a FV scheme. Zhang et al. [75] have proved that this filter satisfies the strict maximum principle and is genuinely high-order and extensible to DG applications. We note that, although the BP filter maintains the bounds of the solution in the range $[m^*, M]$, there is no guarantee that it will remove all the internal oscillations within the cells. However, the combination of the H-WENO limiter and the local BP filter addresses this issue.

4.3 Non-Oscillatory DG Scheme for Cubed-Sphere

The cubed-sphere geometry [57] has become a popular choice for the spherical grid system in global modeling because it offers a quasi-uniform rectangular grid structure on the sphere without pole problems. This grid structure is suitable for high-order element-based Galerkin methods as well as cell-centered FV methods. There are different variants of the cubed-sphere topology, but we will consider the cubed-sphere geometry employing the equiangular central projection as described in [45]. The sphere $\mathcal{D}$ is decomposed into six identical regions by an equiangular central (gnomonic) projection of the faces of an inscribed cube. The central angles of projection $x^1 = x^1(\lambda, \theta)$, $x^2 = x^2(\lambda, \theta)$ are the local coordinates for each face such that $x^1, x^2 \in [-\pi/4, \pi/4]$; where λ and θ are the longitude and latitude of the sphere with radius R, respectively. This results in a nonorthogonal curvilinear coordinate system (x^1, x^2), which is free of singularities; however, the edges of the six faces are discontinuous.

The metric tensor associated with the central mapping is given by

$$g_{ij} = \frac{R^2}{\rho^4 \cos^2 x^1 \cos^2 x^2} \times \begin{bmatrix} 1 + \tan^2 x^1 & -\tan x^1 \tan x^2 \\ -\tan x^1 \tan x^2 & 1 + \tan^2 x^2 \end{bmatrix},$$

where $\rho^2 = 1 + \tan^2 x^1 + \tan^2 x^2$, with tensor indices $i, j \in \{1, 2\}$. The metric term (Jacobian of the transformation) is then $\sqrt{g} = |g_{ij}|^{1/2}$. The horizontal velocity vector on the sphere $\mathbf{v}(\lambda, \theta) = (u, v)$ can be expressed in terms of covariant (u_1, u_2) and contravariant (u^1, u^2) vectors, which are related through $u_i = g_{ij}\, u^j$, $u^i = g^{ij}\, u_j$ where $g^{ij} = g_{ij}^{-1}$. For each face of the cubed-sphere, covariant and contravariant vectors

can be computed from (u, v) as follows:

$$\begin{bmatrix} u^1 \\ u^2 \end{bmatrix} = \mathbf{A}^{-1} \begin{bmatrix} u \\ v \end{bmatrix}, \quad \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = \mathbf{A}^T \begin{bmatrix} u \\ v \end{bmatrix}, \quad \mathbf{A} = R \begin{bmatrix} \cos\theta\,(\partial\lambda/\partial x^1) & \cos\theta\,(\partial\lambda/\partial x^2) \\ \partial\theta/\partial x^1 & \partial\theta/\partial x^2 \end{bmatrix}, \tag{4.3.1}$$

where $\mathbf{A}$ is local to each face of the cubed-sphere such that $\mathbf{A}^T\mathbf{A} = g_{ij}$. The details of the local transformation and the matrix $\mathbf{A}$ are given in [45], and will not be discussed herein.

The general (tensorial) form of the transport equation (4.2.1) in curvilinear co-ordinates for a scalar field ψ, without source term can be written as:

$$\frac{\partial \psi}{\partial t} + \frac{1}{\sqrt{g}} \frac{\partial}{\partial x^i}(u^i \sqrt{g}\psi) = 0.$$

Since the metric term $\sqrt{g}$ is time independent, the above equation on the cubed-sphere ($\mathcal{D}$) can be expressed in the following flux form,

$$\frac{\partial}{\partial t}(\sqrt{g}\psi) + \frac{\partial}{\partial x^1}(u^1 \sqrt{g}\psi) + \frac{\partial}{\partial x^2}(u^2 \sqrt{g}\psi) = 0. \tag{4.3.2}$$

If we consider the scalar $U = \sqrt{g}\,\psi$, then (4.3.2) becomes the standard form as given in (4.2.1), with fluxes $\mathbf{F} = (u^1 U, u^2 U)$. Moreover (4.3.2) may be treated as a simple 2D Cartesian case in (x^1, x^2)-space [38]. For the DG discretization, each face of the computational domain is partitioned into non-overlapping $N_e \times N_e$ cells (elements) such that $N_e \times N_e \times 6$ cells span the cover domain $\mathcal{D}$.

4.4 Numerical Experiments

4.4.1 Cartesian Tests

To validate the non-oscillatory DG scheme, we solve (4.2.1) without source term on Cartesian domains using several benchmark tests including a solid-body rotation test suggested in [37], and a deformational flow test in [6]. The initial scalar fields include smooth and non-smooth distributions to check the effectiveness of the limiter and BP filter. For the following tests we use the 3×3 GL quadrature grid shown in the right panel of Fig. 4.2.1, and the third-order SSP Runge-Kutta [26] method for time integration.

Advection of a 1D irregular signal

To see the effects of the BP filter and the H-WENO limiter on the P^2-DG scheme, we first solve a simple one-dimensional form of the conservation law (4.2.1), $\psi_t + (u\psi)_x = 0$, on a periodic domain $[-\pi, \pi]$. The initial value of the scalar $\psi(x, t = 0) = \psi_0$ is given as a three-level step function (or irregular signal) in $[0, 1]$, with values $\psi_0 = 0, 0.5, 1$ for $x \in [-\pi, \pi]$, as indicated as thin-lines in Fig. 4.4.1. The computational domain consists of 80 cells with a uniform velocity $u = 1$. In Fig. 4.4.1, the numerical solution ψ_n after one revolution (period) is shown as dashed-lines. Although the BP filter keeps the numerical solution within the initial bounds $[0, 1]$ as shown in Fig. 4.4.1b, solution remains oscillatory at the level $\psi_n = 0.5$ similar to the P^2-DG case (Fig. 4.4.1a). However, the H-WENO limiter with the BP filter removes the oscillations (Fig. 4.4.1c) while strictly preserving the initial bounds. The P^2-DG solution combined with the H-WENO limiter (not shown) appears very similar to

Fig. 4.4.1c , but in this case the solution is not strictly positivity-preserving, and there are minor undershoots (overshoots) at $\psi_n = 0$. The BP filter preserves the solution within the initial bounds, but it has no control over the *internal* oscillations of the solution or it cannot enforce that the solution is non-oscillatory. In general, the P^2-DG solution with H-WENO and BP combination is "essentially" non-oscillatory, but preserves the bounds of the initial data.

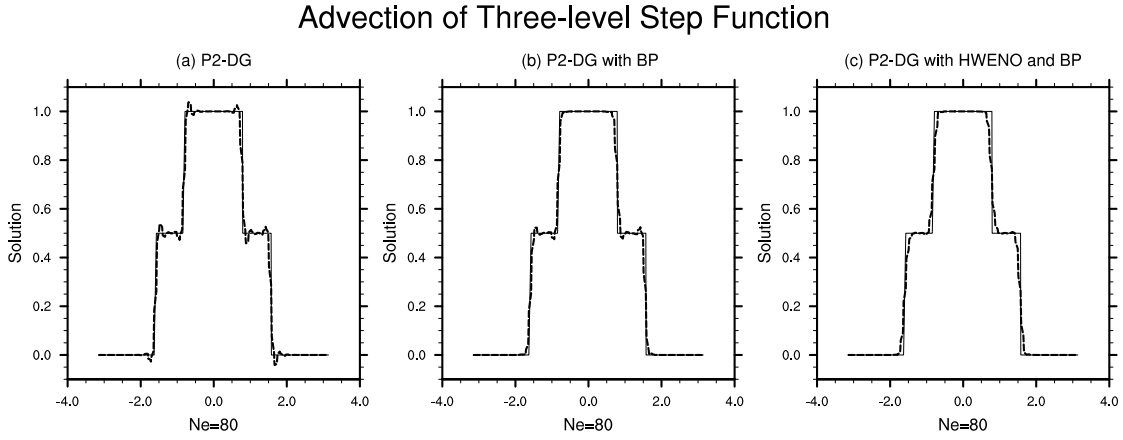


Figure 4.4.1: Results after one complete period for the uniform advection of an irregular signal (three-level step function), on a 1-d periodic domain $[-\pi, \pi]$ employing 80 cells. Reference solution is marked as thin solid-lines and numerical solution is marked as dashed-lines. Numerical solution with (a) P^2-DG (b) P^2-DG and BP filter and (c) P^2-DG with both H-WENO limiter and BP filter.

Solid-body rotation of a Gaussian hill

For the 2D case we consider the conservation law $\psi_t + (u\psi)_x + (v\psi)_y = 0$, on a periodic domain $\mathcal{D}$. The first 2D test-case is the solid-body rotation of a Gaussian hill on a square domain $\mathcal{D} = [-1, 1]^2$. The velocity field is prescribed as $(u, v) = (-\omega y, \omega x)$, where the constant angular velocity $\omega = 1$. For the solid-body rotation test, the initial scalar field translates on a circular trajectory without incurring any deformation. Moreover the exact solution is available at any time. The initial condition is given by:

$$\psi(x, y, t = 0) = a_c \exp\left\{-b_c[(x - x_0)^2 + (y - y_0)^2]\right\}$$

where a_c and b_c is taken to be 1.0 and 100/3 respectively. For the purpose of testing the order of accuracy, we take $x_0 = 0$, $y_0 = 0$, and assume periodic boundary conditions. Note that for the solid-body rotation test we have used small timesteps to minimize the temporal errors. An approximate estimate is given as $\Delta t = 0.5\, C_n\, \Delta$, where $C_n = 0.15$ is a CFL number less than the theoretical maximum (0.2) for the P^2-DG, and Δ is the minimum grid spacing. From Fig. 4.4.2, it is clear that both the H-WENO limiter and the BP filter are third-order accurate.

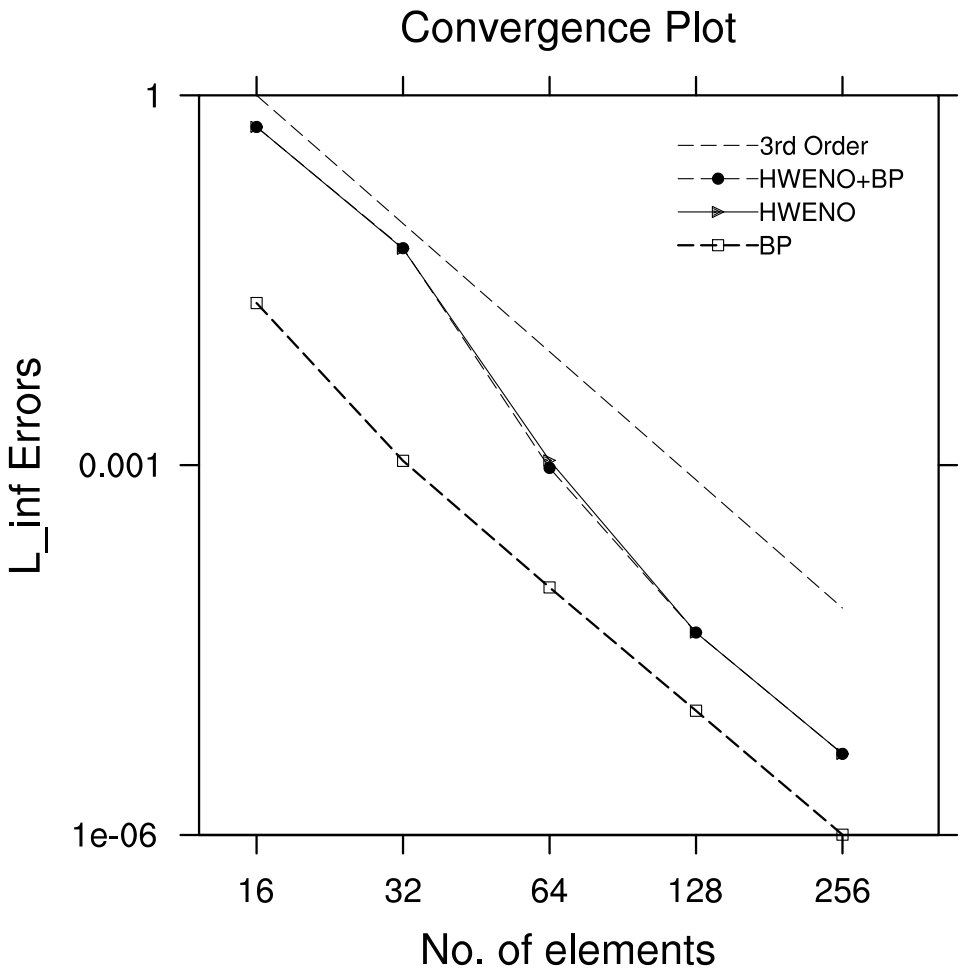


Figure 4.4.2: Convergence plots for 2D Cartesian solid-body rotation of a Gaussian hill. The P^2-DG transport scheme combined with the H-WENO limiter, or BP filter, or both is used for the convergence tests.

Solid-Body Rotation of a Non-Smooth Distribution

The second test-case on the 2D Cartesian domain is also the solid-body rotation, under the same velocity field, but with a non-smooth scalar field comprised of a square block and a cone [37]. This field is given by

$$\psi(x, y, t = 0) = \begin{cases} 1 & \text{if } \max(|x - 0.35|, |y - 0|) \leq 0.25 \\ 1 - \hat{r} & \text{if } \hat{r} < 1 \\ 0 & \text{otherwise} \end{cases}$$

where $\hat{r} = \sqrt{(x - x_0)^2 + (y - y_0)^2}/0.35$ with $x_0 = -0.45$, $y_0 = 0$.

Figure 4.4.3 shows the contour plots of the numerical results and the exact solution after completing a full circle of rotation. The grid resolution is 80×80, which means $\Delta x = \Delta y = 0.025$, on a square domain $[-1, 1]^2$. In Fig. 4.4.4, a 3D projection of the numerical solution is shown. It can be seen clearly from the results that the H-WENO limiter effectively removes the non-physical oscillations which occurs near the discontinuities. Figure 4.4.4 shows a 3D projection of the solution on a 64×64 mesh, (a) with a DG scheme without limiter or filter, (b) with BP filter and (c) with both H-WENO limiter and BP filter. The BP filter alone helps to maintain the numerical solution within the physical bounds $[0, 1]$ as shown in Fig. 4.4.4(b), but there may be oscillations within this bounds. However, the H-WENO and BP combination eliminates oscillations while being strictly positive, as is evident from Fig. 4.4.4(c). Thus, using the non-oscillatory DG scheme achieves a substantial improvement in the quality of the numerical solution over DG scheme alone.

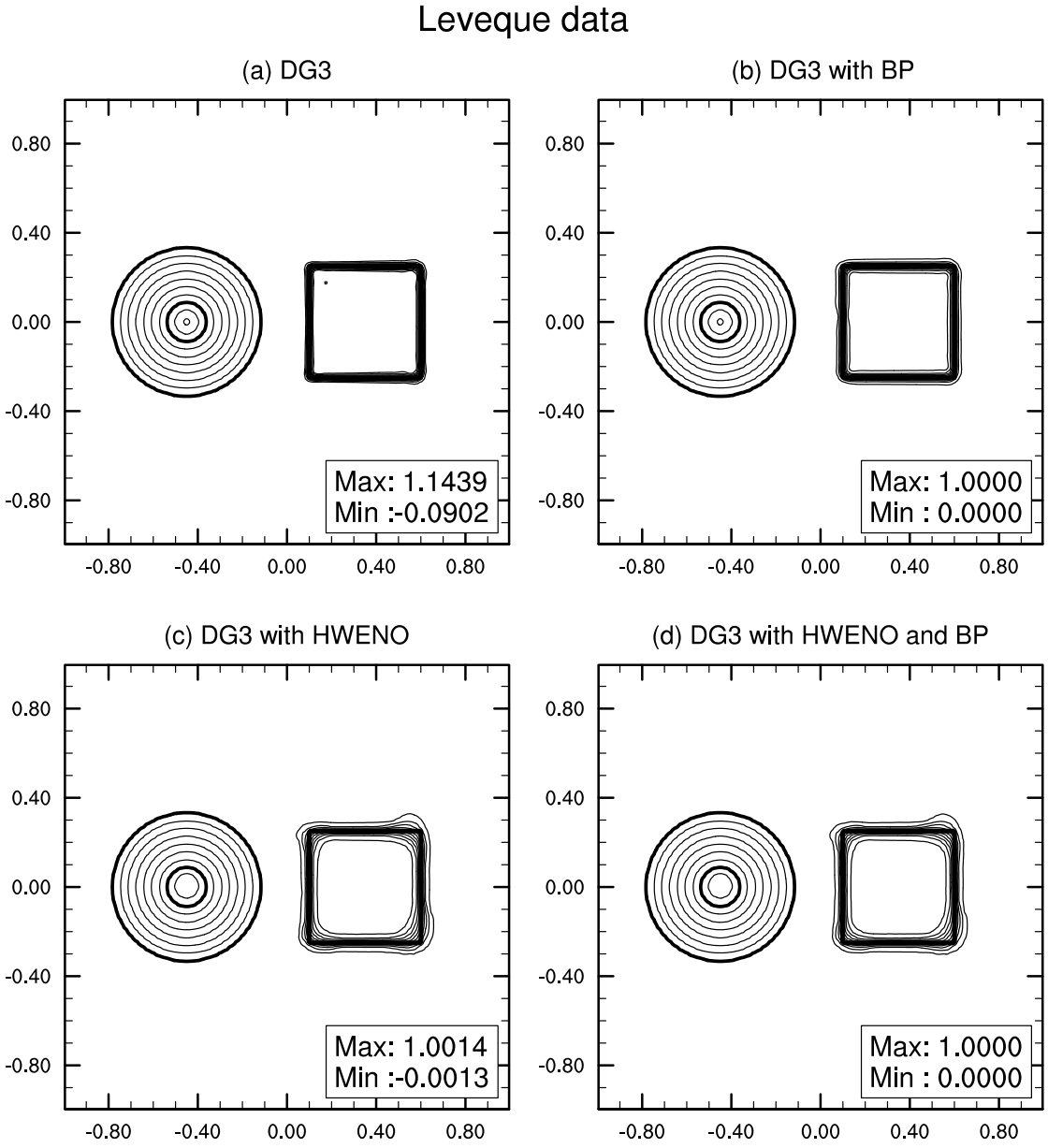


Figure 4.4.3: Numerical results of solid-body rotation of a non-smooth distribution (a circular cone and a square block) after one revolution with $\Delta x = \Delta y = 0.025$ using (a) the P^2-DG scheme only, (b) P^2-DG with the BP filter, (c) P^2-DG with the H-WENO limiter, (d) P^2-DG with the H-WENO limiter and the BP filter. Solid lines show contour values from 0.05 to 0.95 with increments of 0.1. Thick solid lines indicate the contour value of 0.05 and 0.75 of the exact (reference) solution.

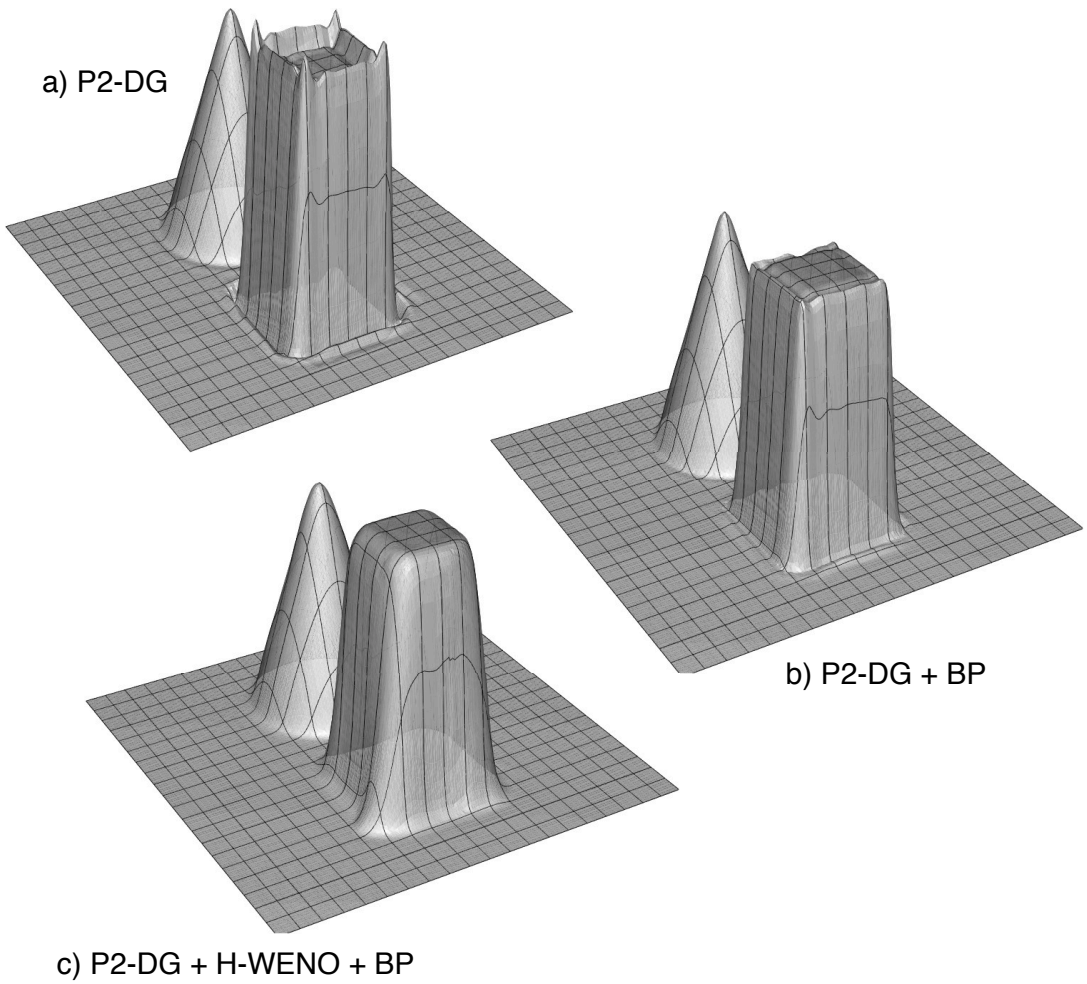


Figure 4.4.4: A 3D perspective of the numerical solution shown in Fig. 4.4.3, but on a 64×64 grid. (a) P^2-DG oscillatory solution, (b) P^2-DG with the BP filter, where the bounds are preserved but oscillations are visible within the bounds $[0, 1]$. (c) A non-oscillatory solution using the P^2-DG scheme combined with the H-WENO limiter and the BP filter.

Deformational Flow Tests

The deformational flow is considered to be a more challenging test-case than the solid-body rotation tests. Instead of a constant wind field, the velocity is time dependent in this test case. Initial circular distribution deforms into a crescent shape, and returns to the initial position when the flow reverses [6]. The analytic solution for this test is only available at the final time T, where it recovers initial condition. The computational domain $\mathcal{D} = [0, 1]^2$, and the velocity field is defined to be

$$u(x, y, t) = u_\theta(r, t)\ \sin(\theta), \quad v(x, y, t) = -u_\theta(r, t)\ \cos(\theta),$$

where $r = \sqrt{(x - 0.5)^2 + (y - 0.5)^2},\ \theta = \tan^{-1}[(y - 0.5)/(x - 0.5)]$, and

$$u_\theta(r, t) = \frac{4\pi r}{T}\left[1 - \cos(\frac{2\pi t}{T}\frac{1 - (4r)^6}{1 + (4r)^6})\right].$$

The initial scalar distribution is given by

$$\psi(x, y, t = 0) = \begin{cases} \psi_0 + (\frac{1+\cos(\pi\tilde{r})}{2})^2 & \text{if } \tilde{r} \le 1 \\ \psi_0 & \text{if } \tilde{r} > 1. \end{cases}$$

where $\tilde{r} = 5\sqrt{(x - 0.3)^2 + (y - 0.5)^2}$. We set the background field $\psi_0 = 0$, and the timestep Δt is set in such a way that the maximum of CFL is 0.25. Figure 4.4.5 shows the numerical solution with the P^2-DG and BP combination, demonstrating how the scalar filed deforms during one full evolution cycle. Figure 4.4.6 shows the contour plots of the exact solution and the numerical solution on a 100×100 grid at $t = T$. From the simulation results, we can see that for this quasi-smooth problem, the DG scheme coupled with the BP filter works better than the other schemes. This is because the H-WENO limiter excessively diffuses the solution near sharp

gradients. A more selective limiting criteria for the oscillatory cell could help avoid this issue.

4.4.2 2D Spherical Tests

For validating transport schemes, a solid-body rotation test and a deformational flow test are often used. We consider the solid-body rotation test suggested in [67] and [49], and a new challenging deformational flow tests described in [42] and [43]. For the P^2-DG scheme with limiter (filter), we solve the transport equation (4.3.2) on the cubed-sphere. All the computations for the P^2-DG scheme are performed using a 4×4 GLL grid as shown in the left panel of Fig. 4.2.1. Normalized standard errors l_1, l_2 and l_∞, and relative minimum ($\psi_{\min}$) and maximum ($\psi_{\max}$) errors [67] are used for validating the numerical scheme. The relative min/max errors are defined as

$$\psi_{\min} = \frac{\min(\psi) - \min(\psi_0)}{\Delta\psi_0}, \quad \psi_{\max} = \frac{\max(\psi) - \max(\psi_0)}{\Delta\psi_0}, \tag{4.4.1}$$

where $\Delta\psi_0 = \max(\psi_0) - \min(\psi_0)$, ψ is the numerical solution and ψ_0 is the initial solution.

Solid-Body Rotation: Cosine-Bell

We first consider the cosine-bell test [67], which is a *de facto* standard test case for spherical advection problems. Since the exact solution is know at all times, error measures can be computed. The initial condition is formulated as:

$$\psi(\lambda, \theta, t = 0) = \begin{cases} (h_0/2)[1 + \cos(\pi r_d / r_0)] & \text{if } r_d < r_0 \\ 0 & \text{if } r_d \geq r_0 \end{cases}$$

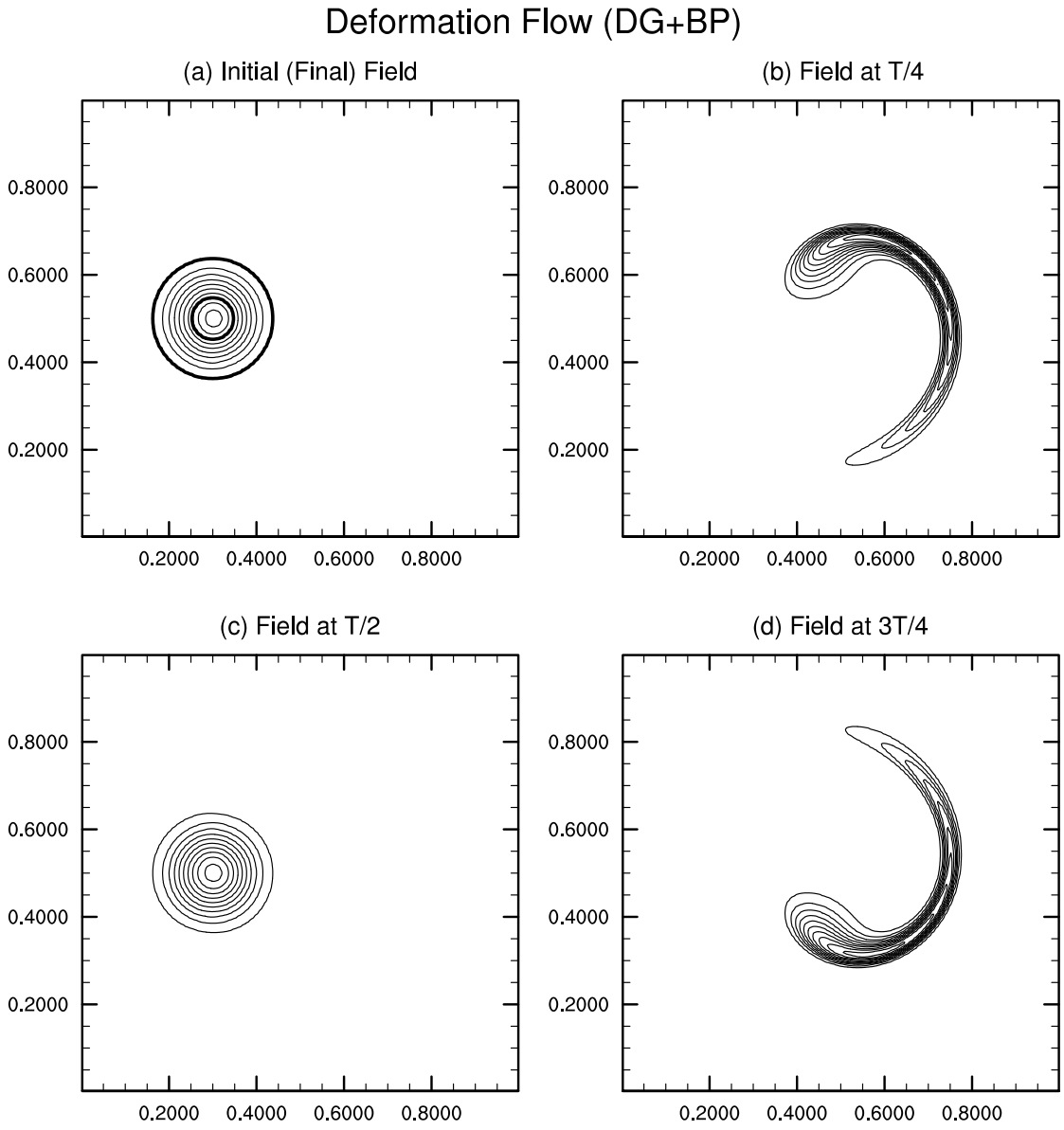


Figure 4.4.5: Results of the 2D Cartesian deformational flow test on a $[0,1]^2$ domain with $\Delta x = \Delta y = 0.01$ (maximum CFL = 0.25), using the P^2-DG scheme combined with the BP filter. Panels (b), (c), (d) and (a) show the deformation of the initial distribution during the simulation at time $T/4$, $T/2$, $3T/4$ and T, respectively. The contours are plotted in the range from -0.05 to 0.95 with increments of 0.1. Thick contours are the highlighted exact (initial) solution for the contour values 0.05 and 0.75.

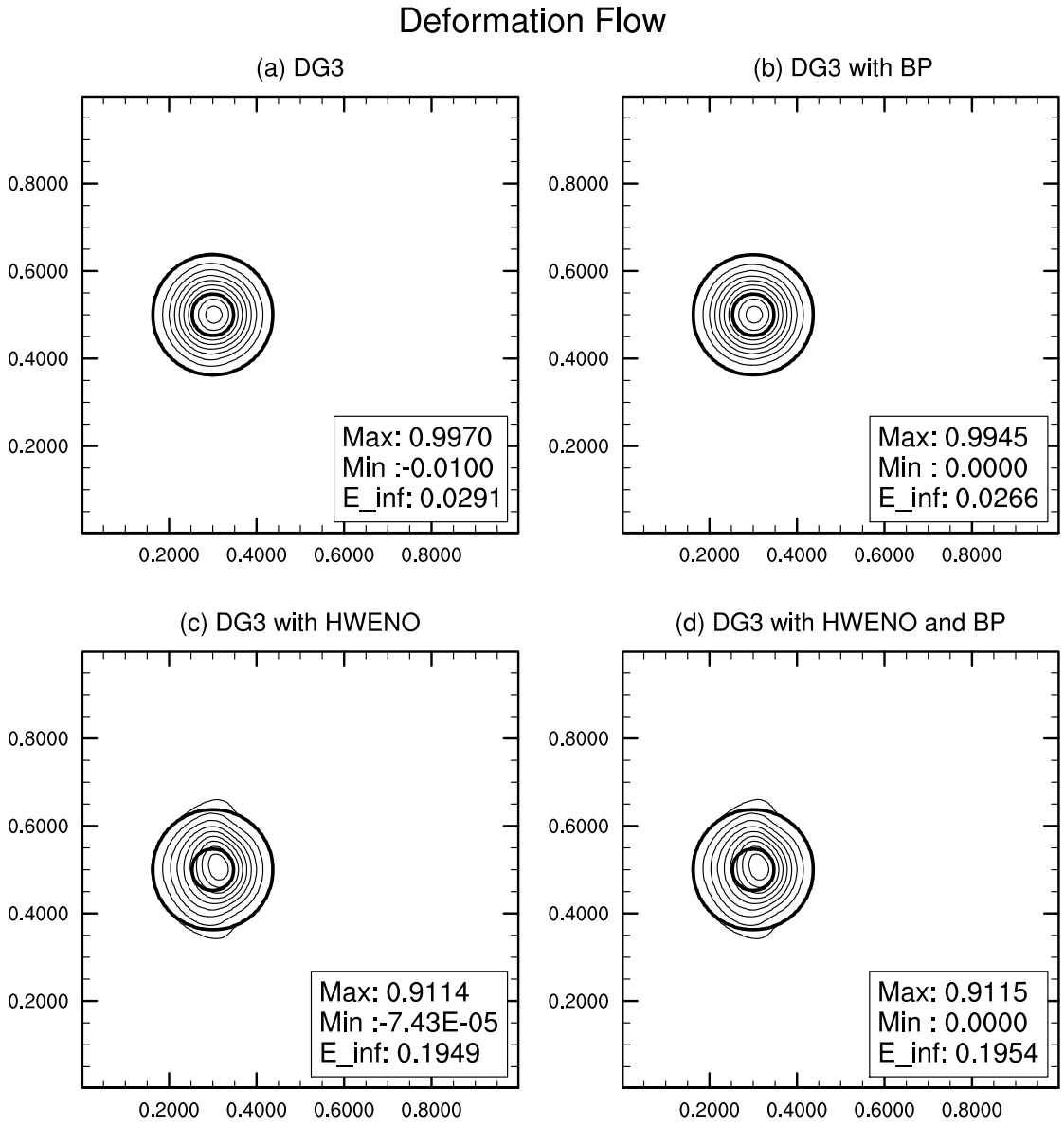


Figure 4.4.6: Same as in the Fig. 4.4.5, but the numerical solution shown at the final time ($t = T$) with various combinations of the filter and limiter. Panel (a) P^2-DG scheme only, (b) with the BP filter, (c) with the H-WENO limiter, (d) with the H-WENO limiter and the BP filter. The contours are plotted from -0.05 to 0.95 with the spacing of 0.1, thick contours are the highlighted exact (reference) solution for the contour values 0.05 and 0.75.

Where $h_0 = 1000m$ is the maximum height, r_d is the great-circle distance from (λ, θ) to the center of the cosine bell which is initially placed at $(3\pi/2, 0)$, and $r_0 = R/3$ is the radius of the cosine-bell with $R = 6.37122 \times 10^6$ denoting the earth's radius. The wind field is non-divergent and defined to be:

$$u = u_0(\cos\alpha \cos\theta + \sin\alpha \cos\lambda \sin\theta) \tag{4.4.2}$$

$$v = -u_0 \sin\alpha \sin\lambda \tag{4.4.3}$$

where $u_0 = 2\pi R/12$ days, so that it takes 12 days to complete a full rotation. The orientation of the wind field can be controlled by setting the parameter α. In the following computation, α is set to be $\pi/4$, so the cosine-bell goes through four vertices and all six faces. This configuration is the most challenging case for the cubed-sphere geometry. The numerical solution is computed on a $32 \times 32 \times 6$ mesh with a relatively small time step $\Delta t = 600$ s, which corresponds to an approximate CFL number 0.02.

In Table. 4.4.1, we give normalized standard error norms at $t = T = 12$ days. Normalized standard errors measures are comparable to those seen in recent high-order FV models [7, 62, 32]. Figure 4.4.7 shows the contour plots of the numerical solution after one full rotation. From the results, it is clear that the non-oscillatory scheme eliminates the negative values produced by the DG scheme near the foot of the bell.

Solid-Body Rotation: Multiscale Signal

Although the cosine-bell advection test is widely popular, it is often considered to be an easy test, especially for checking the monotonicity of the numerical scheme, because, the scalar distribution (cosine-bell) is quasi-smooth and covers only about

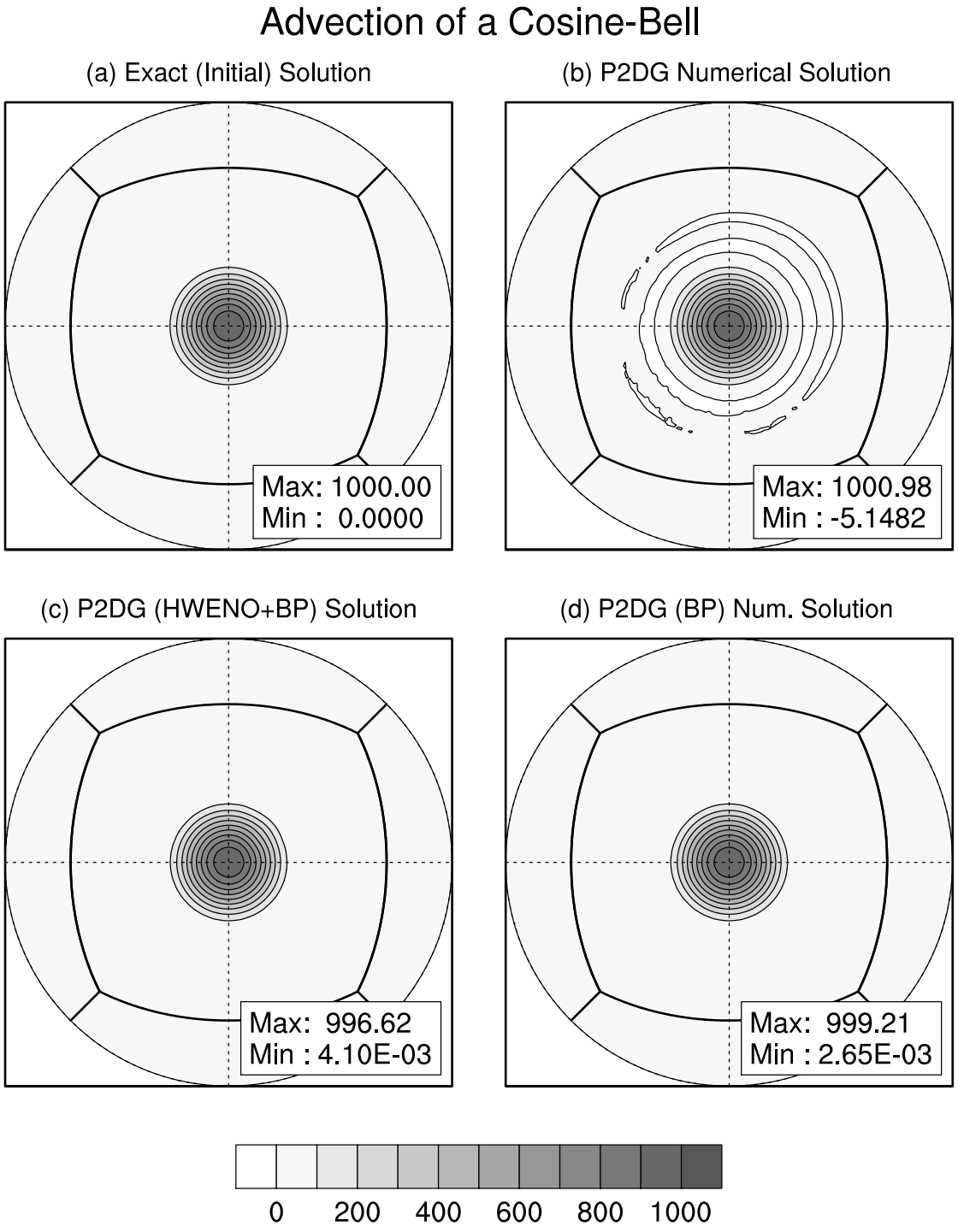


Figure 4.4.7: An orthographic projection of the solution of the solid-body rotation of a cosine-bell on the cubed-sphere. The wind fields are oriented along the north-east direction ($\alpha = \pi/4$), and 12 days are required for a full revolution. (a) Initial height of the cosine-bell which ranges from 0 to 1000 units, (b) P^2-DG solution where spurious undershoots can be seen, (c) numerical solution with the H-WENO limiter and the BP filter, (d) numerical solution with the BP filter. The cubed-sphere with a mesh $32 \times 32 \times 6$, and a time step $\Delta t = 600$ s are used for the simulation.

Table 4.4.1: Normalized standard errors for ψ, for the solid-body (cosine-bell) rotation test after a full revolution. The third-order P^2-DG transport scheme with different limiter (filter) combinations are used for the test. The flow orientation is along the north-east direction ($\alpha = \pi/4$), on a $32\times32\times6$ cubed-sphere mesh with time step $\Delta t = 600$ s. Minimum and maximum heights of the cosine-bell (ψ) after one revolution are indicated by 'Min ht.' and 'Max ht.', respectively.

Scheme	l_1	l_2	l_∞	Min ht.	Max ht.
DG	$9.75E{-}03$	$6.47E{-}03$	$5.88E{-}03$	-5.1482	1000.9833
DG + BP	$8.11E{-}03$	$5.59E{-}03$	$9.49E{-}03$	$2.65E{-}03$	999.2054
DG + HWENO	$1.22E{-}02$	$8.44E{-}03$	$1.38E{-}02$	-5.9211	1000.4346
DG + HWENO +BP	$9.49E{-}03$	$6.87E{-}03$	$1.32E{-}02$	$4.10E{-}03$	996.6210

10% of the entire spherical domain. We consider a more challenging initial condition introduced in [49], where the scalar field is a multiscale signal comprised of continuous and discontinuous functions. The velocity field is the same as in the cosine-bell test (4.4.2)-(4.4.3), but the flow is oriented along the equatorial direction ($\alpha = 0$). The initial scalar field is defined as follows:

$$\psi(\lambda,\theta) = \cos^4\theta\left\{2 + [f_1(\lambda) + f_2(\lambda)]\left[1 + 0.3\,\sin\left(\frac{50\lambda}{9}\right)\right]\left[1 + 0.4\,\sin\left(\frac{50\lambda}{10}\right)\right]\right\}, \tag{4.4.4}$$

where $f_1(\lambda) = -1$ for $\lambda \in D_1 \equiv [8\pi/25, 28\pi/25]$, $f_1(\lambda) = 0$ for $\lambda \in [0, 2\pi] - D_1$; and $f_2(\lambda) = 1$ for $\lambda \in D_2 \equiv (28\pi/25, 39\pi/25]$, $f_2(\lambda) = 0$ for $\lambda \in [0, 2\pi] - D_2$.

Figure 4.4.8a shows the initial condition for the multiscale signal (4.4.4), and Fig. 4.4.8b shows the numerical solution (after 12 days) with the P^2-DG scheme and the H-WENO limiter. The experimental setup is similar to that of the solid-body rotation test, where a mesh with $32 \times 32 \times 6$ cells and time step $\Delta t = 600$ s are used. The P^2-DG numerical solution without the H-WENO limiter is visually indistinguishable from that with the H-WENO limiter (Fig. 4.4.8b) and therefore not shown.

Table 4.4.2 lists the normalized errors for the P^2-DG scheme with or without

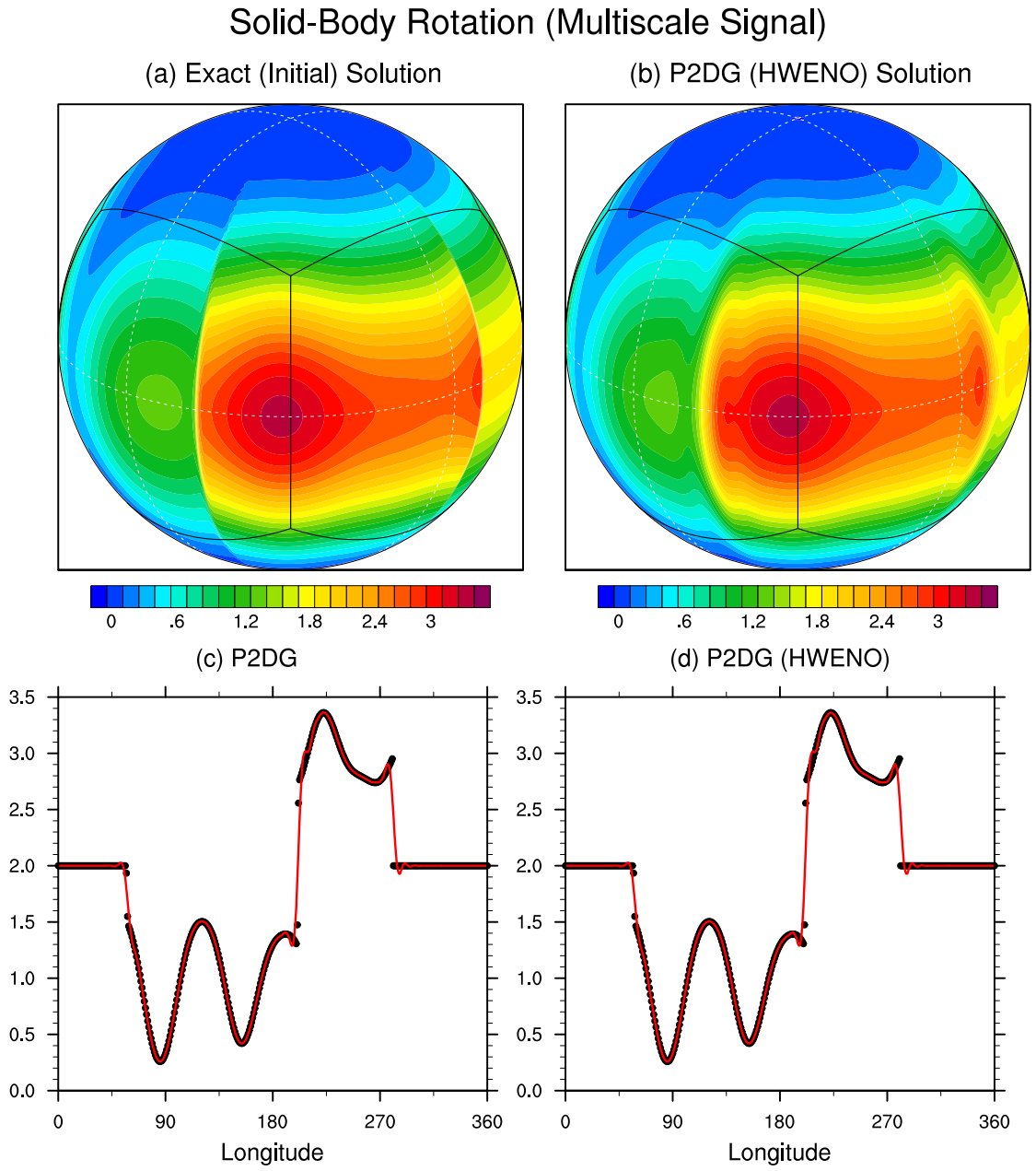


Figure 4.4.8: Solid-body rotation test of a multiscale signal comprised of continuous and discontinuous functions. (a) Exact (initial) solution, (b) numerical solution after a revolution (12 days) with P^2-DG and H-WENO; for this test the wind field is oriented along the equator ($\alpha = 0$). The lower panels show the 1-d numerical solution (solid-lines) sampled along the equator after 12 days, and the exact reference solution is marked as thick black dots. (c) P^2-DG solution and (d) solution with P^2-DG and H-WENO combination. The cubed-sphere mesh with $32 \times 32 \times 6$ cells, and a timestep $\Delta t = 1440$ s are used for the numerical simulations.

the H-WENO limiter. Figures 4.4.8c and 4.4.8d show the value of $\psi(\lambda, \theta)$ sampled along the equator (1d data) after one revolution using the P^2-DG scheme without and with the H-WENO limiter, respectively. The exact solution is indicated using black dots and the numerical solution is displayed as a solid-line. For this test, it is clear that the P^2-DG scheme itself is capable of handling the shock in the multiscale signal, and the H-WENO limiter only has a marginal impact on the solution (see, Fig. 4.4.8d and Table 4.4.2). Moreover, it is found that the BP filter has little or no influence on the solution as is evident from Table 4.4.2.

Table 4.4.2: Normalized standard errors for ψ, for the solid-body rotation test with a multiscale signal on a $32 \times 32 \times 6$ cubed-sphere mesh. The third-order P^2-DG transport scheme with H-WENO limiter and/or BP filter combinations are used for the test. The standard global errors measures are based on [67].

Scheme	l_1	l_2	l_∞	ψ_{max}	ψ_{min}
DG	0.0088	0.0318	0.1662	$6.41E{-}04$	$-1.58E{-}06$
DG + HWENO	0.0195	0.0335	0.1661	$-1.93E{-}04$	$4.71E{-}08$
DG + HWENO + BP	0.0195	0.0335	0.1661	$-1.93E{-}04$	$4.71E{-}08$

Deformational Flow on the Sphere: Vortex Problem

We consider a special case of the deformational flow test, the *moving-vortex* problem, introduced in [42]. This test consists of two steady vortices, created on a sphere with centers located at diametrically opposite sides. The flow field is non-divergent, time-dependent, and highly deformational. The vortices are designed to move along a great-circle trajectory while deforming, and the analytic solution is known at a given time. However, we use a *static* option for the vortices so that vortex-centers remain at the initial position, and numerically integrated for an extended period of time (60 days) as opposed to the recommended 12 days [42]. The purpose of this test is to check the vortex filament formation at the smallest resolvable scale

by the numerical model [50, 20].

The analytic solution at time t is given as follows,

$$\psi(\lambda', \theta', t) = 1 - \tanh\left[\frac{\rho}{\gamma_0}\sin(\lambda' - \omega(\theta')t)\right], \tag{4.4.5}$$

where (λ', θ') are the rotated spherical coordinates with respect to the regular (λ, θ) coordinates, $\rho = \rho_0 \cos\theta'$ is the radial distance of the vortex, and the parameters $\rho_0 = 3$, $\gamma_0 = 5$. For the current tests, the north pole of the rotated sphere is located at $(\lambda_c, \theta_c) = (3\pi/2, 0)$, which is also the center of one of the vortices. The angular velocity $\omega(\theta')$ is defined in terms of the tangential velocity V_t,

$$\omega(\theta') = \begin{cases} V_t/(R\,\rho) & \text{if } \rho \neq 0, \\ 0 & \text{if } \rho = 0, \end{cases}$$

and the tangential velocity of the vortex field is defined by

$$V_t = u_0 \frac{3\sqrt{3}}{2}\mathrm{sech}^2(\rho)\tanh(\rho),$$

where $u_0 = 2\pi R/(12\,\text{days})$. It is scaled such that 12 model days are required for a *full* vortex evolution, for the test recommended in [42]. The uniform wind field (u, v) is given by:

$$u = R\,\omega(\theta')\,[\sin\theta_c\cos\theta - \cos\theta_c\,\cos(\lambda - \lambda_c)\sin\theta], \tag{4.4.6}$$

$$v = R\,\omega(\theta')\,[\cos\theta_c \sin(\lambda - \lambda_c)]. \tag{4.4.7}$$

The numerical experiment is performed with a relatively high-resolution mesh employing $100 \times 100 \times 6$ cells, and with a time step $\Delta t = 600$ s.

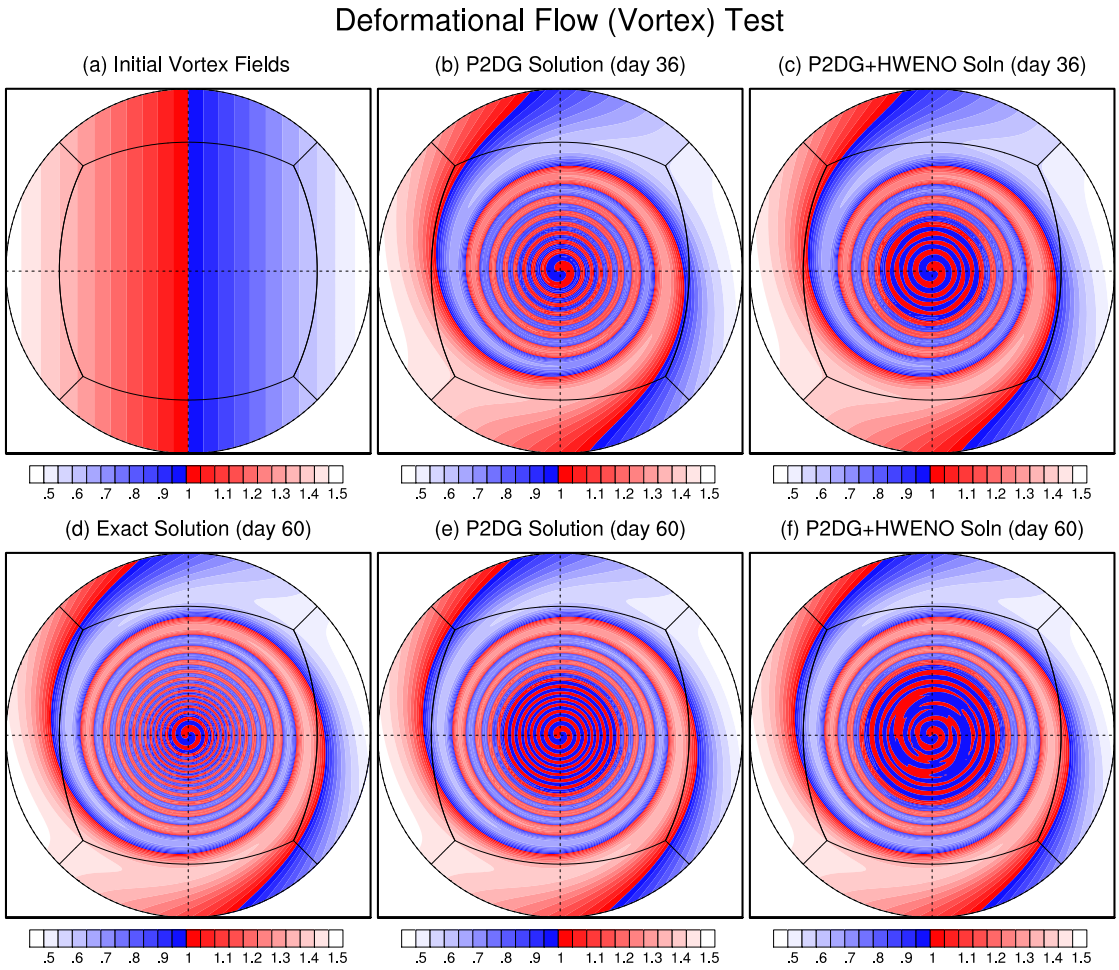


Figure 4.4.9: The solutions for the deformational flow (vortex) test on the cubed-sphere at simulated day 36 and 60. (a) Initial solution, (b) P^2-DG solution at day 36, (c) P^2-DG solution with H-WENO, (d) Exact (reference) solution at day 60, (e) P^2-DG solution at day 60 and (f) P^2-DG solution with H-WENO at day 60. Note that the numerical solutions with or without BP filter is visually indistinguishable. The cubed-sphere mesh with $100 \times 100 \times 6$ cells, and a timestep $\Delta t = 600$ s are used for the numerical simulations

Figure 4.4.9 shows the solution after 36 and 60 model days. The initial condition $\psi(\lambda', \theta', t = 0)$ is shown in Fig. 4.4.9a, and the analytic (reference) solution (4.4.5) at day 60 is shown in Fig. 4.4.9d. The numerical solution with the P^2-DG scheme and the P^2-DG combined with the H-WENO limiter at day 36 are shown in Fig. 4.4.9b and 4.4.9c, respectively, while the results at day 60 are shown in Figs. 4.4.9e and 4.4.9f, respectively. The solution with the P^2-DG and BP filter combination is visually identical to that of the P^2-DG scheme (Fig. 4.4.9b and 4.4.9e), therefore it is not shown. At this resolution the P^2-DG scheme (with or without the BP filter) preserves the fine filaments of the vortex field and its structure appears similar to the exact solution (Fig. 4.4.9d) and comparable to the numerical solution shown in Fig.10 of [50]. At day 36, the solution with P^2-DG combined with H-WENO limiter (Fig. 4.4.9c) shows minor degradation near the narrow filament walls as compared to the P^2-DG case (Fig. 4.4.9c). In Fig. 4.4.9f (at day 60), over the central regions of the vortex fields, some of the fine filament structures are broken or merged together as compared to the unlimited P^2-DG case.

Note that the deformational (vortex) test is a smooth problem for which the DG scheme does not require a limiter. This is also clear from the multiscale signal test, a quasi-smooth case, considered above. The P^2-DG scheme is local and relies only on the cell in question, and even with the application of the BP filter, its local data dependency does not change. However, when the H-WENO limiter is applied with the P^2-DG scheme, sharp gradients (thin filaments) in the solution are smoothed out due to excessive limiting. This is also due to the fact that the H-WENO limiter depends a 3×3 wide stencil. If minor oscillations are present in a cell, the limiter may be activated employing the values from the least oscillatory cells in the stencil. This often leads to flattening of legitimate sharp peaks similar to the effect of a slope limiter. A way to avoid unwanted limiting (excessive dissipation) by the H-WENO

limiter is to employ a better (stringent) criteria for identifying oscillatory (troubled) cells.

We have also computed the normalized standard l_1, l_2 and l_∞ errors after 12 model days. These values, for the P^2-DG case with and without BP filter, are virtually identical. When approximated to two decimal places, they are 6.93×10^{-6}, 3.30×10^{-5} and 8.91×10^{-4}, respectively. The corresponding values for the P^2-DG case combined with the H-WENO and BP combination are 1.31×10^{-5}, 6.81×10^{-5} and 1.61×10^{-3}, respectively. The $\psi_{\max}$ for all the cases are very close and approximately equal to 8.77×10^{-9}. The $\psi_{\min}$ for the P^2-DG case is 1.58×10^{-7} and for the other two cases it is approximately equal to 4.04×10^{-8}.

Deformational Flow on the Sphere: Slotted-Cylinder

To further validate the P^2-DG scheme on the sphere, we use a new challenging benchmark deformational flow test case proposed in [43]. We are particularly interested in two cases with non-smooth (twin slotted-cylinder) and quasi-smooth (twin cosine-bell) initial conditions. Note that this problem is specified in non-dimensional units on a unit sphere ($R=1$). The twin slotted-cylinder is defined by:

$$\psi(\lambda,\theta)=\begin{cases} c & \text{if } r_i\le r \text{ and } |\lambda-\lambda_i|\ge r/6 \text{ for } i=1,2,\\ c & \text{if } r_1\le r \text{ and } |\lambda-\lambda_1|<r/6 \text{ and } \theta-\theta_1<-\frac{5}{12}r\ ,\\ c & \text{if } r_2\le r \text{ and } |\lambda-\lambda_2|<r/6 \text{ and } \theta-\theta_2>\frac{5}{12}r\ ,\\ b & \text{otherwise,}\end{cases} \tag{4.4.8}$$

where $c = 1$, $b = 0.1$, the radius of the cylinder $r = 1/2$ and $r_i = r_i(\lambda, \theta)$ is the great-circle distance between (λ, θ) and a specified center (λ_i, θ_i)

$$r_i(\lambda, \theta) = \arccos[\sin\theta_i \sin\theta + \cos\theta_i \cos\theta \cos(\lambda - \lambda_i)].$$

The initial positions of the centers of the distributions are at $(\lambda_1, \theta_1) = (5\pi/6, 0)$ and $(\lambda_2, \theta_2) = (7\pi/6, 0)$, respectively. The slots are oriented in opposite directions for the two cylinders so that they are symmetric with respect to the flow. Figure 4.4.10 panel (a) shows the initial position of the slotted-cylinders.

For the quasi-smooth case, the slotted cylinders are replaced by two symmetrically located cosine bells, which are defined as follows:

$$\psi(\lambda, \theta) = \begin{cases} b + c\, h_1(\lambda, \theta) & \text{if } r_1 < r, \\ b + c\, h_2(\lambda, \theta) & \text{if } r_2 < r, \\ b & \text{otherwise,} \end{cases} \tag{4.4.9}$$

where $c = 0.9$, $b = 0.1$ and

$$h_i(\lambda, \theta) = \frac{h_{\max}}{2}[1 + \cos(\pi r_i / r)] \quad \text{if} \quad r_i < r, \quad \text{for } i = 1, 2.$$

Other parameters are the same as those used for the slotted-cylinder case.

The wind field is non-divergent but highly deformational. The initial distributions are deformed into thin filaments half way through the simulation while they are being transported along the zonal direction by the solid-body component of the flow. Note that an exact solution for this test is only available at the final time $t = T$, where the initial condition is recovered. The time dependent non-divergent

wind field is defined as:

$$\begin{aligned} u(\lambda,\theta,t) &= \kappa \sin^2(\lambda') \sin(2\theta) \cos(\pi t/T) + 2\pi \cos(\theta)/T \\ v(\lambda,\theta,t) &= \kappa \sin(2\lambda') \cos(\theta) \cos(\pi t/T) \end{aligned}$$

where $\lambda' = \lambda - 2\pi t/T$, $\kappa = 2.0$, and $T = 5$ units.

Figure 4.4.10 shows the results of the deformational flow tests with the P^2-DG scheme combined with different of limiter/filter options. Panel (c)-(d) in Fig. 4.4.10 shows the results for the slotted-cylinder case at the halftime ($t = T/2$) and final time ($t = T$). The P^2-DG scheme combined with the H-WENO limiter and BP filter at resolution $45 \times 45 \times 6$, captures the original shape of the slotted-cylinder as shown in Fig. 4.4.10(d). The P^2-DG scheme with the BP filter produced visibly identical results (not shown). The lower panels (e) and (f) of Fig. 4.4.10 show the simulated results with the P^2-DG and BP combination for the twin cosine-bell problem. Error norms are tabulated in Table 4.4.3.

Table 4.4.3: Normalized standard errors for ψ, for the deformational flow test with the twin slotted-cylinder (SC) and twin cosine-bell (CB) cases on a $45 \times 45 \times 6$ mesh. The third-order P^2-DG transport scheme with different limiter (filter) combinations are used for the test.

Scheme, (Tests)	l_1	l_2	l_∞	ψ_{max}	ψ_{min}
DG, (SC)	0.1498	0.2490	0.8447	0.1748	−0.1985
DG + BP, (SC)	0.1543	0.2711	0.8367	0.0000	0.0000
DG + HWENO + BP, (SC)	0.1543	0.2712	0.8361	0.0000	0.0000
DG, (CB)	0.0117	0.0226	0.0301	−0.0047	−0.0261
DG + BP, (CB)	0.0094	0.0206	0.0383	−0.0085	0.0000
DG + HWENO + BP, (CB)	0.0194	0.0505	0.1298	−0.0274	0.0000

As shown in Fig. 4.4.10, the non-oscillatory DG scheme preserves the discontinuity quite well and completely removes undesirable overshoots and undershoots. In Table. 4.4.3, the normalized standard errors are given for different combinations of

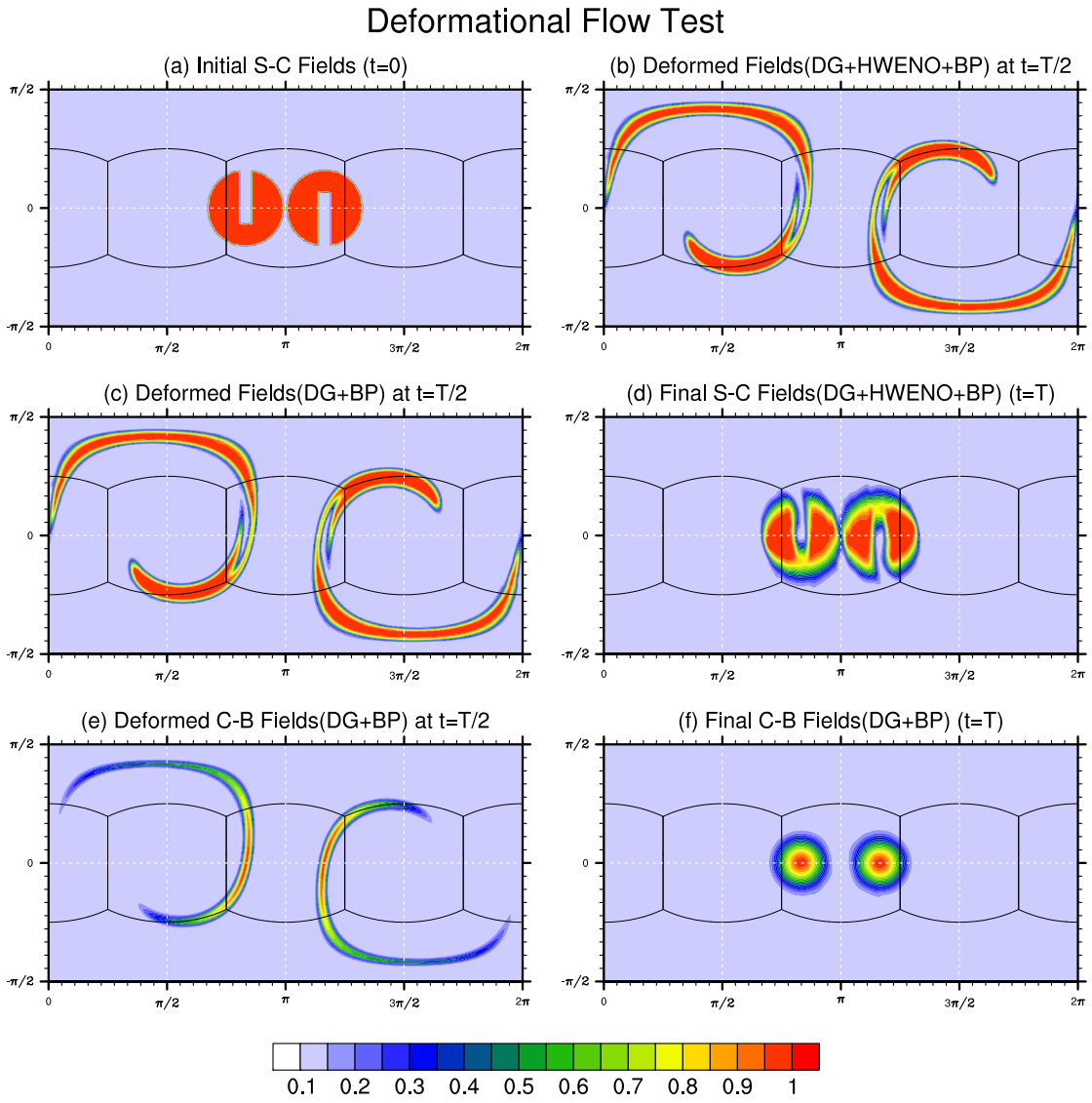


Figure 4.4.10: Numerical solution for the deformational flow test on the cubed-sphere with non-smooth (twin slotted-cylinder) initial conditions (b)-(d); and a quasi-smooth (twin cosine-bell) initial conditions (e)-(f). The initial slotted cylinders (a) move along the zonal direction while deforming, and return to the initial position after making a complete revolution. The mesh size is $45 \times 45 \times 6$ with $\Delta t = 0.00125$. Panels (b) and (c) show the P^2-DG solution at halftime ($t = T/2$) with the BP filter and H-WENO combination and BP filter, respectively. Panel (d) shows the slotted cylinders after one cycle of revolution (deformation) with the BP and H-WENO combination at the final time $T = 5$. Panels (e) and (f), respectively, show the numerical solutions with the BP filter for the twin cosine-bell case at the halftime and the final time.

the limiter (filter) for the non-smooth case. The l_1, l_2 and l_∞ errors for the non-smooth case are significantly higher than those shown for the quasi-smooth case in Table. 4.4.3. This is because the initial data for the slotted-cylinder case is severely non-smooth (C^0 discontinuous). However, for the twin cosine-bell case (Table. 4.4.3), the results compare well with those reported in the [43] paper.

We provide a rough estimate of the additional computational overhead required for the BP filter and H-WENO limiter. Both filter and limiter are applied at every stage of the RK3 time-stepping scheme. As compared with the DG (oscillatory) scheme, it is found that the DG scheme and BP filter combination takes 28% more computational time, while the DG scheme combined with the H-WENO limiter and the BP filter consumes about 40% more time. Note that the H-WENO scheme only is selectively applied to the cells which need limiting, while the BP filter applied to each cell.

We have also tested the effectiveness of the BP limiter in a nodal version of the DG scheme as described in [43] (fourth-order using 4×4 GLL points). The implementation of the BP scheme in the nodal DG version is straightforward because the solution evolves in physical (grid-point) space. It is found that the numerical solution for the non-smooth case is very similar to those shown in Fig. 4.4.10. For fifth- and higher-order DG versions (either nodal or modal), the solutions are still within the physical bounds, but increasingly contaminated by internal oscillations within the elements.

4.5 Concluding remarks

The discontinuous Galerkin (DG) methods are not inherently non-oscillatory. When there are discontinuities and sharp gradients in the solution, Gibbs phenonmenon will lead to spurious oscillations which are unacceptable for many practical applications. The main focus of this chapter is the development of a third-order DG transport scheme which is amenable to limiting and has more lenient CFL stability when used with explicit time stepping. A third-order modal version of the DG scheme (P^2-DG) with six degrees of freedom per element (cell) was developed in Cartesian geometry. In order to suppress oscillations, a limiter based on the H-WENO (Hermite Weighted Essentially Non-Oscillatory) method was applied to the P^2-DG scheme. The H-WENO limiter uses a 3×3 computational stencil with the cell, which need to be limited, located at the center.

Although it suppresses oscillations, the H-WENO limiter cannot guarantee that the legitimate physical bounds of the initial solution are maintained, and oscillations of very small amplitude may still remain in the solution. To address this issue, a Bound-Preserving (BP) filter was combined with the H-WENO limiter. The BP filter is local to the element and computationally efficient. This option provides strict positivity-preservation for the P^2-DG scheme. An explicit third-order Runge-Kutta (RK3) method was adopted for time integration.

A variety of benchmark tests were used to validate the resulting non-oscillatory DG scheme. The H-WENO limiter and the BP filter were optionally applied to remove the non-physical oscillations and to keep the numerical solution within the physical bounds. The non-oscillatory P^2-DG scheme was extended to the spherical (cubed-sphere) geometry. The standard relative errors, and the global maximum

and minimum, show that there is a substantial improvement in the quality of the solution of the non-oscillatory scheme as compared to the solution obtained by the regular DG scheme.

Extending the P^2-DG transport scheme to the 3D case is straightforward. Nevertheless, a dimension-split approach might save computational expenses significantly and ease implementation of the limiter. A better limiting criteria, other than the TVB shock detection method currently used for the H-WENO scheme, might also improve the H-WENO solution further.

Chapter Five

Conclusion

This dissertation is based on the work in [82, 81, 80]. We studied discontinuous Galerkin methods for general nonlinear convection-diffusion equations. Motivated by the general framework for the purely convection case proposed in [75, 79], we designed a second order accurate maximum-principle-satisfying DG scheme for convection-diffusion equations.

It is well known that the discrete maximum principle, is highly desirable for numerical algorithms especially if the quantities of interests are physical quantities like density or temperatures when negative values are simply non-physical. Furthermore, for nonlinear degenerate parabolic equations, negative numerical values may lead to ill-posedness and numerical instability. In the first part of this dissertation, we successfully proved the proposed P1-DG method would satisfy the strict maximum principle while maintaining uniform second order accuracy. The key ingredient for the algorithm is that for simple forward Euler time marching, under suitable CFL conditions which turns out to be only slightly more restrictive than the time step restrictions required for linear stability, the numerical cell averages of the P1-DG scheme can be represented as convex combinations of certain point values. Thus as long as a simple scaling limiter is utilized to ensure that the current point values stay within the physical bounds, the next step cell averages also stay within the bounds. Generalization to explicit high order SSP time integration is immediate.

We have also shown that the same result works on both structured or unstructured triangular meshes without imposing any restrictions on the spatial triangulation. The methodology can be extended to two dimensional incompressible Navier-Stokes equations in the vorticity / streamfunction formulation. It has also been shown that the result is valid for two-dimensional porous medium equations and the vortex patch problems.

In the second part of this dissertation, following ideas of a special finite element method [29] and a multi-scale DG method [65], we propose a multi-scale DG method for two-dimensional second order elliptic equations with curvilinear unidirectional rough coefficients. Applications arise in engineering problems where the coefficients represent the properties of the composite materials. The lack of sufficient regularity in the true solution with respect to ϵ makes most conventional numerical methods such as finite element or finite difference impractical for this type of problem. By exploiting the flexibility of DG methods in choosing local approximation basis, and utilizing the idea of incorporating the explicit oscillatory basis functions indicated by the differential operators into the approximation space, we successfully build a multi-scale DG method which is proven to yield optimal error estimates for the second order approximation space on coarse meshes. Moreover, the convergence rate has been shown to be uniform and independent of the small scale ϵ.

An easy implementation is also proposed. Numerical examples are provided to validate this and to demonstrate the accuracy and the efficiency of the multi-scale DG method.

Finally, the application of a DG method to atmospheric research is addressed in the last part of the dissertation. Combined with an H-WENO limiter and a BP filter, a P^2-DG method produces high-quality, non-oscillatory and positivity-preserving numerical approximations, ideal for the purpose of atmospheric modeling. The accuracy, the non-oscillatory and positivity-preserving properties of the P^2-DG method have been shown through a variety of benchmark tests, both on the Cartesian domain and on the cubed-sphere domain.

To summarize, in this dissertation, we focus on designing DG schemes for general convection-diffusion equations concerning important aspects of the true solution.

One is the positivity-preserving property which can be ensured by a post-process technique under suitable time-step restrictions. The other is the multi-scale resolution, which can be achieved with reasonable computational cost by utilizing the freedom of the DG schemes in choosing local approximation spaces.

Appendix A

HWENO Reconstruction

In Chapter 4, we discussed the P^2-DG discretization for each cell $I_{i,j} = I_{\ell(i,j)}$, where we considered local coordinates (ξ, η) on I_ℓ, and the orthogonal basis set $\mathcal{B}$. For the H-WENO reconstruction on each stencil S_n, $n = 1, 2, \ldots, 8$, as shown in Fig.4.2.2, the polynomials $P_n(\xi, \eta)$ given in (4.2.8) need to be modified. For the P^2-DG scheme, a fourth-order H-WENO reconstruction is required. The reconstructed Hermite quadratic polynomials on each small stencil are derived as follows using the integral constraints associated with the smoothness indicators (4.2.9) and (4.2.10).

- On stencil S_1, we require $P_1(x, y)$ to satisfy:

$$\int_{I_\ell} P_1(x,y)dxdy = a_0\, U_\ell^{0,0}, \quad \ell = 1, 2, 4, 5$$

$$\int_{I_4} P_1(x,y)\varphi_4^{1,0}dxdy = a_1\, U_4^{1,0}, \quad \int_{I_2} P_1(x,y)\varphi_2^{0,1}dxdy = a_2\, U_2^{0,1}$$

  Using the above six constraints P_1 is reconstructed as,

$$\begin{aligned} P_1(\xi,\eta) &= U_5^{0,0} + (-U_4^{0,0} + U_5^{0,0} - U_4^{1,0})\xi + (-U_2^{0,0} + U_5^{0,0} - U_2^{0,1})\eta \\ &+ \frac{(U_1^{0,0} - U_2^{0,0} - U_4^{0,0} + U_5^{0,0})}{4}\xi\eta + \frac{(-U_4^{0,0} + U_5^{0,0} - 2U_4^{1,0})}{6}(\frac{3\xi^2 - 1}{2}) \\ &+ \frac{(-U_2^{0,0} + U_5^{0,0} - 2U_2^{0,1})}{6}(\frac{3\eta^2 - 1}{2}). \end{aligned}$$

- On stencil S_2, we require $P_2(x, y)$ to satisfy:

$$\int_{I_\ell} P_2(x,y)dxdy = a_0 U_\ell^{0,0}, \quad \ell = 2, 3, 5, 6$$

$$\int_{I_6} P_2(x,y)\varphi_6^{1,0}dxdy = a_1 U_6^{1,0}, \quad \int_{I_2} P_2(x,y)\varphi_2^{0,1}dxdy = a_2 U_2^{0,1}$$

$$P_2(\xi,\eta) = U_5^{0,0} + (-U_5^{0,0} + U_6^{0,0} - U_6^{1,0})\xi + (-U_2^{0,0} + U_5^{0,0} - U_2^{0,1})\eta$$

$$+\frac{(U_2^{0,0}-U_3^{0,0}-U_5^{0,0}+U_6^{0,0})}{4}\xi\eta+\frac{(U_5^{0,0}-U_6^{0,0}+2U_6^{1,0})}{6}(\frac{3\xi^2-1}{2})$$
$$+\frac{(-U_2^{0,0}+U_5^{0,0}-2U_2^{0,1})}{6}(\frac{3\eta^2-1}{2}).$$

- On stencil S_3, we require $P_3(x,y)$ to satisfy:

$$\int_{I_\ell} P_3(x,y)dxdy = a_0U_\ell^{0,0}, \quad \ell = 4,5,7,8$$

$$\int_{I_4} P_3(x,y)\varphi_4^{1,0}dxdy = a_1U_4^{1,0}, \quad \int_{I_8} P_3(x,y)\varphi_8^{0,1}dxdy = a_2U_8^{0,1}$$

$$P_3(\xi,\eta) = U_5^{0,0}+(-U_4^{0,0}+U_5^{0,0}-U_4^{1,0})\xi+(-U_5^{0,0}+U_8^{0,0}-U_8^{0,1}))\eta$$
$$+\frac{(U_4^{0,0}-U_5^{0,0}-U_7^{0,0}+U_8^{0,0})}{4}\xi\eta+\frac{(-U_4^{0,0}+U_5^{0,0}-2U_4^{1,0})}{6}(\frac{3\xi^2-1}{2})$$
$$+\frac{(U_5^{0,0}-U_8^{0,0}+2U_8^{0,1})}{6}(\frac{3\eta^2-1}{2}).$$

- On stencil S_4, we require $P_4(x,y)$ to satisfy:

$$\int_{I_\ell} P_4(x,y)dxdy = a_0U_\ell^{0,0}, \quad \ell = 5,6,8,9$$

$$\int_{I_6} P_4(x,y)\varphi_6^{1,0}dxdy = a_1U_6^{1,0}, \quad \int_{I_8} P_4(x,y)\varphi_8^{0,1}dxdy = a_2U_8^{0,1}$$

$$P_4(\xi,\eta) = U_5^{0,0}+(-U_5^{0,0}+U_6^{0,0}-U_6^{1,0})\xi+(-U_5^{0,0}+U_8^{0,0}-U_8^{0,1})\eta$$
$$+\frac{(U_5^{0,0}-U_6^{0,0}-U_8^{0,0}+U_9^{0,0})}{4}\xi\eta+\frac{(U_5^{0,0}-U_6^{0,0}+2U_6^{1,0})}{6}(\frac{3\xi^2-1}{2})$$
$$+\frac{(U_5^{0,0}-U_8^{0,0}+2U_8^{0,1})}{6}(\frac{3\eta^2-1}{2}).$$

- On stencil S_5, we require $P_5(x,y)$ to satisfy:

$$\int_{I_\ell} P_5(x,y)dxdy = a_0 U_\ell^{0,0}, \quad \ell = 1,2,3,4,5,7$$

$$\begin{aligned} P_5(\xi,\eta) = U_5^{0,0} &+ \frac{(U_1^{0,0} - 2U_2^{0,0} + U_3^{0,0} - 2U_4^{0,0} + 2U_5^{0,0})}{4}\xi \\ &+ \frac{(U_1^{0,0} - 2U_2^{0,0} - 2U_4^{0,0} + 2U_5^{0,0} + U_7^{0,0}))}{4}\eta \\ &+ \frac{(U_1^{0,0} - U_2^{0,0} - U_4^{0,0} + U_5^{0,0})}{4}\xi\eta \\ &+ \frac{(U_1^{0,0} - 2U_2^{0,0} + U_3^{0,0})}{12}(\frac{3\xi^2 - 1}{2}). + \frac{(U_1^{0,0} - 2U_4^{0,0} + U_7^{0,0})}{12}(\frac{3\eta^2 - 1}{2}) \end{aligned}$$

- On stencil S_6, we require $P_6(x,y)$ to satisfy:

$$\int_{I_\ell} P_6(x,y)dxdy = a_0 U_\ell^{0,0}, \quad \ell = 1,2,3,5,6,9$$

$$\begin{aligned} P_6(\xi,\eta) = U_5^{0,0} &+ \frac{(-U_1^{0,0} + 2U_2^{0,0} - U_3^{0,0} - 2U_5^{0,0} + 2U_6^{0,0})}{4}\xi \\ &+ \frac{(-2U_2^{0,0} + U_3^{0,0} + 2U_5^{0,0} - 2U_6^{0,0} + U_9^{0,0})}{4}\eta + \\ &\frac{(U_2^{0,0} - U_3^{0,0} - U_5^{0,0} + U_6^{0,0})}{4}\xi\eta \\ &+ \frac{(U_1^{0,0} - 2U_2^{0,0} + U_3^{0,0})}{12}(\frac{3\xi^2 - 1}{2}) + \frac{(U_3^{0,0} - 2U_6^{0,0} + U_9^{0,0})}{12}(\frac{3\eta^2 - 1}{2}). \end{aligned}$$

- On stencil S_7, we require $P_7(x,y)$ to satisfy:

$$\int_{I_\ell} P_7(x,y)dxdy = a_0 U_\ell^{0,0}, \quad \ell = 1,4,5,7,8,9$$

$$P_7(\xi,\eta) = U_5^{0,0} + \frac{(-2U_4^{0,0} + 2U_5^{0,0} + U_7^{0,0} - 2U_8^{0,0} + U_9^{0,0})}{4}\xi$$

$$+\frac{(-U_1^{0,0}+2U_4^{0,0}-2U_5^{0,0}-U_7^{0,0}+2U_8^{0,0})}{4}\eta$$
$$+\frac{(U_4^{0,0}-U_5^{0,0}-U_7^{0,0}+U_8^{0,0})}{4}\xi\eta$$
$$+\frac{(U_7^{0,0}-2U_8^{0,0}+U_9^{0,0}))}{12}(\frac{3\xi^2-1}{2})+\frac{(U_1^{0,0}-2U_4^{0,0}+U_7^{0,0})}{12}(\frac{3\eta^2-1}{2}).$$

- On stencil S_8, we require $P_8(x,y)$ to satisfy:

$$\int_{I_\ell} P_8(x,y)dxdy = a_0 U_\ell^{0,0}, \quad \ell = 3,5,6,7,8,9$$

$$P_8(x,y) = U_5^{0,0}+\frac{(-2U_5^{0,0}+2U_6^{0,0}-U_7^{0,0}+2U_8^{0,0}-U_9^{0,0})}{4}\xi$$
$$+\frac{(-U_3^{0,0}-2U_5^{0,0}+2U_6^{0,0}+2U_8^{0,0}-U_9^{0,0})}{4}\eta$$
$$+\frac{(U_5^{0,0}-U_6^{0,0}-U_8^{0,0}+U_9^{0,0})}{4}\xi\eta$$
$$+\frac{(U_7^{0,0}-2U_8^{0,0}+U_9^{0,0})}{12}(\frac{3\xi^2-1}{2})+\frac{(U_3^{0,0}-2U_6^{0,0}+U_9^{0,0})}{12}(\frac{3\eta^2-1}{2}).$$

Appendix B

Smooth Indicator

Here we give the explicit formulation of the smoothness indicators (β_n) in Eq.(4.2.9)-(4.2.10) associated with the higher-order moments of the P^2-DG for each reconstructed polynomial $P_n(x, y)$. For notational convenience, we assume $P_n(x, y)$ is expressed in the general form:

$$P_n(\xi, \eta) = U_n^{0,0} + U_n^{1,0}\xi + U_n^{0,1}\eta + U_n^{1,1}\xi\eta + U_n^{2,0}\frac{3\xi^2 - 1}{2} + U_n^{0,2}\frac{3\eta^2 - 1}{2}.$$

Combined with the coefficients given in Appendix A, one can compute the smoothness indicators from the formulas listed below.

- For the mode $U_{i,j}^{1,0}$, we require:

$$\beta_n = \sum_{m=1}^{2} |I_{i,j}|^{m-1} \int_{I_{i,j}} \left(\frac{\partial^m}{\partial x^m} P_n(x, y)\right)^2 dx\, dy.$$

  And this gives for each $P_n(x, y)$,

$$\beta_n = 4(U_n^{1,0})^2 + \frac{4}{3}(U_n^{1,1})^2 + 156(U_n^{2,0})^2.$$

- For the mode $U_{i,j}^{0,1}$, we require:

$$\beta_n = \sum_{m=1}^{2} |I_{i,j}|^{m-1} \int_{I_{i,j}} \left(\frac{\partial^m}{\partial y^m} P_n(x, y)\right)^2 dx\, dy.$$

  For each $P_n(x, y)$,

$$\beta_n = 4(U_n^{0,1})^2 + \frac{4}{3}(U_n^{1,1})^2 + 156(U_n^{0,2})^2.$$

- For the higher order modes: $U_{i,j}^{1,1}, U_{i,j}^{2,0}, U_{i,j}^{0,2}$, we require:

$$\beta_n = \sum_{|m|=2}^{2} |I_{i,j}|^{m-1} \int_{I_{i,j}} \left(\frac{\partial^{|m|}}{\partial x^{m_1} \partial y^{m_2}} P_n(x,y) \right)^2 dx\, dy.$$

  For each $P_n(x,y)$,

$$\beta_n = 16((U_n^{1,1})^2 + 9(U_n^{2,0})^2 + 9(U_n^{0,2})^2).$$

Bibliography

[1] D. Arnold. An interior penalty finite element method with discontinuous elements. *SIAM Journal on Numerical Analysis*, 19:742–760, 1982.

[2] D. Arnold, F. Brezzi, B. Cockburn, and L. Marini. Unified analysis of discontinuous galerkin for elliptic problems. *SIAM J. Numer. Anal.*, 39:1749–1779, 2002.

[3] I. Babuška and J. Osborn. Generalized finite element methods: their performance and their relation to mixed methods. *SIAM J. Numer. Anal*, 20:510–536, 1983.

[4] D. S. Balsara, C. Altman, C.-D. Munz, and M. Dumbser. Sub-cell based indicator for troubled zones in RKDG schemes and a novel class of hybrid RKDG+HWENO schemes. *Journal of Computational Physics*, 226:586–620, 2007.

[5] C. E. Baumann and J. T. Oden. A discontinuous hp finite element method for convection-diffusion problems. *Computer Methods in Applied Mechanics and Engineering*, 175:311–341, 1999.

[6] P. N. Blossey and D. R. Durran. Selective monotonicity preservation in scalar advection. *Journal of Computational Physics*, 227:5160–5183, 2008.

[7] C. Chen and F. Xiao. Shallow water model on cubed-sphere by multi-moment finite volume method. *Journal of Computational Physics*, 227:5019–5044, 2008.

[8] Z. Chen and T.Y. Hou. A mixed multiscale finite element method for elliptic problems with oscillating coefficients. *Math. Comp.*, 72:541–576, 2002.

[9] Y. Cheng, I.M. Gamba, and J. Proft. Positivity-preserving discontinuous galerkin schemes for linear vlasov-boltzmann transport equations. *Mathematics of Computation*, 81:153–190, 2012.

[10] Y. Cheng and C.-W. Shu. A discontinuous galerkin finite element method for time dependent partial differential equations with higher order derivatives. *Mathematics of Computation*, 77:699–730, 2008.

[11] B. Cockburn. An introduction to the Discontinuous-Galerkin method for convection-dominated problems. In A. Quarteroni, editor, *Lecture Notes in Mathematics: Advanced Numerical Approximation of Nonlinear Hyperbolic Equations*, volume 1697, pages 151–268. Springer, 1997.

[12] B. Cockburn, S. Hou, and C.-W. Shu. The runge-kutta local projection discontinuous galerkin finite element method for conservation laws iv: the multidimensional case. *Mathematics of Computation*, 54:545–581, 1990.

[13] B. Cockburn and C.-W. Shu. TVB Runge-Kutta local projection Discontinuous Galerkin finite element method for conservation laws II: general framework. *Mathematics of Computation*, 52:411–435, 1989.

[14] B. Cockburn and C.-W. Shu. The local discontinuous galerkin method for time-dependent convection-diffusion systems. *SIAM Journal on Numerical Analysis*, 35:2440–2463, 1998.

[15] B. Cockburn and C.-W. Shu. Runge-kutta discontinuous galerkin methods for convection-dominated problems. *Journal of Scientific Computing*, 16:173–261, 2001.

[16] Y. R. Efendiev, T. Y. Hou, and X. H. Wu. The convergence of nonconforming multi-scale finite element methods. *SIAM J. Numer. Anal.*, 37:888–910, 2000.

[17] R.S. Falk and J.E. Osborn. Remarks on mixed finite element methods for problems with rough coefficient. *Math. Comp.*, 62:1–19, 1994.

[18] I. Faragó and R. Horváth. Discrete maximum principle and adequate discretizations of linear parabolic problems. *SIAM Journal on Scientific Computing*, 28:2313–2336, 2006.

[19] I. Faragó, J. Karátson, and S. Korotov. Discrete maximum principles for nonlinear parabolic pde systems. *IMA Journal of Numerical Analysis, to appear*. DOI:10.1093/imanum/drr050.

[20] N. Flyer and G. B. Wright. Transport schemes on a sphere using radial basis functions. *Journal of Computational Physics*, 226:1059–1084, 2007.

[21] H. Fujii. Theory and practice in finite element structural analysis. *University of Tokyo Press, Tokyo*, pages 91–106, 1973.

[22] A. Genty and C. Le Potier. Maximum and minimum principles for radionuclide transport calculations in geological radioactive waste repository: Comparison between a mixed hybrid finite element method and finite volume element discretizations. *Transport in Porous Media*, 88:65–85, 2011.

[23] F.X. Giraldo, J.S. Hesthaven, and T. Warburton. Nodal high-order discontinuous Galerkin methods for spherical shallow water equations. *Journal of Computational Physics*, 181:499–525, 2002.

[24] K. N. Godev, R. D. Lazarov, V. L. Makarov, and A. A. Samarskii. Homogeneous difference schemes for one-dimensional problems with generalized solutions. *Math USSR SB*, 59:155–179, 1988.

[25] S. Gottlieb, D.I. Ketcheson, and C.-W. Shu. High order strong stability preserving time discretizations. *Journal of Scientific Computing*, 38:251–289, 2009.

[26] S. Gottlieb, C.-W. Shu, and E. Tadmor. Strong Stability-Preserving high-order time discretization methods. *SIAM Review*, 43(1):89–112, 2001.

[27] T. Y. Hou and X. H. Wu. A multiscale finite element method for elliptic problems in composite materials and porous media. *J. Comput. Phys.*, 134:169–189, 1997.

[28] T. Y. Hou, X. H. Wu, and Z. Cai. Convergence of a multiscale finite element method for elliptic problems with rapidly oscillating coefficients. *Math. Comp.*, 68:913–943, 1999.

[29] G. Caloz I. Babuška and J. Osborn. Special finite element methods for a class of second order elliptic problems with rough coefficients. *SIAM J. Numer. Anal*, 31:945–981, 1994.

[30] M. Iskandarani, J.C. Levin, B.-J. Choi, and D.B. Haidgovel. Comparison of advection schemes for high-order h-p finite element and finite volume methods. *Ocean Modeling*, 10:233–252, 2005.

[31] J. Douglas Jr. and T. Dupont. Interior penalty procedures for elliptic and parabolic galerkin methods. In *Lecture Notes in Phys.*, volume 58. Springer-Verlag, Berlin, 1976.

[32] K. K. Katta, R. D. Nair, and V. Kumar. High-order central finite-volume schemes for global atmopsheric models. *Quarterly Journal of the Royal Meteorological Society*, 2011. Submitted.

[33] L. Krivodonova. Limiters for high-order discontinuous Galerkin methods. *Journal of Computational Physics*, 226:879–896, 2007.

[34] A. Kurganov and E. Tadmor. New high-resolution central schemes for nonlinear conservation laws and convection-diffusion equations. *Journal of Computational Physics*, 160:241–282, 2002.

[35] P. Langlo and M. S. Espedal. Macrodispersion for two-phase, immiscible flow in porous media. *Adv. in Water Resources*, 17:297–316, 1994.

[36] M. Läuter, F. X. Giraldo, D. Handorf, and K. Dethloff. A discontinuous Galerkin method for shallow water equations in spherical triangular coordinates. *Journal of Computational Physics*, 227:10226–10242, 2008.

[37] R. J. LeVeque. *Finite Volume Methods for Hyperbolic Problems*. Cambridge University Press, 2002. ISBN 0-19-00924-3, 558 pp.

[38] M. N. Levy, R. D. Nair, and H. M. Tufo. High-order Galerkin method for scalable global atmopsheric models. *Computers and Geoscience*, 33:1022–1035, 2007.

[39] J.-G. Liu and C.-W. Shu. A high-order discontinuous galerkin method for 2d incompressible flows. *Journal of Computational Physics*, 160:577–596, 2000.

[40] X.-D. Liu and S. Osher. Nonoscillatory high-order accurate self similar maximum-principle-satisfying shock capturing scheme. *SIAM Journal on Numerical Analysis*, 33:760–779, 1996.

[41] H. Luo, J. D. Baum, and R. Lohner. A Hermite WENO-based limiter for discontinuous Galerkin method on unstructured grids. *Journal of Computational Physics*, 225:686–713, 2007.

[42] R. D. Nair and C. Jablonowski. Moving vortices on the sphere: A test case for horizontal advection problems. *Monthly Weather Review*, 136:699–711, 2008.

[43] R. D. Nair and P. H. Lauritzen. A class of deformational flow test cases for linear transport problems on the sphere. *Journal of Computational Physics*, 229:8868–8887, 2010.

[44] R. D. Nair, S.J. Thomas, and R.D. Loft. A Discontinuous Galerkin global shallow water model. *Monthly Weather Review*, 133:876–888, 2005.

[45] R. D. Nair, S.J. Thomas, and R.D. Loft. A Discontinuous Galerkin transport scheme on the cubed sphere. *Monthly Weather Review*, 133:814–828, 2005.

[46] Ramachandran D. Nair, Michael N. Levy, and Peter Hjort Lauritzen. Emerging numerical methods for atmospheric modeling. In P.H. Lauritzen, C. Jablonowski, M. Taylor, and R.D. Nair, editors, *Numerical Techniques for Global Atmospheric Models, Lecture Notes in Computational Science and Engineering, Vol.80*, pages 251–311. Springer, 2011.

[47] B. Niceno. EasyMesh Version 1.4: a two-dimensional quality mesh generator. 2001. available from: http://www-dinma.univ.trieste.it/nirftc/research/easymesh/.

[48] B. Perthame and C.-W. Shu. On positivity preserving finite volume schemes for euler equations. *Numerische Mathematik*, 73:119–130, 1996.

[49] J. A. Pudykiewicz. Numerical solution of the reaction-advection-diffusion equation on the sphere. *Journal of Computational Physics*, 213:358–390, 2006.

[50] J. A. Pudykiewicz. On numerical solution of the shallow water equations with chemical reactions on icosahedral geodesic grid. *Journal of Computational Physics*, 230:1956–1991, 2011.

[51] J. Qiu and C.W. Shu. Hermite WENO schemes and their application as limiters for Runge-Kutta discontinuous Galerkin method: Two dimensional case. *Computers & Fluids*, 34:642–663, 2005.

[52] J. Qiu and C.W. Shu. Runge-Kutta discontinuous Galerkin method using WENO limiters. *SIAM Journal on Scientific Computing*, 26:907–929, 2005.

[53] J.-M. Qiu and C.-W. Shu. Positivity preserving semi-lagrangian discontinuous galerkin formulation: Theoretical analysis and application to the vlasov-poisson system. *Journal of Computational Physics*, 230:8386–8409, 2011.

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

[55] B. Riviere, M.F. Wheeler, and V. Girault. Improved energy estimates for interior penalty, constrained and discontinuous galerkin methods for elliptic problems. part i. *Computational Geosciences*, 8:337–360, 1999.

[56] J.A. Rossmanith and D.C. Seal. A positivity-preserving high-order semi-lagrangian discontinuous galerkin scheme for the vlasov-poisson equations. *journal of Computational Physics*, 230:6203–6232, 2011.

[57] R. Sadourny. Conservative finite-difference approximations of the primitive equations on quasi-uniform spherical grids. *Monthly Weather Review*, 100:136–144, 1972.

[58] C.-W. Shu. Essentially non-oscillatory and weighted essentially non-oscillatory schemes for hyperbolic conservation laws. In *Lecture Notes in Mathematics*, volume 1697, pages 325–432. Springer, 1997.

[59] R. Strehl, A. Sokolov, and S. Turek. Efficient, accurate and flexible finite element solvers for chemotaxis problems. *Computers and Mathematics with Applications*, 64:175–189, 2012.

[60] V. Thomée and L.B. Wahlbin. On the existence of maximum principles in parabolic finite element equations. *Mathematics of Computation*, 77:11–19, 2008.

[61] A. N. Tikhonov and A. A. Samarskii. Homogeneous difference schemes. *USSR Computational Mathematics and Mathematical Physics*, 1:5–67, 1962.

[62] P. A. Ullrich, C. Jablonowski, and B. van Leer. High-order finite-volume methods for the shallow-water equations on the sphere. *Journal of Computational Physics*, 229:6104–6134, 2010.

[63] C. Wang, X. Zhang, C.-W. Shu, and J. Ning. Robust high order discontinuous galerkin schemes for two-dimensional gaseous detonations. *Journal of Computational Physics*, 231:653–665, 2012.

[64] W. Wang. Multiscale discontinuous galerkin methods and applications. *Ph.D. Thesis, Brown University*, 2008.

[65] W. Wang, J. Guzmán, and C.-W. Shu. The multiscale discontinuous galerkin method for solving a class of second order elliptic problems with rough coefficients. *International Journal of Numerical Analysis and Modeling*, 8:28–47, 2011.

[66] M. Wheeler. An elliptic collocation finite element method with interior penalties. *SIAM Journal on Numerical Analysis*, 15:152–161, 1978.

[67] D. L. Williamson, J. B. Drake, J. J. Hack, R. Jakob, and P. N. Swarztrauber. A standard test set for numerical approximations to the shallow water equations in spherical geometry. *Journal of Computational Physics*, 102:211–224, 1992.

[68] Y. Xing and C.-W. Shu. High-order finite volume weno schemes for the shallow water equations with dry states. *Advances in Water Resources*, 34:1026–1038, 2011.

[69] Y. Xing, X. Zhang, and C.-W. Shu. Positivity preserving high order well balanced discontinuous galerkin methods for the shallow water equations. *Advances in Water Resources*, 33:1476–1493, 2010.

[70] L. Yuan and C.-W. Shu. Discontinuous galerkin method based on non-polynomial approximation spaces. *J. Comput. Phys.*, 218:295–323, 2006.

[71] L. Yuan and C.-W. Shu. Discontinuous galerkin method for a class of elliptic multi-scale problems. *Int. J. Numer. Meth. Fluids*, 56:1017–1032, 2008.

[72] M. Zhang and C.-W. Shu. An analysis of three different formulations of the discontinuous galerkin method for diffusion equations. *Mathematical Models and Methods in Applied Sciences*, 13:395–413, 2003.

[73] R. Zhang, M. Zhang, and C.-W. Shu. High order positivity-preserving finite volume weno schemes for a hierarchical size-structured population model. *Journal of Computational and Applied Mathematics*, 236:937–949, 2011.

[74] X. Zhang, Y. Liu, and C.-W. Shu. Maximum-principle-satisfying high order finite volume weno schemes for convection-diffusion equations. *SIAM Journal on Scientific Computing*, 34:A627–A658, 2012.

[75] X. Zhang and C.-W. Shu. On maximum-principle-satisfying high order schemes for scalar conservation laws. *Journal of Computational Physics*, 229:3091–2120, 2010.

[76] X. Zhang and C.-W. Shu. Maximum-principle-satisfying and positivity-preserving high order schemes for conservation laws: Survey and new developments. *Proceedings of the Royal Society A*, 467:2752–2776, 2011.

[77] X. Zhang and C.-W. Shu. Positivity-preserving high order discontinuous galerkin schemes for compressible euler equations with source terms. *Journal of Computational Physics*, 230:1238–1248, 2011.

[78] X. Zhang and C.-W. Shu. Positivity-preserving high order finite difference weno schemes for compressible euler equations. *Journal of Computational Physics*, 231:2245–2258, 2012.

[79] X. Zhang, Y. Xia, and C.-W. Shu. Maximum-principle-satisfying and positivity-preserving high order discontinuous galerkin schemes for conservation laws on triangular meshes. *Journal of Scientific Computing*, 50:29–62, 2012.

[80] Y. Zhang and R.-D. Nair. A non-oscillatory discontinuous galerkin transport scheme on the cubed-sphere. *Monthly Weather Review*, 140:3106–3126, 2012.

[81] Y. Zhang, W. Wang, J. Guzmán, and C.-W. Shu. Multi-scale discontinuous galerkin method for solving elliptic problems with curvilinear unidirectional rough coefficients. *In preparation*, 2013.

[82] Y. Zhang, X. Zhang, and C.-W. Shu. Maximum-principle-satisfying second order discontinuous galerkin schemes for convection-diffusion equations on triangular meshes. *Journal of Computational Physics*, 234:295–316, 2013.