Superconvergence, Superaccuracy, and Stability of the Discontinuous Galerkin Finite Element Method

by

Noel Chalmers

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Doctor of Philosophy
in
Applied Mathematics

Waterloo, Ontario, Canada, 2015

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Abstract

This thesis is concerned with the investigation of the superconvergence, superaccuracy, and stability properties of the discontinuous Galerkin (DG) finite element method in one and two dimensions. We propose a novel method for the analysis of these properties. We apply the DG method to a model linear advection problem to derive a partial differential equation (PDE) which is satisfied by the numerical solution itself. This PDE is equivalent to the original advection equation but with a forcing term that is proportional to the jump in the numerical solution at the cell interfaces. We then use classical Fourier analysis to determine the solutions to this PDE with particular temporal frequencies. We find that these Fourier modes are completely determined on each cell by the inflow into that cell and a certain rational function of the mode's frequency. By using local expansions of these modes, we prove several local superconvergence properties of the DG method, as well as superaccurate errors in terms of dissipation and dispersion. Next, by considering a uniform mesh and assuming periodic boundary conditions, we investigate the spectrum of the method. In particular, we show that the spectrum can be partitioned into physical and non-physical modes. The physical modes advect with high-order accuracy while the non-physical modes decay exponentially quickly in time. Finally, using these results we establish several global superconvergence properties of the method on uniform meshes.

In one dimension, we find that the Fourier modes of the numerical solution are closely related to the $\frac{p}{p+1}$ Padé approximant of the exponential function e^z, where p is the order of polynomial approximation. We also find that the local expansion of the Fourier modes of the numerical solution are related to the $(p+1)$-th right-based Radau polynomial R^-_{p+1}. These properties enable us to give a simple new proof of the local superconvergence of the DG method, i.e. the local numerical error is superconvergent of order $p+2$ at the roots of this Radau polynomial, and order $2p+2$ at the downwind point in each cell. We also give a new straight-forward proof that the scheme obtains order $2p+1$ accuracy in dissipation and order $2p+2$ in dispersion. Finally, we prove that on a uniform computational mesh the numerical solution will globally tend towards a superconvergent form which converges at order $p+2$ at the right Radau points in each cell and order $2p+1$ at the downwind point of each cell.

In two dimensions, we establish results analogous to the one-dimensional case. We again find that the Fourier modes of the numerical solution are related to rational approximations of the exponential function e^z. We then use these modes to prove several local superconvergence properties, which depend on the flow direction on a cell. On a uniform mesh of triangles, we symbolically verify that the scheme obtains order $2p+1$ in terms of dissipation and dispersion errors. Finally, we also symbolically verify several global superconvergence properties of the numerical solution on this uniform mesh, and confirm these properties numerically.

Having established these results, we also propose a new family of schemes which can been viewed as a modified version of the DG scheme. These schemes contain $p+1$ free parameters which, a priori, can be freely chosen. By extending our analysis to these new schemes we show that the modifications will affect the formal orders of accuracy of the method in terms of dissipation and dispersion errors. We also show that the superconvergence properties of the method can be manipulated through these parameters. We then find that the size of the spectrum of the method can be effectively altered using particular choices of the parameters. We use this fact to construct schemes with significantly larger stable Courant-Friedrichs-Lewy (CFL) numbers than the classic DG method. We demonstrate through some numerical examples that these modified schemes can be effective in capturing fine structures of the numerical solution when compared with the DG scheme with equivalent computational effort.

Acknowledgements

I would like to thank my fellow students Marty Fuhry and Andrew Giuliani for all their helpful discussions, often into the late hours of the night. I would also like to thank my committee members Hans De Sterck, David Siegel, Sander Rhebergen, Robert Gracie, and Tim Warburton. Above all, I'd like to thank my supervisor Lilia Krivodonova whose guidance and encouragement have made my studies not only successful, but enjoyable.

Dedication

To my grandfather, Sidney Irwin. During my childhood, he would tell stories of his travels and experiences, and could recite innumerable poems from memory. From him, I first heard these words.

> If you can fill the unforgiving minute
> With sixty seconds worth of distance run,
> You'll be a man, my son.

(Rudyard Kipling, *If*)

Table of Contents

List of Tables

List of Figures

Chapter 1

The Discontinuous Galerkin Finite Element Method

> *The obtaining of numerical results in mathematical applications, the so-called 'number crunching' is quite often taken for granted, both in ordinary life and even in relatively more sophisticated settings in science and engineering. We blithely expect our calculators and computers to produce numerical answers, flawlessly and unambiguously. Whatever mathematics is lurking behind those calculations is hidden, obscured, invisible. This invisible mathematics is known as numerical analysis.*
>
> (Anthony Peressini)

1.1 Introduction

This thesis is concerned with the numerical analysis of an algorithm known as the discontinuous Galerkin (DG) finite element method. Introduced in 1973 by Reed and Hill [59] in the context of neutron transport, the DG method was originally developed for the numerical solution of ordinary differential equations (ODEs). LeSaint and Raviart [52] presented the first mathematical analysis of the method soon afterwards in 1974. The DG method was then developed into a form which made it suitable for computational fluid flow problems by Cockburn and Shu in a series of papers [30, 29, 27, 23]. In these works, the original DG spatial discretization was paired with explicit Runge-Kutta [18] time integration methods. Since then, the DG scheme has gained popularity and been rapidly developed into a robust method for solving linear and non-linear partial differential equations (PDEs) in computational fluid dynamics and other areas. The DG method has had applications in gas dynamics [13, 15, 11], compressible flows [12, 55, 32, 69], incompressible flows [16, 25, 24], turbulent flows [14, 10], granular flows [37], electromagnatism [41], magneto-hydrodynamics [68], KdV-type equations [71], and many more topics. A history of the development of the DG method up to 1999 can be found in [26] and the reader is referred to the review papers [31] and [22] and the references therein for a more complete discussion.

The main application of the DG method that we will be concerned with in this thesis is the numerical approximation of general hyperbolic systems of conservation laws in one space dimension

$$\mathbf{u}_t + \mathbf{f}(\mathbf{u})_x = 0$$

and in multiple spatial dimensions

$$\mathbf{u}_t + \nabla \cdot \mathbf{F}(\mathbf{u}) = 0.$$

The approximation of partial differential equations (PDEs) of this type has proven to be a difficult task primarily because their exact solutions are known to develop discontinuities in finite time, and the solutions can exhibit a complex structure near these discontinuities [51]. High-order finite difference and finite volume schemes have had considerable success in this area; in particular, methods implementing the essentially non-oscillatory (ENO) [40, 39, 62, 63] and weighted essentially non-oscillatory (WENO) [54, 44, 35, 42] reconstruction schemes. There are, however, several properties of the DG method which make it an attractive alternative. The numerical solution produced by the DG method is inherently discontinuous between computational cells, allowing for the capture of fine structures in the solution near discontinuities. The method can also achieve an arbitrarily high formal order of accuracy on smooth solutions by simply using a suitably high-order approximation on each element. Unlike finite difference schemes, the DG method can be naturally applied to unstructured computational meshes making it useful for problems with complex geometries. Finally, the method is highly parallelizable and adaptive. The numerical solution on each element only requires information from its own cell and its immediate neighbours in order to be evolved in time. Adaptive strategies can therefore be used to alter the order of approximation and the geometry of the mesh to suit the problem. These properties have made the DG method an ideal candidate for implementation on highly parallel graphical processing unit (GPU) computer architectures. The reader is referred to [45] and [36] for a discussion of the implementation of the nodal and modal forms of the DG method, respectively, on GPUs.

The DG method does, however, suffer from a significant drawback when compared to finite difference and finite volume methods. The linear stability restriction of the method, stated through its Courant-Friedrichs-Lewy (CFL) condition, scales inversely with its order of approximation. The numerical solution therefore requires significantly more time-integration steps to reach the same final time T compared to finite difference and finite volume methods. The computation of the DG spatial discretization also involves both volume integrals and integrals along cell boundaries which must make use of Riemann solvers in flux evaluations. This is in contrast to finite volume schemes, which do not require the computation of volume integrals, and finite difference schemes, which do not require the use of Riemann solvers. Finally, due to the occurrence of discontinuities in the numerical solution, and due to the inherent non-linearities in the general conservation laws, oscillations in the DG numerical solution that form near discontinuities due to the Gibbs phenomenon must be filtered using some form of limiting procedure. Although effective limiting procedures which preserve the DG scheme's local stencil and parallelism have been proposed in one dimension, namely the moment limiter [46], the creation of robust limiters in higher dimensions on unstructured meshes remains an open problem. These issues make the DG scheme a potentially more expensive method for the same theoretical order of convergence when compared to corresponding finite difference and finite volume schemes.

In this thesis we analyse several properties of the DG method and the accuracy of the numerical solutions it produces. The first such property is known as *superconvergence*. Superconvergence is the property that certain points within each computational cell can potentially exhibit a higher rate of convergence than the numerical solution as a whole. It has been conjectured that this property is what is exploited during post-processing algorithms, such as presented in [28] and further developed in [60], where a higher-order approximation is extracted from the current numerical solution. While superconvergence for classical finite element methods has been extensively studied [66, 7], such analysis has only recently been performed for DG schemes and several open questions remain, particularly for the DG scheme on non-cartesian grids. The second property of interest in this thesis is known as *superaccuracy* and refers to the high-order accuracy achieved by the scheme in terms of dissipation and dispersion errors. This topic has been studied by several authors [43, 6] in one dimension but few results exist for higher-dimensional problems. We will also investigate the severe stability restriction of the method by analysing the spectrum of the DG spatial discretization as done in [48]. While these three properties of the scheme appear at first glance to be disparate topics of study, we show that not only is there a connection between the superconvergence, superaccuracy, and stability of the method, but that these properties can be altered to produced schemes with particular superconvergence or superaccuary properties and schemes that are more stable than the classical DG method.

This analysis aims to give insight into how the scheme obtains high-order accuracy and into the origin of the scheme's severe stability restriction.

This thesis is organized as follows. In the remainder of this chapter we will derive the DG method applied to scalar conservation laws, systems, and then multidimensional problems. In Chapter 2, we use classical Fourier analysis to analyse the superconvergence, superaccuracy, and stability of the scheme applied to a model linear problem. In Chapter 3, we propose a modification to the usual DG scheme which we demonstrate to lower the formal accuracy of the scheme while significantly improving its stability. Chapter 4 extends the analysis of the superconvergence, superaccuracy, and stability of the scheme to two-dimensional problems on triangular meshes. Finally, in Chapter 5 we propose analogous modification to the two-dimensional scheme proposed in Chapter 3 in order to obtain similar stability improvements.

1.2 The DG Scheme

1.2.1 One-dimensional Scalar Equations

To illustrate the DG method, we apply the scheme to the scalar hyperbolic conservation law

$$u_t + f(u)_x = 0, \tag{1.2.1}$$

subject to appropriate initial and boundary conditions on interval I. We begin by discretizing the domain into non-overlapping mesh elements $I_j = [x_j, x_{j+1}]$ of size $h_j = x_{j+1} - x_j$, $j = 1, 2, ..., N$ so that

$$I = \bigcup_{j=1}^{N} I_j.$$

We then approximate u on cell I_j by a function $U_j \in \mathcal{S}$, where $\mathcal{S}$ is a finite dimensional subspace of $\mathcal{L}^2[x_j, x_{j+1}]$ which we refer to as the finite element space. Using this approximation in (1.2.1), we multiply (1.2.1) by a test function V and integrate the result on I_j to obtain

$$\frac{d}{dt}\int_{x_j}^{x_{j+1}} U_j V \, dx + \int_{x_j}^{x_{j+1}} f(U_j)_x V \, dx = 0, \tag{1.2.2}$$

$\forall V \in \mathcal{V}$. Here, $\mathcal{V}$ is a finite dimensional subspace of the Sobolev space $\mathcal{H}^1[x_j, x_{j+1}]$ that we refer to as the test function space. We then integrate the second integral in (1.2.2) by parts to obtain,

$$\frac{d}{dt}\int_{x_j}^{x_{j+1}} U_j V\,dx - \int_{x_j}^{x_{j+1}} f(U_j)V_x\,dx + \Big[f(U_j)V\Big]_{x_j}^{x_{j+1}} = 0, \tag{1.2.3}$$

$\forall V \in \mathcal{V}$. To simplify computations, we transform this equation to the canonical element $I_0 = [-1, 1]$ through a linear mapping

$$x(\xi) = \frac{x_j + x_{j+1}}{2} + \frac{h_j}{2}\xi. \tag{1.2.4}$$

This yields

$$\frac{h_j}{2}\frac{d}{dt}\int_{-1}^{1} U_j V\,d\xi - \int_{-1}^{1} f(U_j)V_\xi\,d\xi + \Big[f(U_j)V\Big]_{-1}^{1} = 0, \tag{1.2.5}$$

$\forall V \in \mathcal{V}$, where now both U_j and V are understood to be functions of ξ. At this point, we note that the evaluation of $f(U_j)$ at $\xi = 1$ and -1 is not well defined since the numerical approximation U is potentially double-valued at each cell interface. To resolve this, let us denote the value of U at x_j by U_j^* and write (1.2.5) as

$$\frac{h_j}{2}\frac{d}{dt}\int_{-1}^{1} U_j V\,d\xi - \int_{-1}^{1} f(U_j)V_\xi\,d\xi + f(U_{j+1}^*)V(1) - f(U_j^*)V(-1) = 0, \tag{1.2.6}$$

$\forall V \in \mathcal{V}$. This is known as the *weak formulation* of the conservation law. We call U_j^* a *Riemann state*. Usual practice is to either specify an exact/approximate Riemann state U_j^* at each cell interface using a *Riemann solver* then evaluate the flux function f at that state, or to specify an exact/approximate value $f(U_j^*)$ itself at each cell interface. A detailed discussion regarding Riemann solvers is outside the scope of this text and we refer the reader to the book by Toro [64] and the references therein for more information on this subject. Here, we will simply state how we will specify U_j^* or $f(U_j^*)$. For this general scalar equation we implement a local Lax-Friedrichs flux [53, 29], i.e.

$$f(U_j^*) = \frac{1}{2}\left(f(U_j(x_j)) + f(U_{j-1}(x_j))\right) - \frac{\left|\lambda_{j-\frac{1}{2}}\right|}{2}\left(U_j(x_j) - U_{j-1}(x_j)\right),$$

where $|\lambda_{j-\frac{1}{2}}| = \max(|f'(U_j(x_j)|, |f'(U_{j-1}(x_j)|)$ is the largest wave speed on either side of the cell interface. Note that this flux reduces simply to the upwind flux for linear conservation laws.

Now, to complete the discretization we must specify our finite element space $\mathcal{S}$ and the test function space $\mathcal{V}$. We choose the finite element space to be $\mathcal{S} = \mathbf{P}_p$, i.e. the space of polynomials of degree less than or equal to p. We choose as the basis for this finite element space the Legendre polynomials [1] P_k, $k = 0, \dots, p$. The Legendre polynomials form an orthogonal family on $[-1, 1]$, i.e.

$$\int_{-1}^{1} P_k P_l \, d\xi = \frac{2}{2k+1} \delta_{kl}, \tag{1.2.7}$$

where δ_{kl} is the Kroneker delta. With the chosen normalization (1.2.7), the values of the basis functions at the end points of the interval $[-1, 1]$ are

$$P_k(1) = 1, \qquad P_k(-1) = (-1)^k. \tag{1.2.8}$$

We write the numerical solution in terms of this basis as

$$U_j = \sum_{l=0}^{p} c_{jl} P_l, \tag{1.2.9}$$

where each coefficient c_{jl} is a function of time t. Then using (1.2.9) in (1.2.6) we obtain a Galerkin formulation by also requiring that the test function $\mathcal{V}$ space is equal to the finite element space, i.e. $\mathcal{V} = \mathcal{S} = \mathbf{P}_p$. Therefore, choosing $V = P_k$, $k = 0, 1, \dots, p$, and using (1.2.8) and (1.2.7) we obtain $p + 1$ equations

$$\frac{h_j}{2k+1} \frac{dc_{jk}}{dt} = -\left[f(U_{j+1}^*) - (-1)^k f(U_j^*) \right] + \int_{-1}^{1} f(U_j) P_k' \, d\xi, \tag{1.2.10}$$

for $k = 0, \dots, p$. We complete the discretization by approximating the integral term in (1.2.10) using a quadrature rule [1] and evolving the solution coefficients c_{jk} in time using a suitable time integration scheme. Although the choice of time integration scheme is arbitrary, in this thesis we will only consider the most commonly employed scheme, i.e. an explicit order $p + 1$ Runge-Kutta (RK) method. With this time integration scheme, it is known [31] that the scheme will be linearly stable when the time step Δt satisfies the Courant-Friedrichs-Lewy (CFL) condition

$$\Delta t \lesssim \min_j \frac{h_j}{(2p+1) \max_{I_j} |f'(U_j)|}.$$

We compute the initial values of c_{jk} through some projection of the initial profile $u(x, 0) = u_0(x)$, usually the L^2 projection

$$\int_{-1}^{1} (U_j - u_j) \, P_k \, d\xi = 0,$$

for $k = 0, \ldots, p$. Finally, we note that near discontinuities the numerical solution may develop oscillations. One approach to suppress such oscillations is apply a *limiter*. A complete discussion of limiters and alternative limiting strategies is beyond the scope of this text but the reader is referred to [53] for a preliminary discussion. In this work, unless otherwise stated, we will implement the moment limiter described in [46].

1.2.2 One-dimensional Systems

The implementation of the DG method applied to a one-dimensional hyperbolic system of conservation laws is analogous to the scalar case. We consider the system

$$\mathbf{u}_t + \mathbf{f}(\mathbf{u})_x = 0, \tag{1.2.11}$$

where $\mathbf{u}$ is vector of m variables. Note that the system is assumed to be hyperbolic, that is, the Jacobian matrix $A(\mathbf{u}) = \mathbf{f}(\mathbf{u})_{\mathbf{u}}$ is diagonalizable and its eigenvalues are always real. Following the same procedure as above we multiply (1.2.11) by a test function V and integrate by parts to obtain the weak formulation

$$\frac{h_j}{2}\frac{d}{dt}\int_{-1}^{1} \mathbf{U}_j V \, d\xi - \int_{-1}^{1} \mathbf{f}(\mathbf{U}_j) V_\xi \, d\xi + \mathbf{f}(\mathbf{U}^*_{j+1})V(1) - \mathbf{f}(\mathbf{U}^*_j)V(-1) = 0,$$

$\forall V \in \mathcal{V}$. We again use the local Lax-Friedrichs flux, this time for systems

$$\mathbf{f}(\mathbf{U}^*_j) = \frac{1}{2}\left(\mathbf{f}(\mathbf{U}_j(x_j)) + \mathbf{f}(\mathbf{U}_{j-1}(x_j))\right) - \frac{\left|\lambda_{j-\frac{1}{2}}\right|}{2}\left(\mathbf{U}_j(x_j) - \mathbf{U}_{j-1}(x_j)\right),$$

where $\lambda_{j-\frac{1}{2}}$ is the largest magnitude eigenvalue of the matrices $A(\mathbf{U}_j(x_j))$ and $A(\mathbf{U}_{j-1}(x_j))$. We then approximate the solution $\mathbf{u}$ by a vector of polynomials

$$\mathbf{U}_j = \sum_{l=0}^{p} \mathbf{c}_{jl} P_l, \tag{1.2.12}$$

and using this in the weak formulation and choosing $V = P_k$, $k = 0, \ldots, p$ we obtain $p+1$ equations

$$\frac{h_j}{2k+1}\frac{d\mathbf{c}_{jk}}{dt} = -\left[\mathbf{f}(\mathbf{U}^*_{j+1}) - (-1)^k \mathbf{f}(\mathbf{U}^*_j)\right] + \int_{-1}^{1} \mathbf{f}(\mathbf{U}_j) P'_k \, d\xi, \tag{1.2.13}$$

for $k = 0, \ldots, p$. The linear stability restriction on Δt now scales with the largest eigenvalue of $A(\mathbf{U})$, i.e.

$$\Delta t \lesssim \min_j \frac{h_j}{(2p+1)|\lambda_{j,max}|}.$$

where $\lambda_{j,max}$ is the largest magnitude eigenvalue of $A(\mathbf{U}_j)$.

1.2.3 Two-dimensional Systems

Although the DG scheme can be applied to problems of arbitrary dimension and to arbitrary cell geometries with a suitable choice of finite element space, we restrict our attention to applying the DG scheme to a two-dimensional problem on a mesh of triangular cells. This way we are able to explicitly state the polynomial basis, and the stability restriction.

We consider the two-dimensional hyperbolic system of conservation laws,

$$\mathbf{u}_t + \nabla \cdot \mathbf{F}(\mathbf{u}) = 0 \tag{1.2.14}$$

subject to appropriate initial and boundary conditions in a region $\Omega \subset \mathbb{R}^2$. Here $\mathbf{F} = (\mathbf{F}_1, \mathbf{F}_2)$ is a tensor of two vector valued flux functions $\mathbf{F}_1(\mathbf{u})$ and $\mathbf{F}_2(\mathbf{u})$. Note that the system is assumed to be hyperbolic, that is, the matrix $\alpha A(\mathbf{u}) + \beta B(\mathbf{u})$ is diagonalizable and has real eigenvalues for any α and β such that $\alpha^2 + \beta^2 = 1$, where $A(\mathbf{u}) = (\mathbf{F}_1)_\mathbf{u}$ and $B(\mathbf{u}) = (\mathbf{F}_2)_\mathbf{u}$. We begin as usual by partitioning the domain Ω into non-overlapping triangular cells Ω_j, $j = 1, 2, ..., N$ so that

$$\Omega = \bigcup_{j=1}^{N} \Omega_j.$$

We again replace the exact solution $\mathbf{u}$ in the conservation law (1.2.14) by a polynomial $\mathbf{U}_j$ on every cell Ω_j. Then multiplying (1.2.14) by a test function V and integrating over Ω_j we obtain,

$$\frac{d}{dt}\iint_{\Omega_j} \mathbf{U}_j V \, dA + \iint_{\Omega_j} \nabla \cdot \mathbf{F}(\mathbf{u}) V \, dA = 0, \tag{1.2.15}$$

$\forall V \in \mathcal{V}$. Then, applying the divergence theorem to the second integral in (1.2.15) we obtain,

$$\frac{d}{dt}\iint_{\Omega_j} \mathbf{U}_j V \, dA - \iint_{\Omega_j} \mathbf{F}(\mathbf{U}_j) \cdot \nabla V \, dA + \oint_{\partial\Omega_j} \mathbf{n} \cdot \mathbf{F}(\mathbf{U}_j^*) V \, ds = 0, \tag{1.2.16}$$

$\forall V \in \mathcal{V}$, where $\partial\Omega_j$ is the boundary of Ω_j oriented counter-clockwise and $\mathbf{n}$ is the outward-facing normal vector to $\partial\Omega_j$. Note that we have again used the notation $\mathbf{U}^*$ since the numerical solution is potentially multi-valued along the cell boundary $\partial\Omega_j$ and we must replace $\mathbf{U}_j$ with a Riemann state. We denote by $\mathbf{U}_{j+}$ the value of the numerical solution in the immediate neighbour of Ω_j along each edge of its boundary $\partial\Omega_j$. Using this, the local Lax-Friedrichs flux can be written

$$\mathbf{n} \cdot \mathbf{F}(\mathbf{U}_j^*) = \frac{1}{2}\mathbf{n} \cdot \left(\mathbf{F}(\mathbf{U}_j) + \mathbf{F}(\mathbf{U}_{j+})\right) - \frac{\left|\lambda_{j+\frac{1}{2}}\right|}{2} \left(\mathbf{U}_{j+} - \mathbf{U}_j\right),$$

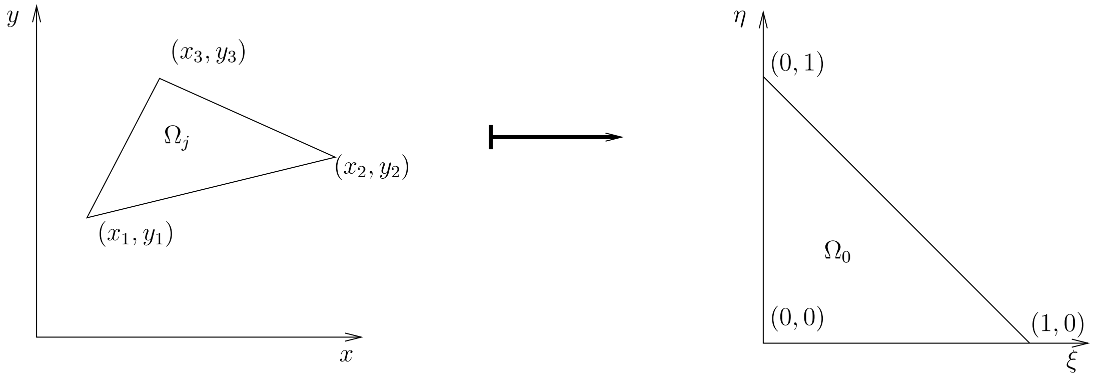


Figure 1.1: The transformation (1.2.17) maps each computational cell Ω_j to the canonical cell Ω_0.

where $\lambda_{j+\frac{1}{2}}$ is the largest magnitude eigenvalue of the matrices $\mathbf{n} \cdot (A(\mathbf{U}_j), B(\mathbf{U}_j))$ and $\mathbf{n} \cdot (A(\mathbf{U}_{j+}), B(\mathbf{U}_{j+}))$. To obtain a more explicit expression for the numerical solution, we proceed as in the one-dimensional case and map each cell Ω_j to a computational cell Ω_0. Suppose the cell Ω_j has vertices at (x_1, y_1), (x_2, y_2), and (x_3, y_3), traveling counter-clockwise. We map Ω_j to the cell Ω_0 in the variables (ξ, η), which has vertices located at (0,0), (1,0), and (0,1). This mapping is given by

$$\begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = \begin{pmatrix} x_1 & x_2 & x_3 \\ y_1 & y_2 & y_3 \\ 1 & 1 & 1 \end{pmatrix} = \begin{pmatrix} 1 - \xi - \eta \\ \xi \\ \eta \end{pmatrix}. \tag{1.2.17}$$

The Jacobian matrix for this transformation is constant and given by

$$J_j = \begin{pmatrix} x_\xi & x_\eta \\ y_\xi & y_\eta \end{pmatrix} = \begin{pmatrix} x_2 - x_1 & x_3 - x_1 \\ y_2 - y_1 & y_3 - y_1 \end{pmatrix}, \tag{1.2.18}$$

and satisfies $\det J_j = 2|\Omega_j|$. Using this mapping in (1.2.16) we can write

$$2|\Omega_j| \frac{d}{dt} \iint_{\Omega_0} \mathbf{U}_j V \, dA - 2|\Omega_j| \iint_{\Omega_0} \mathbf{F}(\mathbf{U}_j) \cdot J_j^{-1} \nabla V \, dA + \oint_{\partial \Omega_j} \mathbf{n} \cdot \mathbf{F}(\mathbf{U}_j^*) V \, ds = 0, \tag{1.2.19}$$

where ∇ is now understood to be an operator in the (ξ, η)-space.

In the new variables ξ and η, we choose the Dubiner basis [33] for the polynomial space $\mathbf{P}_p = \text{span}\{\xi^i \eta^j | i + j \leq p\}$. These basis functions are given by

$$\psi_{ki}(\xi, \eta) = \sqrt{(2i+1)(2k+2)} P_{k-i}^{0,2i+1}(1 - 2\xi)(1 - \xi)^i P_i \left(1 - \frac{2\eta}{1 - \xi}\right), \tag{1.2.20}$$

for $k = 0, \dots, p$ and $i = 0, \dots, k$, where $P_{k-i}^{0,2i+1}$ is the degree $k - i$ Jacobi polynomial with parameters 0 and $2i + 1$ and P_i is the degree i Legendre polynomial. This basis satisfies

$$\iint_{\Omega_0} \psi_{ki}\psi_{lm}\, dA = \delta_{kl}\delta_{im}, \tag{1.2.21}$$

where δ_{kl} is the Kronecker delta function. We write the numerical solution in terms of this basis

$$\mathbf{U}_j = \sum_{m=0}^{p} \sum_{l=0}^{k} \mathbf{c}_{jml}\psi_{ml}(\xi, \eta), \tag{1.2.22}$$

and use this in (1.2.19), taking $V = \psi_{ki}$ for $k = 0, \dots, p$ and $i = 0, \dots, k$, and use the orthogonality (1.2.21) to obtain $\frac{1}{2}(p+1)(p+2)$ equations

$$2|\Omega_j|\frac{d}{dt}\mathbf{c}_{jki} = -\oint_{\partial\Omega_j} \mathbf{n} \cdot \mathbf{F}(\mathbf{U}_j^*)\psi_{ki}\, ds + 2|\Omega_j| \iint_{\Omega_0} \mathbf{F}(\mathbf{U}_j) \cdot J_j^{-1}\nabla\psi_{ki}\, dA, \tag{1.2.23}$$

for $k = 0, \dots, p$ and $i = 0, \dots, k$. The discretization is then completed by approximating the volume integral using a quadrature rule over Ω_0 [34] and approximating the integral along each edge of $\partial\Omega_j$ by using an appropriate one-dimensional quadrature. We then time evolve the solution coefficients using an explicit order $p+1$ Runge-Kutta (RK) time-stepping scheme. The usual CFL condition used in these problems can be written

$$\Delta t \lesssim \min_j \frac{r_j}{(2p+1)|\lambda_{j,max}|},$$

where r_j is the radius of the inscribed circle in Ω_j and $\lambda_{j,max}$ is the largest magnitude eigenvalue of the matrix $\alpha A(\mathbf{u}) + \beta B(\mathbf{u})$ over the cell Ω_j, where $\alpha^2 + \beta^2 = 1$.

Chapter 2

Superconvergence and Superaccuracy of the DG Method

2.1 Introduction

In this chapter we investigate the superconvergence, superaccuracy, and stability properties of DG method in one dimension. To do this, we apply the DG method to the one-dimensional linear hyperbolic problem

$$u_t + au_x = 0, \tag{2.1.1}$$

with $a > 0$ constant, subject to periodic boundary conditions on interval I and sufficiently smooth initial data $u_0(x)$.

We begin by deriving a PDE on the j-th cell which is solved by the polynomial numerical solution U_j exactly. This PDE is equivalent to the original advection equation but with a forcing term. Then, by applying classical Fourier analysis, we find the Fourier modes of this PDE which are polynomial in space. These solutions are completely determined by the inflow into each cell and a rational function of the mode's frequency. These rational functions are also closely related to both the $(p+1)$-th right Radau polynomial R^-_{p+1} and the $\frac{p}{p+1}$ Padé approximant of the exponential function e^z, where p is the degree of the polynomial approximation. We use these particular polynomial solutions to investigate the superconvergence, superaccuracy, and stability of the method. Specifically, we determine the local superconvergence properties of the method by assuming an exact inflow into a cell and considering a local expansion of the Fourier modes. Then, by looking for particular wave solutions, we find the numerical dispersion relation and establish the superaccurate errors of the method in dissipation and dispersion. Finally we establish global superconvergence results and show in particular that for a uniform computational mesh of N elements there exist $(p+1)N$ independent polynomial solutions, N of which can be seen as physical and pN as non-physical. The physical modes of the solution exhibit the familiar superconvergence properties, while the non-physical modes are damped exponentially quickly in time. This property was conjectured by Biswas et al [17] in 1994.

Superconvergence of the DG method for one-dimensional problems has been studied in several papers. Following the conjecture made by Biswas et al, Adjerid et al in [4] proved order $(p+2)$ convergence of the DG solution at the downwind-based Radau points and order $2p+1$ convergence at the downwind end of each cell for ODEs. An order $(p+\frac{3}{2})$ convergence rate of the DG solution to a particular projection of the exact solution was later shown by Cheng and Shu in [20, 21]. Yang and Shu then showed the same superconvergence property with order $p+2$ convergence for linear hyperbolic equations in [72]. Fourier analysis of the DG solution has also been applied to the DG solution in order to investigate superconvergence by symbolically manipulating the discretetization matrices for low order ($p = 1, 2,$ and 3) approximations [73, 38].

Likewise, the connection between the DG scheme and the Padé approximants of the exponential function has been observed in several works. The stability region of the DG method for ODEs was demonstrated by Le Saint and Raviart [52] to be given by $|R(\lambda h)| \leq 1$ where $R(z)$ is the $\frac{p}{p+1}$ Padé approximant of e^z and h is the grid spacing. In [43], Hu and Atkins conjectured that certain polynomials involved in the analysis of the numerical dispersion relation are related to $\frac{p+1}{p}$ Padé approximant of e^z and used this to show that the numerical dispersion relation is accurate to $(\kappa h)^{2p+2}$, where κ is the wavenumber. This conjecture was proven and an extended analysis of the dispersion and dissipation errors was given by Ainsworth in [6]. Later, a connection between the spectrum of the DG method on linear problems and the Padé approximant of the exponential was investigated by Krivodonova and Qin in [48]. In this chapter, we demonstrate that the numerical solutions of the DG method are themselves closely related to this Padé approximant and furthermore both the superconvergent local errors and superaccurate errors in dissipation and dispersion of the method can be seen as resulting from the accuracy of this Padé approximant.

The remainder of this chapter is organized as follows. In Section 2 we apply the DG method to the linear problem (2.1.1) and use the formulation to write a PDE which is solved by the numerical solution U_j on the j-th cell. In Section 3 we decompose the numerical solution into Fourier modes, we use this PDE to find particular numerical solutions and establish our main results. These results are then illustrated numerically in Section 4.

2.2 PDE for the DG Solution

Our goal in this section is to derive a partial differential equation for the numerical solution of the DG method. We begin by applying the DG method (1.2.10) to the linear conservation law (2.1.1) to obtain the scheme

$$\frac{h_j}{2k+1}\frac{dc_{jk}}{dt} = -a\left[U_{j+1}^* - (-1)^k U_j^*\right] + a\int_{-1}^{1} U_j P_k'\,d\xi. \tag{2.2.1}$$

We note that in the case of the linear flux $f(u) = au$, the local Lax-Friedrichs flux we implement reduces to the upwind flux, which can be written as choosing the Riemann state U_j^* to be the value of the numerical solution from the previous cell, i.e. $U_j^* = U_{j-1}(x_j)$. Therefore we can write the scheme (2.2.1) as

$$\frac{h_j}{2k+1}\frac{dc_{jk}}{dt} = -a\left[U_j(x_{j+1}) - (-1)^k U_{j-1}(x_j)\right] + a\int_{-1}^{1} U_j P_k'\,d\xi. \tag{2.2.2}$$

To derive a PDE which the numerical solution U_j solves, we begin by integrating the integral in (2.2.2) by parts to write

$$\frac{h_j}{2k+1}\frac{dc_{jk}}{dt} = -(-1)^k a[[U_j]] - a\int_{-1}^{1} \frac{\partial U_j}{\partial \xi} P_k\,d\xi. \tag{2.2.3}$$

where $[[U_j]] = U_j(x_j) - U_{j-1}(x_j)$ denotes the jump between the endpoints of the numerical solution at the interface of the j-th and $(j-1)$-th cells. Note that the integral in this expression is entirely local, as opposed to the term in (1.2.2). Since we are interested in the equation which U_j itself satisfies we can reconstruct U_j by multiplying each equation in the system by P_k and summing over all k and using the expression for U_j in (1.2.9). We can thereby write the following exact expression for $\frac{\partial}{\partial t}U_j$ after some rearrangement,

$$\frac{\partial}{\partial t}U_j + \frac{2a}{h_j}\sum_{k=0}^{p}\frac{2k+1}{2}\left(\int_{-1}^{1}\frac{\partial U_j}{\partial \xi}P_k\,d\xi\right)P_k = -\frac{a}{h_j}[[U_j]]\left(\sum_{k=0}^{p}(-1)^k(2k+1)P_k\right). \tag{2.2.4}$$

Because the Legendre polynomials are an orthogonal family we have that the first summed term in this expression is simply the projection of $\frac{\partial U_j}{\partial \xi}$ into the finite element space $\mathbf{P}_p$. Moreover, since $\frac{\partial U_j}{\partial \xi}$ is already in the finite element space this projection is exact. Hence we can write,

$$\frac{\partial}{\partial t}U_j + \frac{2a}{h_j}\frac{\partial}{\partial \xi}U_j = -\frac{a}{h_j}[[U_j]]\left(\sum_{k=0}^{p}(-1)^k(2k+1)P_k\right). \tag{2.2.5}$$

To simplify this expression further let us use the following proposition:

Proposition 2.1.

$$-\sum_{k=0}^{p}(-1)^k(2k+1)P_k = 2\frac{d}{d\xi}R^-_{p+1}(\xi),$$

where R^-_{p+1} is the right Radau polynomial [1] of degree $p+1$, which we defined as $R^-_{p+1} = \frac{(-1)^{p+1}}{2}(P_{p+1} - P_p)$.

Proof. It is known that the Legendre polynomials satisfy [1],

$$\frac{d}{d\xi}P_{k+1} = (2k+1)P_k + (2k-3)P_{k-2} + (2k-7)P_{k-4} + \dots$$

Therefore, a simple calculation shows

$$\begin{aligned}\frac{d}{d\xi}R^-_{p+1}(\xi) &= \frac{(-1)^{p+1}}{2}\frac{d}{d\xi}[P_{p+1} - P_p] \\ &= \frac{(-1)^{p+1}}{2}[(2p+1)P_p + (2p-3)P_{p-2} + \dots \\ &\qquad\qquad - (2p-1)P_{p-1} - (2p-5)P_{p-3} - \dots] \\ &= -\frac{1}{2}\sum_{k=0}^{p}(-1)^k(2k+1)P_k,\end{aligned}$$

which completes the proof. □

Using this proposition, we rewrite (2.2.5) compactly as

$$\frac{\partial}{\partial t}U_j + \frac{2a}{h_j}\frac{\partial}{\partial \xi}U_j = \frac{2a}{h_j}[[U_j]]\frac{d}{d\xi}R^-_{p+1}(\xi). \tag{2.2.6}$$

This is an equation which the polynomial approximation U_j will satisfy exactly on the cell I_j. In particular, U_j solves the same advection equation as the exact solution u, except with a forcing term. Note that when approximating smooth solutions the polynomial approximation U_j will be locally order $p+1$ accurate to the exact solution and thus the jump term at the cell interface $[[U_j]]$ will be of order $p+1$ as well. This implies that the solution to (2.2.6) and, hence, the numerical approximation will in a sense be close to a solution of the advection equation since the forcing term is small.

2.3 Fourier Analysis

To investigate the properties of solutions of (2.2.6) we look at a single Fourier mode solution of the form $U_j(\xi, t) = \hat{U}_j(\xi, \omega)e^{-a\omega t}$, where $\hat{U}_j$ is a polynomial of degree p in ξ. Using these assumptions on the form of $U_j(\xi, t)$ in (2.2.6) we have that this Fourier mode satisfies the ODE

$$-a\omega\hat{U}_j + \frac{2a}{h_j}\frac{\partial}{\partial\xi}\hat{U}_j = \frac{2a}{h_j}[[\hat{U}_j]]\frac{d}{d\xi}R^-_{p+1}(\xi). \tag{2.3.1}$$

We can solve this ODE explicitly to obtain

$$\hat{U}_j(\xi, \omega) = \hat{U}_j(-1, \omega)e^{\frac{\omega h_j}{2}(\xi+1)} + [[\hat{U}_j]]\int_{-1}^{\xi} e^{\frac{\omega h_j}{2}(\xi-s)}\frac{d}{ds}R^-_{p+1}(s)\, ds. \tag{2.3.2}$$

The general solution (2.3.2) is not necessarily a polynomial in ξ. Therefore, (2.3.2) is too general for our purposes. Below, we will look for additional restrictions which ensure that this solution is polynomial in ξ. We state two lemmas which will help us to rewrite and investigate the integral term in (2.3.2).

Lemma 2.1. *The integral term in* (2.3.2) *satisfies the following relation*

$$\int_{-1}^{\xi} e^{\frac{\omega h_j}{2}(\xi-s)}\frac{d}{ds}R^-_{p+1}(s)\, ds = -e^{\frac{\omega h_j}{2}(\xi+1)} + \frac{1}{(\omega h_j)^{p+1}}\left(g(\omega h_j)e^{\frac{\omega h_j}{2}(\xi+1)} - f(\omega h_j, \xi)\right), \tag{2.3.3}$$

where $g(\omega h_j)$ is a polynomial of degree $p+1$ in ωh_j and $f(\omega h_j, \xi)$ is a polynomial of degree p in both ωh_j and ξ.

Proof. We begin by integrating the integral in (2.3.2) by parts

$$\begin{aligned}\int_{-1}^{\xi} e^{\frac{\omega h_j}{2}(\xi-s)}\frac{d}{ds}R^-_{p+1}(s)\, ds = &\left(\frac{2}{\omega h_j}\right)\left[\frac{d}{d\xi}R^-_{p+1}(-1)e^{\frac{\omega h_j}{2}(\xi+1)} - \frac{d}{d\xi}R^-_{p+1}(\xi)\right] \\ &+ \left(\frac{2}{\omega h_j}\right)\int_{-1}^{\xi} e^{\frac{\omega h_j}{2}(\xi-s)}\frac{d^2}{ds^2}R^-_{p+1}(s)\, ds,\end{aligned}$$

and then continue integrating by parts until the remaining integral vanishes to obtain

$$\int_{-1}^{\xi} e^{\frac{\omega h_j}{2}(\xi-s)}\frac{d}{ds}R^-_{p+1}(s)\, ds = \frac{1}{(\omega h_j)^{p+1}}\left(\tilde{g}(\omega h_j)e^{\frac{\omega h_j}{2}(\xi+1)} - f(\omega h_j, \xi)\right), \tag{2.3.4}$$

where

$$\tilde{g}(\omega h_j) = \sum_{k=1}^{p+1} 2^k (\omega h_j)^{p+1-k} \frac{d^k}{d\xi^k} R^-_{p+1}(-1), \tag{2.3.5}$$

$$f(\omega h_j, \xi) = \sum_{k=1}^{p+1} 2^k (\omega h_j)^{p+1-k} \frac{d^k}{d\xi^k} R^-_{p+1}(\xi). \tag{2.3.6}$$

Note that $\tilde{g}(\omega h_j)$ is a polynomial of degree p in ωh_j, and $f(\omega h_j, \xi)$ is polynomial of degree p in ωh_j and ξ. Therefore, defining $g(\omega h_j) = (\omega h_j)^{p+1} + \tilde{g}(\omega h_j)$ in (2.3.4) we will obtain equation (2.3.3) which completes the proof. □

Lemma 2.2. *The integral term in* (2.3.2) *also satisfies*

$$\begin{aligned}\int_{-1}^{\xi} e^{\frac{\omega h_j}{2}(\xi-s)} \frac{d}{ds} R^-_{p+1}(s)\, ds = -e^{\frac{\omega h_j}{2}(\xi+1)} + R^-_{p+1}(\xi) \\ + \sum_{k=1}^{\infty} \left(\frac{\omega h_j}{2}\right)^k \frac{1}{(k-1)!} \int_{-1}^{\xi} (\xi - s)^{k-1} R^-_{p+1}(s)\, ds.\end{aligned} \tag{2.3.7}$$

Proof. We prove this lemma in a similar manner to Lemma 1, i.e., by integrating the integral in (2.3.2) by parts, this time in reverse order

$$\int_{-1}^{\xi} e^{\frac{\omega h_j}{2}(\xi-s)} \frac{d}{ds} R^-_{p+1}(s)\, ds = -e^{\frac{\omega h_j}{2}(\xi+1)} + R^-_{p+1}(\xi) + \frac{\omega h_j}{2} \int_{-1}^{\xi} e^{\frac{\omega h_j}{2}(\xi-s)} R^-_{p+1}(s)\, ds.$$

Here, from the definition of the Radau polynomial R^-_{p+1}, we have used $R^-_{p+1}(-1) = 1$. We continue integrating by parts to obtain

$$\begin{aligned}\int_{-1}^{\xi} e^{\frac{\omega h_j}{2}(\xi-s)} \frac{d}{ds} R^-_{p+1}(s)\, ds = -e^{\frac{\omega h_j}{2}(\xi+1)} + R^-_{p+1}(\xi) + \left(\frac{\omega h_j}{2}\right) R^{-,(-1)}_{p+1}(\xi) \\ + \left(\frac{\omega h_j}{2}\right)^2 R^{-,(-2)}_{p+1}(\xi) + \ldots,\end{aligned} \tag{2.3.8}$$

where we define $R^{-,(-k)}_{p+1}$ to be the repeated integrals of the right Radau polynomial, i.e., $R^{-,(0)}_{p+1}(\xi) = R^-_{p+1}(\xi)$ and

$$R^{-,(-(k+1))}_{p+1}(\xi) = \int_{-1}^{\xi} R^{-,(-k)}_{p+1}(s)\, ds. \tag{2.3.9}$$

Finally, using this definition with the Cauchy integration formula we can write the polynomials $R_{p+1}^{-,(-k)}$ as

$$R_{p+1}^{-,(-k)}(\xi) = \frac{1}{(k-1)!} \int_{-1}^{\xi} (\xi - s)^{k-1} R_{p+1}^{-}(s)\, ds, \tag{2.3.10}$$

which, when used in (2.3.8), yields (2.3.7) and completes the proof. □

From these two lemmas we can establish a useful result regarding the polynomials g and f.

Corollary 2.1. *The rational function $\frac{f(\omega h_j,\xi)}{g(\omega h_j)}$ has the expansion*

$$\frac{f(\omega h_j, \xi)}{g(\omega h_j)} = e^{\frac{\omega h_j}{2}(\xi+1)} - \frac{(\omega h_j)^{p+1}}{g(\omega h_j)} \left[R_{p+1}^{-}(\xi) + \sum_{k=1}^{\infty} \left(\frac{\omega h_j}{2} \right)^{k} R_{p+1}^{-,(-k)}(\xi) \right] \tag{2.3.11}$$

and, in particular,

$$\frac{f(\omega h_j, 1)}{g(\omega h_j)} = e^{\omega h_j} + \mathcal{O}((\omega h_j)^{2p+2}), \tag{2.3.12}$$

i.e. $\frac{f(z,1)}{g(z)}$ is the $\frac{p}{p+1}$ Padé approximant of e^z.

Before we state the proof of this corollary let us briefly recall the definition of a Padé approximant [8].

Definition 1. *Given integers m and n and a sufficiently smooth function $F(z)$, the $\frac{m}{n}$ Padé approximant of $F(z)$ is a rational function $\frac{P(z)}{Q(z)}$ where $P(z)$ and $Q(z)$ are polynomials of degree m and n, respectively, and satisfy*

$$\frac{P(z)}{Q(z)} = F(z) + \mathcal{O}(z^{m+n+1}).$$

This Padé approximant is unique up to a constant multiple of the numerator and denominator. It is conventional to take $Q(0) = 1$ so that the Padé approximant is uniquely defined.

We now proceed to prove the Corollary.

Proof of Corollary 2.1. Equating the right hand sides of (2.3.3) and (2.3.7) we obtain

$$-e^{\frac{\omega h_j}{2}(\xi+1)} + \frac{1}{(\omega h_j)^{p+1}}\Big(g(\omega h_j)e^{\frac{\omega h_j}{2}(\xi+1)} - f(\omega h_j,\xi)\Big) = -e^{\frac{\omega h_j}{2}(\xi+1)} + \left[R^-_{p+1}(\xi) + \sum_{k=1}^{\infty}\left(\frac{\omega h_j}{2}\right)^k R^{-,(-k)}_{p+1}(\xi)\right].$$

Solving this expression for $\frac{f(\omega h_j,\xi)}{g(\omega h_j)}$ immediately yields (2.3.11). Subsequently, evaluating (2.3.11) at $\xi = 1$ we obtain

$$\frac{f(\omega h_j,1)}{g(\omega h_j)} = e^{\omega h_j} - \frac{(\omega h_j)^{p+1}}{g(\omega h_j)}\left[R^-_{p+1}(1) + \sum_{k=1}^{\infty}\left(\frac{\omega h_j}{2}\right)^k R^{-,(-k)}_{p+1}(1)\right].$$

From the definition of R^-_{p+1} in terms of the Legendre polynomials we have that $R^-_{p+1}(1) = 0$. Furthermore, from the definition of $R^{-,(-k)}_{p+1}(\xi)$ in (2.3.10) and the orthogonality of R^-_{p+1} to all polynomials of degree $p-1$ we have that $R^{-,(-k)}_{p+1}(1) = 0$ for $k = 1,\dots,p$. We therefore find that

$$\frac{f(\omega h_j,1)}{g(\omega h_j)} = e^{\omega h_j} + \frac{(\omega h_j)^{p+1}}{g(\omega h_j)}\left[\sum_{k=p+1}^{\infty}\left(\frac{\omega h_j}{2}\right)^k R^{-,(-k)}_{p+1}(1)\right],$$

which yields

$$\frac{f(\omega h_j,1)}{g(\omega h_j)} = e^{\omega h_j} + \mathcal{O}((\omega h_j)^{2p+2}).$$

By Lemma 2.1, $f(z,1)$ is a polynomial of degree p while $g(z)$ is a polynomial of degree $p+1$ with the form $g(z) = z^{p+1} + \tilde{g}(z)$. Therefore, we find after a possible rescaling that the rational function $\frac{f(z,1)}{g(z)}$ approximates e^z to order $2p+2$. Therefore it is the unique $\frac{p}{p+1}$ Padé approximant of e^z. □

Using Lemmas 2.1 and 2.2 we can write the general solution (2.3.2) in two ways:

$$\begin{aligned}\hat{U}_j(\xi,\omega) &= \hat{U}_j(-1,\omega)e^{\frac{\omega h_j}{2}(\xi+1)} - [[\hat{U}_j]]e^{\frac{\omega h_j}{2}(\xi+1)} + \frac{[[\hat{U}_j]]}{(\omega h_j)^{p+1}}\Big(g(\omega h_j)e^{\frac{\omega h_j}{2}(\xi+1)} - f(\omega h_j,\xi)\Big)\\ &= \hat{U}_{j-1}(1,\omega)e^{\frac{\omega h_j}{2}(\xi+1)} + \frac{[[\hat{U}_j]]}{(\omega h_j)^{p+1}}\Big(g(\omega h_j)e^{\frac{\omega h_j}{2}(\xi+1)} - f(\omega h_j,\xi)\Big), \qquad (2.3.13)\end{aligned}$$

and

$$\begin{aligned}\hat{U}_j(\xi,\omega) &= \hat{U}_j(-1,\omega)e^{\frac{\omega h_j}{2}(\xi+1)} - [[\hat{U}_j]]e^{\frac{\omega h_j}{2}(\xi+1)} \\ &\qquad + [[\hat{U}_j]]\left[R^{-}_{p+1}(\xi) + \sum_{k=1}^{\infty}\left(\frac{\omega h_j}{2}\right)^k R^{-,(-k)}_{p+1}(\xi)\right] \\ &= \hat{U}_{j-1}(1,\omega)e^{\frac{\omega h_j}{2}(\xi+1)} + [[\hat{U}_j]]\left[R^{-}_{p+1}(\xi) + \sum_{k=1}^{\infty}\left(\frac{\omega h_j}{2}\right)^k R^{-,(-k)}_{p+1}(\xi)\right]. \qquad (2.3.14)\end{aligned}$$

Now, the solution corresponding to the exact advection of the downwind point $\hat{U}_{j-1}(1,\omega)$ is $\hat{U}_{j-1}(1,\omega)e^{\frac{\omega h_j}{2}(\xi+1)}$ and, hence, from (2.3.13) and (2.3.14) the general solution for the numerical approximation in cell I_j consists of two parts: exact advection of the downwind value in cell I_{j-1} and higher-order error terms which are proportional to the magnitude of the jump at that interface. This gives rise to the local superconvergence properties of the method which we state formally in the following theorem.

Theorem 2.1 (Local Superconvergence)**.** *Let $u(x,t)$ be a smooth exact solution of* (2.1.1) *on the interval I with suitable boundary conditions. Let U be the numerical solution of the DG scheme* (2.2.2) *on a mesh of N elements and let U_j be the restriction of the numerical solution to the cell I_j. Let $\epsilon_j(\xi,t) = U_j - u_j$ be the numerical error on I_j (mapped to the canonical element* $[-1,1]$*). Suppose the inflow $U_{j-1}(x_j,t)$ into cell I_j is exact, i.e. $U_{j-1}(x_j,t) = u(x_j,t)$. Then the numerical error on cell I_j satisfies*

$$\epsilon_j(\xi,t) = [[U_j]]R^{-}_{p+1}(\xi) + \mathcal{O}(h_j^{p+2}), \qquad (2.3.15)$$

and

$$\epsilon_j(1,t) = \mathcal{O}(h_j^{2p+2}). \qquad (2.3.16)$$

Proof. We prove this by first considering a single Fourier mode of the error ϵ_j. That is, we consider an exact solution of the form $u_j(\xi,t) = e^{\frac{\omega h_j}{2}(\xi+1)-a\omega t}$ so that the exact inflow into cell I_j is $\hat{U}_{j-1}(1,\omega) = 1$. We then find using (2.3.14) that

$$\epsilon_j(\xi,t) = [[U_j]]\left[R^{-}_{p+1}(\xi) + \sum_{k=1}^{\infty}\left(\frac{\omega h_j}{2}\right)^k R^{-,(-k)}_{p+1}(\xi)\right],$$

and furthermore, evaluating at $\xi = 1$ we have

$$\epsilon_j(1,t) = [[U_j]]\left[\sum_{k=p+1}^{\infty}\left(\frac{\omega h_j}{2}\right)^k R^{-,(-k)}_{p+1}(1)\right].$$

We therefore obtain (2.3.15) and (2.3.16) by summing these expressions over all possible frequencies ω. □

Next, as mentioned above we are interested only in polynomial solutions of (2.3.1). The reason for this is that we know the numerical solution is polynomial in ξ for all times t. Hence, the numerical solution should be composed solely of solutions of (2.3.1) which are polynomials in ξ. By examining (2.3.13) we see that the solutions U_j will be polynomial in ξ only when

$$\hat{U}_{j-1}(1,\omega) + [[\hat{U}_j]]\frac{g(\omega h_j)}{(\omega h_j)^{p+1}} = 0 \tag{2.3.17}$$

is satisfied. Hence, assuming $g(\omega h_j) \neq 0$, we obtain after rearranging that $\hat{U}_j(-1,\omega)$ is related to $\hat{U}_{j-1}(1,\omega)$ by

$$\hat{U}_j(-1,\omega) = \hat{U}_{j-1}(1,\omega)\frac{g(\omega h_j) - (\omega h_j)^{p+1}}{g(\omega h_j)}.$$

Using the above relation in (2.3.13) we obtain after rearranging that the polynomial solutions of (2.3.1) have the form

$$\hat{U}_j(\xi,\omega) = \hat{U}_{j-1}(1,\omega)\frac{f(\omega h_j,\xi)}{g(\omega h_j)}. \tag{2.3.18}$$

Thus, we obtain that the polynomial solutions on each cell are completely determined by the rational function $\frac{f(\omega h_j,\xi)}{g(\omega h_j)}$ and the value of the numerical solution at the downwind point of the previous cell. The relation of the numerical solution to this rational function, which itself is connected to the Padé approximant of e^z is also related to the study of the superaccurate errors in disipation and dispersion of the DG scheme. The same Padé approximant was studied by Hu and Atkins in [43], Ainsworth in [6], and Krivodonova and Qin in [48]. In each paper the authors note that the superaccuracies in dissipation and dispersion errors stem from the accuracy of this Padé approximant. A key difference here, however, is that we have not made the assumption of a uniform mesh. Hence we can extend the previously known results concerning the $2p+1$ order of accuracy in dissipation and $2p+2$ order of accuracy in dispersion of the DG method to non-uniform meshes.

To see this, we consider how the Fourier mode with frequency ω propagates through cell I_j. An exact solution to (2.1.1) of the form $u(x,t) = \exp(\kappa x - a\omega t)$ would satisfy the dispersion relation $\kappa = \omega$ and the relation $u(x_{j+1},t) = u(x_j,t)e^{\kappa h_j}$. The numerical solution with frequency ω, on the other hand, satisfies

$$U_j(1,t) = U_{j-1}(1,t)e^{\tilde{\kappa} h_j}, \tag{2.3.19}$$

for some $\tilde{\kappa}$ which we call the numerical wavenumber. Note that we call errors in $\tilde{\kappa}$ dissipation or dispersion depending on whether the error is real or imaginary. For sinusoidal waves the exact frequency ω is purely imaginary and therefore even powers of ω in the error (2.3.21) contribute dissipation error and odd powers of ω contribute dispersion error. We state the accuracy of this numerical wavenumber in the following theorem.

Theorem 2.2 (Superaccuracy)**.** *The numerical dispersion relation of the DG scheme applied to the linear equation* (2.1.1) *on cell I_j between a frequency ω and the numerical wavenumber $\tilde{\kappa}$ can be written,*

$$\frac{f(\omega h_j, 1)}{g(\omega h_j)} = e^{\tilde{\kappa} h_j}. \tag{2.3.20}$$

The numerical wavenumber then satisfies

$$\tilde{\kappa} = \omega + C_1 \omega^{2p+2} h_j^{2p+1} + C_2 \omega^{2p+3} h_j^{2p+2} + \ldots, \tag{2.3.21}$$

i.e. the scheme has order $2p+1$ accuracy in dissipation and order $2p+2$ accuracy in dispersion.

Proof. We obtain the dispersion relation (2.3.20) immediately by considering a Fourier mode solution of (2.3.1) with frequency ω, which can be written as the rational function (2.3.18), and defining the numerical wavenumber $\tilde{\kappa}$ as in (2.3.19). We then obtain the expansion (2.3.21) by performing a Taylor series of the dispersion relation (2.3.20), using expansion (2.3.11) from Corollary 2.1, and solving for $\tilde{\kappa}$. □

In the above analysis, we find the numerical dispersion relation of the scheme by assuming that ω was an exact frequency and finding the numerical wavenumber $\tilde{k}_n$. Another approach was taken in [48], where the authors were interested in the spectrum of the DG method, i.e. the precise values of the numerical frequencies ω for an exact wavenumber κ. We now show that we can use this approach to give an estimate of the numerical frequencies ω for problems with periodic boundary conditions. To this end we use the relation between the downwind points of U in (2.3.19) and enforce the periodicity of the numerical solution to obtain the following condition on ω

$$\prod_{j=1}^{N} \frac{f(\omega h_j, 1)}{g(\omega h_j)} = 1. \tag{2.3.22}$$

Hence, the admissible numerical frequencies ω must satisfy this relation. Solving (2.3.22) for every value of ω, however, is difficult since it would require finding the roots of a high-order polynomial. An attempt to describe these values for some particular meshes was made in [49], but this is beyond the scope of this chapter. We will instead make the simplifying assumption that the mesh is uniform and obtain that the values of ω are solutions of

$$\frac{f(\omega h, 1)}{g(\omega h)} = e^{\kappa_n h}, \tag{2.3.23}$$

where $e^{\kappa_n h}$ is an N-th root of unity, i.e. $\kappa_n = \frac{2\pi n i}{L}$, $n = 0, \ldots, N-1$, where L is the length of the domain I. Note that since the mesh is uniform and each downwind point of this solution is related by $\hat{U}_j(1,\omega) = \frac{f(\omega h,1)}{g(\omega h)}\hat{U}_{j-1}(1,\omega) = e^{\kappa_n h}\hat{U}_{j-1}(1,\omega)$, the exact physical frequency for this wave is $\omega = \kappa_n$. In the following theorem we give an estimate on the values of ω which are solutions of (2.3.23).

Theorem 2.3 (Physical Spectrum)**.** *Let U be the numerical solution of the DG scheme (2.2.2) on a uniform mesh of N elements on the interval I with periodic boundary conditions, and let U_j be the restriction of the numerical solution to the cell I_j.*

The numerical solution U can be decomposed into $(p+1)N$ solutions. Each of these solutions is polynomial in ξ and has the form $U_j(\xi, t) = \hat{U}_j(\xi,\omega)e^{-a\omega t}$. These solutions also satisfy $\hat{U}_j(1,\omega) = e^{\kappa_n h}\hat{U}_{j-1}(1,\omega)$ for each j where $\kappa_n = \frac{2\pi n i}{L}$, $n = 0,\ldots,N-1$. Corresponding to each κ_n there are $p+1$ spectral values $\omega = \omega_0, \omega_1, \ldots, \omega_p$ which have the expansions

$$\omega_0 = \kappa_n + \mathcal{O}(\kappa_n^{2p+2}h^{2p+1})$$

and

$$\omega_m = \frac{\mu_m}{h} + \mathcal{O}(\kappa_n), \qquad m = 1, \ldots, p,$$

where μ_m are the p non-zero roots of the polynomial $g(z) - f(z)$ and satisfy $\mathrm{Re}(\mu_m) > 0$.

Proof. We begin by noting that from (2.3.12)

$$\frac{f(\omega h, 1)}{g(\omega h)} = e^{\omega h} + \mathcal{O}((\omega h)^{2p+2}),$$

there should be at least one solution of (2.3.23) of the form $\omega = \kappa_n + \mathcal{O}(\kappa_n^{2p+2}h^{2p+1})$. The condition (2.3.23) itself for the numerical frequency ω can be rearranged to obtain

$$g(\omega h)e^{\kappa_n h} - f(\omega h, 1) = 0. \tag{2.3.24}$$

This expression is a polynomial of degree $p+1$ in ω and, therefore, has up to $p+1$ distinct roots. Regarding h as a small parameter, we have from the form of (2.3.24) that we can asymptotically approximate each root using the expansion

$$\omega = \frac{d_{-1}}{h} + d_0 + d_1 h + \dots .$$

Using this expansion in (2.3.24), and expanding $e^{\kappa_n h} = 1 + \kappa_n h + \mathcal{O}((\kappa_n h)^2)$, we obtain

$$g(d_{-1}) - f(d_{-1}, 1) + g(d_{-1})\kappa_n h + g'(d_{-1})d_0 h - f'(d_{-1}, 1)d_0 h + \mathcal{O}((\kappa_n h)^2) = 0. \quad (2.3.25)$$

Setting the powers of h equal to zero we find that we can determine the leading order asymptotic behaviour of each root by finding the possible values of the coefficient d_{-1} which solve

$$g(d_{-1}) - f(d_{-1}, 1) = 0. \quad (2.3.26)$$

Firstly, evaluating (2.3.12) at $\omega h_j = 0$ gives that $f(0, 1) = g(0)$, so $d_{-1} = 0$ is a root of (2.3.26). Furthermore, differentiating (2.3.12) and evaluating at $\omega h_j = 0$ yields that $f'(0, 1) \neq g'(0)$ and, hence, $d_{-1} = 0$ is a simple root. Finally, when this Padé approximant was studied in [48], the authors showed that non-zero roots of the polynomial $g(z) - f(z, 1)$ lay in the right-half complex plane. Therefore we can conclude that there are p roots of the form

$$\omega_m = \frac{\mu_m}{h} + \mathcal{O}(\kappa_n),$$

where $\mathrm{Re}(\mu_m) > 0$, and one root which corresponds to $d_{-1} = 0$. Clearly, the choice of $d_{-1} = 0$ must correspond to the solution $\omega_0 = \kappa_n + \mathcal{O}(\kappa_n^{2p+2} h^{2p+1})$.

We therefore obtain a total of $(p + 1)N$ spectral values for ω. Since every polynomial solution associated with these spectral values satisfies $\hat{U}_j(1, \omega) = e^{\kappa_n h}\hat{U}_{j-1}(1, \omega)$, and since every $\omega_0, \dots, \omega_p$ is distinct for each κ_n, we have that these solutions are linearly independent on the entire interval I. We can therefore decompose the numerical solution U into these $(p + 1)N$ solutions. □

From this theorem we have that for each κ_n there are $p + 1$ independent polynomial solutions of (2.3.1) which satisfy $\hat{U}_j(1, \omega) = e^{\kappa_n h}\hat{U}_{j-1}(1, \omega)$ for all j. One corresponds to $\omega_0 = \kappa_n + \mathcal{O}(\kappa_n^{2p+2} h^{2p+1})$ and can be seen as 'physical' as it propagates with a numerical frequency which is close to the exact frequency. The other, 'non-physical', solutions are dampened out exponentially quickly. This property of the numerical frequencies of the DG method was conjectured by Guo et al in [38], where the authors explicitly calculated similar expansions of the numerical frequencies ω_m for $p = 1, 2$ and 3.

Now, since the non-physical modes are damped out exponentially quickly we see that after sufficiently long times the accuracy of the numerical solution will be completely determined by the accuracy of the physical mode. Hence, if we specifically choose the initial projection of the exact solution to ensure that the physical mode is high-order accurate, we should preserve this accuracy for $t > 0$. We formalize this observation in the following theorem.

Theorem 2.4 (Global Superconvergence)**.** *Let $u(x, t)$ be a smooth exact solution of* (2.1.1) *on the interval I with periodic boundary conditions. Let U be the numerical solution of the DG scheme* (2.2.6) *on a uniform mesh of N elements and let U_j be the restriction of the numerical solution to the cell I_j. Let $\epsilon_j(\xi, t) = U_j - u_j$ be the numerical error on I_j (mapped to the canonical element $[-1, 1]$). Suppose the projection of the initial profile $u(x, 0)$ into the finite element space is chosen such that*

$$\int_{-1}^{1} [U_j(\xi, 0) - u_j(\xi, 0)]\, P_k(\xi)\, d\xi = \mathcal{O}(h^{2p+1-k}), \quad k = 0, \ldots, p, \tag{2.3.27}$$

is satisfied. Then the error on cell I_j will tend exponentially quickly towards the form

$$\epsilon_j(\xi, t) = [[U_j]]R^{-}_{p+1} + \gamma_{p+2}(t)R^{-,(-1)}_{p+1} + \ldots + \gamma_{2p}(t)R^{-,(1-p)}_{p+1} + \mathcal{O}(h^{2p+1}), \tag{2.3.28}$$

where $\gamma_k(t) = \mathcal{O}(h^k)$ and, in particular,

$$\epsilon_j(1, t) = \mathcal{O}(h^{2p+1}).$$

Proof. We begin by assuming for simplicity that the exact solution can be written as the sum

$$u(x, t) = \sum_{n=0}^{N-1} \hat{u}_n e^{\kappa_n (x-at)},$$

where $\kappa_n = \frac{2\pi n i}{L}$ and L is the length of I. The coefficients $\hat{u}_n$ are found by the discrete Fourier transform and satisfy

$$u(x_j, 0) = \sum_{n=0}^{N-1} \hat{u}_n e^{\kappa_n x_j}, \quad j = 1, \ldots, N.$$

Of course, in general the exact solution cannot be written in such a way but provided u is sufficiently smooth and N is sufficiently large the error in such an approximation should be negligible compared to the error in the polynomial approximation on each cell. Without loss of generality, let us consider the numerical approximation of just one of these Fourier modes, $u(x,t) = \hat{u}_n e^{\kappa_n(x-at)}$. Restricting this Fourier mode to the cell I_j and mapping to the canonical element we see that

$$u_j(\xi,t) = \hat{u}_n e^{\kappa_n(x_j-at)} e^{\frac{\kappa_n h}{2}(\xi+1)}.$$

Since the mesh is uniform, the projection of $u_j(x,0)$ into the finite element space will be of the form $U_j(\xi,0) = \hat{u}_n e^{\kappa_n x_j}\hat{U}(\xi)$ for every j and we immediately obtain that $U_j(1,t) = e^{\kappa_n h}U_{j-1}(1,t)$ for every j. We therefore can express the numerical solution as a sum of the $p+1$ independent polynomial solutions found in Theorem 2.3 that satisfy $U_j(\xi,t) = \hat{U}_j(\xi,\omega_m)e^{-a\omega_m t}$ and $\hat{U}_j(1,\omega_m) = e^{\kappa_n h}\hat{U}_{j-1}(1,\omega_m)$, where the ω_m are the $p+1$ distinct values which satisfy $\frac{f(\omega_m h,\xi)}{g(\omega_m h)} = e^{\kappa_n h}$. Hence

$$U_j(\xi,t) = \sum_{m=0}^{p} C_m e^{\kappa_n x_j - a\omega_m t}\frac{f(\omega_m h,\xi)}{g(\omega_m h)}.$$

Since the physical frequency ω_0 is an accurate approximation of the exact frequency κ_n to order $\mathcal{O}(\kappa_n^{2p+2}h^{2p+1})$, we have by Corollary 2.1 the expansion

$$C_0\frac{f(\omega_0 h,\xi)}{g(\omega_0 h)} = C_0 e^{\frac{\kappa_n h}{2}(\xi+1)} + C_0\frac{(\omega_0 h)^{p+1}}{g(\omega_0 h)}\left[R^-_{p+1}(\xi) + \sum_{k=1}^{\infty}\left(\frac{\omega_0 h}{2}\right)^k R^{-,(-k)}_{p+1}(\xi)\right] + \mathcal{O}((\kappa_n h)^{2p+2}).$$

Therefore performing the initial projection and using this expansion together with the orthogonality of the Radau polynomial R^-_{p+1} we find that

$$\int_{-1}^{1}\left[U_j(\xi,0) - u_j(\xi,0)\right]P_k\,d\xi = \int_{-1}^{1}(C_0 - \hat{u}_n)e^{\kappa_n x_j}e^{\frac{\kappa_n h}{2}(x+1)}P_k\,d\xi + \sum_{m=1}^{p} C_m e^{\kappa_n x_j}\int_{-1}^{1}\frac{f(\omega_m h,\xi)}{g(\omega_m h)}P_k\,d\xi + \mathcal{O}(\kappa_n^{2p+2}h^{2p+1-k}).$$

Thus, the requirement of the initial projection to satisfy (2.3.27) will be satisfied by the choice of $C_0 = \hat{u}_n + \mathcal{O}(h^{2p+1})$ and $\sum_{m=1}^{p} C_m\frac{f(\omega_m h,\xi)}{g(\omega_m h)} = \gamma P_p$, where $\gamma = \mathcal{O}(h^{p+1})$. Hence, this initial projection yields a high-order accurate physical mode of the numerical solution. Finally, we know from Theorem 1 that $\omega_1,\ldots,\omega_p$ have positive real parts of order $\mathcal{O}\left(\frac{1}{h}\right)$. Hence these components of the solution are damped out exponentially quickly in time and the numerical solution tends to the form

$$U_j(\xi,t) = \hat{u}_n e^{\kappa_n x_j - a\omega_0 t}\frac{f(\omega_0 h,\xi)}{g(\omega_0 h)} + \mathcal{O}(h^{2p+1}). \qquad (2.3.29)$$

We can write this solution in the form (2.3.14) to find that

$$
\begin{aligned}
U_j(\xi,t) &= \hat{u}_n e^{\kappa_n x_j + \frac{\omega_0 h}{2}(\xi+1) - a\omega_0 t} + [[U_j]]\left[R^-_{p+1}(\xi) + \sum_{k=1}^{\infty}\left(\frac{\omega_0 h}{2}\right)^k R^{-,(-k)}_{p+1}(\xi)\right] + \mathcal{O}(h^{2p+1}), \\
&= u_j(\xi,t) + [[U_j]]\left[R^-_{p+1}(\xi) + \sum_{k=1}^{\infty}\left(\frac{\omega_0 h}{2}\right)^k R^{-,(-k)}_{p+1}(\xi)\right] + \mathcal{O}(h^{2p+1}).
\end{aligned}
$$

From this, we see from that the error for the numerical approximation has the form

$$
\epsilon_j(\xi,t) = [[U_j]]\left[R^-_{p+1}(\xi) + \sum_{k=1}^{\infty}\left(\frac{\omega_0 h}{2}\right)^k R^{-,(-k)}_{p+1}(\xi)\right] + \mathcal{O}(h^{2p+1}),
$$

and since this is true for any Fourier mode $\hat{u}_n e^{\kappa_n x}$, we obtain the result by summing this expression over all Fourier modes. □

Theorem 2.4 provides conditions for when we will observe the entire numerical solution tending towards a superconvergent form on each cell. This superconvergent form will be one order more accurate at points ξ_0 such that $R^-_{p+1}(\xi_0) = 0$, i.e. the roots of the right Radau polynomial. In the proof of this theorem we see that the key requirement for global superconvergence is that the initial projection projects the intial data onto the physical mode with high-order accuracy. For example, an initial projection which consists of simply interpolating the initial data at equidistant points will not satisfy this condition and thus we do not observe superconvergence of the numerical solution at the downwind points at any time.

Examining the superconvergent form (2.3.28) we can establish some useful corollaries. First, we note that once the non-physical modes have been damped out the remaining physical modes will be advected with order h^{2p+1} accuracy. Hence, for the physical modes, the DG method can be viewed as an order $2p+1$ scheme. Second, since the initial projection in Theorem 2.4 produces a high-order accurate physical mode, and due to the orthogonality properties of the Radau polynomials, we also obtain high-order accuracy of the moments of the numerical solution. We state the results formally below.

Corollary 2.2. *The accumulation error of the superconvergent numerical solution* (2.3.29) *is of order* $2p+1$*. That is, after sufficiently long time the non-physical modes of the numerical solution have been damped out and the numerical solution satisfies*

$$
||U_{j+1}(\xi, t+ah) - U_j(\xi,t)|| = \mathcal{O}(h^{2p+1}).
$$

Corollary 2.3. *The superconvergent form of the numerical solution* (2.3.28) *has the property that the m-th moment of the error is order* $2p+1-m$, *i.e.*

$$\int_{-1}^{1} \epsilon_j(\xi,t) P_m \, d\xi = \mathcal{O}(h^{2p+1-m}).$$

In the next section we perform several numerical test to confirm the results of Theorem 2.4 and Corollaries 2.2 and 2.3.

2.4 Numerical Examples

In this section we will perform several numerical experiments to confirm the superconvergence properties stated in the section above for the DG method for the linear advection equation. Specifically, we will confirm that on a uniform mesh the numerical solution of the DG method with a non-superconvergent initial projection will tend exponentially quickly towards the superconvergent form (2.3.28). Moreover, we will show that when t is sufficiently large the superconvergent numerical solution is advected at order $\mathcal{O}(h^{2p+1})$. Finally, we will show that the moments of the numerical error are also high-order accurate after sufficiently long time t.

Our numerical studies were done on the initial value problem

$$\begin{aligned} u_t + u_x &= 0, \qquad -1 \le x < 1, \qquad t \ge 0, \qquad (2.4.1) \\ u(x,0) &= u_0(x), \\ u(-1,t) &= u(1,t), \end{aligned}$$

with

$$u_0(x) = \sin 4\pi x. \qquad (2.4.2)$$

All tests below are calculated using an RK-4 time-stepping scheme and a CFL number of $\frac{0.15}{2p+1}$ to minimize the error incurred in time integration.

Superconvergence from more general initial projections

In the proof of Theorem 1 we showed that the non-physical waves are damped out like $e^{-\frac{a\mu_m t}{h}}$. We therefore expect to observe that a numerical solution with an initial projection satisfying the conditions of Theorem 2, to have converged to the superconvergent form (2.3.28) when

$$e^{-\frac{a\mu_{min}t}{h}} = \mathcal{O}(h^{2p+1}),$$

where μ_{min} is the non-physical numerical frequency with the smallest real part. Therefore, we expect that the numerical solution will be superconvergent when

$$t = -\frac{2p+1}{a\mu_{min}}\mathcal{O}(h \log h).$$

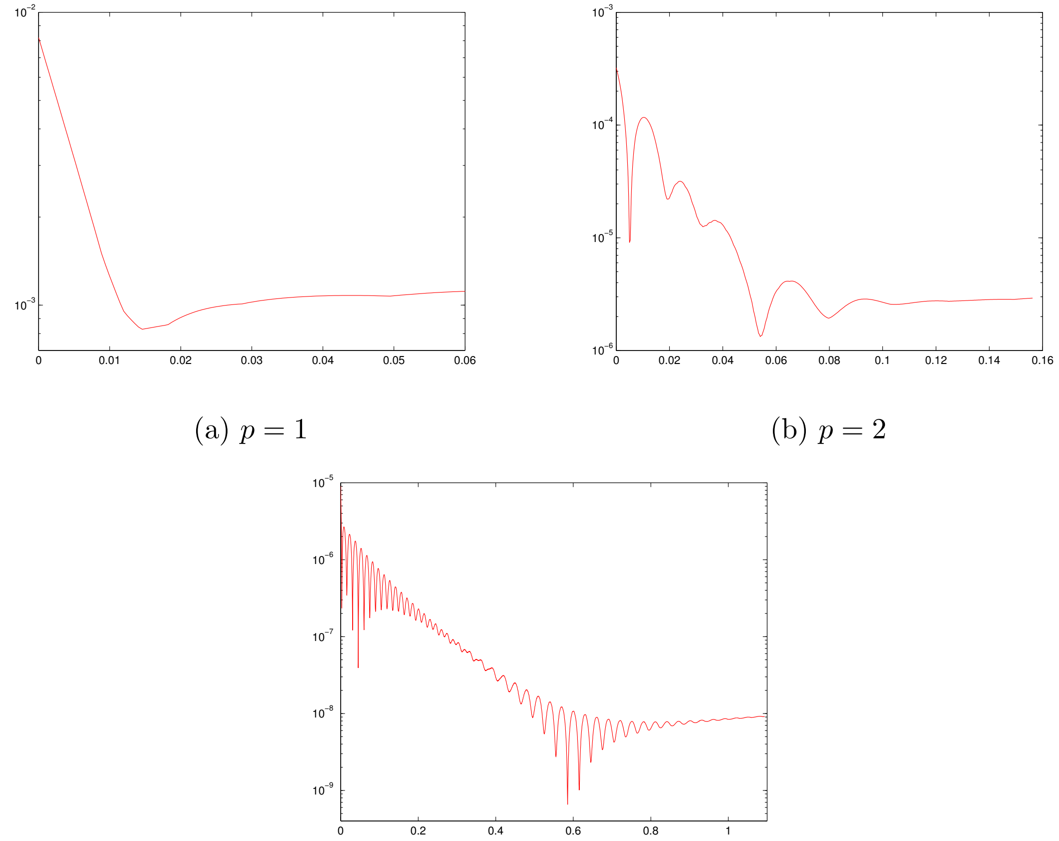
We can estimate the smallest real part of the non-physical numerical frequencies by explicitly calculating the roots of the polynomial $g(z) - f(z, 1)$ and finding the root with the smallest non-zero real part. This calculation for $p = 1, 2, 3$, and 4 yields $\mu_{min} = 6, 3, 0.42$, and 0.058, respectively. Therefore, we see that the smallest real part of the non-physical numerical frequencies is decreasing very rapidly as the order p increases. Hence we expect that it will take significantly longer for the non-physical modes to be damped out as the order of the DG method increases.

In Figure 2.1 we show the error at the downwind point of the numerical solution as a function of time for the $p = 1, 2$, and 3 schemes on a uniform mesh of $N = 64$ elements with the usual L^2 initial projection. The error at the downwind point is calculated using the L^1 norm of the point-wise numerical errors at the downwind points, i.e. $||E|| = h\sum_j |U_j(1,t) - u_j(1,t)|$. We notice from the linear shape of the semi-log plots that the error at the downwind point decays exponentially up to some critical time, at which point the error remains relatively constant. We also notice that due to the scaling of μ_{min} it takes significantly longer for the error at the downwind points to reach this critical time as p increases. In the following numerical test we show that once this critical time is reached the error at the downwind points is $\mathcal{O}(h^{2p+1})$.

In Tables 2.1-2.3 we show the results of our convergence test for $p = 1, 2$, and 3. In each table we present the L^1 errors at the downwind points of the cells $||E||$, and the L^1 norm of the numerical errors in the cell averages, calculated as

$$||\bar{\epsilon}_0|| = h\sum_{j=1}^{N}\left|\int_{-1}^{1} (U_j - u_j)\; d\xi\right|.$$

The errors are calculated at $t = h, 4h$, and $35h$ for the $p = 1, 2$, and 3 methods, respectively, in order to allow sufficient time for the non-physical modes to dampen out. We calculate



(a) $p = 1$

(b) $p = 2$

(c) $p = 3$

Figure 2.1: Semi-log plots of L^1 norm of the point-wise error at the downwind points of the numerical solution with L^2 initial projection as a function of time. Solutions are calculated for the linear advection, (2.4.1)-(2.4.2) on a uniform mesh of $N = 64$ elements, $CFL = \frac{0.05}{2p+1}$.

these errors for two different initial projections. The first is the usual L^2 projection while the second is a left Radau-like projection, which is defined by

$$\int_{-1}^{1} (U_j - u_j) P_k \, d\xi = 0, \quad k = 0, \ldots, p-1,$$

and

$$U_j(-1, 0) = u_j(-1, 0).$$

	L^2 Projection				Left Radau Projection			
N	$\|\|E\|\|$	r	$\|\|\bar{\epsilon}_0\|\|$	r	$\|\|E\|\|$	r	$\|\|\bar{\epsilon}_0\|\|$	r
16	7.02e-02	-	6.66e-02	-	9.63e-02	-	1.22e-01	-
32	8.40e-03	3.06	8.90e-03	2.90	1.22e-02	2.98	1.68e-02	2.86
64	1.04e-03	3.01	1.08e-03	3.04	1.54e-03	2.99	2.13e-03	2.98
128	1.30e-04	3.00	1.34e-04	3.01	1.93e-04	2.99	2.67e-04	3.00
256	1.63e-05	2.99	1.67e-05	3.00	2.43e-05	2.99	3.33e-05	3.00

Table 2.1: Linear advection, (2.4.1)-(2.4.2) with $p = 1$ and with the L^2 and left Radau initial projections. L^1 error of the downwind points $||E||$ and of the cell averages $||\epsilon_0||$ are shown together with convergence rates, r. Errors are calculated at $t = h$.

	L^2 Projection				Left Radau Projection			
N	$\|\|E\|\|$	r	$\|\|\bar{\epsilon}_0\|\|$	r	$\|\|E\|\|$	r	$\|\|\bar{\epsilon}_0\|\|$	r
16	5.87e-03	-	7.96e-03	-	6.65e-03	-	7.66e-03	-
32	1.10e-04	5.72	1.86e-04	5.42	1.38e-04	5.59	2.20e-04	5.12
64	2.74e-06	5.34	4.04e-06	5.52	3.57e-06	5.27	5.54e-06	5.31
128	8.01e-08	5.10	1.10e-07	5.20	1.06e-07	5.07	1.60e-07	5.12
256	2.47e-09	5.01	3.28e-09	5.07	3.31e-09	5.00	4.87e-09	5.03

Table 2.2: Linear advection, (2.4.1)-(2.4.2) with $p = 2$ and with the L^2 and left Radau initial projections. L^1 error of the downwind points $||E||$ and of the cell averages $||\epsilon_0||$ are shown together with convergence rates, r. Errors are calculated at $t = 4h$.

These projections, while satisfying the conditions of Theorem 2, are far from the superconvergent form (2.3.28) which can be viewed as close to a right Radau projection of the exact solution. In each table we observe the expected $2p + 1$ rate of convergence in both the error at the downwind points of the cells and in the cell averages.

Order $2p + 1$ advection of superconvergent solution

Next, we show that once the non-physical modes of the numerical solution have been damped out, the remaining modes are advected at order $2p + 1$. To show this we use the L^2 initial projection and calculate the norm of the difference between numerical solutions

	L^2 Projection				Left Radau Projection			
N	$\|\|E\|\|$	r	$\|\|\bar{\epsilon}_0\|\|$	r	$\|\|E\|\|$	r	$\|\|\bar{\epsilon}_0\|\|$	r
16	5.14e-04	-	1.05e-03	-	5.14e-04	-	1.06e-03	-
32	2.36e-06	7.76	4.39e-06	7.90	2.30e-06	7.80	4.39e-06	7.91
64	9.17e-09	8.00	1.77e-08	7.95	9.09e-09	7.99	1.82e-08	7.91
128	3.63e-11	7.97	6.93e-11	8.00	3.53e-11	8.00	7.13e-11	8.00
256	2.75e-13	7.05	6.53e-13	6.73	2.99e-13	6.88	5.83e-13	6.93

Table 2.3: Linear advection, (2.4.1)-(2.4.2) with $p = 3$ and with the L^2 and left Radau initial projections. L^1 error of the downwind points $||E||$ and of the cell averages $||\epsilon_0||$ are shown together with convergence rates, r. Errors are calculated at $t = 35h$.

N	$\|\|U(x,0) - U(x,2)\|\|$	r	$\|\|U(x,2) - U(x,4)\|\|$	r
16	9.16e-03	-	6.59e-03	-
32	2.34e-03	1.96	8.34e-04	2.98
64	5.90e-04	1.99	1.05e-04	3.00
128	1.48e-04	2.00	1.31e-05	3.00

Table 2.4: Linear advection, (2.4.1)-(2.4.2) with $p = 1$ and L^2 initial projection. L^1 norms of difference in numerical solutions at different times. Differences are measured between U_j initially and at $t = 2$, after one period, then between U_j at $t = 2$ and $t = 4$, after an additional period.

after $0, 1,$ and 2 periods. That is, we calculate these differences as

$$||U(x,0) - U(x,2)|| = h \sum_{j=1}^{N} \int_{-1}^{1} |U_j(\xi, 0) - U_j(\xi, 2)| \; d\xi.$$

In Tables 2.4 and 2.5 we see that that the difference between the numerical solution initially and after one period converges at the usual $p + 1$ rate. This is expected since the non-physical modes of the solution are present initially, and are $\mathcal{O}(h^{p+1})$. However, we also see that that the difference between the numerical solution after one and two periods converges with order $2p + 1$. This shows that once the non-physical modes of the solution have been damped out, the remaining physical modes are advected at order $2p + 1$.

N	$\|\|U(x,0) - U(x,2)\|\|$	r	$\|\|U(x,2) - U(x,4)\|\|$	r
16	2.87e-04	-	1.03e-05	-
32	3.57e-05	3.00	3.24e-07	4.99
64	4.46e-06	3.00	1.01e-08	5.00
128	5.58e-07	3.00	3.17e-10	5.00

Table 2.5: Linear advection, (2.4.1)-(2.4.2) with $p = 2$ and L^2 initial projection. L^1 norms of difference in numerical solutions at different times. Differences are measured between U_j initially and at $t = 2$, after one period, then between U_j at $t = 2$ and $t = 4$, after an additional period.

	L^2 Projection				Left Radau Projection			
N	$\|\|\bar{\epsilon}_1\|\|$	r	$\|\|\bar{\epsilon}_2\|\|$	r	$\|\|\bar{\epsilon}_1\|\|$	r	$\|\|\bar{\epsilon}_2\|\|$	r
16	2.92e-03	-	8.27e-03	-	3.24e-03	-	8.06e-03	-
32	1.12e-04	4.70	1.04e-03	2.99	1.04e-04	4.96	1.04e-03	2.95
64	8.09e-06	3.79	1.29e-04	3.01	7.97e-06	3.70	1.29e-04	3.01
128	5.21e-07	3.96	1.61e-05	3.00	5.19e-07	3.94	1.61e-05	3.00
256	3.28e-08	3.99	2.00e-06	3.00	3.27e-08	3.99	2.01e-06	3.00

Table 2.6: Linear advection, (2.4.1)-(2.4.2) with $p = 2$ and with the L^2 and left Radau initial projections. L^1 norms of the first and second moments of the numerical error are shown together with convergence rates, r. Errors are calculated at $t = 4h$.

Superconvergence of moments

Finally, we demonstrate the high-order accuracy in the moments of the numerical error for $p = 2$ in Table 2.6. We present the L^1 norm of the first and second moments of the numerical error in each cell. The moments are calculated as

$$||\bar{\epsilon}_m|| = h \sum_{j=1}^{N} \left| \int_{-1}^{1} (U_j - u_j) \, P_m \, d\xi \right| .$$

The moments are calculated from the numerical solution using the usual L^2 initial projection and the left Radau-like projection, as above. From this table we see that the m-th moment of the numerical error does indeed achieve the predicted order $2p + 1 - m$ convergence rate.

2.5 Discussion

By finding the Fourier modes of the PDE (2.2.6) that governs the numerical solution we have shown that the polynomial solutions are completely described by the value of the solution at the downwind point of the previous cell and the rational function $\frac{f(\omega h_j,\xi)}{g(\omega h_j)}$. This rational function has a local expansion in h_j in terms of the $(p+1)$-th right Radau polynomial and the anti-derivatives of this polynomial. Furthermore, at the downwind point of the cell, we have that $\frac{f(\omega h_j,\xi)}{g(\omega h_j)}$ is the $\frac{p}{p+1}$ Padé approximant of e^z. As shown in Theorem 2.2, the accuracy of this Padé approximant is what gives rise to the high-order accuracies in both dissipation and dispersion of the DG scheme, known as superaccuracy. Moreover, the expansion of the rational function in terms of the right Radau polynomial and its anti-derivatives is what we observe to be the local superconvergence of the numerical solution at the right Radau points and the order $2p+1$ superconvergence of the downwind point in each cell. Finally, as studied in [48] and shown by equation (2.3.22), the spectrum of the DG discretization matrix is directly related to this rational function. By studying the spectral values of the method, we are able to prove global superconvergence results in Theorem 2.4. These Fourier modes, therefore, provide a direct connection between the three previously disparate properties of superaccuracy, superconvergence, and the stability of the DG method.

We have shown that for a uniform computational mesh of N elements there exist N polynomial solutions that can be viewed as physical components of the numerical wave and pN polynomial solutions that are non-physical components. Moreover, these non-physical solutions are damped out exponentially quickly in time and, therefore, neglecting time integration errors, we can conclude that the accuracy of the numerical solution for sufficiently large times is completely determined by the accuracy of the initial projection of the exact solution onto the physical modes. Beyond this point, the DG scheme can be viewed as order $2p+1$ accurate on these physical solutions. Using this result, we proved that for a class of initial projections of the exact solution we expect to obtain a numerical solution which is superconvergent at both the roots of the right Radau polynomial and the downwind points of the cell, after sufficiently long times. In particular, there is a class of initial projections which do not initially have order h^{2p+1} accuracy at the downwind point, but will obtain this order of accuracy after sufficient time has elapsed. For these projections the points of superconvergence will migrate to the roots of the right Radau polynomial exponentially quickly in time.

Since many properties of the DG scheme are connected to the accuracy of this rational function, we are motivated to consider how these properties may be manipulated through modifications to the scheme. We explore this idea in the next chapter where we propose modifications to the DG method in order to relax its stability restriction. We extend the analysis from this chapter to this modified scheme in order to study what effects the modifications have on the superconvergence and superaccuracy properties of the method.

Chapter 3

The Modified Discontinuous Galerkin Method

3.1 Introduction

It is well known that the DG method applied to convection problems has maximum a CFL number that decreases with the order of approximation p as (approximately) $1/(2p+1)$ when paired with an appropriate order explicit Runge-Kutta scheme. This rather restrictive condition is caused by the growth of the spectrum of the spatial discretization operator of the semi-discrete scheme, which increases slightly slower than $\mathcal{O}(p^2)$ [48]. In contrast, finite difference schemes have a stability restriction that grows with the size of the computational stencil as $\mathcal{O}(p)$. This makes the DG method a more expensive scheme for the same theoretical order of convergence and is often quoted as one of the shortcomings of the DG scheme. A possible solution to this issue was proposed by Warburton and Hagstrom in [67], in which the authors propose the use of a co-volume mesh which allows an order independent CFL number. However, this method is limited to structured grids and requires mappings of the solution between the original and co-volume meshes. The method in [67] shrinks the spectrum of the DG method so that it does not require the usual $1/(2p+1)$ scaling. Another approach is to devise explicit time-integrators with larger absolute stability regions or stability regions which better encapsulate the spectrum of the DG spatial operator [57, 58, 65]. For Runge-Kutta methods this usually comes at the cost of additional stages.

For the same theoretical order of convergence, numerical schemes can have distinctly different global accuracy. It has been pointed out that the discontinuous Galerkin scheme is more accurate than the finite volume scheme, e.g. when applied to the two-dimensional Euler equations [56], in terms of the $\mathcal{L}^2$ norm. One reason for this is the small dispersive and dissipative errors in the DG method, as discussed in Chapter 2. These small errors lead to slower accumulation of the numerical error which is especially noticeable for long time calculations. Since the superaccuracy and superconvergence properties can be seen as arising from the accuracy of a certain rational function $\frac{f(\omega h_j,\xi)}{g(\omega h_j)}$ it is reasonable to assume then that a scheme resulting in a different rational approximation of $\exp(z)$ may have desirable properties, e.g. a less restrictive CFL number. The difficulty is to modify the weak DG form to obtain such a scheme.

In this chapter, we propose modifications to the DG method which involve $p+1$ parameters $\alpha_k,\ k = 0, 1, \ldots, p$, which we call flux multipliers. In the case when $\alpha_k = 1$, for $k = 0, 1, \ldots, p$, we recover the original DG scheme. When a certain α_k is not equal to one we refer to this multiplier as 'modified'. In each equation evolving the k-th degree of freedom on element I_j, c_{jk}, in time (see (3.2.3)), we use the flux multiplier α_k to scale the contribution from the jumps in the numerical flux at cell interfaces to the propagation of c_{jk}. The justification of this operation is that the weak DG formulation consists of integrals over cell volumes plus contributions from jumps in the numerical flux at the cell boundaries. For solutions which belong to the finite element space, the flux jumps are equal to zero and, thus, the proposed modifications will not influence the solution accuracy. More generally, they will not affect the formal results on accuracy and convergence originally established by Cockburn and Shu [29, 27], as long as the equation corresponding to the c_{j0} coefficient (i.e., the one corresponding to the constant basis function) is unchanged. We show that the modifications will affect the eigenvalues of the spatial operator of the semi-discrete scheme, and hence, the CFL number.

In order to relax the time step restriction of the standard DG formulation, we search for a set of flux multipliers α_k that provides the largest increase in the CFL number when using the Legendre polynomial basis. The values for any other polynomial basis could be obtain from the presented ones by a simple transformation. In order to compute this set of values, we use linear algebra software to search for α_k so that the size of the spectrum of the modified scheme is smaller than that of the original DG method. We find that for the orders of approximation considered in this work, the CFL number can be improved by a factor of two or more by modifying only the highest multiplier to be $\alpha_p \approx 0.4$. The modification of more than the highest multiplier generally leads to a larger improvement in the CFL for particular combinations of α_k. Using an energy argument we prove that when only the highest multiplier , α_p, is modified the semi-discrete scheme is linearly stable. In this case small modifications to α_p influence only the size of the spectrum. In a general case where more than one multiplier is modified, a particular choice of multipliers can result in an unstable semi-discrete scheme. However, we are able to numerically find a combination of multipliers which results in stable semi-discrete schemes.

Next, we analyse the superconvergence and superaccuracy of modified schemes. We show that the modifications directly affect the local superconvergence properties. In particular, modifying m highest order multipliers lowers the order of accuracy at the downwind point of the cell by m orders. We also show that the same modifications lower the orders of accuracy in dissipation and dispersion to $\mathcal{O}(h^{2p+2-m})$ and $\mathcal{O}(h^{2p+1-m})$, respectively. Nevertheless, the order of convergence of the scheme in the $\mathcal{L}^1$ norm remains the same regardless of the number of multipliers changed, as long as α_0 remains equal to one. This follows from the standard DG analysis [29, 27], and our numerical experiments. However, we observe in numerical experiments that the magnitude of the global $\mathcal{L}^1$ error increases due to larger dissipative and dispersive errors. In particular, setting a larger number of multipliers to be not equal to one leads to a larger global error.

The proposed schemes can be viewed from a different perspective. Instead of comparing the schemes based on the size of spatial discretization, we can compare them based on the computational effort. That is, instead of increasing the time step size for a fixed mesh, we can fix the time step and proportionally increase the number of cells. We show that with the modified DG scheme, the solution for the same computational effort is noticeably more accurate in terms of the global error. This is especially advantageous for problems which have high frequency waves or fine structures.

The remainder of this chapter is organized as follows: In Section 2 we will introduce a modification of the discontinuous Galerkin method through the introduction of the flux multipliers α_k. We will then prove several superconvergence and superaccuracy results concerning the effects of these multipliers on the accuracy of the DG scheme by using the linear advection equation as a model problem. We will then investigate the stability of the modified scheme and show that we are able to ameliorate the usual stability restriction of the classical DG scheme through suitable choices in the multipliers α_k. We will conclude by showing that the modified scheme preserves the usual order of convergence in the $\mathcal{L}^1$ norm, and we will show how the scheme performs on several test examples including the linear advection equation and the Euler equations. We also give examples where the accuracies of the DG and modified DG schemes are compared on different sized meshes, but equal computation times.

3.2 Modified Discontinuous Galerkin Discretization

We consider again the application of the DG scheme to the one-dimensional scalar conservation law (1.2.1), as written in (1.2.10)

$$\frac{h_j}{2k+1}\frac{dc_{jk}}{dt} = -\left[f(U^*_{j+1}) - (-1)^k f(U^*_j)\right] + \int_{-1}^{1} f(U_j)P'_k\, d\xi.$$

Integrating the integral term by parts we can write,

$$\frac{h_j}{2k+1}\frac{dc_{jk}}{dt} = -[[f(U^*_{j+1})]] - (-1)^k[[f(U^*_j)]] - \int_{-1}^{1} f(U_j)_\xi P_k\, d\xi. \tag{3.2.1}$$

where $[[f(U^*_{j+1})]] = f(U^*_{j+1}) - f(U_j(x_{j+1}))$ and $[[f(U^*_j)]] = f(U_j(x_j)) - f(U^*_j)$ are the jumps between the local value of the flux $f(U_j)$ and the Riemann fluxes at each boundary. Notice that this expression of the scheme can be obtained directly from (1.2.2) by noticing that to include the contributions from the boundary of the cell we have defined $f(U_j)_x$ as a distribution

$$f(U_j)_x = \begin{cases} [[f(U^*_j)]]\delta_{x_j}, & x = x_j \\ f(U_j)_x, & x \in (x_j, x_{j+1}) \\ [[f(U^*_{j+1})]]\delta_{x_{j+1}}, & x = x_{j+1} \end{cases} \tag{3.2.2}$$

where δ_{x_j} is the Dirac delta function at $x = x_j$. The derivative on the interior term is defined classically since U_j is smooth inside I_j.

Notice that the contributions from neighboring cells are concentrated in the two jump terms on the right hand side of (3.2.1), while the integral term is purely local to the cell I_j. Moreover, when the exact solution of (1.2.1) belongs to the finite element space, the two jump terms at the cell boundaries will be equal to zero. Consequently, modifying these terms will not affect the formal accuracy of the solution. This motivates us to consider a modified version of (3.2.1),

$$\frac{h_j}{2k+1}\frac{dc_{jk}}{dt} = -\alpha_k[[f(U^*_{j+1})]] - (-1)^k\alpha_k[[f(U^*_j)]] - \int_{-1}^{1} f(U_j)_\xi P_k\, d\xi. \tag{3.2.3}$$

Here we have introduced the parameters $\alpha_k, k = 0, \ldots, p$, which scale the contributions of the flux discontinuities at the cell interfaces to the propagation of the solution coefficients. This modification can viewed as altering how the δ-functions in the distribution (3.2.2) are projected into the finite element space. Note that when $\alpha_k = 1, \forall k$, we recover the original DG scheme.

Remark 1 (Formal order of Convergence)**.** *We remark that the proposed modified DG scheme should preserve the usual formal order $p+1$ convergence on smooth solutions. Firstly, the modifications to the numerical flux do not affect consistency of the scheme. This is because on smooth solutions the jump term is zero and will not contribute and therefore the numerical flux remains consistent with the exact flux function $f(u)$. Hence the original results established by Cockburn and Shu in [29, 27] on the $(p+1)$-th order consistency of the DG method will carry over to this modified scheme. We can therefore conclude by the equivalence theorem of Lax-Richmeyer that the modified scheme will preserve the usual $p+1$ convergence rate for linear equations, provided the scheme is linearly stable.*

For nonlinear equations the proof of the TVDM property presented in [29] can be verbatim applied to the modified scheme provided $\alpha_0 = 1$. In particular, Lemma 2.1 uses only the equation for the c_{j0} and the values of the solution at the endpoints of the interval. Since the equation c_{j0} is unmodified, and the endpoint values are limited in the same manner, the lemma holds. Moreover Lemma 2.3 in [29] will also hold with $p = 1$ and the minmod limiter. Hence the modified scheme preserves the usual order $p+1$ convergence for smooth nonlinear problems provided it is stable and $\alpha_0 = 1$.

In Section 3.4 we prove linear stability of the modified scheme in the case where only the highest order multiplier is taken not equal to one. However, when more multipliers are modified the scheme may not be linearly stable. In these cases we investigate stability by plotting the spectrum of the spatial operator of the modified DG scheme.

We expect that this scheme, which we will refer to as the modified DG (mDG) scheme, will perform similarly to the original DG on smooth solutions, where the altered jump contributions are small. In the remainder of this chapter we will be interested in establishing what effect these parameters will have on the numerical scheme. Since this analysis is difficult to perform on the general formulation, we will again consider the simple problem of the linear advection equation.

3.3 Fourier Analysis

Applying the modified DG discretization (3.2.3) to the linear advection equation (2.1.1), and again using the upwind flux $U^*_{j+1} = U_j(x_{j+1})$, we obtain

$$\frac{h_j}{2k+1}\frac{dc_{jk}}{dt} = -(-1)^k \alpha_k a[[U_j]] - a\int_{-1}^{1} \frac{\partial U_j}{\partial \xi} P_k \, d\xi, \tag{3.3.1}$$

where $[[U_j]] = U_j(x_j) - U_{j-1}(x_j)$. To extend the analysis performed in chapter 2, we again obtain a PDE which governs the numerical solution itself. Following the same procedure, we can recover a PDE for U_j by multiplying (3.3.1) by P_k and summing over $k = 0, \dots, p$ to obtain, after some rearrangement,

$$\frac{\partial}{\partial t}U_j + \frac{2a}{h_j}\frac{\partial}{\partial \xi}U_j = -\frac{a}{h_j}[[U_j]]\left(\sum_{k=0}^{p}(-1)^k(2k+1)\alpha_k P_k\right). \tag{3.3.2}$$

Before we begin finding the polynomial solutions of (3.3.2) let us first introduce a new polynomial $\tilde{R}_{p+1}(\xi; \boldsymbol{\alpha})$ which depends on all the flux multipliers, $\boldsymbol{\alpha} = (\alpha_0, \dots, \alpha_p)$, and is defined by

$$\frac{d}{d\xi}\tilde{R}_{p+1}(\xi; \boldsymbol{\alpha}) = \frac{1}{2}\sum_{k=0}^{p}(-1)^{k+1}(2k+1)\alpha_k P_k, \tag{3.3.3a}$$

$$\tilde{R}_{p+1}(-1; \boldsymbol{\alpha}) = 1, \tag{3.3.3b}$$

and let us establish some properties of this polynomial which will be useful later.

Proposition 3.1. *Assume $\alpha_0 = 1$ and let α_m be the lowest order multiplier (smallest m) for which $\alpha_m \neq 1$. Then the polynomial $\tilde{R}_{p+1}(\xi; \boldsymbol{\alpha})$, defined in* (3.3.3)*, satisfies*

$$\tilde{R}_{p+1}(1; \boldsymbol{\alpha}) = 0 \tag{3.3.4}$$

and

$$\int_{-1}^{1}\tilde{R}_{p+1}(\xi; \boldsymbol{\alpha})P_k \, d\xi = \frac{(-1)^{k+1}}{2k+1}(\alpha_{k+1} - \alpha_{k-1}), \quad k = 0, \dots, p-1. \tag{3.3.5}$$

Hence, since $\alpha_k = 1$ for all $k < m$, $\tilde{R}_{p+1}(\xi; \boldsymbol{\alpha})$ is orthogonal to all polynomials of degree not exceeding $m - 1$.

Proof. From the assumption $\alpha_0 = 1$, we can obtain (3.3.4) immediately by integrating (3.3.3a) from -1 to 1 and using the orthogonality property of the Legendre polynomials. Next, we use the property that Legendre polynomials satisfy $(2k+1)P_k = \frac{d}{d\xi}[P_{k+1} - P_{k-1}]$ for $k \geq 0$ [1] (where we have chosen $P_{-1} \equiv 1$) in order to write (3.3.3a) as

$$\frac{d}{d\xi}\tilde{R}_{p+1}(\xi; \boldsymbol{\alpha}) = \frac{1}{2}\sum_{k=0}^{p}(-1)^{k+1}\alpha_k \frac{d}{d\xi}[P_{k+1} - P_{k-1}].$$

From this we find

$$\tilde{R}_{p+1}(\xi; \boldsymbol{\alpha}) = \frac{1}{2}\sum_{k=0}^{p}(-1)^{k+1}\alpha_k[P_{k+1} - P_{k-1}]. \tag{3.3.6}$$

Reindexing this sum we can write

$$\tilde{R}_{p+1}(\xi;\boldsymbol{\alpha}) = \frac{(-1)^{p+1}}{2}\left[\alpha_p P_{p+1} - \alpha_{p-1}P_p\right] + \frac{1}{2}\sum_{k=1}^{p-1}(-1)^{k+1}(\alpha_{k+1}-\alpha_{k-1})P_k - \frac{1}{2}(\alpha_1 - 1). \tag{3.3.7}$$

We can then establish relation (3.3.5) by multiplying (3.3.7) by P_k, $k = 0, \ldots, p-1$, and integrating over $[-1, 1]$. □

Using the definition of the polynomial $\tilde{R}_{p+1}(\xi;\boldsymbol{\alpha})$ we can write the PDE (3.3.2) for the numerical solution U_j as

$$\frac{\partial}{\partial t}U_j + \frac{2a}{h_j}\frac{\partial}{\partial \xi}U_j = \frac{2a}{h_j}[[U_j]]\frac{d}{d\xi}\tilde{R}_{p+1}(\xi;\boldsymbol{\alpha}). \tag{3.3.8}$$

Now, we continue as in Chapter 2 and find polynomial solutions of (3.3.8) by looking at a single Fourier mode solution of the form $U_j(\xi,t) = \hat{U}_j(\xi,\omega)e^{-a\omega t}$, where $\hat{U}_j$ is a polynomial of degree p in ξ. Using these assumptions on the form of $U_j(\xi,t)$ in (3.3.8) we have that this Fourier mode satisfies the ODE

$$-a\omega\hat{U}_j + \frac{2a}{h_j}\frac{\partial}{\partial \xi}\hat{U}_j = \frac{2a}{h_j}[[\hat{U}_j]]\frac{d}{d\xi}\tilde{R}_{p+1}(\xi;\boldsymbol{\alpha}). \tag{3.3.9}$$

We then extend the derivations of the polynomial solutions and their properties from Section 2.3 to (3.3.9), the only difference being that the Radau polynomial $R^-_{p+1}(\xi)$ is replaced by the modified polynomial $\tilde{R}_{p+1}(\xi;\boldsymbol{\alpha})$. Hence, we can state that polynomial solutions of (3.3.9) have the form

$$\hat{U}_j(\xi,\omega) = \hat{U}_{j-1}(1,\omega)\frac{\tilde{f}(\omega h_j,\xi)}{\tilde{g}(\omega h_j)}, \tag{3.3.10}$$

where $\tilde{f}(\omega h_j,\xi)$ and $\tilde{g}(\omega h_j)$ are degree p and $p+1$ polynomials, respectively, and are defined using the polynomial $\tilde{R}_{p+1}(\xi;\boldsymbol{\alpha})$ as

$$\tilde{g}(\omega h_j) = (\omega h_j)^{p+1} + \sum_{k=1}^{p+1} 2^k(\omega h_j)^{p+1-k}\frac{d^k}{d\xi^k}\tilde{R}_{p+1}(-1;\boldsymbol{\alpha}), \tag{3.3.11}$$

$$\tilde{f}(\omega h_j,\xi) = \sum_{k=1}^{p+1} 2^k(\omega h_j)^{p+1-k}\frac{d^k}{d\xi^k}\tilde{R}_{p+1}(\xi;\boldsymbol{\alpha}). \tag{3.3.12}$$

Moreover, the polynomial solutions (3.3.10) have the local expansion in ωh_j as

$$\hat{U}_j(\xi,\omega) = \hat{U}_{j-1}(1,\omega)e^{\frac{\omega h_j}{2}(\xi+1)} + [[\hat{U}_j]]\left[\tilde{R}_{p+1}(\xi;\boldsymbol{\alpha}) + \sum_{k=1}^{\infty}\left(\frac{\omega h_j}{2}\right)^k \tilde{R}_{p+1}^{(-k)}(\xi;\boldsymbol{\alpha})\right], \quad (3.3.13)$$

where $\tilde{R}_{p+1}^{(-k)}(\xi;\boldsymbol{\alpha})$ are the successive anti-derivatives of $\tilde{R}_{p+1}(\xi;\boldsymbol{\alpha})$ and can be written using the Cauchy integration formula as

$$\tilde{R}_{p+1}^{(-k)}(\xi;\boldsymbol{\alpha}) = \frac{1}{(k-1)!}\int_{-1}^{\xi}(\xi-s)^{k-1}\tilde{R}_{p+1}(s;\boldsymbol{\alpha})\,ds, \quad (3.3.14)$$

and $\tilde{R}_{p+1}^{(0)}(\xi;\boldsymbol{\alpha}) = \tilde{R}_{p+1}(\xi;\boldsymbol{\alpha})$.

Again we see that the polynomial solutions on each cell are completely determined by a rational function $\frac{\tilde{f}(\omega h_j,\xi)}{\tilde{g}(\omega h_j)}$ and the value of the numerical solution at the downwind point of the previous cell. Using the expansion (3.3.13) we can establish an analogous result to Corollary 2.1 concerning the properties of the rational function $\frac{\tilde{f}(\omega h_j,\xi)}{\tilde{g}(\omega h_j)}$.

Lemma 3.1. *The rational function* $\frac{\tilde{f}(\omega h_j,\xi)}{\tilde{g}(\omega h_j)}$ *has the expansion*

$$\frac{\tilde{f}(\omega h_j,\xi)}{\tilde{g}(\omega h_j)} = e^{\frac{\omega h_j}{2}(\xi+1)} - \frac{(\omega h_j)^{p+1}}{\tilde{g}(\omega h_j)}\left[\tilde{R}_{p+1}(\xi) + \sum_{k=1}^{\infty}\left(\frac{\omega h_j}{2}\right)^k \tilde{R}_{p+1}^{(-k)}(\xi)\right] \quad (3.3.15)$$

and in particular,

$$\frac{\tilde{f}(\omega h_j,1)}{\tilde{g}(\omega h_j)} = e^{\omega h_j} + \mathcal{O}((\omega h_j)^{p+1+m}), \quad (3.3.16)$$

i.e. $\frac{\tilde{f}(\omega h_j,1)}{\tilde{g}(\omega h_j)}$ *is an order* $p+1+m$ *rational approximation of the exponential function* $e^{\omega h_j}$*, where* m *is the index of the lowest order multiplier (smallest* m*) for which* $\alpha_m \neq 1$.

Proof. The expansion (3.3.15) is derived in an entirely analogous way as (2.3.11) was derived in Section 2.3, and is therefore omitted for brevity. To establish (3.3.16), however, we first note that from Proposition 3.1 we have that $\tilde{R}_{p+1}(1;\boldsymbol{\alpha}) = 0$ and $\tilde{R}_{p+1}(\xi;\boldsymbol{\alpha})$ is orthogonal to every polynomial of degree less than $m-1$. Hence, from the definition of the polynomials $\tilde{R}_{p+1}^{(-k)}(\xi;\boldsymbol{\alpha})$ in (3.3.14) we have that $\tilde{R}_{p+1}^{(-k)}(1;\boldsymbol{\alpha}) = 0$ for $k \leq m-1$. Hence, evaluating (3.3.15) at $\xi = 1$, we have that

$$\frac{\tilde{f}(\omega h_j,1)}{\tilde{g}(\omega h_j)} = e^{\frac{\omega h_j}{2}(\xi+1)} - \frac{(\omega h_j)^{p+1}}{\tilde{g}(\omega h_j)}\left[\sum_{k=m}^{\infty}\left(\frac{\omega h_j}{2}\right)^k \tilde{R}_{p+1}^{(-k)}(\xi)\right]$$

which yields

$$\frac{\tilde{f}(\omega h_j, 1)}{\tilde{g}(\omega h_j)} = e^{\omega h_j} + \mathcal{O}((\omega h_j)^{p+1+m}).$$

and establishes the result. □

This lemma concerning the local accuracy of the rational function $\frac{\tilde{f}(\omega h_j,\xi)}{\tilde{g}(\omega h_j)}$ allows us to extend our results from Section 2.3 on the local superconvergence and superaccuracy of the DG method to this modified scheme. We state these results in the following theorems.

Theorem 3.1 (Local Superconvergence)**.** *Let $u(x,t)$ be a smooth exact solution of* (2.1.1) *on the interval I with suitable boundary conditions. Let U be the numerical solution of the modified DG scheme* (3.3.1) *on a mesh of N elements and let U_j be the restriction of the numerical solution to the cell I_j. Let $\epsilon_j(\xi,t) = U_j - u_j$ be the numerical error on I_j (mapped to the canonical element $[-1,1]$). Suppose the inflow $U_{j-1}(x_j,t)$ into cell I_j is exact, i.e. $U_{j-1}(x_j,t) = u(x_j,t)$. Then the numerical error on cell I_j satisfies*

$$\epsilon_j(\xi,t) = [[U_j]]\tilde{R}_{p+1}(\xi;\boldsymbol{\alpha}) + \mathcal{O}(h_j^{p+2}),$$

and

$$\epsilon_j(1,t) = \mathcal{O}(h_j^{p+1+m}). \tag{3.3.17}$$

where m is the index of the lowest order multiplier (smallest m) for which $\alpha_m \neq 1$.

Theorem 3.2 (Superaccuracy)**.** *The numerical dispersion relation of the mDG scheme applied to the linear equation* (2.1.1) *on cell I_j between a frequency ω and the numerical wavenumber $\tilde{\kappa}$ can be written,*

$$\frac{\tilde{f}(\omega h_j, 1)}{\tilde{g}(\omega h_j)} = e^{\tilde{\kappa} h_j}. \tag{3.3.18}$$

The numerical wavenumber then satisfies

$$\tilde{\kappa} = \omega + C_1 \omega^{p+m+1} h_j^{p+m} + C_2 \omega^{p+m+2} h_j^{p+m+1} + \ldots, \tag{3.3.19}$$

where m is the index of the lowest order multiplier (smallest m) for which $\alpha_m \neq 1$. Therefore if $p+m+1$ is odd then the order of the dispersion error of the modified DG scheme is $p+m+1$ and the order of the dissipation error is $p+m$. On the other hand, if $p+m+1$ is even then the order of the dissipation error of the modified DG scheme is $p+m+1$ and the order of the dispersion error is $p+m$.

From Theorem 3.1 we see that, much like the classical DG scheme, the leading order local error of the scheme is the polynomial $\tilde{R}_{p+1}(\xi; \boldsymbol{\alpha})$, which is now directly altered by the parameters α_k, multiplied by the jump at the inflow boundary $[[U_j]]$. Furthermore, we see in (3.3.17) and in the results of Theorem 3.2 that as more flux multipliers are modified the orders of the local superconvergence at the downwind point and the superaccuracies in dissipation and dispersion errors are reduced. This is somewhat expected by recalling that the forcing term in the PDE (2.2.6) for the numerical solution can be seen as resulting from the projection of the δ-functions in $f(U)_x$ at the boundaries of the cell (see (3.2.2)) into the finite element space. The modification of this projection, which produces the forcing term in (3.3.8) for the modified scheme, can be viewed as introducing an error into the usual projection of the δ-function. We will see in Section 3.4, however, that although these modifications reduce the formal orders of accuracy of the DG scheme, they can potentially enable us to choose larger CFL numbers.

We continue the extension of the analysis presented in Chapter 2 to the modified DG scheme by establishing analogous results to Theorems 2.3 and 2.4, which concern the spectrum and global superconvergence of the method on uniform grids. To do this we again consider a uniform mesh and periodic boundary conditions and note that, as a consequence, the spectral values of the scheme must satisfy

$$\frac{\tilde{f}(\omega h, 1)}{\tilde{g}(\omega h)} = e^{\kappa_n h}, \tag{3.3.20}$$

where $e^{\kappa_n h}$ is an N-th root of unity, i.e. $\kappa_n = \frac{2\pi n i}{L}$ where L is the length of the domain I. We use this together with the results of Lemma 3.1 in order to establish an analogous result to Theorem 2.3 for the modified scheme.

Theorem 3.3 (Physical Spectrum)**.** *Let U be the numerical solution of the modified DG scheme* (3.3.1) *on a uniform mesh of N elements on the interval I with periodic boundary conditions, and let U_j be the restriction of the numerical solution to the cell I_j.*

The numerical solution U can be decomposed into $(p+1)N$ solutions. Each of these solutions is polynomial in ξ and has the form $U_j(\xi, t) = \hat{U}_j(\xi, \omega)e^{-a\omega t}$. These solutions also satisfy $\hat{U}_j(1, \omega) = e^{\kappa_n h}\hat{U}_{j-1}(1, \omega)$ for each j where $\kappa_n = \frac{2\pi n i}{L}$, $n = 0, \ldots, N-1$. Corresponding to each κ_n there are $p+1$ spectral values $\omega = \omega_0, \omega_1, \ldots, \omega_p$ which have the expansions

$$\omega_0 = \kappa_n + \mathcal{O}(\kappa_n^{p+1+m} h^{p+m})$$

and

$$\omega_l = \frac{\mu_l}{h} + \mathcal{O}(\kappa_n), \qquad l = 1, \ldots, p,$$

where m is the index of the lowest order multiplier (smallest m) for which $\alpha_m \neq 1$ and μ_l are the p non-zero roots of the polynomial $\tilde{g}(z) - \tilde{f}(z,1)$.

We note that the only difference in the proof of this theorem from the proof of Theorem 2.3 occurs due to the reduced accuracy of the rational approximation $\frac{\tilde{f}(\omega h_j,1)}{\tilde{g}(\omega h_j)}$. The immediate consequence is that the order of accuracy of the physical mode for the modified scheme will be lowered to $p+m$ where α_m is the lowest order multiplier for which $\alpha_m \neq 1$. We also note that Theorem 2.3 also states that $\mathrm{Re}(\mu_l) > 0$, for all l, the proof of which used a known property of the Padé approximant. This, however, cannot be guaranteed in general for the modified scheme. Therefore, to enforce that $\mathrm{Re}(\mu_l) > 0$ for all l we must make the additional assumption that the multipliers α_k are chosen such that the non-zero roots of $\tilde{g}(z) - \tilde{f}(z,1)$ lie in the right-half complex plane. A sufficient, though not explicitly necessary, condition is to enforce that the modified DG discretization remains stable. Under this condition we obtain that we can decompose the spectrum of the scheme into N physical modes, $\omega_0 = \kappa_n + \mathcal{O}(h^{p+m})$, and pN non-physical modes with positive real parts of order $\mathcal{O}(\frac{1}{h})$.

Using this partitioning of the spectrum we finally extend the results of Theorem 2.4 to this modified DG method, making the appropriate changes to the order of accuracy of the physical modes, in order to establish a global superconvergence result for the modified DG method.

Theorem 3.4 (Global Superconvergence)**.** *Let $u(x,t)$ be a smooth exact solution of* (2.1.1) *on the interval I with periodic boundary conditions. Let U be the numerical solution of a modified DG scheme* (3.2.3) *on a uniform mesh of N elements, where the modifiers, α_k, are chosen so that the scheme is stable and $\alpha_0 = 1$. Let m be the smallest index for which $\alpha_m \neq 1$. Let U_j be the restriction of the numerical solution to the cell I_j and let $\epsilon_j(\xi,t) = U_j - u_j$ be the numerical error on I_j (mapped to the canonical element $[-1,1]$). Suppose the projection of the initial profile $u(x,0)$ into the finite element space is chosen such that*

$$\int_{-1}^{1} \left[U_j(\xi,0) - u_j(\xi,0)\right] P_k(\xi)\, d\xi = \mathcal{O}(h^{p+m-k}), \quad k = 0,\ldots,p, \tag{3.3.21}$$

is satisfied. Then the error on cell I_j will tend exponentially quickly towards the form

$$\epsilon_j(\xi,t) = [[U_j]]\tilde{R}_{p+1}(\xi;\boldsymbol{\alpha}) + \gamma_{p+2}(t)\tilde{R}_{p+1}^{(-1)}(\xi;\boldsymbol{\alpha}) + \ldots + \gamma_{p+m-1}(t)\tilde{R}_{p+1}^{(2-m)}(\xi;\boldsymbol{\alpha}) + \mathcal{O}(h^{p+m}), \tag{3.3.22}$$

where $\gamma_k(t) = \mathcal{O}(h^k)$ and, in particular,

$$\epsilon_j(1,t) = \mathcal{O}(h^{p+m}).$$

Proof. The proof of this theorem follows an analogous argument as the proof of Theorem 2.4 in Chapter 2. We again consider an exact solution which consists of only a single discrete Fourier mode, $u(x,t) = \hat{u}_n e^{\kappa_n(x-at)}$, where $\kappa_n = \frac{2\pi n i}{L}$ and L is the length of I. We then argue that the numerical solution on each cell is then a sum of the $p+1$ independent polynomial solutions found in Theorem 1 associated with the wavenumber κ_n, i.e.

$$U_j(\xi,t) = \sum_{l=0}^{p} C_l e^{\kappa_n x_j - a\omega_l t} \frac{\tilde{f}(\omega_l h, \xi)}{\tilde{g}(\omega_l h)}.$$

Next, from the fact that the physical mode has the local expansion (3.3.13) we can conclude that the initial projection (3.3.21) will guarantee that $C_0 = \hat{u}_n + \mathcal{O}(h^{p+m})$. Then, since the modified scheme is assumed to be stable, we have from Theorem 3.3 that the non-physical modes $\omega_1, \ldots, \omega_p$ will have positive real parts of order $\mathcal{O}\left(\frac{1}{h}\right)$ and, hence, will be damped out exponentially quickly and the numerical solution will tend to the form

$$U_j(\xi,t) = \hat{u}_n e^{\kappa_n x_j - a\omega_0 t} \frac{\tilde{f}(\omega_0 h, \xi)}{\tilde{g}(\omega_0 h)} + \mathcal{O}(h^{p+m}).$$

Therefore, using the expansion (3.3.13) and the accuracy of the physical mode ω_0 to the exact wavenumber κ_n we obtain that

$$U_j(\xi,t) = u_j(\xi,t) + [[U_j]] \left[\tilde{R}_{p+1}(\xi;\boldsymbol{\alpha}) + \sum_{k=1}^{\infty} \left(\frac{\omega_0 h}{2}\right)^k \tilde{R}_{p+1}^{(-k)}(\xi;\boldsymbol{\alpha}) \right] + \mathcal{O}(h^{p+m}),$$

which, upon summing over all possible Fourier modes, yields the result. □

This theorem tell us under what conditions we will observe the leading error of the numerical solution tending to the form $[[U_j]]\tilde{R}_{p+1}(\xi;\boldsymbol{\alpha})$. When this occurs, the numerical solution will be superconvergent at the roots of $R_{p+1}(\xi;\boldsymbol{\alpha})$. Note, however, that the accumulation error of the method is order $p+m$, where m is lowest index for which $\alpha_m \neq 1$. Hence if $\alpha_1 \neq 1$ the accumulation error of the scheme will be order $\mathcal{O}(h^{p+1})$ and we will not observe superconvergence at the roots of $R_{p+1}(\xi;\boldsymbol{\alpha})$. We formalize this in the following corollary.

Corollary 3.1. *If, in addition to the conditions of Theorem 3.4, we have that $m \geq 2$, i.e. $\alpha_1 = 1$, then after sufficient time the numerical error will satisfy*

$$\epsilon_j(\xi_l, t) = \mathcal{O}(h^{p+2}),$$

where ξ_l are the roots of the polynomial $\tilde{R}_{p+1}(\xi;\boldsymbol{\alpha})$.

When we recall the PDE (3.3.8) for the numerical solution U_j, where the effects of the modifications the DG scheme were concentrated in the source term on the right hand side, we see that Theorems 3.1 and 3.4 provide a direct link between this source term and the superconvergence properties of the modified scheme. We note in particular that because this source term arises from the projection of the δ-functions at the cell interfaces the modifications to the DG scheme alter this projection and directly alter the superconvergence properties of the method. This reveals that the superconvergence of this family of methods is directly governed by the form and accuracy of the projection of these δ-functions. Further, we can conclude that the classical DG method achieves a certain optimality in the sense that when using the upwind flux the classical projection of the δ-function into the finite element space allows the scheme to achieve the highest possible order of accuracy at the downwind point of the cell.

A direct consequence of these results is that we are able to manipulate the superconvergence properties of the family of methods (3.2.3). Indeed, from Theorem 3.4 and Corollary 1 we see that if we hold α_0 and α_1 to be one then we will obtain a superconvergent numerical solution at the roots of the polynomial $\tilde{R}_{p+1}(\xi; \boldsymbol{\alpha})$ after sufficient time. Hence for $p \geq 2$ we can design schemes which have particular superconvergent points. Moreover, since much of the superaccuracy results of these methods are based on the order of approximation of the rational function $\frac{\tilde{f}(\omega h_j,1)}{\tilde{g}(\omega h_j)}$ to the exponential function $e^{\omega h_j}$, we can also investigate schemes which will have particular rational approximations, not necessarily Padé approximants. We demonstrate these results in the numerical experiments in Section 3.5 below. In the next section, however, we perform a more in-depth study of the effects of the modifications on the method's stability.

3.4 Stability of the mDG Method

In this section we will study what effects modifying the flux multipliers α_k will have on the linear stability of the modified DG scheme. We pair the DG spatial discretization of order p with an order $p+1$ time-integration scheme, e.g. Runge-Kutta-$(p+1)$, in order to ensure a global convergence rate of order $p+1$. For the linear advection equation, when using an explicit order $p+1$ Runge-Kutta time-integration scheme to discretize (3.3.1), it is known [29] that the stability restriction on the size of the time step Δt scales with p as

$$\Delta t \lesssim \frac{h}{a(2p+1)}.$$

This simple estimate is at most 5% smaller than the exact CFL number [31]. From the analysis above we find that this time step restriction can be found by choosing Δt to be small enough that all of the scaled spectral values $\frac{a\lambda_{kn}\Delta t}{h}$ are contained within the absolute stability region of, in this case, Runge-Kutta-$(p+1)$. Here, each λ_{kn} is a solution of

$$\frac{\tilde{f}(-\lambda,1)}{\tilde{g}(-\lambda)} = e^{\kappa_n h}, \tag{3.4.1}$$

where $\kappa_n = \frac{2\pi n i}{L}$ where L is the length of the domain I and $k = 0,\dots,p$ and $n = 0,\dots,N$. Upon altering the multipliers in the modified DG scheme (3.3.1), the spectral values λ_{kn} will be changed. It is therefore possible that this stability restriction can be relaxed by choosing the multipliers α_k in some particular way. Since determining these spectral values explicitly is very difficult, we will resort to numerically calculating them using root-finding software and determine the time step restriction by numerically searching for the largest CFL number such that the scaled spectral values will be contained in the absolute stability region of RK-$(p+1)$. We will begin by only considering changes in the highest multiplier α_p since, as we will see, significant gains can be made in the relaxation of the stability restriction through only modifying the highest multiplier. We will then move on to study the effects of changing more than the highest multiplier.

Case 1: Only highest flux multiplier, α_p, is not equal to one.

Before we begin, let us note that in the particular case that only the highest multiplier of the modified scheme, α_p, is taken to be not equal to 1, we have a corollary of Theorem 2.2.

Corollary 3.2. *If the DG scheme is modified by only changing the highest multiplier, α_p, then the order of the dispersion error of the scheme is lowered by two to $\mathcal{O}(h^{2p})$ and the order of the dissipation error remains $\mathcal{O}(h^{2p+1})$.*

This corollary tells us that upon modifying the highest multiplier the order of accuracy in dissipation and dispersion of the scheme is only minimally affected. Therefore, the improvements in the stability restriction resulting from the modification of only the highest coefficient will have the benefit of only mildly reducing the orders of the error in dissipation and dispersion of the DG scheme. This is particularly true when using a very high-order approximation since for large p the differences between an $\mathcal{O}(h^{2p+2})$ error and an $\mathcal{O}(h^{2p})$ error will be fairly negligible.

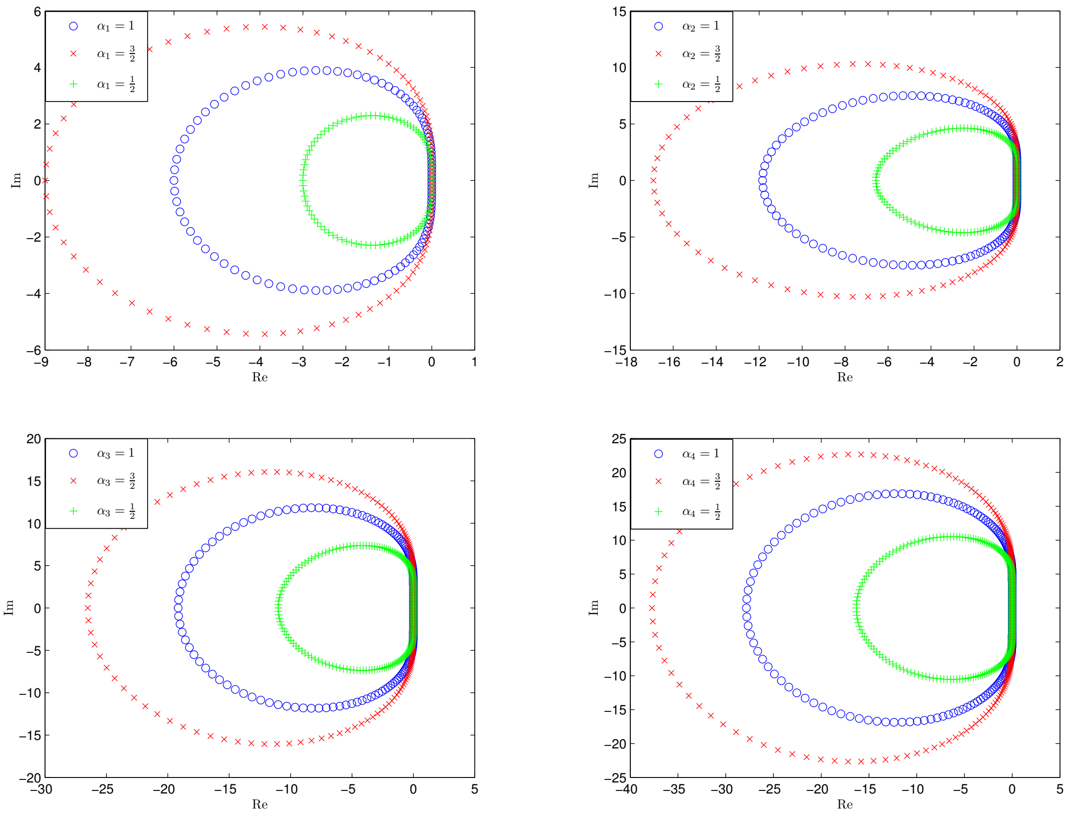


Figure 3.1: Spectral values λ_{kn} of the spatial DG discretization for the linear advection equation, for the $p = 1$ and 2 (top) and $p = 3$, and 4 (bottom), with $N = 50$. We show in each figure the values of λ for $\alpha_p = 1, \frac{3}{2}$, and $\frac{1}{2}$.

In Figure 3.1 we show the spectral values λ_{kn} for the $p = 1, 2, 3,$ and 4 schemes, respectively, with different values for the highest multiplier α_p in each case. In each figure, we show with the 'o' marker the spectrum for $\alpha_p = 1$, which is the spectrum of the original DG scheme, together with the spectra for $\alpha_p = \frac{3}{2}$ and $\alpha_p = \frac{1}{2}$ with the 'x' and '+' markers, respectively. We notice from these figures that, in general, the modification of the highest coefficient has the effect of scaling the spectrum. In particular, upon increasing the α_p multiplier the spectrum of spatial discretization is enlarged, while decreasing the α_p multiplier reduces the size of the spectrum. From this, we immediately see that when $\alpha_p < 1$, and the spectrum is reduced, we are able to choose the CFL number larger and still have a stable scheme. In contrast, when $\alpha_p > 1$ we must choose the CFL number

p	α_p	CFL	Relative Increase
1	1.000	0.33	3.00
	0.333	1.00	
2	1.000	0.21	2.97
	0.210	0.62	
3	1.000	0.14	2.60
	0.260	0.37	
4	1.000	0.11	2.46
	0.270	0.28	
5	1.000	0.09	2.40
	0.330	0.22	
6	1.000	0.08	2.34
	0.345	0.19	
7	1.000	0.07	2.27
	0.360	0.16	
8	1.000	0.06	2.24
	0.380	0.14	
9	1.000	0.05	2.21
	0.385	0.12	
10	1.000	0.05	2.19
	0.395	0.11	

Table 3.1: Largest CFL numbers obtained with the modified DG scheme on the linear advection equation for $p = 1, 2, \ldots, 10$, only modifying the highest order coefficient. Relative increase is calculated as the ratio between the increased CFL of the modified scheme, divided by the CFL number of the original DG scheme.

smaller and the stability condition of the scheme is made more restrictive. Although for completeness we include the cases when $\alpha_p > 1$ in our numerical tests below, we remark that modifying the DG scheme in this way has little benefit since both the stability restriction is tightened and the accuracy of the scheme is reduced.

Now that we have established that the stability restriction of the DG scheme can be relaxed through reducing the highest multiplier α_p, our next pursuit is to determine precisely the degree to which the stability condition can be improved, what choices of α_p give us the most relaxed time-step restriction, and how much of an improvement we can expect to gain for very high-order approximations. To answer these questions, we have used a MATLAB program which calculates the spectral values λ_{kn} for varying values of α_p and uses this spectrum to find the largest CFL number so that the complete spectrum of $CFL \cdot \lambda_{kn}$ is contained within the absolute stability region of RK-$(p+1)$ via a bisection algorithm. In Table 3.1 we present the largest CFL number we were able to obtain using this program for schemes of order $p = 1, 2, \ldots, 10$, together with the value of α_p for which the scheme obtains this CFL number. From this we see that we are able to achieve a significant increase in the usual CFL number of the DG scheme. We conjecture that for very high-order schemes we can expect to obtain a two-fold increase in the CFL number of the DG scheme by only modifying the highest multiplier to be $\alpha_p \approx 0.4$. We note that this significant gain in the CFL number comes at the cost of only one order of accuracy in the form of a dispersive error, while no additional dissipative error is introduced. In fact, we can establish another property of the scheme with this modification: the semi-discrete scheme (3.3.1) is linearly stable for any choice of $\alpha_p > 0$.

Proposition 3.2. *The modified DG scheme* (3.3.1) *with each multiplier* $\alpha_m = 1$, $m = 1, \ldots, p-1$, *and* $\alpha_p > 0$, *is linearly stable.*

Proof. Without loss of generality, we can assume $a = 1$ in the linear advection equation. Using $\alpha_m = 1, m = 1, \ldots, p-1$, the scheme (3.3.1) with the upwind flux can be written

$$\frac{h_j}{2k+1}\frac{dc_{jk}}{dt} = -(-1)^k[[U_j]] - \int_{-1}^{1} \frac{dU_j}{d\xi} P_k \, d\xi, \quad k = 0, 1, \ldots, p-1, \tag{3.4.2}$$

$$\frac{h_j}{2p+1}\frac{dc_{jp}}{dt} = -(-1)^p \alpha_p [[U_j]] - \int_{-1}^{1} \frac{dU_j}{d\xi} P_p \, d\xi. \tag{3.4.3}$$

Multiplying each equation (3.4.2) by $c_{jk}(t)$, then multiplying (3.4.3) by $\frac{1}{\alpha_p} c_{jp}(t)$ and summing, we obtain

$$\begin{aligned} \frac{1}{2}\frac{d}{dt}\left[\left(\sum_{k=0}^{p-1} \frac{h_j}{2k+1} c_{jm}^2\right) + \frac{h_j}{(2p+1)\alpha_p} c_{jp}^2\right] &= -U_j(x_j)[[U_j]] \\ &\quad - \int_{-1}^{1} \frac{dU_j}{d\xi}\left[\left(\sum_{k=0}^{p-1} c_{jk} P_k\right) + \frac{1}{\alpha_p} c_{jp} P_p\right] d\xi. \end{aligned} \tag{3.4.4}$$

Since $\frac{dU_j}{d\xi}$ is a polynomial of degree less than p, the integral $\int_{-1}^{1} \frac{dU_j}{d\xi} P_p \, d\xi = 0$. We then obtain

$$\begin{aligned}\int_{-1}^{1}\frac{dU_j}{d\xi}\left[\left(\sum_{k=0}^{p-1}c_{jk}P_k\right)+\frac{1}{\alpha_p}c_{jp}P_p\right]\,d\xi &= \int_{-1}^{1}\frac{dU_j}{d\xi}\left[\left(\sum_{k=0}^{p-1}c_{jk}P_k\right)+c_{jp}P_p\right]\,d\xi,\\ &= \int_{-1}^{1}\frac{dU_j}{d\xi}U_j\,d\xi,\\ &= \frac{1}{2}U_j^2(x_{j+1})-\frac{1}{2}U_j^2(x_j). \qquad (3.4.5)\end{aligned}$$

Substituting (3.4.5) into (3.4.4) yields

$$\begin{aligned}\frac{1}{2}\frac{d}{dt}\left[\left(\sum_{k=0}^{p-1}\frac{h_j}{2k+1}c_{jk}^2\right)+\frac{h_j}{(2p+1)\alpha_p}c_{jp}^2\right] &= -U_j(x_j)[[U_j]]-\frac{1}{2}U_j^2(x_{j+1})+\frac{1}{2}U_j^2(x_j),\\ &= -\frac{1}{2}U_j^2(x_{j+1})+U_j(x_j)U_{j-1}(x_j)-\frac{1}{2}U_j^2(x_j).\end{aligned}$$

Finally, summing over the entire mesh and using the periodicity of the boundary conditions yields

$$\begin{aligned}\frac{1}{2}\frac{d}{dt}\sum_{j=0}^{N}\left(\sum_{k=0}^{p-1}\frac{h_j}{2k+1}c_{jk}^2+\frac{h_j}{(2p+1)\alpha_p}c_{jp}^2\right)&\\ &= \sum_{j=0}^{N}\left(-\frac{1}{2}U_j^2(x_{j+1})+U_j(x_j)U_{j-1}(x_j)-\frac{1}{2}U_j^2(x_j)\right),\\ &= \sum_{j=0}^{N}\left(-\frac{1}{2}U_{j-1}^2(x_j)+U_j(x_j)U_{j-1}(x_j)-\frac{1}{2}U_j^2(x_j)\right),\\ &= -\frac{1}{2}\sum_{j=0}^{N}\left(U_j(x_j)-U_{j-1}(x_j)\right)^2\leq 0.\end{aligned}$$

Therefore, we find that for any $\alpha_p > 0$, $\sum_{j=0}^{N}||\mathbf{c}_j||$ will be bounded, and hence the semi-discrete scheme is linearly stable. □

Case 2: Several flux multipliers are not equal to one.

When several multipliers in the modified scheme (3.3.1) are taken to be not equal to one, we encounter several difficulties. Firstly, as we have established above, as we alter more multipliers the order of accuracy diminishes as we introduce larger dispersive and dissipative errors into the scheme. Secondly, the search for the choices of the multipliers which will yield the largest gain in the CFL number becomes computationally expensive. Thirdly, in our tests we observed that when more than one multiplier is modified, some spectral values λ_{kn} may have positive real parts. Therefore, a linear stability analysis of the type presented in Proposition 3.2 is not possible.

To understand why the scheme can become unstable, we consider the specific case when $p = 2$ and consider modifications to the second highest multiplier, α_1. Following the arguments of Theorem 3.2 we explicitly calculate the relation between the numerical wavenumber $\tilde{\kappa}$ and the exact frequency ω (for simplicity we set $\alpha_2 = 1$) to find

$$\tilde{\kappa} = \omega + \frac{1-\alpha_1}{120}\omega^4 h^3 + \frac{\alpha_1(1-\alpha_1)}{1200}\omega^5 h^4 + \mathcal{O}(h^5).$$

From this equation, we see that when $\alpha_1 < 1$ the coefficient in front of ω^4 will be positive. Since the numerical solution associate to this frequency and wavenumber satisfies $U(x_{j+1}, t) = e^{\tilde{\kappa}h} U_{j-1}(x_j, t)$ with ω purely imaginary, this error term will cause the magnitude of the solution to grow with j, rather than remain bounded. Hence, this order 3 error in $\tilde{\kappa}$ is the cause of the instability that can be observed when solving (3.3.1) numerically.

In general, we can use these expansions of $\tilde{\kappa}$ to determine what choices of α_k will produce an unstable scheme. For example, if we calculate the complete expansion of $\tilde{\kappa}$ for $p = 3$ we find

$$\begin{aligned}\tilde{\kappa} = \omega &+ \frac{\alpha_1 - 1}{1680\alpha_3}\omega^5 h^4 + \frac{7\alpha_3(\alpha_2-1) + 3\alpha_2(\alpha_1-1)}{70560\alpha_3^2}\omega^6 h^5 \\ &+ \frac{49\alpha_3^2(\alpha_3-1) + 35\alpha_3\alpha_2(\alpha_2-1) + (147\alpha_3^2 - 21\alpha_3\alpha_1 + 15\alpha_2^2)(\alpha_1-1)}{4939200\alpha_3^3}\omega^7 h^6 + \mathcal{O}(h^7),\end{aligned}$$

and the condition that the coefficient on ω^6 is positive can be written

$$\alpha_1 \geq \frac{7\alpha_3(1-\alpha_2)}{3\alpha_2} + 1.$$

Therefore, if we alter the highest three multipliers for the $p = 3$ scheme we can expect that the scheme will be stable if this condition is met. In general, the condition that the coefficient of ω^{2p} in the expansion of $\tilde{\kappa}$ will not cause an instability can be written

$$\alpha_{p-2} \geq \frac{(2p+1)\alpha_p(1-\alpha_{p-1})}{(2p-3)\alpha_{p-1}} + 1.$$

p	α_p	α_{p-1}	α_{p-2}	CFL	Relative Increase
3	1.00	1.00	1.00	0.14	5.40
	0.04	0.39	1.15	0.78	
4	1.00	1.00	1.00	0.11	4.06
	0.04	0.41	1.16	0.47	
5	1.00	1.00	1.00	0.09	3.88
	0.07	0.52	1.16	0.36	

Table 3.2: Largest CFL numbers obtained with the modified DG scheme on the linear advection equation for $p = 3, 4$, and 5 modifying the three highest order coefficients. Relative increase is calculated as the ratio between the increased CFL of the modified scheme, divided by the CFL number of the original DG scheme.

Hence, this condition tells us that we can expect to obtain a stable scheme when reducing the second highest multiplier, α_{p-1}, so long as the third highest multiplier, α_{p-2}, is chosen to be sufficiently large. Using this information, we again use our MATLAB program to search for the optimal choices of the three highest multipliers. More specifically, we construct a mesh of test values for α_p, α_{p-1}, and α_{p-2} and search for the specific point in this mesh which yields the largest CFL number in the modified scheme. The mesh is then refined and the process is repeated until a desired amount of accuracy for this optimal point is obtained. The obvious downside of this modification is that we must now alter the highest *three* multipliers, rather than just the highest two. This modification will therefore have a more severe effect on the overall accuracy of the scheme. We show the results of this search in Table 3.2 where we see that we can again substantially improve the usual CFL number of the DG scheme. However, this large increase in the CFL number appears to diminish as the order of the scheme rises, and the effects of this modification become less disruptive.

3.5 Numerical Examples

In this section we present two distinct applications of the analysis in the sections above. Our primary goal is to apply the modified DG scheme to several test examples to confirm its convergence rate and superconvergence properties as well as observe the general performance of the scheme in comparison with the standard DG scheme. We will begin by testing the modified scheme with several choices of the multipliers $\alpha_k, k = 1, \ldots, p$, to show that we retain the usual $p+1$ convergence rate on smooth solutions. We will also demonstrate that we are able to specifically alter the superconvergence properties of the method. We will then show how the modified scheme performs for a linear problem with several different waveforms. We will also present an example where the accuracy of the modified scheme on a fine mesh is compared to the accuracy of the DG scheme on a coarse mesh, but the computational effort of both schemes is relatively equivalent. These examples are specifically chosen with initial conditions with fine structure where mesh refinement may be more beneficial to accuracy than the higher-order dissipation and dispersion errors of the DG scheme. We will conclude the section by applying the modified scheme to some non-linear problems, in which we will again confirm the convergence rate and show that the demonstrated gains in the CFL condition do indeed carry over to non-linear problems.

3.5.1 Superconvergence of the mDG scheme

In this section we will apply the modified DG scheme to a linear test problem to confirm its global convergence rate. We will then show that we are able to control its superconvergence properties through specific choices of the flux multipliers α_k.

Convergence Study

Our convergence studies were done on the same linear advection initial value problem (2.4.1) in Section 2.4, this time with the initial condition

$$u_0(x) = \frac{1}{2}\sin \pi x. \tag{3.5.1}$$

In Tables 3.3-3.5 we show the results of the convergence tests for the $p = 1, 2$, and 3 schemes. In each table, we present errors ϵ_1 in the $\mathcal{L}^1$ norm at $t = 2$ after one full period on uniform meshes having 16, 32, 64, 128, and 256 elements. To obtain a proper comparison of the accuracy of the numerical solution for each choice of the α_m multipliers, the CFL number was chosen to be as large as possible, with the exception of the case $p = 1$ and $\alpha_1 = \frac{1}{3}$. In this case, a simple calculation can show that when the time step is chosen to be precisely $\Delta t = \frac{h}{a}$ this scheme will perfectly advect, i.e. with no numerical error committed, the piecewise linear numerical solution of the linear advection equation[1]. For

[1] It is worth noting that for $p = 2$ we are able to construct a scheme which also perfectly advects the piecewise quadratic solution to (2.1.1) by choosing $\alpha_0 = 1$, $\alpha_1 = \frac{1}{2}$, $\alpha_2 = \frac{1}{10}$ and $CFL = 1$. However, as discussed in section 3.4, because $\alpha_1 < 1$ and $\alpha_0 = 1$, the scheme is linearly unstable for $CFL \neq 1$.

	$\alpha_1 = 1, CFL = \frac{1}{3}$		$\alpha_1 = \frac{4}{3}, CFL = \frac{1}{4}$		$\alpha_1 = \frac{2}{3}, CFL = \frac{1}{2}$		$\alpha_1 = \frac{1}{3}, CFL = 0.9$	
N	ϵ_1	r	ϵ_1	r	ϵ_1	r	ϵ_1	r
16	1.26e-02	-	1.97e-02	-	6.63e-03	-	2.14e-02	-
32	3.00e-03	2.07	4.88e-03	2.01	1.73e-03	1.93	5.77e-03	1.89
64	7.29e-04	2.04	1.21e-03	2.01	4.45e-04	1.96	1.47e-03	1.98
128	1.80e-04	2.02	3.02e-04	2.01	1.12e-04	1.99	3.73e-04	1.98
256	4.47e-05	2.01	7.54e-05	2.00	2.80e-05	2.00	9.39e-05	1.99

Table 3.3: Linear advection, (2.4.1), (3.5.1). $\mathcal{L}^1$ errors ϵ_1 and convergence rates, r, for the sine wave initial condition, $p = 1$. Errors are calculated at $t = 2$, after one full period.

	$\alpha_2 = 1, CFL = \frac{1}{5}$		$\alpha_2 = \frac{7}{5}, CFL = \frac{1}{10}$		$\alpha_2 = \frac{2}{5}, CFL = \frac{2}{5}$		$\alpha_2 = \frac{1}{5}, CFL = \frac{3}{5}$	
N	ϵ_1	r	ϵ_1	r	ϵ_1	r	ϵ_1	r
16	1.66e-04	-	1.07e-04	-	8.10e-04	-	2.44e-03	-
32	2.06e-05	3.01	1.31e-05	3.04	9.93e-05	3.03	3.02e-04	3.02
64	2.57e-06	3.00	1.62e-06	3.02	1.23e-05	3.01	3.76e-05	3.01
128	3.21e-07	3.00	2.01e-07	3.01	1.53e-06	3.01	4.70e-06	3.00
256	4.01e-08	3.00	2.51e-08	3.00	1.91e-07	3.00	5.87e-07	3.00

Table 3.4: Linear advection, (2.4.1), (3.5.1). $\mathcal{L}^1$ errors ϵ_1 and convergence rates, r, for the sine wave initial condition, $p = 2$. Errors are calculated at $t = 2$, after one full period.

this reason, we choose a CFL number that is slightly less than the maximum possible. In these convergence tests, when choosing the multipliers in the modified scheme to be not equal to 1, we obtain that the scheme is less accurate in terms of the $\mathcal{L}^1$ error. This is expected, since these modifications result in increased dispersion and dissipation errors as compared to the original DG scheme and these errors lead to a faster growth of the accumulated error. The temporal component of the error also increases due to a larger time step. We also see from these tables that for any stable scheme of order $p + 1$, we retain the full $p + 1$ order convergence rate regardless of the choices for the multipliers α_k, $k = 1, \ldots, p$.

Our numerical experiments revealed that when the lowest multiplier α_0 was changed, the order of convergence of the scheme was reduced by one. This was to be expected, as

	$\alpha_3 = 1, CFL = 0.14$		$\alpha_3 = 0.33, CFL = 0.35$		$\alpha_3 = 0.04, \alpha_2 = 0.39,$ $\alpha_1 = 1.15, CFL = 0.78$	
N	ϵ_1	r	ϵ_1	r	ϵ_1	r
16	3.38e-06	-	1.74e-05	-	5.15e-04	-
32	2.11e-07	4.00	1.08e-06	4.01	3.27e-05	3.97
64	1.32e-08	4.00	6.72e-08	4.00	2.04e-06	4.00
128	8.27e-10	4.00	4.20e-09	4.00	1.28e-07	4.00
256	5.17e-11	4.00	2.62e-10	4.00	7.99e-09	4.00

Table 3.5: Linear advection, (2.4.1), (3.5.1). $\mathcal{L}^1$ errors ϵ_1 and convergence rates, r, for the sine wave initial condition, $p = 3$. Errors are calculated at $t = 2$, after one full period.

remarked above, and hence was not reported.

Superconvergence

In the following numerical examples we present two distinct applications of the superconvergence results of Section 2.2. In our first example we choose to alter the δ-function projection (i.e. choose the flux multipliers α_k) so that the rational function $\frac{\tilde{f}(\omega h_j,1)}{\tilde{g}(\omega h_j)}$ takes a particular form. In this way we are able to construct a scheme with particular superaccuracy properties and predict its superconvergence properties. In the second example we will choose a δ-function projection such that the superconvergent points of the modified scheme are located at particular points. In both examples we perform a convergence study on the same linear advection initial value problem (2.4.1)-(3.5.1). All tests are completed using an RK-4 time-stepping scheme and a CFL number of $\frac{0.15}{2p+1}$ to minimize the error incurred in time integration. The initial projections are also chosen to be the usual L^2 projection, which satisfies the conditions of Theorem 3.4.

From the discussion above, we have seen that the superconvergence and superaccuracy of the DG method is directly linked to the accuracy of the rational approximation of the exponential function, $\frac{\tilde{f}(\omega h,1)}{\tilde{g}(\omega h)}$. In the first example we show that we can choose the highest order flux modifier α_p so that $\frac{\tilde{f}(\omega h,1)}{\tilde{g}(\omega h)}$ is the $\frac{p-1}{p+1}$ Padé approximant of $e^{\omega h}$. We can then determine the superconvergence properties of this modified DG scheme. This process can be analogously extended for other rational approximations, not necessarily Padé approximants.

In order to construct a modified DG scheme such that its associated rational function $\frac{\tilde{f}(\omega h,1)}{\tilde{g}(\omega h)}$ is the $\frac{p-1}{p+1}$ Padé approximant of $e^{\omega h}$ we choose the parameters α_k such that $\tilde{f}(\omega h, 1)$ is a polynomial of degree $p-1$, and $\frac{\tilde{f}(\omega h,1)}{\tilde{g}(\omega h)}$ approximates $e^{\omega h}$ to order $2p+1$. By Lemma 3.1 above, this rational function will have this order of approximation if α_p is the only flux multiplier chosen not equal to one. In Proposition 1 of [19] it was shown that the polynomials $\tilde{f}(\omega h, 1)$ and $\tilde{g}(\omega h)$ can be generated through certain recursion relations. Using these relations, one can show that the coefficient on $(\omega h)^p$ in $\tilde{f}(\omega h, 1)$ is $\sum_{k=0}^{p}(-1)^{p+k}(2k+1)\alpha_k$. Hence, fixing $\alpha_k = 1$ for $k = 0, \ldots, p-1$ we can choose

$$\begin{aligned}\alpha_p &= \frac{1}{2p+1}\sum_{k=0}^{p-1}(-1)^{p+k-1}(2k+1),\\ &= \frac{p}{2p+1},\end{aligned}$$

in order to obtain that $\tilde{f}(\omega h, 1)$ will be a polynomial of degree $p-1$ and $\frac{\tilde{f}(\omega h,1)}{\tilde{g}(\omega h)}$ will be the $\frac{p-1}{p+1}$ Padé approximant of $e^{\omega h}$.

For this specific choice of flux multipliers, we can determine the superconvergence properties of the modified scheme through the analysis above. In particular, using (3.3.7) we determine the polynomial $\tilde{R}_{p+1}(\xi; \boldsymbol{\alpha})$ to be

$$\begin{aligned}\tilde{R}_{p+1}(\xi;\boldsymbol{\alpha}) &= \frac{1}{2}\sum_{k=0}^{p-1}(-1)^{k+1}[P_{k+1}-P_{k-1}] + \frac{(-1)^{p+1}}{2}\frac{p}{2p+1}[P_{p+1}-P_{p-1}],\\ &= \frac{(-1)^p}{2}[P_p - P_{p-1}] + \frac{(-1)^{p+1}}{2}\frac{p}{2p+1}[P_{p+1}-P_{p-1}],\\ &= \frac{(-1)^{p+1}}{2}\left[\frac{p}{2p+1}P_{p+1} - P_p + \frac{p+1}{2p+1}P_{p-1}\right],\end{aligned}$$

and from Corollary 3.1 we know that for $p \geq 2$ the numerical error will tend towards being proportional to this polynomial. Hence, for $p \geq 2$ the numerical solution will converge at a rate of $p+2$ at the roots of this polynomial, and converge at a rate of $2p$ at the downwind point of the cell. In fact, in this special case the polynomial $\tilde{R}_{p+1}(\xi;\boldsymbol{\alpha})$ has a double root at the downwind point which implies that the spatial derivative of the numerical error will also be order $2p$ at the downwind point of the cell.

In Table 3.6 we show the results of our convergence tests for this particular modified scheme. At each of the roots of $\tilde{R}_{p+1}(\xi;\boldsymbol{\alpha})$, including the downwind point, we calculate

		Downwind point		Derivative at Downwind point		1st Root		2nd Root	
p	N	Error	Order	Error	Order	Error	Order	Error	Order
1	20	3.10e-02	-	1.10e-01	-				
	40	8.30e-03	1.90	2.96e-02	1.90				
	60	3.75e-03	1.96	1.33e-02	1.96				
	80	2.13e-03	1.98	7.55e-03	1.98				
	100	1.37e-03	1.98	4.85e-03	1.98				
2	20	4.70e-05	-	1.72e-04	-	4.07e-05	-		
	40	2.97e-06	3.98	1.09e-05	3.98	2.54e-06	4.00		
	60	5.90e-07	3.99	2.17e-06	3.99	5.02e-07	4.00		
	80	1.87e-07	3.99	6.86e-07	4.00	1.59e-07	4.00		
	100	7.67e-08	4.00	2.81e-07	4.00	6.50e-08	4.00		
3	20	3.48e-08	-	1.24e-07	-	2.00e-07	-	2.57e-07	-
	40	5.48e-10	5.99	2.01e-09	5.95	5.90e-09	5.08	8.45e-09	4.93
	60	4.84e-11	5.99	1.78e-10	5.99	7.61e-10	5.05	1.13e-09	4.96
	80	8.70e-12	5.96	3.21e-11	5.95	1.76e-10	5.04	2.70e-10	4.97
	100	2.32e-12	5.92	8.73e-12	5.83	5.82e-11	5.06	8.89e-11	4.98

Table 3.6: Linear advection, (2.4.1)-(3.5.1) with modified scheme associated to the $\frac{p-1}{p+1}$ Padé approximant. L^1 norm of the point-wise error of the numerical solution and the derivative of the numerical solution at the downwind points are shown with the L^1 norm of the point-wise error of the numerical solution at the interior roots of $\tilde{R}^{-}_{p+1}(\xi;\boldsymbol{\alpha})$.

the error as the L^1 norm of the vector of point-wise errors $U_j - u_j$. We also calculate the error in the spacial derivative of the numerical solution at the downwind point by taking the L^1 norm of the vector of point-wise errors $\frac{d}{dx}(U_j - u_j)$. For the $p = 2$ scheme the final root of $\tilde{R}_{p+1}(\xi;\boldsymbol{\alpha})$ is located at $\xi = -1/2$, while for the $p = 3$ scheme the roots are located at $\xi = -\frac{5\pm2\sqrt{10}}{15}$. From the table we see that for $p \geq 2$ we indeed achieve the expected order $2p$ rate of convergence in the numerical solution and the spacial derivative of the numerical solution at the downwind point of the cell. We also achieve the expected order $p + 2$ convergence of the numerical solution at the roots of $\tilde{R}_{p+1}(\xi;\boldsymbol{\alpha})$ in the interior of the cell.

In our second superconvergence test we show that we can choose the projection of

		Downwind point		1st Root		2nd Root		3rd Root	
p	N	Error	Order	Error	Order	Error	Order	Error	Order
1	20	3.20e-02	-	3.29e-02	-				
	40	8.20e-03	1.97	8.27e-03	1.99				
	60	3.66e-03	1.99	3.67e-03	2.00				
	80	2.06e-03	2.00	2.07e-03	2.00				
	100	1.32e-03	2.00	1.32e-03	2.00				
2	20	7.22e-05	-	7.33e-05	-	7.25e-05	-		
	40	4.56e-06	3.99	4.59e-06	4.00	4.52e-06	4.00		
	60	9.00e-07	4.00	9.05e-07	4.00	8.92e-07	4.00		
	80	2.85e-07	4.00	2.86e-07	4.00	2.82e-07	4.00		
	100	1.17e-07	4.00	1.17e-07	4.00	1.16e-07	4.00		
3	20	3.16e-08	-	2.62e-08	-	1.55e-07	-	1.67e-07	-
	40	4.67e-10	5.99	3.28e-10	6.32	4.41e-09	5.13	5.67e-09	4.88
	60	4.40e-11	5.98	2.31e-11	6.54	5.62e-10	5.08	7.66e-10	4.94
	80	8.09e-12	5.89	3.68e-12	6.39	1.31e-10	5.05	1.83e-10	4.96
	100	2.24e-12	5.75	9.83e-13	5.92	4.27e-11	5.04	6.06e-11	4.97

Table 3.7: Linear advection, (2.4.1)-(3.5.1) with modified scheme associated to the choice $\tilde{R}_{p+1}(\xi;\boldsymbol{\alpha}) = \frac{(-1)^{p+1}}{2}(\xi-1)P_p(\xi)$. L^1 norm of the point-wise error of the numerical solution at the downwind points are shown with the L^1 norm of the point-wise error of the numerical solution at the interior roots of $\tilde{R}_{p+1}(\xi;\boldsymbol{\alpha})$.

the interface δ-functions such that the modified scheme has certain potentially desirable superconvergent properties. Specifically, we can choose parameters α_k such that the superconvergent points, i.e. the roots of the polynomial $\tilde{R}_{p+1}(\xi;\boldsymbol{\alpha})$, are located at particularly chosen points. The rate of superconvergence at the downwind point can then be determined from the orthogonality properties of this polynomial $\tilde{R}_{p+1}(\xi;\boldsymbol{\alpha})$.

Suppose we wish to choose our modified scheme such that the interior roots of the polynomial $\tilde{R}_{p+1}(\xi;\boldsymbol{\alpha})$ are located at the roots of the Legendre polynomial P_p. Then $\tilde{R}_{p+1}(\xi;\boldsymbol{\alpha})$ takes the form

$$\tilde{R}_{p+1}(\xi;\boldsymbol{\alpha}) = \frac{(-1)^{p+1}}{2}(\xi-1)P_p(\xi). \tag{3.5.2}$$

Note that this choice of $\tilde{R}_{p+1}(\xi;\boldsymbol{\alpha})$ satisfies $\tilde{R}_{p+1}(-1;\boldsymbol{\alpha}) = 1$ and $\tilde{R}_{p+1}(1;\boldsymbol{\alpha}) = 0$ and, hence, the resulting modified scheme should preserve $\alpha_0 = 1$. To determine what choices of the flux multipliers α_k will yield a polynomial $\tilde{R}_{p+1}(\xi; \boldsymbol{\alpha})$ of this form we note that that the Legendre polynomials satisfy the following recursion relation [1]

$$\xi P_p = \frac{p+1}{2p+1} P_{p+1} + \frac{p}{2p+1} P_{p-1}.$$

Using this relation in (3.5.2) we obtain

$$\begin{aligned} \tilde{R}_{p+1}(\xi; \boldsymbol{\alpha}) &= \frac{(-1)^{p+1}}{2} \left[\frac{p+1}{2p+1} P_{p+1} + \frac{p}{2p+1} P_{p-1} - P_p \right], \\ &= \frac{(-1)^{p+1}}{2} \left[\frac{p+1}{2p+1} (P_{p+1} - P_{p-1}) - (P_p - P_{p-1}) \right]. \end{aligned} \tag{3.5.3}$$

Comparing (3.5.3) to (3.3.7) we obtain that the choice of $\alpha_k = 1$ for $k = 0, \ldots, p-1$ and $\alpha_p = \frac{p+1}{2p+1}$ will yield this polynomial $\tilde{R}_{p+1}(\xi; \boldsymbol{\alpha})$.

Using this choice of $\tilde{R}_{p+1}(\xi; \boldsymbol{\alpha})$ in the definition of the polynomials $\tilde{R}_{p+1}^{(-k)}(\xi; \boldsymbol{\alpha})$ in (2.3.10), and using the orthogonality of the Legendre polynomial P_p, we obtain that $\tilde{R}_{p+1}^{(-k)}(1; \boldsymbol{\alpha}) = 0$ for $k = 0, \ldots, p-1$ and hence the local error at the downwind point of the cell will be $\mathcal{O}(h^{2p+1})$. Furthermore, because we have particularly chosen the roots of $\tilde{R}_{p+1}(\xi; \boldsymbol{\alpha})$ to be located at the roots of the Legendre polynomial P_p we obtain the the local error of the numerical solution at these points will be $\mathcal{O}(h^{p+2})$. We therefore expect to obtain a global rate of convergence of $2p$ at the downwind point of the numerical solution and a rate of convergence of $p+2$ at the roots of P_p inside the each cell when $p \geq 2$.

In Table 3.7 we show the results of our convergence tests for this particular modified scheme. At each of the roots of $\tilde{R}_{p+1}(\xi; \boldsymbol{\alpha})$, which in this case are the roots of P_p plus the downwind point, we calculate the error as the L^1 norm of the vector of point-wise errors $U_j - u_j$. From the table we see that for $p \geq 2$ we indeed achieve the expected order $2p$ rate of convergence in the numerical solution at the downwind point of the cell. We also achieve the expected order $p+2$ convergence of the numerical solution at the roots of P_p in the interior of the cell.

3.5.2 Performance on Linear Problems

The next test - with which we can more directly observe the effects of modifying the DG scheme on a variety of waveforms - involves solving the linear advection problem (2.4.1) with the following initial conditions [44]:

$$u_0(x) = \begin{cases} \frac{1}{6}(G(x,\beta,z-\delta)+G(x,\beta,z+\delta)+4G(x,\beta,z)) & -0.8 \le x \le -0.6, \\ 1 & -0.4 \le x \le -0.2, \\ 1-|10(x-0.1)| & 0 \le x \le 0.2, \\ \frac{1}{6}(F(x,\alpha,a-\delta)+F(x,\alpha,a+\delta)+4F(x,\alpha,z)) & 0.4 \le x \le 0.6, \\ 0 & \text{otherwise,} \end{cases} \tag{3.5.4a}$$

$$G(x,\beta,z) = e^{-\beta(x-z)^2}, \tag{3.5.4b}$$

$$F(x,\alpha,a) = \sqrt{\max(1-\alpha^2(x-a)^2, 0)}, \tag{3.5.4c}$$

where $a = 0.5$, $z = -0.7$, $\delta = 0.005$, $\alpha = 10$, and $\beta = \frac{\log 2}{36\delta^2}$. This initial profile consists of a combination of Gaussians, a square pulse, a sharp triangle, and a combination of half-ellipses. We present the results with out limiting in order to discuss the effect of the induced dispersive and dissipative errors in the modified scheme. These effects are better seen in the spurious oscillations near solution discontinuities - which limiting would destroy - and in the dissipation of local extrema, to which limiters heavily contribute. We then present an example where the limiter has been applied and note that there is little difference between the schemes in terms of accuracy. Implementation of limiters, e.g. the minmod [29] or moment limiter [46], is straightforward and analogous to their implementation in classical DG schemes.

The results of test (2.4.1)-(3.5.4) for the $p = 1, 2$, and 3 schemes are shown in Figures 3.2-3.4 at $t = 2$ after one full period, on a uniform mesh of $N = 200$ cells. In each figure we show several choices of the highest multiplier α_p and for the $p = 3$ scheme in Figure 3.4 we show an example where the three highest multipliers have been modified to their optimal values listed in Table 3.2. In Figure 3.2, we observe a slight shift to the left and right for $\alpha_1 = \frac{1}{3}$ and $\alpha_1 = \frac{4}{3}$, respectively, of the entire wave front for the $p = 1$ scheme. This is especially noticeable for the Gaussians and ellipses. The modified scheme for which $\alpha_1 = \frac{2}{3}$ is visually closer to the original DG scheme. This can be explained once we explicitly calculate the expansion of the numerical wavenumber $\tilde{\kappa}$ in terms of the exact frequency ω from Theorem 3.2 for the $p = 1$ scheme,

$$\tilde{\kappa} = \omega + \frac{\alpha_1 - 1}{12\alpha_1}\omega^3 h^2 + \mathcal{O}(h^3). \tag{3.5.5}$$

Hence, since ω is purely imaginary, choosing $\alpha_1 > 1$ will introduce an additional dispersive error of negative sign into the usual DG scheme. On the other hand, decreasing α_1 to $\frac{2}{3}$ introduces an additional positive dispersive error.

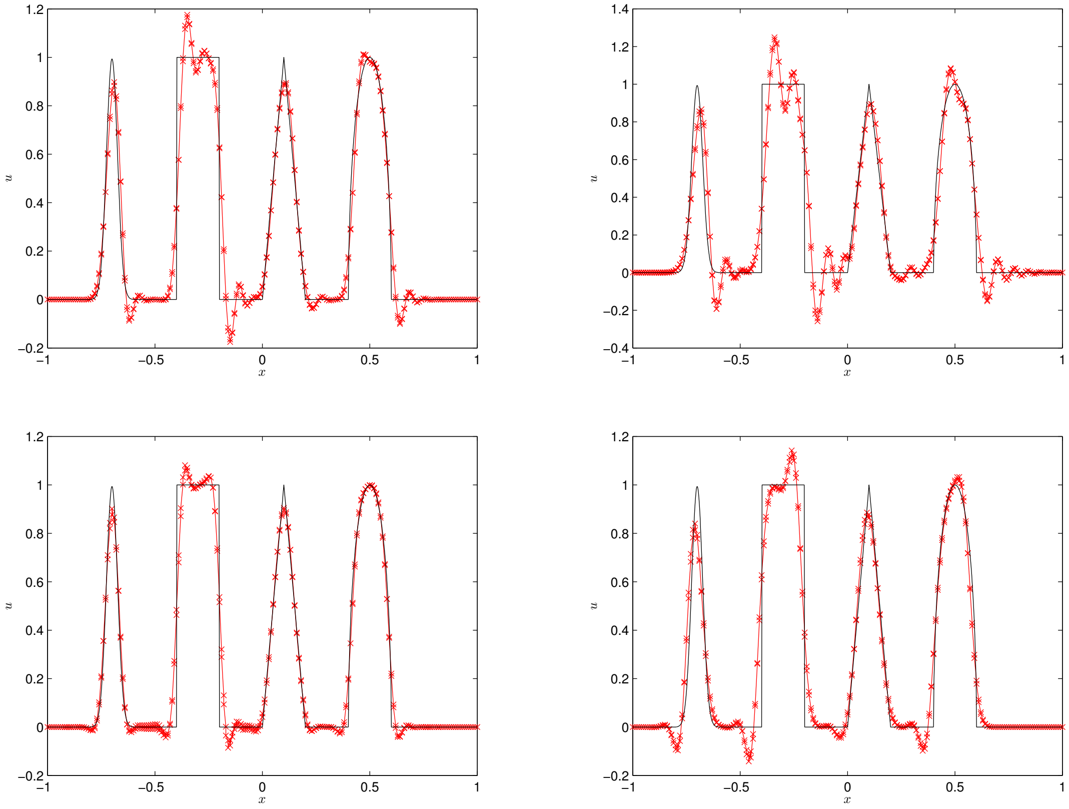


Figure 3.2: Linear advection, (2.4.1), (3.5.4), $p = 1$ on a mesh of $N = 200$ elements. Shown at $t = 2$, after one full period. Solid line shows the exact solution, line with 'x' markers shows the numerical solution. Top left: $\alpha_1 = 1, CFL = \frac{1}{3}$, Top Right: $\alpha_1 = \frac{4}{3}, CFL = \frac{1}{4}$, Bottom left: $\alpha_1 = \frac{2}{3}, CFL = \frac{1}{2}$, Bottom right: $\alpha_1 = \frac{1}{3}, CFL = 0.9$.

This property is true in general for the modified scheme, i.e. in the expansion of $\tilde{\kappa}$ for the order p scheme, when each multiplier is taken to be equal to one except the highest, the coefficient of ω^{2p+2} will have a similar form to (3.5.5). Therefore, choosing $\alpha_p > 1$ will add a negative dispersive error and shift the wave fronts to the right, while choosing $\alpha_p < 1$ will add a positive dispersive error and shift the wave fronts to the left. For example, the full expansion of $\tilde{\kappa}$ in the $p = 2$ scheme is calculated to be

$$\tilde{\kappa} = \omega + \frac{1-\alpha_1}{120\alpha_2}\omega^4 h^3 - \frac{5\alpha_2(\alpha_2 - 1) + 3\alpha_1(\alpha_1 - 1)}{3600\alpha_2^2}\omega^5 h^4 + \mathcal{O}(h^5), \tag{3.5.6}$$

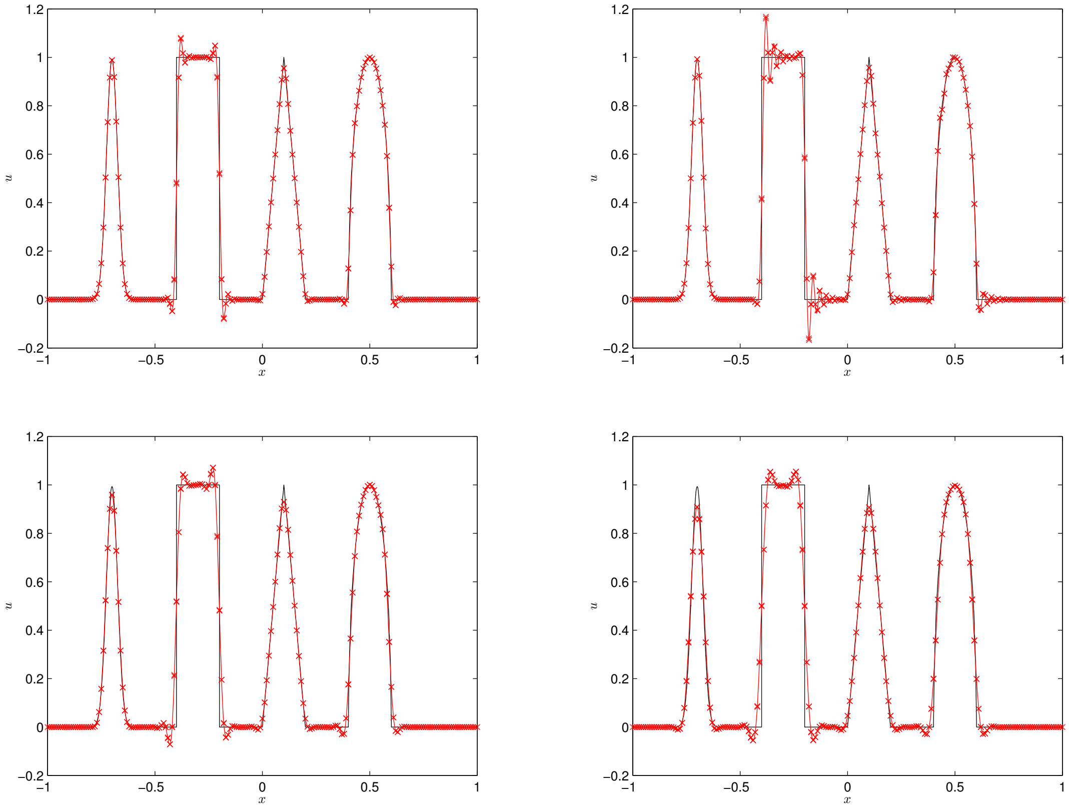


Figure 3.3: Linear advection, (2.4.1), (3.5.4), $p = 2$ on a mesh of $N = 200$ elements. Shown at $t = 2$, after one full period. Solid line shows the exact solution, line with 'x' markers shows the numerical solution. Top left: $\alpha_2 = 1, CFL = \frac{1}{5}$, Top Right: $\alpha_2 = \frac{7}{5}, CFL = \frac{1}{10}$, Bottom left: $\alpha_2 = \frac{2}{5}, CFL = \frac{2}{5}$, Bottom right: $\alpha_2 = \frac{1}{5}, CFL = \frac{3}{5}$.

and therefore when $\alpha_1 = 1$,

$$\tilde{\kappa} = \omega - \frac{\alpha_2 - 1}{720\alpha_2}\omega^5 h^4 + \mathcal{O}(h^5), \tag{3.5.7}$$

and the effects of altering α_2 in the $p = 2$ scheme will be analogous to the effects of altering α_1 in the $p = 1$ scheme.

We note that although the order of the leading errors of $\tilde{\kappa}$ may stay the same for different choices of the multipliers in (3.5.5)-(3.5.7), the magnitude of the error changes with different choices. Indeed from these examples it is clear that although the formal

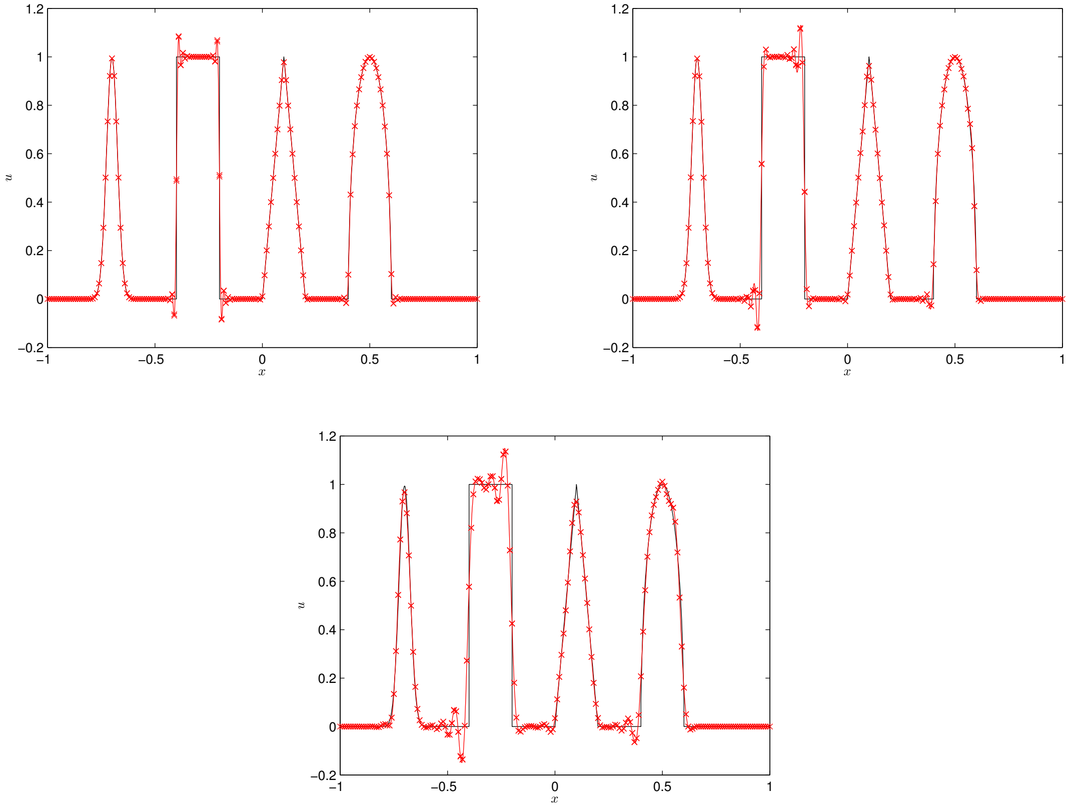


Figure 3.4: Linear advection, (2.4.1), (3.5.4), $p = 3$ on a mesh of $N = 200$ elements. Shown at $t = 2$, after one full period. Solid line shows the exact solution, line with 'x' markers shows the numerical solution. Top left: $\alpha_3 = 1, CFL = 0.14$, Top Right: $\alpha_3 = 0.33, CFL = 0.36$, Bottom: $\alpha_3 = 0.04, \alpha_2 = 0.39, \alpha_1 = 1.15, CFL = 0.78$.

order of accuracy remains the same, larger modifications may introduce larger errors in accuracy. In practice, care should be taken to choose the multipliers to obtain a balance between the stability gains and the deteriorating effects of the loss of accuracy.

Finally, we show in Figure 3.5 the results of this test for $p = 1$ with a minmod limiter implemented. We measure the errors to be 0.070, 0.079, 0.068, 0.117 for the DG, mDG with $\alpha_1 = 4/3, 2/3, 1/3$, respectively. Visually the solutions look similar, with the exception of the $\alpha_1 = 1/3$ case where the error is greater. This would seem to indicate that in the presence of discontinuities when a limiter is used there is little difference in accuracy

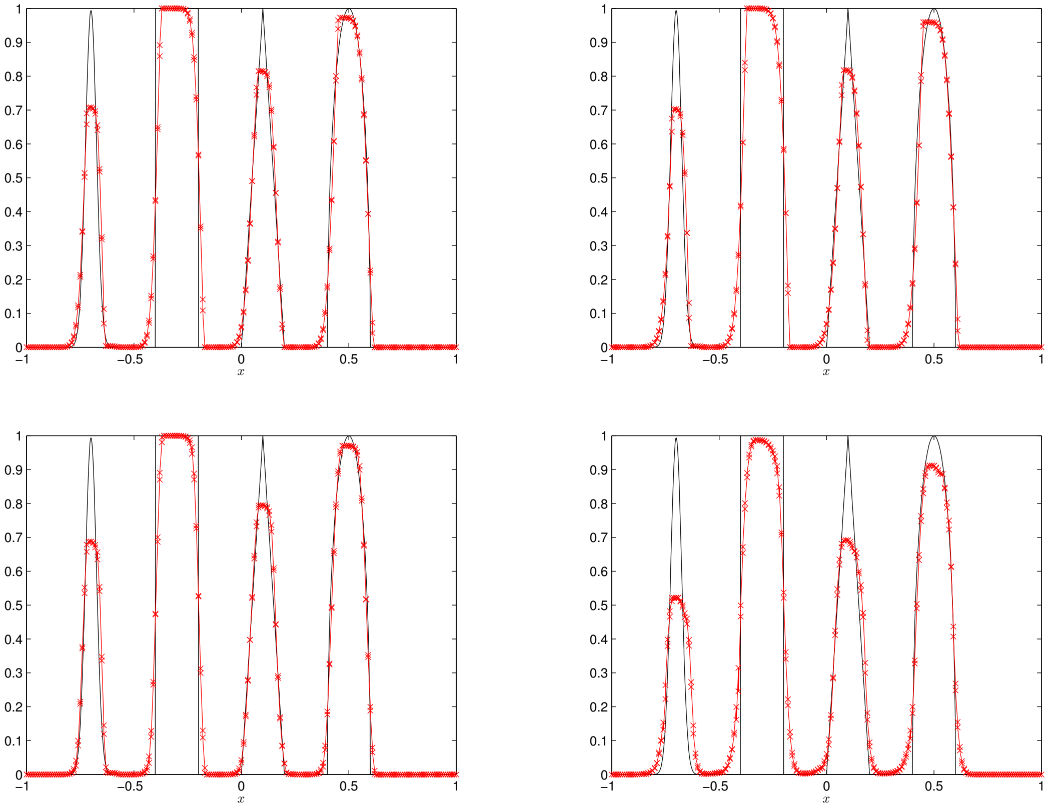


Figure 3.5: Linear advection, (2.4.1), (3.5.4), $p = 1$ on a mesh of $N = 200$ elements with minmod limiter. Shown at $t = 2$, after one full period. Solid line shows the exact solution, line with 'x' markers shows the numerical solution. Top left: $\alpha_1 = 1, CFL = \frac{1}{3}$, Top Right: $\alpha_1 = \frac{4}{3}, CFL = \frac{1}{4}$, Bottom left: $\alpha_1 = \frac{2}{3}, CFL = \frac{1}{2}$, Bottom right: $\alpha_1 = \frac{1}{3}, CFL = 0.9$.

of the solutions, i.e. for non-smooth problems the numerical error is almost completely determined by the errors introduced by the limiter. Hence, after limiting the detrimental effects on accuracy introduced by the modifications do not impact the overall accuracy of the scheme. This would seem to imply an immediate performance benefit of the modified scheme compared to the classical DG scheme since the modified scheme requires significantly fewer time-steps.

	$\alpha_1 = 1, CFL = \frac{1}{3}$		$\alpha_1 = \frac{4}{3}, CFL = \frac{1}{4}$		$\alpha_1 = \frac{2}{3}, CFL = \frac{1}{2}$		$\alpha_1 = \frac{1}{3}, CFL = 0.9$	
N	ϵ_1	r	ϵ_1	r	ϵ_1	r	ϵ_1	r
16	3.83e-03	-	3.54e-03	-	5.79e-03	-	1.58e-02	-
32	1.17e-03	1.71	9.92e-04	1.83	1.74e-03	1.73	3.78e-03	1.58
64	3.24e-04	1.84	2.67e-04	1.89	4.99e-04	1.81	1.10e-03	1.78
128	8.63e-05	1.91	7.01e-05	1.93	1.37e-04	1.87	3.02e-04	1.87
256	2.24e-05	1.95	1.80e-05	1.96	3.58e-05	1.93	7.96e-05	1.93

Table 3.8: Burgers' equation (3.5.8), (3.5.1). $\mathcal{L}^1$ errors ϵ_1 and convergence rates, r, $p = 1$. Errors are calculated at $t = 0.3$, before a shock wave forms.

	$\alpha_2 = 1, CFL = \frac{1}{5}$		$\alpha_2 = \frac{7}{5}, CFL = \frac{1}{10}$		$\alpha_2 = \frac{2}{5}, CFL = \frac{2}{5}$		$\alpha_2 = \frac{1}{5}, CFL = \frac{3}{5}$	
N	ϵ_1	r	ϵ_1	r	ϵ_1	r	ϵ_1	r
16	2.58e-04	-	2.02e-04	-	7.04e-04	-	1.40e-03	-
32	3.43e-05	2.91	2.76e-05	2.87	9.45e-05	2.90	1.95e-04	2.84
64	4.63e-06	2.89	3.56e-06	2.95	1.22e-05	2.95	2.62e-05	2.90
128	6.16e-07	2.91	4.54e-07	2.97	1.60e-06	2.93	3.49e-06	2.91
256	8.03e-08	2.94	5.78e-08	2.97	2.10e-07	2.93	4.61e-07	2.92

Table 3.9: Burgers' equation (3.5.8), (3.5.1). $\mathcal{L}^1$ errors ϵ_1 and convergence rates, r, $p = 2$. Errors are calculated at $t = 0.3$, before a shock wave forms.

3.5.3 Performance on Nonlinear Problems

To test the modified scheme on a non-linear problem, we consider Burgers' equation,

$$u_t + uu_x = 0, \tag{3.5.8}$$

on $[-1, 1]$, with periodic boundary conditions and with the sine wave initial condition, (3.5.1). We perform our convergence tests on this problem for the $p = 1$ and $p = 2$ schemes for various choices of the multipliers α_k and show the results in Tables 3.8 and 3.9. We use the same choices of multipliers as in our convergence study for the linear advection equation above, and present errors ϵ_1 in the $\mathcal{L}^1$ norm at $t = 0.3$, before the shock wave has formed. No limiter is used in these tests. From these tables we see that the modified scheme indeed retains the usual order of convergence for this nonlinear problem, for any choices of the multipliers α_k. We again observe that the performance of the DG

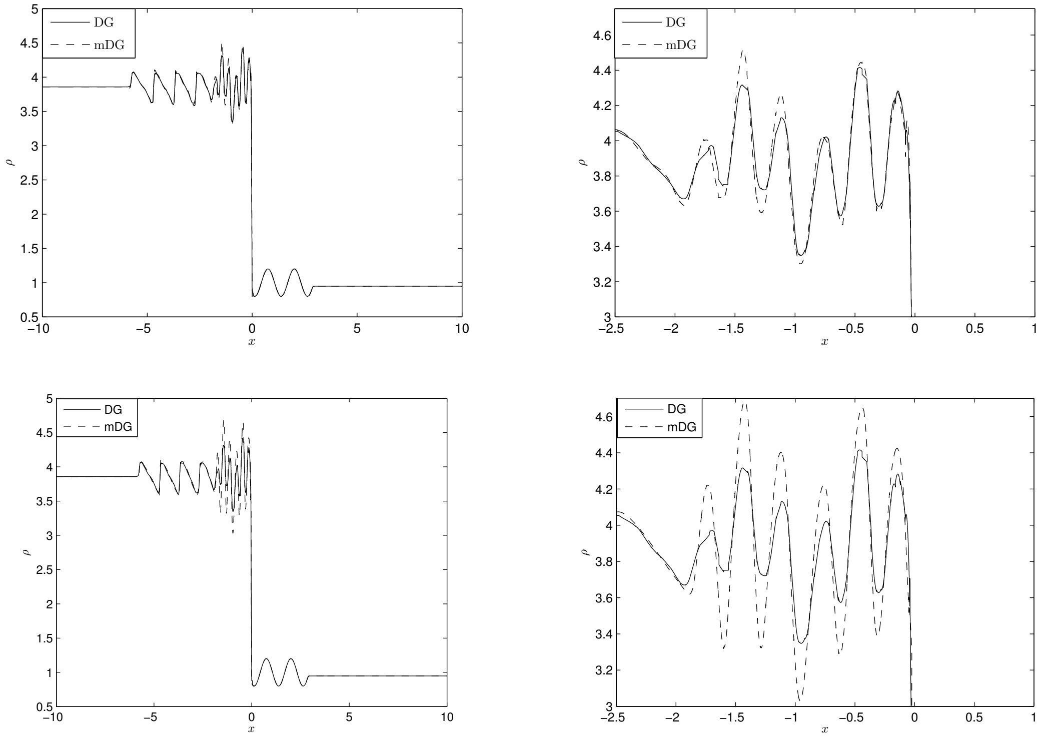


Figure 3.6: Euler equations, (3.5.9)- (3.5.10), $p = 2$, shown at $t = 2$. Top: DG and mDG with $\alpha_2 = \frac{1}{5}$ on a mesh of $N = 500$ elements. Bottom: DG on a 500-element mesh and mDG with $\alpha_2 = \frac{1}{5}$, on a mesh of $N = 866$ elements. Right plots are zooms of left plots.

scheme is roughly the same with that of the mDG method with increased CFL number for a fixed computation effort.

3.5.4 Performance on Nonlinear Systems

To test the modified DG method for a system of equations, we consider the Euler equations, $\mathbf{u}_t + \mathbf{f}(\mathbf{u})_x = 0$ with

$$\mathbf{u} = (\rho, \rho q, E)^T, \quad \mathbf{f}(\mathbf{u}) = q\mathbf{u} + (0, P, qP)^T, \tag{3.5.9a}$$

and an equation of state

$$P = (\gamma - 1)\left(E - \frac{1}{2}\rho q^2\right), \tag{3.5.9b}$$

for which we take $\gamma = 1.4$, and subject to the initial data [63]

$$(\rho, q, P)(x, 0) = \begin{cases} (3.857143, -0.920279, 10.333333), & x \leq 0, \\ (1 + 0.2\sin(5x), -3.549648, 1.000000), & 0 < x < 10, \\ (1.000000, -3.549648, 1.000000), & x \geq 10. \end{cases} \tag{3.5.10}$$

This example involves the interaction of a stationary shock at $x = 0$ with a leftward-moving flow having a sinusoidal density variation. As the density perturbation passes through the shock, it produces oscillations developing into shocks of smaller amplitude. We choose this test problem since it gives us a good example of the interaction between a shock and the fine structure of the produced oscillations. In our tests we chose to use the moment limiter [46]. In Figure 3.6, we present the numerical solutions of the $p = 2$ scheme at $t = 2$. In the top left figure we show the unmodified DG scheme, $\alpha_2 = 1$ with $CFL = \frac{1}{5}$, and the modified scheme with $\alpha_2 = \frac{1}{5}$ and $CFL = \frac{3}{5}$, on a mesh of $N = 500$ elements. In the top right figure we show a zoomed view of the fine structure of the solution to the left of the shock wave. In each figure we show the schemes together with a reference solution computed using the DG scheme with $p = 2$ and $N = 2500$ with the moment limiter. Surprisingly, the mDG solution is more accurate, i.e. suffers from less numerical diffusion. While a rigorous explanation of this is still an open question, one possible explanation is that the limiter destroys some of the accuracy of the fine structure at each iteration. Hence, since the modified solution is obtained using a larger time-step, the solution is less damaged by the limiter and is able to better resolve the fine structure to the left of the shock wave. Finally, in the bottom left figure we show again the unmodified DG scheme, $\alpha_2 = 1$ with $CFL = \frac{1}{5}$, on the same mesh of $N_{DG} = 500$ elements, together with the modified scheme with $\alpha_2 = \frac{1}{5}$ and $CFL = \frac{3}{5}$ on a mesh of $N_{mDG} = 866 \approx \sqrt{3}N_{DG}$ elements. The bottom right figure shows a zoomed view of the fine structure of the solution. This example demonstrates the increase in accuracy we can obtain by implementing the modified DG scheme on a refined mesh, for equivalent computation effort.

3.6 Discussion

In this chapter, we have proposed a family of numerical schemes obtained through a modification of the discontinuous Galerkin finite element method. It is known that the choice of numerical flux influences the spectrum of the DG scheme. For example, the central flux results in the spectrum being entirely located on the imaginary axis and the upwind flux produces a spectrum which lies in the left half-plane and grows with order of approximation p. Here, we propose a modification to the DG scheme that does not change the type of flux, but rather alters the contribution of this flux to the solution coefficients c_{jk}. This modification is obtained by multiplying the jump contributions of the numerical flux for the solution coefficient c_{jk} by a multiplier α_k. Since for one-dimensional problems, with a basis of Legendre polynomials, the coefficient c_{jk} is a numerical approximation of the k-th derivative of the solution on cell j, scaled by $C_k h^k$ where C_k is a constant, our method modifies the amount of numerical flux that is being contributed the k-th derivative of the solution. In the specific case that $\alpha_k = 1, \forall k$, we obtain the usual DG method.

The modifications to the DG scheme can also be viewed as changing how the δ-functions at the cell interfaces are projected into the finite element space. By extending the analysis performed in Chapter 2, we have shown that these modifications alter the superconvergence properties of the scheme. Specifically, by using classical Fourier analysis we have shown that the Fourier modes of the numerical method which are polynomials are closely related to a rational approximation $\frac{\tilde{f}(\omega h_j,\xi)}{\tilde{g}(\omega h_j)}$ of the exponential function. The order of approximation of this rational function is determined by the orthogonality properties of the δ-function projection. Moreover, this rational function has a local expansion in terms of the degree $p+1$ polynomial $\tilde{R}_{p+1}(\xi;\boldsymbol{\alpha})$ and its antiderivatives. We have shown that for a family of initial projections on a uniform mesh the superconvergent points of the numerical solution will tend exponentially quickly towards the roots of this polynomial. Therefore these modifications reveal a strong connection between the projection of the δ-functions at the cell interfaces and the superconvergence properties of the method.

The results of our study of this modified method can be summarized as follows. Firstly, the modification of the lowest order coefficient α_0 in the order p scheme immediately results in a severe accuracy loss and the order of convergence of the scheme is reduced by one. We therefore avoid such a modification and focus on modifying only the equations for the higher-order coefficients of the scheme. Secondly, by analyzing how the modified scheme performs on the linear advection equation we can establish that when the coefficient α_k is modified the order of accuracy of dispersion and dissipation of the scheme is $p+k$, i.e. the accuracy is reduced from the usual accuracy of order $2p+1$ in dissipation and order $2p+2$ in dispersion. This reduction of accuracy introduces additional dispersive and diffusive errors to the numerical solution. Thirdly, when modifying only the highest multiplier we can prove that the method is linearly stable for any choice of α_p. Furthermore, we can expect to obtain a more relaxed stability restriction by choosing $\alpha_p \approx 0.4$. The relaxed condition allows us to take a time step twice as large, compared to the usual DG method. Finally, more multipliers may be altered and a larger improvement in the usual CFL number can be made for specific choices of α_k, but as more multipliers are altered the accuracy of the scheme is reduced as more dispersion and dissipation errors are added. Additionally, the increased time step introduces a larger temporal error into the solution.

We present a number of numerical experiments demonstrating the performance of the mDG method. In our examples, the mDG method preserves the convergence rate of the original DG method in the usual L^1 norm. For the linear advection equation with a very smooth profile, the mDG method performs similarly to the DG method for a fixed computational effort, i.e. the number of cells times the number of time steps. On the other hand, when the solution has discontinuities and limiters are applied, the mDG scheme provides a comparable solution on the same mesh, but in less computation time. Additionally, fewer time steps results in less limiting which can result in fine structures of the solution from being overly smoothed by the limiter. In particular, for the Euler equations example, the mDG method results in a better solution with the CFL number being three times larger than in the usual DG method.

Our numerical test have also shown that it is possible to alter the δ-function projection so as to obtain a modified scheme which has certain superconvergence properties. In particular, we can create schemes whose associated rational approximation of the exponential function has a particular form/order, or we can design schemes so that the numerical solution has specific superconvergent points. In doing so, however, care must be taken to ensure that the downwind point remains $\mathcal{O}(h^{p+2})$ or else this error will dominate.

It is hoped that further study will illuminate a better understanding of the effects of these modifications to the DG method. In particular, more testing is necessary to determine what choices of the multipliers will be optimal in the sense of the trade-off between accuracy and the CFL number. It would also be useful to compare the mDG scheme with finite-volume and finite-difference schemes in terms of accuracy. Additionally, the results in Tables 3.1 and 3.2 indicate that there may be a pattern in the choices of the multipliers α_k which give us the largest CFL improvement, as p increases. This suggests that these choices may be related to some specific rational approximation of $\exp(z)$. Further, the optimal choices of the multipliers α_k should also be investigated with the application of different limiters in the presence of shock waves.

Chapter 4

Superconvergence and Superaccuracy of the DG Method on Triangular Elements

4.1 Introduction

In this chapter we will extend the superconvergence, superaccuracy, and stability analysis presented in Chapter 2 to the DG scheme for two-dimensional problems on triangular elements. We again simplify the analysis by considering the application of the DG method to a linear problem

$$u_t + \mathbf{a} \cdot \nabla u = 0, \tag{4.1.1}$$

where $\mathbf{a} = (a, b)$, on a two-dimensional domain $\Omega \subset \mathbb{R}^2$ subject to the initial condition $u(x, y, 0) = u_0(x, y)$ and suitable boundary conditions.

We derive a PDE which is solved by the numerical solution itself and apply classical Fourier analysis to find the Fourier modes of this PDE that are polynomial in space. Analogously to the one-dimensional case, we find that the Fourier modes of the numerical solution are completely determined by a projection of the inflow into the cell and are rational functions of the mode's frequency ω and a parameter h_j. Geometrically, h_j is shown to be the width of the cell Ω_j along the direction of flow $\mathbf{a}$. We use these Fourier modes to investigate the superconvergence, superaccuracy, and stability of the method. Specifically, we determine the local superconvergence properties of the method on triangular cells by assuming exact inflow into a cell. Then, by considering a simple uniform mesh of triangles, we use these Fourier modes to symbolically calculate the numerical dispersion relation of the method and verify that the numerical dispersion relation agrees with the exact dispersion relation to order $2p+1$, establishing the superaccuracies of the method in terms of dissipation and dispersion errors. Finally, we investigate the global superconvergence properties of the method by using the numerical dispersion relation to show that the spectrum of the method can be decomposed into physical and non-physical modes. The non-physical modes are damped out exponentially quickly in time, and the physical modes are advected with high-order accuracy. We then symbolically calculate the superconvergence properties of physical modes to establish the global superconvergence properties of the DG method on this uniform mesh.

We also derive a new, tighter CFL condition for the DG method for two-dimensional problems. The stability condition which is usually implemented when the DG discretization is paired with an explicit Runge-Kutta method can be written as

$$\Delta t \leq \frac{1}{2p+1} \min_j \frac{r_j}{||\mathbf{a}_j||},$$

where r_j is the radius of the inscribed circle in each element and $||\mathbf{a}_j||$ is the largest wave speed. This condition was proposed and supported with numerical evidence by Cockburn et al in [23] and provides a stable time step. However, it is known to not be a tight bound [50] and in some cases a much larger stable time step exists. Here, we note that the appearance of the parameter h_j in the Fourier modes of the numerical solution has the effect of scaling the size of the spectrum. Consequently, we propose that a more natural CFL condition for the method can be written as

$$\Delta t \leq CFL \ \min_j \frac{h_j}{||\mathbf{a}_j||}.$$

When pairing the spatial discretization with an explicit Runge-Kutta-$(p+1)$ time integration, our numerical experiments have revealed that taking

$$CFL = \frac{1}{(2p+1)\left(1+\frac{4}{(p+2)^2}\right)}$$

provides a fairly tight bound on the time step Δt.

Previous studies of the superconvergence properties of the DG method in two dimensions have been applied to two types of mesh elements: quadrilaterals and triangles. Following the one-dimensional superconvegence study of Adjerid et al in [4] where the authors found that the local error of the DG scheme is superconvergent at the right-based Radau points, Adjerid and Massey [5] performed the natural extension of these results to the DG scheme on rectangular meshes. The authors showed that the local error of the DG scheme on rectangular cells with a tensor product basis is spanned by two degree $p+1$ Radau polynomials in the x and y directions. The local error of the DG scheme on triangular elements was then studied by Krivodonova and Flaherty in [47] and by Adjerid and Baccouch in [2, 3]. In this chapter we use the Fourier modes of the numerical solution to provide new simple proofs of the local superconvergence results presented in these works.

This chapter is organized as follows. In Section 4.2 we apply the DG method to a simple linear advection problem and derive a PDE for the numerical solution. In Section 4.3 we apply Fourier analysis to this PDE and derive the Fourier modes of the numerical solution that are polynomials in space. We then use these Fourier modes in Section 4.4 to give simple proofs of the local superconvergence properties of the DG method on triangular elements. We then consider a uniform mesh and prove the superaccuracy of the DG method in terms of the dissipation and dispersion errors in Section 4.5. In Section 4.6 we examine the spectrum of the DG method on this uniform mesh and show that our proposed CFL condition arises naturally when considering the stability of the method. We then proceed in Section 4.7 to establish that the spectrum of the method on this uniform mesh can be partitioned into physical and non-physical frequencies and use this result to establish several global superconvergence properties of the method. Finally, we provide several numerical examples in Section 4.8 which confirm the superconvergence properties of the method as well as demonstrate the efficacy of the proposed CFL condition.

4.2 The DG method in 2D

We begin our analysis as in 1D by applying the DG scheme to the linear problem (4.1.1) on a mesh of triangles and derive a PDE for the numerical solution. First, we discretize the domain Ω into a mesh of N triangles $\Omega_j, j = 1, \ldots, N$. Recall that we consider the equation (4.1.1) over a single cell Ω_j and map this cell using the mapping (1.2.17) to a canonical triangle Ω_0 whose vertices are located at (0,0), (1,0), and (0,1). The Jacobian matrix for this transformation is constant and given in (1.2.18). Upon mapping this linear problem (4.1.1) to the canonical triangle we obtain the scaled problem

$$u_t + \boldsymbol{\alpha} \cdot \nabla u = 0, \tag{4.2.1}$$

where the ∇ operator is now understood as a gradient in (ξ, η)-space and

$$\boldsymbol{\alpha} = (\alpha, \beta) = \mathbf{a}(J_j^{-1})^T. \tag{4.2.2}$$

Applying the formulation of the DG scheme in two dimensions (1.2.23) to this scaled problem and using the upwind flux, i.e

$$U_j^* = \begin{cases} U_j & \boldsymbol{\alpha} \cdot \mathbf{n} \geq 0, \\ U_{j+} & \boldsymbol{\alpha} \cdot \mathbf{n} < 0, \end{cases} \tag{4.2.3}$$

we obtain the DG scheme for this linear problem

$$\frac{d}{dt} c_{jki} + \oint_{\partial\Omega_0} (\boldsymbol{\alpha} \cdot \mathbf{n}) U_j^* \psi_{ki} \, ds - \iint_{\Omega_0} \boldsymbol{\alpha} \cdot \nabla \psi_{ki} U_j \, dA = 0, \tag{4.2.4}$$

for $k = 0, \ldots, p$ and $i = 0, \ldots, k$, recalling that the ψ_{ki} are the orthonormal polynomial basis functions defined in (1.2.20) and U_{j+} is the value of the numerical solution in the immediate neighbour of Ω_j along each edge of its boundary $\partial\Omega_j$. We now proceed to find the Fourier modes of the scheme by deriving a PDE for the DG solution satisfied by the polynomial U_j. To this end, we apply the divergence theorem to the volume integral in (4.2.4) and move the boundary integrals to the right hand side to obtain

$$\frac{d}{dt} c_{jki} + \iint_{\Omega_0} \boldsymbol{\alpha} \cdot \nabla U_j \psi_{ki} \, dA = - \oint_{\partial\Omega_0} (\boldsymbol{\alpha} \cdot \mathbf{n}) [[U_j]] \psi_{ki} \, ds, \tag{4.2.5}$$

where $[[U_j]] = U_j^* - U_j$ is the jump between the Riemann states and the numerical solution on the boundary. Note that from the choice of the upwind flux (4.2.3) this jump is zero along edges where $\boldsymbol{\alpha} \cdot \mathbf{n} \geq 0$, i.e. outflow edges. Hence, we can partition the boundary of Ω_0 as $\partial\Omega_0 = \partial\Omega_0^+ \cup \partial\Omega_0^-$, where $\partial\Omega_0^-$ is the inflow boundary along which $\boldsymbol{\alpha} \cdot \mathbf{n} < 0$, and $\partial\Omega_0^+$ is the outflow boundary. Since $U_j^* = U_j$ along the out flow boundaries, the integral along the entire boundary $\partial\Omega_0$ in (4.2.5) reduces to the integral along the inflow boundary $\partial\Omega_0^-$.

We will make a simplifying assumption about which edges of $\partial\Omega_0$ are outflow edges. We label the edges of the canonical triangle Ω_0 travelling counter-clockwise as E_1, E_2, and E_3, where the first edge, E_1, is the edge connecting (0,0) to (1,0). We then assume, without loss of generality, that the vertices of Ω_j have been indexed specifically so that when Ω_j is mapped to the canonical cell Ω_0 the second edge E_2 is either the only inflow edge or the only outflow edge. That is, we assume either α and β have the same sign, or one of them is equal to zero.

We proceed by multiplying (4.2.5) by ψ_{ki}, summing over $k = 0, \ldots, p$ and $i = 0, \ldots, k$ and, using (1.2.22), obtain an equation for $\frac{\partial}{\partial t} U_j$

$$\frac{\partial}{\partial t} U_j + \sum_{k=0}^{p} \sum_{i=0}^{k} \left[\iint_{\Omega_0} \boldsymbol{\alpha} \cdot \nabla U_j \psi_{ki} \, dA \right] \psi_{ki} = - \sum_{k=0}^{p} \sum_{i=0}^{k} \left[\int_{\partial\Omega_0^-} (\boldsymbol{\alpha} \cdot \mathbf{n}) [[U_j]] \psi_{ki} \, ds \right] \psi_{ki}.$$

Note that because $\{\psi_{ki}\}$ is an orthonormal family the first sum in this expression is a projection of ∇U_j into the space of polynomials $\mathbf{P}_p$. Furthermore, since U_j is a polynomial of degree at most p, $\boldsymbol{\alpha} \cdot \nabla U_j$ will be polynomial of degree $p-1$ and the projection will be exact. Therefore we can write

$$\frac{\partial}{\partial t} U_j + \boldsymbol{\alpha} \cdot \nabla U_j = -\sum_{k=0}^{p} \sum_{i=0}^{k} \left[\int_{\partial \Omega_0^-} (\boldsymbol{\alpha} \cdot \mathbf{n})[[U_j]] \psi_{ki} \, ds \right] \psi_{ki}. \tag{4.2.6}$$

Hence, we obtain a PDE that is solved exactly by the polynomial numerical approximation U_j on cell Ω_j. This PDE is equivalent to the original linear problem, except for a forcing term proportional to the size of the jumps at the cell boundaries. Note that the jump is a function of ξ and η. We now proceed to find exact solutions of this PDE using classical Fourier analysis.

4.3 Fourier Analysis

We look for a single Fourier mode solution of (4.2.6) of the form $U_j(\xi, \eta, t) = \hat{U}_j(\xi, \eta) e^{-||\mathbf{a}||\omega t}$ where $\hat{U}_j(\xi, \eta)$ is a polynomial in ξ and η and $||\mathbf{a}||$ is the L^2 norm of the flow vector $\mathbf{a}$. Using this assumption in (4.2.6) we have that the Fourier mode satisfies

$$-||\mathbf{a}||\omega \hat{U}_j + \boldsymbol{\alpha} \cdot \nabla \hat{U}_j = -\sum_{k=0}^{p} \sum_{i=0}^{k} \left[\int_{\partial \Omega_0^-} (\boldsymbol{\alpha} \cdot \mathbf{n})[[\hat{U}_j]] \psi_{ki} \, ds \right] \psi_{ki}. \tag{4.3.1}$$

To proceed, we make a change of variables that transforms this PDE into an ODE. The change of variables is

$$\begin{pmatrix} \zeta \\ \sigma \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ \theta - 1 & \theta \end{pmatrix} \begin{pmatrix} \xi \\ \eta \end{pmatrix}, \tag{4.3.2}$$

where $\theta = \frac{\alpha}{\alpha + \beta}$ is a parameter in $[0, 1]$ which gives a measure of the flow direction $\boldsymbol{\alpha}$. We will use the check accent to denote objects in this new coordinates. Hence, we denote Ω_0 in these new coordinates as $\check{\Omega}_0$ and observe that $\check{\Omega}_0$ has vertices at $(0,0)$, $(1, \theta - 1)$, and $(1, \theta)$. Note also that the transformation (4.3.2) preserves the area of Ω_0. The flow direction $\check{\boldsymbol{\alpha}}$ in the (ζ, σ)-coordinates is given by

$$\begin{aligned} \check{\boldsymbol{\alpha}} &= \boldsymbol{\alpha} \begin{pmatrix} 1 & \theta - 1 \\ 1 & \theta \end{pmatrix} \\ &= (\alpha + \beta, 0). \end{aligned}$$

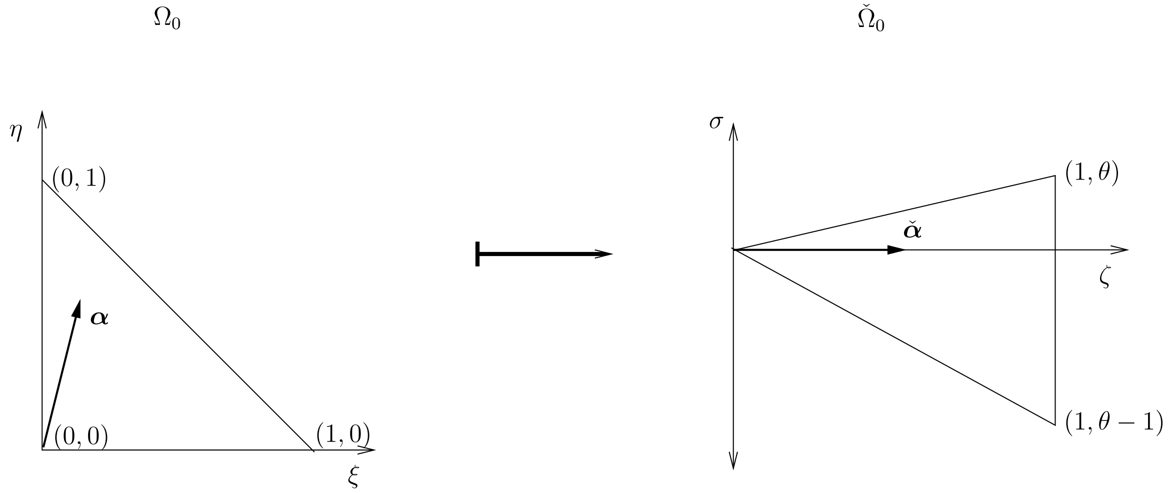


Figure 4.1: The transformation (4.3.2) maps the coordinate system so that the direction of flow is along the ζ-axis.

Hence, $\check{\Omega}_0$ is aligned such that the flow direction $\check{\boldsymbol{\alpha}}$ is along the ζ-axis (see Figure 4.1). Consequently, upon transforming the ∇ operator to the (ζ, σ) coordinates we find that the operator $\boldsymbol{\alpha} \cdot \nabla$ in the PDE (4.3.1) becomes $(\alpha + \beta)\frac{\partial}{\partial \zeta}$ in the new coordinate system.

We also introduce a new small parameter $h_j = \frac{||\mathbf{a}||}{\alpha+\beta}$ which will tend zero under mesh refinement. In fact, h_j is the width of the cell Ω_j along the direction of flow $\mathbf{a}$. To see this, first note that from the definition of the scaled velocity $\boldsymbol{\alpha}$ in (4.2.2) we can write

$$\alpha = \frac{a(y_3 - y_1) - b(x_3 - x_1)}{(x_2 - x_1)(y_3 - y_1) - (x_3 - x_1)(y_2 - y_1)},$$

$$\beta = \frac{-a(y_2 - y_1) + b(x_2 - x_1)}{(x_2 - x_1)(y_3 - y_1) - (x_3 - x_1)(y_2 - y_1)}.$$

Hence, we can write

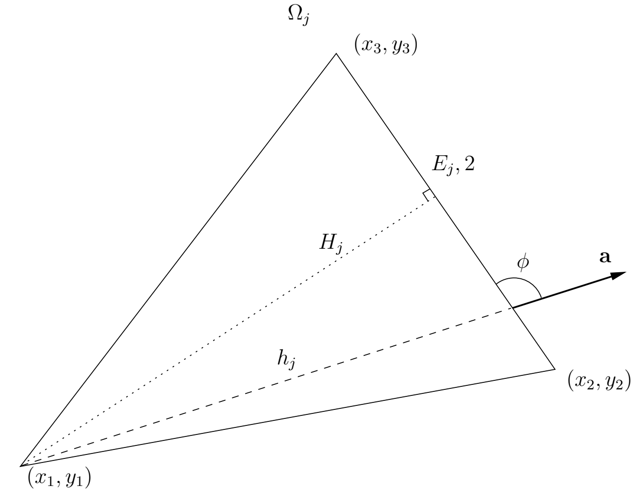


Figure 4.2: Diagram of the cell Ω_j showing the parameter h_j. We see that h_j is the width of Ω_j along the direction of flow $\mathbf{a}$.

$$\begin{aligned} h_j &= \frac{||\mathbf{a}||}{\alpha + \beta} \\ &= ||\mathbf{a}|| \frac{(x_2 - x_1)(y_3 - y_1) - (x_3 - x_1)(y_2 - y_1)}{a(y_3 - y_2) - b(x_3 - x_2)} \\ &= ||\mathbf{a}|| \frac{\det J_j}{\mathbf{a} \cdot (y_3 - y_2, -(x_3 - x_2))} \\ &= \frac{2|\Omega_j|}{|E_{j,2}| \sin \phi}. \end{aligned} \tag{4.3.3}$$

Here we have used the notation that ϕ is the angle between $\mathbf{a}$ and $E_{j,2}$, where $E_{j,2}$ is the side of Ω_j which connects (x_2, y_2) and (x_3, y_3) (see Figure 4.2). We have also used that $\det J_j = 2|\Omega_j|$. If we write the area of cell Ω_j as $|\Omega_j| = \frac{1}{2}|E_{j,2}|H_j$, where H_j is the height of cell Ω_j measured from the vertex (x_1, y_1) to the edge $E_{j,2}$, then we see that (4.3.3) implies that $h_j \sin \phi = H_j$. From the definition of ϕ we see that h_j is the width of cell Ω_j along the direction of flow $\mathbf{a}$.

Transforming the PDE (4.3.1) to this new coordinate system, and multiplying the entire expression by $\frac{h_j}{||\mathbf{a}||}$, we obtain the following ODE for the Fourier modes of the numerical solution

$$-\omega h_j \hat{U}_j + \frac{\partial}{\partial \zeta} \hat{U}_j = -\sum_{k=0}^{p} \sum_{i=0}^{k} \left[\int_{\partial \check{\Omega}_0^-} n_\zeta [[\hat{U}_j]] \check{\psi}_{ki} \, ds \right] \check{\psi}_{ki}. \tag{4.3.4}$$

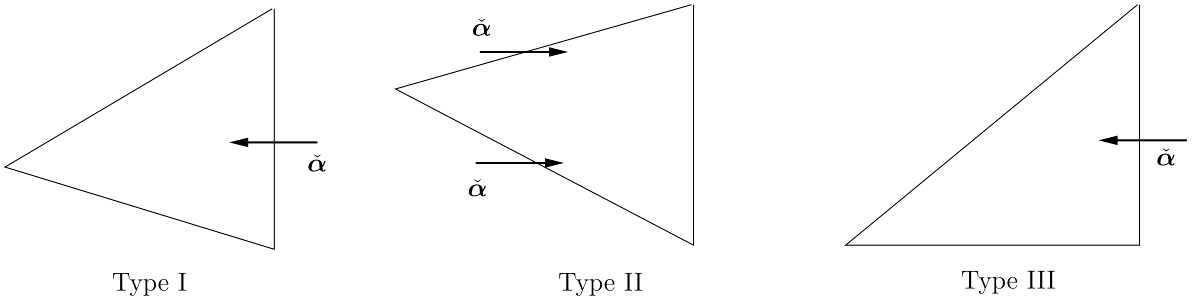

$\check{\boldsymbol{\alpha}}$
Type I
$\check{\boldsymbol{\alpha}}$
$\check{\boldsymbol{\alpha}}$
Type II
$\check{\boldsymbol{\alpha}}$
Type III

Figure 4.3: We label cells Type I, II, or III, based on the direction of the flow $\boldsymbol{\alpha}$ into the cell.

Here we have used the notation that n_ζ is the first coordinate of the normal vector $\check{\mathbf{n}} = (n_\zeta, n_\sigma)^T$ in the (ζ, σ)-coordinates, and $\check{\psi}_{ki}$ are the polynomial basis functions now evaluated using the (ζ, σ)-coordinates. The Fourier mode $\hat{U}_j$ is now understood to be evaluated in (ζ, σ)-coordinates as well.

Next, we aim to write the right-hand side of (4.3.4) in a more compact form. Specifically, we want to express this forcing term as the ζ-derivative of some polynomial $\mathcal{R}_{p+1}$, as it was done in one dimension with the right Radau polynomial. We notice, however, that because of the integral in (4.3.4) is over the inflow boundary $\check{\Omega}_0^-$ this polynomial $\mathcal{R}_{p+1}$ necessarily depends on how many inflow edges $\check{\Omega}_0$ has. Hence, first we must separate this problem into cases depending upon the number of inflow edges of the cell $\check{\Omega}_j$. We label these cases in the same way as in [2, 3] and denote a cell with only one inflow edge a type I cell, a cell with two inflow edges a type II cell, and a cell with a characteristic edge a type III cell (see Figure 4.3). Since, by assumption, α and β have the same sign we see that $\alpha, \beta > 0$ corresponds to a type I cell, $\alpha, \beta < 0$ corresponds to a type II cell, and the special case when α or β is zero corresponds to a type III cell. Using this labelling, we write the forcing on the right hand side of this equation in a more useful form in following proposition.

Proposition 4.1. *We define a projection of the jump function $[[\hat{U}_j]]$, which we denote as $\mathcal{R}_{p+1}[[\hat{U}_j]](\zeta, \sigma)$, into the space of polynomials in ζ and σ satisfying the following conditions,*

$$\int_{\partial\check{\Omega}_0^+} n_\zeta \mathcal{R}_{p+1}[[\hat{U}_j]]\check{\psi}_{ki}\, ds = \iint_{\check{\Omega}_0} \mathcal{R}_{p+1}[[\hat{U}_j]]\frac{\partial}{\partial\zeta}\check{\psi}_{ki}\, dA, \tag{4.3.5}$$

for $k = 0, \ldots, p$ and $i = 0, \ldots, k$, and

$$\int_{\partial\check{\Omega}_0^-} n_\zeta \left(\mathcal{R}_{p+1}[[\hat{U}_j]]\right)\check{\psi}_{ki}\, ds = \int_{\partial\check{\Omega}_0^-} n_\zeta [[\hat{U}_j]]\check{\psi}_{ki}\, ds, \tag{4.3.6}$$

for $k = 0, \ldots, p$ and $i = 0, \ldots, k$. When Ω_j is a type I or III cell we require that $\mathcal{R}_{p+1}[[\hat{U}_j]](\zeta, \sigma)$ is a polynomial of degree $p+1$ in ζ and of degree p in σ. When Ω_j is a type II cell we require that $\mathcal{R}_{p+1}[[\hat{U}_j]](\zeta, \sigma)$ is a polynomial of degree $p+1$ in ζ and of degree $2p$ in σ. $\mathcal{R}_{p+1}[[\hat{U}_j]](\zeta, \sigma)$ is then uniquely determined by (4.3.5) *and* (4.3.6)*.*

Then, using this projection, the forcing term on the right hand side of (4.3.4) *can be written as*

$$\sum_{k=0}^{p} \sum_{i=0}^{k} \left[\int_{\partial \check{\Omega}_0^-} n_\zeta [[\hat{U}_j]] \check{\psi}_{ki} \, ds \right] \check{\psi}_{ki} = \frac{\partial}{\partial \zeta} \mathcal{R}_{p+1}[[\hat{U}_j]](\zeta, \sigma). \tag{4.3.7}$$

Proof. We first confirm the existence of such a projection by checking that (4.3.5) and (4.3.6) are sufficient to uniquely define $\mathcal{R}_{p+1}[[\hat{U}_j]]$. First, $\mathcal{R}_{p+1}[[\hat{U}_j]]$ is required to be a polynomial of degree $p+1$ in ζ and either degree p or $2p$ in σ depending on what type of cell Ω_j is. For now, let us say $\mathcal{R}_{p+1}[[\hat{U}_j]]$ is of degree M_p in σ. We can then write $\mathcal{R}_{p+1}[[\hat{U}_j]]$ as a linear combination of monomials

$$\mathcal{R}_{p+1}[[\hat{U}_j]](\zeta, \sigma) = \sum_{k=0}^{p} \sum_{i=0}^{k} r_{k-i+1,i} \zeta^{k-i+1} \sigma^i + \sum_{i=0}^{M_p} r_{0,i} \sigma^i. \tag{4.3.8}$$

From this we see that $\mathcal{R}_{p+1}[[\hat{U}_j]]$ contains $\frac{1}{2}(p+1)(p+2)$ monomials which are of degree at least 1 in ζ and $M_p + 1$ monomials in only σ. Hence $\mathcal{R}_{p+1}[[\hat{U}_j]]$ can be uniquely determined through $\frac{1}{2}(p+1)(p+2) + M_p + 1$ independent equations. From the orthogonality of the basis functions ψ_{ki} we see that (4.3.5) contains $\frac{1}{2}(p+1)(p+2)$ independent conditions on $\mathcal{R}_{p+1}[[\hat{U}_j]]$. It therefore only remains to determine the $M_p + 1$ additional independent conditions.

Examining equation (4.3.6) we see that this expression involves an integral of each basis function $\check{\psi}_{ki}$ along the inflow boundary $\partial \check{\Omega}_0^-$. Note that since $\check{\psi}_{ki}$ span all polynomials of degree p in $\check{\Omega}_0$, the restriction of these basis functions to a single edge (parametrized by a single variable, say s) spans all polynomials of degree p in the variable s. Therefore, if Ω_j is a type I or III cell the inflow boundary $\partial \check{\Omega}_0^-$ will consists of only a single edge and (4.3.6) gives $p+1$ independent conditions on $\mathcal{R}_{p+1}[[\hat{U}_j]]$. Hence, when Ω_j is a type I or III cell we have $M_p = p$ and (4.3.6) provides the remaining $p+1$ conditions necessary to uniquely determine $\mathcal{R}_{p+1}[[\hat{U}_j]]$.

Similarly, the restriction of the basis functions $\check{\psi}_{ki}$ to an inflow boundary which consists of two edges will span a space of dimension $2p+1$. To see this note that on a single edge the basis functions span all polynomials of degree p or less, but every basis function is continuous on the whole cell $\check{\Omega}_0$. Hence, the restriction of the basis functions to two edges (again parametrized by a variable s) will span all piecewise continuous polynomials of degree p or less on each edge. Therefore, we obtain that if Ω_j is a type II cell the inflow boundary $\partial\check{\Omega}_0^-$ will consists of two edges and (4.3.6) gives $2p+1$ independent conditions on $\mathcal{R}_{p+1}[[\hat{U}_j]]$. Hence, when Ω_j is a type II cell we have $M_p = 2p$ and (4.3.6) provides the remaining $2p+1$ conditions necessary to uniquely determine $\mathcal{R}_{p+1}[[\hat{U}_j]]$.

Finally, having shown the existence of $\mathcal{R}_{p+1}[[\hat{U}_j]]$ through the relations (4.3.6) and (4.3.6), we proceed to verify (4.3.7). Writing $\mathcal{R}_{p+1}[[\hat{U}_j]]$ as a sum of monomials in (4.3.8) we see that $\frac{\partial}{\partial\zeta}\mathcal{R}_{p+1}[[\hat{U}_j]]$ is a polynomial of degree p in both ζ and σ,

$$\frac{\partial}{\partial\zeta}\mathcal{R}_{p+1}[[\hat{U}_j]](\zeta,\sigma) = \sum_{k=0}^{p}\sum_{i=0}^{k}(k-i+1)r_{k-i+1,i}\zeta^{k-i}\sigma^i. \tag{4.3.9}$$

Therefore, we can verify (4.3.7) by multiplying the expression by $\check{\psi}_{ki}$, integrating over $\check{\Omega}_0$, applying the divergence theorem, and using the orthogonality relations (4.3.5) to obtain

$$\begin{aligned}\int_{\partial\check{\Omega}_0^-} n_\zeta[[\hat{U}_j]]\check{\psi}_{ki}\,ds &= \iint_{\check{\Omega}_0}\left(\frac{\partial}{\partial\zeta}\mathcal{R}_{p+1}[[\hat{U}_j]]\right)\check{\psi}_{ki}\,dA \\ &= \oint_{\partial\check{\Omega}_0} n_\zeta\mathcal{R}_{p+1}[[\hat{U}_j]]\check{\psi}_{ki}\,ds - \iint_{\check{\Omega}_0}\mathcal{R}_{p+1}[[\hat{U}_j]]\frac{\partial}{\partial\zeta}\check{\psi}_{ki}\,dA. \\ &= \int_{\partial\check{\Omega}_0^-} n_\zeta[[\hat{U}_j]]\check{\psi}_{ki}\,ds,\end{aligned}$$

which is true by (4.3.6). □

The orthogonality relations (4.3.5) which define the projection $\mathcal{R}_{p+1}$ were also considered in an analogous form when the local error of the DG method applied to a elliptic boundary value problem was examined by Krivodonova and Flaherty in [47] and Ajerid and Baccouch in [2]. We will make use of some of their results in order to establish certain properties of the projection $\mathcal{R}_{p+1}$.

Proposition 4.2. *When Ω_j is a cell of type I the projection $\mathcal{R}_{p+1}[[\hat{U}_j]]$ of the jump function along the inflow boundary satisfies the following orthogonality relation*

$$\int_{\partial\check{\Omega}_0^+} n_\zeta\mathcal{R}_{p+1}[[\hat{U}_j]]\sigma^k\,ds = 0, \tag{4.3.10}$$

for all $k = 0,\ldots,p$.

Proof. Cf. Ajerid and Baccouch [2]. □

Proposition 4.3. *When Ω_j is a cell of type II or III the projection $\mathcal{R}_{p+1}[[\hat{U}_j]]$ of the jump function along the inflow boundary satisfies the following orthogonality relations*

$$\iint_{\check{\Omega}_0} \mathcal{R}_{p+1}[[\hat{U}_j]]\check{\psi}_{ki}\, dA = 0, \tag{4.3.11}$$

for all $k = 0, \ldots, p-1$ and $i = 0, \ldots, k$, and

$$\int_{\partial\check{\Omega}_0^+} n_\zeta \mathcal{R}_{p+1}[[\hat{U}_j]]\check{\psi}_{ki}\, ds = 0, \tag{4.3.12}$$

for all $k = 0, \ldots, p$ and $i = 0, \ldots, k$.

Proof. Cf. Krivodonova and Flaherty [47]. □

Using Proposition 4.1, we can write (4.3.4) in the following compact form

$$-\omega h_j \hat{U}_j + \frac{\partial}{\partial \zeta}\hat{U}_j = -\frac{\partial}{\partial \zeta}\mathcal{R}_{p+1}[[\hat{U}_j]]. \tag{4.3.13}$$

We can solve this ODE exactly by integrating from the inflow boundary $\partial\check{\Omega}_0^-$. For clarity, let us parametrize the inflow boundary as $\partial\check{\Omega}_0^- = \{(\zeta_0(\sigma), \sigma)\}$, observing from Figure 4.1 that $(\theta - 1) \leq \sigma \leq \theta\}$. From Figure 4.1 we also see that when Ω_j is a type I or III cell we have

$$\zeta_0(\sigma) = 1$$

and when Ω_j is a type II cell we have

$$\zeta_0(\sigma) = \begin{cases} \frac{\sigma}{\theta} & 0 \leq \sigma \leq \theta, \\ \frac{\sigma}{\theta-1} & \theta - 1 \leq \sigma \leq 0. \end{cases}$$

Using this notation, we solve (4.3.13) exactly by considering σ as a fixed parameter and integrating the ODE from the boundary $\partial\check{\Omega}_0^-$ to write the solution as

$$\hat{U}_j(\zeta, \sigma) = \hat{U}_j(\zeta_0, \sigma)e^{\omega h_j(\zeta - \zeta_0)} - \int_{\zeta_0}^{\zeta} \frac{\partial}{\partial z}\mathcal{R}_{p+1}[[\hat{U}_j]](z, \sigma)e^{\omega h_j(\zeta - z)}\, dz. \tag{4.3.14}$$

These exact solutions are the general Fourier modes of the DG scheme in two dimensions. The modes consist of an exact advection of the initial value $\hat{U}_j(\zeta_0, \sigma)$ on $\partial\check{\Omega}_0^-$ along the direction of flow and an integral term resulting from the forcing term in (4.3.13). However, these solutions are not necessarily polynomials in ζ and σ. We therefore will look for particular solutions which are polynomials. First, however, let us define a new space

$$\mathbf{P}_p^- = \text{span}\{\check{\psi}_{ki}(\zeta_0, \sigma) | k = 0, \ldots, p, \text{and } i = 0, \ldots, k\}. \tag{4.3.15}$$

This is the restriction of the polynomial space $\mathbf{P}_p$ to the inflow boundary $\partial\check{\Omega}_0^-$. As noted in the proof of Proposition 4.1, when Ω_j is a type I or III cell the inflow boundary $\partial\check{\Omega}_0^-$ consists of a single edge and the space $\mathbf{P}_p^-$ has dimension $p+1$. On the other hand, when Ω_j is a type II cell the inflow boundary $\partial\check{\Omega}_0^-$ consists of two edges and the space $\mathbf{P}_p^-$ has dimension $2p+1$. Next, let us define a projection $\mathcal{I}_p$ into this space.

Definition 2. *Let $\mathcal{I}_p V(\zeta_0, \sigma)$ be the projection of the function V into the space $\mathbf{P}_p^-$, defined by*

$$\int_{\partial\Omega_0^-} (\mathcal{I}_p V)\, \psi_{ki}\, ds = \int_{\partial\Omega_0^-} V \psi_{ki}\, ds,$$

for $k = 0, \ldots, p$ and $i = 0, \ldots, k$.

Notice that if the function V is in the space $\mathbf{P}_p^-$ then $\mathcal{I}_p$ acts as an identity operator. Next, we state two lemmas which will help to investigate the general solutions (4.3.14) of the PDE (4.3.13).

Lemma 4.1. *The integral term in* (4.3.14) *can be written as*

$$\int_{\zeta_0}^{\zeta} \frac{\partial}{\partial z} \mathcal{R}_{p+1}[[\hat{U}_j]](z, \sigma) e^{\omega h_j (\zeta - z)}\, dz = \frac{1}{(\omega h_j)^{p+1}} \left[\mathcal{F}_p[[\hat{U}_j]](\zeta_0, \sigma) e^{\omega h_j (\zeta - \zeta_0)} - \mathcal{F}_p[[\hat{U}_j]](\zeta, \sigma) \right], \tag{4.3.16}$$

where

$$\mathcal{F}_p[[\hat{U}_j]](\zeta, \sigma) = \sum_{k=1}^{p+1} (\omega h_j)^{p+1-k} \frac{\partial^k}{\partial \zeta^k} \mathcal{R}_{p+1}[[\hat{U}_j]](\zeta, \sigma). \tag{4.3.17}$$

Proof. Viewing σ as a fixed parameter in this integral, we can prove this lemma using the same procedure as the proof of Lemma 2.1 in Chapter 2. Namely, we integrate the integral term in (4.3.14) by parts repeatedly. □

By its construction in (4.3.17), $\mathcal{F}_p[[\hat{U}_j]]$ is a polynomial of degree p in ζ and σ and also a polynomial of degree p in ωh_j. Furthermore, since we can view $\mathcal{R}_{p+1}$ as a projection operator, we can also view $\mathcal{F}_p$ as a projection operator to the space $\mathbf{P}_p$.

Lemma 4.2. *The integral term in* (4.3.14) *can also be written as*

$$\int_{\zeta_0}^{\zeta} \frac{\partial}{\partial z}\mathcal{R}_{p+1}[[\hat{U}_j]](z,\sigma)e^{\omega h_j(\zeta-z)}\,dz = -\mathcal{R}_{p+1}[[\hat{U}_j]](\zeta_0,\sigma)e^{\omega h_j(\zeta-\zeta_0)} + \mathcal{R}_{p+1}[[\hat{U}_j]](\zeta,\sigma) + \sum_{k=1}^{\infty}(\omega h_j)^k\mathcal{R}_{p+1}^{(-k)}[[\hat{U}_j]](\zeta,\sigma), \quad (4.3.18)$$

where $\mathcal{R}_{p+1}^{(-k)}[[\hat{U}_j]]$ *are the repeated anti-derivatives of the polynomial* $\mathcal{R}_{p+1}[[\hat{U}_j]]$ *and can be written using the Cauchy integration formula as*

$$\mathcal{R}_{p+1}^{(-k)}[[\hat{U}_j]](\zeta,\sigma) = \frac{1}{(k-1)!}\int_{\zeta_0}^{\zeta}(\zeta-z)^{k-1}\mathcal{R}_{p+1}[[\hat{U}_j]](z,\sigma)\,dz. \quad (4.3.19)$$

Proof. As in Lemma 4.1, (4.3.18) is obtained by integrating the integral term in (4.3.14) by parts repeatedly, this time in the opposite direction. □

Using these lemmas we can establish our first result.

Theorem 4.1. *The Fourier modes of the DG method* (4.2.4) *for linear hyperbolic problems in two dimensions which are polynomials in* ζ *and* σ *can be written as*

$$\hat{U}_j(\zeta,\sigma) = \mathcal{F}_p \circ \mathcal{G}_p^{-1}\hat{U}_{j+}, \quad (4.3.20)$$

where $\circ$ *denote the composition of operators. Here* $\mathcal{G}_p^{-1}\hat{U}_{j+}$ *is a projection of the inflow function* U_{j+} *to* $\mathbf{P}_p^-$ *which satisfies* $[(\omega h_j)^{p+1}\mathcal{I}_p + \ \mathcal{F}_p] \circ \mathcal{G}_p^{-1}\hat{U}_{j+} = \mathcal{I}_p\hat{U}_{j+}$. *In addition to being in* $\mathbf{P}_p$, *these modes are rational functions of* ωh_j. *They also have the following expansion*

$$\hat{U}_j(\zeta,\sigma) = \hat{U}_{j+}(\zeta_0,\sigma)e^{\omega h_j(\zeta-\zeta_0)} + \left[\mathcal{R}_{p+1}[[\hat{U}_j]](\zeta_0,\sigma) - [[\hat{U}_j]]\right]e^{\omega h_j(\zeta-\zeta_0)} - \mathcal{R}_{p+1}[[\hat{U}_j]](\zeta,\sigma) - \sum_{k=1}^{\infty}(\omega h_j)^k\mathcal{R}_{p+1}^{(-k)}[[\hat{U}_j]](\zeta,\sigma). \quad (4.3.21)$$

Proof. Using Lemma 1 in the general solution (4.3.14) to the PDE (4.3.13) we find that the solutions can be written

$$\hat{U}_j(\zeta,\sigma) = \hat{U}_j(\zeta_0,\sigma)e^{\omega h_j(\zeta-\zeta_0)} - \frac{1}{(\omega h_j)^{p+1}}\left[\mathcal{F}_p[[\hat{U}_j]](\zeta_0,\sigma)e^{\omega h_j(\zeta-\zeta_0)} - \mathcal{F}_p[[\hat{U}_j]](\zeta,\sigma)\right].$$

Examining this expression, we see that solutions of (4.3.13) will be polynomial in ζ and σ when

$$(\omega h_j)^{p+1}\hat{U}_j(\zeta_0,\sigma) = \mathcal{F}_p[[\hat{U}_j]](\zeta_0,\sigma), \tag{4.3.22}$$

is satisfied. These polynomial solutions have the form

$$\hat{U}_j(\zeta,\sigma) = \frac{1}{(\omega h_j)^{p+1}}\mathcal{F}_p[[\hat{U}_j]](\zeta,\sigma). \tag{4.3.23}$$

Adding $(\omega h_j)^{p+1}\mathcal{I}_p[[\hat{U}_j]]$ to both sides of (4.3.22), and using the fact that from the definition of the jump function $[[\hat{U}_j]]$ we have that $\mathcal{I}_p[[\hat{U}_j]] = \mathcal{I}_p\hat{U}_{j+} - \hat{U}_j$, we obtain that

$$(\omega h_j)^{p+1}\mathcal{I}_p\hat{U}_{j+1}(\zeta_0,\sigma) = \left[(\omega h_j)^{p+1}\mathcal{I}_p + \mathcal{F}_p\right][[\hat{U}_j]](\zeta_0,\sigma). \tag{4.3.24}$$

At this point we define a new projection, denoted by $\mathcal{G}_p^{-1}\hat{U}_{j+}$, of the inflow function $\hat{U}_{j+}$ to $\mathbf{P}_p^-$ which satisfies

$$\int_{\partial\Omega_0^-}\left(\left[(\omega h_j)^{p+1}\mathcal{I}_p + \mathcal{F}_p\right]\circ\mathcal{G}_p^{-1}\hat{U}_{j+}\right)\psi_{ki}\,ds = \int_{\partial\Omega_0^-}\hat{U}_{j+}\psi_{ki}\,ds,$$

for $k = 0,\ldots,p$, and $i = 0,\ldots,k$. That is, this projection is defined so that $[(\omega h_j)^{p+1}\mathcal{I}_p + \mathcal{F}_p]\circ\mathcal{G}_p^{-1}\hat{U}_{j+} = \mathcal{I}_p\hat{U}_{j+}$. Using this new projection, the expression (4.3.24) can be rewritten as

$$\mathcal{I}_p[[\hat{U}_j]] = (\omega h_j)^{p+1}\mathcal{G}_p^{-1}\hat{U}_{j+}.$$

and using this in (4.3.23) (first noting that, by the definition of $\mathcal{I}_p$, $\mathcal{F}_p[[\hat{U}_j]] = \mathcal{F}_p\circ\mathcal{I}_p[[\hat{U}_j]]$) we have that polynomial solutions of (4.3.13) can be written

$$\hat{U}_j(\zeta,\sigma) = \mathcal{F}_p\circ\mathcal{G}_p^{-1}\hat{U}_{j+}.$$

which establishes (4.3.20).

Finally, we can establish the expansion (4.3.21) by using Lemma 4.2 in the general solution (4.3.14) to the PDE (4.3.13) to obtain

$$\begin{aligned}\hat{U}_j(\zeta,\sigma) = \hat{U}_j(\zeta_0,\sigma)e^{\omega h_j(\zeta-\zeta_0)} &+ \mathcal{R}_{p+1}[[\hat{U}_j]](\zeta_0,\sigma)e^{\omega h_j(\zeta-\zeta_0)} \\ &- \mathcal{R}_{p+1}[[\hat{U}_j]](\zeta,\sigma) - \sum_{k=1}^{\infty}(\omega h_j)^k\mathcal{R}_{p+1}^{(-k)}[[\hat{U}_j]](\zeta,\sigma).\end{aligned}$$

Adding and subtracting $\hat{U}_{j+}(\zeta_0,\sigma)e^{\omega h_j(\zeta-\zeta_0)}$ from the right hand side of this equation yields (4.3.21). □

From (4.3.20) in this theorem, we have that the Fourier modes of the DG scheme applied to the linear problem (4.1.1) take the form of rational functions ωh_j on each cell. Moreover, we have that these rational functions have the expansion (4.3.21) in ωh_j. This expansion consists of an exact advection of the inflow function along the direction of flow $\hat{U}_{j+} e^{\omega h_j(\zeta-\zeta_0)}$, and higher-order terms involving the projection $\mathcal{R}_{p+1}[[\hat{U}_j]]$ of the jump function along the inflow boundary.

4.4 Local Superconvergent Error

Using the results from Theorem 4.1, we can state new and simple proofs of the local superconvergence properties of the DG method applied to linear hyperbolic problems in two dimensions.

Theorem 4.2. *Suppose Ω_j is a type I cell and the suppose the inflow function into Ω_j is exact and given by $\hat{U}_{j+} = e^{\omega h_j \zeta_0 + \kappa h_j \sigma}$. Then the local error $\epsilon_j = \hat{U}_j - \hat{u}$ of the DG method* (4.2.4) *applied to the linear problem* (4.1.1) *in this cell has the expansion*

$$\epsilon_j(\zeta,\sigma) = -\mathcal{R}_{p+1}[[\hat{U}_j]](\zeta,\sigma) - \gamma_{p+1} P_{p+1}(2\sigma - 2\theta + 1) + \mathcal{O}(h_j^{p+2}), \tag{4.4.1}$$

where

$$\gamma_{p+1} = (2p+3) \int_{\partial\check{\Omega}_0^-} \hat{U}_{j+} P_{p+1}(2\sigma - 2\theta + 1)\, ds. \tag{4.4.2}$$

The error also satisfies

$$\int_{\partial\check{\Omega}_0^+} n_\zeta \epsilon_j \sigma^m \, ds = \mathcal{O}(h_j^{p+2}). \tag{4.4.3}$$

for $m = 0, \ldots, p$.

Proof. Using the inflow function $\hat{U}_{j+} = e^{\omega h_j \zeta_0 + \kappa h_j \sigma}$ in the expansion (4.3.21) of the Fourier modes we obtain that

$$\begin{aligned}\hat{U}_j(\zeta,\sigma) = e^{\omega h_j \zeta + \kappa h_j \sigma} + \Big[\mathcal{R}_{p+1}[[\hat{U}_j]](\zeta_0,\sigma) - [[\hat{U}_j]]\Big]\, e^{\omega h_j(\zeta-\zeta_0)} \\ - \mathcal{R}_{p+1}[[\hat{U}_j]](\zeta,\sigma) - \sum_{k=1}^{\infty} (\omega h_j)^k \mathcal{R}_{p+1}^{(-k)}[[\hat{U}_j]](\zeta,\sigma),\end{aligned}$$

and since the exact advection of this inflow is $\hat{u} = e^{\omega h_j \zeta + \kappa h_j \sigma}$ we see that

$$
\begin{aligned}
\epsilon_j(\zeta,\sigma) = \Big[\mathcal{R}_{p+1}[[\hat{U}_j]](\zeta_0,\sigma) - [[\hat{U}_j]]\Big]\, e^{\omega h_j(\zeta-\zeta_0)} \\
- \mathcal{R}_{p+1}[[\hat{U}_j]](\zeta,\sigma) - \sum_{k=1}^{\infty}(\omega h_j)^k \mathcal{R}_{p+1}^{(-k)}[[\hat{U}_j]](\zeta,\sigma). \quad (4.4.4)
\end{aligned}
$$

Next, from the definition of the projection $\mathcal{R}_{p+1}$, and the fact that $\partial\check{\Omega}_0^-$ consists of only one edge, we have that $\mathcal{R}_{p+1}[[\hat{U}_j]](\zeta_0,\sigma) = \mathcal{I}_p[[\hat{U}_j]]$. Hence,

$$
\begin{aligned}
\mathcal{R}_{p+1}[[\hat{U}_j]](\zeta_0,\sigma) - [[\hat{U}_j]] &= \mathcal{I}_p[[\hat{U}_j]](\zeta_0,\sigma) - [[\hat{U}_j]] \\
&= \mathcal{I}_p\hat{U}_{j+} - \hat{U}_{j+} \\
&= -\gamma_{p+1}P_{p+1}(2\sigma - 2\theta + 1) + \mathcal{O}(h_j^{p+2}),
\end{aligned}
$$

where γ_{p+1} is $\mathcal{O}(h_j^{p+1})$ and can be found by (4.4.2). Using this leading order estimate in (4.4.4) we obtain (4.4.1).

We can establish (4.4.3) by multiplying (4.4.4) by $n_\zeta\sigma^m$ and integrating over $\partial\check{\Omega}_0^+$. Then, using the orthogonality property (4.3.10) of the projection $\mathcal{R}_{p+1}[[\hat{U}_j]]$ shown in Proposition 4.2, we can write

$$
\int_{\partial\check{\Omega}_0^+} n_\zeta \epsilon_j \sigma^m \, ds = -\int_{\partial\check{\Omega}_0^+} n_\zeta \gamma_{p+1} P_{p+1}(2\sigma - 2\theta + 1) e^{\omega h_j(\zeta-\zeta_0)} \sigma^m \, ds + \mathcal{O}(h_j^{p+2}).
$$

Expanding $e^{\omega h_j(\zeta-\zeta_0)}$ as $1 + \omega h_j(\zeta - \zeta_0) + \mathcal{O}(h_j^2)$, we see that the leading order term on the integrand on the right hand side of this expression has no dependence on ζ. We can therefore write the integral along this boundary as an integral in σ to obtain

$$
\begin{aligned}
\int_{\partial\check{\Omega}_0^+} n_\zeta \epsilon_j \sigma^m \, ds &= -\int_{\partial\check{\Omega}_0^+} n_\zeta \gamma_{p+1} P_{p+1}(2\sigma - 2\theta + 1)\sigma^m \, ds + \mathcal{O}(h_j^{p+2}) \\
&= \int_{\theta-1}^{\theta} \gamma_{p+1} P_{p+1}(2\sigma - 2\theta + 1)\sigma^m \, d\sigma + \mathcal{O}(h_j^{p+2}) \\
&= \mathcal{O}(h_j^{p+2}),
\end{aligned}
$$

where in the last line the integral vanishes by the orthogonality of the Legendre polynomial P_{p+1}. □

Theorem 4.3. *Suppose Ω_j is a type II cell and the suppose the inflow function into Ω_j is exact and given by $\hat{U}_{j+} = e^{\omega h_j \zeta_0 + \kappa h_j \sigma}$. Then the local error $\epsilon_j = \hat{U}_j - \hat{u}$ of the DG method* (4.2.4) *applied to the linear problem* (4.1.1) *in this cell has the expansion*

$$\epsilon_j(\zeta,\sigma) = -\mathcal{R}_{p+1}[[\hat{U}_j]](\zeta,\sigma) + \mathcal{O}(h_j^{p+2}). \tag{4.4.5}$$

The error also satisfies

$$\int_{\partial\check{\Omega}_0^+} \epsilon_j \sigma^m \, ds = \mathcal{O}(h_j^{2p+2-m}), \tag{4.4.6}$$

for $m = 0, \ldots, p$ and

$$\iint_{\check{\Omega}_0} \epsilon_j \check{\psi}_{mi} \, ds = \mathcal{O}(h_j^{2p+1-m}), \tag{4.4.7}$$

for $m = 0, \ldots, p-1$ and $i = 0, \ldots, m$.

Proof. Again, using the inflow function $\hat{U}_{j+} = e^{\omega h_j \zeta_0 + \kappa h_j \sigma}$ in the expansion (4.3.21) of the Fourier modes we obtain the expansion of the error ϵ_j, (4.4.4). In this case, however, since in its definition the projection $\mathcal{R}_{p+1}[[\hat{U}_j]]$ is polynomial of degree $p+1$ in ζ and $2p$ in σ, we have that $\mathcal{R}_{p+1}[[\hat{U}_j]](\zeta_0,\sigma) = [[\hat{U}_j]] + \mathcal{O}(h_j^{p+2})$. Hence we immediately obtain (4.4.5). To prove (4.4.6), we multiply (4.4.4) by σ^m and integrate along the outflow edge $\partial\check{\Omega}_0^+$ and use the orthogonality property (4.3.12) of $\mathcal{R}_{p+1}[[\hat{U}_j]]$ to obtain

$$\begin{aligned}\int_{\partial\check{\Omega}_0^+} \epsilon_j \sigma^m \, ds = \int_{\partial\check{\Omega}_0^+} \Big[\mathcal{R}_{p+1}[[\hat{U}_j]](\zeta_0,\sigma) - [[\hat{U}_j]]\Big] e^{\omega h_j(\zeta-\zeta_0)} \sigma^m \, ds \\ - \sum_{k=1}^{\infty} (\omega h_j)^k \int_{\partial\check{\Omega}_0^+} \mathcal{R}_{p+1}^{(-k)}[[\hat{U}_j]](\zeta,\sigma) \sigma^m \, ds.\end{aligned} \tag{4.4.8}$$

Since $\zeta = 1$ along $\partial\check{\Omega}_0^+$, we can show that the first integral on the right hand side of (4.4.8) vanishes to $\mathcal{O}(h_j^{2p+2-m})$ by re-writing it as an integral along $\partial\check{\Omega}_0^-$ in the following way,

$$\begin{aligned}&\int_{\partial\check{\Omega}_0^+} \Big[\mathcal{R}_{p+1}[[\hat{U}_j]](\zeta_0,\sigma) - [[\hat{U}_j]]\Big] e^{\omega h_j(\zeta-\zeta_0)} \sigma^m \, ds \\ &\qquad = \int_{\partial\check{\Omega}_0^+} \Big[\mathcal{R}_{p+1}[[\hat{U}_j]](\zeta_0,\sigma) - [[\hat{U}_j]]\Big] e^{\omega h_j(1-\zeta_0)} \sigma^m \, ds \\ &\qquad = -\int_{\partial\check{\Omega}_0^-} n_\zeta \Big[\mathcal{R}_{p+1}[[\hat{U}_j]](\zeta_0,\sigma) - [[\hat{U}_j]]\Big] e^{\omega h_j(1-\zeta_0)} \sigma^m \, ds.\end{aligned}$$

Expanding $e^{\omega h_j(1-\zeta_0)} = 1 + \omega h_j(1-\zeta_0) + \ldots$, and using the property (4.3.6) from the defintition of $\mathcal{R}_{p+1}$ we get that the integral of the first $p+1-m$ terms in this expansion will vanish, and hence the whole integral is $\mathcal{O}(h_j^{2p+2-m})$.

We then show that the sum on the left hand side of (4.4.8) vanishes to $\mathcal{O}(h_j^{2p+2-m})$ by using the definition of the $\mathcal{R}_{p+1}^{(-k)}[[\hat{U}_j]]$ projections in (4.3.19) in order to write

$$\begin{aligned}\int_{\partial\check{\Omega}_0^+} \mathcal{R}_{p+1}^{(-k)}[[\hat{U}_j]](\zeta,\sigma)\sigma^m\, ds &= \oint_{\partial\check{\Omega}_0} n_\zeta \mathcal{R}_{p+1}^{(-k)}[[\hat{U}_j]](\zeta,\sigma)\sigma^m\, ds \\ &= \oint_{\partial\check{\Omega}_0} \frac{n_\zeta}{(k-1)!}\int_{\zeta_0}^{\zeta}(\zeta-z)^{k-1}\mathcal{R}_{p+1}[[\hat{U}_j]](z,\sigma)\sigma^m\, dz\, ds \\ &= \frac{1}{(k-1)!}\iint_{\check{\Omega}_0}(\zeta-z)^{k-1}\sigma^m\mathcal{R}_{p+1}[[\hat{U}_j]](z,\sigma)\, dA.\end{aligned}$$

Finally, by the orthogonality property of $\mathcal{R}_{p+1}[[\hat{U}_j]]$ in (4.3.5) we have that this vanishes for $k+m \le p$ and hence the entire sum in (4.4.8) vanishes to $\mathcal{O}(h_j^{2p+2-m})$, and we have established (4.4.6).

We follow a similar argument to prove (4.4.7). We multiply (4.4.4) by $\zeta^{m-l}\sigma^l$, where $l \le m$ and $m \le p-1$, integrate over $\check{\Omega}_0$, and use the orthogonality of $\mathcal{R}_{p+1}[[\hat{U}_j]]$ in (4.3.5) to obtain

$$\begin{aligned}\iint_{\check{\Omega}_0}\epsilon_j\zeta^{m-l}\sigma^l\, dA = \iint_{\check{\Omega}_0}&\left[\mathcal{R}_{p+1}[[\hat{U}_j]](\zeta_0,\sigma) - [[\hat{U}_j]]\right]e^{\omega h_j(\zeta-\zeta_0)}\zeta^{m-l}\sigma^l\, dA \\ &- \sum_{k=1}^{\infty}(\omega h_j)^k\iint_{\check{\Omega}_0}\mathcal{R}_{p+1}^{(-k)}[[\hat{U}_j]](\zeta,\sigma)\zeta^{m-l}\sigma^l\, dA. \quad (4.4.9)\end{aligned}$$

We then show that the first integral term on the right hand side of (4.4.9) is $\mathcal{O}(h_j^{2p+1-m})$ by using the divergence theorem to write

$$\begin{aligned}&\iint_{\check{\Omega}_0}\left[\mathcal{R}_{p+1}[[\hat{U}_j]](\zeta_0,\sigma) - [[\hat{U}_j]]\right]e^{\omega h_j(\zeta-\zeta_0)}\zeta^{m-l}\sigma^l\, dA = \\ &\qquad\frac{1}{\omega h_j}\oint_{\partial\check{\Omega}_0} n_\zeta\left[\mathcal{R}_{p+1}[[\hat{U}_j]](\zeta_0,\sigma) - [[\hat{U}_j]]\right]e^{\omega h_j(\zeta-\zeta_0)}\zeta^{m-l}\sigma^l\, ds \\ &\qquad - \frac{m-l}{\omega h_j}\iint_{\check{\Omega}_0}\left[\mathcal{R}_{p+1}[[\hat{U}_j]](\zeta_0,\sigma) - [[\hat{U}_j]]\right]e^{\omega h_j(\zeta-\zeta_0)}\zeta^{m-l-1}\sigma^l\, dA. \quad (4.4.10)\end{aligned}$$

By a similar argument used above, the surface integral on the right hand side is $\mathcal{O}(h_j^{2p+1-l})$. Applying the divergence theorem again, we will find that the surface integral term is now

$\mathcal{O}(h_j^{2p+1-l})$. Hence, repeatedly applying the divergence theorem to this integral a total of $m-l$ times, we will arrive at a surface integral which will be $\mathcal{O}(h_j^{2p+1-m})$.

The second integral term on the right hand side of (4.4.9) can be seen to be $\mathcal{O}(h_j^{2p+1-m})$ by a similar argument. We apply the divergence theorem to write

$$\iint_{\breve{\Omega}_0} \mathcal{R}_{p+1}^{(-k)}[[\hat{U}_j]](\zeta,\sigma)\zeta^{m-l}\sigma^l \, dA = \oint_{\partial\breve{\Omega}_0} n_\zeta \mathcal{R}_{p+1}^{(-k-1)}[[\hat{U}_j]](\zeta,\sigma)\zeta^{m-l}\sigma^l \, ds \\ -(m-l)\iint_{\breve{\Omega}_0} \mathcal{R}_{p+1}^{(-k-1)}[[\hat{U}_j]](\zeta,\sigma)\zeta^{m-l-1}\sigma^l \, dA. \quad (4.4.11)$$

Again, by a similar argument used above we know that the surface integral in this expression will vanish when $k+m \leq p-1$. Applying the divergence theorem again we will again find that the surface integral will vanish when $k+m \leq p-1$ and, therefore, by repeatedly applying the divergence theorem we can conclude that the entire expression will vanish for $k+m \leq p-1$ and the sum on the right hand side of (4.4.9) will vanish to $\mathcal{O}(h_j^{2p+1-m})$ which concludes the proof. □

Theorem 4.4. *Suppose Ω_j is a type III cell and the suppose the inflow function into Ω_j is exact and given by $\hat{U}_{j+} = e^{\omega h_j \zeta_0 + \kappa h_j \sigma}$. Then the local error $\epsilon_j = \hat{U}_j - \hat{u}$ of the DG method* (4.2.4) *applied to the linear problem* (4.1.1) *in this cell has the expansion*

$$\epsilon_j(\zeta,\sigma) = -\mathcal{R}_{p+1}[[\hat{U}_j]](\zeta,\sigma) - \gamma_{p+1}P_{p+1}(2\sigma - 2\theta + 1) + \mathcal{O}(h_j^{p+2}), \quad (4.4.12)$$

where

$$\gamma_{p+1} = (2p+3)\int_{\partial\breve{\Omega}_0^-} \hat{U}_{j+}P_{p+1}(2\sigma - 2\theta + 1)\, ds. \quad (4.4.13)$$

The error also satisfies

$$\int_{\partial\breve{\Omega}_0^+} \epsilon_j \sigma^m \, ds = \mathcal{O}(h_j^{2p+2-m}), \quad (4.4.14)$$

for $m = 0, \ldots, p$ and

$$\iint_{\breve{\Omega}_0} \epsilon_j \breve{\psi}_{mi} \, ds = \mathcal{O}(h_j^{2p+1-m}), \quad (4.4.15)$$

for $m = 0, \ldots, p-1$ and $i = 0, \ldots, m$.

Proof. The proof of this theorem combines elements from the proofs of the previous theorems in this section. To begin, we can establish expressions (4.4.12) and (4.4.13) in an entirely analogous way as (4.4.1) and (4.4.2) were established in the proof of Theorem 4.2.

We then establish the orthogonality properties (4.4.14) and (4.4.15) through an argument analogous to the one used to prove the orthogonality properties (4.4.6) and (4.4.7) in the proof of Theorem 4.3, instead using that the inflow edge is located along $\zeta_0 = 1$. The full argument is repetitive and is omitted. □

4.5 Superaccuracy

While the expansion of the Fourier modes found in Theorem 4.1 proves useful in determining the local superconvergence properties of the DG solution, the Fourier modes (4.3.20) are themselves useful in investigating the superaccuracies of the method in terms of dissipation and dispersion errors and in analysing the spectrum of the method. To do this, let us consider the linear problem (4.1.1) on the unit square domain Σ with periodic boundary conditions. We consider a particularly simple uniform computational mesh found by partitioning Σ into $N \times M$ rectangles $\Sigma_{lj} = [x_l, x_{l+1}] \times [y_j, y_{j+1}]$ of size $\Delta x \times \Delta y$, then dividing each square into two triangles along lines connecting the points (x_{l+1}, y_j) and (x_l, y_{j+1}). We begin by mapping each rectangle Σ_{lj} to a canonical square element $\Sigma_0 = [0, 1] \times [0, 1]$ using the mapping

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} \Delta x \xi + x_l \\ \Delta y \eta + y_j \end{pmatrix}. \tag{4.5.1}$$

On each cell, the flow direction is given by $\boldsymbol{\alpha} = \left[\frac{a}{\Delta x}, \frac{b}{\Delta y}\right]$. For convenience we will assume that the mesh is refined in such a way that $\frac{\Delta x}{\Delta y}$ remains constant so that the direction of this vector remains constant under refinement. We label the two triangles which make up this canonical square Ω_1 and Ω_2. Now, as is usual in the analysis of dispersion and dissipation errors, we assume that the numerical solution has the form of a plane wave along each cell. In particular, we assume the numerical solution $U_{lj}(\xi, \eta, t)$ on each square Σ_{lj} has the form

$$U_{lj}(\xi, \eta, t) = \hat{U}(\xi, \eta) \exp\left(l \Delta x \tilde{\kappa}_1 + j \Delta y \tilde{\kappa}_2 - ||\mathbf{a}|| \omega t\right). \tag{4.5.2}$$

Here $\tilde{\kappa}_1$ and $\tilde{\kappa}_1$ are numerical wavenumbers. Note that the exact dispersion relation of a plane wave of this form would be $a\kappa_1 + b\kappa_2 = ||\mathbf{a}||\omega$. Our goal in this section is to show that this dispersion relation holds up to order $2p + 2$ for the numerical wavenumbers.

From the analysis above, we can find solutions of the form (4.5.2) on the square Σ_0 by finding the Fourier mode solutions with frequency ω on Ω_1 and Ω_2. These solutions will be completely determined by the inflow into their cells. We compute these solutions symbolically in order to determine a condition on ω, $\tilde{\kappa}_1$ and $\tilde{\kappa}_2$ which must be satisfied

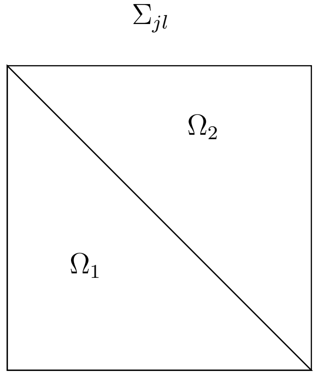


Figure 4.4: Diagram of a square cell Σ_{lj} and the two triangular sub cells Ω_1 Ω_2.

in order for solutions of the form (4.5.2) to exist. We will restrict our attention to the case that $\alpha, \beta > 0$ since every other case can be seen as being equivalent to this case after possibly some linear transformation. When $\alpha, \beta > 0$ we have that the bottom and left edges of Σ_0 are inflow edges which means Ω_1 is a type II cell while Ω_2 is a type I cell. We transform to the (ζ, σ)-coordinate system and consider an inflow $\hat{U}_+$ along $\partial\Omega_1^-$ and compute the numerical solution on the triangle Ω_1 using this inflow. We find the numerical solution on Ω_1 to be rational functions of ωh and θ, where $h = \frac{||\mathbf{a}||\Delta x \Delta y}{a\Delta y + b\Delta x}$ and $\theta = \frac{a\Delta y}{a\Delta y + b\Delta x}$. We then can compute the numerical solution in Ω_2 by using the value of the numerical solution in Ω_1 along its diagonal edge as the inflow to Ω_2. On Ω_2 the numerical solution will be a rational function of $-\omega h$ and θ. In this way we can write the numerical solution $\hat{U}(\xi, \eta)$ over the entire square Σ_0 as a projection of the inflow $\hat{U}_+$ which is polynomial in ξ and η in Ω_1 and Ω_2. We denote this numerical solution on Σ_0 by

$$\hat{U}(\xi, \eta) = \mathcal{H}_p \hat{U}_+(\xi, \eta). \tag{4.5.3}$$

Finally, note that solutions of the form (4.5.2) on different cells are simply scalar multiples of each other. Therefore, since the solutions (4.5.3) are completely determined by their inflow, for these to be of the form (4.5.2) they must satisfy that their value along an outflow edge is a scalar multiple of their value along the opposite inflow edge. We therefore have that solutions of the form (4.5.2) will exist when the system

$$\begin{cases} \mathcal{H}_p \hat{U}_+(1, \eta) = \frac{1}{\nu} \hat{U}_+(0, \eta), \\ \mathcal{H}_p \hat{U}_+(\xi, 1) = \frac{1}{\mu} \hat{U}_+(\xi, 0), \end{cases} \tag{4.5.4}$$

has non-trivial solutions, where $\nu = \exp(-\Delta x \tilde{\kappa}_1)$ and $\mu = \exp(-\Delta y \tilde{\kappa}_2)$. If we view this system as an eigenvalue problem on the inflow polynomial $\hat{U}_+$ we find that the system will have non-trivial solutions when its determinant is zero. Moreover, since the projections are rational functions of ωh the determinant will be a polynomial of h, ω, ν, and μ. We denote the determinant of system (4.5.4) as C_p and compute it symbolically for several values of p. The resulting expressions are quite large and are excluded but we provide the Mathematica source code of the computation of the determinant C_p in the appendix. We use this determinant to propose the following theorem.

Theorem 4.5. *Let U be a numerical solution of the DG method* (4.2.4) *applied to the linear problem*(4.1.1) *on the square domain Σ with a uniform computational mesh and suppose U is of the form* (4.5.2). *Then the numerical wavenumbers $\tilde{\kappa}_1$ and $\tilde{\kappa}_2$ satisfy*

$$a\tilde{\kappa}_1 + b\tilde{\kappa}_2 = ||\mathbf{a}||\omega + \mathcal{O}(h^{2p+1}). \tag{4.5.5}$$

That is, the local orders of errors in dissipation and dispersion of the scheme along the direction of flow are $2p+1$.

We have verified this theorem through symbolic computations for $p \leq 5$ in the following way. Recall that when $\alpha, \beta > 0$ solutions of the form (4.5.2) will exist when the determinant C_p of the system (4.5.4) is equal to zero. This determinant depends on the small parameter $h = \frac{||\mathbf{a}||\Delta x \Delta y}{a\Delta y + b\Delta x}$ and the parameter $\theta = \frac{a\Delta y}{a\Delta y + b\Delta x}$. Therefore, after computing this determinant symbolically we make the substitution $\nu = \exp\left(\frac{\Delta x}{h}\tilde{\kappa}_1\right)$ and $\mu = \exp\left(\frac{\Delta y}{h}\tilde{\kappa}_2\right)$ and form a Taylor expansion of the equation $C_p = 0$ around $h = 0$, recalling that by assumption the mesh is refined in such a way that $\frac{\Delta x}{\Delta y}$ remains constant and therefore $\frac{\Delta x}{h}$, $\frac{\Delta y}{h}$, and θ also remain constant.

Examining this Taylor expansion we find that the constant term is identically zero and the coefficient on h is

$$\theta\frac{\Delta x}{h}\tilde{\kappa}_1 + (1-\theta)\frac{\Delta y}{h}\tilde{\kappa}_2 - \omega.$$

Further more, we find that this expression is a factor in each coefficient of this Taylor expansion up to and including the coefficient of h^{2p+1}. Therefore we find that

$$\theta\frac{\Delta x}{h}\tilde{\kappa}_1 + (1-\theta)\frac{\Delta y}{h}\tilde{\kappa}_2 = \omega + \mathcal{O}(h^{2p+1}),$$

must be true for $C_p = 0$ to hold. Using $h = \frac{||\mathbf{a}||\Delta x \Delta y}{a\Delta y + b\Delta x}$ and $\theta = \frac{a\Delta y}{a\Delta y + b\Delta x}$ we arrive at (4.5.5). We note that this has been verified symbolically up to $p = 5$ but we conjecture that it is true for all p.

4.6 Spectrum of DG in 2D

Let us again consider the DG method for solving (4.1.1) on uniform mesh of the square domain Σ with periodic boundary conditions. In the previous section, we investigated solutions of the form (4.5.2) in order to determine the numerical dispersion relation of the scheme. In this sections, let us again examine the solutions (4.5.2) and note that since the outflow edge of every cell is simply a scalar multiple of its opposite inflow edge, the periodicity of the solution implies that these scalar multiples must be roots of unity. More specifically, we must have that $\nu = \exp(-\Delta x \kappa_1)$ is an N-th root of unity and $\mu = \exp(-\Delta y \kappa_2)$ is an M-th root of unity. Furthermore, since solutions of the form (4.5.2) will exist when the determinant C_p is equal to zero, we can investigate the spectrum of values ω admitted by the scheme on this mesh by finding what values of ω will satisfy $C_p = 0$. In fact, upon computing the determinant C_p we find that that it is a degree $(p+1)(p+2)$ polynomial of ωh. Therefore solving either $C_p = 0$ for the NM possible choices of ν and μ will yield $(p+1)(p+2)NM$ spectrum values. Since there are $(p+1)(p+2)NM$ degree of freedom for the scheme on this mesh, this will be the complete spectrum of the method.

We note that the determinant C_p is a polynomial function of ωh and hence every spectral value for the scheme is scaled by h. This is particularly interesting since geometrically h_j is the length of the cell Ω_j along the direction of flow $\mathbf{a}$ and not the size of the inscribed circle in Ω_j or the length of the smallest edge in Ω_j, which are commonly implemented to scale the minimum timestep Δt. To investigate this in more detail we introduce a variable $\lambda = -\omega h$ to scale the spectral values of the DG method on this mesh. Then, solving $C_p = 0$ with $\nu = \exp\left(\frac{2\pi n i}{N}\right)$ and $\mu = \exp\left(\frac{2\pi m i}{M}\right)$ for every root λ we obtain $(p+1)(p+2)NM$ spectral values which we denote λ_{knm} for $0 \leq k < (p+1)(p+1)$, $n = 0, \ldots, N$, and $m = 0, \ldots, M$. When we pair the DG spatial discretization with an explicit order $p+1$ Runge-Kutta time integration scheme we will have that the scheme will be stable if Δt is sufficiently small so that

$$\frac{||\mathbf{a}||\Delta t \lambda_{knm}}{h_j} \in \mathcal{A}_{p+1},$$

where $\mathcal{A}_{p+1}$ is the absolute stability region of the RK-$(p+1)$ scheme. Note that the spectral values λ_{knm} still depend on the parameter θ which give a measure of the direction of flow in each cell. We demonstrate how the parameter θ alters the spectrum of the method in Figures 4.5 and 4.6 for the $p = 1$ and $p = 2$ schemes. Our numerical computations of the spectral values reveals that the overall size of the spectrum is not very sensitive to this parameter, indicating that the size of the spectrum of the method is determined primarily

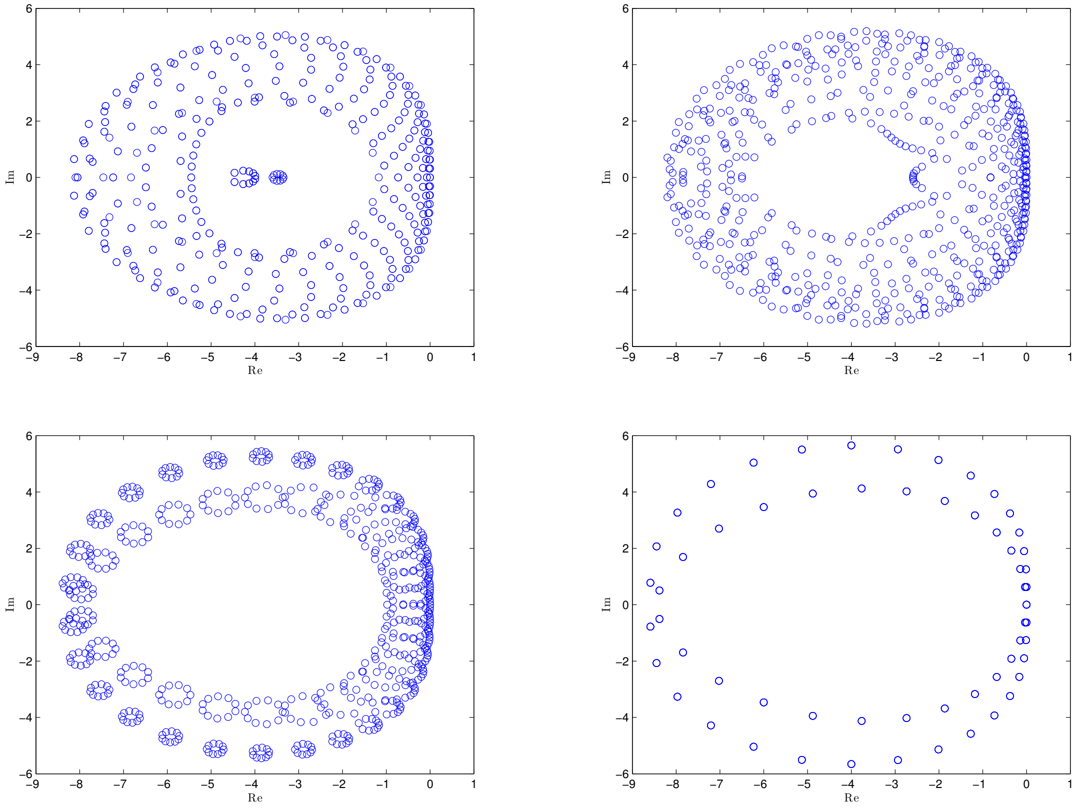


Figure 4.5: Spectral values λ_{knm} of the 2D spatial DG discretization for the linear advection equation, for the $p = 1$ for several values of θ with $N = M = 10$. We show the spectral values with $\theta = 0.5$ and 0.65 (top) and $\theta = 0.85$ and 1 (bottom).

by the parameter h and not by the direction of flow. This is in contrast to previous studies [50, 41] which showed that using parameters such as the smallest cell edge or smallest cell height in the CFL condition leads to CFL numbers which depend heavily on the direction of flow. This strong dependence of the size of the spectrum on the parameter h motivates us to propose a new CFL condition

$$\Delta t \leq CFL \min_j \frac{h_j}{||\mathbf{a}||}, \tag{4.6.1}$$

where the minimum stable time step Δt is now scaled by this new parameter h_j. Our

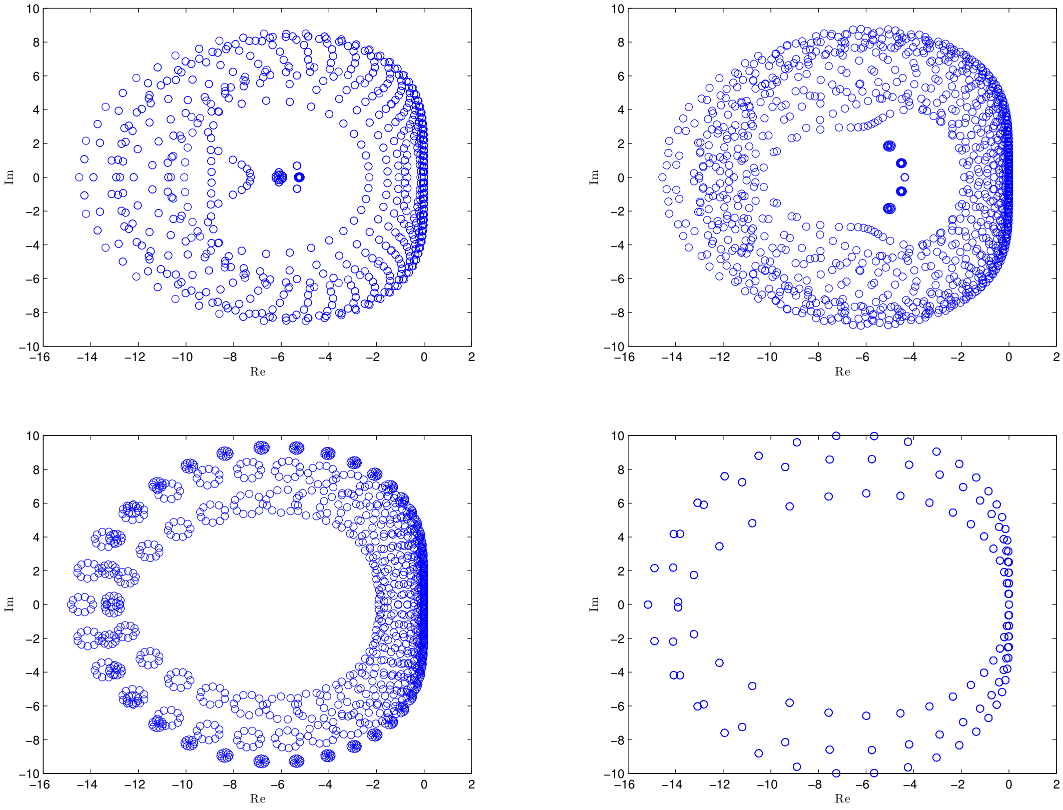


Figure 4.6: Spectral values λ_{knm} of the 2D spatial DG discretization for the linear advection equation, for the $p = 2$ for several values of θ with $N = M = 10$. We show the spectral values with $\theta = 0.5$ and 0.65 (top) and $\theta = 0.85$ and 1 (bottom).

numerical tests have revealed that taking a CFL number given by

$$CFL = \frac{1}{(2p+1)\left(1 + \frac{4}{(p+2)^2}\right)}, \tag{4.6.2}$$

provides a fairly tight bound on the maximum step size Δt, regardless of the value of θ. This has been tested numerically up to $p = 5$. The CFL condition usually implemented for the DG scheme would bound Δt roughly as $\frac{1}{2p+1}\frac{r_j}{||\mathbf{a}||}$ where r_j is the radius of the inscribed circle in Ω_j or the length of the smallest edge in Ω_j, (4.6.1)-(4.6.2) would seem to be a significant improvement over the usual CFL condition. We see geometrically that r_j is a least half the size of h_j. In particular we note that on our uniform mesh if the flow is parallel to an axis, say $b = 0$, we find that $\frac{h}{||\mathbf{a}||} = \frac{\Delta x}{a}$ and the time step restriction (4.6.1) will have no dependence on Δy. This implies that we are able to use a very fine mesh in the y direction without sacrificing the size of the time step Δt, a fact we will demonstrate in the numerical examples section below.

4.6.1 CFL Condition for Non-linear Systems

While the proposed CFL condition (4.6.1) is valid for scalar linear problems in 2D, it is unclear how such a condition should be extended to more general non-linear problems. Here we briefly extend the above analysis to these more general problems, using a usual linearization argument, and propose a more general CFL condition.

We consider the general two-dimensional hyperbolic system (1.2.14) on cell Ω_j. As is usually done when considering the CFL condition for non-linear problems, we linearize the flux tensor $\mathbf{F}(\mathbf{U}_j)$ around the cell averages $\bar{\mathbf{U}}_j$ in Ω_j. We assume that the higher-order terms in this linearization are sufficiently small to be negligible so that we can consider the stability of the linear system

$$\frac{\partial \mathbf{U}_j}{\partial t} + A\frac{\partial \mathbf{U}_j}{\partial x} + B\frac{\partial \mathbf{U}_j}{\partial y} = 0,$$

on Ω_j where $A = \frac{d\mathbf{F}_1}{d\mathbf{u}}(\bar{\mathbf{U}}_j)$ and $B = \frac{d\mathbf{F}_2}{d\mathbf{u}}(\bar{\mathbf{U}}_j)$. Now in the particular case when the matrices A and B commute we will have that they are also simultaneously diagonalizable. Hence, through a change of variables $\mathbf{W}_j = R\mathbf{U}_j$, where the columns of R are the simultaneous eigenvectors of A and B, we can decouple the system into M equations where M is size of the system (1.2.14), i.e.

$$\frac{\partial \mathbf{W}_j}{\partial t} + D_1\frac{\partial \mathbf{W}_j}{\partial x} + D_2\frac{\partial \mathbf{W}_j}{\partial y} = 0,$$

where D_1 and D_2 are diagonal matrices. Denoting the vector $\mathbf{W}_j$ as $\mathbf{W}_j = [W_{1,j}, W_{2,j}, \ldots, W_{M,j}]^T$, and the diagonal elements of D_1 and D_2 as a_k and b_k, respectively, for $k = 1, \ldots, M$, we can write this system as the M characteristic equations

$$\frac{\partial W_{k,j}}{\partial t} + a_k\frac{\partial W_{k,j}}{\partial x} + b_k\frac{\partial W_{k,j}}{\partial y} = 0,$$

for $k = 1, \ldots, M$. We can then consider the CFL condition associated with each of these independent equations. That is, we calculate the parameter $h_{k,j}$ associated with the k-th characteristic field's velocity $\mathbf{a}_k = [a_k, b_k]^T$, and take the time step Δt to be bounded by the smallest of all the $h_{k,j}$,

$$\Delta t \leq CFL \min_{j,k} \frac{h_{k,j}}{||\mathbf{a}_k||}. \tag{4.6.3}$$

In the more general case when the matrices A and B do not commute, some characteristic fields will not have a unique flow direction. Rather, these fields will have an infinite set of flow directions forming a characteristic Monge cone. For these fields, we can extend our proposed CFL condition by considering all directions and minimizing the cell width divided by the flow velocity of each field over all directions. An alternative option is simply to take $h_{k,j}$ to be the minimum height of the cell Ω_j, $h_{\min,j}$, and $||\mathbf{a}_k||$ to be the largest flow velocity of the field over all possible directions. Using either of these approaches in the calculation of h_j in (4.6.3) should be sufficient to obtain a linearly stable time step.

4.7 Global Superconvergence

We conclude the analysis of the DG scheme on triangular elements by extending the global superconvergence results of Chapter 2 to these problems. We begin with an analogous result to Theorem 2.3.

Theorem 4.6 (Physical Spectrum)**.** *Let U be the numerical solution of the DG scheme* (4.2.4) *on a unit square domain Σ with periodic boundary conditions and with a uniform mesh of NM rectangles Σ_{lj} which have been subsequently subdivided into two triangular elements, and let U_{lj} be the restriction of the numerical solution to a rectangle Σ_{lj}.*

The numerical solution U can be decomposed into $(p+1)(p+2)NM$ solutions. Each of these solutions is polynomial in ξ and η on each triangular element and has the form $U_{lj}(\xi, \eta, t) = \hat{U}_{lj}(\xi, \eta)e^{-||\mathbf{a}||\omega t}$. These solutions also satisfy $\hat{U}_{lj}(1, \eta) = e^{\kappa_n \Delta x}\ \hat{U}_{l-1,j}(1, \eta)$ and $\hat{U}_{lj}(\xi, 1) = e^{\kappa_m \Delta y}\ \hat{U}_{l,j-1}(\xi, 1)$ for each l and j where $\kappa_n = 2\pi n i$, $n = 0, \dots, N-1$ and $m = 0, \dots, M-1$. If $a \neq 0$ and $b \neq 0$ then corresponding to each n and m there are $(p+1)(p+2)$ spectral values $\omega = \omega_k$, $0 \leq k < (p+1)(p+2)$ which have the expansions

$$\omega_0 = \frac{a\kappa_n + b\kappa_m}{||\mathbf{a}||} + \mathcal{O}(h^{2p+1}),$$

and

$$\omega_k = \frac{\mu_k}{h} + \mathcal{O}(\kappa_n) + \mathcal{O}(\kappa_m),$$

for $1 \leq k < (p+1)(p+2)$, where every μ_{nm} satisfies $\mathrm{Re}(\mu_k) > 0$.

On the other hand, if either $a = 0$ or $b = 0$ then the $(p+1)(p+2)$ spectral values have the expansions

$$\omega_k = \frac{a\kappa_n + b\kappa_m}{||\mathbf{a}||} + \mathcal{O}(h^{2(p-k)+1}),$$

for $k = 0, \ldots, p$ and

$$\omega_k = \frac{\mu_k}{h} + \mathcal{O}(\kappa_n) + \mathcal{O}(\kappa_m),$$

for $p+1 \leq k < (p+1)(p+2)$, where every μ_k satisfies $\mathrm{Re}(\mu_k) > 0$.

As with the results on concerning the superaccuracies and spectrum of the DG method on triangular elements, this result relies on the explicit computation of the polynomial solutions $\hat{U}_{lj}(\xi, \eta)$ and thus has only been symbolically verified for $p \leq 5$, but is conjectured to be true for all p. We verify this result in the following way: considering again the determinant C_p of the system (4.5.4), whose roots ω are the spectral values of the method, we asymptotically expand each root as

$$\omega = \frac{d_{-1}}{h} + d_0 + d_1 h + \ldots,$$

and solve for the coefficients $d_{-1}, d_0, d_1, \ldots$ by setting like powers of h to zero.

As with Theorem 2.3 for one-dimensional problems, this theorem reveals that the spectrum of the DG method in two dimension on a uniform mesh with periodic boundary conditions can be partitioned into physical and non-physical modes. The modes which satisfy $\omega_k = \frac{a\kappa_n + b\kappa_m}{||\mathbf{a}||} + \mathcal{O}(h^l)$ are viewed as physical since they propagate with numerical frequencies which agree with the exact frequencies to a high-order of accuracy. The modes which satisfy $\omega_k = \frac{\mu_k}{h} + \mathcal{O}(\kappa_n) + \mathcal{O}(\kappa_m)$ are subsequently viewed as non-physical and are damped out exponentially quickly since $\mathrm{Re}(\mu_k) > 0$.

To determine the global superconvergence properties of the DG method for this problem we must know the superconvergence properties of the physical modes. We therefore return to the system (4.5.4) and symbolically calculate the eigenvectors associated to the physical frequencies. Beginning with the case where $a \neq 0$ and $b \neq 0$ we find that the physical mode associated to the frequency $\omega_0 = \frac{a\kappa_n + b\kappa_m}{||\mathbf{a}||} + \mathcal{O}(h^{2p+1})$ has the expansion

$$\hat{U}_j = e^{\kappa_n h \xi + \kappa_m h \eta} + \mathcal{O}(h^{p+1}).$$

For $p = 1, 2$, and 3 we have symbolically verified the following orthogonality properties of this physical mode. On each triangular cell Ω_j the physical mode satisfies

$$\int_{\partial\Omega_0^+} (\mathbf{a} \cdot \mathbf{n}) \left[\hat{U}_j - e^{\kappa_n h \xi + \kappa_m h \eta}\right] \, ds = \mathcal{O}(h^{p+2}), \tag{4.7.1}$$

and

$$\iint_{\Omega_0} \left[\hat{U}_j - e^{\kappa_n h\xi + \kappa_m h\eta}\right] \, dA = \mathcal{O}(h^{p+2}), \tag{4.7.2}$$

Also, on each square Σ_{jl} the physical mode satisfies

$$\iint_{\Sigma_0} \left[\hat{U}_{jl} - e^{\kappa_n h\xi + \kappa_m h\eta}\right] \, dA = \mathcal{O}(h^{2p+1}). \tag{4.7.3}$$

Using these properties of the physical modes, we propose the following global superconvergence result.

Theorem 4.7 (Global Superconvergence)**.** *Let $u(x,y,t)$ be a smooth exact solution of* (4.1.1) *with neither a nor b equal to zero on the unit square Σ with periodic boundary conditions. Let U be the numerical solution of the DG scheme* (4.2.4) *on a uniform mesh of NM rectangles Σ_{lj} which have been subsequently subdivided into two equal triangular elements.*

Let U_{jl} be the restriction of the numerical solution to the square Σ_{jl} and let U_j be the restriction of the numerical solution to a triangular element Ω_j. Let $\epsilon_{jl} = U_{jl} - u_{jl}$ and $\epsilon_j = U_j - u_j$ be the numerical error on the square Σ_{jl} and triangle Ω_j, respectively, mapped to the canonical cells via the mapping (1.2.17)*. Suppose the projection of the initial profile $u(x,y,0)$ into the finite element space is chosen so that on each triangular element Ω_j*

$$\iint_{\Omega_0} [U_j - u_j] \, \psi_{ki} \, dA = \mathcal{O}(h^{2p-k+1}), \tag{4.7.4}$$

for $k = 0, \ldots, p-1$ and $i = 0, \ldots k$. Then after sufficiently long time that the non-physical modes of the numerical solution have been damped out, the numerical error ϵ_j on each triangular cell satisfies

$$\int_{\partial\Omega_0^+} (\mathbf{a} \cdot \mathbf{n}) \epsilon_j \, ds = \mathcal{O}(h^{p+2}), \tag{4.7.5}$$

and

$$\iint_{\Omega_0} \epsilon_j \, dA = \mathcal{O}(h^{p+2}). \tag{4.7.6}$$

Furthermore, over each square Σ_{jl} the numerical error ϵ_{jl} satisfies

$$\iint_{\Sigma_0} \epsilon_{jl} \, dA = \mathcal{O}(h^{2p+1}). \tag{4.7.7}$$

Sketch of Proof. The proof of this theorem relies on the results of Theorem 4.6 and our symbolic verification of the orthogonality properties of the physical modes and is therefore only conjectured to be true for $p > 3$. Assuming these results hold, however, we give a brief sketch of the proof of Theorem 4.7. We again consider an exact solution which consists of a single discrete Fourier mode $u(x, y, t) = \hat{u}_{nm} e^{\kappa_n x + \kappa_m y - (a\kappa_n + b\kappa_m)t}$. Then on the square Σ_{lj} the numerical solution can be written as the sum of the $(p+1)(p+2)$ independent solutions described in Theorem 4.6 associated with κ_n and κ_m. Furthermore, as described in the theorem, one of these solutions can be viewed as physical while the remaining non-physical modes are damped a out exponentially quickly. The numerical solution tends exponentially quickly to a scalar multiple of the physical mode.

Using the symbolically verified properties of the physical modes in (4.7.2) we can proceed as in the proof of the one-dimensional global superconvergence results in Theorems 2.4 and 3.4 and argue that the initial projection (4.7.4) projects the initial data onto the physical mode to order $2p + 1$. Therefore, after sufficient time that the non-physical modes of the numerical solution have been damped out, we use the orthogonality properties (4.7.1)-(4.7.3) and the high-order accuracy of the physical frequency ω_0 in order to establish that (4.7.5)-(4.7.7) will hold for $t > 0$. □

In the special case when either $a = 0$ or $b = 0$, every cell in the uniform mesh of the square Σ will be type III. Without loss of generality we will assume that $b = 0$ so that $\theta = 1$ in every cell. In this case, from the results in Theorem 4.6 from our symbolic computations of the spectrum of the DG spatial operator, we will have for every κ_n and κ_m, $p + 1$ spectral values ω_k, $k = 0, \dots, p$, which can be viewed as physical and have the expansions $\omega_k = \frac{a\kappa_n}{||\mathbf{a}||} + \mathcal{O}(h^{2(p-k)+1})$. Upon symbolically computing the physical modes associated to these physical spectral values we find that the have the form

$$\hat{U}_j = h^k P_k(2\eta - 1) e^{\kappa_n h \xi} + \mathcal{O}(h^{p+1}),$$

and we have symbolically verified for $p = 1, 2$, and 3 the following properties of these modes

$$\int_{\partial\Omega_0^+} \left[\hat{U}_j - h^k P_k(2\eta - 1) e^{\kappa_n h \xi} \right] \eta^k \, ds = \mathcal{O}(h^{2p-k+1}), \tag{4.7.8}$$

for $k = 0, \dots, p$ and

$$\iint_{\Omega_0} \left[\hat{U}_j - h^k P_k(2\eta - 1) e^{\kappa_n h \xi} \right] \psi_{ki} \, dA = \mathcal{O}(h^{2p-k}), \tag{4.7.9}$$

for $k = 0, \dots, p - 1$ and $i = 0, \dots, k$. Using these properties of the physical modes we can establish a stronger superconvergence result in this special case.

Theorem 4.8. *Suppose the conditions of Theorem 4.7 hold but assume instead that either $a = 0$ or $b = 0$. Then after sufficiently long time that the non-physical modes of the numerical solution have been damped out, the numerical error ϵ_j on each cell satisfies*

$$\iint_{\Omega_0} \epsilon_j \psi_{ki} \, dA = \mathcal{O}(h^{2p-k}), \tag{4.7.10}$$

for $k = 0, \ldots, p-1$ and $i = 0, \ldots k$, and

$$\int_{\partial\Omega_0^+} (\mathbf{a} \cdot \mathbf{n}) \epsilon_j (a\eta - b\xi)^k \, ds = \mathcal{O}(h^{2p-k+1}), \tag{4.7.11}$$

for $k = 0, \ldots, p$.

The proof of this theorem follows a similar argument to the proof of Theorem 4.7 and uses the orthogonality properties (4.7.8) and (4.7.9).

We finish this section with a useful corollary, analogous to Corollary 2.2, regarding the accumulation error of the numerical solution which holds when $a \neq 0$ and $b \neq 0$ since by Theorem 4.6 every frequency of the physical modes is accurate to the exact frequency to order $2p+1$.

Corollary 4.1. *The accumulation error of the superconvergent numerical solution described in Theorem 4.7 is of order $2p+1$. That is, let U_{lj} be the numerical solution on the rectangle Σ_{lj} and suppose there exists a time τ such that $r = a\tau$ and $s = b\tau$ are integers. Then after sufficiently long time the non-physical modes of the numerical solution have been damped out and the numerical solution satisfies*

$$||U_{l+r,j+s}(\xi, \eta, t+\tau) - U_{lj}(\xi, \eta, t)|| = \mathcal{O}(h^{2p+1}).$$

In the next section we perform several numerical tests to confirm the global superconvergence results of this section as well as test the efficacy of the new CFL condition proposed in section 4.6.

4.8 Numerical Examples

In this section we will perform several numerical experiments to confirm the superconvergence properties stated in the section above for the DG method for the two-dimensional linear advection equation. Specifically, we will confirm the superconvergence properties described in Theorems 4.7 and 4.8 for the linear advection equation with $a \neq 0$ and $b \neq 0$ on a uniform mesh with periodic boundary conditions. We also show that when t is sufficiently large the superconvergent numerical solution is advected at order $\mathcal{O}(h^{2p+1})$. We will then establish the stronger superconvergence results detailed in Theorem 4.8 in the special case when $b = 0$. We will then give several examples which show the improvement in the number of time steps required in time integration when using the new CFL condition proposed in Section 4.6 when compared to the regularly implemented CFL condition.

4.8.1 Superconvergence Tests

Our first numerical study was done on the initial value problem

$$\begin{aligned} u_t + 2u_x + u_y &= 0, \qquad 0 < x < 1, \quad 0 < y < 1, \qquad t > 0, \\ u(x,y,0) &= u_0(x,y), \\ u(0,y,t) &= u(1,y,t), \\ u(x,0,t) &= u(x,1,t), \end{aligned} \tag{4.8.1}$$

with

$$u_0(x,y) = \sin 2\pi x. \tag{4.8.2}$$

All test below are performed using the classical RK-4 time-stepping scheme and a CFL number of $\frac{0.1}{(2p+1)\left(1+\frac{4}{(p+2)^2}\right)}$ to minimize the error incurred in time integration. We also use an initial projection which satisfies the conditions of Theorems 4.7 and 4.8. In each test, we calculate the numerical solution on the uniform mesh of triangles described at the beginning of Section 4.5. We also take the number of cells in the x-direction to be equal to the number of cells in the y-direction, i.e. $N = M$, for a total of $N_\Omega = 2N^2$ triangular cells.

In Tables 4.1-4.3 we show the results of this convergence test for $p = 1$, 2, and 3, respectively. We present the L^1 norm of the integral of the numerical error along the outflow edges of each cell, i.e.

$$||E|| = \sum_{j=1}^{N_\Omega} \det J_j \left| \int_{\partial\Omega_0^+} (\mathbf{a}\cdot\mathbf{n})[U_j - u_j]\, ds \right|,$$

N_Ω	h	$\vert\vert E\vert\vert$	r	$\vert\vert E_\Omega\vert\vert$	r	$\vert\vert E_\Sigma\vert\vert$	r
32	1.25e-01	2.98e-01	-	2.43e-01	-	1.72e-01	-
64	8.33e-02	1.31e-01	2.03	9.74e-02	2.26	6.89e-02	2.26
128	6.25e-02	6.27e-02	2.56	4.54e-02	2.66	3.21e-02	2.66
256	5.00e-02	3.39e-02	2.76	2.42e-02	2.81	1.71e-02	2.81
512	4.17e-02	2.01e-02	2.85	1.43e-02	2.88	1.01e-02	2.88

Table 4.1: Linear advection, (4.8.1)-(4.8.2) for $p = 1$. L^1 norms of the integral of the error along the outflow edges $||E||$, over each triangluar cell $||E_\Omega||$, and over each square cell $||E_\Sigma||$ are shown together with convergence rates, r, with respect to the parameter h. Errors are calculated at $t = 2$.

N_Ω	h	$\vert\vert E\vert\vert$	r	$\vert\vert E_\Omega\vert\vert$	r	$\vert\vert E_\Sigma\vert\vert$	r
32	1.25e-01	1.05e-02	-	7.83e-03	-	5.54e-03	-
64	8.33e-02	1.67e-03	4.53	1.11e-03	4.81	7.88e-04	4.81
128	6.25e-02	4.33e-04	4.69	2.72e-04	4.91	1.92e-04	4.91
256	5.00e-02	1.50e-04	4.74	9.02e-05	4.94	6.38e-05	4.94
512	4.17e-02	6.33e-05	4.75	3.65e-05	4.96	2.58e-05	4.96

Table 4.2: Linear advection, (4.8.1)-(4.8.2) for $p = 2$. L^1 norms of the integral of the error along the outflow edges $||E||$, over each triangluar cell $||E_\Omega||$, and over each square cell $||E_\Sigma||$ are shown together with convergence rates, r, with respect to the parameter h. Errors are calculated at $t = 2$.

along with the error in the cell averages over each triangular cell Ω_j

$$||E_\Omega|| = \sum_{j=1}^{N_\Omega} \det J_j \left| \iint_{\Omega_0} [U_j - u_j] \, dA \right|,$$

and over each square Σ_{jl}

$$||E_\Sigma|| = \sum_{j=1}^{N} \sum_{l=1}^{N} 2 \det J_j \left| \iint_{\Sigma_0} [U_{jl} - u_{jl}] \, dA \right|.$$

We calculate this error on uniform meshes of $N_\Omega = 32, 64, 128, 256,$ and 512 triangular cells, and we report the parameter h, calculated by its definition in (4.3.3). Errors are calculated

N_Ω	h	$\|\|E\|\|$	r	$\|\|E_\Omega\|\|$	r	$\|\|E_\Sigma\|\|$	r
32	1.25e-01	1.20e-04	-	9.94e-05	-	7.03e-05	-
64	8.33e-02	8.33e-06	6.57	7.03e-06	6.53	4.37e-06	6.85
128	6.25e-02	1.28e-06	6.50	1.25e-06	5.99	5.69e-07	7.09
256	5.00e-02	3.57e-07	5.74	3.49e-07	5.73	1.19e-07	7.02
512	4.17e-02	1.23e-07	5.84	1.30e-07	5.42	3.33e-08	6.97

Table 4.3: Linear advection, (4.8.1)-(4.8.2) for $p = 3$. L^1 norms of the integral of the error along the outflow edges $||E||$, over each triangular cell $||E_\Omega||$, and over each square cell $||E_\Sigma||$ are shown together with convergence rates, r, with respect to the parameter h. Errors are calculated at $t = 2$.

N_Ω	h	$\|\|U(x,y,0) - U(x,y,1)\|\|$	r	$\|\|U(x,y,1) - U(x,y,2)\|\|$	r
32	1.25e-01	6.00e-02	-	5.22e-02	-
64	8.33e-02	2.10e-02	2.59	2.03e-02	2.33
128	6.25e-02	9.78e-03	2.65	9.29e-03	2.72
256	5.00e-02	5.50e-03	2.57	4.92e-03	2.85
512	4.17e-02	3.53e-03	2.43	2.90e-03	2.91

Table 4.4: Linear advection, (4.8.1)-(4.8.2) for $p = 1$. L^1 norms of difference in numerical solutions at different times. Differences are measured between U initially and at $t = 1$, then between U at $t = 1$ and $t = 2$.

at $t = 2$ in order to allow sufficient time for the non-physical modes to be damped out. We then calculated the rates of convergence r with respect to the h parameter for each of the methods. In each test we observe the expected order $p + 2$ convergence rate of the error $||E||$ along the outflow edges of each cell and the error in the cell averages in each cell $||E_\Omega||$. We also observe the predicted order $2p + 1$ convergence rate of the error over each square $||E_\Sigma||$.

Next, we verify the results of Corollary 4.1 by verifying that once the non-physical modes of the numerical solution have been damped out, the remaining modes are advected at order $2p+1$. To do this we calculate the L^1 norm of the difference between the numerical solutions at $t = 0$ and $t = 1$, and the numerical solutions at $t = 1$ and $t = 2$. We calculate

N_Ω	h	$\|\|U(x,y,0)-U(x,y,1)\|\|$	r	$\|\|U(x,y,1)-U(x,y,2)\|\|$	r
32	1.25e-01	3.33e-03	-	1.73e-03	-
64	8.33e-02	9.07e-04	3.21	2.33e-04	4.95
128	6.25e-02	3.64e-04	3.18	5.63e-05	4.93
256	5.00e-02	1.85e-04	3.04	1.86e-05	4.96
512	4.17e-02	1.05e-04	3.07	7.51e-06	4.98

Table 4.5: Linear advection, (4.8.1)-(4.8.2) for $p = 2$. L^1 norms of difference in numerical solutions at different times. Differences are measured between U initially and at $t = 1$, then between U at $t = 1$ and $t = 2$.

these differences as

$$||U(x,y,0) - U(x,y,1)|| = \sum_{j=1}^{N} \det J_j \iint_{\Omega_0} |U_j(\xi,\eta,0) - U_j(\xi,\eta,1)| \, dA.$$

In Tables 4.4 and 4.5 we see that the difference between the initial numerical solution at $t = 0$ and the numerical solution at $t = 1$ converges as h^{p+1}, as usual, while the difference between the numerical solutions at $t = 1$ and $t = 2$ converges at the predicted h^{2p+1} rate. This shows that once the non-physical modes of the numerical solution have been damped out the remaining physical modes are advected at order $2p + 1$.

In our next superconvergence study, we aim to verify the results of Theorem 4.8 which concerns the special case where every cell in the mesh is type III. To do this we consider the initial value problem

$$\begin{aligned} u_t + u_x &= 0, \qquad 0 < x < 1, \quad 0 < y < 1, \qquad t > 0, \qquad (4.8.3) \\ u(x,y,0) &= u_0(x,y), \\ u(0,y,t) &= u(1,y,t), \\ u(x,0,t) &= u(x,1,t), \end{aligned}$$

with

$$u_0(x,y) = \sin 2\pi(x+y). \qquad (4.8.4)$$

N_Ω	h	$\lvert\lvert E_0\rvert\rvert$	r
32	1.25e-01	1.22e-01	-
64	8.33e-02	4.29e-02	2.59
128	6.25e-02	1.91e-02	2.82
256	5.00e-02	9.98e-03	2.90
512	4.17e-02	5.84e-03	2.94

Table 4.6: Linear advection, (4.8.3)-(4.8.4) for $p = 1$. L^1 norms of the integral of the error along the outflow edges $||E_0||$ are shown together with convergence rates, r, with respect to the parameter h. Errors are calculated at $t = 2$.

N_Ω	h	$\lvert\lvert E_0\rvert\rvert$	r	$\lvert\lvert E_1\rvert\rvert$	r	$\lvert\lvert M_{00}\rvert\rvert$	r
32	1.25e-01	2.99e-03	-	2.63e-03	-	3.29e-03	-
64	8.33e-02	4.32e-04	4.76	5.08e-04	4.06	6.67e-04	3.93
128	6.25e-02	1.06e-04	4.89	1.53e-04	4.16	2.06e-04	4.08
256	5.00e-02	3.52e-05	4.93	6.13e-05	4.11	8.32e-05	4.06
512	4.17e-02	1.42e-05	4.95	2.91e-05	4.08	4.02e-05	3.99

Table 4.7: Linear advection, (4.8.3)-(4.8.4) for $p = 2$. L^1 norms of the moments of the error along the outflow edges $||E_k||$, $k = 0, 1$, and the L^1 norms of the moments of the error over the whole cell $||M_{ki}||$, $k = i = 0$, are shown together with convergence rates, r, with respect to the parameter h. Errors are calculated at $t = 2$.

In Tables 4.6-4.8 we show the results of this convergence test for $p = 1, 2$, and 3, respectively. We present the L^1 norms of two different kinds of errors. The first is the moment of the error along the outflow edges, i.e.

$$||E_k|| = \sum_{j=1}^{N_\Omega} \det J_j \left| \int_{\partial\Omega_0^+} [U_j - u_j]\eta^k \, ds \right| .$$

The second error is the moment of the error over the whole cell,

$$||M_{ki}|| = \sum_{j=1}^{N_\Omega} \det J_j \left| \iint_{\Omega_0} [U_j - u_j]\psi_{ki} \, ds \right| .$$

Errors are calculated at $t = 2$ in order to allow sufficient time for the non-physical modes to be damped out. We then calculated the rates of convergence r with respect to the h

N_Ω	h	$\|\|E_0\|\|$	r	$\|\|E_1\|\|$	r	$\|\|E_2\|\|$	r
32	1.25e-01	3.46e-05	-	3.11e-05	-	4.45e-05	-
64	8.33e-02	2.48e-06	6.50	3.19e-06	5.61	5.09e-06	5.34
128	6.25e-02	3.75e-07	6.56	5.42e-07	6.16	1.14e-06	5.21
256	5.00e-02	8.08e-07	6.88	1.49e-07	5.79	3.65e-07	5.08
512	4.17e-02	2.12e-08	7.32	4.90e-08	6.10	1.46e-07	5.03

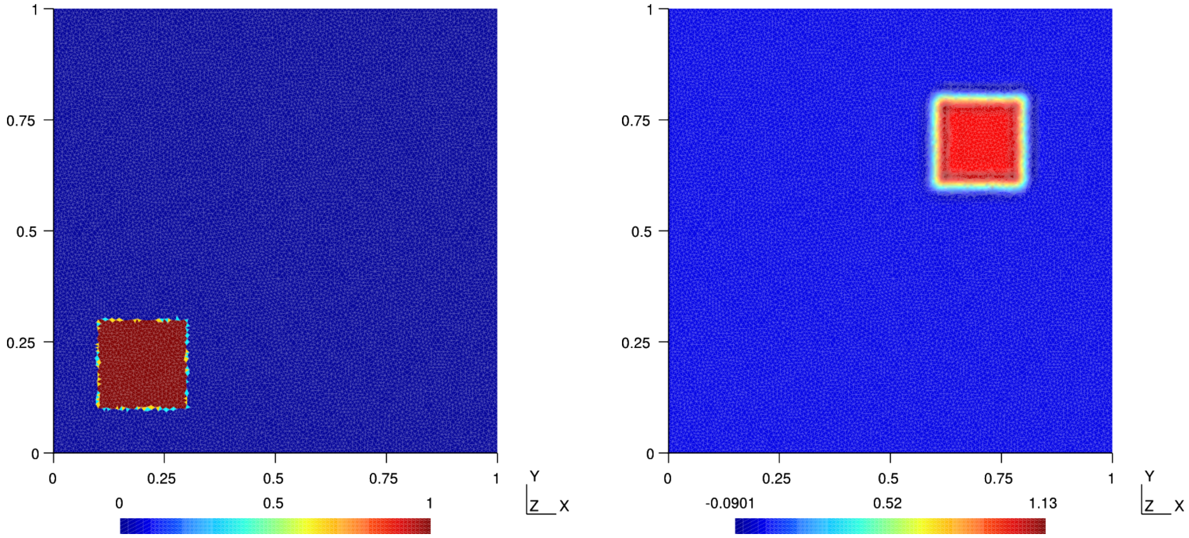
N_Ω	h	$\|\|M_{00}\|\|$	r	$\|\|M_{10}\|\|$	r	$\|\|M_{11}\|\|$	r
32	1.25e-01	4.89e-05	-	1.33e-04	-	9.73e-05	-
64	8.33e-02	4.07e-06	6.13	2.02e-05	4.66	1.24e-05	5.08
128	6.25e-02	7.29e-07	5.97	4.95e-06	4.88	2.91e-06	5.04
256	5.00e-02	1.84e-07	6.18	1.64e-06	4.95	9.53e-07	5.01
512	4.17e-02	6.19e-08	5.97	6.64e-07	4.97	3.82e-07	5.01

Table 4.8: Linear advection, (4.8.3)-(4.8.4) for $p = 3$. L^1 norms of the moments of the error along the outflow edges $||E_k||$, $k = 0, 1, 2$, and the L^1 norms of the moments of the error over the whole cell $||M_{ki}||$, $k = 0, 1$, $i = 0, \ldots k$, are shown together with convergence rates, r, with respect to the parameter h. Errors are calculated at $t = 2$.

parameter for each of the methods. In each test we observe the expected order $2p - k + 1$ convergence of the error $||E_k||$ along the outflow edges of the cells, and we observe the expected order $2p - k$ convergence of the error $||M_{ki}||$ over all of the cells.

4.8.2 Proposed CFL Condition

Our second set of numerical tests aims to compare the CFL condition proposed in Section 4.6, which is scaled by the parameter h_j, to the usual CFL condition implemented which is scaled by the radius of the inscribed circle in the cell. We will present both linear and non-linear examples at several orders p to demonstrate its efficacy. In each example, we pair the degree p DG discretization with an explicit RK-$(p + 1)$ time integration scheme and do not apply a limiter. We apply the method to each problem twice, once with the usual CFL condition and again with our proposed condition noting that both choices are indeed stable, and compare the number of time steps required to reach the final time.



(a) $t = 0$

(b) $t = 0.5$

Figure 4.7: Linear advection equation (4.8.5) with square pulse initial profile (4.8.6) on an unstructured mesh of 26524 elements. Shown initially at $t = 0$ (left) and at $t = 0.5$ (right).

For the first linear test we consider the initial value problem

$$\begin{aligned} u_t + u_x + u_y &= 0, \qquad 0 < x < 1, \quad 0 < y < 1, \qquad t > 0, \\ u(x, y, 0) &= u_0(x, y), \\ u(0, y, t) &= u(1, y, t), \\ u(x, 0, t) &= u(x, 1, t), \end{aligned} \tag{4.8.5}$$

with the initial condition consisting of a square pulse

$$u_0(x, y) = \begin{cases} 1, & 0.1 \le x \le 0.3, \quad 0.1 \le y \le 0.3 \\ 0, & \text{otherwise.} \end{cases} \tag{4.8.6}$$

We apply the DG method on an unstructured mesh of 26524 elements up to a final time of $t = 0.5$ for several orders p. Using the usual time step restriction

$$\Delta t \le \frac{1}{2p+1} \min_j \frac{r_j}{||\mathbf{a}||},$$

where r_j is the radius of the inscribed circle in Ω_j we find that the $p = 1, 2,$ and 3 schemes required 1187, 1978, and 2768 time steps, respectively, to reach the final time. In comparison, using our proposed CFL condition

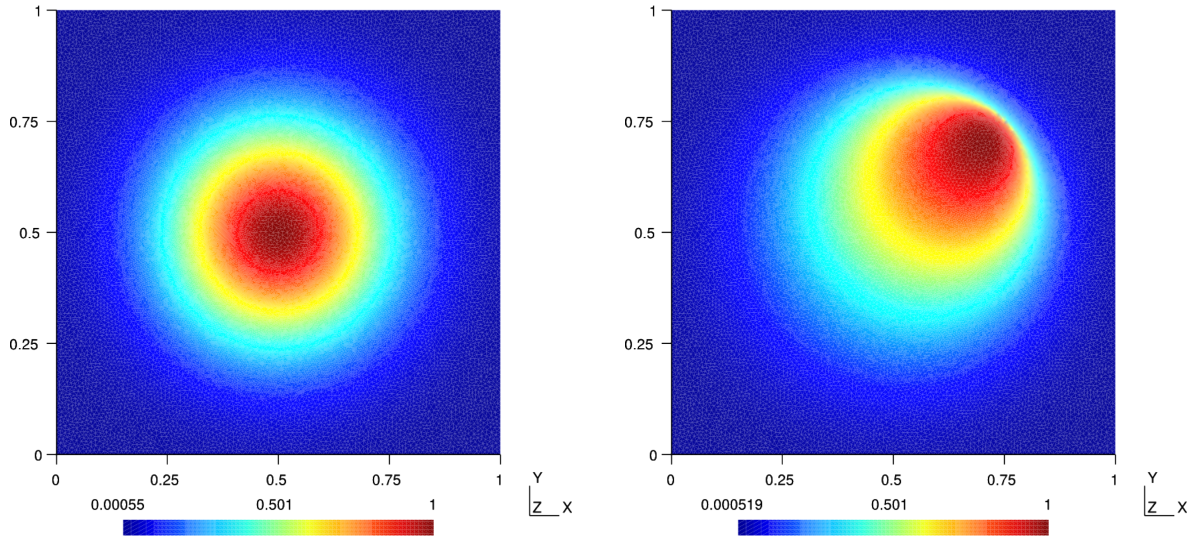
$$\Delta t \leq \frac{1}{(2p+1)\left(1+\frac{4}{(p+2)^2}\right)} \min_j \frac{h_j}{||\mathbf{a}||},$$

we find that the $p = 1, 2$, and 3 schemes required 715, 1032, and 1340 time steps, respectively, to reach the final time. Since the computational cost of the scheme is directly proportional to the number of time steps required, this marks a significant increase in the efficiency of the algorithm. Indeed, at $p = 3$ the proposed time step restriction reduces the amount of time steps required by more than 50%.

It was remarked in Section 4.6 that since h_j is geometrically the length of the cell Ω_j along the direction of flow $\mathbf{a}$ then h_j has no dependence on the size of Ω_j in an orthogonal direction. This implies that the mesh can be refined indefinitely in this dimension without further restricting the time step of the method. To observe this we consider a particularly simple example. We again consider the linear advection equation (4.8.3) with the square pulse initial profile (4.8.6), which has a flow direction parallel to the x-axis. We apply the DG method on the uniform mesh described at the beginning of Section 4.5 with $\Delta x = \frac{1}{50}$ and $\Delta y = \frac{1}{250}$ for a total of 25000 cells. The mesh is therefore five times more refined in the y direction. Classically we would expect this level of refinement to be the limiting factor in the time step restriction. Hence, when using the usual time step restriction involving the radius of the inscribed circle we find that the $p = 1, 2$, and 3 schemes required 833, 1388, and 1943 time steps, respectively, to reach the final time of $t = 0.5$. On the other hand, upon computing h_j thorough (4.3.3) we find that it has no dependence on Δy. Therefore the scheme using our proposed time step restriction requires only 109, 157, and 204 time steps, respectively to reach the same final time of $t = 0.5$. We see that our proposed CFL condition achieves more than an 85% reduction in computational cost in this special case. We note that the percentage of computational cost reduced would increase asymptotically to 100% in this example with continued refinements to Δy.

We proceed with two non-linear examples. For the first, we consider Burgers' equation in two dimensions,

$$\begin{aligned} u_t + uu_x + uu_y &= 0, \qquad 0 < x < 1, \quad 0 < y < 1, \qquad t > 0, \qquad (4.8.7) \\ u(x,y,0) &= u_0(x,y), \\ u(0,y,t) = u(1,y,t) &= 0, \\ u(x,0,t) = u(x,1,t) &= 0, \end{aligned}$$



(a) $t = 0$

(b) $t = 0.5$

Figure 4.8: Burgers' equation (4.8.7) with Gaussian pulse initial profile (4.8.8) on an unstructured mesh of 41416 elements. Shown initially at $t = 0$ (left) and at $t = 0.1$ (right).

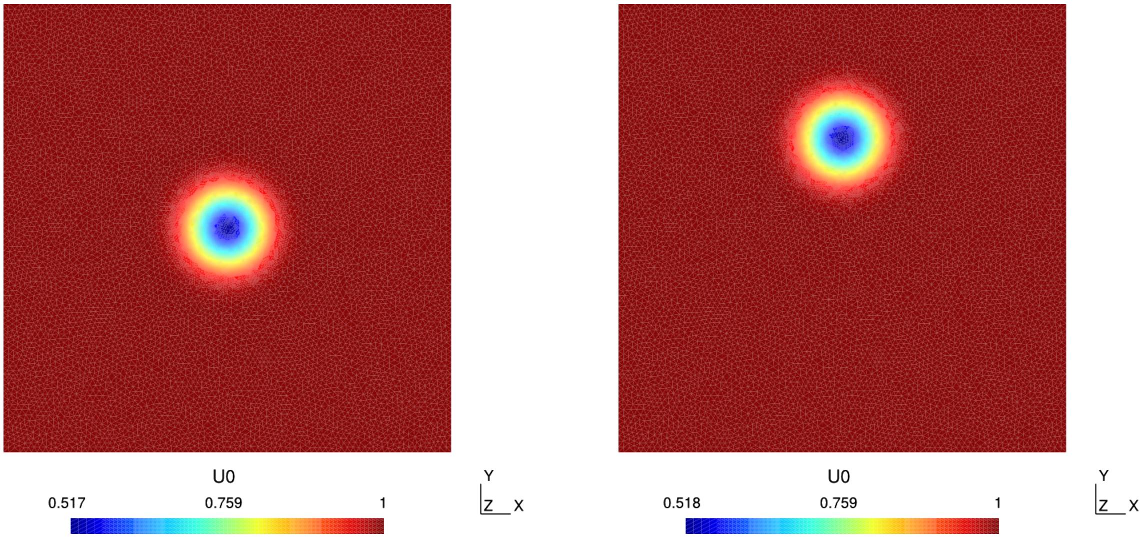
with the initial condition consisting of a centred Gaussian pulse

$$u_0(x, y) = \exp\left(-15((x - 0.5)^2 + (y - 0.5)^2)\right). \tag{4.8.8}$$

We apply the DG method on an unstructured mesh of 41416 elements up to a final time of $t = 0.1$, before the formation of a shock. Using the usual time step restriction involving the radius of the inscribed circle in Ω_j we find that the $p = 1, 2$, and 3 schemes required 265, 442, and 619 time steps, respectively, to reach the final time. In comparison, using our proposed CFL condition we find that the schemes required 148, 213, and 276 time steps, respectively.

Finally, in our second non-linear example we consider the Euler equations in two dimensions,

$$\frac{\partial}{\partial t}\begin{pmatrix}\rho \\ \rho u \\ \rho v \\ E\end{pmatrix} + \frac{\partial}{\partial x}\begin{pmatrix}\rho u \\ \rho u^2 + P \\ \rho uv \\ u(E + P)\end{pmatrix} + \frac{\partial}{\partial y}\begin{pmatrix}\rho v \\ \rho uv \\ \rho v^2 + P \\ v(E + P)\end{pmatrix} = 0, \tag{4.8.9}$$



(a) $t = 0$

(b) $t = 0.1$

Figure 4.9: Euler equations (4.8.9) with smooth vortex initial profile (4.8.11)-(4.8.14) on an unstructured mesh of 16954 elements. Shown initially at $t = 0$ (left) and at $t = 4$ (right).

on the square $-10 \leq x \leq 10$ and $-10 \leq y \leq 10$ with the equation of state

$$P = (\gamma - 1)\left(E - \frac{1}{2}\rho(u^2 + v^2)\right), \tag{4.8.10}$$

and the initial condition consisting of a smooth vortex [61] centred at the origin and moving upward, i.e.

$$\rho_0 = \left[1 - \frac{S^2M^2}{8\pi^2}\exp\left(\frac{1 - x^2 - y^2}{R^2}\right)\right]^{\frac{1}{\gamma-1}}, \tag{4.8.11}$$

$$u_0 = \frac{Sy}{2\pi R}\exp\left(\frac{1 - x^2 - y^2}{2R^2}\right), \tag{4.8.12}$$

$$v_0 = 1 - \frac{Sx}{2\pi R}\exp\left(\frac{1 - x^2 - y^2}{2R^2}\right), \tag{4.8.13}$$

$$P_0 = \frac{1}{\gamma M^2}\left[1 - \frac{S^2M^2}{8\pi^2}\exp\left(\frac{1 - x^2 - y^2}{R^2}\right)\right]^{\frac{\gamma}{\gamma-1}}, \tag{4.8.14}$$

where $S = 13.5$, $M = 0.4$, $R = 1.5$, and $\gamma = 1.4$. We use the constant boundary condition

$$\begin{pmatrix} \rho \\ u \\ v \\ P \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ \frac{1}{\gamma M^2} \end{pmatrix} \tag{4.8.15}$$

along the boundary of the mesh.

We apply the DG method on an unstructured mesh of 16954 elements up to a final time of $t = 4$. Note that, as per the discussion in Section 4.6.1, since this system is not simultaneously diagonalizable there exist characteristics which do not degenerate to lines. We therefore take h_j to be the minimum height of the cell for these fields. Using the usual time step restriction involving the radius of the inscribed circle in Ω_j we find that the $p = 1, 2$, and 3 schemes required 1387, 2312, and 3236 time steps, respectively, to reach the final time. In comparison, using our proposed CFL condition we find that the schemes required 847, 1222, and 1587 time steps, respectively.

4.9 Discussion

In this chapter we have investigated the superconvergence, superaccuracy, and stability of the DG method on triangular grids. By examining a PDE for the numerical solution we find that the Fourier modes of the numerical solution on each cell are completely determined by a projection of their inflow, a rational function of their frequency, the parameter h_j which can be seen to be the length of the cell Ω_j along the direction of flow $\mathbf{a}$, and a parameter θ which gives a measure of the direction of flow in each cell. These Fourier modes have a local expansion in h_j which involves a projection $\mathcal{R}_{p+1}[[U_j]]$ of the jumps in the numerical solution along the inflow boundary of each cell. By assuming an exact inflow and using a local expansion of these Fourier modes in terms of h_j we are able to give a simple proof of the local superconvergence properties of the method studied in [47, 2, 3].

When we consider a uniform mesh of triangles, we use these Fourier modes to symbolically calculate a polynomial condition $C_p = 0$ which relates the frequencies ω to the numerical wavenumbers $\tilde{\kappa_1}$ and $\tilde{\kappa_2}$. We then perform an expansion of this condition in h in order to verify the order $2p + 1$ superaccuracy of the numerical wavenumbers to the exact wavenumbers in terms of dissipation and dispersion. We also use the condition $C_p = 0$ to calculate the spectrum of the DG method on this uniform mesh. Examining the spectrum over a range of values of θ reveals that the size of the spectrum is not very sensitive to this parameter. Meanwhile, from the form of the polynomial C_p we find that the spectral values are scaled by $\frac{||\mathbf{a}||}{h}$. We therefore propose a new CFL condition for the DG method in which the time step is scaled by the parameter h_j, rather than the radius of the inscribed circle in each cell r_j as is usually implemented. We then extend this CFL condition to general non-linear hyperbolic systems. We see in our numerical tests that this new CFL condition performs significantly better in that it provides much larger stable time steps. Indeed, as p increases our tests with the proposed CFL condition required less than 50% of the usual amount of time steps.

Finally, we show that the spectrum of the DG method on this uniform mesh can be partitioned into frequencies which can be considered either physical of non-physical. The physical modes of the numerical solution propagate with frequencies which agree with the exact frequency to high-order, while the non-physical modes are damped out exponentially quickly in time. We therefore obtain an analogous result to that established in Chapter 2 for the DG scheme in 1D: the accuracy of the numerical solution will be completely determined by the accuracy of the initial projection onto these physical modes. Using this result and symbolically examining the superconvergence properties of the physical modes, we prove that for a class of initial projections the numerical solution will globally tend exponentially quickly towards a superconvergent form.

The analysis of the DG method through the derivation and a PDE which governs the numerical solution, and the Fourier analysis of said PDE, has proven effective in establishing several useful superconvergence, superaccuracy, and stability results of the DG method. The extension of this analysis to other cell geometries in 2D, and to higher-dimensional problems is subject of future study. In particular the extension of the proposed CFL number to these problems. Finally, in this analysis we observe an analogy with the the one-dimensional case, in that we see that much of the error analysis relies on the orthogonality properties of the projection $\mathcal{R}_{p+1}[[U_j]]$ of the jumps in the numerical solution along the inflow boundaries of the cell. As demonstrated in Chapter 3, modifications to this projection can yield significant relaxation of the scheme's stability restriction. We are therefore motivated to extend this idea to the DG scheme in two dimensions, which we explore in the next chapter.

Chapter 5

The Modified DG scheme in 2D

5.1 Introduction

In one dimension, and when paired with an appropriate order explicit Runge-Kutta scheme, the CFL number of the DG scheme decreases with the order of approximation p as roughly $1/(2p+1)$. This restrictive condition is caused by the growth of the spectrum of the spatial discretization operator of the DG scheme [48]. In Chapter 3 we explored a possible approach to reducing the severity of this time step restriction, in which the one-dimensional DG scheme was modified through the introduction of $p+1$ parameters α_k, $k = 0, \ldots, p$, called 'flux-multipliers'. Using specific choices of these flux multipliers, particularly the multipliers associated with the highest order modes, we were able to take significantly larger CFL numbers while only introducing additional dispersive and diffusive errors.

In two dimensions, we encounter a similar scaling of the CFL number of the DG method. Indeed, as discussed in Chapter 4, the DG scheme of order p on triangular meshes, when paired with an explicit Runge-Kutta-$(p+1)$ scheme, has a CFL number that scales roughly as $\frac{1}{(2p+1)\left(1+\frac{4}{(p+2)^2}\right)}$. Again this restrictive condition is due to the growth of the spectrum of the spatial discretization operator. The similarity with the scaling of the CFL number in one dimension motivates us to consider an analogous modification of the DG scheme in 2D. To this end, we propose in this chapter a modified DG scheme that involves $p+1$ flux multipliers γ_k, $k = 0, \ldots, p$. These multipliers act analogously to the multipliers in the one-dimensional mDG scheme to scale the contributions from the jumps along the cell boundary to the numerical flux for the modes of the numerical solution. By following the procedures used in Chapter 4, we show that these modifications alter the spectrum of the DG method and allow us to take larger CFL numbers.

We also investigate how the local superconvergence and superaccuracy properties of the DG method are modified through the multipliers γ_k. Since the results of Chapter 4 rely on the properties of the Fourier modes of the numerical solution, we can extend them immediately by investigating how the modes are modified by the multipliers. In particular, we show that modifying the m highest order multipliers lowers the accuracy in dissipation and dispersion to $\mathcal{O}(h^{2p+1-m})$. We observe these additional errors in several numerical examples. We also present an example that compares the DG and modified DG methods with equivalent computational effort and demonstrate that the modified method can obtain significantly more accuracy on fine structures due to its smaller cell size.

The rest of this chapter is organized as follows. In Section 5.2 we introduce the modified DG scheme in two dimensions, and proceed to extend our Fourier analysis used in Chapter 4 to this scheme in Section 5.3. We then investigate the effects of the modifications to the spectrum of the DG method in Section 5.4, and we use linear algebra software to determine what choices of the multipliers γ_k will yield the largest CFL numbers. We then present several numerical examples in Section 5.5 in order to observe the effects of the modifications on the dissipation and dispersion errors of the scheme, and the potential benefits of the scheme to reduce computational effort or capture fine structures of the numerical solution.

5.2 Modified DG Discretization

Following an analogous procedure to that used in Chapter 3, we consider the application of the DG scheme to the two-dimensional system of conservation laws (1.2.14) in the form similar to (1.2.23)

$$2|\Omega_j|\frac{d}{dt}\mathbf{c}_{jki} = -\oint_{\partial\Omega_j} \mathbf{n}\cdot\mathbf{F}(\mathbf{U}_j^*)\psi_{ki}\,ds + 2|\Omega_j|\iint_{\Omega_0} \mathbf{F}(\mathbf{U}_j)\cdot J_j^{-1}\nabla\psi_{ki}\,dA.$$

Applying the divergence theorem to the volume integral again we obtain

$$2|\Omega_j|\frac{d}{dt}\mathbf{c}_{jki} = -\oint_{\partial\Omega_j} \mathbf{n}\cdot[[\mathbf{F}(\mathbf{U}_j)]]\psi_{ki}\,ds - 2|\Omega_j|\iint_{\Omega_0} J_j^{-1}\nabla\cdot\mathbf{F}(\mathbf{U}_j)\psi_{ki}\,dA, \qquad (5.2.1)$$

where $[[\mathbf{F}(\mathbf{U}_j)]] = \mathbf{F}(\mathbf{U}_j^*)-\mathbf{F}(\mathbf{U}_j)$ is the jump in numerical flux along the cell boundary $\partial\Omega_j$. We proceed by noticing again that these jump terms should be small on smooth solutions and their modification should not affect the formal accuracy of the scheme. Hence, we propose modifications to the DG scheme (5.2.1) through the introduction of parameters

γ_k, for $k = 0, \ldots, p$, in the following way

$$2|\Omega_j| \frac{d}{dt} \mathbf{c}_{jki} = -\gamma_k \oint_{\partial\Omega_j} \mathbf{n} \cdot [[\mathbf{F}(\mathbf{U}_j)]] \psi_{ki} \, ds - 2|\Omega_j| \iint_{\Omega_0} J_j^{-1} \nabla \cdot \mathbf{F}(\mathbf{U}_j) \psi_{ki} \, dA. \tag{5.2.2}$$

We again expect that this 2D mDG scheme will perform similarly to the original DG scheme on smooth solutions where the jump terms are small. Note that we have chosen the multipliers γ_k to depend only on the order of the polynomial basis function k and not on the specific index of the basis function i. This, of course, makes the implementation of these modifications simpler and helps with our analysis by reducing the number of different parameters which we must consider. More importantly, however, as we detail in the sections below there does not seem to be any benefit in terms of superaccuracies or stability when considering multipliers which depend on both k and i.

In the remainder of this chapter we will extend some of our superconvergence and superaccuracy analysis in Chapter 4 to this mDG scheme and we will investigate what increases to the CFL number of the scheme we can achieve using the flux multipliers.

5.3 Fourier Analysis

Following the same procedure as in Section 4.2 we can apply the mDG scheme (5.2.2) to the linear advection equation (4.1.1) to obtain

$$\frac{d}{dt} c_{jki} + \iint_{\Omega_0} \boldsymbol{\alpha} \cdot \nabla \psi_{ki} U_j \, dA = -\gamma_k \int_{\partial\Omega_0^-} (\boldsymbol{\alpha} \cdot \mathbf{n}) [[U_j]] \psi_{ki} \, ds, \tag{5.3.1}$$

where $[[U_j]] = U_j^* - U_j$ and the Riemann state U_j^* is defined using the upwind flux (4.2.3). We continue as in Section 4.2 and derive the PDE which the numerical solution U_j satisfies by multiplying (5.3.1) by ψ_{ki} and summing over $k = 0, \ldots, p$ and $i = 0, \ldots, k$ to obtain

$$\frac{\partial}{\partial t} U_j + \boldsymbol{\alpha} \cdot \nabla U_j = -\sum_{k=0}^{p} \sum_{i=0}^{k} \gamma_k \left[\int_{\partial\Omega_0^-} (\boldsymbol{\alpha} \cdot \mathbf{n}) [[U_j]] \psi_{ki} \, ds \right] \psi_{ki}. \tag{5.3.2}$$

Next, we look for solutions of the form $U_j(\xi, \eta, t) = \hat{U}_j(\xi, \eta) e^{-||\mathbf{a}||\omega t}$ where $\hat{U}_j(\xi, \eta)$ is a polynomial in ξ and η. We also transform (5.3.2) into the (ζ, σ)-coordinates using the transformation (4.3.2) in order to write

$$-\omega h_j \hat{U}_j + \frac{\partial}{\partial \zeta} \hat{U}_j = -\sum_{k=0}^{p} \sum_{i=0}^{k} \gamma_k \left[\int_{\partial\breve{\Omega}_0^-} n_\zeta [[\hat{U}_j]] \breve{\psi}_{ki} \, ds \right] \breve{\psi}_{ki}. \tag{5.3.3}$$

This process has been entirely analogous to what was done in Chapter 4, but now we see in (5.3.3) that the modifiers γ_k have altered the right hand side of (4.3.4). As a consequence, the projection $\mathcal{R}_{p+1}$ will be altered. We denote the new projection, which depends on the modifiers γ_k, by $\tilde{\mathcal{R}}_{p+1}$ and use it to simplify the right hand side of (5.3.3) using the following proposition.

Proposition 5.1. *We define a projection of the jump function* $[[\hat{U}_j]]$, *which we denote as* $\tilde{\mathcal{R}}_{p+1}[[\hat{U}_j]](\zeta,\sigma)$, *into the space of polynomials in* ζ *and* σ *satisfying the following conditions,*

$$\int_{\partial\check{\Omega}_0^+} n_\zeta \tilde{\mathcal{R}}_{p+1}[[\hat{U}_j]]\check{\psi}_{ki}\,ds - \iint_{\check{\Omega}_0} \tilde{\mathcal{R}}_{p+1}[[\hat{U}_j]]\frac{\partial}{\partial\zeta}\check{\psi}_{ki}\,dA = (\gamma_k - 1)\int_{\partial\check{\Omega}_0^-} n_\zeta \tilde{\mathcal{R}}_{p+1}[[\hat{U}_j]]\check{\psi}_{ki}\,ds, \quad (5.3.4)$$

for $k = 0, \dots, p$ *and* $i = 0, \dots, k$, *and*

$$\int_{\partial\check{\Omega}_0^-} n_\zeta \tilde{\mathcal{R}}_{p+1}[[\hat{U}_j]]\check{\psi}_{ki}\,ds = \int_{\partial\check{\Omega}_0^-} n_\zeta [[\hat{U}_j]]\check{\psi}_{ki}\,ds, \quad (5.3.5)$$

for $k = 0, \dots, p$ *and* $i = 0, \dots, k$. *When* Ω_j *is a type I or III cell we require that* $\tilde{\mathcal{R}}_{p+1}[[\hat{U}_j]](\zeta,\sigma)$ *is a polynomial of degree* $p+1$ *in* ζ *and of degree* p *in* σ. *When* Ω_j *is a type II cell we require that* $\tilde{\mathcal{R}}_{p+1}[[\hat{U}_j]](\zeta,\sigma)$ *is a polynomial of degree* $p+1$ *in* ζ *and of degree* $2p$ *in* σ. $\tilde{\mathcal{R}}_{p+1}[[\hat{U}_j]](\zeta,\sigma)$ *is then uniquely determined by* (5.3.4) *and* (5.3.5).

Then, using this projection, the forcing term on the right hand side of (5.3.3) *can be written as*

$$\sum_{k=0}^{p}\sum_{i=0}^{k}\gamma_k\left[\int_{\partial\check{\Omega}_0^-} n_\zeta [[\hat{U}_j]]\check{\psi}_{ki}\,ds\right]\check{\psi}_{ki} = \frac{\partial}{\partial\zeta}\tilde{\mathcal{R}}_{p+1}[[\hat{U}_j]](\zeta,\sigma). \quad (5.3.6)$$

Proof. The proof of this proposition follows the same argument as the proof of Proposition 4.1. This time, however, we verify (5.3.6) by multiplying the expression by $\check{\psi}_{ki}$, integrating over $\check{\Omega}_0$, applying the divergence theorem, and using the orthogonality relations (5.3.4) to obtain

$$\begin{aligned}\gamma_k\int_{\partial\check{\Omega}_0^-} n_\zeta [[\hat{U}_j]]\check{\psi}_{ki}\,ds &= \iint_{\check{\Omega}_0}\left(\frac{\partial}{\partial\zeta}\tilde{\mathcal{R}}_{p+1}[[\hat{U}_j]]\right)\check{\psi}_{ki}\,dA \\ &= \oint_{\partial\check{\Omega}_0} n_\zeta \tilde{\mathcal{R}}_{p+1}[[\hat{U}_j]]\check{\psi}_{ki}\,ds - \iint_{\check{\Omega}_0}\tilde{\mathcal{R}}_{p+1}[[\hat{U}_j]]\frac{\partial}{\partial\zeta}\check{\psi}_{ki}\,dA. \quad (5.3.7) \\ &= \gamma_k\int_{\partial\check{\Omega}_0^-} n_\zeta [[\hat{U}_j]]\check{\psi}_{ki}\,ds, \quad (5.3.8)\end{aligned}$$

which is true by (5.3.5). □

Using this proposition we can write (5.3.3) in the following compact form

$$-\omega h_j \hat{U}_j + \frac{\partial}{\partial \zeta}\hat{U}_j = -\frac{\partial}{\partial \zeta}\tilde{\mathcal{R}}_{p+1}[[\hat{U}_j]](\zeta, \sigma). \tag{5.3.9}$$

Note that because the defining relations of the projection $\mathcal{R}_{p+1}$ have been altered, any orthogonality properties of $\tilde{\mathcal{R}}_{p+1}$ will be directly determined by the choice of multipliers in (5.3.4). As a consequence, since much of the superconvergence properties of the scheme result from the orthogonality of this projection (e.g. Theorems 4.2-4.4), we have the possibility that the superconvergence properties of the modified scheme can be specifically manipulated. To determine this more precisely we repeat the procedures used in [2] and [47] in determining the orthogonality of the $\mathcal{R}_{p+1}$ projection in order to establish some properties of the modified projection $\tilde{\mathcal{R}}_{p+1}$.

Proposition 5.2. *Let Ω_j be a cell of type I and let q be the smallest index for which $\gamma_q \neq 1$. Then the projection $\tilde{\mathcal{R}}_{p+1}[[\hat{U}_j]]$ of the jump function along the inflow boundary satisfies the following orthogonality relation*

$$\int_{\partial\check{\Omega}_0^+} n_\zeta \left(\tilde{\mathcal{R}}_{p+1}[[\hat{U}_j]]\right) \sigma^k \, ds = 0, \tag{5.3.10}$$

for all $k = 0, \ldots, q-1$.

Proof. We follow the same arguments as in the proof of expression (3.62) of Theorem 3.5 in [2]. We can also prove (5.3.10) directly by simply replacing $\check{\psi}_{ki}$ by σ^k in (5.3.4). □

Proposition 5.3. *Let Ω_j be a cell of type II or III and let q be the smallest index for which $\gamma_q \neq 1$. Then the projection $\tilde{\mathcal{R}}_{p+1}[[\hat{U}_j]]$ of the jump function along the inflow boundary satisfies the following orthogonality relations*

$$\iint_{\check{\Omega}_0} \tilde{\mathcal{R}}_{p+1}[[\hat{U}_j]]\check{\psi}_{ki} \, dA = 0, \tag{5.3.11}$$

for all $k = 0, \ldots, q-2$ and $i = 0, \ldots, k$, and

$$\int_{\partial\check{\Omega}_0^+} n_\zeta \tilde{\mathcal{R}}_{p+1}[[\hat{U}_j]]\check{\psi}_{ki} \, ds = 0, \tag{5.3.12}$$

for all $k = 0, \ldots, q-1$ and $i = 0, \ldots, k$.

Proof. We follow similar arguments as used in the proof of Theorem 1 in [47]. In particular, we can prove both (5.3.11) and (5.3.12) directly by considering the monomial basis $\{(\zeta - 1)^m, (\zeta-1)^{q-1}\sigma, \dots, \sigma^m\}$ for $m = 0, \dots, p$ in (5.3.4). □

Now that we have obtained the PDE (5.3.9) for the numerical solution of the modified DG scheme, analogous to the PDE (4.3.13) considered for the classical DG scheme in Chapter 4, and we have established some of the orthogonality properties of the modified $\tilde{\mathcal{R}}_{p+1}$ projection, we can extend much of the analysis performed in Chapter 4 to the modified scheme. Since much of the analysis can be repeated almost verbatim, replacing $\mathcal{R}_{p+1}$ by $\tilde{\mathcal{R}}_{p+1}$ and changing the orthogonality properties to those of $\tilde{\mathcal{R}}_{p+1}$ where appropriate, we will not repeat these results in full. Instead we will simply refer to these theorems and note important differences in their results as they pertain to the modified scheme.

To begin, we follow the same procedure leading to Theorem 4.1 to obtain that the Fourier modes of the modified DG method (5.3.1) for linear hyperbolic problems in two dimensions which are polynomials in ζ and σ can be written as

$$\hat{U}_j(\zeta,\sigma) = \tilde{\mathcal{F}}_p \circ \tilde{\mathcal{G}}_p^{-1} \hat{U}_{j+}, \tag{5.3.13}$$

where the projections $\tilde{\mathcal{F}}_p$ and $\tilde{\mathcal{G}}_p^{-1}$ are defined in analogous ways to the definition of $\mathcal{F}_p$ in (4.3.17) and the definition of $\mathcal{G}_p^{-1}$ in Theorem 4.1. The Fourier modes also have the expansion

$$\begin{aligned}\hat{U}_j(\zeta,\sigma) = \hat{U}_{j+}(\zeta_0,\sigma)e^{\omega h_j(\zeta-\zeta_0)} + \left[\tilde{\mathcal{R}}_{p+1}[[\hat{U}_j]](\zeta_0,\sigma) - [[\hat{U}_j]]\right] e^{\omega h_j(\zeta-\zeta_0)} \\ - \tilde{\mathcal{R}}_{p+1}[[\hat{U}_j]](\zeta,\sigma) - \sum_{k=1}^{\infty} (\omega h_j)^k \tilde{\mathcal{R}}_{p+1}^{(-k)}[[\hat{U}_j]](\zeta,\sigma).\end{aligned} \tag{5.3.14}$$

Using this expansion and the orthogonality properties of $\tilde{\mathcal{R}}_{p+1}$ in Propositions 5.2 and 5.3 we can extend the local superconvergence properties presented in Section 4.4 to the modified scheme. We omit restating these results in full and instead note the important difference for their analogous modified results. For what follows, we assume q is the smallest index for which $\gamma_q \neq 1$. Beginning with the results of Theorem 4.2 concerning the local superconvergence of type I cells, we note that for the modified scheme the orthogonality property (4.4.3) will only hold for $m = 0, \dots, q-1$. Next, for the results of Theorem 4.3 concerning the local superconvergence of type II cells we note that (4.4.6) will only hold to $\mathcal{O}(h_j^{p+q+1-m})$ for $m = 0, \dots, q-1$, and (4.4.7) will only hold to $\mathcal{O}(h_j^{p+q-m})$ for $m = 0, \dots, q-2$. Similarly, for the results of Theorem 4.4 concerning type III cells, (4.4.14) will only hold to $\mathcal{O}(h_j^{p+q+1-m})$ for $m = 0, \ldots, q-1$, and (4.4.15) will only hold to $\mathcal{O}(h_j^{p+q-m})$ for $m = 0, \ldots, q-2$.

Finally, we extend our results concerning the superaccuracy of the DG method in terms of dissipation and dispersion errors to the modified scheme. We again consider a uniform mesh on the unit square domain Σ and we look for solutions of the form (4.5.2). We compute the Fourier mode solutions on each rectangle Σ_{jl} and find that solutions of this form will exist when the determinant of the system (4.5.4) is zero. We symbolically compute these determinants, which now depend of the modifiers γ_k, and compute their Taylor series around $h = 0$ in order to verify the following result up to $p = 3$.

Theorem 5.1. *Let U be a numerical solution of the modified DG method* (5.3.1) *applied to the linear problem*(4.1.1) *on the square domain Σ with a uniform computational mesh and suppose U is of the form* (4.5.2). *Then the numerical wavenumbers $\tilde{\kappa}_1$ and $\tilde{\kappa}_2$ satisfy*

$$a\tilde{\kappa}_1 + b\tilde{\kappa}_2 = ||\mathbf{a}||\omega + \mathcal{O}(h^{p+q}), \tag{5.3.15}$$

where q is the smallest index for which $\gamma_q \neq 1$. That is, the local orders of errors in dissipation and dispersion of the scheme along the direction of flow are $p + q$.

It is interesting to note that this theorem holds even in the case when we take the modifiers to depend on the index i in (5.3.1), or even take different choices of multipliers in each sub-triangle of Σ_{jl}. The orders of dissipation and dispersion errors are still determined by the smallest index q for which $\gamma_{qi} \neq 1$, regardless of i.

5.4 Stability

In this section we will investigate what improvements to the usual CFL number of the DG scheme can be obtained using the multipliers γ_k. We pair the mDG spatial discretization with an explicit order $p + 1$ Runge-Kutta time integration method. As discussed in the previous chapter for the classical DG method, with this time integration scheme the CFL condition

$$\Delta t \leq \frac{1}{(2p+1)\left(1 + \frac{4}{(p+2)^2}\right)} \frac{h}{||\mathbf{a}||},$$

provides a fairly tight bound on the time step Δt in order to ensure linear stability. We found this condition by computing each spectral value λ_{knm} that is a root of the determinant

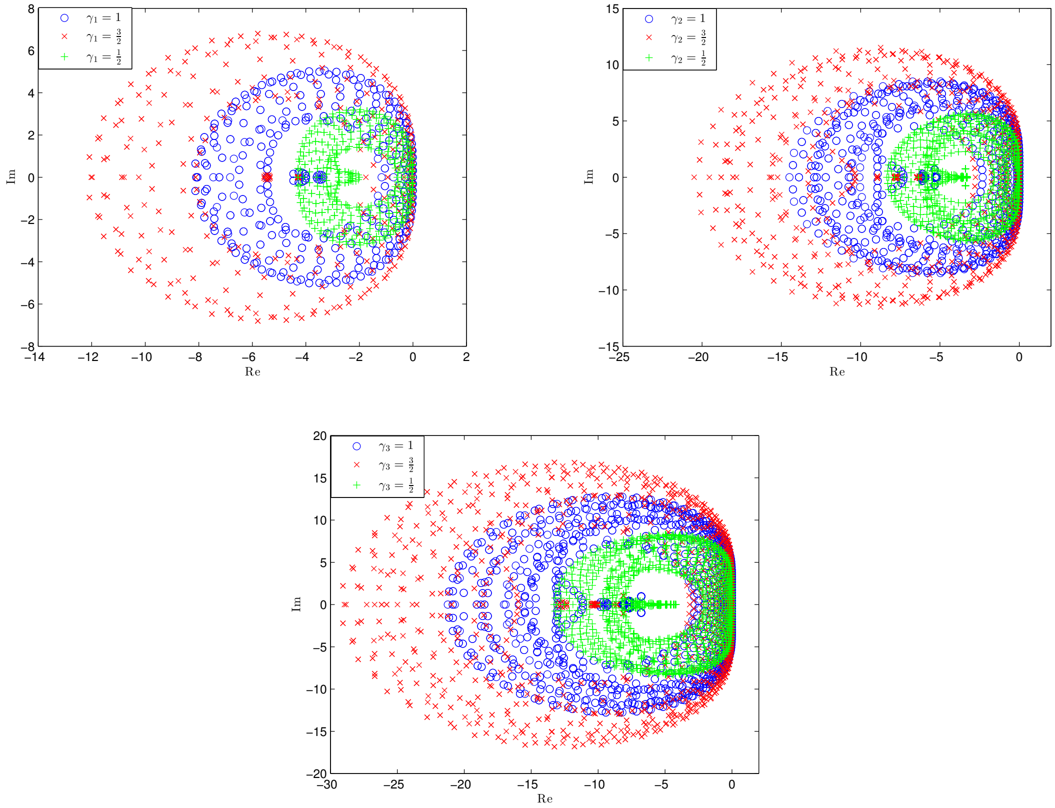


Figure 5.1: Spectral values λ_{knm} of the spatial 2D DG discretization for the linear advection equation, for the $p = 1$ and 2 (top) and $p = 3$ (bottom), with $N = M = 10$. We show in each figure the spectral values for $\gamma_p = 1, \frac{3}{2}$, and $\frac{1}{2}$.

C_p of the system (4.5.4) with $\nu = \exp\left(\frac{2\pi ni}{N}\right)$ and $\mu = \exp\left(\frac{2\pi mi}{M}\right)$. For the modified DG scheme (5.3.1) this determinant will depend on the multipliers γ_k. We again resort to numerically calculating these roots for various choices of multipliers and determine the time step restriction of the resulting scheme by finding the largest CFL number such that the scaled spectral values are contained in the absolute stability region of the RK-$(p+1)$ scheme.

In Figure 5.1 we show the spectral values λ_{knm} of the two-dimensional DG spatial discretization for the $p = 1, 2$, and 3 schemes, respectively, with different values for the highest multiplier γ_p in each case. In each figure, we show with the 'o' marker the spectrum

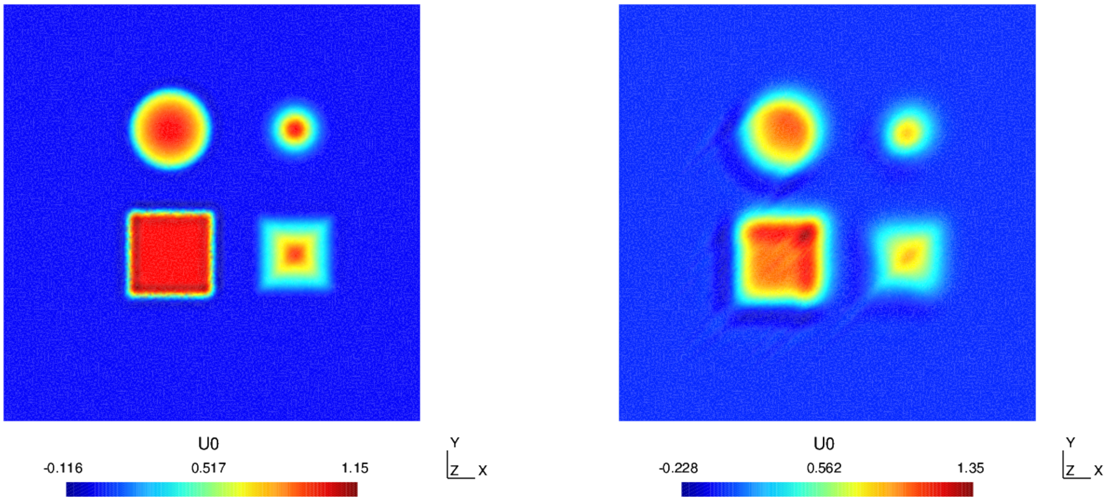
p	γ_p	CFL	Relative Increase
1	1.000	0.232	3.58
	0.182	0.834	
2	1.000	0.165	2.07
	0.240	0.342	
3	1.000	0.124	1.89
	0.278	0.234	

Table 5.1: Largest CFL numbers obtained with the modified DG scheme on the 2D linear advection equation for $p = 1, 2,$ and 3, only modifying the highest order coefficient. Relative increase is calculated as the ratio between the increased CFL of the modified scheme, divided by the CFL number of the original DG scheme.

for $\gamma_p = 1$, which is the spectrum of the original DG scheme, together with the spectra for $\gamma_p = \frac{3}{2}$ and $\gamma_p = \frac{1}{2}$ with the 'x' and '+' markers, respectively. As with the one-dimensional mDG scheme in Chapter 3 we again see from these figures that, in general, the modification of the highest modifier has the effect of scaling the spectrum. Specifically, increasing the γ_p multiplier increases the size of the spectrum, while decreasing the γ_p multiplier reduces the size of the spectrum. From this, we again see that we are able to choose a larger CFL number then the usual DG scheme when $\gamma_p < 1$.

To determine what choices of γ_p give us the most relaxed time-step restriction, we implement a Nelder-Mead optimization algorithm in MATLAB. At each iteration, for a given choice of multiplier γ_p, the program calculates the spectral values λ_{knm} for varying values of θ and uses this spectrum to find the largest CFL number so that the complete spectrum of $CFL \cdot \lambda_{knm}$ is contained within the absolute stability region of RK-$(p+1)$ via a bisection algorithm. In Table 5.1 we present the largest CFL number we were able to obtain using this program for schemes of order $p = 1, 2,$ and 3, together with the value of γ_p for which the scheme obtains this CFL number. From this table we see that we are able to achieve a significant increase in the usual CFL number of the DG scheme through modification to only the highest order multiplier.

When we consider modifications of more than just the highest order multiplier, and extend our search for choices of these multipliers that allow us to take larger CFL numbers, we find that there is little improvement to the CFL numbers reported in Table 5.1. For example, considering the $p = 3$ scheme and searching for an optimal set of multipliers γ_1, γ_2, and γ_3, returns that both γ_1 and γ_2 are very close to 1 and the optimal CFL number is very close to 0.234. This result is in direct contrast with the results seen for



(a) $\gamma_1 = 1$

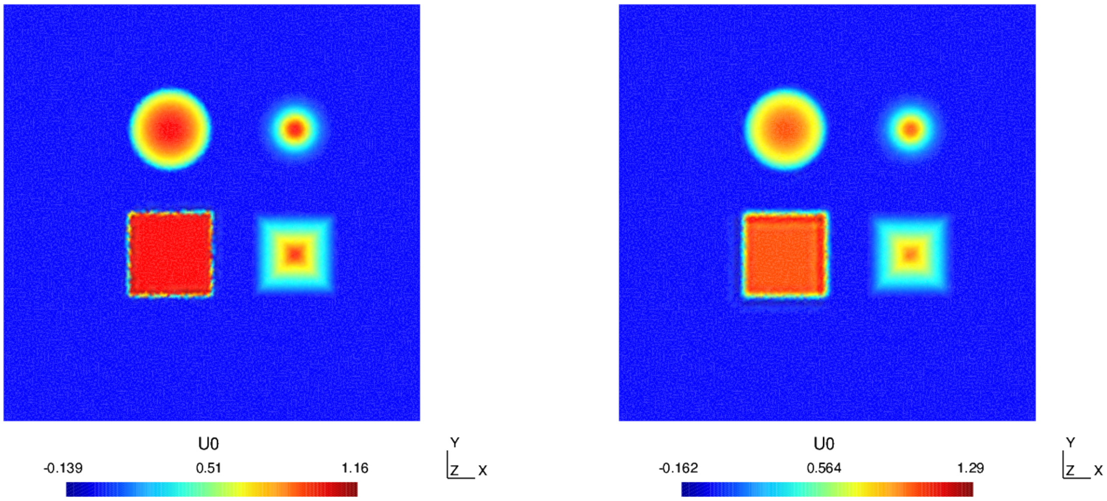
(b) $\gamma_1 = 0.182$

Figure 5.2: Linear advection equation (4.8.5)-(5.5.1), $p = 1$ on a mesh of 26524 triangles. Shown at $t = 0.3$. We show the solution of the mDG scheme with $\gamma_1 = 1$ and $CFL = 0.232$ (left) and with $\gamma_1 = 0.182$ and $CFL = 0.834$ (right).

the one-dimensional mDG scheme where modifying several multipliers had the potential to yield even larger CFL numbers. It is possible that such modifications are still possible but require higher-order schemes more multipliers to be modified. For now, we proceed to numerically verify our results concerning the mDG scheme in two-dimensional in the following section.

5.5 Numerical Examples

In this section we will present several numerical examples of the modified DG scheme in two-dimensional to demonstrate its potentially more relaxed stability restriction and qualitatively observe its effects on the dissipation and dispersion errors. In each test below we pair the DG spatial discretization with a explicit order $p+1$ RK time-integration scheme and use the largest CFL number possible.



(a) $\gamma_2 = 1$

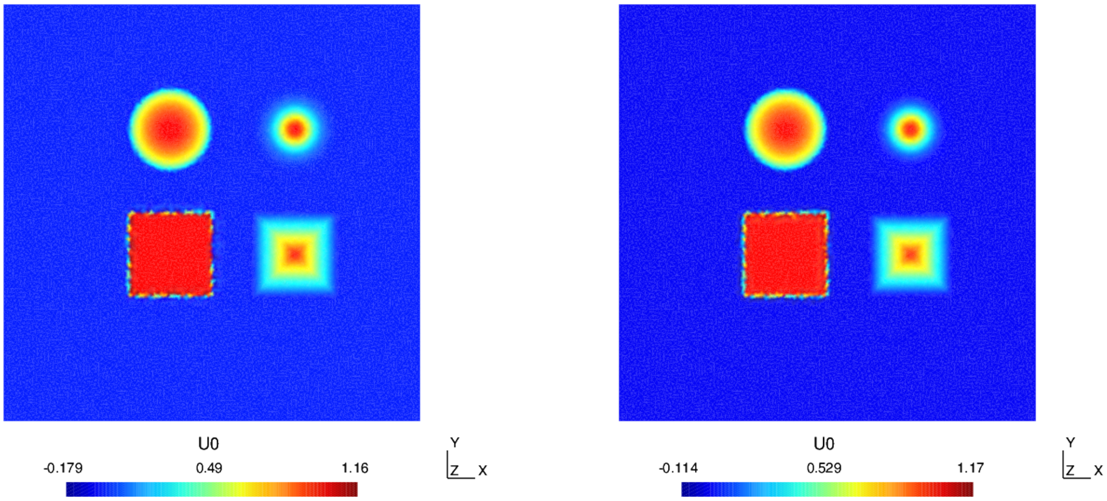
(b) $\gamma_2 = 0.24$

Figure 5.3: Linear advection equation (4.8.5)-(5.5.1), $p = 2$ on a mesh of 26524 triangles. Shown at $t = 0.3$. We show the solution of the mDG scheme with $\gamma_2 = 1$ and $CFL = 0.165$ (left) and with $\gamma_2 = 0.24$ and $CFL = 0.342$ (right).

In our first example we consider a simple linear problem with and initial condition which contains several different waveforms. That is, we consider again the linear advection initial value problem (4.8.5) with the initial condition

$$u_0(x,y) = \begin{cases} 1 & 0 \le x \le 0.2,\ 0 \le y \le 0.2, \\ 10\min(x-0.3, 0.5-x, y, 0.2-y) & 0.3 \le x \le 0.5,\ 0 \le y \le 0.2, \\ \sqrt{\max(1-100((x-0.1)^2+(y-0.4)^2), 0)} & 0 \le x \le 0.2,\ 0.3 \le y \le 0.5, \\ \exp(-500((x-0.4)^2+(y-0.4)^2) & 0.3 \le x \le 0.5,\ 0.3 \le y \le 0.5, \\ 0 & \text{otherwise.} \end{cases} \tag{5.5.1}$$

This initial profile consists of a square wave, a square base pyramid, a half-ellipse, and a Gaussian pulse. We implement the mDG scheme for $p = 1, 2$ and 3, modifying only the highest order multiplier γ_p using the values detailed in Table 5.1. We implement the method on an unstructured mesh of 26524 triangles. Since limiting can potentially destroy



(a) $\gamma_3 = 1$

(b) $\gamma_3 = 0.278$

Figure 5.4: Linear advection equation (4.8.5)-(5.5.1), $p = 3$ on a mesh of 26524 triangles. Shown at $t = 0.3$. We show the solution of the mDG scheme with $\gamma_3 = 1$ and $CFL = 0.124$ (left) and with $\gamma_3 = 0.278$ and $CFL = 0.234$ (right).

fine structures of the numerical approximation, we do not implement a limiter in these tests so that we can observe the effects of the dissipation and dispersion errors that the modifications will have.

In Figures 5.2, 5.3, and 5.4 we show the results of this test for $p = 1, 2$, and 3, respectively, at $t = 0.3$. In the first test, we see that the modification of the highest multiplier has introduced a significant amount of dissipation and dispersion errors which appears to smear the waveforms and even create a trailing error along the direction of flow. However, with the error has come a significant increase in the size of the time step Δt. Using the CFL condition proposed in Chapter 4 we find that the classical DG scheme requires 429 time steps to reach the final time, while the modified scheme requires only 120. Examining the $p = 2$ and $p = 3$ tests we see similar results to those observed for the one-dimensional mDG scheme. The effects of modifying the highest order multiplier on the dissipation and dispersion errors of the scheme become less severe as the order p increases, while we are still able to obtain significant increases in time-step sizes. We note that the classical DG

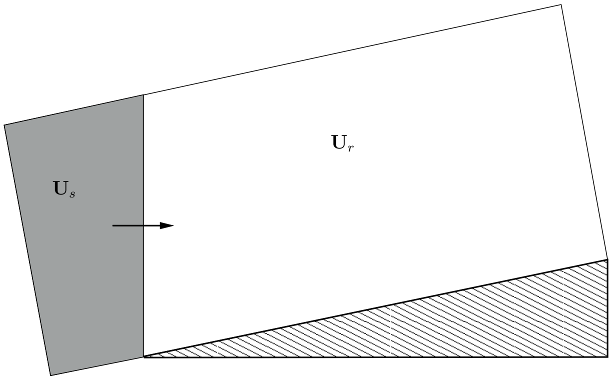


Figure 5.5: Diagram of the double Mach reflection test problem for the Euler equations.

scheme with $p = 2$ required 619 time steps to reach final time, while the modified scheme only required 299. Similarly, the DG scheme with $p = 3$ required 804 time steps while the mDG scheme required only 426.

To demonstrate the efficacy of the modified scheme for a nonlinear problem we consider a more involved example. A common test for the two-dimensional Euler equations is the so-called double Mach reflection problem, discussed in [70]. In this problem, we consider a rectangular domain $\Omega = [0, 4] \times [0, 1]$ containing a right-moving shock wave impinging on a reflecting wedge, as depicted in Figure 5.5. We solve the two-dimensional Euler equations (4.8.9)-(4.8.10) with an initial condition consisting of a shock state $\mathbf{U}_s$ to the left of the shock front which has $\rho = 8$, $u = 8.25\cos(\frac{\pi}{6})$, $v = 8.25\sin(\frac{\pi}{6})$, and $P = 116.5$ so that the shock wave is traveling at Mach 10. The state to the right of the shock front $\mathbf{U}_r$ is taken to be at rest with $\rho = 1$, $u = v = 0$, and $P = 1$. The boundary conditions along the upper edge are taken to match the position of the shock wave, inflow of the shock state along the left edge, and outflow along the right edge. Finally, the bottom boundary is assigned to be a reflecting boundary. Since this problem involves strong shocks we implement a limiter in our experiments, namely the Barth-Jesperson limiter [9].

In Figure 5.6 we show the results of the classical DG scheme on an unstructured mesh of 120926 triangles at the final time of $T = 0.2$, which required 3865 time steps. We also show the results of the mDG scheme with $\gamma_1 = \frac{1}{2}$ and a CFL number 2.5 times larger than in the classical DG scheme on the same mesh in Figure 5.7. The two numerical solutions appear visually identical, however the mDG scheme requires only 1545 time steps to reach the final time, thus marking a significant performance benefit. A possible explanation for

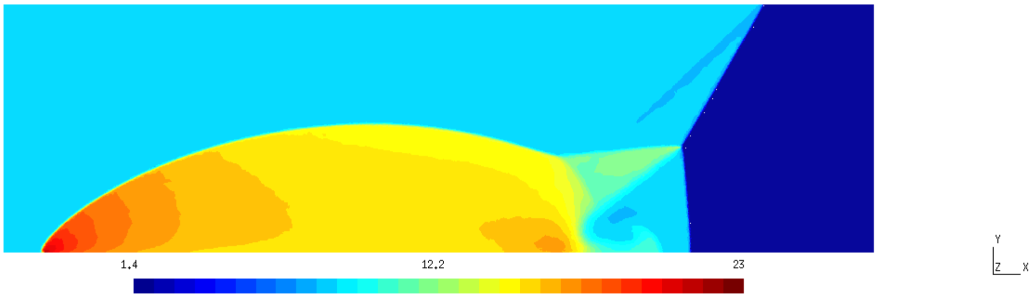


Figure 5.6: Double Mach reflection test. We show the results of the DG method with $p = 1$ on a mesh of 120926 triangles at time $t = 0.2$.

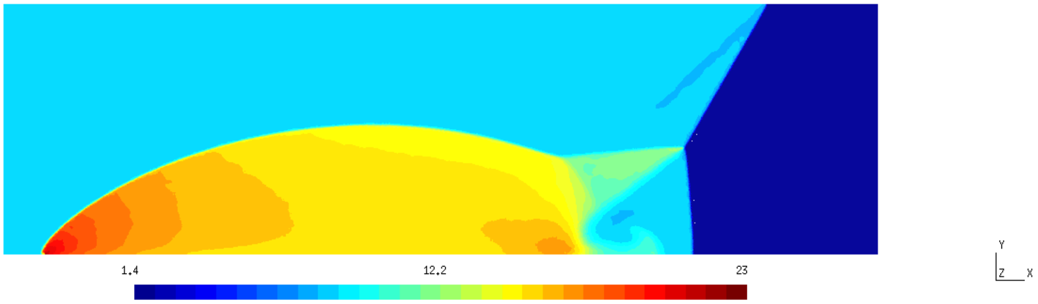


Figure 5.7: Double Mach reflection test. We show the results of the mDG method with $p = 1$ and $\gamma_1 = \frac{1}{2}$ on a mesh of 120926 triangles at time $t = 0.2$.

this could be an analogous phenomenon to what was observed for the Euler equations in one dimension presented in Section 3.5.4. In both examples it is possible that the scheme which requires fewer time-steps is less damaged by repeated applications of the limiter.

In Figure 5.8 we show the results of the mDG method with with $\gamma_1 = \frac{1}{2}$ and a CFL number 2.5 times larger than in the classical DG scheme on an unstructured mesh of 228654 triangles, which requires 2262 time-steps to reach the final time. Since the computational effort of each scheme should be proportional to the number of cells in the mesh multiplied by the number of time steps required, we find that this scheme has a similar computational effort to example with the classical DG scheme consider in Figure 5.6. We see, however, that the mDG scheme seems to obtain better accuracy on fine structures of the solution,

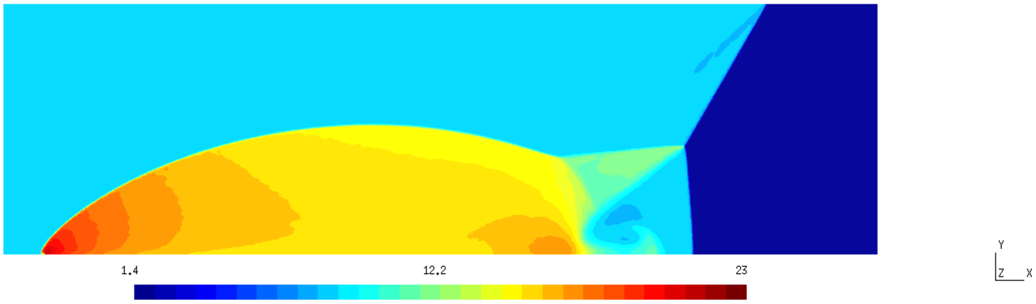


Figure 5.8: Double Mach reflection test. We show the results of the mDG method with $p = 1$ and $\gamma_1 = \frac{1}{2}$ on a mesh of 228654 triangles at time $t = 0.2$.

particularly noticeable in the contact region, due to its more refined mesh. This suggests a potential benefit to implementing the mDG scheme on a more refined mesh, for an equivalent computational effort.

5.6 Discussion

In this chapter we have extended our modified DG scheme proposed in Chapter 3 to two-dimensional systems on triangular meshes. The modifications scale the contributions from the jumps in the numerical flux along the cell boundaries through the introduction of so-called flux multipliers γ_k.

By applying the modified scheme to a simple linear problem, we show that the multipliers alter the $\tilde{\mathcal{R}}_{p+1}[[U_j]]$ projection of the jump along cell boundary. Since much of the superconvergence analysis performed in Chapter 4 relies of the properties of this projection, we can immediately extend our local superconvergence results to the modified scheme by making the appropriate changes to the proofs. In particular, we show that when the γ_q modifier is taken not equal to one, the order of accuracy of the method in terms of dissipation and dispersion is $p + q$. The modifications therefore introduce additional dispersive and diffusive errors to the numerical solution, which we observe in our numerical examples.

When we search for what choices of the multipliers will yield the largest CFL number we find that we can take significantly larger CFL numbers when only the highest order multiplier γ_p is modified. We also find that there is little benefit to modifying two or three of the highest order modifiers, since this result in very little gains in the maximum CFL number, or results in an unstable semi-discrete scheme. It is possible that gains to the CFL number may still be possible through modifying more than three modifiers, but this remains an open question.

We present several numerical examples to observe the performance of the modified scheme. On linear problems we observe that the modified scheme performs similarly to the DG method on smooth profiles, but incurs additional dissipative and diffusive errors which are particularly visible on strong discontinuities. In our non-linear examples, we observe that when a limiter is applied there is very little difference between the DG and mDG numerical solutions, while for a fixed computational effort the mDG achieves significantly more accuracy on the fine structures of the solution due to its greater cell refinement.

Several questions remain open for further study. First, more testing is necessary to determine what choices of multipliers will be optimal in the sense of the trade-off between accuracy and efficiency. Additionally, more testing is required to extend the search for optimal choices of multipliers presented in Table 5.1 for higher-order schemes. Finally, more study is required to investigate the application of the modified scheme on other cell geometries, as well as in higher-dimensional problems.

Chapter 6

Conclusions and Future Work

In this thesis we have investigated the superconvergence, superaccuracy, and stability properties of the discontinuous Galerkin finite element method. By considering a simple linear hyperbolic problem we have established several results in one and two dimensions on both uniform and non-uniform meshes. We have also demonstrated, through simple modifications to the scheme, that the superconvergence, superaccuracy, and stability properties of the DG method may be manipulated. We use these proposed modifications to construct schemes that are formally less accurate but significantly more stable than the classical DG method. Here we briefly summarize the work presented in the above chapters and discuss how these topics may be further investigated in future works.

Our analysis began in Chapter 2 where we applied the DG method to a simple linear advection problem. The key steps in the analysis were the derivation of a PDE that the numerical solution satisfies on each cell and the application of classical Fourier analysis to investigate the solutions of this PDE. After finding the Fourier modes of the numerical solution we established our primary results. We showed that the local error of the scheme is order $p+2$ superconvergent at the right Radau points inside the cell, and order $2p+2$ at the downwind points. We also showed that the Fourier modes of the numerical solution are closely related to the $\frac{p}{p+1}$ Padé approximant of the exponential e^z at the downwind points and used this result to establish the scheme's superaccurate order $2p+1$ and $2p+2$ errors in dissipation and dispersion, respectively. While these properties have been shown previously on uniform meshes [4][43] our analysis extends these results to non-uniform meshes. We then considered a uniform mesh with periodic boundary conditions and showed that under these assumptions the spectrum of the method can be decomposed into non-physical and physical modes, where the non-physical modes are damped out exponentially quickly in time. Using this decomposition, we proved that for a class of initial projection the DG solution will tend to a globally superconvergent form which obtains order $p + 2$ rate of convergence at the right Radau points on the interior of the cells and order $2p + 1$ rate of convergence at the downwind points of each cell.

In Chapter 3, we analysed the superconvergence, superaccuracy, and stability of the DG method under a slightly different perspective. Specifically, we investigated how these properties may be manipulated through modifications to the scheme. We proposed a new family of schemes, which we name the modified DG (mDG) scheme, obtained via simple modifications to the numerical fluxes through the introduction of several parameters, α_k, $k = 0, \ldots, p$. The original DG scheme can be seen as the special case of the mDG scheme when $\alpha_k = 1$ $\forall k$. By repeating the analysis preformed in Chapter 2 we showed that the superconvergence, superaccuracy, and stability of the mDG method can be manipulated by choosing different values for the parameters α_k. We also proved that the DG scheme is optimal in the sense that the modifications can only reduce the formal orders of accuracy in dissipation and dispersion. The modifications do, however, allow us to construct schemes which are more stable than the classical DG method. Indeed, we have constructed schemes which allow for time steps which are twice as large as usual through only taking $\alpha_p \neq 1$. We demonstrate with some numerical examples that there may be benefits to using the mDG method over the classical DG method, in particular when limiting is applied or the solution contains fine structures which would be more well-resolved under mesh refinement.

We continued our study of the DG method by moving to two-dimensional equations in Chapter 4. In previous works, the local superconvegence of the DG method had been investigated for meshes of quadrilaterals in [5] and triangles in [47] and [2]. In this Chapter, we used the approach developed in Chapter 2 to provide new proofs of the results in these works. We then made the simplifying assumption of a uniform mesh and periodic boundary conditions in order to extend our other one-dimensional results. Through symbolic calculation, we proved that the scheme obtains superaccurate order $2p + 1$ errors in dissipation and dispersion. We also proved that the spectrum of the method can be decomposed into non-physical and physical modes, analogously to the one-dimensional scheme. We then used this decomposition to prove that for a class of initial projections the numerical solution will tend towards a superconvergent form. Finally, we also proposed a new CFL condition for the scheme, motivated by the appearance of a parameter h_j in each cell, which geometrically can be seen to be the width of that cell along the direction of flow. We showed through some linear and non-linear numerical examples that this CFL condition can yield time steps which are significantly larger than those given using the commonly-used CFL condition which uses the inscribed radii of each cell.

Our final topic was then presented in Chapter 5, where we extend our modified DG scheme to two-dimensional problems. We modify the two-dimensional DG scheme in an analogous way to the one-dimensional scheme through the introduction of $p+1$ parameters γ_k, $k = 0, \ldots, p$. We then showed that these modifications to the DG method allow us once again to manipulate the superconvergence and superaccuracy properties of the method. We also proved, as in one dimension, that the DG scheme is the optimal choice in terms of formal orders of accuracy in dissipation and dispersion. We then used the modifications to construct schemes which are more stable than the usual DG method. In contrast to the one-dimensional scheme, we found that the stability is only significantly improved by modifying the highest order multiplier γ_p.

An immediate topic of further study is whether the spectral decomposition and global superconvergence results of Chapters 2 and 4 may be extended to non-uniform meshes. In one-dimension, it is known that on non-uniform meshes a numerical solution that is initially in a superconvergent form will remain in a superconvergent form [72]. It is therefore reasonable to assume that an extension of our analysis is plausible. Further topics of research arise when we note that the analysis performed in Chapters 2 and 4, while yielding interesting results for the linear equations, must be further extended in order to yield results for more practical problems. In particular, non-linear equations and systems must be considered. As is usual for this type or analysis, a possible approach to non-linear equations could be a linearisation argument in each cell, followed by our previously-used linear analysis. This approach may then be successful in yielding leading order error estimates and superconvergence/superaccuracy for these equations. The extension to systems should then be fairly straight-forward by considering the linearisation of the characteristic equations of the system. Another topic, concerning the analysis for Chapter 4, is whether the results verified through symbolic computation will hold for all orders p. It is therefore important that general proofs of these results be investigated.

Several questions remain regarding the modified DG methods proposed in Chapters 3 and 5. In particular, the affect the modifiers have on the global accuracy and what choices of modifiers are optimal in the sense of the trade-off between accuracy and efficiency. While some preliminary benchmarking was performed in [19], the results are not conclusive and no such testing has yet been performed for the two-dimensional scheme. It is also not clear how the limiter affects the modified scheme. While the examples we present show that there is little difference between the numerical solutions produced by the DG and mDG schemes when a limiter is applied these examples only consider the moment limiter. This interaction is still an open question, especially if other limiting strategies, such as shock-detection and artificial viscosity approaches, are employed. Finally, there is a possibility that the modifications to the DG scheme could serve other purposes than obtaining larger CFL numbers. Specifically, since the multipliers introduced scale the contributions to the modes of the numerical solution from the jumps at the cell boundaries it is possible that the multipliers could be chosen adaptively to serve as a limiting strategy.

One final topic of further study is the extension of this work to three-dimensional problems. In addition to the superconvergence and superaccuracy studies, the question of whether an analogous CFL condition to the one proposed in Chapter 4 can be derived for these higher-dimensional problems would be of significant practical interest.

APPENDICES

Appendix A

Mathematica Source Code

Here we give the Mathematica source code used to compute the determinant C_p of the system (4.5.4) described in Section 4.5. We note that in the code below we can calculate C_p for different values of p simply by changing its value in the first line. Moreover, we can compute the determinants C_p for the modified DG scheme described in Chapter 5 by changing the values given to the variable `gamma`, currently defaulted to a list of ones which corresponds to the classical DG scheme.

```
p = 1;
phi[xi_,eta_,k_,i_] := JacobiP[k-i,0,2*i+1,1-2*xi]*(1-xi)^i*
                       LegendreP[i,1-2*eta/(1-xi)];
J1[s_] := Sum[(2*m+1)*Subscript[J,1,m]*LegendreP[m,2*s-1],{m,0,p}];
J3[s_] := Sum[(2*m+1)*Subscript[J,3,m]*LegendreP[m,2*s-1],{m,0,p}];
J2[s_] := Sum[(2*m+1)*Subscript[J,2,m]*LegendreP[m,2*s-1],{m,0,p}];
U1[s_] := Sum[Subscript[U,1,m]*s^m,{m,0,p}];
U3[s_] := Sum[Subscript[U,3,m]*s^m,{m,0,p}];
gamma = Table[1,{p+1}];
C2[m_,l_] := -theta*Integrate[J3[s]*phi[0,1-s,m,l],{s,0,1}] -
             (1-theta)*Integrate[J1[s]*phi[s,0,m,l],{s,0,1}];
C1[m_, l_] := Integrate[J2[s]*phi[1-s,s,m,l],{s,0,1}];
```

```
f2 = Simplify[Sum[Sum[(2*m+2)*(2*l+1)*gamma[[m+1]]*C2[m,l]*
        Sum[(omega*h)^(-n-1)*D[phi[theta*zeta-sigma,(1-theta)*zeta
        +sigma,m,l],{zeta,n}],{n,0,p}],{l,0,m}],{m, 0, p}]];
S2 = Simplify[Solve[Union[
        Table[Integrate[U3[s]*LegendreP[m,2*s-1],{s, 0, 1}] ==
              Integrate[(J3[s]+f2/.{zeta->1-s,sigma->theta*(1-s)})
              *LegendreP[m,2*s-1],{s,0,1}],{m,0,p}],
        Table[Integrate[U1[s]*LegendreP[m,2*s-1],{s,0,1}] ==
              Integrate[(J1[s]+f2/.{zeta->s,sigma->(theta-1)*s})
              *LegendreP[m,2*s-1],{s,0,1}],{m,0,p}]],
        Union[Table[Subscript[J,1,m],{m,0,p}],
              Table[Subscript[J,3,m],{m,0,p}]]][[1]]];
F2 = Factor[f2/.S2];

f1 = Simplify[Sum[Sum[(2*m+2)*(2*l+1)*gamma[[m+1]]*C1[m,l]*
      Sum[(-omega*h)^(-n-1)*D[phi[theta*zeta-sigma,(1-theta)*zeta
      +sigma,m,l],{zeta,n}],{n,0,1}],{l,0,m}],{m,0,1}]];
S1 = Factor[Solve[
         Table[Integrate[(F2/.{zeta->1,sigma->theta-s})*
               LegendreP[m,2*s-1],{s,0,1}] ==
               Integrate[(J2[s]+f1/.{zeta->1,sigma->theta-1+s})*
               LegendreP[m,2*s-1],{s,0,1}],{m,0,p}],
         Table[Subscript[J,2,m],{m,0,p}]][[1]]];
F1 = Factor[f1/.S1];

M = Factor[CoefficientArrays[Union[
        Table[Integrate[U3[s]*LegendreP[m,2*s-1],{s,0,1}] -
              lambda*Integrate[(F1/.{zeta->s,sigma->theta*s})*
              LegendreP[m,2*s-1],{s,0,1}],{m,0,p}],
        Table[Integrate[U1[s]*LegendreP[m,2*s-1],{s,0,1}] -
              mu*Integrate[(F1/.{zeta->1-s,sigma->(theta-1)*(1-s)})*
              LegendreP[m,2*s-1],{s,0,1}],{m,0,p}]],
        Union[Table[Subscript[U,1,m],{m,0,p}],
              Table[Subscript[U,3,m],{m,0,p}]]][[2]]];

Cp = Numerator[Factor[Det[M]]];
```

References

[1] M. Abramowitz and I.A. Stegun, editors. *Handbook of Mathematical Functions.* Dover, New York, 1965.

[2] S. Adjerid and M. Baccouch. The discontinuous Galerkin method for two-dimentional hyperbolic problems. I: Superconvergence error analysis. *SIAM Journal on Scientific Computing*, 33:75–113, 2007.

[3] S. Adjerid and M. Baccouch. The discontinuous Galerkin method for two-dimentional hyperbolic problems. II: A posteriori error estimation. *SIAM Journal on Scientific Computing*, 38:15–49, 2009.

[4] S. Adjerid, K. Devine, J.E. Flaherty, and L. Krivodonova. A posteriori error estimation for discontinuous Galerkin solutions of hyperbolic problems. *Computer Methods in Applied Mechanics and Engineering*, 191:1097–1112, 2002.

[5] S. Adjerid and T.C. Massey. Superconvergence of discontinuous finite element solutions for nonlinear hyperbolic problems. *Computer methods in applied mechanics and engineering*, 191:3331–3346, 2006.

[6] M. Ainsworth. Dispersive and dissipative behaviour of high order discontinuous Galerkin finite element methods. *Journal of Computational Physics*, 198:106–130, 2004.

[7] M. Ainsworth and J.T. Oden. *A posteriori Error Estimation in Finite Element Analysis.* Computational and Applied Mathematics. Wiley-Interscience, 2000.

[8] G. A. Baker and P. R. Graves-Morris. *Padé Approximants.* Addison-Wesley, Reading, Mass.; Don Mills, Ont., 1981.

[9] T.J. Barth and D.C. Jespersen. The design and application of upwind schemes on unstructured meshes. In *27th Aerospace Sciences Meeting*, AIAA 89-0366, Reno, Nevada, 1989.

[10] F. Bassi, A. Crivellini, S. Rebay, and M. Savini. Discontinuous Galerkin solution of the Reynolds-averaged Navier–Stokes and k–ω turbulence model equations. *Computers & Fluids*, 34(4):507–540, 2005.

[11] F. Bassi and S. Rebay. Accurate 2D Euler computations by means of a high order discontinuous finite element method. In *XIV International conference on numerical Methods in Fluid Dynamics*, volume 453 of *Lecture Notes in Physics*, pages 234–240. Springer, 1994.

[12] F. Bassi and S. Rebay. A high-order accurate discontinuous finite element method for the numerical solution of the compressible Navier-Stokes equations. *Journal of Computational Physics*, 131:267–279, 1997.

[13] F. Bassi and S. Rebay. High-order accurate discontinuous finite element solution of the 2D Euler equations. *Journal of Computational Physics*, 138:251–285, 1997.

[14] F. Bassi and S. Rebay. A high order discontinuous Galerkin method for compressible turbulent flows. In *Discontinuous Galerkin Methods*, pages 77–88. Springer, 2000.

[15] 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.

[16] C.E. Baumann and J.T. Oden. A discontinuous *hp* finite element method for the Euler and Navier–Stokes equations. *International Journal for Numerical Methods in Fluids*, 31(1):79–95, 1999.

[17] R. Biswas, K. Devine, and J.E. Flaherty. Parallel adaptive finite element methods for conservation laws. *Applied Numerical Mathematics*, 14:255–284, 1994.

[18] J.C. Butcher. *The numerical analysis of ordinary differential equations: Runge-Kutta and general linear methods.* Wiley-Interscience, 1987.

[19] N. Chalmers, L. Krivodonova, and R. Qin. Relaxing the CFL number of the discontinuous Galerkin method. *SIAM Journal on Scientific Computing*, 36(4):A2047–A2075, 2014.

[20] Y. Cheng and C.-W. Shu. Superconvergence and time evolution of discontinuous Galerkin finite element solutions. *Journal of Computational Physics*, v227:9612–9627, 2008.

[21] Y. Cheng and C.-W. Shu. Superconvergence of discontinuous Galerkin and local discontinuous Galerkin schemes for linear hyperbolic and convection-diffusion equations in one space dimension. *SIAM Journal on Numerical Analysis*, 47(6):4044–4072, 2010.

[22] B. Cockburn. Discontinuous Galerkin methods for convection-dominated problems. In *High-order methods for computational physics*, pages 69–224. Springer, 1999.

[23] B. Cockburn, S. Hou, and C.-W. Shu. The Runge-Kutta local projection discontinuous Galerkin finite element method for the conservation laws IV: The multidimensional case. *Mathematics of Computation*, 54:545–581, 1990.

[24] B. Cockburn, G. Kanschat, and D. Schötzau. The local discontinuous Galerkin method for the Oseen equations. *Mathematics of Computation*, 73(246):569–593, 2004.

[25] B. Cockburn, G. Kanschat, D. Schötzau, and C. Schwab. Local discontinuous Galerkin methods for the Stokes system. *SIAM Journal on Numerical Analysis*, 40(1):319–343, 2002.

[26] B. Cockburn, G.E. Karniadakis, and C.-W. Shu. The development of discontinuous Galerkin methods. In *Discontinuous Galerkin Methods*, pages 3–50. Springer, 2000.

[27] B. Cockburn, S.Y. Lin, and C.-W. Shu. TVB Runge-Kutta local projection discontinuous Galerkin methods for scalar conservation laws III: One dimensional systems. *Journal of Computational Physics*, 84:90–113, 1989.

[28] B. Cockburn, M. Luskin, C.-W. Shu, and E. Süli. Enhanced accuracy by post-processing for finite element methods for hyperbolic equations. *Math. Comp.*, 72(242):577–606 (electronic), 2003.

[29] B. Cockburn and C.-W. Shu. TVB Runge-Kutta local projection discontinuous Galerkin methods for scalar conservation laws II: General framework. *Mathematics of Computation*, 52:411–435, 1989.

[30] B. Cockburn and C.-W. Shu. The Runge-Kutte local projection P^1 discontinuous Galerkin method for scalar conservation laws. *RAIRO Model. Math. Anal. Numer.*, 25:337–361, 1991.

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

[32] A. Dervieux, B. van Leer, J. Periaux, and A. Rizzi, editors. *Numerical simulation of compressible Euler flows*, volume 26 of *Notes on Numerical fluid mechanics*, Braunschweig/Wiesbaden, 1989. Friedr. Vieweg & Sohn.

[33] M. Dubiner. Spectral methods on triangles and other domains. *Journal of Scientific Computing*, 6(4):345–390, 1991.

[34] D.A. Dunavant. High degree efficient symmetrical Gaussian quadrature rules for the triangle. *International Journal for Numerical Methods in Engineering*, 21(6):1129–1148, 1985.

[35] O. Friedrich. Weighted essentially non-oscillatory schemes for the interpolation of mean values on unstructured grids. *Journal of Computational Physics*, 144(1):194–212, 1998.

[36] M. Fuhry, A. Giuliani, and L. Krivodonova. Discontinuous Galerkin methods on graphics processing units for nonlinear hyperbolic conservation laws. *International Journal for Numerical Methods in Fluids*, 76(12):982–1003, 2014.

[37] P.A. Gremaud and J.V. Matthews. On the computation of steady hopper flows: I. stress determination for coulomb materials. *Journal of Computational Physics*, 166(1):63–83, 2001.

[38] W. Guo, X. Zhong, and J.-M. Qiu. Superconvergence of discontinuous Galerkin and local discontinuous Galerkin methods: Eigen-structure analysis based on Fourier approach. *Journal of Computational Physics*, 235:458–485, 2013.

[39] A. Harten, B. Engquist, S. Osher, and S. Chakravarthy. Uniformly high-order accurate essentially nonoscillatory schemes III. *Journal of Computational Physics*, 71:231–303, 1987.

[40] A. Harten and S. Osher. Uniformly high-order accurate nonoscillatory schemes.I. *SIAM Journal on Numerical Analysis*, 24:279–309, 1987.

[41] J.S. Hesthaven and T. Warburton. Nodal high-order methods on unstructured grids:: I. time-domain solution of Maxwell's equations. *Journal of Computational Physics*, 181(1):186–221, 2002.

[42] C. Hu and C.-W. Shu. Weighted essentially non-oscillatory schemes on triangular meshes. *Journal of Computational Physics*, 150(1):97–127, 1999.

[43] F. Q. Hu and H. L. Atkins. Eigensolution analysis of the discontinuous Galerkin method with nonuniform grids. *Journal of Computational Physics*, 182:516–545, 2002.

[44] G.-S. Jiang and C.-W. Shu. Efficient implementation of weighted ENO schemes. *Journal of Computational Physics*, 126:202–228, 1996.

[45] A. Klöckner, T. Warburton, J. Bridge, and J.S. Hesthaven. Nodal discontinuous Galerkin methods on graphics processors. *Journal of Computational Physics*, 228(21):7863–7882, 2009.

[46] L. Krivodonova. Moment limiters for discontinuous Galerkin methods. *Journal of Computational Physics*, 226:276–296, 2007.

[47] L. Krivodonova and J.E. Flaherty. Error estimation for discontinuous Galerkin solutions of multidimensional hyperbolic problems. *Advances in Computational Mathematics*, 19:57–71, 2003.

[48] L. Krivodonova and R. Qin. An analysis of the spectrum of the discontinuous Galerkin method. *Applied Numerical Mathematics*, 64:1–18, 2013.

[49] L. Krivodonova and R. Qin. An analysis of the spectrum of the discontinuous Galerkin method II: Nonuniform grids. *Applied Numerical Mathematics*, 71:41–62, 2013.

[50] E.J. Kubatko, C. Dawson, and J.J. Westerink. Time step restrictions for Runge–Kutta discontinuous Galerkin methods on triangular grids. *Journal of Computational Physics*, 227(23):9697–9710, 2008.

[51] P. Lax. *Hyperbolic systems of conservation laws and the mathematical theory of shock waves*, volume 11. SIAM, 1973.

[52] P. Lesaint and P. Raviart. On a finite element method for solving the neutron transport equation. In C. de Boor, editor, *Mathematical Aspects of Finite Elements in Partial Differential Equations*, pages 89–145, New York, 1974. Academic Press.

[53] R.J. LeVeque. *Numerical methods for conservation laws*, volume 132. Springer, 1992.

[54] X.-D. Liu and S. Osher. Nonoscillatory high order accurate self-similar maximum principle satisfying shock capturing schemes I. *SINUM*, 33:760–779, 1996.

[55] I. Lomtev and G.E. Karniadakis. A discontinuous Galerkin method for the Navier-Stokes equations. *Intnational Journal on Numerical Methods in Fluids*, 29:587–603, 1999.

[56] H. Luo, J. Baum, and R. Lohner. On the computation of steady-state compressible flows using a discontinuous Galerkin method. *International Journal for Numerical Methods in Engineering*, 73(5):597–623, 2008.

[57] J. Niegemann, R. Diehl, and K. Busch. Efficient low-storage Runge–Kutta schemes with optimized stability regions. *Journal of Computational Physics*, 231(2):364 – 372, 2012.

[58] M. Parsani, D. I. Ketcheson, and W. Deconinck. Optimized explicit Runge–Kutta schemes for the spectral difference method applied to wave propagation problems. *SIAM Journal on Scientific Computing*, 35(2):A957–A986, 2013.

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

[60] J. Ryan, C.-W. Shu, and H. Atkins. Extension of a postprocessing technique for the discontinuous Galerkin method for hyperbolic equations with application to an aeroacoustic problem. *SIAM Journal on Scientific Computing*, 26(3):821–843, 2005.

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

[62] C.-W. Shu and S. Osher. Efficient implementation of essentially non-oscillatory shock-capturing schemes. *Journal of Computational Physics*, 77:439–471, 1988.

[63] C.-W. Shu and S. Osher. Efficient implementation of essentially non-oscillatory shock-capturing schemes, II. *Journal of Computational Physics*, 83:32–78, 1989.

[64] E. Toro. *Riemann solvers and Numerical Methods for fluid dynamics*. Springer, 1999.

[65] T. Toulorge and W. Desmet. Optimal Runge-Kutta schemes for discontinuous Galerkin space discretizations applied to wave propagation problems. *Journal of Computational Physics*, 231(4):2067 – 2091, 2012.

[66] L.B. Wahlbin. *Superconvergence in Galerkin finite element methods.* Springer, 1995.

[67] T. Warburton and T. Hagstrom. Taming the CFL number for discontinuous Galerkin methods on structured meshes. *SIAM Journal on Numerical Analysis*, 46(6):3151–3180, 2008.

[68] T. Warburton and G.E. Karniadakis. A discontinuous Galerkin method for the viscous MHD equations. *Journal of computational Physics*, 152(2):608–641, 1999.

[69] T. Warburton, I. Lomtev, R.M. Kirby, and G.E. Karniadakis. A discontinuous Galerkin method for the Navier-Stokes equations on hybrid grids. *Center for Fluid Mechanics*, pages 97–14, 1998.

[70] P. Woodward and P. Colella. The numerical simulation of two-dimensional fluid flow with strong shocks. *Journal of Computational Physics*, 54:115–173, 1984.

[71] J. Yan and C.-W. Shu. A local discontinuous Galerkin method for KdV type equations. *SIAM Journal on Numerical Analysis*, 40(2):769–791, 2002.

[72] Y. Yang and C.-W. Shu. Analysis of optimal superconvergence of discontinuous Galerkin method for linear hyperbolic equations. *SIAM Journal on Numerical Analysis*, 50(6):3110–3133, 2012.

[73] X. Zhong and C.-W. Shu. Numerical resolution of discontinuous Galerkin methods for time dependent wave equations. *Computer Methods in Applied Mechanics and Engineering*, 200(41):2814–2827, 2011.