

**HIGH ORDER SCHEMES FOR TRANSPORT PROBLEMS:
SEMI-LAGRANGIAN SCHEMES WITH APPLICATIONS TO
PLASMA PHYSICS AND ATMOSPHERIC SCIENCES, AND
SUPERCONVERGENCE**

A Dissertation
Presented to
the Faculty of the Department of Mathematics
University of Houston

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy

By
Wei Guo
August 2014

**HIGH ORDER SCHEMES FOR TRANSPORT PROBLEMS:
SEMI-LAGRANGIAN SCHEMES WITH APPLICATIONS TO
PLASMA PHYSICS AND ATMOSPHERIC SCIENCES, AND
SUPERCONVERGENCE**

Wei Guo

APPROVED:

Dr. Jingmei Qiu, Chairman

Dr. Tsorng-Whay Pan

Dr. Jiwen He

Dr. Guoning Chen,
Department of Computer Science,
University of Houston

Dean, College of Natural Sciences and Mathematics

Acknowledgements

This dissertation work would not have been possible without support from lots of people. I am most grateful to my dissertation advisor, Dr. Jingmei Qiu, for her guidance and support throughout the time of my dissertation research. Dr. Qiu is always patient with me and guiding me through this endeavor. Her deep insight, wide knowledge, and accurate foresight made the development of this work significantly easier. Another special thanks goes to my committee members, Dr. Tsorng-Whay Pan, Dr. Jiwen He, and Dr. Guoning Chen, for their valuable feedback and helpful comments to improve this work. I also would like to express my warm thanks to many collaborators I have had the pleasure of working with for the past few years; in particular, Dr. Andrew Christlieb and Dr. Xinghui Zhong at the Michigan State University, Dr. Maureen Morton at the Stark State College, Dr. Ram Nair at NCAR, and Dr. Jianxian Qiu at the Xiamen University. I look forward to working with them in the future. In addition, a special mention goes to fellow group members, Dr. Tao Xiong, Mr. Pei Yang, and Mr. Tan Ren for many insightful discussions and enjoyable experience both inside and outside the office. Most importantly, I would like to thank my parents, Fandi Guo and Jianmin Zhou, for their consistent encouragement and valuable teachings since I was born.

**HIGH ORDER SCHEMES FOR TRANSPORT PROBLEMS:
SEMI-LAGRANGIAN SCHEMES WITH APPLICATIONS TO
PLASMA PHYSICS AND ATMOSPHERIC SCIENCES, AND
SUPERCONVERGENCE**

An Abstract of a Dissertation
Presented to
the Faculty of the Department of Mathematics
University of Houston

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy

By
Wei Guo
August 2014

Abstract

High order schemes for transport gain lots of popularity in scientific computing community due to their superior properties, such as high efficiency and high resolution. In this dissertation, we systematically investigate the efficient high order numerical schemes for solving transport equations.

In the first part, we develop and implement a class of high order semi-Lagrangian (SL) schemes for linear transport equations, which are further applied to the Vlasov simulations and global transport modeling. Compared with Eulerian type schemes, the SL schemes can take arbitrary large time steps without stability issue, leading to improved computational efficiency. For solving the Vlasov-Poisson (VP) system, a high order hybrid methodology, which couples discontinuous Galerkin (DG) schemes and finite difference weighted essentially non-oscillatory (WENO) schemes, is proposed in the Strang splitting framework. The hybrid scheme can take advantage of the numerical ingredients in order to attain good numerical performance. Furthermore, an integral deferred correction method is used to correct the splitting error. Then, the proposed SL method for linear transport is extended for solving spherical transport equations. In particular, a SLDG scheme is formulated on the cubed-sphere geometry. A collection of benchmark numerical tests demonstrate reliability and efficiency of the scheme.

In the second part, via classic Fourier approach, we systematically study superconvergence properties of DG schemes. Superconvergence analysis is important for understanding long time behaviors of DG errors. Based on the eigen-structure analysis of the amplification matrix, we can explain that DG errors will not significantly

grow for long time simulations. Specifically, the part of the error that grows linearly in time comes from the dispersion and dissipation errors of the physically relevant eigenvalue. Such an error is superconvergent of order $2k+1$, compared with the standard $(k+1)^{th}$ order accuracy. Furthermore, we investigate superconvergence for DG schemes coupled with a Lax-Wendroff (LW) type time discretization. The numerical evidence shows that the original LWDG scheme, does not numerically exhibit superconvergence. In order to restore such properties, we propose a new LWDG scheme by borrowing techniques from a local DG (LDG) scheme. Numerical experiments are presented to verify superconvergence of the newly proposed scheme.

Contents

1	Introduction	1
1.1	Transport Modeling	2
1.1.1	Vlasov Model for Plasma Physics	2
1.1.2	Global Multi-tracer Transport in Atmosphere Sciences	6
1.2	Numerical Schemes for Transport: Review	9
1.2.1	Three Types of Numerical Approaches	9
1.2.2	Operator Splitting Semi-Lagrangian Scheme	10
1.2.3	High Order Numerical Schemes	11
1.3	Numerical Property: Superconvergence of DG Schemes	12
1.4	Scope of the Work	15
2	DG and LDG Schemes	16
2.1	Algorithm Formulation of DG Schemes	17
2.2	Algorithm Formulation of LDG schemes	20
2.3	Time Discretization: Runge-Kutta Methods	22

3	High Order Semi-Lagrangian Schemes	25
3.1	High Order Semi-Lagrangian DG Schemes	26
3.1.1	First Formulation	27
3.1.2	Second Formulation	29
3.1.3	The Bound-Preserving (BP) Filter	33
3.2	High Order Semi-Lagrangian WENO Schemes	34
3.3	Extension to Multi-dimensional Problems via Dimensional Splitting .	40
3.4	Applications to Vlasov Simulations	42
3.4.1	A Hybrid Methodology	43
3.4.2	Integral Deferred Correction in Correcting Splitting Error . . .	79
3.5	Applications to Global Transport	113
3.5.1	Cubed-Sphere Geometry	114
3.5.2	A Semi-Lagrangian DG Scheme on the Cubed Sphere	115
4	Superconvergence of Discontinuous Galerkin Schemes	137
4.1	Fourier Analysis for DG and LDG Schemes	138
4.1.1	Fourier Analysis for DG Schemes	138
4.1.2	Fourier Analysis for LDG schemes	140
4.2	Eigen-Structures of G: Error Estimate	141
4.2.1	DG Schemes	141
4.2.2	LDG Schemes	155
4.2.3	Supraconvergence of DG and LDG Schemes	164
4.2.4	Fully Discrete RKDG Schemes	165
4.3	A New LWDG Scheme with Superconvergence	195
4.3.1	Lax-Wendroff Type Time Discretization	196
4.3.2	Numerical Evidences Regarding Superconvergence	198
4.3.3	A New LWDG Scheme	203

CONTENTS

4.3.4	Fourier Analysis for the New LWDG Schemes	211
5	Conclusions	232
	Bibliography	236

CHAPTER 1

Introduction

The work of this dissertation consists of two parts. First, a class of high order semi-Lagrangian (SL) schemes are formulated for solving transport problems. Second, superconvergence properties of discontinuous Galerkin (DG) methods are investigated in different settings. In this chapter, we briefly introduce the motivation as well as the background of the research work which provide the context of the novel work presented in subsequent chapters. Some state-of-the-art numerical approaches are reviewed and discussed in terms of their strength and weakness.

1.1 Transport Modeling

Transport problems are ubiquitous in the real world where many natural phenomena can be described by a set of transport equations. The hyperbolic conservation laws arising from one of the most important physics laws, namely conservation, such as mass, momentum, and energy, are known as nonlinear transport equations. Unfortunately, the analytical solutions of transport equations are not always available. On the other hand, in many applications, people seek to obtain approximate solutions in order to understand the behavior of transport equations. To this end, the numerical algorithm comes in with the aid of computers or supercomputers, and plays an increasingly important role in academic research and real applications. In this dissertation, we are interested in two types of transport problems: One is the nonlinear Vlasov-Poisson (VP) system in plasma physics. The other is the global multi-tracer transport model in atmosphere science.

1.1.1 Vlasov Model for Plasma Physics

Plasma is known as the fourth state of fundamental matter in the universe after solid, liquid, and gas. In particular, it describes a typical state at high temperature and high pressure, in which the electrons are completely dissociated from their nucleus. Generally, there are two classes of descriptions of a plasma including the fluid models and the kinetic models. The fluid model tracks the macroscopic quantities such as density and average velocity at each spatial location. The well known ideal magnetohydrodynamics (MHD) equations belong to this category. One obvious advantage of

fluid models is the low dimensionality (3-D in space at most plus time). However, it is required that the plasma is sufficiently close to the thermodynamical equilibrium in phase space. In other words, under the assumption of ample collisions of particles such that the mean free path is much smaller than the characteristic length scale of the plasma, the fluid models are valid. Kinetic models in describing the physics with meso-scale lie in between the microscopic particles dynamics by Newton's laws of motion and macroscopic fluid models. The thermodynamical equilibrium in phase space is not required for kinetic models at the price of high dimensionality. In fact, a kinetic model provides a complete phase-space description of plasma by introducing velocities as new independent variables. Therefore, the computational cost of the kinetic model (3-D in space and 3-D in velocity plus time) is generally more expensive than that of the fluid model. On the other hand, if a collisionless plasma is considered, a kinetic model is necessary to describe the dynamics, since the plasma may significantly deviate from the thermodynamical equilibrium.

Vlasov-Poisson System

A simple non-relativistic and collisionless model in describing a plasma with a single species is given by the VP system,

$$f_t + \mathbf{v} \cdot \nabla_{\mathbf{x}} f + \mathbf{E}(t, \mathbf{x}) \cdot \nabla_{\mathbf{v}} f = 0, \quad (1.1)$$

$$\mathbf{E}(t, \mathbf{x}) = -\nabla_{\mathbf{x}} \phi, \quad -\Delta_{\mathbf{x}} \phi = -1 + \rho(t, \mathbf{x}), \quad (1.2)$$

where $f(t, \mathbf{x}, \mathbf{v}) \geq 0$ describes the probability of finding a particle with velocity $\mathbf{v} \in \mathcal{R}^3$ at position $\mathbf{x} \in \mathcal{R}^3$ at time t , \mathbf{E} is the self-consistent electrostatic field,

ϕ is the self-consistent electrostatic potential, and $\rho(t, \mathbf{x}) = \int_{\mathcal{R}^3} f(t, \mathbf{x}, \mathbf{v}) d\mathbf{v}$ is the electron charge density. Here a uniform distribution of ions is assumed to form a neutralizing background. All physical constants in (1.1) have been normalized to one. Note that, the VP system (1.1)-(1.2) is a simplified model of the full Vlasov-Maxwell (VM) system under the assumption of negligible magnetic field.

Moments and Conservation Properties

The plasma moments are a set of the macroscopic quantities defined by the distribution function f via integrations in velocity space, after multiplication of specific functions of \mathbf{v} . Note that the plasma moments are physically measurable and often provide the information of interest. The first three moments are defined as

$$\begin{pmatrix} \rho \\ \mathbf{j} \\ W \end{pmatrix} = \int_{\mathcal{R}^3} f \begin{pmatrix} 1 \\ \mathbf{v} \\ \frac{1}{2}|\mathbf{v}|^2 \end{pmatrix} d\mathbf{v}, \quad (1.3)$$

which are commonly known as mass density, momentum density and kinetic energy density. Average velocity \mathbf{u} is defined by

$$\mathbf{u} = \frac{\mathbf{j}}{\rho}. \quad (1.4)$$

The VP system conserves a bunch of physics quantities in the evolution, such as

total mass, momentum and total energy which are defined by

$$\text{total mass} = \int_{\mathcal{R}^3} \int_{\mathcal{R}^3} f d\mathbf{x}d\mathbf{v}, \quad (1.5)$$

$$\text{momentum} = \int_{\mathcal{R}^3} \int_{\mathcal{R}^3} f \mathbf{v} d\mathbf{x}d\mathbf{v}, \quad (1.6)$$

$$\text{total energy} = \frac{1}{2} \int_{\mathcal{R}^3} \int_{\mathcal{R}^3} f |\mathbf{v}|^2 d\mathbf{x}d\mathbf{v} + \frac{1}{2} \int_{\mathcal{R}^3} |\mathbf{E}|^2 d\mathbf{x}. \quad (1.7)$$

These important conservation properties can be derived directly from the VP system (1.1)-(1.2) via simple algebraic computation. Moreover, any functional in the form of $\int_{\mathcal{R}^3} \int_{\mathcal{R}^3} G(f) d\mathbf{x}d\mathbf{v}$ remains constant in time, and all of these invariants are called Casimir invariants, see e.g., [35, 72]. Here, we want to point out one of the most invariants of this type, namely total entropy $S = - \int_{\mathcal{R}^3} \int_{\mathcal{R}^3} f \log(f) d\mathbf{x}d\mathbf{v}$. The fact that the total entropy stays constant in time implies the information of the whole system is not lost during the evolution, and hence the VP system is time reversible. Note that it is not the case for the Boltzmann equation, for which the total entropy grows in time due to the well known H-theorem, see e.g., [35].

In the numerical level, it is very difficult to conserve all the physical invariants mentioned above. However, it provides a good and practical criterion to assess the performance of numerical schemes by tracking the time evolution of these quantities. A ‘good’ numerical scheme is supposed to ‘approximately’ conserve the quantities in a ‘satisfactory’ manner.

1.1.2 Global Multi-tracer Transport in Atmosphere Sciences

During the past few decades, the atmospheric models have provided substantial insight into understanding of the behaviors of atmosphere, and predicting the weather as well as the trend of climate. The multi-tracer transport process plays a central role in the global atmospheric modeling. A tracer, such as humidity, mixed ratio, or a chemical species, is a typical entity moving along with the air's motion. In the multi-tracer transport model, $\mathcal{O}(100)$ tracer species need to be tracked. In other words, the associated transport equations must be solved $\mathcal{O}(100)$ times at the same time. Consequently, the total computational expense is always dominated by the tracer transport schemes. Moreover, the large scale global transport modeling also imposes great challenges on the numerical simulations. Therefore, efficient and reliable transport schemes on the sphere are extremely desirable.

Cubed-Sphere Geometry

Traditionally, the numerical grid based on the latitude-longitude (Lat-Lon) or polar coordinates is considered a natural choice for transport simulations on the sphere. However, the Lat-Lon grid tends to merge towards the north and south poles, and such polar singularity leads to computational inefficiency due to the highly non-uniform nature of the grid. Numerous efforts are made to seek other spherical geometry with weaker singularities for the numerical simulations, such as icosahedral grid [71], cubed-sphere grid [95], Ying-Yang grid [79], among many others. In this work, we consider the cubed-sphere geometry.

The cubed-sphere geometry was first introduced by Sadourny [100] and reintroduced by Ronchi et al. [95] with some improved features such as the orthogonality. The cubed-sphere grid is constructed via a bijective mapping from a sphere to identical six faces (patches) of a cube, see Figure 1.1. Note that, unlike the Lat-Lon grid, the cubed-sphere counterpart is free of polar singularities. Instead, a weaker singularity on the internal edges of the cube is generated. Generally, there are two types of the cubed-sphere grids [91] constructed via the conformal projection and the gnomonic (equi-angular central) projection, respectively. In this work, we adopt the cubed-sphere geometry based on the gnomonic projection, which offers a more isotropic spherical grid. In the past two decades, the cubed-sphere-based simulations attract lots of interest in the global atmosphere research community [76, 62, 96], since it is known to be well-suited for the ‘local’ numerical methods including finite volume schemes [96, 62, 50], finite element DG schemes [76, 123], and spectral element schemes [109].

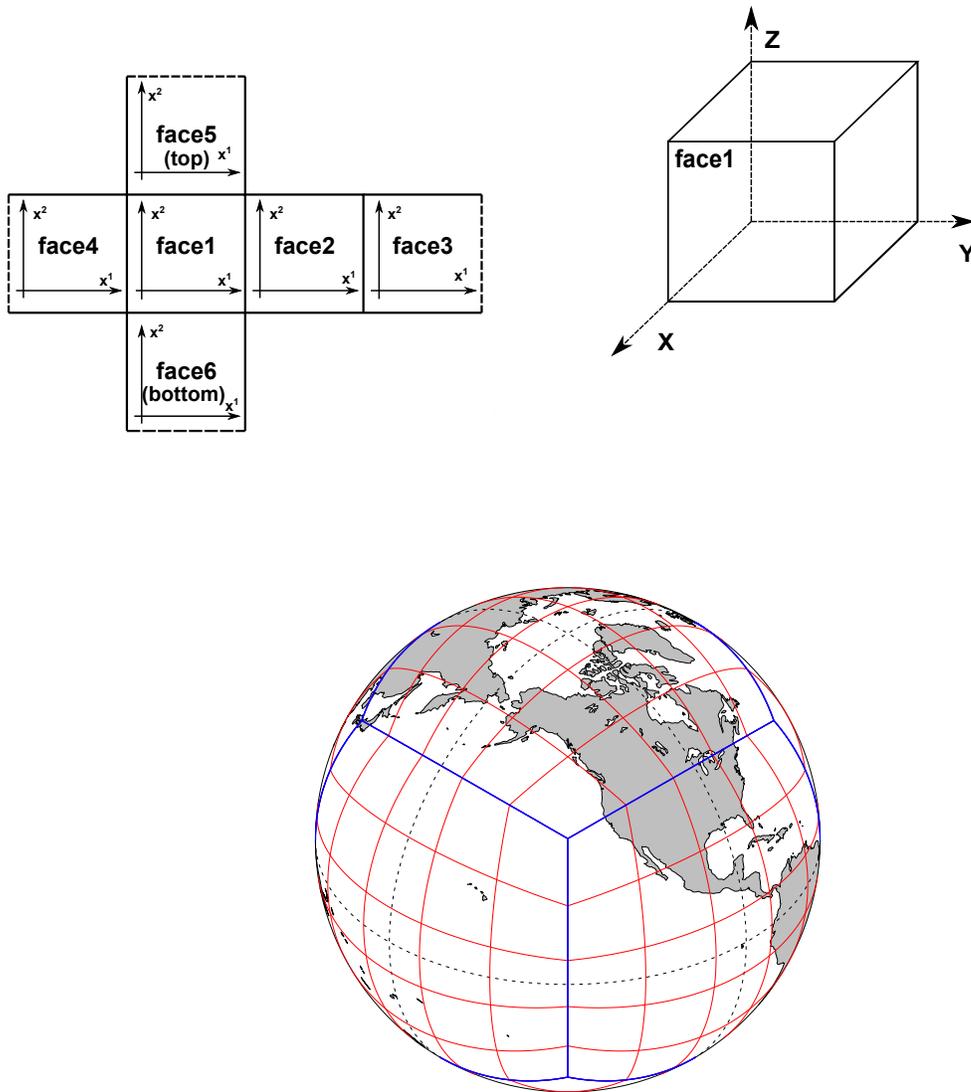


Figure 1.1: Schematic for the cubed-sphere geometry, the relative positions of six cube faces (from face1 to face6) and their local connectivity.

1.2 Numerical Schemes for Transport: Review

In this section, we review the numerical approaches for transport equations as well as their applications to plasma physics and global atmospheric modeling.

1.2.1 Three Types of Numerical Approaches

Popular numerical approaches in solving transport equations can be classified into three types: Eulerian, Lagrangian, and SL. The Lagrangian-type particle methods, such as the very famous particle-in-cell (PIC) scheme for the Vlasov simulations, see, e.g., [8, 54, 111], evolve the solution by following the trajectories of some sampled macro-particles, while the Eulerian approach, e.g., [104, 28, 66, 125, 14], evolves the state variable according to the underlying PDEs on a fixed numerical grid. The SL approach is considered a mixed approach of Lagrangian and Eulerian in an attempt to combine the advantages of both. In particular, the SL approach has a fixed numerical grid; however, over each time step the state variable is evolved by propagating information along characteristics. The Eulerian and the SL approaches can be conveniently designed to be of very high order accuracy, an advantage when compared with the Lagrangian approach. On the other hand, because of the evolution mechanism, the SL method does not suffer the Courant-Friedrichs-Lewy (CFL) [33] time step restriction as in an explicit Eulerian approach, allowing for extra large time step evolution, and therefore less computational effort. We focus on the SL-type schemes in this work.

1.2.2 Operator Splitting Semi-Lagrangian Scheme

The SL approach is very popular in kinetic [6, 7, 56, 38, 110] and global atmospheric [106, 46, 90, 93, 75, 62] simulations. In kinetic simulations of plasma, a very popular approach is the Strang splitting SL method, first proposed by Cheng and Knorr in [13]. In particular, the Vlasov equation (1.1) is split into two lower dimensional transport equations:

$$f_t + \mathbf{v} \cdot \nabla_{\mathbf{x}} f = 0 \quad \text{spatial advection,} \quad (1.8)$$

$$f_t + \mathbf{E}(t, x) \cdot \nabla_{\mathbf{v}} f = 0 \quad \text{velocity ac/deceleration.} \quad (1.9)$$

The advantage of performing such a splitting is that the decoupled equations of spatial advection and velocity ac/deceleration respectively are linear and are much easier to evolve numerically. Consequently, numerical schemes with high order of spatial accuracy have been designed, and demonstrated numerically to be more efficient than lower order schemes in finite difference framework with different reconstruction/interpolation strategies, such as, the cubic spline interpolation [105], the cubic interpolated propagation [78], the weighted essentially non-oscillatory (WENO) interpolation [12, 84, 85]; in finite volume framework for the VP system [39] and for the guiding center Vlasov equation [34]; and in finite element DG framework [97, 87]. In the atmospheric simulation, the operator splitting-based SL scheme for transport is also one of prominent numerical approaches at present [68, 65], since the complicated regridding procedure in a non-splitting SL scheme, e.g., the conservative SL multi-tracer transport (CSLAM) scheme [62], is completely avoided. On the other

hand, despite great computational efficiency and convenience of the operator splitting SL approaches, the numerical error in time is dominated by the splitting error, which is relatively low order ($\mathcal{O}(\Delta t^2)$) [108]. It is therefore important to circumvent such low order splitting errors in the SL framework, in order to have numerical algorithms that attain higher orders of accuracy in both space and time.

1.2.3 High Order Numerical Schemes

Here, we would like to point out the advantages of using high order numerical schemes for transport simulations. Historically, the monotone scheme for solving transport equations was commonly used in real applications due to its remarkable features, such as the provable convergence property in very general settings. However, people later realized that the monotone scheme is first order accurate at most, and the numerical solution is smeared very quickly owing to excessive numerical viscosity, see e.g., [67]. Then, a class of high resolution schemes were developed in order to achieve improved numerical performance [66, 67]. This type of schemes features a famous stability property, namely total variation diminishing (TVD), i.e., the total variation of the numerical solution is always non-increasing when evolving the scheme. Meanwhile, the numerical viscosity is significantly reduced compared with the monotone schemes, and hence much more accurate and reliable results are provided by the high resolution schemes. Even today, high resolution schemes are considered prominent schemes in scientific computing community. In spite of the great success [66], high resolution schemes also suffer the relatively low order accuracy. In fact, the scheme is only first order accurate in the vicinity of extrema of the solution. A breakthrough of

developing uniformly high order accurate schemes with good stability properties was made by Harten et al. in [51], where the famous ENO scheme was proposed. Later, an improved version, WENO scheme was developed in [69, 60]. Note that ENO/WENO schemes are in the finite volume or finite difference framework, in which the high order accuracy is obtained by the adoption of wide stencils. On the other hand, in the finite element framework, DG schemes developed by Cockburn, Shu, and their collaborators are known to be high order accurate and nonlinearly stable owing to the cell entropy inequality [59]. The high order accuracy of DG is obtained by increasing the degrees of used approximation space. Nowadays, high order schemes gain lots of popularity in research and real applications, see, e.g., [24]. Compared with low order schemes, the high order ones have relatively low numerical dissipation and superior ability in resolving fine solution structures in long time integrations with relatively coarse numerical meshes. Therefore, when considering large scale numerical simulations, the high order numerical schemes are more desirable since convergence study with very fine mesh refinement for low order schemes is unaffordable.

1.3 Numerical Property: Superconvergence of DG Schemes

The DG and local DG (LDG) methods are a class of finite element methods, designed for solving hyperbolic and parabolic problems, among many others [32]. These methods use piecewise polynomial spaces of degree k that could be discontinuous across

cell boundaries as solution and test function spaces. These methods have the advantages of being compact and flexible for unstructured meshes, and being suitable for h-p adaptivity. Moreover, it has been rigorously proved that both DG and LDG methods are $(k+1)^{th}$ order accurate in the L^2 norm for linear problems with smooth enough solutions [32] for 1-D cases. For general meshes, DG solutions are proved to be $(k + \frac{1}{2})^{th}$ order accurate for linear hyperbolic problems [81]. These methods also have some inherent dissipation mechanism for L^2 stability of nonlinear problems in the semi-discrete sense, see e.g., [59, 114] and references therein.

In this work, our focus is on superconvergence properties of DG and LDG solutions. Superconvergence properties of DG and LDG methods for hyperbolic and parabolic problems have been intensively investigated in the past. Lowrie et al. [70] discovered that when the piecewise polynomial space of degree k is used, “a component of error” of the DG method converges with order $2k + 1$ in the L^2 norm. It is shown in [27, 57] that the DG and LDG solutions converge with order $2k + 1$ in terms of the negative-order norm if the exact solution is globally smooth. Based on the negative-order norm estimate, the DG and LDG solutions on translation invariant grids can be post-processed via a kernel convolution with B-spline functions. The post-processed solution, which is proved to converge with order $2k + 1$ in the L^2 norm, is much more accurate than the original numerical solution ($(k + 1)^{th}$ order), see [27, 99, 57]. Adjerid et al. in [1] analyzed the DG method in the setting of ordinary differential equations (ODEs) with a conclusion that the DG solution converges with order $k + 2$ at Radau points of each element, and with order $2k + 1$ at downwind points. In [2], Adjerid et al. numerically investigated superconvergence of DG and

LDG schemes for convection-diffusion equations at Radau points. Cheng and Shu in [15, 16, 17] showed that the DG or LDG solution is closer to the Radau projection of the exact solution than the exact solution itself. As a result, the error of a DG or LDG solution will not grow over a long time period $\mathcal{O}(\frac{1}{\sqrt{\Delta x}})$. In [55, 3, 4, 101, 103], Fourier analysis has been adopted to investigate superconvergence properties of DG schemes in terms of dispersion and dissipation error of physically relevant eigenvalues, see e.g., [55, 3, 4]. Zhong and Shu [124] used the Fourier analysis and symbolic computation to show that the DG method is superconvergent at Radau and downwind points with order of $k + 2$ and $2k + 1$, respectively. In [117], the DG solutions are proved to converge at the optimal rate of $k + 2$ at Radau points under a general assumption of non-uniform meshes. More recently, Cao et al. constructed another special projection of the exact solution, which is even closer to the DG solution than the Radau projection ($(2k + 1)^{th}$ order) [11]. Because of these superconvergence properties, the method is considered to be very competitive in resolving waves propagation with long time integrations.

Different approaches have been adopted to analyze superconvergence properties of DG schemes, such as the negative-order norm estimate [27, 57], by considering the problem as an initial or boundary value problem [1, 55, 3, 4], by special decomposition of error and playing with test functions in the weak formulation [15, 117, 11], Fourier analysis [124, 101, 103, 116] etc. Fourier analysis has been known to be limited to linear problems with periodic boundary conditions and uniform mesh. However, it provides a sufficient condition for instability of ‘bad’ schemes [119] as well as a quantitative error estimate. It can be used as a guidance to study the numerical

properties in more general settings [124].

1.4 Scope of the Work

In this dissertation, we focus on two primary topics: (1) the development of a class of high order SL methods for the Vlasov simulations and global transport modeling; and (2) the theoretical study of superconvergence properties of DG schemes.

This dissertation is organized as follows. In Chapter 2 we review the formulations of DG and LDG schemes. In Chapter 3 we construct a class of high order SL methods for solving the VP system and global transport equations on the cubed sphere. Chapter 4 pursues theoretical analysis of superconvergence properties of DG schemes in different settings. Finally, conclusions and future work are presented in Chapter 5.

CHAPTER 2

DG and LDG Schemes

In this chapter, we review the finite element DG schemes and LDG schemes for solving hyperbolic conservation laws and parabolic equations. We only consider 1-D cases for simplicity. Both schemes are demonstrated to be effective, robust, and reliable in real applications. Moreover, they serve as the building blocks for the schemes developed in the subsequent chapters. Another purpose of this chapter is to introduce notations used throughout the dissertation.

Historically, the DG schemes was proposed by Reed and Hill in [92] for solving the neutral transport equations. Then, the DG schemes were generalized to hyperbolic problems, arising from many areas of science and engineering, especially in the field of

computational fluid dynamics, by Cockburn, Shu, and their collaborators through a series of papers [29, 28, 26, 23, 31]. A breakthrough of extending the DG schemes for PDEs with high order derivatives was made by Bassi and Rebay in the framework of the compressible Navier-Stokes equations, see [5]. Later, Cockburn and Shu proposed the LDG schemes which can deal with general convection-dominated systems [30, 32].

2.1 Algorithm Formulation of DG Schemes

In this section, we review the algorithm formulation of the DG schemes. Some implementation details are also provided.

Consider the following one-dimensional (1-D) hyperbolic conservation laws problem

$$u_t + f(u)_x = 0, \quad x \in [a, b], \quad t > 0, \quad (2.1)$$

with suitable initial and boundary conditions. $f(u)$ here is the flux function. Note that there are two different meanings of f throughout the dissertation. One is the flux function of a hyperbolic conservation law, the other is the distribution function of the VP system. The readers can easily figure out the meaning of f from the given context without any confusion. To define the DG schemes, we consider a partition of the computational domain $[a, b]$ into N cells as follows:

$$a = x_{\frac{1}{2}} < x_{\frac{3}{2}} < \cdots < x_{N+\frac{1}{2}} = b.$$

2.1. ALGORITHM FORMULATION OF DG SCHEMES

Denote the cell by $I_j = [x_{j-\frac{1}{2}}, x_{j+\frac{1}{2}}]$ and the cell center by $x_j = \frac{1}{2} (x_{j+\frac{1}{2}} + x_{j-\frac{1}{2}})$, for $j = 1, \dots, N$. The length of the cell is denoted by $\Delta x_j = x_{j+\frac{1}{2}} - x_{j-\frac{1}{2}}$ and the mesh size $\Delta x = \max_{1 \leq j \leq N} \Delta x_j$. Define the approximation space as

$$V_h^k = \{v : v|_{I_j} \in P^k(I_j); 1 \leq j \leq N\}, \quad (2.2)$$

where $P^k(I_j)$ denotes the set of polynomials of degree up to k on cell I_j . The semi-discrete DG method for solving (2.1) is defined as follows: find $u_h \in V_h^k$ such that $\forall v \in V_h^k$, we have

$$\int_{I_j} (u_h)_t v \, dx - \int_{I_j} f(u_h) v_x \, dx + \hat{f}_{j+\frac{1}{2}} v_{j+\frac{1}{2}}^- - \hat{f}_{j-\frac{1}{2}} v_{j-\frac{1}{2}}^+ = 0, \quad j = 1, \dots, N. \quad (2.3)$$

Here and below $u_h^+ = u_h(x^+)$, $u_h^- = u_h(x^-)$ denote the right and left limits of the function u_h at a spatial location x , respectively. The numerical flux $\hat{f} = \hat{f}(u_h^-, u_h^+)$ is defined at the cell interface and should satisfy the following requirements:

- Consistency: $\hat{f}(u, u) = f(u)$;
- Continuity: $\hat{f}(\cdot, \cdot)$ is Lipschitz continuous with respect to both arguments;
- Monotonicity: $\hat{f}(\cdot, \cdot)$ is non-decreasing with the first argument and non-increasing with the second argument.

Below, we list several commonly used monotone numerical fluxes.

- Godunov flux:

$$\hat{f}(a, b) = \begin{cases} \min_{a \leq u \leq b} f(u), & \text{if } a \leq b, \\ \max_{b \leq u \leq a} f(u), & \text{if } b < a; \end{cases}$$

- Engquist-Osher flux:

$$\hat{f}(a, b) = \int_0^a \max(f'(u), 0) du + \int_0^b \min(f'(u), 0) du + f(0);$$

- Lax-Friedrichs flux:

$$\hat{f}(a, b) = \frac{1}{2} (f(a) + f(b) - \alpha(b - a)),$$

where $\alpha = \max_u |f'(u)|$ with the maximum taken in the computational domain.

When implementing the DG scheme (2.3), we need to choose a set of basis functions of V_h^k . For example, the local orthogonal Legendre basis function:

$$\phi_j^0(x) = 1, \quad \phi_j^1(x) = \frac{x - x_j}{\Delta x_j}, \quad \phi_j^2(x) = \left(\frac{x - x_j}{\Delta x_j} \right)^2 - \frac{1}{12}, \dots, \quad x \in I_j,$$

for $j = 1, \dots, N$ can be used. Then, the numerical solution can be represented as

$$u_h(t, x) = \sum_{l=0}^k u_j^l(t) \phi_j^l(x), \quad x \in I_j, \quad (2.4)$$

where $u_j^l(t)$ are the degrees of freedom. Denote by

$$\mathbf{u}_j(t) = (u_j^0(t), \dots, u_j^k(t))^T. \quad (2.5)$$

Plugging (2.4) into the DG scheme (2.3), we arrive at an ODE system for the degrees of freedom:

$$\frac{d}{dt}u_h(t) = L(u_h(t)), \quad (2.6)$$

where $L(\cdot)$ is the spatial discretization operator of the DG scheme.

2.2 Algorithm Formulation of LDG schemes

In this subsection, we review the LDG scheme [30] for solving the following 1-D diffusion equation:

$$u_t = (c(u)u_x)_x, \quad x \in [a, b], \quad t > 0, \quad (2.7)$$

with suitable initial and boundary conditions. Here we assume $c(u) \geq 0$. The domain discretization is same as the DG scheme. In order to define the LDG scheme, we rewrite equation (2.7) into the following system by introducing an auxiliary variable $p = \sqrt{c(u)}u_x$:

$$u_t = (q'p)_x, \quad (2.8)$$

$$p = q_x, \quad (2.9)$$

2.2. ALGORITHM FORMULATION OF LDG SCHEMES

where $q(u) = \int^u \sqrt{c(s)} ds$. The semi-discrete LDG method for solving (2.8)-(2.9) is defined as follows: find $u_h, p_h \in V_h^k$ such that $\forall v, w \in V_h^k$, we have

$$\begin{aligned} \int_{I_j} (u_h)_t v dx + \int_{I_j} q' p_h v_x dx - \widehat{q' p_h}_{j+\frac{1}{2}} v_{j+\frac{1}{2}}^- + \widehat{q' p_h}_{j-\frac{1}{2}} v_{j-\frac{1}{2}}^+ &= 0, \\ \int_{I_j} p_h w dx + \int_{I_j} q w_x dx - \hat{q}_{j+\frac{1}{2}} w_{j+\frac{1}{2}}^- + \hat{q}_{j-\frac{1}{2}} w_{j-\frac{1}{2}}^+ &= 0, \end{aligned} \quad (2.10)$$

for $j = 1, \dots, N$. $\widehat{q' p_h}$ and \hat{q} are numerical fluxes that are carefully designed for attaining provable L^2 stability in the semi-discrete sense. For example, in [30], the following numerical fluxes were proposed,

$$\widehat{q' p_h} = \frac{[[q(u_h)]]}{[[u_h]]} p_h^-, \quad (2.11)$$

$$\hat{q} = q(u_h^+), \quad (2.12)$$

where

$$[[u_h]] = u_h^+ - u_h^-$$

denotes the jump of function u_h at a spatial location.

Note that the LDG solution can be represented in the form of (2.4). Plugging (2.4) into the LDG scheme (2.10), we obtain a similar ODE system as (2.6) for the degrees of freedom of the LDG scheme.

2.3 Time Discretization: Runge-Kutta Methods

The time variable in the DG and LDG schemes can be further discretized through an explicit high order Runge-Kutta (RK) method in a method of lines (MOL) framework. The resulting fully discrete schemes are usually termed the RKDG and RKLDG schemes in the literature. In the simulations, a strong stability preserving (SSP) RK method is preferred [44, 43], since it can help enhance numerical stability and control spurious oscillations by maintaining the strong stability properties of the forward Euler time stepping. Despite great success in many applications, the RKDG and RKLDG schemes are known to suffer a CFL time step restriction. For instance, the time step Δt of a RKDG scheme obeys

$$\Delta t \leq \frac{\text{CFL}}{|a_x|} \min_j \Delta x_j \quad (2.13)$$

where $|a_x|$ is the maximum wave propagation speed and CFL is a constant depending on the DG scheme and RK method being used.

Below, we list several SSP RK methods. For illustrative purposes, we consider the following ODE systems:

$$\frac{d}{dt}u(t) = L(u(t)).$$

Denote a p_1 -stage p_2 -order SSP RK method by SSPRK(p_1, p_2).

- SSPRK(2,2):

$$\begin{aligned} u^{(1)} &= u^n + \Delta t L(u^n), \\ u^{n+1} &= \frac{1}{2}u^n + \frac{1}{2}(u^{(1)} + \Delta t L(u^{(1)})); \end{aligned}$$

- SSPRK(3,3) [104]:

$$\begin{aligned} u^{(1)} &= u^n + \Delta t L(u^n), \\ u^{(2)} &= \frac{3}{4}u^n + \frac{1}{4}(u^{(1)} + \Delta t L(u^{(1)})), \\ u^{n+1} &= \frac{1}{3}u^n + \frac{2}{3}(u^{(2)} + \Delta t L(u^{(2)})); \end{aligned}$$

- SSPRK(5,4) [43]:

$$\begin{aligned} u^{(1)} &= u^n + 0.391752226571890\Delta t L(u^n), \\ u^{(2)} &= 0.444370493651235u^n + 0.555629506348765u^{(1)} \\ &\quad + 0.368410593050371\Delta t L(u^{(1)}), \\ u^{(3)} &= 0.620101851488403u^n + 0.379898148511597u^{(2)} \\ &\quad + 0.251891774271694\Delta t L(u^{(2)}), \\ u^{(4)} &= 0.178079954393132u^n + 0.821920045606868u^{(3)} \\ &\quad + 0.544974750228521\Delta t L(u^{(3)}), \\ u^{n+1} &= 0.517231671970585u^{(2)} + 0.096059710526147u^{(3)} \\ &\quad + 0.063692468666290\Delta t L(u^{(3)}) + 0.386708617503269u^{(4)} \\ &\quad + 0.226007483236906\Delta t L(u^{(4)}) \end{aligned}$$

2.3. TIME DISCRETIZATION: RUNGE-KUTTA METHODS

- SSPRK(9,9) [44], denote by $u^{(0)} = u^n$:

$$\begin{aligned}u^{(i)} &= u^{(i-1)} + \Delta t L(u^{(i-1)}), \quad i = 1, \dots, 8, \\u^{n+1} &= \sum_{i=0}^7 \alpha_{9,i} u^{(i)} + \alpha_{9,8} (u^{(8)} + \Delta t L(u^{(8)})),\end{aligned}$$

where

$$\begin{aligned}\alpha_{9,0} &= \frac{16687}{45360}; & \alpha_{9,1} &= \frac{2119}{5760}; & \alpha_{9,2} &= \frac{103}{560}; \\ \alpha_{9,3} &= \frac{53}{864}; & \alpha_{9,4} &= \frac{11}{720}; & \alpha_{9,5} &= \frac{1}{320}; \\ \alpha_{9,6} &= \frac{1}{2160}; & \alpha_{9,7} &= \frac{1}{10080}; & \alpha_{9,8} &= \frac{1}{362880}.\end{aligned}$$

Note that SSPRK(9,9) is only valid when operator L is linear.

CHAPTER 3

High Order Semi-Lagrangian Schemes

The aim of this chapter is to formulate a class of high order SL methods for solving the VP system and global transport equations on the cubed sphere. First, two formulations of high order SLDG schemes are developed for solving linear transport equations. Note that the two formulations are mathematically equivalent, but they differ in the numerical discretization and hence produce different numerical results. Second, high order SL finite difference WENO (SLWENO) schemes for linear transport equations are introduced. Then, these high order SL schemes are applied to the Vlasov simulations and global transport modeling in different settings.

3.1 High Order Semi-Lagrangian DG Schemes

As we discussed in Chapter 1, the RKDG schemes suffer the stringent CFL time step restriction which leads to computational inefficiency in many applications. Moreover, the situation deteriorates when increasing the degree of the used approximation space. In particular, it is pointed out in [32] that

$$\text{CFL} \approx \frac{1}{2k+1}$$

for DG discretizations using V_h^k combined with $(k+1)^{\text{th}}$ order accurate RK schemes. In order to overcome the shortcoming, we consider coupling the SL approach with the DG spatial discretizations. The resulting SLDG scheme differs from the RKDG scheme in its time evolution mechanism. Specifically, characteristics are being tracked over time in the SLDG formulation. However, we note that the tracking of characteristics can be theoretically and computationally complicated for nonlinear and multi-dimensional problems. First, we consider the following 1-D linear transport equations in the conservative form:

$$u_t + (a(t, x)u)_x = 0, \tag{3.1}$$

where a can be variable coefficient depending on t and x . In the following two subsections, we will formulate two types of SLDG schemes for solving (3.1). Notations used in this section are consistent with Chapter 2.

3.1.1 First Formulation

The first formulation we will introduce below is propose by Qiu and Shu in [85] is designed based on the semi-discrete DG scheme (2.3) with flux function $f(u) = a(t, x)u$. Specifically, we integrate equation (2.3) in time from $[t^n, t^{n+1}]$ and obtain,

$$\begin{aligned} \int_{I_j} u_h^{n+1} v dx &= \int_{I_j} u_h^n \cdot v dx + \int_{t^n}^{t^{n+1}} \int_{I_j} a u_h(t, x) v_x dx dt \\ &\quad - \int_{t^n}^{t^{n+1}} \left(\widehat{a u_h}(t, x_{j+\frac{1}{2}}) v_{j+\frac{1}{2}} - \widehat{a u_h}(t, x_{j-\frac{1}{2}}) v_{j-\frac{1}{2}} \right) dt, \end{aligned} \quad (3.2)$$

where $u_h^n = u_h(t^n, x)$ denotes the numerical solution at time level t^n . In order to attain unconditional stability for scheme (3.2), the time integration of the flux function terms have to be evaluated in the SL fashion. To do that, we observe in Figure 3.1 that at a fixed spatial location at time level t^{n+1} , say $(x_{j-\frac{1}{2}}, t^{n+1})$, there exists a backward characteristic curve, with the foot (departure point) located on time level t^n at $x_{j-\frac{1}{2}}^*$. We denote the region $\Omega_{j-\frac{1}{2}}$ to be the region bounded by the three points $(x_{j-\frac{1}{2}}, t^{n+1})$, $(x_{j-\frac{1}{2}}, t^n)$ and $(x_{j-\frac{1}{2}}^*, t^n)$. We apply the divergence theorem to the integral form of equation (3.1) over the region $\Omega_{j-\frac{1}{2}}$, and obtain

$$\int_{t^n}^{t^{n+1}} \widehat{a u_h}(t, x_{j-\frac{1}{2}}) dt = \int_{x_{j-\frac{1}{2}}^*}^{x_{j-\frac{1}{2}}} u_h^n(x) dx, \quad (3.3)$$

which can be used to evaluate the time integration of the flux terms on the r.h.s. of equation (3.2). Note that the numerical solution u_h^n is available everywhere in the computational domain. The volume integral in (3.2) can be evaluated in a similar way. First, we denote x_{ig} for $ig = 1, 2, \dots, k+1$ are $k+1$ local Gaussian points on

cell I_j and w_{ig} are the corresponding quadrature weights over a unit length interval.

The volume integral can be approximated by

$$\begin{aligned} \int_{t^n}^{t^{n+1}} \int_{I_j} au_h(t, x)v_x dx dt &\approx \sum_{ig=1}^{k+1} w_{ig}v_x(x_{ig}) \int_{t^n}^{t^{n+1}} au_h(t, x_{ig}) dt \\ &= \Delta x_j \sum_{ig=1}^{k+1} w_{ig}v_x(x_{ig}) \int_{x_{ig}^*}^{x_{ig}} u_h^n(x) dx, \end{aligned} \quad (3.4)$$

where x_{ig}^* denotes the foot of the characteristic curve emanating from (t^{n+1}, x_{ig}) .

In summary, the SLDG scheme is formulated as follows: given $u_h^n \in V_h^k$, find $u_h^{n+1} \in V_h^k$ such that $\forall v \in V_h^k$, equation (3.2) holds with the flux integrals and the volume integral specified in (3.3) and (3.4), respectively.

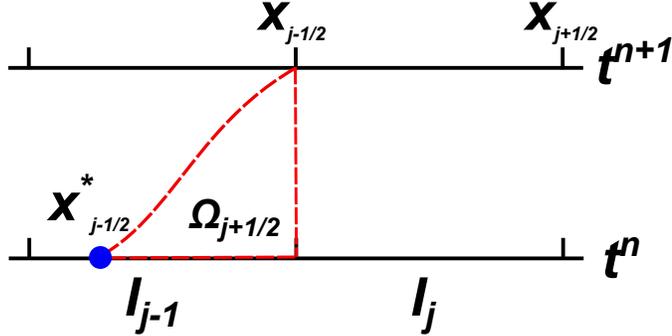


Figure 3.1: SL scheme approximates equation (3.1).

3.1.2 Second Formulation

The second SLDG formulation is based on the weak formulation of characteristic Galerkin method of [18, 98], which is proposed in [46]. We also consider the 1-D linear transport equation (3.1). To update the solution at the time level t^{n+1} over a cell I_j from the solution at t^n , we let the test function $V(t, x)$ satisfy the adjoint problem with $v \in V_h^k$,

$$\begin{cases} V_t + a(t, x)V_x = 0, \\ V(t^{n+1}) = v(x). \end{cases} \quad (3.5)$$

We remark that for the above advective form of equation (3.5), the solution stays constant along the characteristic curve; while for the conservative form of equation (3.1), the solution varies along the curve. It can be shown that

$$\frac{d}{dt} \int_{I_j(t)} u(t, x)V(t, x)dx = 0, \quad (3.6)$$

where $I_j(t)$ is a dynamic interval bounded by characteristics emanating from cell boundaries of I_j at $t = t^{n+1}$. Equation (3.6) can be proved by the following:

$$\begin{aligned} & \frac{d}{dt} \int_{I_j(t)} u(t, x)V(t, x)dx \\ = & \frac{dx}{dt} \Big|_{x_{j+\frac{1}{2}}} u(t, x_{j+\frac{1}{2}}(t))V(t, x_{j+\frac{1}{2}}(t)) - \frac{dx}{dt} \Big|_{x_{j-\frac{1}{2}}} u(t, x_{j-\frac{1}{2}}(t))V(x_{j-\frac{1}{2}}(t), t) \\ & - \int_{I_j(t)} (a(t, x)u(t, x))_x V(t, x)dx + \int_{I_j(t)} u(t, x)V_t(t, x)dx \end{aligned}$$

$$\begin{aligned}
 &= auV|_{(t,x_{j+\frac{1}{2}}(t))} - auV|_{(t,x_{j-\frac{1}{2}}(t))} - \left(auV|_{(t,x_{j+\frac{1}{2}}(t))} - auV|_{(t,x_{j-\frac{1}{2}}(t))} \right) \\
 &\quad + \int_{I_j(t)} a(t,x)u(t,x)V_x(t,x)dx - \int_{I_j(t)} u(t,x)a(t,x)V_x(t,x)dx \\
 &= 0.
 \end{aligned}$$

A SL time discretization of (3.6) leads to

$$\int_{I_j} u^{n+1}vdx = \int_{I_j^*} u^nV(t^n, x)dx, \tag{3.7}$$

where $I_j^* \doteq [x_{j-\frac{1}{2}}^*, x_{j+\frac{1}{2}}^*]$ with $x_{j\pm\frac{1}{2}}^*$ being the foot of the characteristic curve emanating from $(t^{n+1}, x_{j\pm\frac{1}{2}})$ at time t^n . To update the numerical solution u_h^{n+1} , the following procedures are performed.

1. Locate the foot of the characteristics $x_{j\pm\frac{1}{2}}^*$ (see, Figure 3.2 (left)). We numerically solve the following final-value problem (characteristic equation):

$$\frac{d}{dt}x(t) = a(t, x(t)) \tag{3.8}$$

with the final-value $x(t^{n+1}) = x_{j\pm\frac{1}{2}}$ by a high order numerical integrator such as a classical fourth order RK method.

2. Detect intervals/sub-intervals within $I_j^* = \cup_l I_{j,l}^*$, which are all the intersections between I_j^* and the grid elements. (l is the index for sub-interval). For example, in Figure 3.2 (left), there are two sub-intervals: $I_{j,1}^* = [x_{j-1/2}^*, x_{j-1/2}]$ and $I_{j,2}^* = [x_{j-1/2}, x_{j+1/2}^*]$.
3. Locate the $k + 1$ local Gaussian-Lobatto (GL) points over each $I_{j,l}^*$. We denote

them as $x_{j,l,ig}^*$ (ig is the index for GL points). See the red circles as 4 GL points per sub-interval in Figure 3.2 (right).

4. Trace the characteristics forward in time from $(t^n, x_{j,l,ig}^*)$ to $(t^{n+1}, x_{j,l,ig})$. Especially, similar to the final-value problem above, we use a high order time integrator to numerically solve an initial value problem (3.8) with the initial-values $x(t^n) = x_{j,l,ig}^*$ (see the green curve and circles in Figure 3.2 (right)). From the advective form of the adjoint problem (3.5) one has

$$V(t^n, x_{j,l,ig}^*) = v(x_{j,l,ig}).$$

5. Use the GL quadrature rule to evaluate

$$\int_{I_j^*} u_h^n V(t^n, x) dx \approx \sum_l \left(\sum_{ig=1}^{k+1} w_{ig} u_h^n(x_{j,l,ig}^*) v(x_{j,l,ig}) \Gamma(I_{j,l}^*) \right), \quad (3.9)$$

with w_{ig} being the quadrature weights for a unit length interval and $\Gamma(I_{j,l}^*)$ being the length of interval $I_{j,l}^*$. Note that the accuracy of the GL quadrature rule is order of $2k$ when $k+1$ GLL points are used. As in the classical DG formulation, the evaluation of volume integral will not destroy the $(k+1)^{th}$ order of accuracy for the SLDG scheme. Moreover, the mass conservation properties are not affected since the numerical integration is exact for a polynomial of degree k with the test function $v = 1$. As an alternative, the Gaussian quadrature rule can be used to compute the integrals.

6. Finally, find $u_h^{n+1} \in V_h^k$, s.t. (3.7) is satisfied $\forall v \in V_h^k$ with the r.h.s term

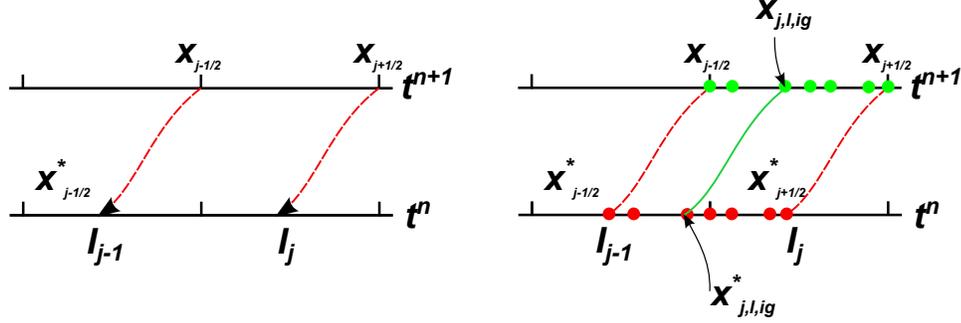


Figure 3.2: Schematic showing the 1-D SLDG scheme (3.7)-(3.9) (second formulation). Step i and Step ii (left); Step iii and Step iv (right). 4 GL points per cell are used as an example.

evaluated as described above.

Remark 3.1. Note that the proposed 1-D algorithm is fourth order accurate in time in the sense of local truncation error, i.e.

$$\frac{1}{\Delta t} |x_{j\pm\frac{1}{2}}^* - x(t^n; x_{j\pm\frac{1}{2}}, t^{n+1})| = O(\Delta t^4), \quad \frac{1}{\Delta t} |x_{j,l,ig} - x(t^{n+1}; x_{j,l,ig}^*, t^n)| = O(\Delta t^4)$$

with a fourth order RK method for tracing characteristics. Here $\Delta t = t^{n+1} - t^n$; $x(t^n; x_{j\pm\frac{1}{2}}, t^{n+1})$ and $x(t^{n+1}; x_{j,l,ig}^*, t^n)$ denote the exact solutions of the characteristic equation (3.8) with final-value $x(t^{n+1}) = x_{j\pm\frac{1}{2}}$ and initial-value $x(t^n) = x_{j,l,ig}^*$, respectively.

Remark 3.2. The proposed second SLDG formulation differs from the first formulation (3.2) and the one proposed in [94], which follow an Eulerian volume (fixed spatial cell) with numerical fluxes obtained from tracing the characteristics. However, the second formulation follows the Lagrangian volume dynamically moving with

the characteristics, in the same spirit as in [97]. Also note that, the proposed formulation is more general than that in [97], in the sense that it permits a non-uniform velocity field and is extendable to curvilinear coordinates, whereas the SLDG method in [97] can only be applied to problems with constant velocity.

3.1.3 The Bound-Preserving (BP) Filter

It is known that equation (3.1) is mass conservative. However, the solution does not fulfill the maximum principle, i.e. the solution in the future time will not be bounded by the lower and upper bound of the initial condition. On the other hand, if the initial condition for (3.1) is positive ($u(t = 0, x) \geq 0, \forall x$), then the future solution stays positive ($u(t, x) \geq 0, \forall x, t$). We call such property as positivity preserving (PP). In the SLDG schemes (the first formulation and the second formulation), it can be easily checked that the updated cell average at t^{n+1} stays positive, if the numerical solution u_h^n (piecewise polynomial function) at t^n is positive. However, the numerical solution u_h^{n+1} at t^{n+1} does not necessarily stay positive. To ensure PP of the numerical solution, we apply a BP filter [121, 122, 87, 123], if the initial condition stays positive. The procedure of the BP filter can be viewed as ensuring the positivity of the numerical solution by a linear rescaling around the cell averages, with the assumption that the cell averages are positivity preserving. In particular, the numerical solution is modified from $u(x)$ to $\tilde{u}(x)$ such that it maintains the high order accuracy of the original approximation, conserves the cell average (mass), and

preserves positivity:

$$\tilde{u}(x) = \theta(u(x) - \bar{u}) + \bar{u}, \quad \theta = \min \left\{ \left| \frac{\bar{u}}{m' - \bar{u}} \right|, 1 \right\}, \quad (3.10)$$

where \bar{u} is the cell average of the numerical solution and m' is the minimum of $u(x)$ over a given cell. A formal proof can be found in [121] (Lemma 2.4). To implement the BP filter, the minimum of the numerical solution m' is needed. In our numerical tests, we use up to P^3 polynomials, the minimum of which can be easily found by locating the zeros of their derivatives. The proposed SLDG methods with the BP filter enjoy the L^1 (mass) conservation, the proof of which can be found in [87].

3.2 High Order Semi-Lagrangian WENO Schemes

Another successful numerical method, in the class of finite difference schemes, for computational fluid dynamics as well as kinetic simulations, is the high order finite difference WENO scheme [60]. The SSP RK method can be used for time evolution. The most distinctive feature of finite difference WENO scheme is its ability in resolving complicated solution structures in a robust and stable way when compared with a DG scheme; as well as its ease and flexibility in multi-dimensional implementations by working with point values in a dimension-by-dimension fashion when compared with a finite volume method.

Similar to the SLDG schemes, instead of using SSP RK time integrator, one can use the SL evolution mechanism to evaluate the time integral of the flux functions.

In this section, we formulate a high order SLWENO scheme for the 1-D transport equation (3.1), which has been shown to be successful due to the conservative nature of the scheme formulation, the flexibility of the finite difference framework, the robustness of WENO reconstruction and the large time step efficiency of SL methods [85].

Unlike finite element schemes, the computational domain in a finite difference framework is discretized by the uniformly distributed grid points

$$a = x_0 < \cdots < x_i \cdots < x_N = b.$$

We still denote $\Delta x = (b-a)/N$ as the mesh size, and denote by $x_{i+1/2} = (x_i + x_{i+1})/2$. Let u_j^n be the point value of numerical solution at grid point x_j and time level t^n .

Similar to the first SLDG formulation (3.2), the SLWENO scheme is based on integrating the conservative form of equation (3.1) over $[t^n, t^{n+1}]$,

$$u(t^{n+1}, x) = u(t^n, x) - \left(\int_{t^n}^{t^{n+1}} a(t, x) u(t, x) dt \right)_x.$$

Evaluating the above equation at the grid point x_j gives

$$u_j^{n+1} = u_j^n - \left(\int_{t^n}^{t^{n+1}} a(t, x) u(t, x) dt \right) \Big|_{x=x_j} = u_j^n - \mathcal{F}_x \Big|_{x=x_j}, \quad (3.11)$$

where $\mathcal{F}(x) \doteq \int_{t^n}^{t^{n+1}} a(t, x) u(t, x) dt$. Let $\mathcal{H}(x)$ be a function whose sliding average is

$\mathcal{F}(x)$, i.e.,

$$\mathcal{F}(x) = \frac{1}{\Delta x} \int_{x-\frac{\Delta x}{2}}^{x+\frac{\Delta x}{2}} \mathcal{H}(\xi) d\xi. \quad (3.12)$$

Taking the x derivative of the above equation gives

$$\mathcal{F}_x = \frac{1}{\Delta x} \left(\mathcal{H}\left(x + \frac{\Delta x}{2}\right) - \mathcal{H}\left(x - \frac{\Delta x}{2}\right) \right).$$

Therefore the equation (3.11) can be written in a conservative form as

$$u_j^{n+1} = u_j^n - \frac{1}{\Delta x} (\mathcal{H}(x_{j+\frac{1}{2}}) - \mathcal{H}(x_{j-\frac{1}{2}})), \quad (3.13)$$

where $\mathcal{H}(x_{j+\frac{1}{2}})$ is called the flux function. Again, $\mathcal{H}(x_{j+\frac{1}{2}})$ should be approximated in a SL fashion in order to attain the unconditional stability. By following information along characteristics, i.e., (3.3), $\mathcal{F}(x_j) = \int_{t^n}^{t^{n+1}} a(t, x_j) u(t, x_j) dt = \int_{x_j^*}^{x_j^i} u(t^n, x) dx$ can be evaluated by reconstructing function u at t_n from neighboring point values, where x_j^* denotes the foot of the characteristic curve at t^n . We denote this reconstruction procedure as \mathcal{R}_1 . Similar to the idea in the finite difference WENO scheme, $\mathcal{H}(x_{j+\frac{1}{2}})$ can be reconstructed in high order from several of its neighboring cell averages

$$\bar{\mathcal{H}}_k = \frac{1}{\Delta x} \int_{x_{k-\frac{1}{2}}}^{x_{k+\frac{1}{2}}} \mathcal{H}(\xi) d\xi \stackrel{(3.12)}{=} \mathcal{F}(x_k), \quad k = j-p, \dots, j+q.$$

We denote this reconstruction procedure as \mathcal{R}_2 .

In summary, a SL finite difference scheme in evolving equation (3.1) from t^n to t^{n+1} can be designed as follows:

1. At each of the grid points at time level t^{n+1} , say (x_i, t_{n+1}) , trace the characteristic back to time level t^n by solving the characteristic equation (3.6) and get the foot x_j^* .
2. Reconstruct $\mathcal{F}(x_j) = \int_{t^n}^{t^{n+1}} au(t, x_j)dt = \int_{x_j^*}^{x_j} udx$ from $\{u_j^n\}_{j=1}^N$. We use \mathcal{R}_1 to denote this reconstruction procedure $\mathcal{R}_1[x_j^*, x_j](f_{j-p_1}^n, \dots, u_{j+q_1}^n)$ in approximating $\mathcal{F}(x_j)$, where $(j - p_1, \dots, j + q_1)$ indicates the stencil used in the reconstruction. $\mathcal{R}_1[c, d]$ indicates the reconstruction of $\int_c^d u(t, x)dx$.
3. Reconstruct $\{\mathcal{H}(x_{j+\frac{1}{2}})\}_{j=0}^N$ from $\{\bar{\mathcal{H}}_j\}_{j=1}^N$. We use \mathcal{R}_2 to denote this reconstruction procedure $\hat{F}_{j+\frac{1}{2}} \doteq \mathcal{R}_2(\bar{\mathcal{H}}_{j-p_2}, \dots, \bar{\mathcal{H}}_{j+q_2})$ in approximating $\mathcal{H}(x_{j+\frac{1}{2}})$, where $(j - p_2, \dots, j + q_2)$ indicates the stencil used in the reconstruction.
4. Update the solution $\{u_j^{n+1}\}_{j=1}^N$ by

$$u_j^{n+1} = u_j^n - \frac{1}{\Delta x}(\hat{F}_{j+\frac{1}{2}} - \hat{F}_{j-\frac{1}{2}}), \quad (3.14)$$

with numerical fluxes $\hat{F}_{j\pm\frac{1}{2}}$ computed in the previous step.

When the reconstruction stencils in \mathcal{R}_1 and \mathcal{R}_2 above only involve one neighboring point value of the solution, then the scheme is first order accurate in space. In fact, the scheme reduces to a first order upwind scheme when the time step is under the CFL restriction. The proposed SL finite difference scheme can be designed to be of high order accuracy by including more points in the stencil for $\mathcal{R}_2 \circ \mathcal{R}_1$, the composition of \mathcal{R}_1 and \mathcal{R}_2 , to reconstruct the numerical flux

$$\hat{F}_{j+\frac{1}{2}} = \mathcal{R}_2 \circ \mathcal{R}_1(u_{j-p}^n, \dots, u_{j+q}^n), \quad (3.15)$$

where $(j-p, \dots, j+q)$ indicates the stencil used in the reconstruction process. The WENO mechanism can be introduced in reconstruction procedures as a stable and non-oscillatory method to capture fine-scale structures. It is numerically demonstrated in [84, 85, 86] that the high order SLWENO method works very well in the Vlasov and incompressible flow simulations with extra large time step evolution.

There are different conservative SL finite difference procedures proposed in [84, 85, 86]. These approaches have similar performance for linear advection equations with constant coefficients. Below, we briefly review another SLWENO scheme proposed in [84]. Assume the velocity a is a *constant* and let $\xi_0 = a\frac{\Delta t}{\Delta x}$. Without loss of generality, we assume $\xi_0 \in [0, \frac{1}{2}]$. When $\xi_0 \in [-\frac{1}{2}, 0]$, a similar but symmetric procedure can be applied. Otherwise, ξ_0 can be shifted to $[-\frac{1}{2}, \frac{1}{2}]$ via grid shifting.

By using the fact that the solution stays constant along characteristics, i.e.,

$$u(t^{n+1}, x_j) = u(t^n, x_j - a\Delta t),$$

the solution can be updated via a high order reconstruction of $u(x_j - a\Delta t, t^n)$ from neighboring point values. Taking a third order reconstruction for example, see Figure 3.3, we obtain

$$\begin{aligned} u_j^{n+1} &= u_j^n + \left(-\frac{1}{6}u_{j-2}^n + u_{j-1}^n - \frac{1}{2}u_j^n - \frac{1}{3}u_{j+1}^n\right)\xi_0 \\ &\quad + \left(\frac{1}{2}u_{j-1}^n - u_j^n + \frac{1}{2}u_{j+1}^n\right)\xi_0^2 \\ &\quad + \left(\frac{1}{6}u_{j-2}^n - \frac{1}{2}u_{j-1}^n + \frac{1}{2}u_j^n - \frac{1}{6}u_{j+1}^n\right)\xi_0^3, \end{aligned} \tag{3.16}$$

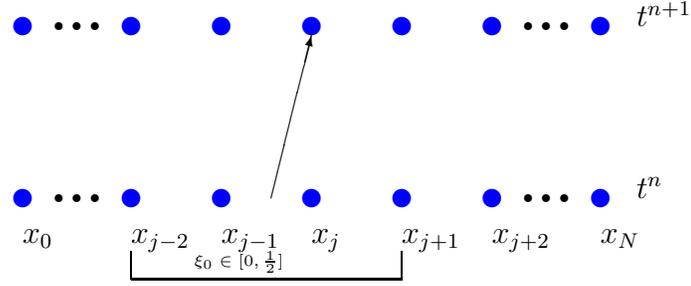


Figure 3.3: SL finite difference WENO reconstruction.

Based on the linear reconstruction (3.16), a nonlinear WENO reconstruction can be introduced to suppress numerical oscillations when the solution is under-resolved [12]. However, the mass conservation, which is a very important property for solving hyperbolic conservation laws, may be lost. To ensure the mass conservation, equation (3.16) is rewritten in a flux difference form:

$$u_j^{n+1} = u_j^n - \xi_0(\hat{F}_{j+\frac{1}{2}} - \hat{F}_{j-\frac{1}{2}}). \quad (3.17)$$

$\hat{F}_{j+\frac{1}{2}}$ is the numerical flux defined as

$$\hat{F}_{j+\frac{1}{2}} = (u_{j-1}^n, u_j^n, u_{j+1}^n) \cdot C_3^L \cdot (1, \xi_0, \xi_0^2)', \quad (3.18)$$

where

$$C_3^L = \begin{pmatrix} -\frac{1}{6} & 0 & \frac{1}{6} \\ \frac{5}{6} & \frac{1}{2} & -\frac{1}{3} \\ \frac{1}{3} & -\frac{1}{2} & \frac{1}{6} \end{pmatrix}.$$

3.3. EXTENSION TO MULTI-DIMENSIONAL PROBLEMS VIA DIMENSIONAL SPLITTING

Similarly, the WENO mechanism can be incorporated into the reconstruction procedure of the numerical flux $\hat{F}_{j+\frac{1}{2}}$ in order to control non-physical oscillations. The algorithm can be generalized to higher order reconstructions, e.g., 5th, 7th, and 9th order, see [84].

Remark 3.3. *Note that the SLWENO scheme (3.14)-(3.15) from [85] is more general in the sense that it can be applied to linear transport equations with variable coefficients. For solving transport equations with a constant coefficient, such as the split Vlasov equations (1.8)-(1.9), the SLWENO scheme from [84], e.g., (3.17)-(3.18) for a third order scheme, can also be applied. However, for the guiding center Vlasov model, the split equations have variable coefficients. Hence, the SLWENO scheme (3.14)-(3.15) is needed.*

3.3 Extension to Multi-dimensional Problems via Dimensional Splitting

The 1-D SL schemes can be extended to multi-dimensional algorithms via the second order Strang dimension splitting based on a Cartesian mesh, see, e.g., [108]. Below, we sketch the idea of the algorithm for SLDG schemes, which also applies to SLWENO schemes with some minor modifications.

Consider a 2-D transport equation in the conservative form

$$u_t + (a(t, x, y)u)_x + (b(t, x, y)u)_y = 0. \quad (3.19)$$

3.3. EXTENSION TO MULTI-DIMENSIONAL PROBLEMS VIA DIMENSIONAL SPLITTING

When the velocity field (a, b) is non-divergent, i.e., $\nabla \cdot (a, b) = 0$, the equation is equivalent to the advective form,

$$u_t + a(t, x, y)u_x + b(t, x, y)u_y = 0. \quad (3.20)$$

The SLDG scheme for (3.19) is outlined as follows:

Algorithm 3.4. A dimensional splitting SLDG scheme for 2-D transport equations:

1. Split the equation (3.19) into two 1-D advection problems:

$$u_t + (a(t, x, y)u)_x = 0, \quad (3.21)$$

$$u_t + (b(t, x, y)u)_y = 0. \quad (3.22)$$

2. Locate $k+1$ GL points in both x - and y -directions in each rectangular cell as (x_{ig}, y_{jg}) . See Figure 3.4 (left).
3. Perform the Strang dimension-splitting strategy, for which the numerical update over a time step Δt is as follows:
 - (a) Evolve 1-D equation (3.21) at different y_{jg} locations with corresponding velocities $u(x, y_{jg}, t)$ for a half time step $\Delta t/2$, see Figure 3.4(middle).
 - (b) Evolve 1-D equation (3.22) at different x_{ig} locations with corresponding velocities $v(x_{ig}, y, t)$ for a whole step Δt , see Figure 3.4(right).
 - (c) Evolve 1-D equation (3.21) for another half time step $\Delta t/2$ as (a).

Note that the BP filter is applied separately in each direction and the resulting

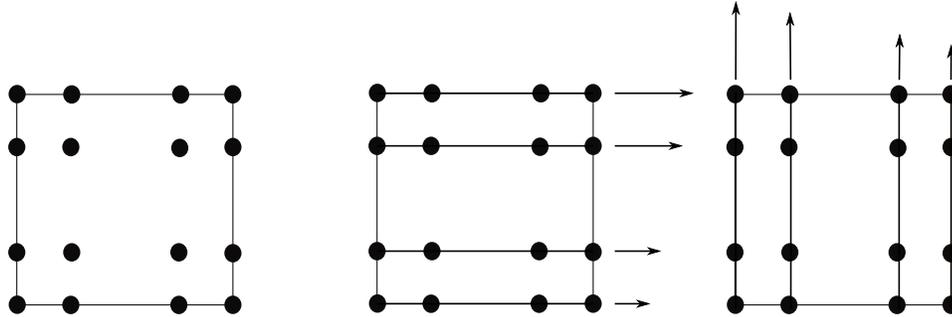


Figure 3.4: Schematic showing the 2-D SLDG scheme via Strang splitting, as described in the text. Locate $k + 1$ GL points in both x - and y - directions (left); Evolution in x -direction at different y_{jg} (middle). Evolution in y -direction at different x_{ig} (right). 4×4 GL points per cell are used as an example.

scheme can preserve positivity, see [97, 87]. It is difficult to design a numerical scheme that preserves a constant field in the dimensional splitting framework. It is our ongoing work to design a non-splitting SLDG scheme that preserves the constant field when the velocity field is non-divergent.

3.4 Applications to Vlasov Simulations

In this section, we develop a class of high order dimensional splitting SL schemes coupling the SLDG schemes in Section 3.1 and the SLWENO schemes in Section 3.2 for Vlasov simulations. In particular, we will formulate a hybrid methodology for solving the VP system in Subsection 3.4.1. Then, the integral deferred correction (IDC) method is used in correcting the dimensional splitting error, which is discussed in Subsection 3.4.2.

3.4.1 A Hybrid Methodology

For Vlasov simulations, a very popular time stepping strategy is the Strang splitting [13]. There are many advantages associated with such splitting. For example, the nonlinearity of the VP system is being decoupled. As a result, the split equations can be independently evolved, for which a ‘best’ numerical method can be chosen for each of these independent evolutions. In the proposed hybrid method we adopt the Strang splitting idea again. In particular, the six-dimensional Vlasov equation (1.1) is split into two equations, for spatial advection (1.8) and velocity ac/deceleration (1.9), respectively. The splitting of equation (1.1) can be designed to be second order accurate in time by advecting the equation (1.8) in spatial direction for a half time step $\Delta t/2$, then evolving equation (1.9) in velocity direction for a full time step Δt , followed by solving equation (1.8) for a second half time step $\Delta t/2$. The two advection equations (1.8) and (1.9) are linear. They can be solved by independent numerical solvers, wisely chosen to be the most suitable ones for individual problems. For many application problems such as tokamak, the spatial domain may have complicated geometry and general boundary conditions. Depending on the shape of spatial domain, we propose to apply the SLDG scheme or the Eulerian RKDG scheme with local time stepping for evolving equation (1.8) based on a structured rectangular or an unstructured triangular discretization of the spatial domain. On the other hand, the computational domain for velocity space is simply a cuboid $\mathbf{v} \in [v_{1,min}, v_{1,max}] \times [v_{2,min}, v_{2,max}] \times [v_{3,min}, v_{3,max}]$ with zero boundary conditions. Because of this, we propose to apply the high order SLWENO scheme to evolve equation (1.9), which is demonstrated to be very robust in resolving the filamentation

3.4. APPLICATIONS TO VLASOV SIMULATIONS

solution structures in Vlasov simulations.

Below, we consider the split Vlasov equation with 1-D in space and 1-D in velocity (1D1V) to illustrate the idea of the proposed hybrid scheme, i.e.,

$$f_t + v f_x = 0 \quad \text{spatial advection,} \quad (3.23)$$

$$f_t + E(t, x) f_v = 0 \quad \text{velocity ac/deceleration.} \quad (3.24)$$

The numerical mesh is based on a tensor product of the following discretization in x and v directions respectively,

$$\mathcal{D}_x : [a, b] = \bigcup_{i=1}^{N_x} I_i,$$

where I_i are non-overlapping intervals, not necessarily uniform, the union of which is the domain $[a, b]$; and

$$\mathcal{D}_v : -v_{max} = v_0 < v_1 < \dots < v_{N_v/2-1} < v_{N_v} = v_{max},$$

where the distribution of grid points are uniform. The solution space for this 1D1V problem is defined as

$$V_{\mathcal{D}_{x,v}}^k = \{f(x, v = v_j)|_{I_i} \doteq f_{i,j}(x) \in P^k(I_i), \quad i = 1, \dots, N_x, \quad j = 0, \dots, N_v\},$$

which consists of piecewise polynomials defined on intervals of x , I_i , and at fix locations of v , v_j , $\forall i = 1, \dots, N_x$ and $j = 0, \dots, N_v$. The evolution procedure per time

step Δt of the proposed splitting scheme for the VP system is described in the rest of this subsection. We remark that the time step Δt is not restricted by the CFL stability constrain in the proposed framework.

Advection in Spatial Direction

We adopt a RKDG scheme or a SLDG scheme for advection in spatial direction, denoted as DG_x , discussed in details in Section 2.1-2.3 and Section 3.1, respectively. For each fixed grid point in velocity, say v_j , we advect equation (3.23) for half a time step $\Delta t/2$.

- If the SLDG scheme is adopted, there is no CFL stability time step restriction. Therefore, the time stepping for $\Delta t/2$ can be performed by calling $DG_x(\Delta t/2)$.
- If the RKDG scheme is applied, then there is time step restriction due to the linear stability of the algorithm. We denote such the time step restriction as $\delta t(v_j)$, related to the wave propagation speed v_j . In this case, the time stepping for $\Delta t/2$ can be performed by calling $DG_x(\delta t(v_j))$ for several times, until $\Delta t/2$ is reached. Such time stepping strategy maximizes the size of local time steps that can be applied in the RKDG scheme for (3.23) *per* v_j , hence optimizes the time stepping efficiency. Here, by ‘local’, we mean that the size of time stepping can be chosen based on the wave propagation speed v ’s. Much larger time step δt can be taken for smaller v ’s.

Extension of the DG algorithm to multi-dimensional \mathbf{x} can be done either with a rectangular mesh, or with a triangular mesh via a truly multi-dimensional approach, depending on the shape of the spatial domain Ω . However, a nonlinear limiter may

be desirable to robustly resolve the filamentation solution structures of solution of the VP system. We will discuss this issue below.

Advection in Velocity Direction

A conservative high order SLWENO scheme, denoted as $SLWENO_v(\Delta t)$, discussed in Section 3.2, is applied to advect equation (3.24) for a time step Δt . As an SL approach is applied, no time step restriction is involved in this step. Recall that the domain discretization of (x, v) is the tensor product of intervals (finite element discretization in x) and point values (finite difference discretization in v). In order to advect (3.24) in the finite difference framework, we propose to apply one-to-one linear transformations from/to k^{th} degree polynomials $f_{i,j}(x)$ over the interval I_i , to/from $k + 1$ point values of the solution at Gaussian quadrature nodes on I_i at v_j , denoted as $\{f_{i,ig;j}\}_{ig=1}^{k+1}$. Below we illustrate the advection procedure (see Figure 3.5).

1. *From polynomials to point values.* For each spatial interval I_i , transform the k^{th} degree polynomial $f_{i,j}(x)$ to $k + 1$ Gaussian point values $\{f_{i,ig;j}\}_{ig=1}^{k+1}$ on I_i per grid point v_j .
2. *SLWENO evolution.* Apply the SLWENO scheme, discussed in details in Section 3.2, to equation (3.24) for $x_{i,ig}$, $\forall i = 1, \dots, N_x$ and $ig = 1, \dots, k + 1$.
3. *From point values to polynomials.* For each spatial interval I_i , transform the $k + 1$ Gaussian quadrature point values $\{f_{i,ig;j}\}_{ig=1}^{k+1}$ back to a k^{th} degree polynomial $f_{i,j}(x)$, on I_i per grid point v_j .

Extension to multi-dimensional \mathbf{v} can be done by dimensional splitting without introducing splitting errors. We remark that one has the flexibility to combine any

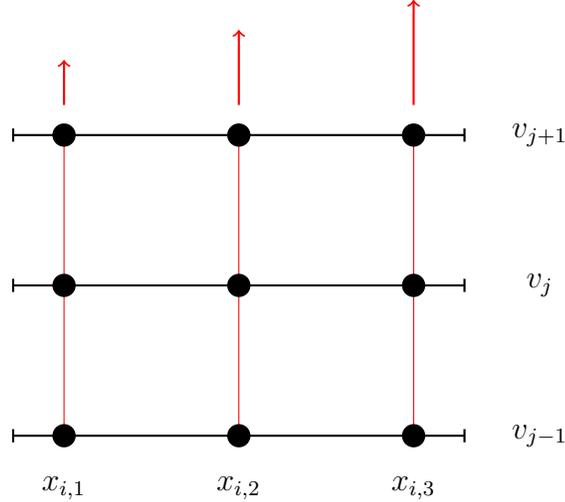


Figure 3.5: Advection in velocity direction by the SLWENO scheme at Gaussian points. Piecewise polynomial space V_h^2 (3 Gaussian points per cell) is used as an example.

order of SLDG/RKDG schemes with any order of SLWENO schemes in the hybrid setting. In the simulations, we adopt the combination of P^1 with WENO3, P^2 with WENO5 and P^3 with WENO9 to better preserve the order of accuracy of DG schemes. We decide to use higher order WENO schemes than DG schemes, since DG schemes have better numerical resolution than WENO schemes based on the same set of used meshes. Specifically, a DG scheme using V_h^k as the approximation space has $k+1$ degrees of freedom per cell compared with a finite difference WENO scheme which only has one point value.

In order to evolve the advection equation (3.24), the electric field E needs to be computed. One can use a LDG scheme to solve the Poisson's equation (1.2). If the partition is uniform and periodic boundary conditions are imposed, an alternative approach is to perform a fast Fourier transform (FFT). Firstly, we evaluate the

3.4. APPLICATIONS TO VLASOV SIMULATIONS

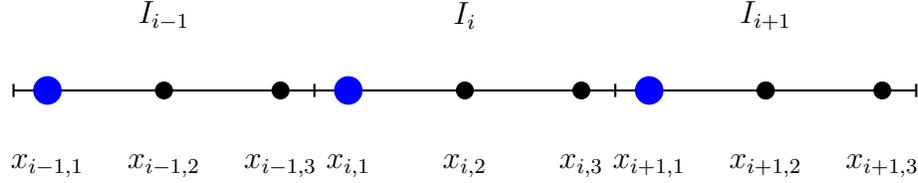


Figure 3.6: Evaluation of electric field by FFT at Gaussian quadrature nodes $\{x_{i,1}\}_{i=1}^{N_x}$. A piecewise polynomial space V_h^2 is used as an example.

charge density at all Gaussian points. Then we group the points as $\{x_{i,ig}\}_{i=1}^{N_x}$, $ig = 1, 2 \dots k + 1$. Note that in each group, the nodes are evenly distributed, see blue circles in Figure 3.6. Therefore, we can conveniently apply FFT in each group to computing the electric field as showed in Figure 3.6. At each time step, FFT will be performed $k + 1$ times.

It is well known that filamentation solution structures may be developed in the VP system. Therefore, nonlinear limiters might be needed for DG schemes in spatial advection in order to control numerical oscillations. We propose to apply the WENO limiter [83] before DG evolution as a pre-processing procedure. In our implementation presented in the next section, we use the TVB limiter with problem dependent TVB constants M to identify troubled cells. The flow chart of the algorithm is outlined below:

Algorithm 3.5. A hybrid method for the VP system:

1. Evolve the solution in spatial directions by a SLDG scheme (or a RKDG scheme with local time stepping) for $\Delta t/2$ by calling $DG_x(\Delta t/2)$.
2. Solve E at all Gaussian points by FFT (or a LDG scheme) from the Poisson's equation.
3. Evolve the solution in velocity directions by SLWENO for Δt by calling $SLWENO_v(\Delta t)$.
4. Apply the WENO limiter as a pre-processing procedure for DG evolution.
5. Evolve the solution as in step 1 for $\Delta t/2$.

Note that step 1 and step 5 can be merged as one time step in order to save computational cost.

In the context of solving the Vlasov equations, the WENO limiter would be useful when the initial data contain discontinuities. Even with smooth initial data, it is well-known that filamentation solution structures might be developed after some time. Such filamentation solution structures will be under-resolved by a given set of meshes. Numerical (rather than physical) oscillations might appear due to the Gibbs phenomenon. In the simulations, the WENO limiter is adopted to control such oscillations and enhance the numerical stability. Notice that one can not hope for accuracy for an under-resolved numerical solution.

3.4. APPLICATIONS TO VLASOV SIMULATIONS

Finally, we remark that the proposed hybrid method is mass conservative, since the RKDG, SLDG and SLWENO schemes are all mass conservative. The mass conservation of a DG scheme can be checked by choosing the test function $v = 1$ in the DG weak formulation (2.3), (3.2), (3.7), and (3.9). The mass conservation of a SLWENO scheme is due to the fact that the equation (3.17) is in a flux difference form. Readers are referred to [84, 85, 86] for detailed analysis.

Proposition 3.1 The proposed hybrid method conserves the total mass subject to periodic boundary conditions.

Proof: It is sufficient to prove that the hybrid method is mass conservative without a WENO limiter. The WENO limiter preserves the cell average, and hence the overall mass. We denote $\bar{f}_{i,j}^n$ as the cell average of the numerical solution $f_{i,j}$ over interval I_i at velocity v_j at time level t^n . Denote $f_{i,j}^{n,(1)}$ as the numerical solution after advection

of Step 1. Denote $f_{i,j}^{n,(2)}$ as the numerical solution after advection in Step 3.

$$\begin{aligned}
 & \sum_j \sum_i \bar{f}_{i,j}^{n+1} \Delta x_i \Delta v \\
 \stackrel{\text{Step 5}}{=} & \sum_j \sum_i \bar{f}_{i,j}^{n,(2)} \Delta x_i \Delta v && \text{(mass conservation of DG)} \\
 = & \sum_j \sum_i \Delta x_i \sum_{ig} f_{i,ig;j}^{n,(2)} w_{ig} \Delta v \\
 = & \sum_i \Delta x_i \sum_{ig} w_{ig} \sum_j f_{i,ig;j}^{n,(2)} \Delta v \\
 \stackrel{\text{Step 3}}{=} & \sum_i \Delta x_i \sum_{ig} w_{ig} \sum_j f_{i,ig;j}^{n,(1)} \Delta v && \text{(mass conservation of SLWENO)} \\
 = & \sum_j \sum_i \Delta x_i \sum_{ig} f_{i,ig;j}^{n,(1)} w_{ig} \Delta v \\
 = & \sum_j \sum_i \bar{f}_{i,j}^{n,(1)} \Delta x_i \Delta v \\
 \stackrel{\text{Step 1}}{=} & \sum_j \sum_i \bar{f}_{i,j}^n \Delta x_i \Delta v. && \text{(mass conservation of DG)}
 \end{aligned}$$

■

Numerical Results

Below, several numerical examples are presented to illustrate high order accuracy and reliability of the proposed hybrid method. We test linear advection and rigid body rotation problems, and the VP system. In the numerical experiments, two types of hybrid methods are considered. In particular, we first perform the dimensional splitting, then (1) a SLDG scheme or (2) a RKDG scheme with local time stepping for advection in x -direction are coupled with a SLWENO scheme for advection in y - or v -direction.

3.4. APPLICATIONS TO VLASOV SIMULATIONS

Example 3.6. Consider a 2-D linear advection equation:

$$u_t + u_x + u_y = 0 \tag{3.25}$$

with an initial condition $u(t = 0, x, y) = \sin(x + y)$ and periodic boundary conditions. The errors and numerical orders of accuracy of the proposed hybrid methods are shown in Table 3.1 for the SLDG schemes combined with the SLWENO schemes, and in Table 4.2 for the RKDG schemes combined with the SLWENO schemes. Since the x -shifting and y -shifting operators commute, there is no splitting error in time and the spatial error will dominate. We remark that the order of used WENO reconstruction in y -direction is higher than that of a DG scheme in x -direction. Such choice of combination makes the DG error dominant. Expected orders of accuracy from DG schemes are observed in Table 3.1 and 4.2. Comparable numerical results are observed for coupling SLDG schemes or RKDG schemes in the hybrid framework.

Table 3.1: Linear advection. Hybrid methods with SLDG schemes and SLWENO schemes. L^2 errors and numerical orders of accuracy on uniform meshes with $N_x \times N_y$ cells. TVB constant $M= 1.0$. CFL=2.2. T=10.

	$P^1+WENO3$		$P^2+WENO5$		$P^3+WENO9$	
mesh	L^2 error	order	L^2 error	order	L^2 error	order
40×40	5.58E-2	–	8.67E-5	–	6.63E-7	–
80×80	1.87E-2	1.57	1.04E-5	3.06	4.05E-8	4.04
120×120	9.03E-3	1.79	3.12E-6	2.97	8.73E-9	3.78
160×160	4.96E-3	2.08	1.41E-6	2.75	2.77E-9	3.99
200×200	2.88E-3	2.44	6.81E-7	3.27	1.17E-9	3.87

3.4. APPLICATIONS TO VLASOV SIMULATIONS

Table 3.2: Linear advection. Hybrid methods with RKDG schemes and SLWENO schemes. L^2 errors and numerical orders of accuracy on uniform meshes with $N_x \times N_y$ cells. TVB constant $M=1.0$. $CFL=2.2$. $T=10$.

	$P^1+WENO3$		$P^2+WENO5$		$P^3+WENO9$	
mesh	L^2 error	order	L^2 error	order	L^2 error	order
40×40	5.04E-2	–	1.34E-4	–	8.10E-7	–
80×80	1.48E-2	1.77	1.36E-5	3.30	5.07E-8	4.00
120×120	7.03E-3	1.83	3.89E-6	3.09	1.00E-8	4.00
160×160	3.95E-3	2.00	1.62E-6	3.04	3.17E-9	4.00
200×200	2.40E-3	2.24	8.25E-7	3.03	1.30E-9	4.00

Example 3.7. Consider the solid body rotation problem:

$$u_t - yu_x + xu_y = 0, \quad (x, y) \in [-2\pi, 2\pi]^2. \quad (3.26)$$

We first consider a smooth initial condition

$$u(t=0, x, y) = \begin{cases} \cos^6\left(\frac{r}{2}\right) & r < \pi, \\ 0 & \text{otherwise,} \end{cases} \quad (3.27)$$

where $r = \sqrt{0.8(x-1)^2 + 1.2y^2}$. This is to test orders of convergence of the hybrid schemes. Again, we report the L^2 error and the orders of accuracy for two hybrid methods: SLDG schemes combined with SLWENO schemes in Table 3.3 and Table 3.4, RKDG schemes combined with SLWENO schemes in Table 3.5 and Table 3.6 for $CFL=0.3$ and $CFL=6.2$, respectively. The spatial error is observed to be the dominant error when $CFL=0.3$, see Table 3.3 and 3.5. The second order splitting error in time becomes the dominant error when a relatively large CFL number, i.e., $CFL=6.2$, is used, see Table 3.4 and 3.6. We remark that there is a certain range of

3.4. APPLICATIONS TO VLASOV SIMULATIONS

Table 3.3: Solid body rotation. Hybrid methods with SLDG schemes and SLWENO schemes. L^2 errors and numerical orders of accuracy on uniform meshes with $N_x \times N_y$ cells. TVB constant $M=1.0$. $CFL=0.3$. $T=\pi/2$.

	$P^1+WENO3$		$P^2+WENO5$		$P^3+WENO9$	
mesh	L^2 error	order	L^2 error	order	L^2 error	order
10×30	2.57E-01	–	6.09E-02	–	5.36E-03	–
20×60	7.33E-02	1.81	6.63E-03	3.20	3.41E-04	3.97
30×90	3.28E-02	1.98	1.90E-03	3.09	6.44E-05	4.11
40×120	1.88E-02	1.94	7.92E-04	3.03	2.01E-05	4.04
50×150	1.24E-02	1.88	4.15E-04	2.89	8.35E-06	3.95

Table 3.4: Solid body rotation. Hybrid methods with SLDG schemes and SLWENO schemes. L^2 errors and numerical orders of accuracy on uniform meshes with $N_x \times N_y$ cells. TVB constant $M=1.0$. $CFL=6.2$. $T=\pi/2$

	$P^1+WENO3$		$P^2+WENO5$		$P^3+WENO9$	
mesh	L^2 error	order	L^2 error	order	L^2 error	order
10×30	1.87E-01	–	5.37E-02	–	2.67E-02	–
20×60	5.57E-02	1.75	8.03E-03	2.74	6.46E-03	2.05
30×90	2.53E-02	1.94	3.12E-03	2.33	2.87E-03	2.01
40×120	1.46E-02	1.92	1.66E-03	2.20	1.61E-03	2.00
50×150	9.71E-03	1.82	1.03E-03	2.12	1.03E-03	2.00

CFL numbers above which the splitting error in time is the dominant error. Such range is problem and scheme dependent. An attempt to increase the temporal order accuracy will be discussed in Subsection 3.4.2. Note that we set $3N_x = N_y$ in order to observe a clean order of convergence from DG schemes.

We then consider an initial condition which includes a slotted disk, a cone as well as a smooth hump, similar to the one used in [66] to test the proposed schemes' ability in resolving complicated structures. In Figure 3.7 we plot the initial condition

3.4. APPLICATIONS TO VLASOV SIMULATIONS

Table 3.5: Solid body rotation. Hybrid methods with RKDG schemes and SLWENO schemes. L^2 errors and numerical orders of accuracy on uniform meshes with $N_x \times N_y$ cells. TVB constant $M=1.0$. CFL=0.3. $T=\pi/2$.

	$P^1+WENO3$		$P^2+WENO5$		$P^3+WENO9$	
mesh	L^2 error	order	L^2 error	order	L^2 error	order
10×30	2.71E-01	–	6.75E-02	–	6.85E-03	–
20×60	7.87E-02	1.78	7.38E-03	3.19	6.15E-04	3.48
30×90	3.53E-02	1.98	2.14E-03	3.06	1.21E-04	4.01
40×120	2.00E-02	1.98	8.96E-04	3.02	3.83E-05	4.00
50×150	1.29E-02	1.95	4.66E-04	2.93	1.59E-05	3.94

Table 3.6: Solid body rotation. Hybrid methods with RKDG schemes and SLWENO schemes. L^2 errors and numerical orders of accuracy on uniform meshes with $N_x \times N_y$ cells. TVB constant $M=1.0$. CFL= 6.2. $T=\pi/2$

	$P^1+WENO3$		$P^2+WENO5$		$P^3+WENO9$	
mesh	L^2 error	order	L^2 error	order	L^2 error	order
10×30	2.34E-01	–	6.57E-02	–	2.49E-02	–
20×60	6.92E-02	1.76	9.04E-03	2.86	6.15E-03	2.01
30×90	3.10E-02	1.98	3.36E-03	2.44	2.72E-03	2.01
40×120	1.76E-02	1.96	1.74E-03	2.29	1.53E-03	2.00
50×150	1.14E-02	1.95	1.06E-03	2.20	9.77E-04	2.00

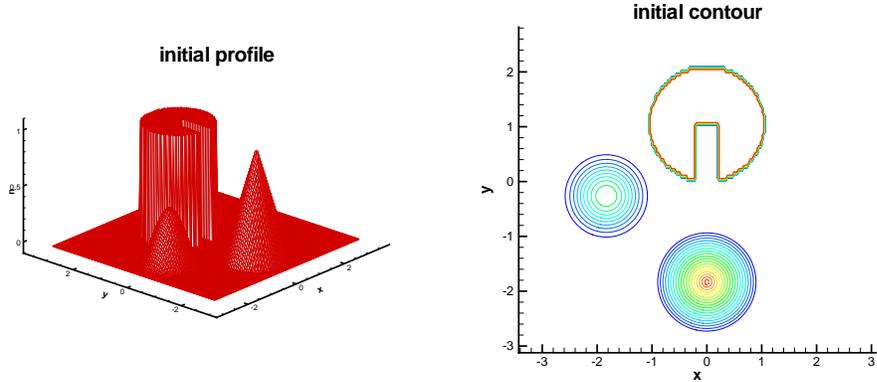


Figure 3.7: Initial condition. Profile (left) and contour (right).

in mesh and contour. The numerical solution at $T=12\pi$ is reported in Figure 3.8, which returns to the initial state after six full evolutions. Non-oscillatory numerical capturing of discontinuities is observed. It is clear that higher order methods resolve the solution structure better. As the performance of the hybrid schemes using RKDG schemes in x -direction is very similar to those in Figure 3.8, we omit them for brevity.

Below, we further examine the performance of the proposed hybrid methods when applied to the VP systems. Periodic boundary conditions are imposed in x -direction, while zero boundary conditions are imposed in v -direction. We only present the numerical results of the hybrid methods with SLDG schemes in x -direction and SLWENO schemes in v -direction, as the performance of the scheme using RKDG schemes for spatial advection is very similar. In realistic high-dimensional problems with complicated spatial domains, the RKDG schemes will offer more flexibility than the SLDG counterparts.

Throughout this section, we consider solving the VP system with the following

3.4. APPLICATIONS TO VLASOV SIMULATIONS

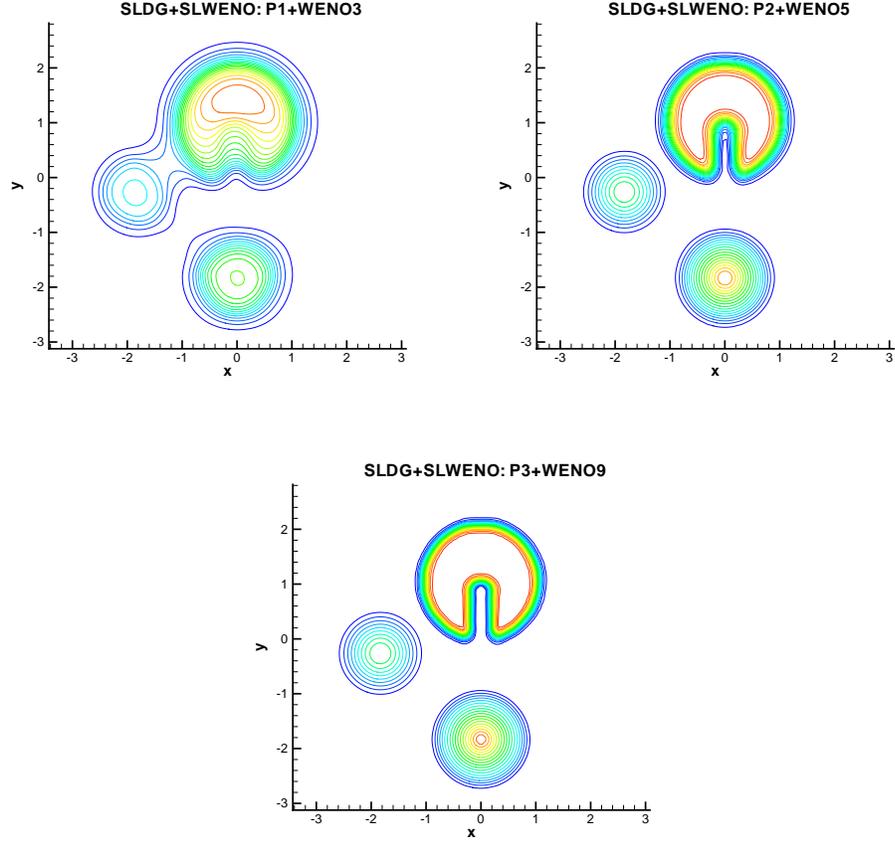


Figure 3.8: Solid body rotation with the initial condition in Figure 3.7. Uniform meshes with $N_x \times N_y = 100 \times 100$. TVB constant $M= 1.0$. $CFL=2.2$. $T=12\pi$. Hybrid schemes: SLDG schemes combined with SLWENO schemes.

four initial conditions.

- Landau damping:

$$f(t = 0, x, v) = \frac{1}{\sqrt{2\pi}} (1 + \alpha \cos(kx)) \exp\left(-\frac{v^2}{2}\right), \quad (3.28)$$

where $\alpha = 0.01$ for the weak case and $\alpha = 0.5$ for the strong case.

- Two stream instability I:

$$\begin{aligned}
 f(t = 0, x, v) & \tag{3.29} \\
 &= \frac{2}{7\sqrt{2\pi}}(1 + 5v^2) (1 + \alpha ((\cos(2kx) + \cos(3kx)) / 1.2 + \cos(kx))) \exp\left(-\frac{v^2}{2}\right),
 \end{aligned}$$

where $\alpha = 0.01$, $k = 0.5$.

- Two stream instability II:

$$f(t = 0, x, v) = \frac{1}{\sqrt{2\pi}}(1 + \alpha \cos(kx))v^2 \exp\left(-\frac{v^2}{2}\right), \tag{3.30}$$

where $\alpha = 0.05$, $k = 0.5$.

- Two stream instability III:

$$\begin{aligned}
 f(t = 0, x, v) & \tag{3.31} \\
 &= \frac{1}{2v_{th}\sqrt{2\pi}} \left[\exp\left(-\frac{(v-u)^2}{2v_{th}^2}\right) + \exp\left(-\frac{(v+u)^2}{2v_{th}^2}\right) \right] (1 + \alpha \cos(kx)).
 \end{aligned}$$

Example 3.8. We first present the performance of the proposed hybrid scheme for two stream instability I (3.29). In this example, we will demonstrate (1) the high order spatial accuracy and the second order temporal accuracy of the proposed scheme; (2) the performance of the proposed hybrid method in resolving solutions and in preserving theoretically conserved physical norms; (3) the robustness of the SLWENO scheme for velocities compared with a spectral method and a pure SLDG scheme. The length of the domain in the x -direction is $L = \frac{2\pi}{k}$.

3.4. APPLICATIONS TO VLASOV SIMULATIONS

Firstly, we want to test spatial and temporal accuracy of the hybrid methods. To minimize the error from truncating the domain in v -direction, we let $v_{max} = 2\pi$. We compute a reference numerical solution with a very fine mesh. We remark that the overall numerical error consists of two parts: spatial and temporal errors. To test spatial accuracy of DG in x -direction, we make the mesh in v -direction to be fine enough and time step to be sufficiently small, so that spatial error dominates. We report the L^2 error and the orders of accuracy for two hybrid methods: the SLDG schemes combined with the SLWENO schemes in Table 3.7 and RKDG schemes combined with SLWENO schemes in Table 3.8. Expected $k + 1$ orders of spatial accuracy are observed. To test temporal accuracy, there is a lower bound in the time step size that we need to respect. When the time step is too small, spatial error becomes dominant. In this case, we let $\Delta t = \Delta x$ to test the temporal accuracy. In Table 3.9 we report the L^2 error and the orders of accuracy for two hybrid methods: the SLDG/RKDG P^3 combined with the SLWENO9. Expected second order temporal accuracy due to the Strang splitting is observed. Note that the errors from two hybrid methods are comparable, owing to the fact that splitting error dominates for using a large time step.

Then we test the reliability of the hybrid methods after a long time integration with $v_{max} = 5$ and $\Delta t = \Delta x$. In Figure 3.9, we show the numerical results of hybrid methods at $T = 53$. The filamentation solution structures are well resolved by the hybrid methods. Time evolution of discrete L^1 norm, L^2 norm, energy and entropy by hybrid methods with different orders are reported in Figure 3.10. As expected, the physical quantities, which are conserved in the continuous VP system,

3.4. APPLICATIONS TO VLASOV SIMULATIONS

are better preserved by a higher order hybrid method. In Figures 3.11 we show 3-D plots of the distribution function f at different instances of time. Good resolutions can be observed without oscillations. Our results are comparable to those that have been reported by Filbet and Sonnendrücker in [38] where the performance of several numerical schemes are compared. Since a DG method is used in x -direction, a non-uniform meshes is allowed. Figure 3.12 presents the mesh and the numerical solution of a hybrid method (a RKDG scheme with P^3 in x -direction and a SLWENO9 scheme in v -direction) at $T= 53$. Here we use a non-uniform mesh from 20% random perturbation of the uniform mesh in x -direction with $N_x \times N_v = 64 \times 160$. Due to the non-uniform distribution of the mesh, we use a LDG scheme instead of FFT to solve the Poisson's equation.

An alternative way of advection for the velocity ac/deceleration is using a spectral method. The numerical results of a scheme coupling a SLDG scheme in x -direction and a spectral method in v -direction are presented in Figure 3.13 at different times. When the structure of the solution is simple (see the left panel in Figure 3.13), the spectral method performs well. However, as the numerical mesh no longer supports the filamentation solution structures which are developed over a long time period, serious spurious oscillations are observed (see the right panel of Figure 3.13).

Finally, we use this example to draw a comparison between the method using in both spatial and velocity directions and the proposed hybrid method. We set $N_x \times N_v = 64 \times 128$ for a pure SLDG scheme (P^3 polynomial in both x - and v -directions) and $N_x \times N_v = 64 \times 512$ for the hybrid method (a SLDG scheme with P^3 in x -direction and SLWENO9 scheme in v -direction) for a comparable

3.4. APPLICATIONS TO VLASOV SIMULATIONS

Table 3.7: Two stream instability I. Hybrid method with SL DG and SL WENO. L^2 errors and numerical orders of accuracy on uniform meshes with $N_x \times N_y$ cells. $\Delta t = 0.005\Delta x$. $T=0.5$. The reference solution is computed with $N_x \times N_y = 192 \times 630$.

	$P^2+WENO5$		$P^3+WENO9$	
mesh	L^2 error	order	L^2 error	order
16×210	1.49E-04	–	1.05E-05	–
32×210	1.85E-05	3.01	6.55E-07	4.00
48×210	5.50E-06	3.00	1.29E-07	4.00
64×210	2.31E-06	3.02	4.17E-08	3.94

Table 3.8: Two stream instability I. Hybrid methods with RKDG schemes and SLWENO schemes. L^2 errors and numerical orders of accuracy on uniform meshes with $N_x \times N_y$ cells. $\Delta t = 0.005\Delta x$. $T=0.5$. The reference solution is computed with $N_x \times N_y = 192 \times 630$

	$P^2+WENO5$		$P^3+WENO9$	
mesh	L^2 error	order	L^2 error	order
16×210	1.54E-04	–	1.06E-05	–
32×210	1.92E-05	3.01	6.64E-07	4.00
48×210	5.69E-06	2.99	1.31E-07	4.00
64×210	2.39E-06	3.02	4.22E-08	3.94

resolution. Figure 3.14 gives 3-D plots of the numerical solution in the region where filamentation structures are developed. Milder numerical sawtooth-shaped oscillations are observed for the hybrid method than that for the pure SLDG scheme in both x - and v -directions, when the numerical solution is under-resolved.

Example 3.9. Consider weak Landau damping (3.28) with $\alpha = 0.01$. When the perturbation parameter α is small enough, the VP system can be approximated by linearization around the Maxwellian equilibrium $f^0(v) = \frac{1}{\sqrt{2\pi}}e^{-\frac{v^2}{2}}$. The analytical damping rate of the electric field can be derived accordingly [41]. We test the hybrid

3.4. APPLICATIONS TO VLASOV SIMULATIONS

Table 3.9: Two stream instability I. Hybrid methods with the SLDG/RKDG P^3 schemes and the SLWENO9 scheme. L^2 errors and numerical orders of accuracy on uniform meshes with $N_x \times N_y$ cells. The reference is computed with $N_x \times N_v = 630 \times 630$. $\Delta t = \Delta x$. $T=1$.

	SLDG P^3 +SLWENO9		RKDG P^3 +SLWNEO9	
mesh	L^2 error	order	L^2 error	order
70×70	3.23E-04	–	3.23E-04	–
90×90	1.93E-04	2.05	1.93E-04	2.05
126×126	9.59E-05	2.08	9.59E-05	2.08
210×210	3.20E-05	2.15	3.20E-05	2.15

scheme with different k 's and compare the numerical damping rates with theoretical values. In the simulations, we set $N_x = 64$, $N_v = 160$, $v_{max} = 5$, and $\Delta t = \Delta x$. We plot the evolution of the electric field in L^2 norm in Figure 3.15 for $k = 0.5$, $k = 0.4$ and $k = 0.3$. The correct decay rates of the electric field are observed, benchmarked with all the theoretical values (solid black lines in the figure). The time evolution of discrete L^1 norm, L^2 norm, energy and entropy by hybrid methods with different orders are reported in Figure 3.16. For brevity, we only report the result when $k = 0.5$. In this case, the total mass is observed to be exactly conserved. Other physical quantities are well preserved.

Example 3.10. Consider strong Landau damping (3.28) with $\alpha = 0.5$ and $k = 0.5$. In Figure 3.17, the evolution of L^2 norms of the electric field is provided. The linear decay rate of different orders of hybrid methods are all approximately $\gamma_1 = -0.2812$, which is identical to the value computed by Cheng and Knorr [13]. It is a little smaller than the value of -0.292 reported by Rossmannith and Seal in [97] and -0.287 reported by Heath et. al in [53]. We also compute the growth rates of hybrid

3.4. APPLICATIONS TO VLASOV SIMULATIONS

methods, which are approximately $\gamma_2 = 0.0813$ of the SLDG P^3 scheme combined with the SLWENO9 schemes, $\gamma_2 = 0.0778$ of the SLDG P^2 scheme combined with the SLWENO5 scheme and $\gamma_2 = 0.0770$ of the SLDG P^1 scheme combined with the SLWENO3 scheme. They are all consistent with the value of 0.0815 computed by Rossmannith and Seal in [97] and 0.0746 by Heath et al. in [53]. Numerical solutions of the hybrid methods with different orders at $T=30$ are plotted in Figure 3.18. Better resolutions are observed with higher order methods. Figure 3.19 gives the numerical solution of strong Landau damping at different instances of time using hybrid methods of the SLDG P^3 scheme combined with the SLWENO9 scheme, the results are comparable to what are shown in [84]. The time evolution of discrete L^1 norms, L^2 norm, energy and entropy by hybrid methods with different orders are reported in Figure 3.20. It is clear that higher order hybrid methods better preserve physical quantities than lower order methods, as expected. We remark that because the proposed hybrid methods are not positivity preserving, L^1 norms of numerical solutions are not exactly preserved, although the methods are mass conservative.

Example 3.11. Consider the two stream instability III. The background ion distribution function is fixed, uniform, and chosen so that the total net charge density for the system is zero. We first let $\alpha = 0.001$, $u = 2.4$, $v_{th} = 1$, and $k = 0.2$. The linear growth rate of electric field after some time is 0.2258, which can be derived by the same procedure as the Landau damping. In Figure 3.21, we plot the evolution of the electric field in the L^2 norm. The correct growth rates of the electric field are observed, benchmarked with the theoretical value. We then let $\alpha = 0.05$, $u = 0.99$, $v_{th} = 0.3$ and $k = \frac{2}{13}$. This case is studied in [86, 87] by SLWENO schemes

3.4. APPLICATIONS TO VLASOV SIMULATIONS

and SLDG schemes, respectively. In Figures 3.22, we report the numerical results approximating the distribution solution f . Time evolution of discrete L^1 norm, L^2 norm, energy and entropy by hybrid methods with different orders are reported in Figure 3.23. Again, the higher order methods in general do a better job in resolving filamentation structures and preserving the physical quantities than low order ones. Finally, we remark that with strong perturbation, nonlinear effects of higher modes become dominant. The growth rate of the electric field does not agree with the theoretical value obtained in the linear analysis. Thus we omit to present the evolution of the electric field.

3.4. APPLICATIONS TO VLASOV SIMULATIONS

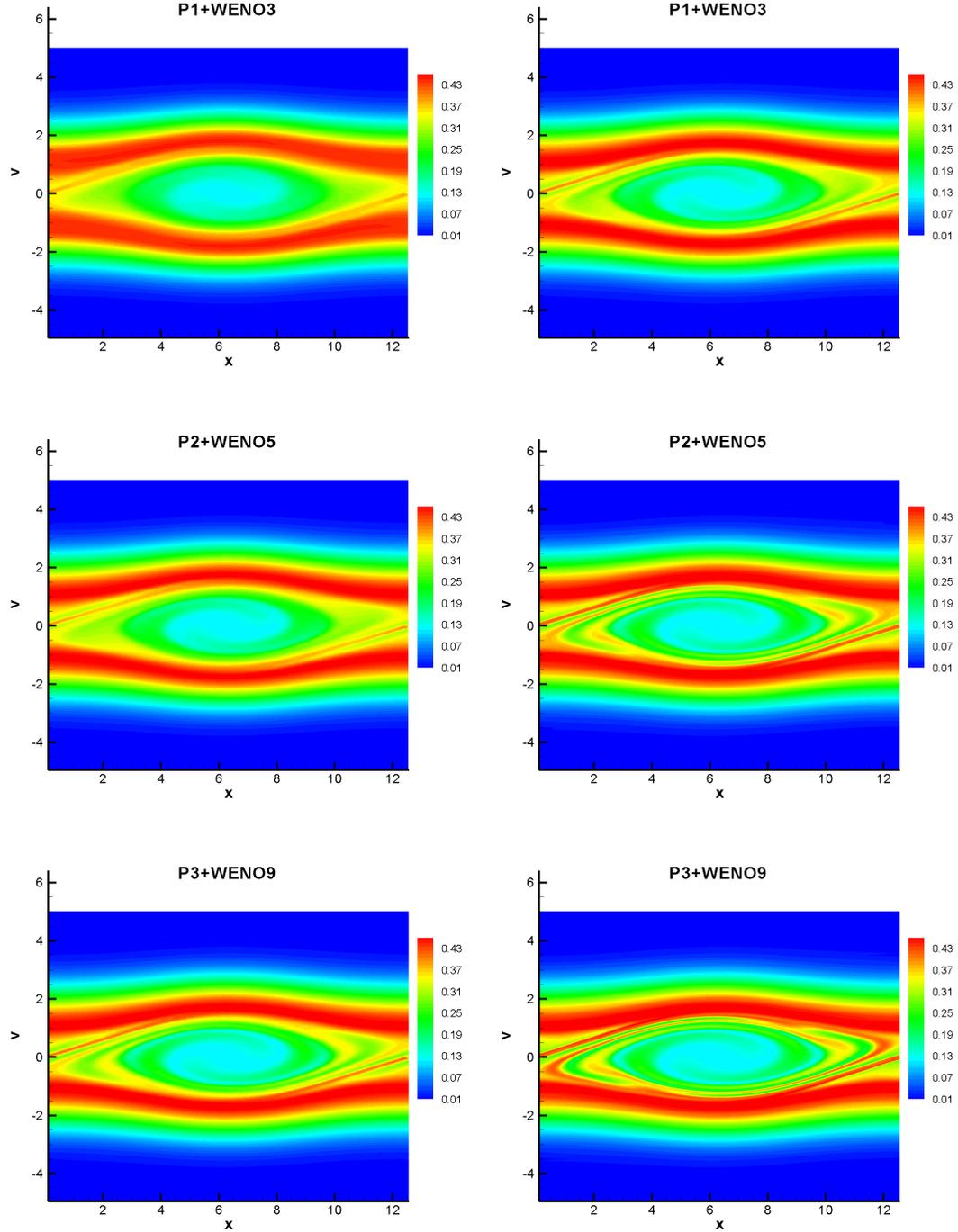


Figure 3.9: Two stream instability I. TVB constant $M=1.0$. $T=53$. Hybrid scheme: the SLDG scheme combined with SLWENO scheme. $N_x \times N_v = 64 \times 160$ (left), $N_x \times N_v = 128 \times 320$ (right).

3.4. APPLICATIONS TO VLASOV SIMULATIONS

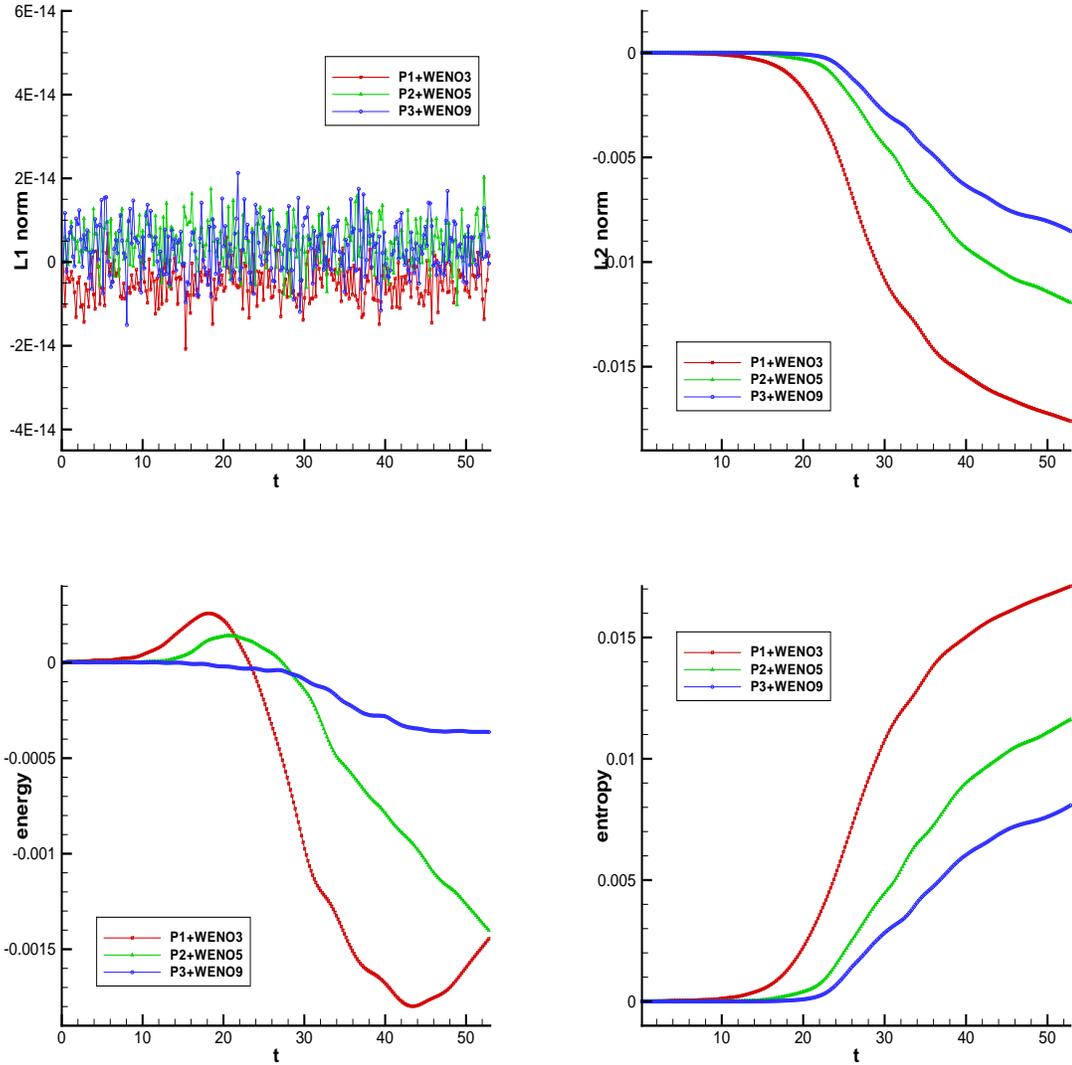


Figure 3.10: Two stream instability I. Time evolution of the relative deviations of discrete L^1 norms (upper left), L^2 norms (upper right), energy (lower left) and entropy (lower right). TVB constant $M= 1.0$. Hybrid scheme: the RKDG scheme combined with the SLWENO scheme.

3.4. APPLICATIONS TO VLASOV SIMULATIONS

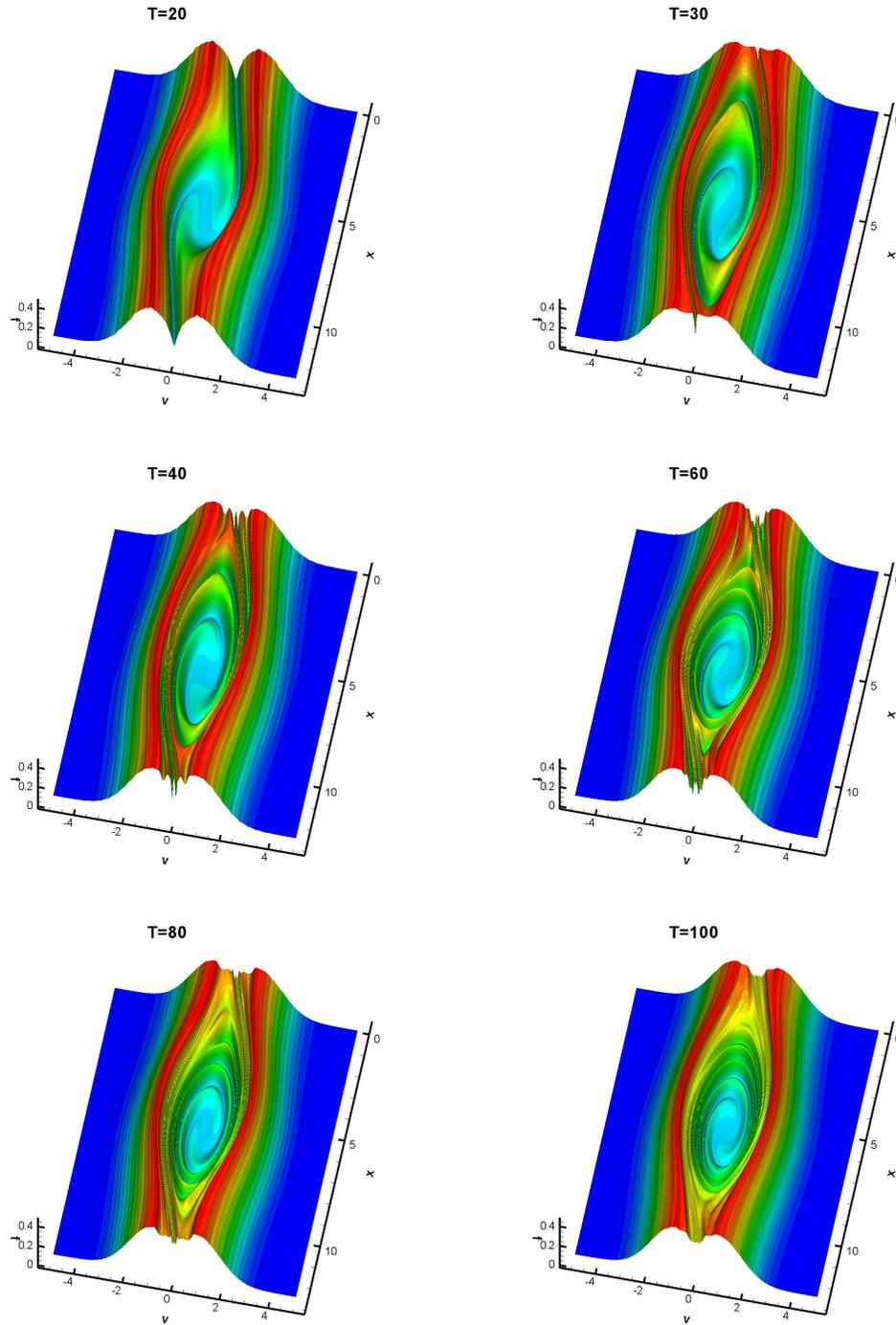


Figure 3.11: Two stream instability I. TVB constant $M=1.0$. $N_x \times N_v = 128 \times 320$. 3-D contour plot of distribution function. $T=20, 30, 40, 60, 80,$ and 100 .

3.4. APPLICATIONS TO VLASOV SIMULATIONS

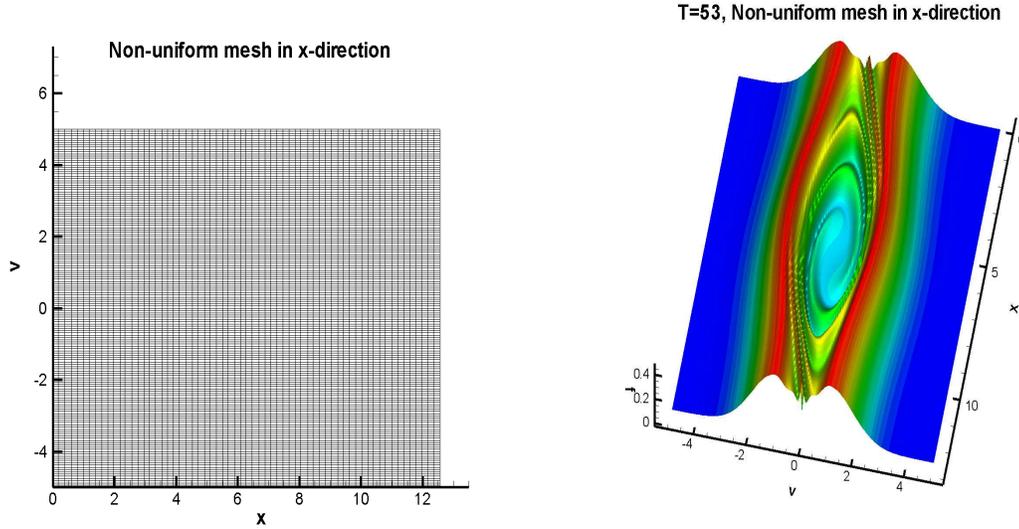


Figure 3.12: Two stream instability I. A non-uniform meshes with 20% random perturbation in x -direction of a uniform mesh $N_x \times N_v = 64 \times 160$. $T = 53$. Hybrid scheme: the RKDG scheme P^3 combined with the SLWENO9 scheme.

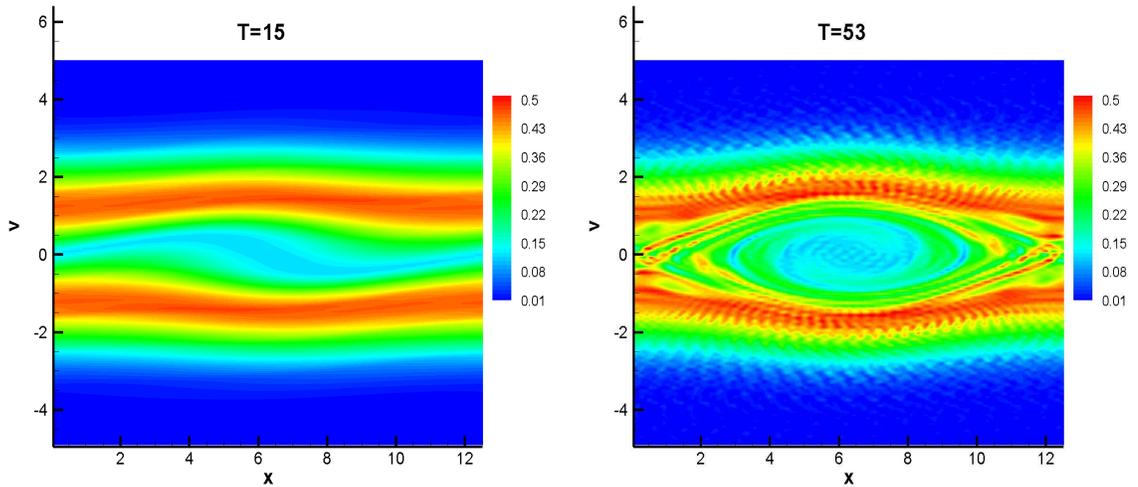


Figure 3.13: Two stream instability I. A spectral method is adopted for velocity ac/deceleration. When $T=15$, the structure of solution is simple, the performance of the spectral method is acceptable (left); when $T=53$, the filamentation structures are generated, serious oscillations appear (right).

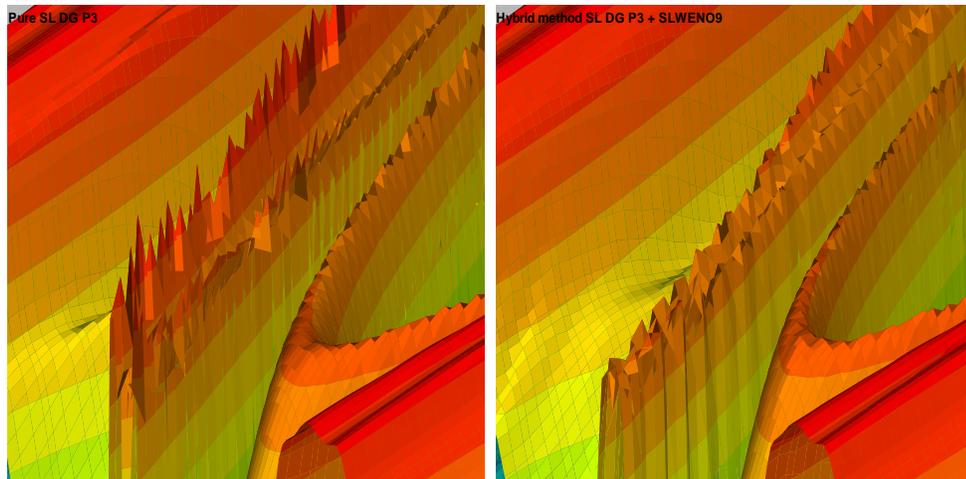


Figure 3.14: Two stream instability I. Comparison between the pure SLDG scheme and the hybrid method. $T=53$. The pure SLDG P^3 scheme (left), the hybrid method combined SLDG P^3 scheme and SLWENO9 scheme (right). Zoomed-in region to show more details. Much fewer artificial oscillations are observed of the hybrid method.

3.4. APPLICATIONS TO VLASOV SIMULATIONS

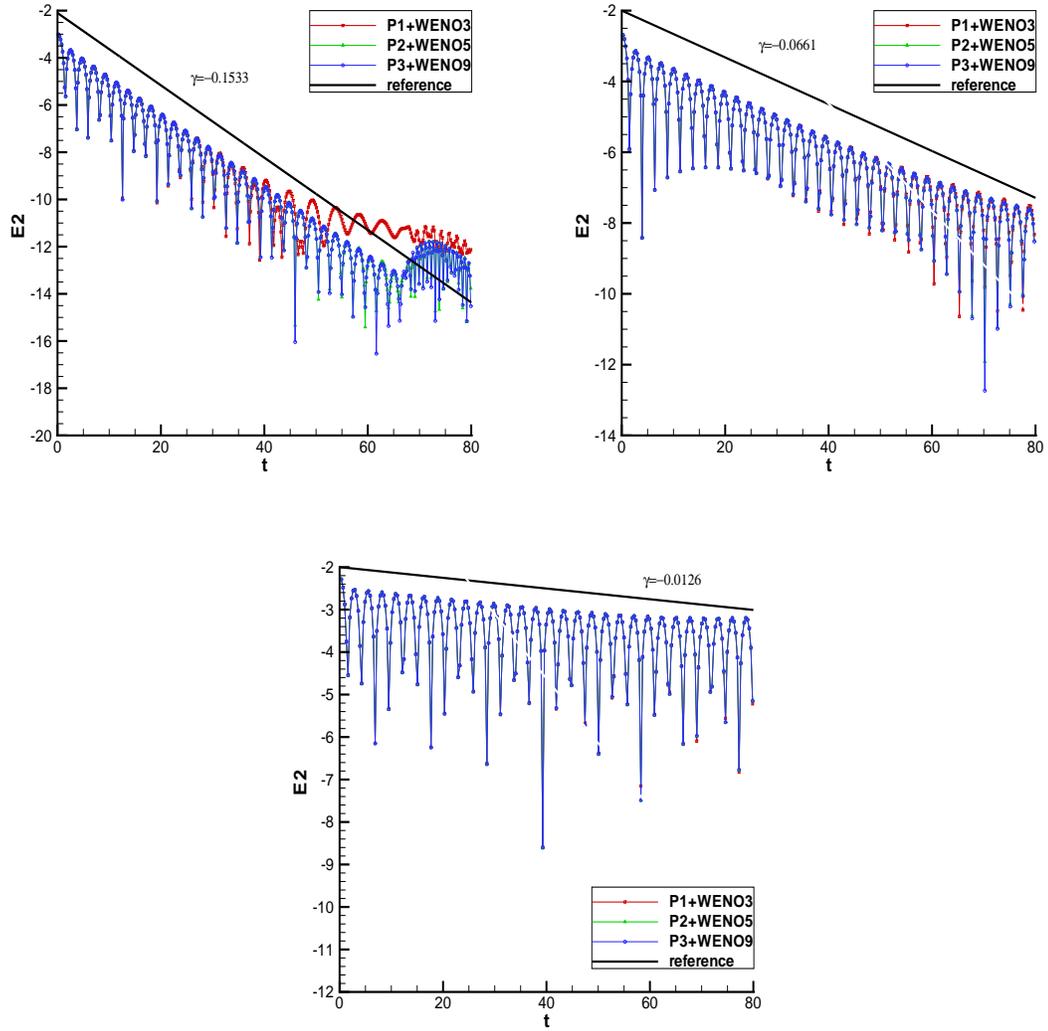


Figure 3.15: Weak Landau damping. Time evolution of the electric field in the L^2 norm. $k = 0.5$ (upper left), $k = 0.4$ (upper right), and $k = 0.3$ (lower)

3.4. APPLICATIONS TO VLASOV SIMULATIONS

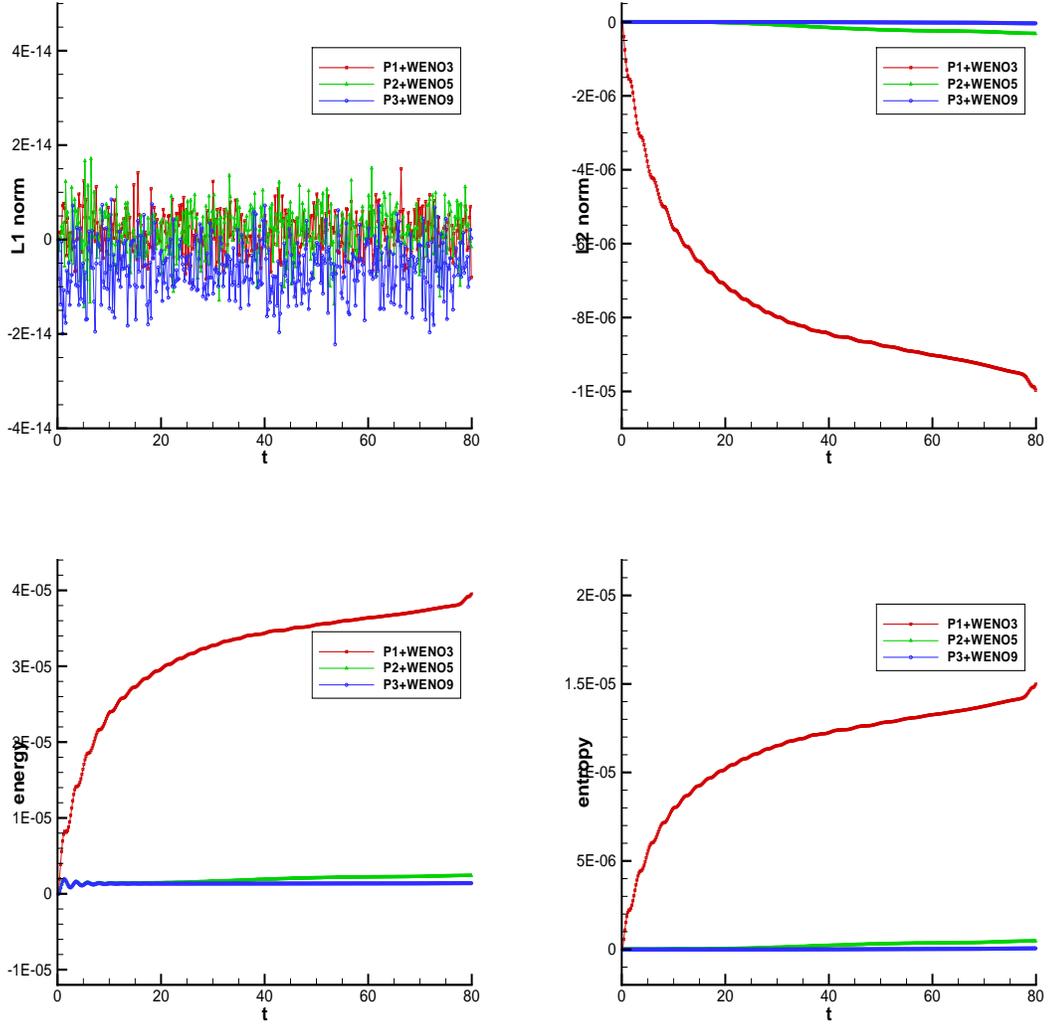


Figure 3.16: Weak Landau damping. Time evolution of the relative deviations of discrete L^1 norms (upper left), L^2 norms, energy (lower left), and entropy (lower right). TVB constant $M= 1.0$. Hybrid schemes: SLDG schemes combined with SLWENO schemes.

3.4. APPLICATIONS TO VLASOV SIMULATIONS

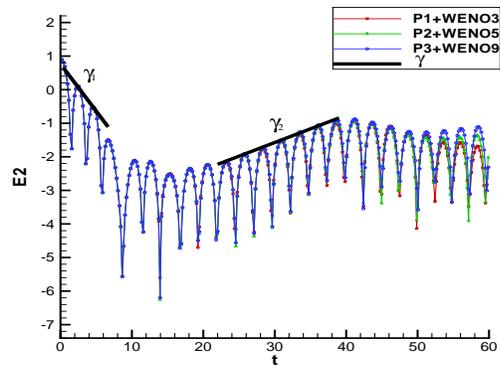


Figure 3.17: Strong Landau damping. Time evolution of electric field in L^2 norm.

3.4. APPLICATIONS TO VLASOV SIMULATIONS

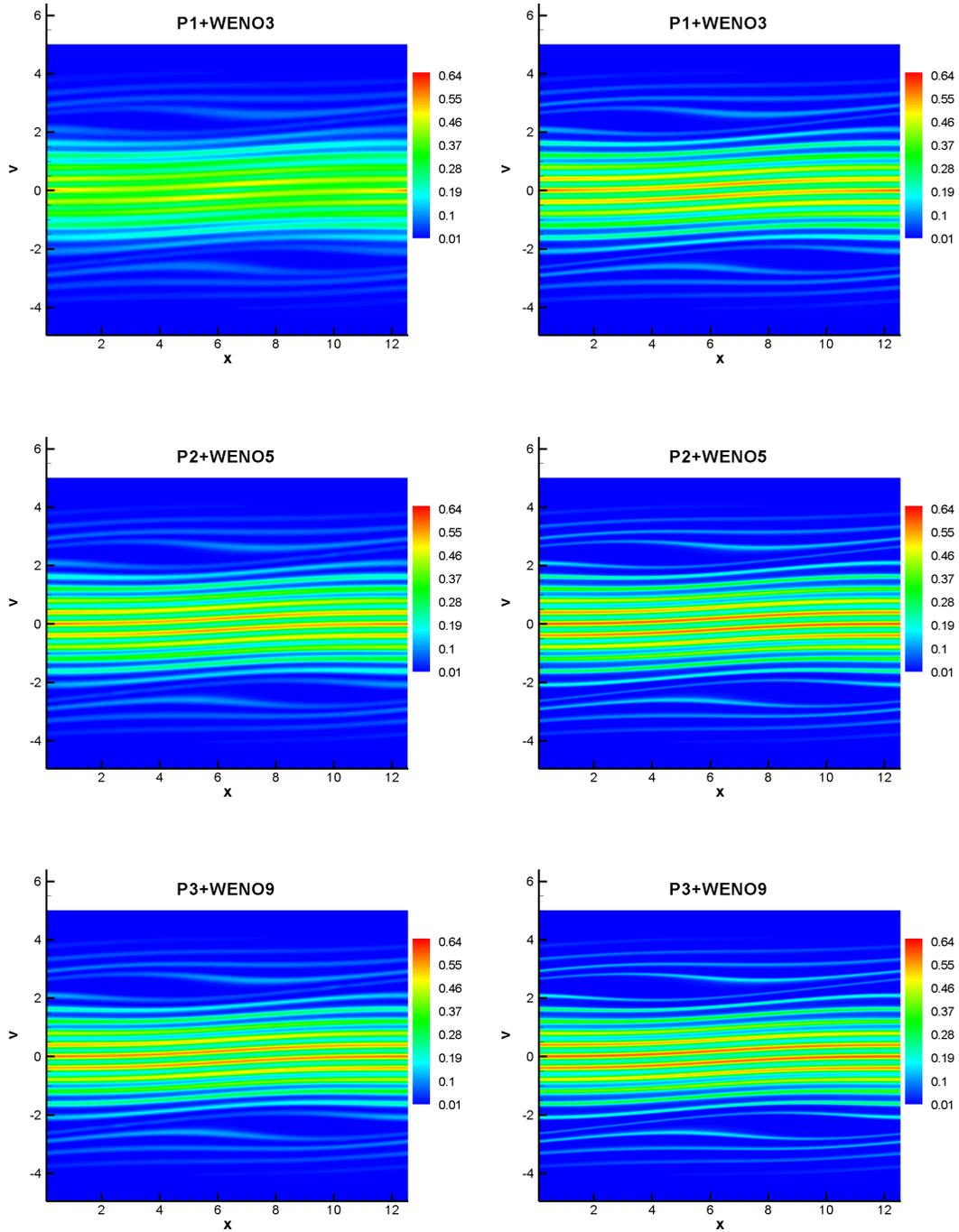


Figure 3.18: Strong Landau damping. TVB constant $M= 1.0$. $T=30$. Hybrid scheme: the SLDG scheme combined with the SLWENO scheme. $N_x \times N_v = 64 \times 160$ (left), $N_x \times N_v = 128 \times 320$ (right).

3.4. APPLICATIONS TO VLASOV SIMULATIONS

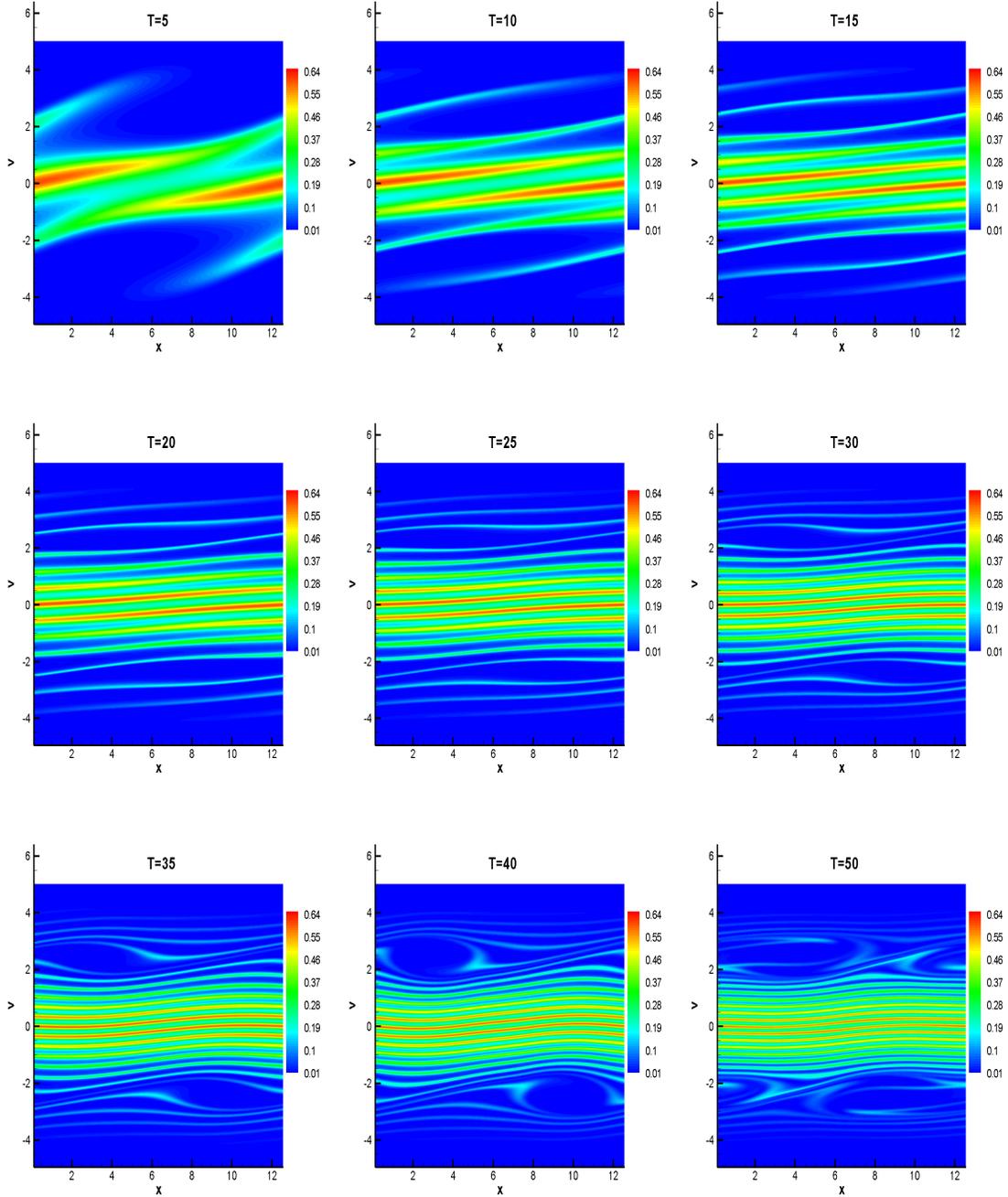


Figure 3.19: Strong Landau damping. TVB constant $M=1.0$. Time evolution of numerical solutions at $T=5, 10, 15, 20, 25, 30, 35, 40,$ and 50 . Hybrid scheme of the SLDG P^3 scheme combined with the SLWENO9 scheme. $N_x \times N_v = 128 \times 320$.

3.4. APPLICATIONS TO VLASOV SIMULATIONS

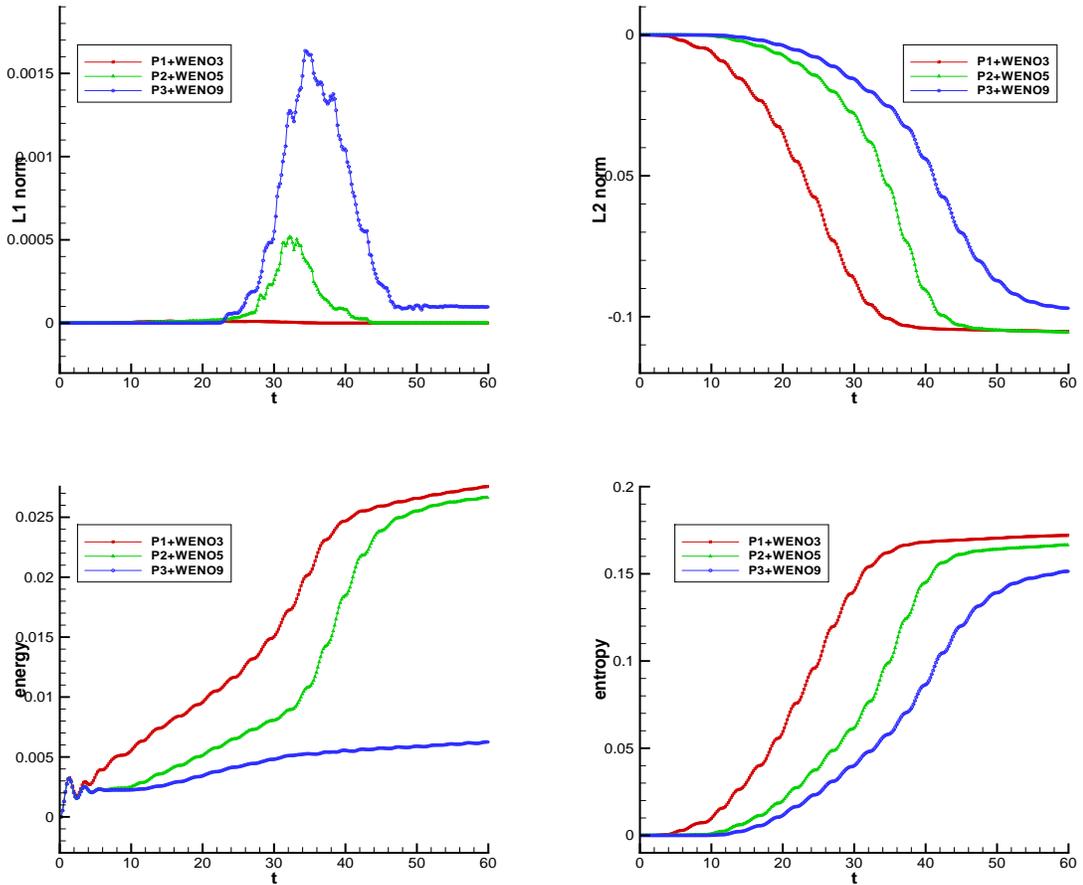


Figure 3.20: Strong Landau damping. Time evolution of the relative deviations of discrete L^1 norms (upper left), L^2 norms, energy (lower left), and entropy (lower right). TVB constant $M=1.0$. Hybrid scheme: the SLDG scheme combined with the SLWENO scheme.

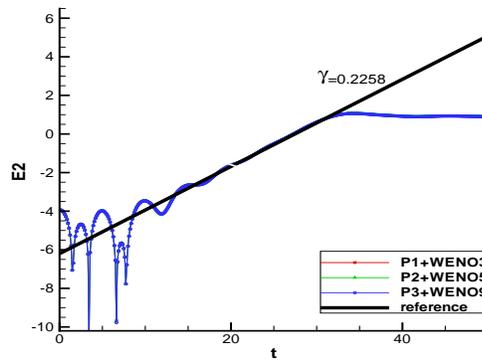


Figure 3.21: Two stream instability III. Time evolution of the electric field in the L^2 norm. $\alpha = 0.001$, $u = 2.4$, $v_{th} = 1$, and $k = 0.2$.

3.4. APPLICATIONS TO VLASOV SIMULATIONS

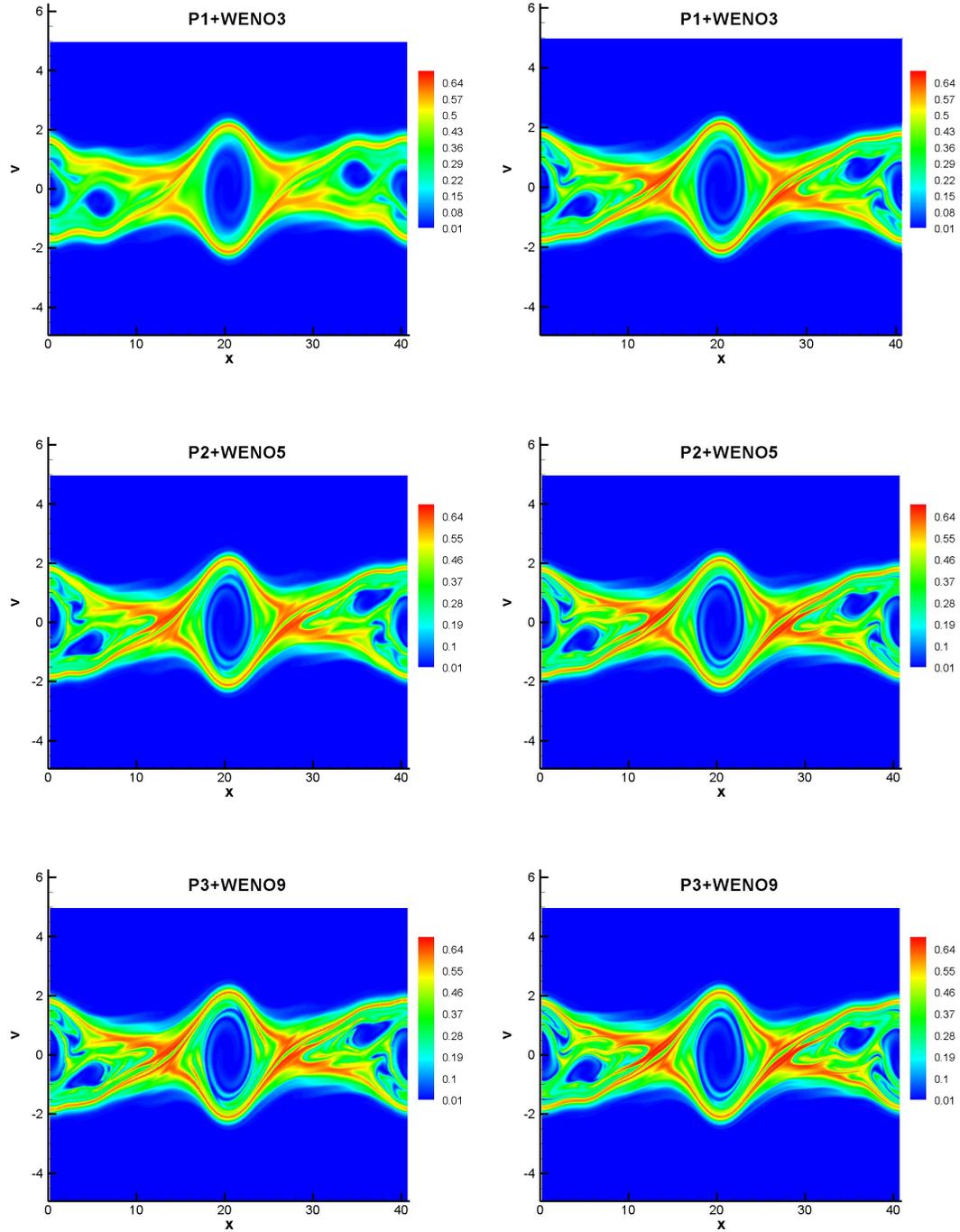


Figure 3.22: Two stream instability III. TVB constant $M= 3.0$. $T=70$. Hybrid scheme: the SLDG schemes combined with the SLWENO schemes. $N_x \times N_v = 128 \times 320$ (left), $N_x \times N_v = 256 \times 640$ (right).

3.4. APPLICATIONS TO VLASOV SIMULATIONS

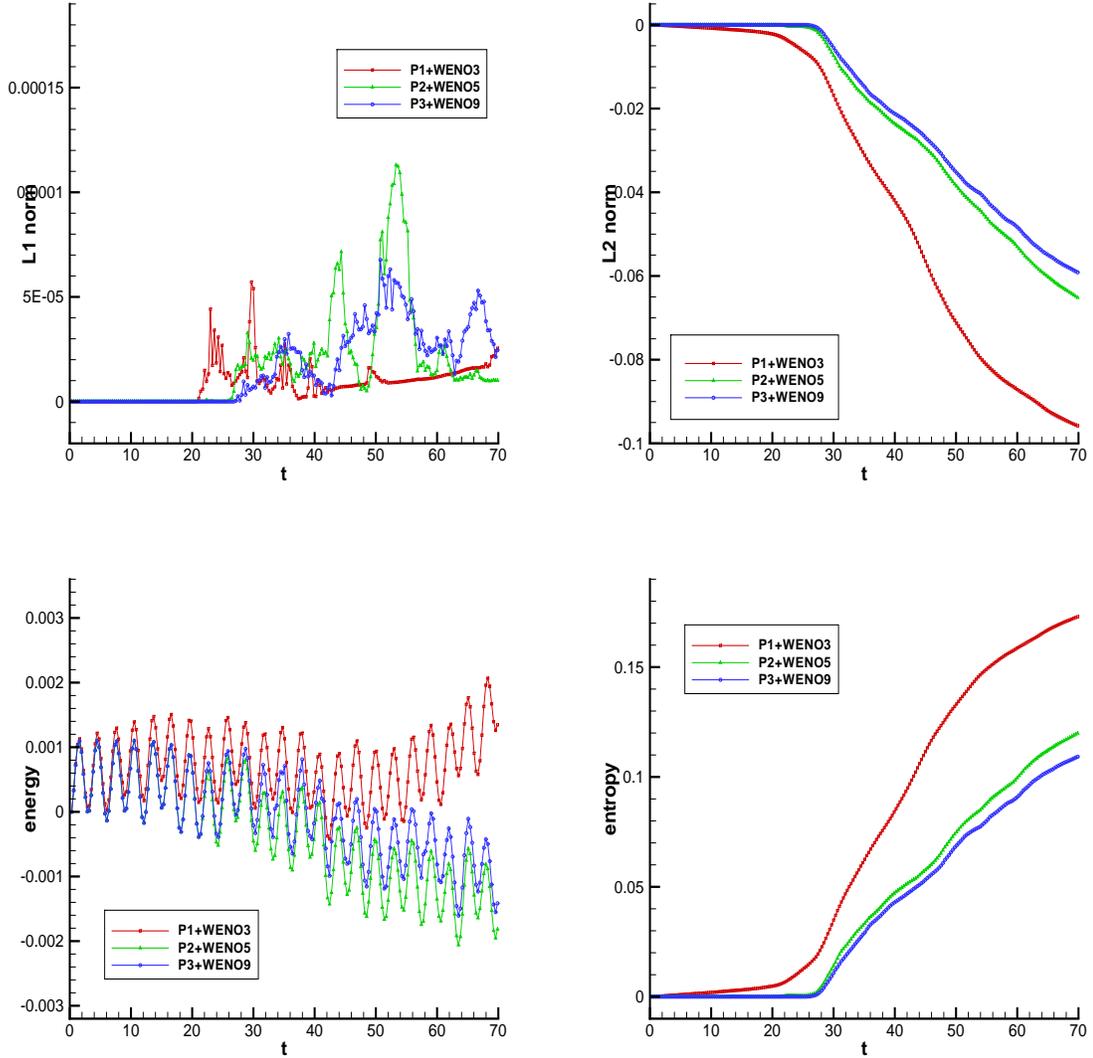


Figure 3.23: Two stream instability III. Time evolution of the relative deviations of discrete L^1 norms (upper left), L^2 norms, energy norms (lower left), and entropy (lower right). TVB constant $M=3.0$. Hybrid scheme: the SLDG schemes combined with SLWENO schemes.

3.4.2 Integral Deferred Correction in Correcting Splitting Error

Despite great efficiency and convenience of the dimensional splitting approach, the method is subject to low order splitting error in time ($\mathcal{O}(\Delta t^2)$) compared with the high order spatial accuracy. It is therefore important to overcome the issue, in order to have numerical algorithms that achieve high order accuracy in both space and time. In this subsection, we consider adopting the idea of IDC to correct lower order dimensional splitting error. The IDC methods are considered one-step, multi-stage integrators for solving initial value problems (IVPs). They are motivated by the defect/deferred correction (DC) methods [107, 40, 80], and more recently the spectral deferred correction (SDC) methods [36]. By construction, the IDC framework can systematically extend simple low order time integrators to high order ones by correcting provisional solutions. The DC/SDC/IDC methods and their variants have been applied to many application areas such as chemical rate equations, reactive flow [64, 9], hyperbolic equations [113], and parabolic equations [19]. Additionally, recent developments in IDC algorithms have opened up new possibilities for increased computational speed via parallelization [20].

Below, we investigate applying the IDC framework to correct the dimensional splitting error in solving the VP system, and guiding center models with plasma applications, as well as in simulating incompressible flows. We choose the dimensional splitting SLWENO scheme (3.14)-(3.15) reviewed in Section 3.2 as the base solver in the prediction and correction steps of IDC, but point out that SL finite difference

3.4. APPLICATIONS TO VLASOV SIMULATIONS

schemes with different reconstruction procedures such as those in [105, 77, 12] can also be used. In the IDC framework, the low order temporal accuracy from dimensional splitting is increased by iteratively approximating error functions via solving error equations. In particular, the temporal error accuracy is lifted by r in each correction loop, where $r = 1, 2$ for coupling the first order splitting and the Strang splitting, respectively. Our proposed SL scheme coupled with the IDC method enjoys the simplicity of the dimensional splitting algorithm, maintains the high order spatial accuracy, and can be designed to be of high order in time. However, the IDC methods with correction steps will render some CFL time step restrictions. We perform linear stability analysis via the classic Fourier approach and provide upper bounds of the CFL numbers for the proposed schemes. The CFL time step restriction is comparable to that for Eulerian methods using RK time discretization, leading to the computational cost at a similar scale.

We would like to remark that prior to the proposed IDC method in correcting the dimensional splitting error, several constructions of high order splitting methods have been developed. The methods proposed in [118, 47] are in the spirit of composition methods, requiring backward steppings (negative time steps). The number of intermediate stages, hence the computational cost, increases exponentially with the order of the splitting method. A fourth order splitting in [118] is applied to the VP system in [97]. Higher order splitting methods that do not require backward steppings by using complex coefficients are proposed in [49]. The number of intermediate stages scales similarly to [118]. A fourth order splitting method for a linear Vlasov equation was presented in [102]. However, the generalization of this fourth

order method to a nonlinear problem is not straightforward.

Overview of IDC Methods

We provide a brief review of IDC methods [22], designed for scalar/system of IVPs in the following form,

$$\begin{aligned} y'(t) &= g(t, y), \quad t \in [0, T], \\ y(0) &= y_0. \end{aligned} \tag{3.32}$$

The time domain, $[0, T]$, is discretized into intervals,

$$0 = t^1 < t^2 < \dots < t^n < \dots < t^N = T,$$

and each interval, $J^n = [t^n, t^{n+1}]$, is further discretized into sub-intervals,

$$t^n = t_0^n < t_1^n < \dots < t_m^n < \dots < t_M^n = t^{n+1}. \tag{3.33}$$

The IDC method on each time interval $[t^n, t^{n+1}]$ is described below. We drop the superscript n , e.g., $\tau_0 \doteq t_0^n$ in (3.33), with the understanding that the IDC method is described for one time interval. We also refer to $\tau_m \doteq t_m^n$ as grid points or quadrature nodes, whose index m runs from 0 to M . In the IDC method, the size of sub-intervals are uniform. Let $\Delta\tau \doteq \frac{t_{n+1} - t_n}{M}$, then $\tau_m = t^n + m\Delta\tau$, $m = 0, \dots, M$. The procedure of an IDC method with $M + 1$ uniformly distributed quadrature nodes as in equation (3.33) and with K correction loops denoted as IDC($M + 1$)J(K) is the following:

1. (prediction step) Use a low order numerical method to obtain a numerical

3.4. APPLICATIONS TO VLASOV SIMULATIONS

solution, $\vec{\eta}^{[0]} = (\eta_0^{[0]}, \dots, \eta_m^{[0]}, \dots, \eta_M^{[0]})$, which is a low order approximation to the exact solution at quadrature points. For example, applying a first order forward Euler method to (3.32) gives

$$\eta_{m+1}^{[0]} = \eta_m^{[0]} + \Delta\tau g(t, \eta_m^{[0]}), \quad m = 0, \dots, M - 1.$$

2. (correction loops) Use the error function to improve the accuracy of the scheme at each iteration.

For $k = 1, \dots, K$ (K is the number of correction steps)

- (a) Denote *the error function* from the previous step as

$$e^{(k-1)}(t) = y(t) - \eta^{(k-1)}(t),$$

where $y(t)$ is the exact solution and $\eta^{(k-1)}(t)$ is an M^{th} degree polynomial interpolating $\vec{\eta}^{[k-1]}$.

- (b) Denote *the residual function* as

$$\epsilon^{(k-1)}(t) = (\eta^{(k-1)})'(t) - g(t, \eta^{(k-1)}(t)).$$

- (c) Compute *the numerical error vector*, $\vec{\delta}^{[k]} = (\delta_0^{[k]}, \dots, \delta_m^{[k]}, \dots, \delta_M^{[k]})$, using a low order numerical method to discretize the following *error equation*

with a zero initial condition,

$$\left(e^{(k-1)} + \int_0^t \epsilon^{(k-1)}(\tau) d\tau \right)'(t) = g(t, \eta^{(k-1)}(t) + e^{(k-1)}(t)) - g(t, \eta^{(k-1)}(t)). \quad (3.34)$$

For example, applying a first order forward Euler scheme to equation (3.34) gives

$$\begin{aligned} \delta_{m+1}^{[k]} = \delta_m^{[k]} + \Delta\tau(g(\tau_m, \eta_m^{[k-1]} + \delta_m^{[k]}) - g(\tau_m, \eta_m^{[k-1]})) + \sum_{\ell=0}^M \alpha_{m,\ell} g(\tau_\ell, \eta_\ell^{[k-1]}) \\ + \eta_m^{[k-1]} - \eta_{m+1}^{[k-1]}, \quad m = 0, \dots, M-1. \end{aligned} \quad (3.35)$$

To get equation (3.35) from the discretization of (3.34), $\int_{\tau_m}^{\tau_{m+1}} \epsilon^{(k-1)}(t) dt$ is approximated by

$$\eta_{m+1}^{[k-1]} - \eta_m^{[k-1]} - \sum_{\ell=0}^M \alpha_{m,\ell} g(\tau_\ell, \eta_\ell^{[k-1]}),$$

where $\sum_{j=0}^M \alpha_{m,\ell} g(\tau_\ell, \eta_\ell^{[k-1]})$ approximates $\int_{\tau_m}^{\tau_{m+1}} g(t, \eta^{(k-1)}(t)) dt$ by quadrature formulas. We note that such a way of evolving the error function in the IDC algorithm is advantageous compared with that in the traditional DC algorithm. It introduces more stability by using the integral form, rather than the differentiation form, of the residual.

- (d) Update the numerical solution $\vec{\eta}^{[k]} = \vec{\eta}^{[k-1]} + \vec{\delta}^{[k]}$.

Remark 3.12. (About notations.) In our description of IDC, we let English letters

3.4. APPLICATIONS TO VLASOV SIMULATIONS

$y_m, e_m^{(k)}$ denote the exact solutions and exact error functions, and Greek letters $\eta_m^{[k]}, \delta_m^{[k]}$ denote the numerical approximations to the exact solutions and error functions.

We use subscript m to denote the location $t = \tau_m$. The superscripts k with round brackets ((k)) and square brackets ($[k]$) are for functions and vectors (or their components) respectively at the prediction ($k = 0$) and correction loops ($k = 1, \dots, K$).

We let $\vec{\cdot}$ denote the vector on IDC quadrature nodes. For example, $\vec{y} = (y_0, \dots, y_M)$.

Remark 3.13. (About the distribution of the quadrature nodes.) The IDC methods reviewed above adopt the uniformly distributed quadrature nodes to compute the residual. In [22], it is proved under some mild assumption, that the order accuracy of an IDC method can be increased by r order when an r^{th} order RK integrator is used to solve the error equation in each correction loop. The numerical results reported in [21] show that such high order accuracy lifting property does not always hold for an SDC method, which is constructed with Gaussian quadrature nodes.

Remark 3.14. (About computational cost and storage requirement.) In terms of the order of accuracy, per $\Delta\tau$ (a subinterval size in the IDC method), numerical solutions $\eta_m^{[K]}$ are $(K + 1)^{\text{th}}$ order approximations to the exact solution at quadrature nodes y_m for $m = 0, \dots, M$; in terms of computational cost, per $\Delta\tau$, there are $K + 1$ function evaluations for a $(K + 1)^{\text{th}}$ order IDC $(M + 1)$ J (K) method. In this sense, the IDC method is considered to be efficient with relatively low computational cost among RK methods with the same order of accuracy. We remark that an IDC method can be considered as a one-step RK method with its Butcher table constructed in [21]. At the same time, an IDC $(M + 1)$ J (K) method requires storage space for numerical solutions at $M + 1$ quadrature nodes.

IDC methods for the VP system

The dimensional splitting SL methods described in Section 3.1 (SLDG) and Section 3.2 (SLWENO) are very high order accurate in space, but only low order in time. Therefore, we use the IDC framework to increase the temporal order of the dimensional splitting method. We will only consider using first order splitting for brevity, but also comment on the use of Strang splitting. Consider the VP system with only 1-D in space and 1-D in velocity for simplicity of notation. Here, the SLWENO scheme is used as a base scheme. Note that the IDC framework also applies to the SLDG scheme.

1. (prediction step) Use the dimensional splitting SLWENO method described in Section 3.2 in the prediction step of the IDC framework. More specifically, predict solution $\vec{\eta}^{[0]} = (\eta_0^{[0]}, \dots, \eta_M^{[0]})$ at time step subintervals (3.33) for each spatial and velocity grid point, say (x_i, v_j) , $\forall i = 1, \dots, N_x, \quad j = 0, \dots, N_v$.
2. (correction loops) Use the error function to reduce the dimensional splitting error at each iteration. Our correction procedure is based on a fixed location, say (x_i, v_j) .

For $k = 1, \dots, K$ (K is the number of correction steps)

- (a) *The error function* is defined as $e^{(k-1)}(t, x_i, v_j) = f(t, x_i, v_j) - \eta^{(k-1)}(t, x_i, v_j)$, where $f(t, x_i, v_j)$ is the exact solution and $\eta^{(k-1)}(t, x_i, v_j)$ is the polynomial interpolating $\vec{\eta}^{[k-1]} = (\eta_0^{[k-1]}(x_i, v_j), \dots, \eta_M^{[k-1]}(x_i, v_j))$ at quadrature points (3.33) over a time interval $[t_n, t_{n+1}]$.

(b) *The residual function* is defined as

$$\epsilon(t, x, v) = -(\eta_t + v \cdot \eta_x + E^\eta(x) \cdot \eta_v), \quad (3.36)$$

where E^η is the electrostatic field induced by the numerical distribution function $\eta(t, x, v)$.

(c) *The error equation* about the error function is obtained by adding the residual equation (3.36) to the Vlasov equation (1.1),

$$e_t + v \cdot e_x + (E^\eta + E^e) \cdot e_v + E^e \cdot \eta_v = \epsilon, \quad (3.37)$$

where E^e is the electrostatic field induced by the error function $e(t, x, v)$.

(d) Evolve the error equation (3.37) with zero initial condition by the same dimensional splitting SLWENO method for spatial advection and velocity ac/deceleration as that for the Vlasov equation. Specifically, we split the error equation (3.37) into three parts,

$$e_t + v \cdot e_x = 0 \quad (\text{spatial advection}) \quad (3.38)$$

$$e_t + E^{\eta+e} \cdot e_v = 0 \quad (\text{velocity ac/deceleration}) \quad (3.39)$$

$$e_t + E^e \cdot \eta_v = \epsilon \quad (\text{source term}). \quad (3.40)$$

To evolve the error equation (3.37) from τ_m to τ_{m+1} in a splitting fashion, we first evolve the solution $\delta_m^{[k]}$ by approximating equation (3.38) with the SLWENO scheme. The solution after the evolution is denoted by $\delta_{m,*}^{[k]}$.

3.4. APPLICATIONS TO VLASOV SIMULATIONS

After that, we solve the electrostatic field $E^{\eta_m^{[k-1]} + \delta_{m,*}^{[k]}}$ induced by $\eta_m^{[k-1]} + \delta_{m,*}^{[k]}$ from the Poisson's equation, then get $E^{\delta_{m,*}^{[k]}} = E^{\eta_m^{[k-1]} + \delta_{m,*}^{[k]}} - E^{\eta_m^{[k-1]}}$. We then evolve $\delta_{m,*}^{[k]}$ by approximating equation (3.39) again with the SLWENO scheme. The numerical solution is denoted by $\delta_{m,**}^{[k]}$. Finally we solve equation (3.40), but using the integral form of the residual in an SDC/IDC fashion,

$$\left(e(t, x, v) - \int_{t_n}^t \epsilon(\tau, x, v) d\tau \right)'(t) = -E^e \cdot \eta_v. \quad (3.41)$$

Note that $E^e \cdot \eta_v$, as well as the terms $v \cdot \eta_x$ and $E^\eta \cdot \eta_v$ in equation (3.36) for the residual ϵ , are approximated in a flux difference form to ensure mass conservation. In the simulation, we use a fifth order WENO procedure to reconstruct all the numerical fluxes. Similar to (3.35), we approximate (3.41) at (x_i, v_j) by

$$\begin{aligned} \delta_{m+1}^{[k]} = \delta_{m,**}^{[k]} - \frac{\Delta\tau}{\Delta v} E^{\delta_{m,*}^{[k]}} (\widehat{\eta_m^{[k-1]}}_{i,j+\frac{1}{2}} - \widehat{\eta_m^{[k-1]}}_{i,j-\frac{1}{2}}) - \sum_{\ell=0}^M \alpha_{m,\ell} g(\eta_\ell^{[k-1]}) \\ + \eta_{m+1}^{[k-1]} - \eta_m^{[k-1]}, \quad m = 0, \dots, M-1, \end{aligned} \quad (3.42)$$

with

$$g(\eta_\ell^{[k-1]}) = v \frac{\widehat{\eta_\ell^{[k-1]}}_{i+\frac{1}{2},j} - \widehat{\eta_\ell^{[k-1]}}_{i-\frac{1}{2},j}}{\Delta x} + E^{\eta_\ell^{[k-1]}} \frac{\widehat{\eta_\ell^{[k-1]}}_{i,j+\frac{1}{2}} - \widehat{\eta_\ell^{[k-1]}}_{i,j-\frac{1}{2}}}{\Delta v}, \quad (3.43)$$

where we omit the (i, j) dependence when there is no confusion.

3.4. APPLICATIONS TO VLASOV SIMULATIONS

- (e) Update the solution by the approximate error function computed from the correction step,

$$\vec{\eta}^{[k]}(x_i, v_j) = \vec{\eta}^{[k-1]}(x_i, v_j) + \vec{\delta}^{[k]}(x_i, v_j). \quad (3.44)$$

The flow chart of SLWENO coupled with IDC($M + 1$)J(K) for the VP system is outlined below.

Algorithm 3.15. A SLWENO scheme coupled in the IDC framework for the VP system:

1. Find the prediction solution $\vec{\eta}^{[0]} = (\eta_0^{[0]}, \dots, \eta_M^{[0]})$ at time step subintervals for each spatial and velocity location (x_i, v_j) , $\forall i = 1, \dots, N_x$, $j = -N_v/2, \dots, N_v/2$, by using the dimensional splitting SLWENO scheme.

2. **For** $k = 1, \dots, K$

Perform the correction loop to update the solution $\vec{\eta}^{[k]}$.

- Solve the numerical error vector $\vec{\delta}^{[k]} = (\delta_0^{[k]}, \dots, \delta_M^{[k]})$ by evolving the split error equations (3.38)-(3.40) with zero initial condition at time step subintervals for each spatial and velocity location.

Specifically,

For $m = 0, \dots, M - 1$,

(a) Evolve $\delta_m^{[k]}$ by solving equation (3.38) to get $\delta_{m,*}^{[k]}$ by the SLWENO scheme.

(b) Evolve $\delta_{m,*}^{[k]}$ by solving equation (3.39) to get $\delta_{m,**}^{[k]}$ by the SLWENO scheme.

(c) Evolve $\delta_{m,**}^{[k]}$ by solving equation (3.40) in an IDC fashion using equation (3.42) to get $\delta_{m+1}^{[k]}$.

End For

- update the solution by $\vec{\eta}^{[k]} = \vec{\eta}^{[k-1]} + \vec{\delta}^{[k]}$.

End For

3.4. APPLICATIONS TO VLASOV SIMULATIONS

The proposed IDC-dimensional splitting SLWENO scheme enjoys the mass conservation property; see the following proposition.

Proposition 3.16. **Algorithm 3.15** *conserves the total mass for solving the VP system, if the boundary conditions are periodic.*

Proof: First note that the SLWENO scheme conserves the total mass when periodic boundary conditions are imposed. Thus, we have

$$\sum_i \sum_j \eta_0^{[0]}(x_i, v_j) = \sum_i \sum_j \eta_1^{[0]}(x_i, v_j) = \dots = \sum_i \sum_j \eta_M^{[0]}(x_i, v_j), \quad (3.45)$$

for the prediction. By (3.44), to prove the mass conservation, it is sufficient to prove

$$\sum_i \sum_j \delta_m^{[k]}(x_i, v_j) = 0, \quad k = 1, \dots, K, \quad m = 0, \dots, M. \quad (3.46)$$

Let's prove (3.46) for $k = 1$. A similar argument carries over for general k . The split error equations (3.38)-(3.40) are solved with zero initial condition, hence $\sum_i \sum_j \delta_0^{[1]}(x_i, v_j) = 0$. Now assume $\sum_i \sum_j \delta_m^{[1]}(x_i, v_j) = 0$, we will prove

$$\sum_i \sum_j \delta_{m+1}^{[1]}(x_i, v_j) = 0.$$

Due to the total mass conservative property of SLWENO, we get

$$\sum_i \sum_j \delta_{m,*}^{[1]}(x_i, v_j) = 0, \quad \sum_i \sum_j \delta_{m,**}^{[1]}(x_i, v_j) = 0. \quad (3.47)$$

Since the derivatives in equation (3.42) are written into a flux difference form, it

follows that

$$\begin{aligned} \sum_i \sum_j \delta_{m+1}^{[1]}(x_i, v_j) &= \sum_i \sum_j \delta_{m,**}^{[1]}(x_i, v_j) - \sum_i \sum_j \frac{\Delta\tau}{\Delta v} E^{\delta_{m,*}^{[k]}} (\widehat{\eta_m^{[k-1]}}_{i,j+\frac{1}{2}} - \widehat{\eta_m^{[k-1]}}_{i,j-\frac{1}{2}}) \\ &\quad - \sum_i \sum_j \sum_{\ell=0}^M \alpha_{m,\ell} g(\eta_\ell^{[0]}(x_i, v_j)) + \sum_i \sum_j \eta_{m+1}^{[0]}(x_i, v_j) - \sum_i \sum_j \eta_m^{[0]}(x_i, v_j). \end{aligned}$$

Thanks to (3.45), (3.47) and the cancellation of the flux difference form, we get

$$\begin{aligned} \sum_i \sum_j \delta_{m+1}^{[1]}(x_i, v_j) &= - \sum_i \sum_j \sum_{\ell=0}^M \alpha_{m,\ell} g(\eta_\ell^{[0]}(x_i, v_j)) \\ &= - \sum_{\ell=0}^M \alpha_{m,\ell} \sum_i \sum_j g(\eta_\ell^{[0]}(x_i, v_j)) \\ &= 0, \end{aligned}$$

where the last equality holds due to the flux difference form of g in (3.43). By induction, we complete the proof. ■

Remark 3.17. *Algorithm 3.15* can be extended to IDC methods coupled with second order Strang splitting without additional complication. The only modification is to employ the Strang splitting SLWENO scheme to get a prediction, and again employ the Strang splitting to solve the split error equations (3.38)-(3.40). The procedure of an IDC method coupled with the second order Strang splitting is denoted as IDC-Strang($M+1$) $J(K)$, when $M+1$ uniformly distributed quadrature nodes and K correction loops are used. The numerical results reported below indicate that the temporal order accuracy of the IDC-Strang method for the VP system can be increased with second order per correction. The modified IDC-Strang splitting SLWENO scheme

also enjoys the mass conservation property. The proof is quite similar, therefore we omit it.

IDC Methods for the Guiding Center Model

We consider the guiding center model, which describes a highly magnetized plasma in the transverse plane of a tokamak [105, 34]. We consider equation

$$\rho_t + \mathbf{E}^\perp \cdot \nabla \rho = 0. \quad (3.48)$$

where ρ is the particle density function, $\mathbf{E}^\perp = (-E_2, E_1)$ with the electrostatic field $\mathbf{E} = (E_1, E_2)$ satisfying a 2-D Poisson's equation

$$\Delta \Phi = \rho, \quad \mathbf{E} = -\nabla \Phi. \quad (3.49)$$

Compared to the VP system, the 1-D equations obtained from dimensional splitting of equation (3.48) are variable coefficient equations. We apply the SLWENO algorithm in [85] as described in Section 3.2 as 1-D solvers. The 2-D Poisson's equation is again solved by a 2-D FFT. The computational procedure of applying the IDC method to reduce the splitting error is similar to that for the VP system, except that we need to formulate a new residual function and an error equation for the guiding center model (3.48). Below we provide the residual function and the error equation for the guiding center Vlasov equation with notations that are consistent with the previous subsection.

- The residual function is defined as

$$\epsilon(t, x, y) = -(\eta_t - (E_2^\eta \eta)_x + (E_1^\eta \eta)_y) \quad (3.50)$$

where $\mathbf{E}^\eta = -\nabla \Phi^\eta$ with $\Delta \Phi^\eta = \eta$.

- The error equation about the error function $e(t, x, y) = \rho(t, x, y) - \eta(t, x, y)$ is obtained by adding (3.50) to (3.48),

$$e_t - (E_2^\rho - E_2^\eta)_x + (E_1^\rho - E_1^\eta)_y = \epsilon, \quad (3.51)$$

where \mathbf{E}^ρ and \mathbf{E}^η are the electrostatic field induced by the exact solution $\rho(t, x, y)$ and the numerical solution $\eta(t, x, y)$ respectively. From $\rho = \eta + e$ and $\mathbf{E}^\rho = \mathbf{E}^\eta + \mathbf{E}^e$, we have

$$e_t - ((E_2^e + E_2^\eta)e)_x + ((E_1^e + E_1^\eta)e)_y - (E_2^e \eta)_x + (E_1^e \eta)_y = \epsilon, \quad (3.52)$$

where \mathbf{E}^e is the electrostatic field induced by the error function $e(t, x, y)$. Similar to the proposed scheme for the VP system, the error equation is evolved with zero initial conditions by dimensional splitting,

$$e_t - ((E_2^e + E_2^\eta)e)_x = 0, \quad (3.53)$$

$$e_t + ((E_1^e + E_1^\eta)e)_y = 0, \quad (3.54)$$

$$e_t - (E_2^e \eta)_x + (E_1^e \eta)_y = \epsilon. \quad (3.55)$$

Again, the 1-D equations (3.53)-(3.54) are solved by the SLWENO method, and the last equation above is solved in a similar manner as equation (3.41) for the VP system. Similar to Proposition 3.16, the algorithm enjoys the mass conservation property, since the splitting is performed in a conservative way.

Numerical Stability

IDC is a numerical approach in generating time stepping algorithms with high order accuracy, yet numerical stability of the IDC method using SLWENO as the base scheme remains to be investigated. In the following, we investigate stability properties of the proposed IDC-SLWENO method via classical Fourier analysis. We provide the CFL restriction for stability when the method is applied to a linear problem as guidance for choosing numerical time steps for general nonlinear problems below.

We consider the linear model problem (3.1) with constant coefficient $a = 1$. Assume the mesh is uniform and boundary conditions are periodic. We consider a subinterval in IDC with time step size $\Delta\tau$. An explicit linear scheme for equation (3.1) can be written in the following form:

$$u_j^{n+1} = \sum_{k=-r}^l C_k u_{j+k}^n, \quad (3.56)$$

where C_k , $k = -r, \dots, l$ are constants, that depend on the CFL number $\lambda \doteq \Delta\tau/\Delta x$ but are independent of the solution. For example, the third order linear SL scheme

3.4. APPLICATIONS TO VLASOV SIMULATIONS

combined with IDC2J0 reads:

$$\begin{aligned} u_j^{n+1} = & \frac{1}{6} \lambda^* ((\lambda^*)^2 - 1) u_{j^*-2}^n + \frac{1}{2} \lambda^* (-(\lambda^*)^2 + \lambda^* + 2) u_{j^*-1}^n \\ & + \frac{1}{2} ((\lambda^*)^3 - 2(\lambda^*)^2 - \lambda^* + 2) u_{j^*}^n - \frac{1}{6} \lambda^* ((\lambda^*)^2 - 3\lambda^* + 2) u_{j^*+1}^n \end{aligned}$$

where $j^* = j - \lfloor \lambda \rfloor$ and $\lambda^* = \lambda - \lfloor \lambda \rfloor$. The third order linear SL scheme combined with IDC2J1 reads:

$$\begin{aligned} u_j^{n+1} = & -\frac{1}{72} \lambda^2 (\lambda^2 - 1) u_{j-4}^n + \frac{1}{24} \lambda^2 (3\lambda^2 - \lambda - 4) u_{j-3}^n \\ & + \frac{1}{12} \lambda (-4\lambda^3 + 4\lambda^2 + 7\lambda - 2) u_{j-2}^n + \frac{1}{36} \lambda (13\lambda^3 - 24\lambda^2 - 16\lambda + 36) u_{j-1}^n \\ & - \frac{1}{24} (3\lambda^4 - 10\lambda^3 + 5\lambda^2 + 12\lambda - 24) u_j^n + \frac{1}{24} \lambda (-\lambda^3 + \lambda^2 + 4\lambda - 8) u_{j+1}^n \\ & + \frac{1}{36} \lambda^2 (\lambda^2 - 3\lambda + 2) u_{j+2}^n \end{aligned}$$

when $0 < \lambda < 1$; and

$$\begin{aligned} u_j^{n+1} = & -\frac{1}{72} \lambda^2 (\lambda^2 - 3\lambda + 2) u_{j-5}^n + \frac{1}{24} \lambda^2 (3\lambda^2 - 10\lambda + 7) u_{j-4}^n \\ & - \frac{1}{12} \lambda^2 (4\lambda^2 - 16\lambda + 13) u_{j-3}^n + \frac{1}{36} \lambda (13\lambda^3 - 63\lambda^2 + 71\lambda - 6) u_{j-2}^n \\ & + \frac{1}{24} \lambda (-3\lambda^3 + 19\lambda^2 - 34\lambda + 24) u_{j-1}^n - \frac{1}{24} (\lambda^4 - 4\lambda^3 + \lambda^2 + 12\lambda - 24) u_j^n \\ & + \frac{1}{36} \lambda (\lambda^3 - 6\lambda^2 + 11\lambda - 12) u_{j+1}^n \end{aligned}$$

when $1 \leq \lambda < 2$.

3.4. APPLICATIONS TO VLASOV SIMULATIONS

Classical Fourier analysis is performed to linear schemes in the form of (3.56). We substitute $u_j^n = g^n e^{ij\xi}$ into linear schemes in the form of (3.56) and compute the corresponding amplification factors g . We use Mathematica to derive the explicit form of numerical schemes and the corresponding amplification factors g . The time step restriction from linear stability can be obtained by maximizing the CFL number λ with the constraint that $g \leq 1$ for any $\xi \in [0, 2\pi]$. We remark that it is exceedingly tedious and sometimes very difficult to derive the upper bounds of the CFL number analytically, especially for high order schemes. Hence we rely on numerical approaches to obtain such upper bounds: we find the maximum λ such that $g \leq 1$ for 2000 evenly distributed points $\{\xi_n\}$ over $[0, 2\pi]$. In Table 3.10, we list the upper bounds of the CFL numbers for several numerical schemes which use linear SL schemes as base solvers in the IDC framework, of different orders. It is observed that the CFL upper bounds for an IDC subinterval are comparable to those of the Eulerian RK WENO scheme with the same orders of accuracy. As commented in Remark 3.14, there are $(K + 1)$ function evaluations for the IDC $(M + 1)$ J (K) method per IDC subinterval $\Delta\tau$, where $(K + 1)^{th}$ order accuracy is achieved at all quadrature nodes $\tau_m, m = 0, \dots, M$. The IDC method is considered to be efficient among Eulerian RK methods with the same order of accuracy. We remark that Fourier analysis can be extended to arbitrary order cases, however the algebraic manipulations may become prohibitively complicated.

Numerical Results

Now, we present some simulation results for solving the VP system, the guiding center Vlasov model and the 2-D incompressible Euler equations. Through these

3.4. APPLICATIONS TO VLASOV SIMULATIONS

Table 3.10: The upper bounds of the CFL numbers. SL3 and SL5 are SL schemes with the third and fifth order linear reconstructions. $\text{IDC}(M + 1)\text{J}(K)$ denotes an IDC procedure with $M + 1$ uniformly distributed quadrature nodes and K correction loops.

Scheme	IDC2J0	IDC2J1	IDC3J0	IDC3J1	IDC3J2
SL3	No restriction	1.50	No restriction	0.73	0.67
SL5	No restriction	1.55	No restriction	0.71	0.66

examples, we numerically demonstrate the low order dimensional splitting error, and the IDC’s ability to correct these errors. In the simulations,

$$\text{CFL} = \Delta\tau \left(\frac{|a_x|}{\Delta x} + \frac{|a_v|}{\Delta v} \right),$$

where $|a_x|$ and $|a_v|$ are maximum wave propagation speeds in the x - and the v -directions respectively, and $\Delta\tau$ is the size of a sub-interval in the IDC method. Again, $\text{IDC}(M + 1)\text{J}(K)$ and $\text{IDC-Strang}(M + 1)\text{J}(K)$ denote IDC procedures with $M + 1$ uniformly distributed quadrature nodes and K correction loops coupling the first order splitting and the Strang splitting, respectively. The length of the domain in the x -direction is $L = \frac{2\pi}{k}$ and the background ion distribution function is fixed, uniform and chosen so that the total net charge density for the system is zero. To minimize the error from truncating the domain in the v -direction, we let $v_{max} = 2\pi$. A fifth order SLWENO scheme is employed as a base scheme to achieve fifth order spatial accuracy.

Example 3.18. (Accuracy validation.) We validate the capacity of the proposed scheme in correcting the low order splitting error. We set the computational mesh

3.4. APPLICATIONS TO VLASOV SIMULATIONS

as $N_x \times N_v = 400 \times 400$. In the simulation, we fix the spatial mesh and compute the numerical solutions up to $T = 0.1$ with different CFL numbers. A reference solution is computed with $\text{CFL} = 0.01$ by using IDC3J3. In Table 3.11-3.13, we report the L^1 error and the orders of accuracy when IDC methods of different orders coupled with first order splitting strategy are used for the VP system with three sets of initial conditions including the strong Landau damping (3.28), the two stream instability I (3.29) and the two stream instability II (3.30). It is observed that the dimensional splitting error in time is significantly reduced when the IDC framework is applied, and $(K + 1)^{\text{th}}$ order of accuracy is clearly achieved for IDC3JK ($K \leq 3$). In Table 3.14, the L^1 error and the orders of accuracy for IDC methods coupled with Strang splitting are reported, where $(2K + 2)^{\text{th}}$ order of accuracy is observed for IDC-Strang3JK ($K \leq 1$). We remark that: in terms of stability, the Strang splitting is advantageous as it is known to be unconditionally stable, whereas the proposed scheme with IDC3J1 is only conditionally stable, see Table 3.10. When the CFL number is small enough, e.g., $\text{CFL} = 0.6$, the magnitudes of errors from IDC3J1 and the second order Strang splitting (IDC-Strang3J0 in Table 3.14) are comparable; however, the computational cost from the Strang splitting scheme is less than that of the IDC3J1 scheme. Also note that the performance of IDC methods with first order splitting and Strang splitting are very similar when the same order accuracy and CFL numbers are considered; hence, we only report the numerical results from IDC methods with first order splitting below for brevity.

Example 3.19. (Performance assessment.) We assess the ability of the proposed IDC-SLWENO schemes in preserving the physical norms of the VP system. The

3.4. APPLICATIONS TO VLASOV SIMULATIONS

Table 3.11: Strong Landau damping. $T=0.1$. L^1 error and orders of accuracy.

CFL	IDC3J0		IDC3J1		IDC3J2		IDC3J3	
	L^1 error	order						
0.6	3.85E-06	–	9.71E-09	–	2.22E-11	–	1.83E-13	–
0.5	3.23E-06	0.97	6.79E-09	1.96	1.30E-11	2.95	8.90E-14	3.95
0.4	2.60E-06	0.98	4.37E-09	1.97	6.67E-12	2.98	3.66E-14	3.98
0.3	1.94E-06	1.02	2.44E-09	2.03	2.79E-12	3.03	1.15E-14	4.03
0.2	1.30E-06	0.99	1.09E-09	1.98	8.34E-13	2.98	2.31E-15	3.95

Table 3.12: Two stream instability I. $T=0.1$. L^1 error and orders of accuracy.

CFL	IDC3J0		IDC3J1		IDC3J2		IDC3J3	
	L^1 error	order						
0.6	4.46E-07	–	2.36E-09	–	9.23E-12	–	1.25E-13	–
0.5	3.73E-07	0.98	1.66E-09	1.94	5.43E-12	2.91	6.16E-14	3.88
0.4	2.99E-07	1.00	1.06E-09	2.00	2.78E-12	3.00	2.52E-14	4.00
0.3	2.24E-07	1.00	5.94E-10	2.02	1.17E-12	3.02	7.93E-15	4.02
0.2	1.50E-07	0.99	2.65E-10	1.99	3.48E-13	2.98	1.60E-15	3.94

mesh is reset as $N_x \times N_v = 256 \times 256$ and $CFL = 0.6$. We track the evolution histories of these quantities and only show the results of total mass and energy here. In Figure 3.24, we report time evolution of the relative deviation in the total mass and the total energy for all three problems. It is observed that the proposed scheme can preserve the total mass up to machine error as stated in Proposition 3.16. Moreover, the higher order schemes can better preserve the total energy than the first order scheme (IDC3J0). The performance of the second (IDC3J1), third (IDC3J2), and fourth (IDC3J3) order schemes are qualitatively similar. The reason may be that the error from the spatial discretization dominates the temporal error in these test cases. At last, we show the contour plots of the numerical solution for the strong

3.4. APPLICATIONS TO VLASOV SIMULATIONS

Table 3.13: Two stream instability II. T= 0.1. L^1 error and orders of accuracy.

CFL	IDC3J0		IDC3J1		IDC3J2		IDC3J3	
	L^1 error	order						
0.6	6.72E-07	–	2.33E-09	–	3.61E-12	–	1.79E-14	–
0.5	5.56E-07	1.04	1.61E-09	2.04	2.08E-12	3.02	8.60E-15	4.01
0.4	4.46E-07	0.99	1.03E-09	2.00	1.07E-12	3.00	3.53E-15	3.99
0.3	3.36E-07	0.98	5.84E-10	1.98	4.53E-13	2.98	1.13E-15	3.95
0.2	2.24E-07	1.00	2.60E-10	2.00	1.34E-13	3.00	2.65E-16	3.58

Table 3.14: IDC methods with Strang splitting for the VP systems. T= 0.1. L^1 error and orders of accuracy.

CFL	Strong Landau damping				Two stream instability II			
	IDC-Strang3J0		IDC-Strang3J1		IDC-Strang3J0		IDC-Strang3J1	
	L^1 error	order	L^1 error	order	L^1 error	order	L^1 error	order
0.6	5.27E-09	–	7.97E-14	–	1.07E-09	–	5.62E-14	–
0.5	3.68E-09	1.96	3.88E-14	3.94	7.55E-10	1.91	2.78E-14	3.87
0.4	2.37E-09	1.98	1.60E-14	3.97	4.89E-10	1.94	1.14E-14	3.98
0.3	1.32E-09	2.03	5.05E-15	4.01	2.82E-10	1.91	3.64E-15	3.97
0.2	5.93E-10	1.98	1.04E-15	3.90	1.37E-10	1.78	7.99E-16	3.74

Landau damping in Figure 3.25, two stream instability I and II in Figure 3.26 and 3.27 respectively to demonstrate the performance of the proposed scheme. We only report the numerical results by the high order IDC3J3 method with CFL = 0.6 for brevity. The results agree well with those by the hybrid schemes in previous subsection and by other methods presented in the literature [52, 97, 84].

Example 3.20. (2-D guiding center Vlasov equation.) We consider the guiding center model (3.48) with the initial condition [34]

$$\rho(0, x, y) = \sin(y) + 0.015 \cos(kx),$$

3.4. APPLICATIONS TO VLASOV SIMULATIONS

and periodic boundary conditions. We let $k = 0.5$, thereby creating a Kelvin-Helmholtz instability. In the simulations, we set a mesh as $N_x \times N_y = 128 \times 128$. We use a third order SLWENO scheme as a base scheme to achieve third order spatial accuracy. In Figure 3.28 (top), we report the contour plots of numerical solutions at time $T = 40$ for the first order splitting scheme (IDC3J0) and the third order scheme (IDC3J2) with $CFL = 0.67$. A noticeable difference is observed. In Figure 3.29 (left), 1-D cuts of the numerical solutions at $y = \pi$ are plotted. The reference solution is computed by IDC3J2 with $CFL = 0.05$. It can be observed that the numerical solution obtained by IDC3J2 with $CFL = 0.67$ qualitatively matches the reference. However, a significant difference between the numerical solution obtained by the first order scheme and the reference solution is observed. Then we reduce the CFL number to 0.05 for the first order splitting scheme. The 2-D contour plot by the first order scheme approximately matches the reference by IDC3J2, see Figure 3.28 (bottom). A more precise match is also observed when the 1-D cuts of the numerical solutions are compared. The presented numerical evidence shows better performance from higher order numerical schemes in time. Note that the continuous guiding center model (3.48) preserves the L^2 norms of ρ (enstrophy) and the L^2 norms of \mathbf{E} (energy), i.e.,

$$\frac{d}{dt} \|\rho(t)\|_{L^2} = \frac{d}{dt} \|\mathbf{E}(t)\|_{L^2} = 0.$$

We track the relative deviations of these invariants numerically as a measurement of the quality of numerical schemes. In Figure 3.30, the time evolutions of the enstrophy and the energy for the first order splitting scheme (IDC3J0), the second

3.4. APPLICATIONS TO VLASOV SIMULATIONS

order scheme (IDC3J1) and the third order scheme (IDC3J2) are reported. It is observed that these quantities are better preserved by higher order schemes in time. Also note that little difference can be observed between the IDC3J1 and IDC3J2 solutions. In this case, the spatial error may dominate.

Example 3.21. (2-D incompressible Euler equations.) We consider 2-D incompressible Euler equations in the vorticity stream-function formulation as follows

$$\omega_t + \nabla \cdot (\mathbf{u}\omega) = 0, \quad x \in [0, 2\pi], \quad y \in [0, 2\pi]. \quad (3.57)$$

Here $\mathbf{u} = \nabla^\perp \Phi = (-\Phi_y, \Phi_x)$, where Φ is solved from the Poisson's equation $\Delta \Phi = \omega$. We note that equation (3.57) is in the same form as the 2-D guiding center Vlasov equation (3.48). We first test the accuracy of the schemes with the following initial conditions:

$$\omega(t = 0, x, y) = -2 \sin(x) \sin(y),$$

and periodic boundary conditions. Note that the exact solution is identical to the initial condition. We use this example to check the orders of accuracy of our proposed schemes when the IDC framework is adopted to correct the splitting error. In the simulation, we fix the spatial mesh as $N_x \times N_y = 300 \times 300$ and compute the numerical solutions with different CFLs. We evolve the solutions up to time $T = 1$. In Table 3.15, we report the L^1 error and orders of accuracy for the first order splitting scheme (IDC3J0), the second order scheme (IDC3J1) and the third order scheme (IDC3J2). Expected orders of accuracy are observed. Note that third order convergence is not clearly observed for IDC3J2 when the CFL numbers are relatively small ($\text{CFL} \leq 0.5$).

3.4. APPLICATIONS TO VLASOV SIMULATIONS

In this case, the spatial error begins to dominate when the numerical error is around $9.00E - 9$. The L^1 error for IDC3J2 with CFL = 0.01 is $9.31E - 09$. We remark that there is a certain range of CFL numbers (which could be a very small interval), related to the spatial resolution and accuracy, where the temporal order of accuracy can be numerically observed. Above that range, the scheme is either numerically unstable or the order of convergence can not be observed yet; below that range, the spatial error could dominate, with which the temporal order of convergence can no longer be observed.

Table 3.15: 2-D incompressible Euler equation. A third order SLWENO scheme coupled in the IDC framework. L^1 norms of errors and orders of accuracy. $N_x \times N_y = 300 \times 300$. T= 1.

CFL	IDC3J0		IDC3J1		IDC3J2	
	L^1 error	L^1 order	L^1 error	L^1 order	L^1 error	L^1 order
0.67	2.16E-03	–	8.89E-06	–	5.39E-08	–
0.62	2.00E-03	0.97	7.63E-06	1.97	4.26E-08	3.02
0.57	1.83E-03	1.02	6.43E-06	2.03	3.29E-08	3.07
0.52	1.67E-03	0.99	5.36E-06	1.99	2.51E-08	2.96
0.47	1.51E-03	1.01	4.37E-06	2.01	1.92E-08	2.63

We then consider two benchmark tests. One is shear flow with the initial condition given by:

$$\begin{cases} \omega(t = 0, x, y) = \delta \cos(x) - \frac{1}{\rho} \operatorname{sech}^2((y - \pi/2)/\rho), & \text{if } y \leq \pi, \\ \omega(t = 0, x, y) = \delta \cos(x) + \frac{1}{\rho} \operatorname{sech}^2((3\pi/2 - y)/\rho), & \text{if } y > \pi, \end{cases} \quad (3.58)$$

where $\delta = 0.05$ and $\rho = \frac{\pi}{15}$. The other is a vortex patch, and the initial condition is

given by:

$$\left\{ \begin{array}{l} \omega(t = 0, x, y) = -1, \quad \text{if } (x, y) \in \left[\frac{\pi}{2}, \frac{3\pi}{2}\right] \times \left[\frac{\pi}{4}, \frac{3\pi}{4}\right], \\ \omega(t = 0, x, y) = 1, \quad \text{if } (x, y) \in \left[\frac{\pi}{2}, \frac{3\pi}{2}\right] \times \left[\frac{5\pi}{4}, \frac{7\pi}{4}\right], \\ \omega(t = 0, x, y) = 0, \quad \text{otherwise.} \end{array} \right. \quad (3.59)$$

In the simulations for the shear flow, we set the mesh as $N_x \times N_y = 128 \times 128$. The third order SLWENO scheme is used to obtain solutions in the IDC prediction step. In Figure 3.31, we report the contours of the numerical solution at time $T = 8$ for the first order splitting scheme and IDC3J2. Little difference can be observed from the contour plot. To better see the difference, in Figure 3.32, 1-D cuts at $x = \pi$ of the numerical solution are reported. We use the solution computed by IDC3J2 with $CFL = 0.05$ as a reference. It is observed that the solution by IDC3J2 with $CFL = 0.67$ matches the reference solution very well, whereas there is a noticeable difference between the solution by the first order scheme with $CFL = 0.67$ and the reference. Then we reduce the CFL to 0.05 for the first order scheme and the corresponding temporal error is reduced. A more precise match is observed. The comparison shows the better performance of coupling the high order IDC methods to correct low order splitting errors. For the vortex patch test, we set the mesh size as $N_x \times N_y = 256 \times 256$. Figure 3.33 gives the numerical solutions at time $T = 5$ (top) and $T = 10$ (bottom) for the first order scheme (IDC3J0) (left) and IDC3J2 (right). The solution structure is observed to be slightly better resolved when the high order IDC framework is used.

3.4. APPLICATIONS TO VLASOV SIMULATIONS

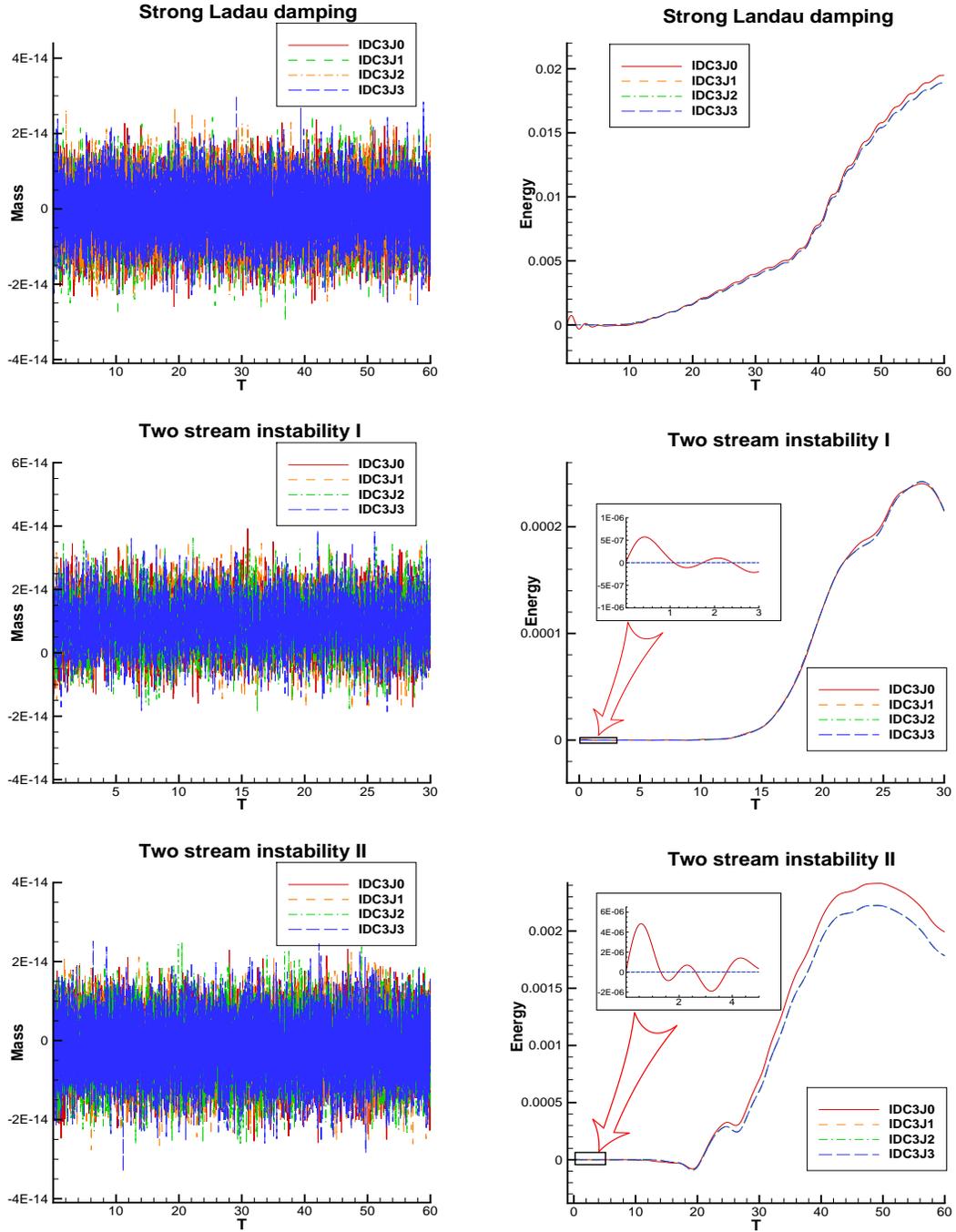


Figure 3.24: The time evolution of the relative deviation in total mass (left) and total energy (right). $N_x \times N_v = 256 \times 256$. CFL = 0.6.

3.4. APPLICATIONS TO VLASOV SIMULATIONS

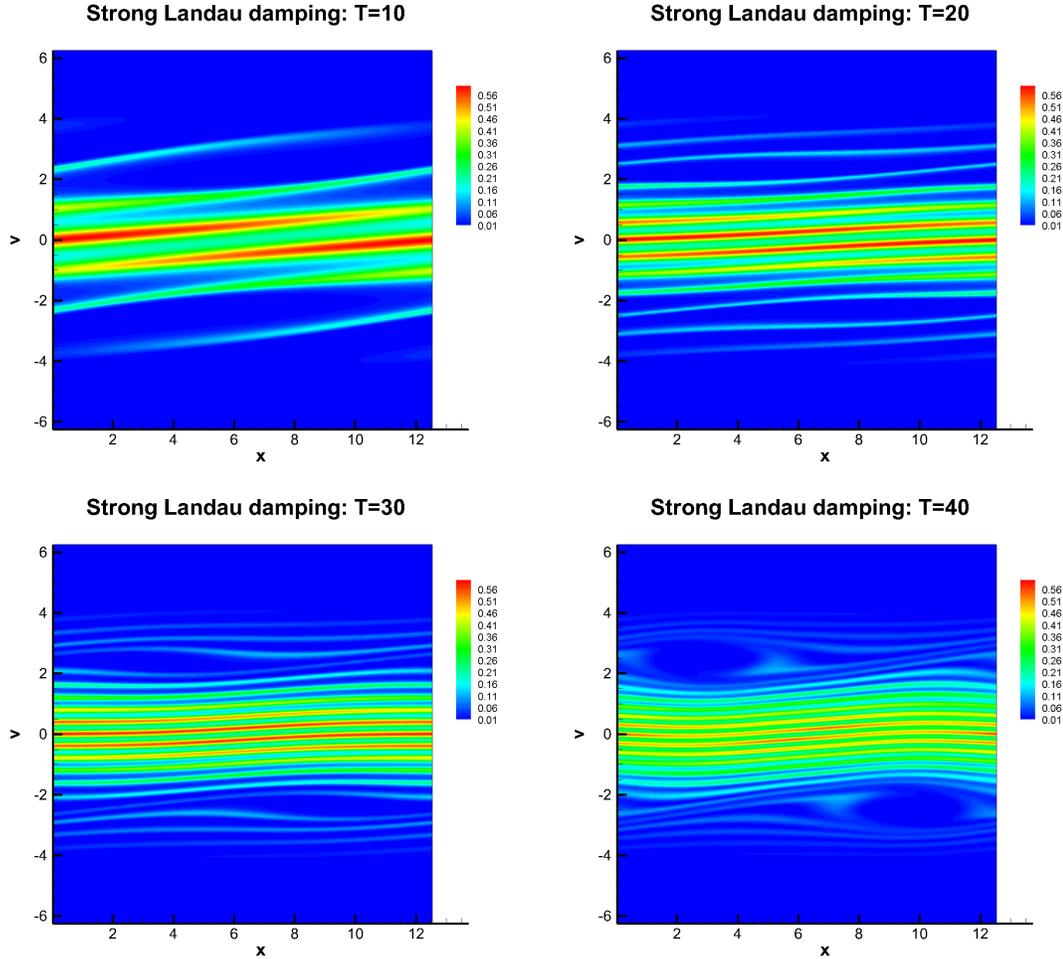


Figure 3.25: Contour plots of the numerical solutions for the strong Landau damping. $N_x \times N_v = 256 \times 256$. CFL = 0.6. IDC3J3.

3.4. APPLICATIONS TO VLASOV SIMULATIONS

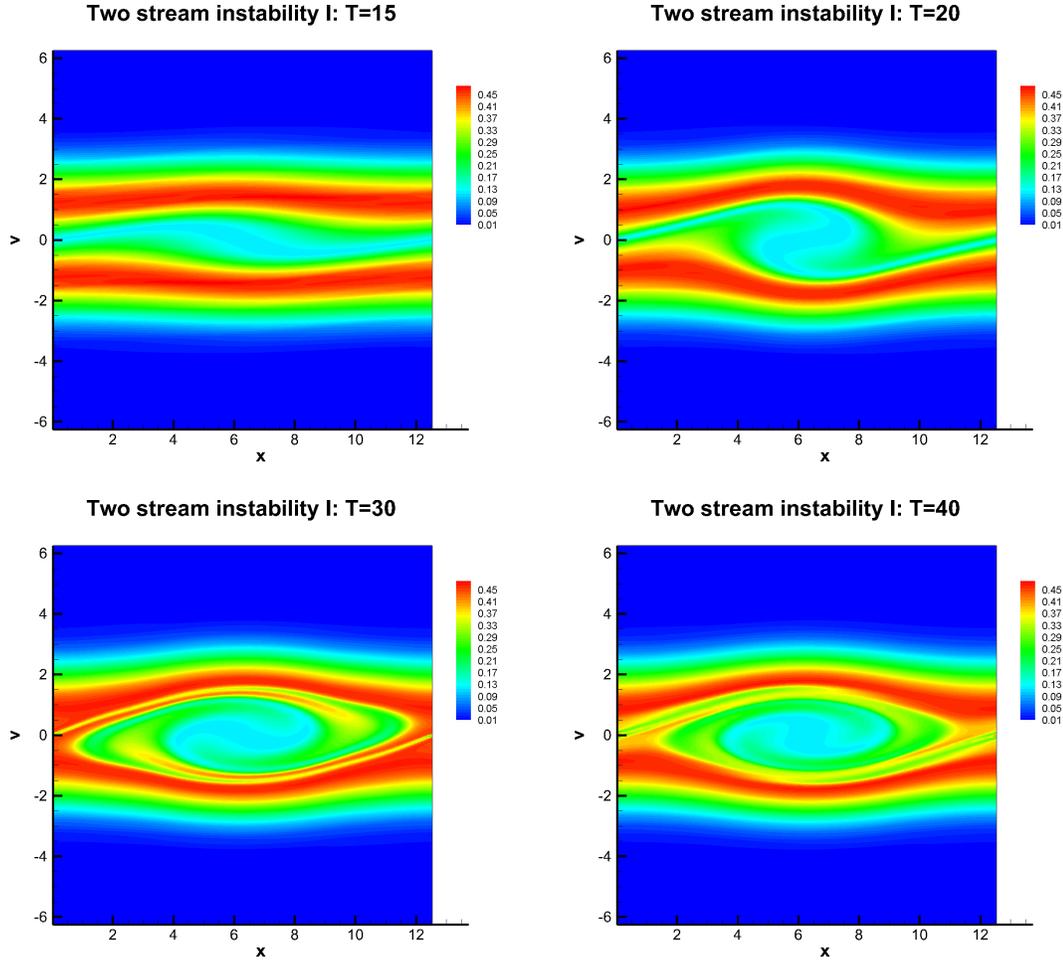


Figure 3.26: Contour plots of the numerical solutions for the two stream instability I. $N_x \times N_v = 256 \times 256$. CFL = 0.6. IDC3J3.

3.4. APPLICATIONS TO VLASOV SIMULATIONS

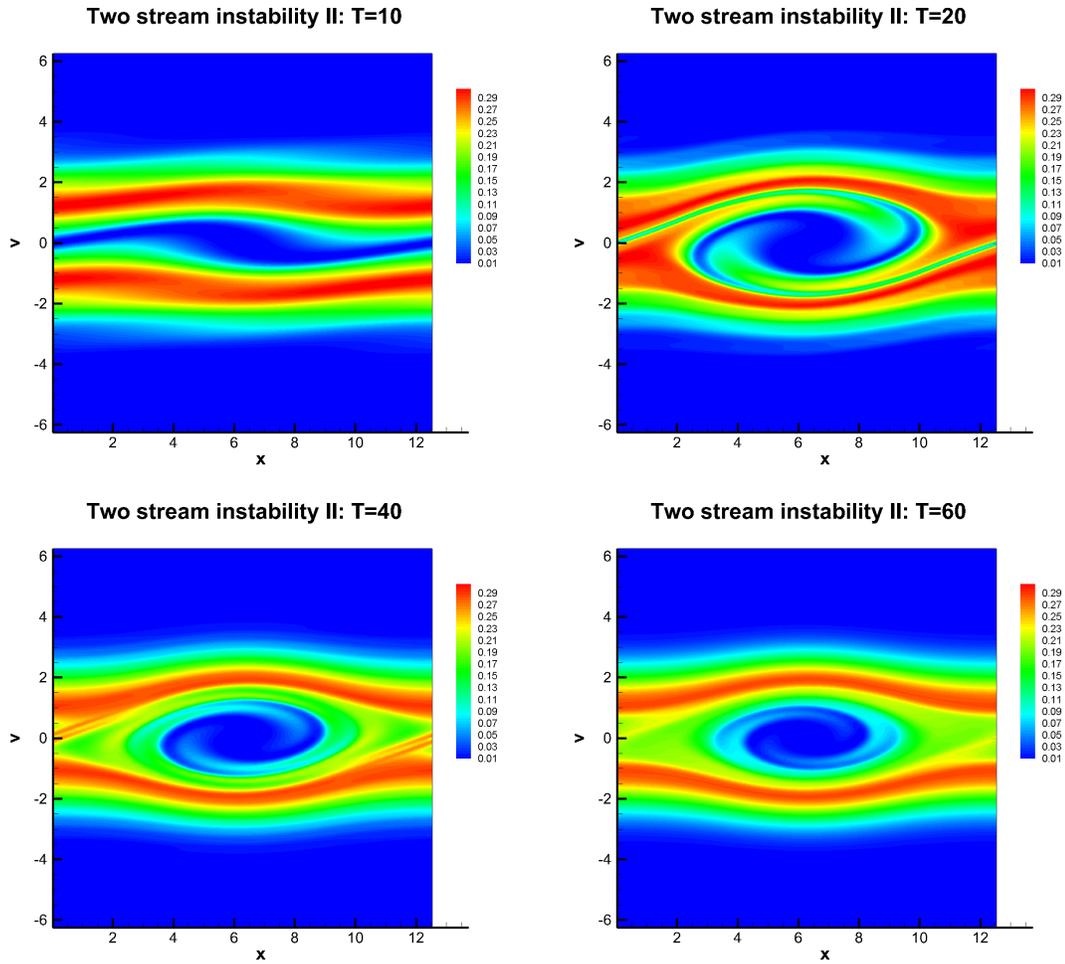


Figure 3.27: Contour plots of the numerical solutions for the two stream instability II. $N_x \times N_v = 256 \times 256$. CFL = 0.6. IDC3J3.

3.4. APPLICATIONS TO VLASOV SIMULATIONS

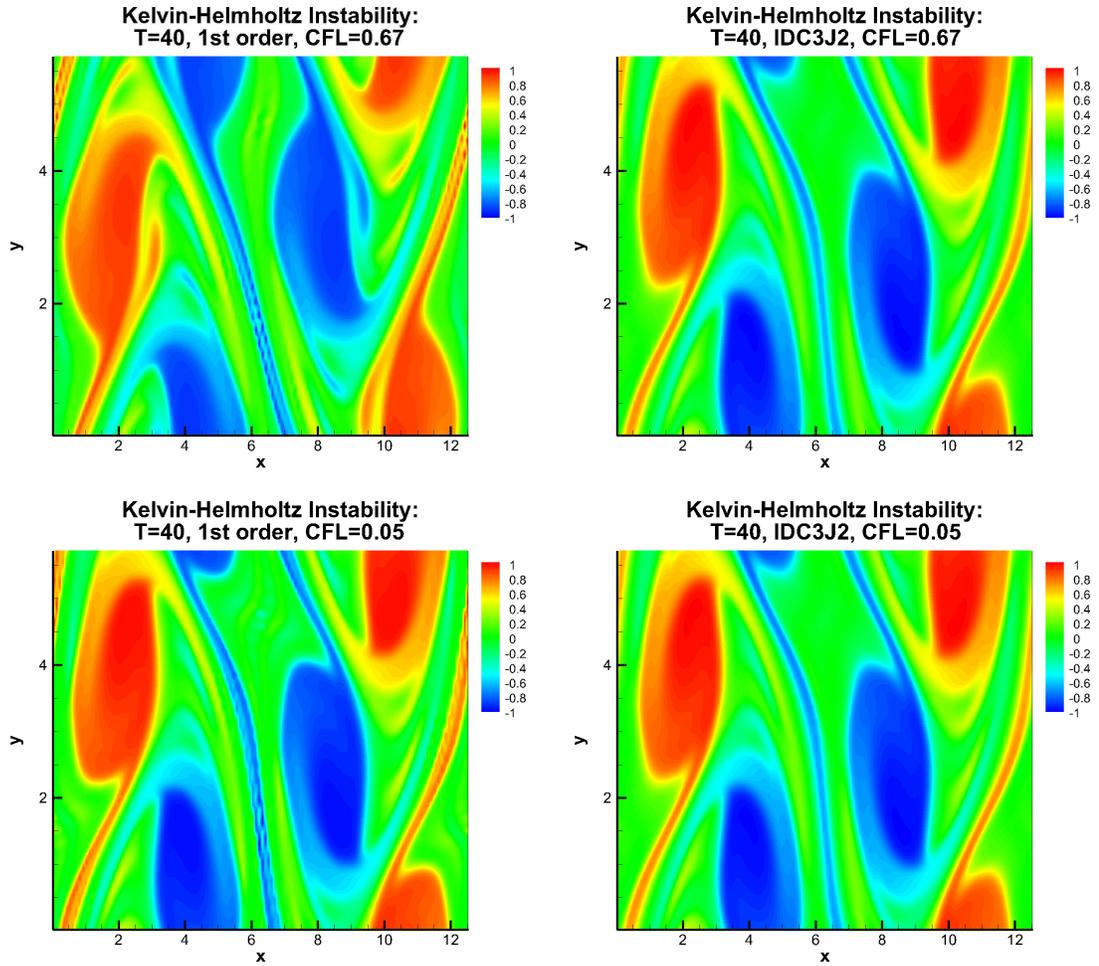


Figure 3.28: Contour plots of the numerical solutions for the Kelvin-Helmholtz instability. $N_x \times N_y = 128 \times 128$. CFL = 0.67 (top) and CFL = 0.05 (bottom) at $T=40$. First order scheme (IDC3J0) (left); Third order scheme (IDC3J2) (right).

3.4. APPLICATIONS TO VLASOV SIMULATIONS

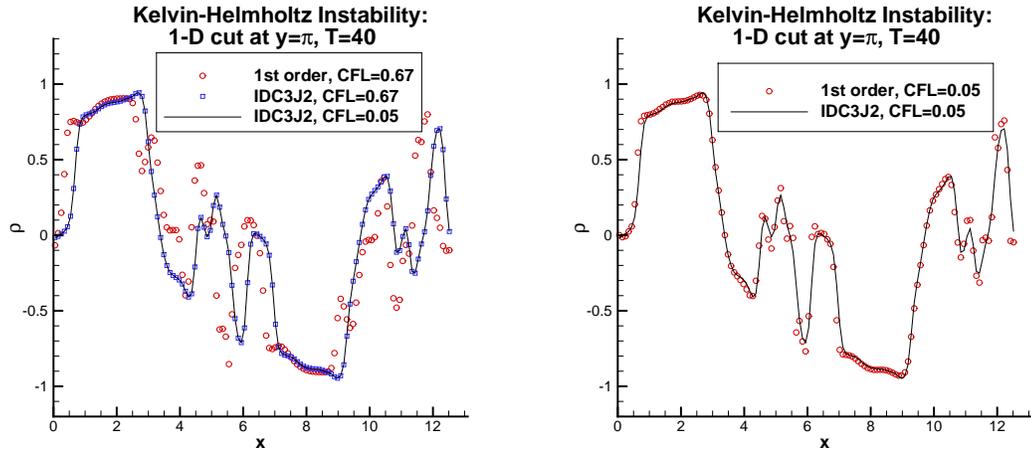


Figure 3.29: 1-D cuts of the numerical solutions at $y = \pi$ for the Kelvin-Helmholtz instability. $N_x \times N_y = 128 \times 128$. $T = 40$. $CFL = 0.67$ (left); $CFL = 0.05$ (right). The reference solution is computed by IDC3J2 with $CFL = 0.05$.

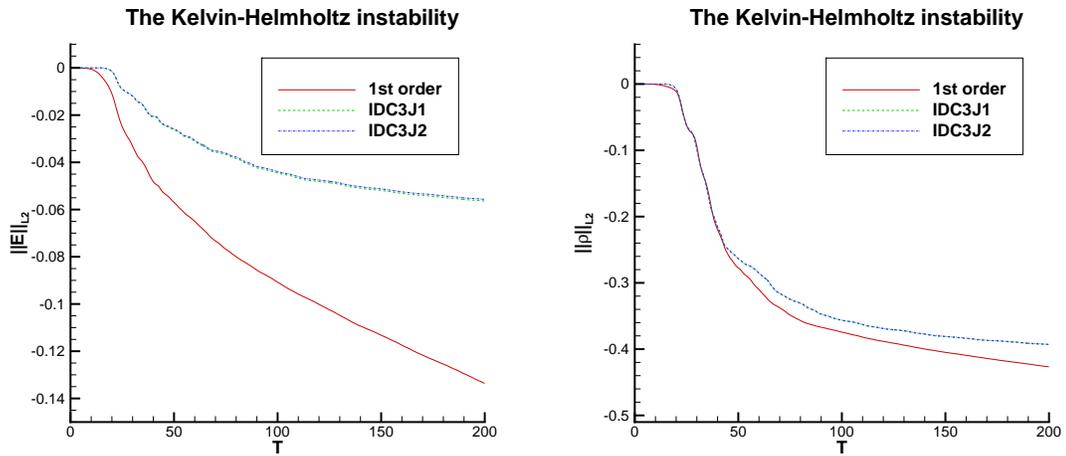


Figure 3.30: The Kelvin-Helmholtz instability. The time evolutions of the energy $\|\mathbf{E}\|_{L^2}$ (left) and the enstrophy $\|\rho\|_{L^2}$ (right). $N_x \times N_y = 128 \times 128$. $CFL = 0.67$.

3.4. APPLICATIONS TO VLASOV SIMULATIONS

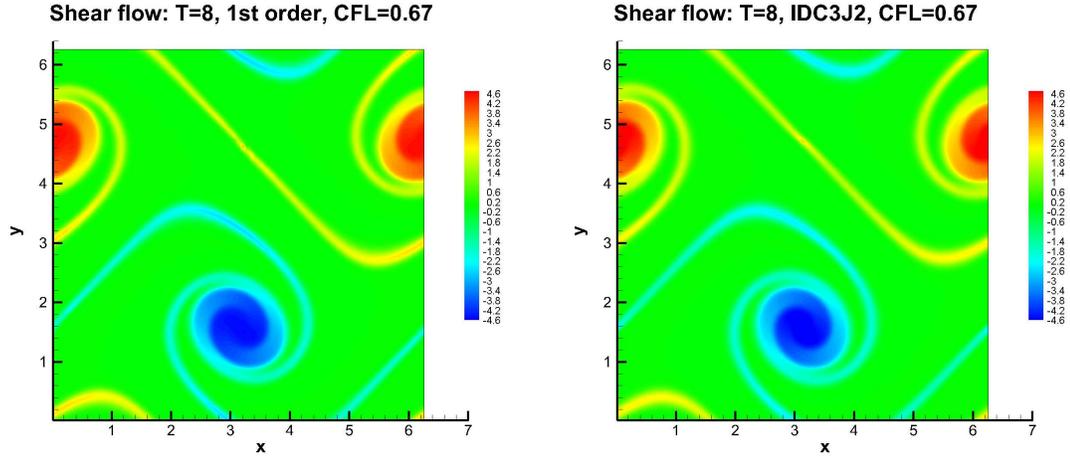


Figure 3.31: Contour plots of the numerical solutions for the shear flow test. $N_x \times N_y = 128 \times 128$. CFL = 0.67 at T= 8. First order scheme (IDC3J0) (left); Third order scheme (IDC3J2) (right).

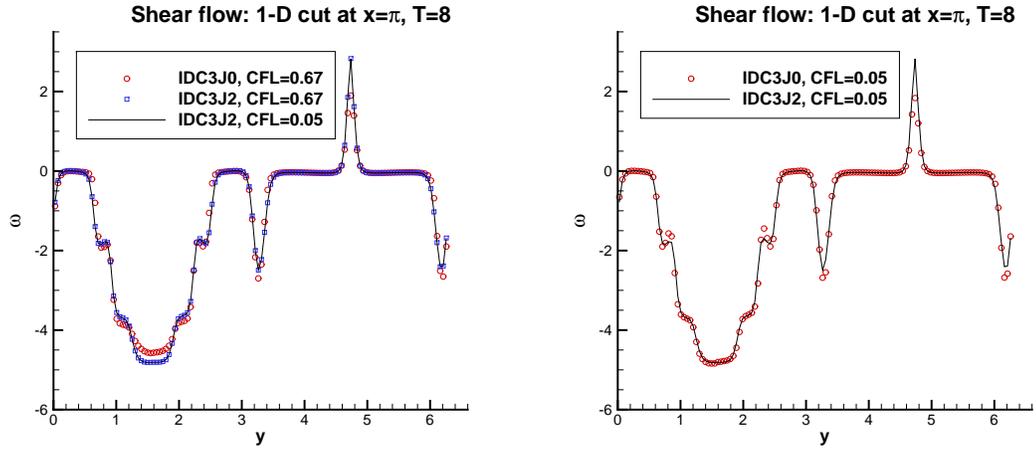


Figure 3.32: 1-D cuts of the numerical solutions at $x = \pi$ for the shear flow test. $N_x \times N_y = 128 \times 128$. T= 8. CFL = 0.67 (left); CFL = 0.05 (right). The reference solution is computed by IDC3J2 with CFL = 0.05.

3.4. APPLICATIONS TO VLASOV SIMULATIONS

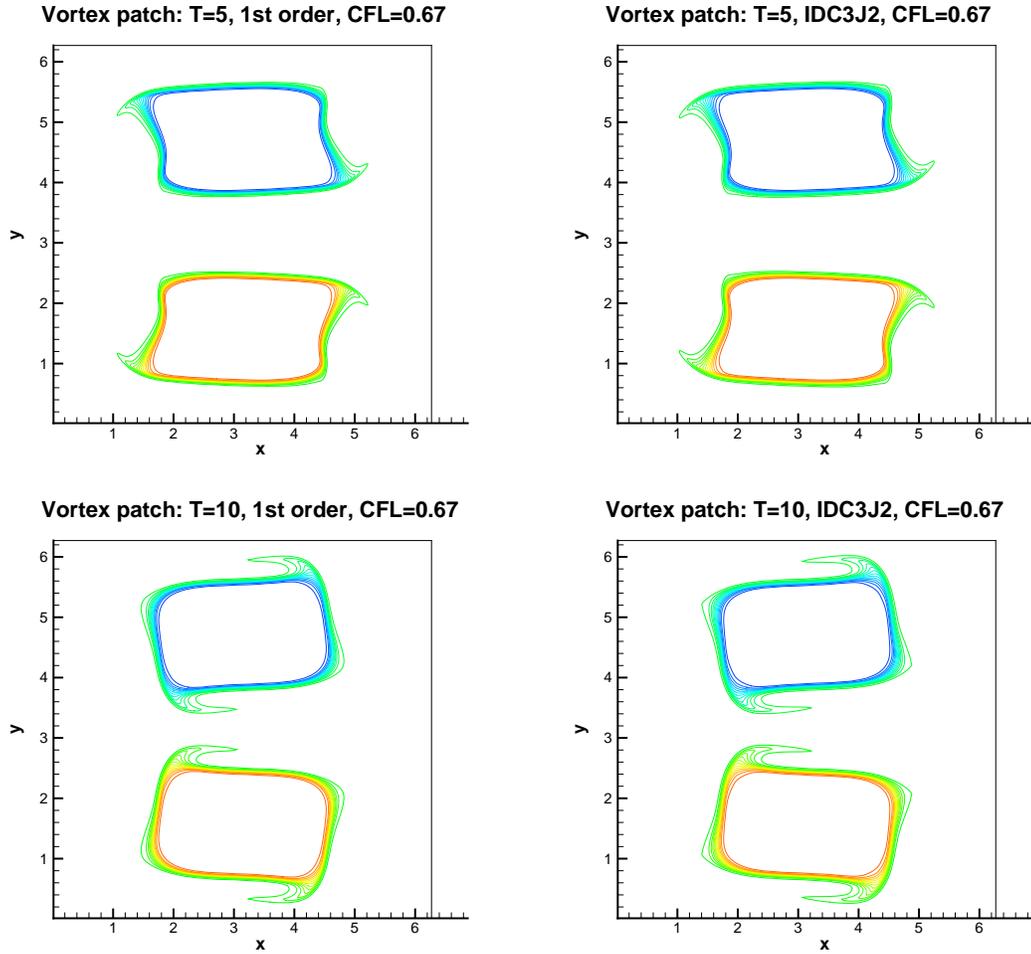


Figure 3.33: Contour plots of the numerical solutions for the vortex patch test. $N_x \times N_y = 256 \times 256$. CFL = 0.67 at T= 5 (top) and T= 10 (bottom). First order scheme (IDC3J0) (left); Third order scheme (IDC3J2) (right). 30 equally spaced contours from -1.1 to 1.1.

3.5 Applications to Global Transport

In an operational climate model, the total computational expense is dominated by that of the tracer transport scheme, which is accountable for $\mathcal{O}(100)$ tracer species including several moisture variables. Recently, [37] showed that the multi-tracer transport scheme CSLAM [62] based on the finite-volume SL philosophy is an efficient alternative to the native SE transport scheme based on the Eulerian approach in CAM framework. As the number of tracer species increases to more than six or so, the semi-Lagrangian scheme becomes significantly more efficient. This is due to the fact that once the upstream trajectory and other geometric information are computed, they can be reused for each tracer field. Although the semi-Lagrangian scheme can take a larger time step, a *moderate* value $CFL \approx 1$ would be desirable to maintain the parallel efficiency. However, the CSLAM scheme employs its own uniform finite-volume cells within each spectral element defined by highly non-uniform GL quadrature points [37]. This necessitates two grids system, one for the SE dynamics (GL) and the other one for the CSALM transport, requiring a grid-to-grid remapping.

The objective of this section is to develop a conservative SLDG scheme based on a dimensional splitting strategy on the cubed-sphere geometry. The scheme is particularly designed for a nodal DG discretization employing GL grids with the CFL number approximately 1. The SLDG scheme can be directly implemented for SE grids, and does not require two grids system as in the case of SE and CSLAM combination.

3.5.1 Cubed-Sphere Geometry

The cubed-sphere geometry [100, 95] is free from polar singularities, and well-suited for element-based Galerkin methods. Below, we consider the cubed-sphere mesh generated by an equiangular central projection as described in [76]. In the cubed-sphere grid system, the spherical domain is partitioned into six identical regions (faces). On each face, gridlines follow nonorthogonal curvilinear coordinate system (x^1, x^2) such that $x^1, x^2 \in [-\pi/4, \pi/4]$, see Figure 1.1. Each face of the cubed sphere is tiled with $N_e \times N_e$ elements (cells) so that $6 \times N_e^2$ elements span the entire spherical domain. See Figure 1.1 for the relative position of six faces and the equiangular central projection from the Cartesian mesh on the cube to the curvilinear mesh on the sphere with $N_e = 5$. The metric tensor associated with the equiangular central (gnomonic) mapping is given by

$$g_{ij} = \frac{R^2}{\rho^4 \cos^2 x^1 \cos^2 x^2} \times \begin{bmatrix} 1 + \tan^2 x^1 & -\tan x^1 \tan x^2 \\ -\tan x^1 \tan x^2 & 1 + \tan^2 x^2 \end{bmatrix}, \quad (3.60)$$

where $i, j, \in \{1, 2\}$, $\rho = 1 + \tan^2 x^1 + \tan^2 x^2$ and R is radius of the sphere. Denote $g = \det(g_{ij})$, then the Jacobian of the transformation is given by \sqrt{g} , which is identical on each face. We refer to [76] for all the conversion formulas between the usual Lat-Lon velocity components a, b and the contravariant components a^1, a^2 on the cubed sphere.

3.5.2 A Semi-Lagrangian DG Scheme on the Cubed Sphere

The transport equation for a scalar u on the cubed sphere can be re-written in (x^1, x^2) -space on each face as follows [123]:

$$\frac{\partial U}{\partial t} + \frac{\partial F_1(U)}{\partial x^1} + \frac{\partial F_2(U)}{\partial x^2} = 0, \quad (3.61)$$

where $U = u\sqrt{g}$, and the fluxes $F_1 = a^1 U$, and $F_2 = a^2 U$; the Jacobian \sqrt{g} is a given continuous function of the curvilinear coordinate transform. For the DG spatial discretization, each element is further mapped onto $N_p \times N_p$ GL grids. Figure 1.1 (bottom) shows a cubed-sphere discretized with uniform size elements with $N_e = 5$ and $N_p = 4$. The equation (3.61) is in a conservative form, similar to equation (3.19) for a Cartesian mesh. Therefore, the SLDG scheme can be implemented in a similar fashion as that for a Cartesian mesh previously described. As we directly work with U , the mass conservation property of numerical solution is preserved. Below, we focus our description on the differences of implementation.

The cubed-sphere grid lines may be interpreted as three families of piecewise closed great-circle arcs (ξ, η, ζ) on the sphere, see Figure 1.1 and Figure 3.34. We exploit this idea for solving transport equations on the cubed sphere.

Due to this special geometric feature of the cubed-sphere grid system, there exists three logical transport directions denoted as ξ -, η -, and ζ -directions as shown in Figure 3.34. Specifically,

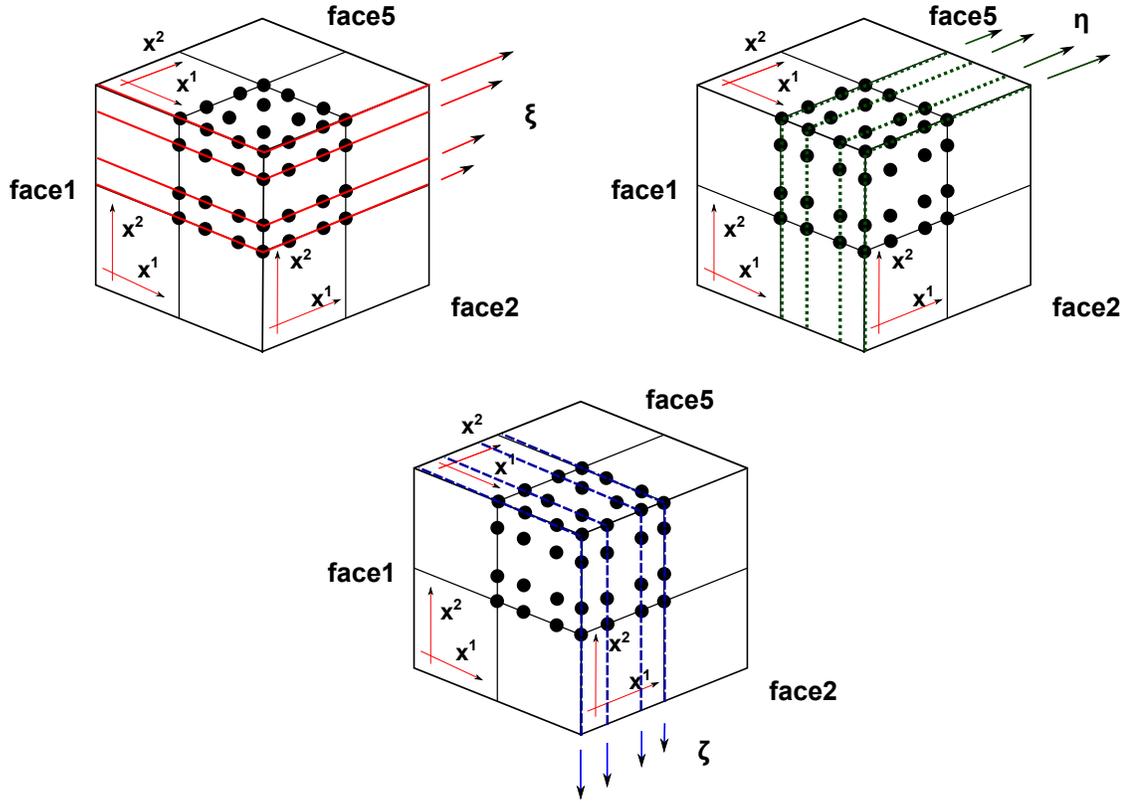


Figure 3.34: Schematic showing of the SLDG scheme on mesh $2 \times 2 \times 6$ of the cubed-sphere geometry. 4×4 GL points per element are used as an example. Advection of ξ -direction (upper left) ; advection of η -direction (upper right); advection of ζ -direction (bottom).

- ξ is the direction along the x^1 -axis of face1, face2, face3, and face4, see Figure 1.1 and Figure 3.34 (upper left).
- η is the direction along the x^2 -axis of face1, face3, face5, and face6, see Figure 3.34 (upper right);
- ζ is the direction along the x^2 -axis of face2, and face4, and along the x^1 -axis of face5 and face6, see Figure 3.34 (bottom).

3.5. APPLICATIONS TO GLOBAL TRANSPORT

In the (ξ, η, ζ) coordinate, the transport equation (3.61) on the cubed sphere can be re-written as

$$\frac{\partial U}{\partial t} + \frac{\partial \tilde{F}_1(U)}{\partial \xi} + \frac{\partial \tilde{F}_2(U)}{\partial \eta} + \frac{\partial \tilde{F}_3(U)}{\partial \zeta} = 0, \quad (3.62)$$

where $\tilde{F}_1, \tilde{F}_2, \tilde{F}_3 \in \{F_1, F_2\}$ such that (3.61) and (3.62) are equivalent on any given face. We note that the unknown function U *only* depends on two variables (x^1, x^2) on each cube face. For example, U is dependent only on ξ (x^1) and η (x^2) on face1 and only on ξ (x^1) and ζ (x^2) on face2, also see Figure 3.34. Thus, equation (3.62) is essentially identical to (3.61) on each face. A second order Strang-type splitting strategy for multiple operators proposed in [42] can be used to equation (3.62). We summarize the procedure as follows:

1. The equation (3.62) is split into three 1-D advection problems on the cubed sphere:

$$\frac{\partial U}{\partial t} + \frac{\partial \tilde{F}_1(U)}{\partial \xi} = 0, \quad (3.63)$$

$$\frac{\partial U}{\partial t} + \frac{\partial \tilde{F}_2(U)}{\partial \eta} = 0, \quad (3.64)$$

$$\frac{\partial U}{\partial t} + \frac{\partial \tilde{F}_3(U)}{\partial \zeta} = 0. \quad (3.65)$$

2. The numerical solution is updated by a Strang-type splitting strategy for one time step Δt :

- (a) Evolve 1-D equation (3.63) in the direction ξ for $\Delta t/2$, see Figure 3.34

- (upper left);
- (b) Evolve 1-D equation (3.64) in the direction η for $\Delta t/2$, see Figure 3.34 (upper right);
- (c) Evolve 1-D equation (3.65) in the direction ζ for Δt , see Figure 3.34 (bottom);
- (d) Evolve 1-D equation (3.64) in the direction η for $\Delta t/2$ as (b);
- (e) Evolve 1-D equation (3.63) in the direction ξ for $\Delta t/2$ as (a).

The evolution of each 1-D equation follows a similar procedure as that for the Cartesian mesh, except for the tracing of characteristics across face edges. As before, we employ a fourth order RK method for solving the characteristic equations, e.g.,

$$\frac{dx^1(t)}{dt} = a^1(x^1(t), x_{jg}^2, t), \quad (3.66)$$

where x_{jg}^2 is a fixed GL point in x^2 -direction. Below we only demonstrate the initial value problem case (forward trajectory), while the backward case is similar. Note that special treatment is needed, since the velocity a^1 takes different expressions on different faces. Below is the procedure implemented in our code in the case of characteristics emanating from (x^*, t^n) crossing the edge of face1 and face2, see Figure 3.35.

1. Find the time point $t^* \in [t^n, t^{n+1}]$, when the trajectory reaches the face edge, denoted as x_e . The following gives a second order way of approximating t^* when the u_s^1 ($s = 1$ or 2 being the index for a face) is time independent, see

Figure 3.35 (left):

$$\begin{aligned} x_e - x^* &= \frac{t^* - t^n}{2}(a_1^1(x_e) + a_1^1(x^*)), \quad \text{or} \\ t^* &= t^n + \frac{2(x_e - x^*)}{a_1^1(x_e) + a_1^1(x^*)}. \end{aligned} \quad (3.67)$$

In some practical applications, such as the multi-tracer transport, u_s^1 is only given at t^n , so it is reasonable to assume that a_s^1 is constant in $[t^n, t^{n+1}]$. In the time dependent case, assume we have a high order RK method (e.g. a fourth order RK method) to solve the initial value problem (3.66) with $x^1(t^n) = x^*$. Denote the numerical solution of equation (3.66) at time t as $RK(t; x^*, t^n)$, then we want to find t^* such that

$$RK(t^*; x^*, t^n) = x_e. \quad (3.68)$$

We adopt the Newton's type method to solve (3.68):

(a) Using (3.67) to get:

$$t^{*,0} = t^n + \frac{2(x_e - x^*)}{a_1^1(x_e, t^n) + a_1^1(x^*, t^n)},$$

which is a good prediction of t^* .

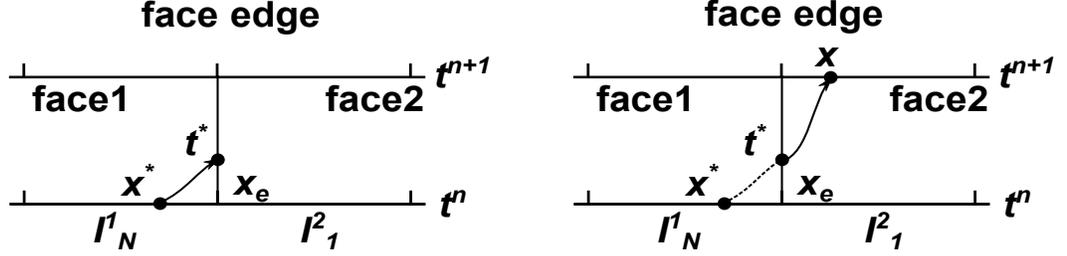


Figure 3.35: Get the arrival point x of the SLDG scheme at the edge of face1 and face2. Find the time t^* when the trajectory emanating from x^* reaches the edge (left); get the arrival point x starting at time t^* (right).

(b) Set a threshold ϵ and do the following iteration:

$$x_e^j = RK(t^{*,j}; x^*, t^n);$$

If $|x_e^j - x_e| < \epsilon$, stop, and let $t^* = t^{*,j}$,

else

$$t^{*,j+1} = t^{*,j} + \frac{x_e^j - x_e}{a_1^1(x_e^j, t^{*,j})}.$$

In the simulation, it takes about 3-5 iterations to reach an error tolerance of $\epsilon = 10^{-10}$.

2. Continue evolving the characteristic curve on face2 to locate the arrival point x at the next time step t^{n+1} , see Figure 3.35 (right).

Numerical Results

In the following, we consider two types of 2-D spherical advection tests for the

SLDG scheme on the sphere. The tests include a solid body rotation and two deformation flow tests.

Example 3.22. (Solid body rotation of a cosine bell.) Solid body rotation of a cosine bell is a widely used standard test for 2-D spherical advection problem [112]. The initial scalar distribution (cosine bell) is defined by

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

where r_d is the great-circle distance between (λ, θ) and the bell center which is $(3\pi/2, 0)$ at $t = 0$, $h_0 = 1000\text{m}$ is the maximum height of the cosine bell, and $r_0 = R/3$ represents the radius of the bell, here $R = 6.37122 \times 10^6$ is the earth's radius. The wind components in the longitudinal (λ) and latitudinal (θ) directions are defined as follows:

$$\begin{aligned} a &= a_0(\cos \alpha \cos \theta + \sin \alpha_0 \cos \lambda \sin \theta), \\ b &= -a_0 \sin \alpha \sin \lambda, \end{aligned}$$

where $a_0 = 2\pi R/(12 \text{ days})$ and α is the rotation angle, which is between the axis of the solid body rotation and the polar axis of the spherical coordinate. The flow is oriented along the equatorial direction when $\alpha = 0$ and the northeast direction when $\alpha = \pi/4$. Note that the configuration with $\alpha = \pi/4$ is more challenging for the cubed-sphere geometry. In this case, the cosine bell goes through four vertices, two edges and all six faces. The wind field is non-divergent, which means the maximum principle holds. The exact solution is available at all times and the cosine bell

reaches its initial state after a complete (12 days) rotation, thus error measures can be computed.

We apply the SLDG P^3 scheme to the solid body rotation problem with mesh $20 \times 20 \times 6$ corresponding to 1.5° resolution at the equator for the cubed-sphere geometry. The time step is set as $\Delta t = 3600$ sec, which is 6 times larger than that used by the RKDG P^2 scheme in [123]. In Table 3.16, we report the standard normalized error norms based on [112] with $\alpha = 0$ and $\alpha = \pi/4$. The error measured are comparable to those by the RKDG scheme in [76, 123]. Note that the proposed scheme is exactly mass conservative. In Figure 3.36, the contour plots of the numerical solution are reported for $\alpha = \pi/4$. The results are observed to be comparable to those produced by a non-oscillatory RKDG scheme presented in [123]. The evolution of error norms are given in Figure 3.37 for $\alpha = 0$ (panel (a) and (b)) and $\alpha = \pi/4$ (panel (c)-(f)). Note that the L^∞ error grows significantly when a cosine bell goes through a corner for $\alpha = \pi/4$ with $\Delta t = 3600$ sec. However, the L^∞ error drops back quickly when it is away from the corner, see Figure 3.37 (panel (c) and (d)). If the time step Δt is reduced to 1800 sec, the L^∞ error is also reduced, see Figure 3.37 (panel (e) and (f)). This indicates that the L^∞ error around corners comes from time discretization. In Figure 3.38 (top), numerical error is plotted when the cosine bell approaches (left), reaches (middle) and passes (right) a corner on the cubed sphere. It is observed that error magnitude at cube edges is much larger than elsewhere. The error grows as the peak of the cosine bell approaches a corner on the cubed sphere; the error starts to drop as the peak passes the corner. We then consider a different ordering for dimensional splitting: we first evolve (3.65), then (3.63), and finally

3.5. APPLICATIONS TO GLOBAL TRANSPORT

Table 3.16: Normalized standard errors for ϕ , for the 2-D solid body rotation test with $\alpha = 0$ and $\alpha = \pi/4$ on the cubed-sphere geometry. The SLDG P^3 scheme is used on a $20 \times 20 \times 6$ (1.5° resolution) and time step is set as $\Delta t = 3600$ sec. The numerical solution is computed after a full rotation.

	Scheme	L^1	L^2	L^∞	mass error
$\alpha = 0$	SLDG	$1.04E-02$	$7.03E-03$	$6.53E-03$	$-5.20E-13$
	SLDG + BP	$8.50E-03$	$5.82E-03$	$6.72E-03$	$-5.20E-13$
$\alpha = \pi/4$	SLDG	$1.17E-02$	$7.70E-03$	$7.20E-03$	$-4.13E-13$
	SLDG + BP	$8.93E-03$	$6.06E-03$	$7.46E-03$	$-4.12E-13$

(3.64). Similar error patterns but with opposite signs are observed in Figure 3.38 (bottom). This is an indication that such error comes from the dimensional splitting, and there exist certain symmetry property for different dimensional splitting orderings. Such symmetry property, together with the symmetry of the cosine bell profile, may contribute to the dropping of the L^∞ error after the cosine bell passed the corner.

We compare the proposed SLDG scheme with the RKDG scheme and the semi-Lagrangian “CSLAM” scheme by [62], in terms of error norms. In Table 3.17, we show the comparison between the SLDG scheme and the RKDG scheme when P^3 is used. The mesh is set as $30 \times 30 \times 6$ ($N_e = 30$) which corresponds to approximately 1° resolution at the equator. Note that the SLDG scheme can take a very large time step but the RKDG scheme suffers from the time step restriction (1400 sec is nearly the limit for time step in this case). The CPU time for the SLDG scheme is three times smaller than that of the RKDG scheme. We remark that the computational cost per time step of the SLDG scheme is larger than that of the RKDG scheme. It is due to the need to evaluate volume integrals in several sub-intervals for one

3.5. APPLICATIONS TO GLOBAL TRANSPORT

Table 3.17: Comparison between the RKDG scheme and the SLDG scheme in terms of error norms and CPU time when solving the solid body rotation of a cosine bell. The mesh is set as $30 \times 30 \times 6$ corresponding 1° resolution. DG P^3 is adopted. The numerical solutions are computed after a full rotation.

	Scheme	time step	CPU time	L^2	L^∞
$\alpha = 0$	RKDG P^3	1440 sec	9.76 sec	$1.14E-02$	$7.55E-03$
	SLDG P^3	7200 sec	3.32 sec	$3.70E-03$	$4.33E-03$
	SLDG P^3	3600 sec	6.31 sec	$2.80E-03$	$3.12E-03$
$\alpha = \pi/4$	RKDG P^3	1440 sec	9.76 sec	$1.21E-02$	$7.97E-03$
	SLDG P^3	7200 sec	3.32 sec	$1.72E-02$	$3.56E-02$
	SLDG P^3	3600 sec	6.31 sec	$5.15E-03$	$9.25E-03$

element, see equation (3.9). We also remark that the SL scheme is significantly efficient for the multi-tracer transport because the geometric information such as upstream trajectories can be reused for each field. In spite of the larger time steps used with the SLDG scheme, the error norms are still comparable to those of the RKDG scheme. This shows the SLDG scheme to be more efficient when solving the solid body rotation problem. Finally, we compare the SLDG scheme with the CSLAM [62], when the horizontal resolution and the time step are comparable. For this experiment, the mesh for the SLDG P^3 scheme is set to $10 \times 10 \times 6$ ($N_e = 10$), which corresponds to 3° resolution, and a $32 \times 32 \times 6$ mesh is set for the CSLAM scheme, which corresponds to 2.8125° resolution. Note that we use a little lower resolution mesh for the SLDG scheme. Table 3.18 shows the error norms of the two schemes performing the solid body rotation of a cosine bell with $\alpha = 0$ and $\alpha = \pi/4$. The time step is set as 4050 sec. It is observed that the error norms of SLDG P^3 are a little smaller than those of CSLAM. The comparison with the two popular transport schemes as discussed above shows the SLDG scheme is very competitive.

3.5. APPLICATIONS TO GLOBAL TRANSPORT

Table 3.18: Comparison between SLDG P^3 and CSLAM [62] in terms of error norms when solving the solid body rotation of a cosine bell. The mesh is set as $10 \times 10 \times 6$ for SLDG corresponding to 3° resolution and $32 \times 32 \times 6$ for CSLAM corresponding to 2.8125° resolution. The numerical solutions are computed after a full rotation with time step $\Delta t = 4050$ sec.

	Scheme	L^1	L^2	L^∞
$\alpha = 0$	SLDG P^3	$7.52E-02$	$4.20E-02$	$3.31E-02$
	CSLAM	$7.9E-02$	$4.6E-02$	$3.4E-02$
$\alpha = \pi/4$	SLDG P^3	$7.15E-02$	$3.66E-02$	$2.25E-02$
	CSLAM	$7.6E-02$	$4.1E-02$	$2.5E-02$

Example 3.23. (Deformational flow on the sphere: moving-vortex problem.) The second test is a challenging deformational flow test, the moving-vortex problem, proposed by [73]. The test represents the roll-up of an idealized moving atmospheric vortex such as hurricane or tropical cyclone [48]. In this test case, two vortices are generated located on the diametrically opposite sides of the sphere. The wind field is a combination of wind vectors of the solid body rotation, which is considered in the previous case, and that of the deformational flow. The two vortices move along a great circle and the exact solution is available at any time. In a rotated coordinate system (λ', θ') , the scaled tangential velocity V_t of the vortex field is defined by

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

where $\rho = \rho_0 \cos(\theta')$ is the radial distance of the vortex with the parameters $\rho_0 = 3$, and rotational speed $u_0 = 2\pi R/12$ (days) such that 12 days are required for a full

3.5. APPLICATIONS TO GLOBAL TRANSPORT

vortex evolution. The associated angular velocity is defined to be

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

The exact solution in rotated coordinates is

$$u(t, \lambda', \theta') = 1 - \tanh \left[\frac{\rho}{\gamma_0} \sin(\lambda' - \omega(\theta')t) \right], \quad (3.69)$$

here parameter $\gamma_0 = 5$. The time dependent wind field for the moving vortex is given by

$$\begin{aligned} a(t) &= a_0(\cos \theta \cos \alpha + \sin \theta \cos \lambda \sin \alpha) \\ &\quad + R\omega(\theta')[\sin \theta_c(t) \cos \theta - \cos \theta_c(t) \cos(\lambda - \lambda_c(t)) \sin \theta], \\ b(t) &= -a_0 \sin \lambda \sin \alpha + R\omega(\theta')[\cos \theta_c(t) \sin(\lambda - \lambda_c(t))], \end{aligned}$$

where α is the flow orientation parameter described earlier and $(\lambda_c(t), \theta_c(t))$ is the center of one of the moving vortices, which is directed along a great circle trajectory. For the current test, the initial vortex center is located at $(\lambda_c(0), \theta_c(0)) = (3\pi/2, 0)$, which is also the location of the north pole of the rotated sphere.

We applied the SLDG P^3 scheme to the moving-vortex problem with a mesh $30 \times 30 \times 6$ (1° resolution) on the cubed-sphere geometry. The time step is set as $\Delta t = 3600$ sec which is six times larger than that used in [123]. In Figure 3.39, the evolution of numerical solution is shown for $\alpha = \pi/4$ as a series of orthographic projections

centered on one of the vortices. The initial condition is shown in Figure 3.39a and the numerical solution at day 3, day 6 and day 12 are shown in Figure 3.39b, Figure 3.39c, Figure 3.39d, respectively. The numerical solutions are visually indistinguishable from the exact solution, which is not shown for saving space. At this resolution, the SLDG scheme resolves the fine filaments of the vortex field and is comparable to the results shown in Figure 2 of [73] and Figure 8 of [82]. When approximated to two decimal places, the normalized standard L^1 , L^2 and L^∞ errors are $4.40E-4$, $1.06E-3$ and $9.55E-3$, respectively. The histories of error norms evolution are plotted in Figure 3.40 for $\alpha = 0$ and $\alpha = \pi/4$. A similar phenomenon is observed that the L^∞ error norm grows when the vortices go through the corners, then it drops back when they are away from the corners.

We employ the SLDG scheme with high order elements (P^6) to study its performance. For this test a mesh $15 \times 15 \times 6$ is adopted, which corresponds to 1° resolution on the cubed-sphere geometry. The time step is set to be 3600 sec as before. The numerical solution is visually indistinguishable from the exact solution, so it is not shown to save space. Here we give the normalized standard L^1 , L^2 and L^∞ errors after 12 model days. They are $3.53E-4$, $7.28E-4$ and $8.28E-3$, respectively. Note that the error norms are a little smaller than those computed by the SLDG P^3 scheme with the same resolution, this is an expected behavior of high order methods with smooth problems.

Example 3.24. (Deformational flow on the sphere: slotted-cylinder.) The last benchmark test we consider is a challenging test from a class of deformational flow tests proposed by [74]. This test is quite challenging for any advection scheme on

3.5. APPLICATIONS TO GLOBAL TRANSPORT

the sphere because the flow field is extremely deformational (non-divergent) with a non-smooth initial condition. Since the analytic solution is available at the final time, errors measures can be made available for comparison.

The non-divergent wind field is defined to be

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

where $\lambda' = \lambda - 2\pi t/T$, $\kappa = 2$ and $T = 5$ units. Note that the wind field is a combination of a deformational field and a zonal background flow, avoiding the possible cancellations of errors due to the reversal of the flow along the same flow path after the half time $T/2$, see [74]. The initial condition is the twin slotted-cylinder defined by:

$$u(t = 0, \lambda, \theta) = \begin{cases} c & \text{if } r_i \leq r \text{ and } |\lambda - \lambda_i| \geq r/6 \text{ for } i = 1, 2, \\ c & \text{if } r_1 \leq r \text{ and } |\lambda - \lambda_1| < r/6 \text{ and } \theta - \theta_1 < -\frac{5}{12}r, \\ c & \text{if } r_2 \leq r \text{ and } |\lambda - \lambda_2| < r/6 \text{ and } \theta - \theta_2 > \frac{5}{12}r, \\ d & \text{otherwise,} \end{cases} \quad (3.70)$$

where $c = 1$, $d = 0$, $r = 1/2$ is the radius of the cylinder and $r_i = r_i(\lambda, \theta)$ is the great-circle distance between (λ, θ) and a specified center (λ_i, θ_i) of the cylinder, which is given by

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

The centers of the initial distribution are located at $(\lambda_1, \theta_1) = (5\pi/6, 0)$ and $(\lambda_2, \theta_2) =$

3.5. APPLICATIONS TO GLOBAL TRANSPORT

$(7\pi/6, 0)$, respectively. The slots are oriented in opposite directions for the two cylinders so that they are symmetric with respect to the flow. The wind field and initial distributions are defined in non-dimensional units on the unit sphere ($R = 1$). Note that the distribution is stretched into thin filaments and at half time $T/2$ while they are being transported along the zonal direction by the background flow. The exact solution is only available at the final time $t = T$ which is identical to the initial condition.

We apply the SLDG P^3 scheme with the BP filter to the deformational flow problem with a mesh $30 \times 30 \times 6$ corresponding to 1° resolution at the equator for the cubed-sphere geometry. The time step is set as $\Delta t = T/800$ such that it takes 800 step to complete a full evolution. Note that Δt is chosen as 5 times larger than that used for the RKDG P^2 scheme in [123]. Figure 3.41 shows the initial condition (a), the numerical solution at half time $t = T/2$ (b) and final time $t = T$. It is clearly observed that the SLDG scheme resolves the very thin filament solution structures at half time and the original shape of the twin slotted-cylinder at final time is also captured. Moreover, the numerical solution preserves the positivity exactly and compares to that reported in the [123].

We consider a variant of this test by changing the initial condition where the non-smooth twin slotted-cylinder is replaced by two symmetrically located quasi-smooth

cosine bells defined as follows [74]:

$$u(t = 0, \lambda, \theta) = \begin{cases} d + ch_1(\lambda, \theta) & \text{if } r_1 < r, \\ d + ch_2(\lambda, \theta) & \text{if } r_2 < r, \\ d & \text{otherwise,} \end{cases} \quad (3.71)$$

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

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

Other parameters are the same as those used for the slotted-cylinder case. Note that the initial condition is quasi-smooth (C^1 smoothness). We want to use this test case to compare the proposed SLDG scheme with the CSLAM scheme in terms of error norms. The mesh is set as $20 \times 20 \times 6$ corresponding to 1.5° resolution at the equator and the time step is set as $\Delta t = T/600$. In Table 3.19, we show the L^1 , L^2 and L^∞ error norms of SLDG P^3 and those of CSLAM reported in [74] with the same resolution and time step. It is observed that the L^1 , L^2 , and L^∞ error norms by SLDG are approximately 2/3-3/4 of those by CSLAM. We conclude that when the CFL number approximately equals 1 and the same resolutions are considered, the errors by SLDG are smaller than those by CSLAM when solving the deformational flow problem.

3.5. APPLICATIONS TO GLOBAL TRANSPORT

Table 3.19: Comparison between the SLDG P^3 scheme and the CSLAM scheme [62] in terms of error norms when solving the deformational flow of two symmetrically located cosine bells. The mesh is set as $20 \times 20 \times 6$ for SLDG corresponding to 1.5° resolution and $60 \times 60 \times 6$ for CSLAM corresponding to the same resolution. The result of CSLAM is from [74]. The numerical solutions are computed after a full rotation with time step $\Delta t = T/600$.

Scheme	L^1	L^2	L^∞
SLDG P^3	0.0393	0.0673	0.1109
CSLAM	0.0533	0.1088	0.1421

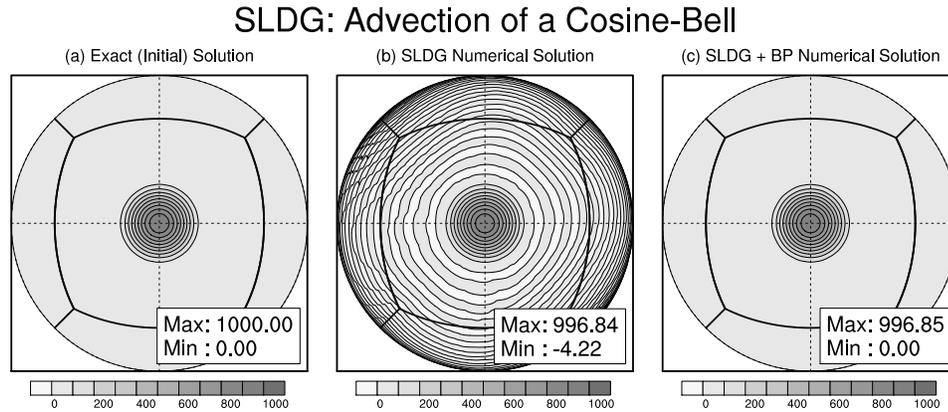


Figure 3.36: Solid body rotation of a cosine bell with $\alpha = \pi/4$. The SLDG P^3 scheme on a cubed-sphere mesh $20 \times 20 \times 6$ (1.5° resolution), $\Delta t = 3600$ sec. (a) Exact (initial) solution; (b) the numerical solution without the BP filter; (c) the numerical solution with the BP filter.

3.5. APPLICATIONS TO GLOBAL TRANSPORT

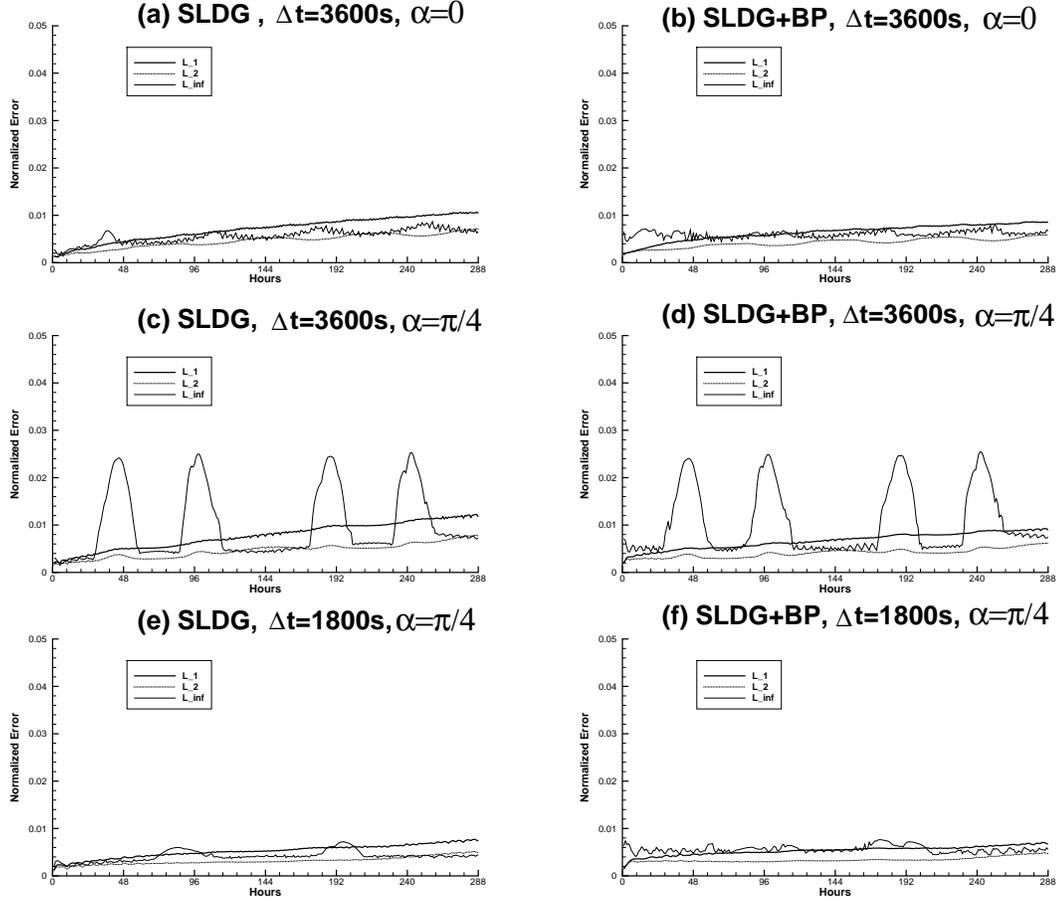


Figure 3.37: The histories of error norms evolution for the solid body rotation. The SLDG P^3 scheme is applied on a cubed-sphere mesh $20 \times 20 \times 6$ (1.5° resolution). The time step is set as $\Delta t = 3600$ sec for panel (a)-(d) and $\Delta t = 1800$ sec for panel (e)-(f). (a) Evolution of error norms for the SLDG scheme without the BP filter when $\alpha = 0$; (b) evolution of error norms for the SLDG scheme with the BP filter when $\alpha = 0$. Comparable result is observed to Panel (a); (c) evolution of error norms for the SLDG scheme without the BP filter when $\alpha = \pi/4$. The L^∞ grows when the cosine bell goes through a corner, where the the splitting error is larger than elsewhere; (d) evolution of error norms for the SLDG scheme with the BP filter when $\alpha = \pi/4$; (e) evolution of error norms for the SLDG scheme without the BP filter when $\alpha = \pi/4$ and $\Delta t = 1800$ sec; the L^∞ does not excessively grows when the cosine bell goes through the corner; (f) evolution of error norms for the SLDG scheme with the BP filter when $\alpha = \pi/4$ and $\Delta t = 1800$.

Advection of a Cosine-Bell: Difference from the Exact Solution

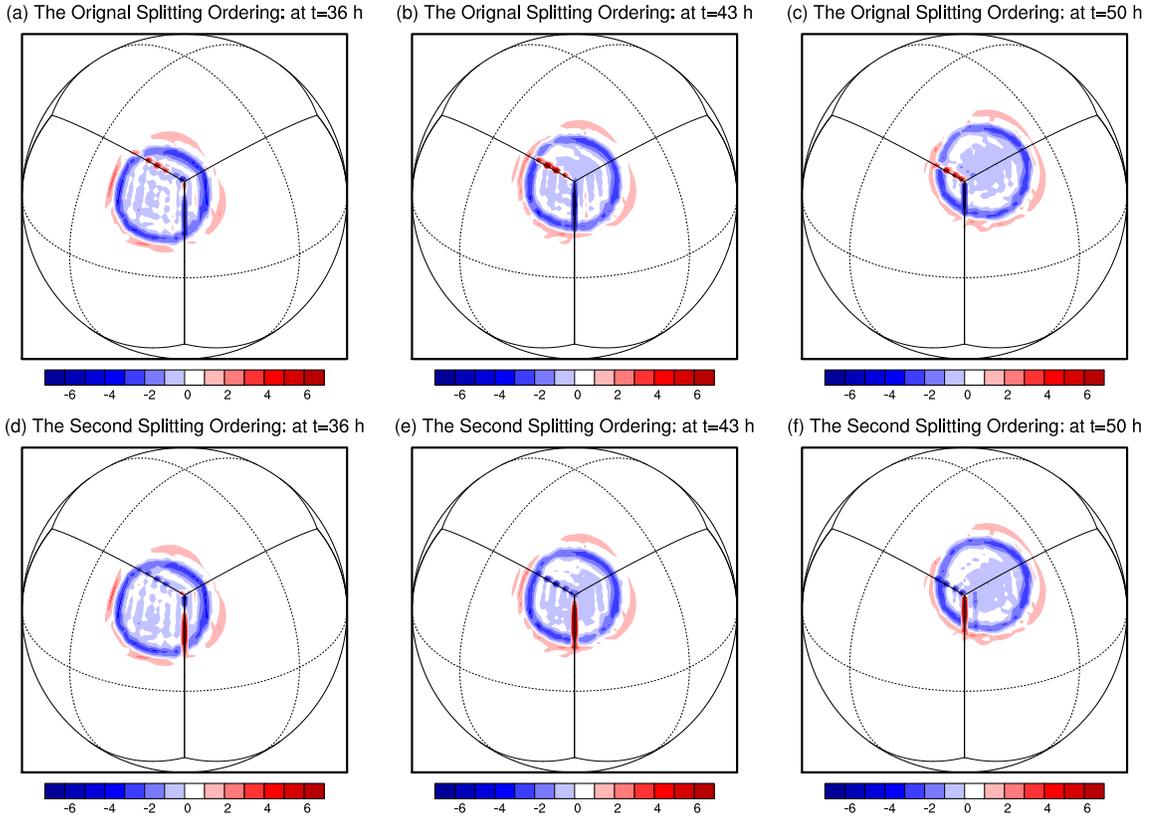


Figure 3.38: The error pattern for the solid body rotation problem when the cosine bell passes one corner. The SLDG P^3 scheme is applied on a cubed-sphere mesh $20 \times 20 \times 6$ (1.5° resolution). The time step is set as $\Delta t = 3600$ sec. Panel (a)-(c) are from the method using the dimensional splitting as described in Subsection 3.5.2; panel (d)-(f) are from the method with another ordering for dimensional splitting as described in Example 3.22. Panel (a) and (d) are for $t = 36$ h; panel (b) and (e) are for $t = 43$ h; panel (c) and (f) are for $t = 50$ h.

SLDG: Moving Vortex Test

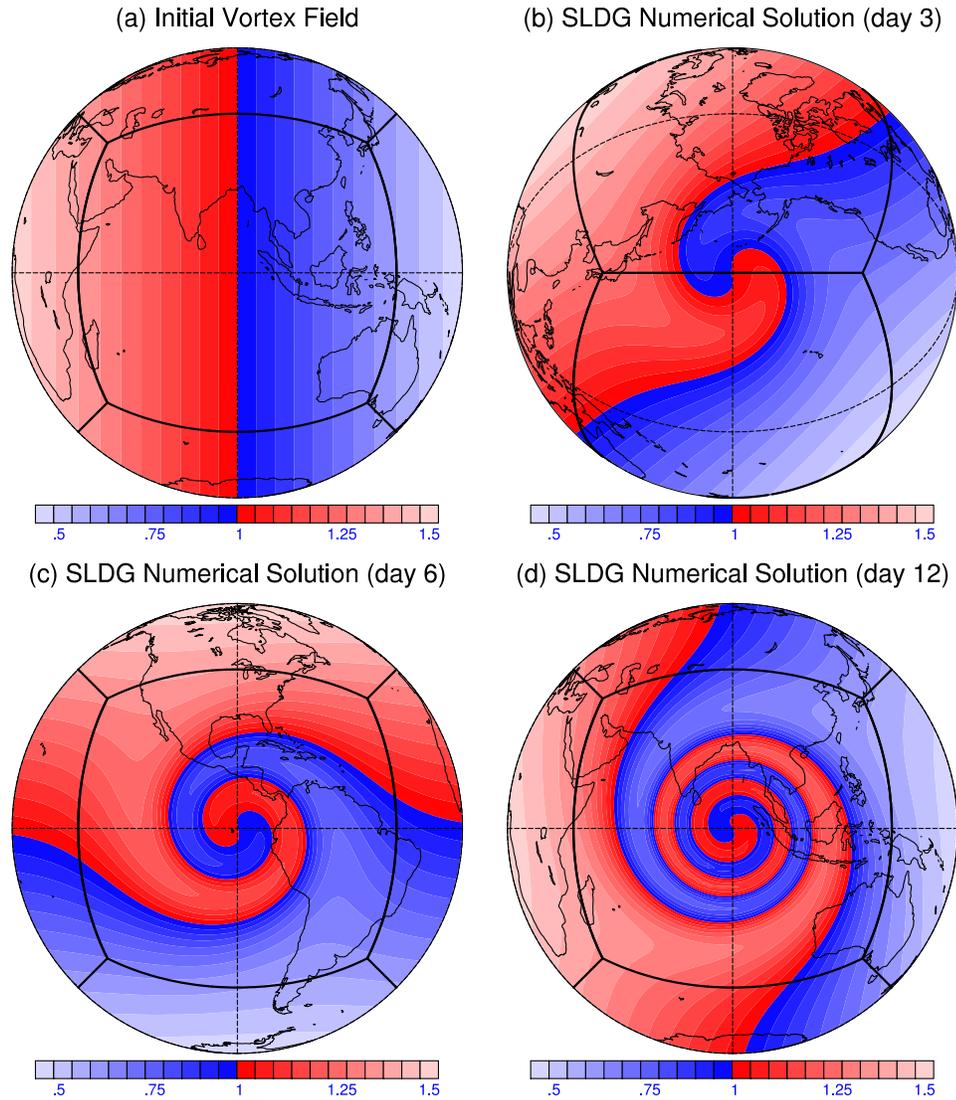


Figure 3.39: An orthographic projection of the solution of moving-vortex test with $\alpha = \pi/4$. The SLDG P^3 scheme is applied on a cubed-sphere mesh $30 \times 30 \times 6$ (1° resolution) and time step is set as $\Delta t = 3600$ sec. (a) Initial vortex field; (b) the numerical solution by SLDG P^3 at day 3; (c) the numerical solution by SLDG P^3 at day 6; (d) the numerical solution by SLDG P^3 at day 12. The numerical solutions are visually identical to the exact solution, therefore it is not shown.

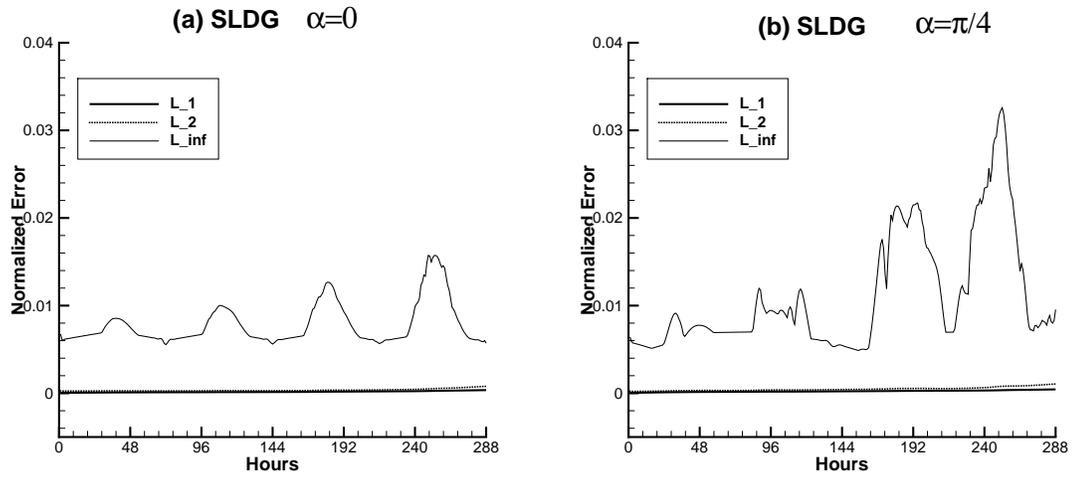


Figure 3.40: The histories of error norms evolution for the moving-vortex test. The SLDG P^3 scheme is applied on a cubed-sphere mesh $30 \times 30 \times 6$ (1° resolution) and time step is set as $\Delta t = 3600$ sec. (a) Evolution of error norms for SLDG P^3 when $\alpha = 0$; (b) evolution of error norms for SLDG P^3 when $\alpha = \pi/4$. The L^∞ error grows when the vortices go through the edges and corners, where the the splitting error is larger than elsewhere. However, the it drops back when the vortex is away from the edges or corners.

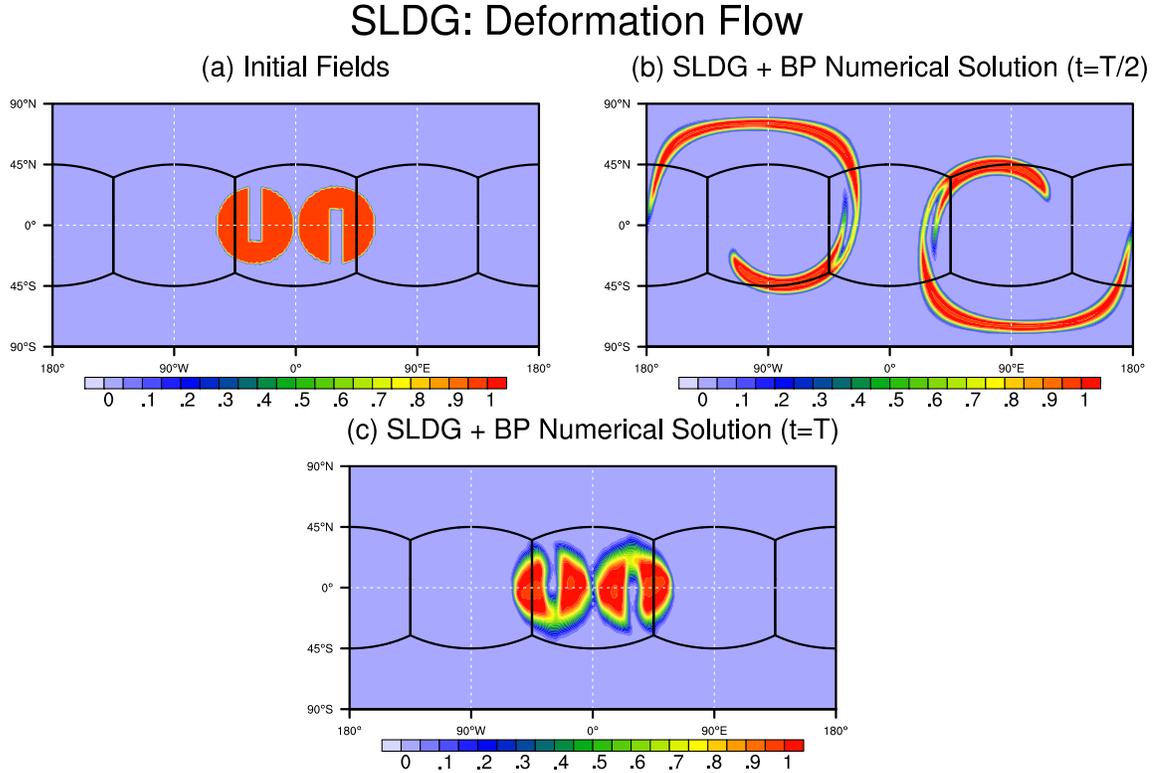


Figure 3.41: Numerical solution for the deformational flow test with twin slotted-cylinder as the initial condition. The SLDG P^3 scheme with BP filter is applied on a cubed-sphere mesh $30 \times 30 \times 6$ (1° resolution) and time step is set as $t = T/800$. (a) Initial condition: twin slotted-cylinder; (b) the numerical solution by SLDG P^3 at half time $t = T/2$. The thin filaments is resolved; (c) the numerical solution by SLDG P^3 at the final time $T = 5$. The slotted cylinders are captured by the SLDG scheme. T is set as 5. Also note that the numerical solution is exactly positivity preserving.

CHAPTER 4

Superconvergence of Discontinuous Galerkin Schemes

In this chapter, we investigate superconvergence properties of DG schemes. First, Fourier analysis for DG and LDG schemes is reviewed. Then, we study superconvergence of DG and LDG schemes via the eigenstructure analysis in the Fourier approach framework. The last part pursues superconvergence analysis for the DG schemes with a high order Lax-Wendroff (LW) time discretization.

4.1 Fourier Analysis for DG and LDG Schemes

In this section, we review Fourier analysis for the DG scheme (2.3) and the LDG scheme (2.10). Fourier analysis is also named von Neumann analysis in the literature, which can provide a sufficient condition of a ‘bad’ scheme, i.e., the amplification factor or the eigenvalues of the amplification matrix of a scheme should have a negative real part for linear stability. Even though Fourier analysis only applies to linear schemes for linear equations with periodic conditions and uniform meshes, it provides a guideline to analyze the underlying schemes in more general settings. Below, we will use this technique to analyze superconvergence properties of DG and LDG schemes in different frameworks.

4.1.1 Fourier Analysis for DG Schemes

First of all, we consider the following linear hyperbolic equation

$$\begin{aligned}u_t + au_x &= 0, \quad x \in [0, 2\pi], \quad t > 0, \\u(0, x) &= \exp(i\omega x), \quad x \in [0, 2\pi],\end{aligned}\tag{4.1}$$

with periodic boundary conditions. Note that we let $f(u) = au$ in (2.1) with a indicating wave propagation speed. The initial condition is a plane wave and ω is the wave number. For convenience, assume that $a > 0$.

After applying the DG scheme (2.3), we obtain a ODEs system (2.6) which can

be rewritten as

$$\frac{d\mathbf{u}_j}{dt} = \frac{a}{\Delta x} (A\mathbf{u}_j + B\mathbf{u}_{j-1}), \quad (4.2)$$

where A and B are $(k+1) \times (k+1)$ constant matrices.

Under the assumption of uniform meshes and periodic boundary conditions, Fourier analysis starts by making the following ansate

$$\mathbf{u}_j(t) = \hat{\mathbf{u}}(t) \exp(i\omega x_j), \quad (4.3)$$

substituting which into the DG scheme (4.2) provides the following ODE system for the coefficient vector $\hat{\mathbf{u}}(t)$,

$$\frac{d}{dt} \hat{\mathbf{u}}(t) = aG\hat{\mathbf{u}}(t), \quad (4.4)$$

where G is the amplification matrix, given by

$$G = \frac{1}{\Delta x} (A + B e^{-i\xi}), \quad \xi = \omega \Delta x. \quad (4.5)$$

If G is diagonalizable, denote the eigenvalues of G as $\lambda_0, \dots, \lambda_k$ and the corresponding eigenvectors as $\tilde{V}_0, \dots, \tilde{V}_k$. Then the general solution of the ODE system (4.4) is

$$\hat{\mathbf{u}}(t) = C_0 e^{a\lambda_0 t} \tilde{V}_0 + \dots + C_k e^{a\lambda_k t} \tilde{V}_k, \quad (4.6)$$

where C_0, \dots, C_k can be determined by the initial condition. Let $V_l = C_l \tilde{V}_l$, $l = 0, \dots, k$, then

$$\hat{\mathbf{u}}(t) = e^{a\lambda_0 t} V_0 + \dots + e^{a\lambda_k t} V_k, \quad (4.7)$$

which is an explicit representation of a DG solution for future time $t > 0$.

4.1.2 Fourier Analysis for LDG schemes

Below we review Fourier analysis of LDG schemes (2.10). We consider the following linear heat equation

$$\begin{aligned} u_t &= u_{xx}, \quad x \in [0, 2\pi], \quad t > 0, \\ u(0, x) &= \exp(i\omega x), \quad x \in [0, 2\pi], \end{aligned} \quad (4.8)$$

with periodic boundary conditions.

Similar to the DG scheme, the LDG scheme for equation (4.8) with uniform meshes can be rewritten in the following form:

$$\frac{d\mathbf{u}_j}{dt} = \frac{1}{\Delta x^2} (A_1 \mathbf{u}_{j-1} + B \mathbf{u}_j + A_2 \mathbf{u}_{j+1}), \quad (4.9)$$

where \mathbf{u}_j is a vector of degrees of freedom on cell I_j defined in (2.5). A_1 , B and A_2 are $(k+1) \times (k+1)$ constant matrices. Assume that the LDG solution is in the form

of equation (4.3). Substituting (4.3) into (4.9) gives

$$\frac{d}{dt}\hat{\mathbf{u}}(t) = G\hat{\mathbf{u}}(t), \quad (4.10)$$

with the amplification matrix G given by

$$G = \frac{1}{\Delta x}(A_1 e^{-i\xi} + B + A_2 e^{i\xi}), \quad \xi = \omega \Delta x. \quad (4.11)$$

As in equation (4.7) for the DG solution, the explicit form of the LDG solution can also be expressed based on eigenvalues and eigenvectors of G in equation (4.11) if G is diagonalizable.

4.2 Eigen-Structures of G : Error Estimate

4.2.1 DG Schemes

One-dimensional Scalar and System of Linear Hyperbolic Equations

Depending on different choices of basis functions in DG implementation, the amplification matrix G could be different. The eigenvalues of G however will stay the same, since the DG method is independent of the choice of basis functions. However, the eigenvectors will be basis-dependent. In order to reveal the superconvergence at shifted Radau points, we analyze G matrix based on the basis functions that are the

Lagrangian polynomials

$$\phi_j^l(x) = \prod_{i \neq l} \frac{x - x_j^i}{x_j^l - x_j^i}, \quad (4.12)$$

where

$$x_j^l = x_j + \frac{\zeta_{k,l}}{2} \Delta x, \quad l = 0, \dots, k$$

are the $k + 1$ shifted Radau points. $\{\zeta_{k,l}\}$ are the roots of the Radau polynomial $P_{k+1}(\zeta) - P_k(\zeta)$, where $P_k(\zeta)$ is the Legendre polynomial of degree k normalized such that

$$\int_{-1}^1 P_i(x) P_j(x) = \frac{2}{2i+1} \delta_{ij},$$

where δ_{ij} is the Kroneker delta. Such choice of basis functions will help to reveal the superconvergence properties at Radau and downwind points [1, 15].

Proposition 4.1. *Consider DG methods with polynomial space P^k ($k = 1, 2,$ and 3) for linear hyperbolic problem (4.1) with uniform meshes. Consider Fourier analysis of the DG method using Lagrangian polynomials (4.12) based on shifted Radau points as basis functions. The amplification matrix G is diagonalizable with $k + 1$ distinct eigenvalues. One of these eigenvalues denoted as λ_0 is the physically relevant one; it approximates the analytical value $-i\omega$ with dissipation error on the order of $2k + 1$ and dispersion error on the order of $2k + 2$. The rest of eigenvalues $\lambda_1, \dots, \lambda_k$ have a negative real part with the magnitude on the order of $\frac{1}{\Delta x}$.*

Proof. We perform symbolic computations via Mathematica to analyze eigenvalues of G . Here is a summary of our results:

- P^1 case

$$\lambda_0 = -i\omega - \frac{\omega^4}{72}\Delta x^3 - \frac{i\omega^5}{270}\Delta x^4 + \mathcal{O}(\Delta x^5)$$

$$\lambda_1 = -\frac{6}{\Delta x} + \mathcal{O}(1)$$

- P^2 case

$$\lambda_0 = -i\omega - \frac{\omega^6}{7200}\Delta x^5 - \frac{i\omega^7}{42000}\Delta x^6 + \mathcal{O}(\Delta x^7)$$

$$\lambda_{1,2} = \frac{-3 \pm \sqrt{51}i}{\Delta x} + \mathcal{O}(1)$$

- P^3 case

$$\lambda_0 = -i\omega - 7.08 \times 10^{-7}\omega^8\Delta x^7 - 9.00 \times 10^{-8}i\omega^9\Delta x^8 + \mathcal{O}(\Delta x^9)$$

$$\lambda_{1,2} = \frac{-0.42 \pm 6.61i}{\Delta x} + \mathcal{O}(1), \quad \lambda_3 = -\frac{19.15}{\Delta x} + \mathcal{O}(1)$$

It can be checked from above that for $k = 1, 2,$ and 3

$$\mathcal{R}(-i\omega - \lambda_0) = \mathcal{O}(\Delta x^{2k+1}), \quad \mathcal{I}(-i\omega - \lambda_1) = \mathcal{O}(\Delta x^{2k+2}).$$

$$\mathcal{R}(\lambda_l) < 0, \quad |\mathcal{R}(\lambda_l)| = \mathcal{O}\left(\frac{1}{\Delta x}\right), \quad l = 1, \dots, k.$$

■

Remark 4.2. *The fact that the non-physically relevant eigenvalues have large negative real part on the order of $\frac{1}{\Delta x}$ indicates that the corresponding eigenvectors will*

be damped exponentially fast with respect to Δx over time.

Proposition 4.3. *With the same assumption as Proposition 4.1, the eigenvector V_0 corresponding to the physically relevant eigenvalue λ_0 approximates $\hat{\mathbf{u}}(0)$ in equation (4.7) with order $k+2$ at Radau points and with order $2k+1$ at downwind points. The non-physically relevant eigenvectors V_l , $l = 1, \dots, k$ are of order $k+2$ at Radau points.*

Proof. We perform symbolic computations via Mathematica. Below is a summary of our results:

- P^1 case

$$V_0 - \hat{\mathbf{u}}(0) = \begin{pmatrix} \frac{i\omega^3}{162}\Delta x^3 + \mathcal{O}(\Delta x^4) \\ -\frac{i\omega^3}{54}\Delta x^3 + \mathcal{O}(\Delta x^4) \end{pmatrix}$$

$$V_1 = \begin{pmatrix} -\frac{i\omega^3}{162}\Delta x^3 + \mathcal{O}(\Delta x^4) \\ \frac{i\omega^3}{54}\Delta x^3 + \mathcal{O}(\Delta x^4) \end{pmatrix}$$

- P^2 case

$$\begin{aligned}
 V_0 - \hat{\mathbf{u}}(0) &= \begin{pmatrix} \frac{(3 + 8\sqrt{6})\omega^4}{20000} \Delta x^4 + \mathcal{O}(\Delta x^5) \\ \frac{(3 - 8\sqrt{6})\omega^4}{20000} \Delta x^4 + \mathcal{O}(\Delta x^5) \\ -\frac{i\omega^5}{3000} \Delta x^5 + \mathcal{O}(\Delta x^6) \end{pmatrix} \\
 V_1 &= \begin{pmatrix} -\frac{(153 + 408\sqrt{6} + i18\sqrt{34} - i29\sqrt{51})\omega^4}{2040000} \Delta x^4 + \mathcal{O}(\Delta x^5) \\ -\frac{(153 - 408\sqrt{6} - i18\sqrt{34} - i29\sqrt{51})\omega^4}{2040000} \Delta x^4 + \mathcal{O}(\Delta x^5) \\ -\frac{i\omega^4}{160\sqrt{51}} \Delta x^4 + \mathcal{O}(\Delta x^5) \end{pmatrix} \\
 V_2 &= \begin{pmatrix} -\frac{(153 + 408\sqrt{6} - i18\sqrt{34} + i29\sqrt{51})\omega^4}{2040000} \Delta x^4 + \mathcal{O}(\Delta x^5) \\ -\frac{(153 - 408\sqrt{6} + i18\sqrt{34} + i29\sqrt{51})\omega^4}{2040000} \Delta x^4 + \mathcal{O}(\Delta x^5) \\ \frac{i\omega^4}{160\sqrt{51}} \Delta x^4 + \mathcal{O}(\Delta x^5) \end{pmatrix}
 \end{aligned}$$

- P^3 case

$$V_0 - \hat{\mathbf{u}}(0) = \begin{pmatrix} -4.58 \times 10^{-5} i\omega^5 \Delta x^5 + \mathcal{O}(\Delta x^6) \\ 4.81 \times 10^{-5} i\omega^5 \Delta x^5 + \mathcal{O}(\Delta x^6) \\ -2.61 \times 10^{-5} i\omega^5 \Delta x^5 + \mathcal{O}(\Delta x^6) \\ -2.43 \times 10^{-6} i\omega^7 \Delta x^7 + \mathcal{O}(\Delta x^8) \end{pmatrix}$$

$$V_1 = \begin{pmatrix} (2.13 \times 10^{-5} + i1.19 \times 10^{-5})\omega^5 \Delta x^5 + \mathcal{O}(\Delta x^6) \\ (1.55 \times 10^{-6} - i1.86 \times 10^{-5})\omega^5 \Delta x^5 + \mathcal{O}(\Delta x^6) \\ (-1.73 \times 10^{-5} + i9.61 \times 10^{-6})\omega^5 \Delta x^5 + \mathcal{O}(\Delta x^6) \\ (6.53 \times 10^{-6} + i2.31 \times 10^{-5})\omega^5 \Delta x^5 + \mathcal{O}(\Delta x^6) \end{pmatrix}$$

$$V_2 = \begin{pmatrix} (-2.13 \times 10^{-5} + i1.19 \times 10^{-5})\omega^5 \Delta x^5 + \mathcal{O}(\Delta x^6) \\ (-1.55 \times 10^{-6} - i1.86 \times 10^{-5})\omega^5 \Delta x^5 + \mathcal{O}(\Delta x^6) \\ (1.73 \times 10^{-5} + i9.61 \times 10^{-6})\omega^5 \Delta x^5 + \mathcal{O}(\Delta x^6) \\ (-6.53 \times 10^{-6} + i2.31 \times 10^{-5})\omega^5 \Delta x^5 + \mathcal{O}(\Delta x^6) \end{pmatrix}$$

$$V_3 = \begin{pmatrix} 2.20 \times 10^{-5}i\omega^5 \Delta x^5 + \mathcal{O}(\Delta x^6) \\ -1.09 \times 10^{-5}i\omega^5 \Delta x^5 + \mathcal{O}(\Delta x^6) \\ 6.85 \times 10^{-6}i\omega^5 \Delta x^5 + \mathcal{O}(\Delta x^6) \\ -4.62 \times 10^{-5}i\omega^5 \Delta x^5 + \mathcal{O}(\Delta x^6) \end{pmatrix}$$

Clearly, for $k = 1, 2$, and 3 , V_0 approximates $\hat{\mathbf{u}}(0)$ with order $2k + 1$ at downwind points and with order $k + 2$ at other Radau points. V_l , $l = 1, \dots, k$ are of order $(k + 2)$ at Radau points. ■

Remark 4.4. *We remark that the choice of Lagrangian basis functions based on shifted Radau points is crucial in estimating superconvergence properties at Radau points. If we choose other basis functions, e.g. Lagrangian basis functions based on uniformly distributed points as in [119], then the eigenvector V_0 approximates $\hat{\mathbf{u}}(0)$*

4.2. EIGEN-STRUCTURES OF G : ERROR ESTIMATE

with order $k + 1$ at all points, and V_l , $l \geq 1$, is of order $k + 1$ at all points. Details of symbolic computation are omitted for brevity.

Proposition 4.5. *With the same assumption as Proposition 4.1, let $\mathbf{u}(T) = \hat{\mathbf{u}}(0) \exp(i\omega x_j - \omega^2 T)$ and $\mathbf{u}_h(T) = \hat{\mathbf{u}}(T) \exp(i\omega x_j)$ be point values of exact solution and numerical solution at shifted Radau points on a cell I_j . Let $\mathbf{e} = \mathbf{u} - \mathbf{u}_h$. Then for $T > 0$,*

$$\|\mathbf{e}(T)\| \leq C_1 a T \Delta x^{2k+1} + C_2 \Delta x^{k+2} + C_3 \exp(-C \frac{aT}{\Delta x}) \Delta x^{k+2}, \quad (4.13)$$

where C , C_1 , C_2 , and C_3 are positive constants independent of Δx and $\|\cdot\|$ can be any norm for vectors.

Proof. Note that in (4.7), $\hat{\mathbf{u}}(0) = \sum_{l=0}^k V_l$. By Equation (4.7), Proposition 4.1 and 4.3, we have

$$\begin{aligned} \|\mathbf{e}(T)\| &= \|\mathbf{u}(T) - \mathbf{u}_h(T)\| \\ &= \|\exp(-i\omega a T) \hat{\mathbf{u}}(0) - \sum_{l=0}^k \exp(\lambda_l a T) V_l\| \\ &\stackrel{(4.7)}{=} \|\exp(-i\omega a T) \sum_{l=0}^k V_l - \sum_{l=0}^k \exp(\lambda_l a T) V_l\| \\ &\leq \|(\exp(-i\omega a T) - \exp(\lambda_0 a T)) V_0\| + |\exp(-i\omega a T)| \|\sum_{l=1}^k V_l\| \\ &\quad + \sum_{l=1}^k \|\exp(\lambda_l a T) V_l\| \end{aligned}$$

$$\begin{aligned} &\leq |\exp(-i\omega aT) - \exp(\lambda_0 aT)| \|V_0\| + \|\hat{\mathbf{u}}(0) - V_0\| + \sum_{l=1}^k |\exp(\lambda_l aT)| \|V_l\| \\ &\leq C_1 aT \Delta x^{2k+1} + C_2 \Delta x^{k+2} + C_3 \exp(-C \frac{aT}{\Delta x}) \Delta x^{k+2}, \end{aligned}$$

where C , C_1 , C_2 , and C_3 are positive constants independent of Δx . Notice that $\|V_1\|$ is of order 1 by Proposition 4.3. ■

Remark 4.6. *From Proposition 4.5, it can be seen that under the assumption of uniform mesh, the error of the DG solution for a linear hyperbolic problem can be decomposed as three parts:*

1. *Dissipation and dispersion errors of the physically relevant eigenvalue. This part of error will grow linearly in time and is of order $2k + 1$.*
2. *Projection error $\|\mathbf{u}^* - \mathbf{u}\|$. That is, there exists a special projection of the solution,*

$$\mathbf{u}^*(T) = P_h^* \mathbf{u}(T) = \exp(i\omega(x_j - aT)) V_1$$

on cell I_j , such that the numerical solution is much closer to the special projection of exact solution ($\|\mathbf{u}_h - \mathbf{u}^\| = \mathcal{O}(\Delta x^{2k+1})$), than the exact solution itself.*

The projection error

$$\|\mathbf{u}^* - \mathbf{u}\| = \mathcal{O}(\Delta x^{k+2})$$

will not grow in magnitude in time. By Proposition 4.3, such special projection approximates the exact solution at Radau points with order $k+2$ with the exception of Radau point at downwind end, which is of order $2k + 1$. Unfortunately, the analytical form of such special projection is only known symbolically and is

4.2. EIGEN-STRUCTURES OF G: ERROR ESTIMATE

subject to further investigation.

3. *Dissipation of non-physically relevant eigenvectors.* This part of error will decay exponentially fast over time with respect to Δx if $a \geq a_0 > 0$.

Remark 4.7. When $a > a_0 \geq 0$, the error $e(T)$ in Proposition 4.5 is of order $k + 2$ at Radau points and is of order $2k + 1$ at downwind points.

Remark 4.8. Based on the error estimate (4.13), one can conclude

(a) when $T = o(\frac{1}{\Delta x^{k-1}})$, $\mathcal{O}(\Delta x^{k+2})$ is the dominant term in (4.13); this term will not grow with time;

(b) when $T = \mathcal{O}(\frac{1}{\Delta x^{k-1}})$ (very long time integration), $C_1 a T \Delta x^{2k+1}$ is the dominant term in (4.13); this term grows linearly with time and is of order $2k + 1$.

Since it is hard to check numerically the long time behavior of the error of DG solutions, we propose to use the following Corollary as a way to numerically assess our theoretical results discussed in this section.

Corollary 4.9. Consider DG methods with polynomial space P^k ($k = 1, 2$, and 3) for linear problem (4.1) with uniform mesh. Let $n > 1$ be an integer, then

$$\|\mathbf{u}_h(t + \frac{2n\pi}{a}) - \mathbf{u}_h(t)\| \leq C_1 n (\Delta x^{2k+1}) + C_2 \exp(-\frac{C a t}{\Delta x}) \Delta x^{k+2}, \quad (4.14)$$

where C , C_1 , and C_2 are positive constants independent of Δx .

Proof.

$$\begin{aligned}
 \|\mathbf{u}_h(t + \frac{2n\pi}{a}) - \mathbf{u}_h(t)\| &= \left\| \sum_{l=0}^k (\exp(\lambda_l(at + 2n\pi)) - \exp(\lambda_l at)) V_l \right\| \\
 &\leq |\exp(\lambda_0(at + 2n\pi)) - \exp(\lambda_0 at)| \|V_0\| \\
 &\quad + \sum_{l=1}^k |\exp(\lambda_l(at + 2n\pi)) - \exp(\lambda_l at)| \|V_l\| \\
 &\leq |\exp(\lambda_0 2n\pi) - \exp(i\omega 2n\pi)| |\exp(\lambda_0 at)| \|V_0\| \\
 &\quad + C_2 \exp(-\frac{Cat}{\Delta x}) \Delta x^{k+2} \\
 &\leq C_1 n |\lambda_1 - i\omega| + C_2 \exp(-\frac{Cat}{\Delta x}) \Delta x^{k+2} \\
 &\leq C_1 n \Delta x^{2k+1} + C_2 \exp(-\frac{Cat}{\Delta x}) \Delta x^{k+2}.
 \end{aligned}$$

■

Remark 4.10. Assume $a = 1$, let $T = 2n\pi + t$ with $t = \mathcal{O}(1)$, then the first term on the r.h.s. of (4.14) is the dominant term.

$$\begin{aligned}
 \|e(T)\| &= \|\mathbf{u}(T) - \mathbf{u}_h(T)\| \\
 &= \|\mathbf{u}(t) - \mathbf{u}_h(2n\pi + t)\| \\
 &\leq \|\mathbf{u}_h(2n\pi + t) - \mathbf{u}_h(t)\| + \|\mathbf{u}(t) - \mathbf{u}_h(t)\|.
 \end{aligned}$$

Since $\|\mathbf{u}_h(2n\pi + t) - \mathbf{u}_h(t)\|$ is order of $2k + 1$ and grows linearly with n , we can conclude that the error $e(T)$ will not grow linearly in time until $T = \mathcal{O}(\frac{1}{\Delta x^{k-1}})$.

Remark 4.11. When a is of order 1, the dominant error on the r.h.s. of (4.14) is of order $2k + 1$. This observation is numerically verified in many examples below. When a is very small, the dominant error is of order $k + 2$, rather than $2k + 1$. This

observation is important in explaining the numerical performance of DG schemes for a nonlinear Burgers' equation in Example 4.32 and for a rotational problem in Example 4.36.

Remark 4.12. Proposition 4.5 and Corollary 4.9 can be extended to a linear hyperbolic system $U_t + AU_x = 0$, where $A_{n \times n}$ is a constant diagonalizable matrix with real eigenvalues. This is due to the fact that the hyperbolic system can be decoupled into n scalar equations.

DG Schemes for 2-D Problems: Q^k

Now, we analyze the DG method for a 2-D linear advection equation

$$u_t + au_x + bu_y = 0, \quad (x, y) \in [0, 2\pi]^2, \quad (4.15)$$

via Fourier analysis. We assume that $a, b > 0$ are constant. Consider a uniform partition of the computational domain as $[0, 2\pi]^2 = \cup_{i,j} I_{ij} = \cup_{i,j} [x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}] \times [y_{j-\frac{1}{2}}, y_{j+\frac{1}{2}}]$. The basis functions are chosen to be 2-D functions Q^k which are tensor products of 1-D ones on each cell I_{ij} . Define the approximation space

$$V_h^k = \{v : v|_{I_{ij}} \in Q^k(I_{ij}); 1 \leq i \leq N_x, 1 \leq j \leq N_y\}. \quad (4.16)$$

The semi-discrete DG method using the upwind flux for solving (4.15) is defined as

follows: find $u_h \in V_h^k$, such that

$$\begin{aligned}
 & \int_{I_{ij}} (u_h)_t v \, dx dy \\
 = & a \int_{I_{ij}} u_h v_x \, dx dy + b \int_{I_{ij}} u_h v_y \, dx dy \\
 & - a \left(\int_{y_{j-\frac{1}{2}}}^{y_{j+\frac{1}{2}}} (u_h)_{i+\frac{1}{2},j}^- (y) v(x_{i+\frac{1}{2}}^-, y) \, dy - \int_{y_{j-\frac{1}{2}}}^{y_{j+\frac{1}{2}}} (u_h)_{i-\frac{1}{2},j}^- (y) v(x_{i-\frac{1}{2}}^+, y) \, dy \right) \\
 & - b \left(\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} (u_h)_{i,j+\frac{1}{2}}^- (x) v(x, y_{j+\frac{1}{2}}^-) \, dx - \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} (u_h)_{i,j-\frac{1}{2}}^- (x) v(x, y_{j-\frac{1}{2}}^+) \, dx \right)
 \end{aligned} \tag{4.17}$$

for all $v \in V_h^k$, and $1 \leq i \leq N_x$, $1 \leq j \leq N_y$. As in the 1-D case, to reveal the superconvergence properties at Radau nodes, we use the following basis functions

$$B_{m,n} = L_m(x)L_n(y), \quad m, n = 0, \dots, k,$$

where $L_m(x)$ and $L_n(y)$ are shifted Radau Lagrangian basis functions (4.12) in x - and y - directions respectively. Please see Figure 4.1 for distribution of shifted Radau points in a cell.

In Fourier analysis, the DG solution is assumed to be of the form

$$\mathbf{u}(t) = \hat{\mathbf{u}}(t) \exp(i\omega_x x_i + i\omega_y y_j), \tag{4.18}$$

where $\hat{\mathbf{u}}(t)$ is the coefficient vector of $(k+1)^2$ elements. We substitute equation (4.18) into the DG scheme (4.17) to obtain an ODE system for the coefficient vector

$$\frac{d}{dt} \hat{\mathbf{u}}(t) = G \hat{\mathbf{u}}(t).$$

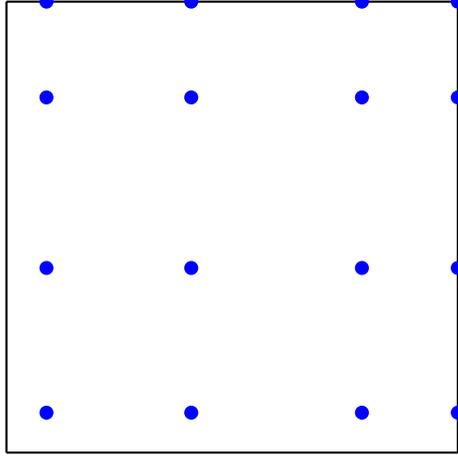


Figure 4.1: Radau points in two-dimensional case.

Let G^{x,ω_x} and G^{y,ω_y} be the 1-D amplification matrix in x - and y -direction, respectively. Then the 2-D amplification matrix G can be written as

$$G = aI \otimes G^{x,\omega_x} + bG^{y,\omega_y} \otimes I, \quad (4.19)$$

where \otimes is the tensor product or Kronecker product of two matrices.

Proposition 4.13. *The 2-D amplification matrix G in equation (4.19) has $(k+1)^2$ eigenvalues,*

$$a\lambda^{x,(p)} + b\lambda^{y,(q)}, \quad p, q = 0, \dots, k,$$

with the corresponding eigenvectors

$$V^{y,(q)} \otimes V^{x,(p)},$$

where $\lambda^{x,(p)}$ and $V^{x,(p)}$, $p = 0, \dots, k$, $\lambda^{y,(q)}$ and $V^{y,(q)}$, $q = 0, \dots, k$ are eigenvalues

4.2. EIGEN-STRUCTURES OF G : ERROR ESTIMATE

and eigenvectors for G^{x,ω_x} and G^{y,ω_y} , respectively.

Proof. Since $(A \otimes B)(\mathbf{x} \otimes \mathbf{y}) = A\mathbf{x} \otimes B\mathbf{y}$, we have

$$\begin{aligned}
 G(V^{y,(q)} \otimes V^{x,(p)}) &= (aI \otimes G^{x,\omega_x} + bG^{y,\omega_y} \otimes I)(V^{y,(q)} \otimes V^{x,(p)}) \\
 &= aV^{y,(q)} \otimes (G^{x,\omega_x} V^{x,(p)}) + bG^{y,\omega_y} V^{y,(q)} \otimes V^{x,(p)} \\
 &= a\lambda^{x,(p)}(V^{y,(q)} \otimes V^{x,(p)}) + b\lambda^{y,(q)}(V^{y,(q)} \otimes V^{x,(p)}) \\
 &= (a\lambda^{x,(p)} + b\lambda^{y,(q)})(V^{y,(q)} \otimes V^{x,(p)}).
 \end{aligned}$$

■

Similar to the 1-D case, based on our understanding on the eigen-structures of 2-D amplification matrices, we have the following error estimate for the DG method with Q^k basis functions for a 2-D linear advection problem (4.15). The proof is similar to the 1-D case, and thus is omitted.

Proposition 4.14. *Consider DG methods with polynomial space Q^k ($k = 1, 2,$ and 3) for the 2-D linear hyperbolic problem (4.15) with uniform mesh sizes Δx and Δy in x - and y -directions respectively. Let $\mathbf{u}(T) = \hat{\mathbf{u}}(0)\exp(i\omega_x(x_i - aT) + i\omega_y(y_j - bT))$ and $\mathbf{u}_h(T) = \hat{\mathbf{u}}(T)\exp(i\omega_x x_j + i\omega_y y_j)$ be point values of exact and numerical solution at shifted Radau points in a cell I_{ij} . Let $\mathbf{e} = \mathbf{u} - \mathbf{u}_h$, then for $T > 0$,*

$$\begin{aligned}
 \|\mathbf{e}(T)\| &\leq C_1 T(a\Delta x^{2k+1} + b\Delta y^{2k+1}) + C_2(\Delta x^{k+2} + \Delta y^{k+2}) \\
 &\quad + C_3 \exp(-CT(\frac{a}{\Delta x} + \frac{b}{\Delta y}))(\Delta x^{k+2} + \Delta y^{k+2}),
 \end{aligned} \tag{4.20}$$

where $C, C_1, C_2,$ and C_3 are positive constants independent of Δx and Δy .

Remark 4.15. Consider a DG method for the 2-D linear hyperbolic problem (4.15) using polynomial spaces

$$P^k(I_{ij}) = \left\{ \sum_{i+j \leq k} c_{ij} x^i y^j \right\} \quad (4.21)$$

as polynomials up to degree k on each cell I_{ij} . Unlike the Q^k case, the eigen-structure for a 2-D amplification matrix cannot be analyzed via our understanding on a 1-D case, since the number of basis functions increase quadratically with k and it is difficult to obtain the roots of an algebraic equation of degree higher than 4.

4.2.2 LDG Schemes

In this subsection, we discuss the eigen-structure of the amplification matrix from an LDG scheme for linear parabolic problem (4.8). As in a DG scheme, we formulate the amplification matrix with Lagrangian basis functions (4.12) based on the $k+1$ shifted Radau points. Such choice of basis functions will help to reveal superconvergence properties at Radau points [2, 16]. The direction of the Radau points is determined by the choice of the numerical flux. In the following analysis and simulations, we choose $\hat{u} = u^-$ and $\hat{q} = q^+$. In this case, the right-shifted Radau points are used and the corresponding downwind points are $x_{j+\frac{1}{2}}^-$. Note that the amplification matrix of the LDG scheme for equation (4.8) can be derived from the amplification matrix of DG scheme for equation (4.1) with $a = 1$ directly. Specifically, let G_{DG} and G_{LDG} denote the amplification matrix of DG and LDG respectively. Then

$$G_{LDG} = -W\bar{G}_{DG}WG_{DG}. \quad (4.22)$$

4.2. EIGEN-STRUCTURES OF G : ERROR ESTIMATE

Here \bar{G}_{DG} is the conjugate of G_{DG} , i.e.,

$$G_{DG} = \frac{1}{\Delta x}(A + Be^{-i\xi}), \quad \bar{G}_{DG} = \frac{1}{\Delta x}(A + Be^{i\xi}),$$

with the notations introduced in (4.5). W is the change of basis matrix, which maps function values at right-shifted Radau points to left-shifted ones. By the symmetry of right-shifted and left-shifted Radau points distribution, W features the property $W^{-1} = W$. For example, for P^1 case, the right-shifted Radau points are ordered as $(x_{j-\frac{1}{6}}, x_{j+\frac{1}{2}})$, the left-shifted ones are ordered as $(x_{j+\frac{1}{6}}, x_{j-\frac{1}{2}})$, and

$$W = \begin{pmatrix} \frac{1}{2} & \frac{1}{2} \\ \frac{3}{2} & -\frac{1}{2} \end{pmatrix}.$$

Notice the symmetry of these two set of points with respect to x_j . Due to such symmetry, it can be checked that $W^{-1} = W$ as claimed above.

Proposition 4.16. *Consider LDG methods with polynomial space P^k ($k = 1, 2$, and 3) for linear parabolic problem (4.8) with uniform mesh. Consider Fourier analysis of the LDG method using Lagrangian basis functions (4.12) based on shifted Radau points. The amplification matrix G is diagonalizable with $(k+1)$ distinct eigenvalues. One of these eigenvalues denoted as λ_0 is the physically relevant one, approximating $-\omega^2$ with dissipation error on the order of $2k+2$. The rest of eigenvalues $\lambda_1, \dots, \lambda_k$ have negative real part with the magnitude on the order of $\frac{1}{\Delta x^2}$.*

Proof. We perform symbolic computations on Mathematica. Below is a summary

4.2. EIGEN-STRUCTURES OF G : ERROR ESTIMATE

of eigenvalues of G .

- P^1 case

$$\lambda_0 = -\omega^2 + \frac{\omega^6}{540}\Delta x^4 + \frac{11\omega^8}{108864}\Delta x^6 + \mathcal{O}(\Delta x^8)$$

$$\lambda_1 = -\frac{36}{\Delta x^2} + \mathcal{O}(1)$$

- P^2 case

$$\lambda_0 = -\omega^2 + \frac{\omega^8}{128000}\Delta x^6 + \frac{\omega^{10}}{8100000}\Delta x^8 + \mathcal{O}(\Delta x^{10})$$

$$\lambda_{1,2} = \frac{-78 \pm 6\sqrt{69}}{\Delta x^2} + \mathcal{O}(1)$$

- P^3 case

$$\lambda_0 = -\omega^2 - 2.25 \times 10^{-8}\omega^{10}\Delta x^8 - 7.51 \times 10^{-11}\omega^{12}\Delta x^{10} + \mathcal{O}(\Delta x^{12})$$

$$\lambda_1 = -\frac{438.91}{\Delta x^2} + \mathcal{O}(1)$$

$$\lambda_2 = -\frac{46.58}{\Delta x^2} + \mathcal{O}(1)$$

$$\lambda_3 = -\frac{34.51}{\Delta x^2} + \mathcal{O}(1)$$

It can be checked from above that for $k = 1, 2,$ and 3

$$\mathcal{R}(-\omega^2 - \lambda_0) = \mathcal{O}(\Delta x^{2k+2}).$$

$$\mathcal{R}(\lambda_l) < 0, \quad |\mathcal{R}(\lambda_l)| = \mathcal{O}\left(\frac{1}{\Delta x^2}\right), \quad l = 1, \dots, k.$$

■

Remark 4.17. *In our symbolic computation, we find that the eigenvalues of the amplification matrix G for LDG methods are real for $k = 1, 2$, and 3 . However, this fact is difficult to prove based on equation (4.22).*

Remark 4.18. *The fact that the other eigenvalues have large negative real part on the order of $\frac{1}{\Delta x^2}$ indicates that the corresponding errors will be damped out exponentially fast with respect to Δx over time.*

Proposition 4.19. *With the same assumption as Proposition 4.16, the eigenvector V_0 corresponding to the physically relevant eigenvalue λ_0 approximates $\hat{\mathbf{u}}(0)$ with order $k + 2$ at Radau points and with order $2k + 1$ at downwind points. The non-physically relevant eigenvectors V_l , $l = 1, \dots, k$ are of order $k + 2$ at Radau points.*

Proof. We perform symbolic computations on Mathematica. Below is a summary of our results.

- P^1 case

$$V_0 - \hat{\mathbf{u}}(0) = \begin{pmatrix} -\frac{i\omega^3}{324}\Delta x^3 + \mathcal{O}(\Delta x^4) \\ \frac{i\omega^3}{108}\Delta x^3 + \mathcal{O}(\Delta x^4) \end{pmatrix}$$

$$V_1 = \begin{pmatrix} \frac{i\omega^3}{324}\Delta x^3 + \mathcal{O}(\Delta x^4) \\ -\frac{i\omega^3}{108}\Delta x^3 + \mathcal{O}(\Delta x^4) \end{pmatrix}$$

- P^2 case

$$V_0 - \hat{\mathbf{u}}(0) = \begin{pmatrix} -\frac{(3 + 8\sqrt{6})\omega^4}{60000}\Delta x^4 + \mathcal{O}(\Delta x^5) \\ -\frac{(3 - 8\sqrt{6})\omega^4}{60000}\Delta x^4 + \mathcal{O}(\Delta x^5) \\ -\frac{i\omega^5}{18000}\Delta x^5 + \mathcal{O}(\Delta x^6) \end{pmatrix}$$

$$V_1 = \begin{pmatrix} \frac{(207 + 552\sqrt{6} + 162\sqrt{46} - 11\sqrt{69})\omega^4}{8280000}\Delta x^4 + \mathcal{O}(\Delta x^5) \\ \frac{(207 - 552\sqrt{6} - 162\sqrt{46} + 11\sqrt{69})\omega^4}{8280000}\Delta x^4 + \mathcal{O}(\Delta x^5) \\ \frac{\omega^4}{480\sqrt{69}}\Delta x^4 + \mathcal{O}(\Delta x^5) \end{pmatrix}$$

$$V_2 = \begin{pmatrix} \frac{(207 + 552\sqrt{6} - 162\sqrt{46} + 11\sqrt{69})\omega^4}{8280000}\Delta x^4 + \mathcal{O}(\Delta x^5) \\ \frac{(207 - 552\sqrt{6} + 162\sqrt{46} - 11\sqrt{69})\omega^4}{8280000}\Delta x^4 + \mathcal{O}(\Delta x^5) \\ -\frac{\omega^4}{480\sqrt{69}}\Delta x^4 + \mathcal{O}(\Delta x^5) \end{pmatrix}$$

- P^3 case

$$V_0 - \hat{\mathbf{u}}(0) = \begin{pmatrix} 1.15 \times 10^{-5} i \omega^5 \Delta x^5 + \mathcal{O}(\Delta x^6) \\ -1.20 \times 10^{-5} i \omega^5 \Delta x^5 + \mathcal{O}(\Delta x^6) \\ 6.52 \times 10^{-6} i \omega^5 \Delta x^5 + \mathcal{O}(\Delta x^6) \\ 4.05 \times 10^{-7} i \omega^7 \Delta x^7 + \mathcal{O}(\Delta x^8) \end{pmatrix}$$

$$V_1 = \begin{pmatrix} -3.06 \times 10^{-6} i \omega^5 \Delta x^5 + \mathcal{O}(\Delta x^6) \\ -1.20 \times 10^{-5} i \omega^5 \Delta x^5 + \mathcal{O}(\Delta x^6) \\ -7.02 \times 10^{-7} i \omega^5 \Delta x^5 + \mathcal{O}(\Delta x^6) \\ 1.19 \times 10^{-5} i \omega^5 \Delta x^5 + \mathcal{O}(\Delta x^6) \end{pmatrix}$$

$$V_2 = \begin{pmatrix} -5.96 \times 10^{-6} i \omega^5 \Delta x^5 + \mathcal{O}(\Delta x^6) \\ 1.24 \times 10^{-6} i \omega^5 \Delta x^5 + \mathcal{O}(\Delta x^6) \\ 3.40 \times 10^{-6} i \omega^5 \Delta x^5 + \mathcal{O}(\Delta x^6) \\ -4.61 \times 10^{-6} i \omega^5 \Delta x^5 + \mathcal{O}(\Delta x^6) \end{pmatrix}$$

$$V_3 = \begin{pmatrix} -2.45 \times 10^{-6} i \omega^5 \Delta x^5 + \mathcal{O}(\Delta x^6) \\ 1.04 \times 10^{-5} i \omega^5 \Delta x^5 + \mathcal{O}(\Delta x^6) \\ -9.22 \times 10^{-6} i \omega^5 \Delta x^5 + \mathcal{O}(\Delta x^6) \\ -7.31 \times 10^{-6} i \omega^5 \Delta x^5 + \mathcal{O}(\Delta x^6) \end{pmatrix}$$

It can be checked that for $k = 1, 2$, and 3 , V_0 approximates $\hat{\mathbf{u}}(0)$ with order $2k + 1$ at downwind points and with order $k + 2$ at other Radau points. $V_l, l = 1, \dots, k$ are of order $(k + 2)$ at Radau points. ■

Proposition 4.20. *Consider LDG methods with polynomial space P^k ($k=1, 2$, and 3) for the linear parabolic problem (4.8) with uniform mesh. Let $\mathbf{u}(T) = \hat{\mathbf{u}}(0) \exp(i\omega x_j - \omega^2 T)$ and $\mathbf{u}_h(T) = \hat{\mathbf{u}}(T) \exp(i\omega x_j)$ be point values of exact and LDG solutions at shifted Radau points on a cell I_j . Let $\mathbf{e} = \mathbf{u} - \mathbf{u}_h$, then for $T > 0$,*

$$\|\mathbf{e}(T)\| \leq C_1 T \Delta x^{2k+2} + C_2 \Delta x^{k+2} + C_3 \exp\left(-C \frac{T}{\Delta x^2}\right) \Delta x^{k+2}, \quad (4.23)$$

where C, C_1, C_2 , and C_3 are positive constants independent of Δx .

Proof. The result can be derived based on Proposition 4.16 and 4.19. The proof is similar to Proposition 4.5.

Remark 4.21. *Similar to the case of DG scheme, the error of the LDG solution can be decomposed as three parts:*

- (a) Dissipation error of the physically relevant eigenvalue in the order of $2k + 2$;
- (b) Projection error $\|\mathbf{u}^* - \mathbf{u}\|$: the numerical solution is much closer to the special

4.2. EIGEN-STRUCTURES OF G: ERROR ESTIMATE

projection of exact solution denoted as \mathbf{u}^* ($\|\mathbf{u}^* - \mathbf{u}_h\| = \mathcal{O}(\Delta x^{2k+2})$), than the exact solution itself ($\|\mathbf{u} - \mathbf{u}_h\| = \mathcal{O}(\Delta x^{k+2})$);

(c) Dissipation of non-physically relevant eigenvectors.

Remark 4.22. Based on the error estimate (4.23), one can conclude

(a) when $T = o(\frac{1}{\Delta x^k})$, $\mathcal{O}(\Delta x^{k+2})$ is the dominant term in (4.23); this term will not grow with time;

(b) when $T = \mathcal{O}(\frac{1}{\Delta x^k})$, $\mathcal{O}(\Delta x^{2k+2})T$ is the dominant term in (4.23); this term grows linearly with time and is of order $2k + 2$.

As for the DG scheme, it is hard to check numerically the long time behavior of the error of LDG solutions, we propose to use the following corollary as a way to numerically assess our theoretical results above. Our numerical results on LDG method in the next section is based on the following corollary.

Corollary 4.23. Consider LDG methods with polynomial space P^k ($k = 1, 2,$ and 3) for linear parabolic problem (4.8) with uniform mesh. Let $T > t$ and $t = \mathcal{O}(1)$, then

$$\|\mathbf{u}_h(T) - \mathbf{u}_h(t)\exp(-\omega^2(T-t))\| \leq C_1(T-t)\Delta x^{2k+2} + C_2\exp(-\frac{Ct}{\Delta x^2})\Delta x^{k+2},$$

where C , C_1 , and C_2 are positive constants independent of Δx .

Proof. The proof is similar to Corollary 4.9.

$$\begin{aligned}
 & \| \mathbf{u}_h(T) - \mathbf{u}_h(t) \exp(-\omega^2(T-t)) \| \\
 = & \left\| \sum_{l=1}^{k+1} (\exp(\lambda_l T) - \exp(\lambda_l t - (T-t)\omega^2)) V_l \right\| \\
 \leq & \left| \exp(\lambda_1 T) - \exp(\lambda_1 t - \omega^2(T-t)) \right| \|V_1\| \\
 & + \left\| \sum_{l=2}^{k+1} (\exp(\lambda_l T) - \exp(\lambda_l t - \omega^2(T-t))) V_l \right\| \\
 \leq & \left| \exp(\lambda_1(T-t) - \omega^2(T-t)) \right| \exp(\lambda_1 t) \|V_1\| \\
 & + C_2 \exp\left(-\frac{Ct}{\Delta x^2}\right) \Delta x^{k+2} \\
 \leq & C_1(T-t) \Delta x^{2k+2} + C_2 \exp\left(-\frac{Ct}{\Delta x^2}\right) \Delta x^{k+2}.
 \end{aligned}$$

■

Remark 4.24. Similar to Remark 4.10, let $T > t$ and $t = \mathcal{O}(1)$, then

$$\begin{aligned}
 \|e(T)\| &= \| \mathbf{u}(T) - \mathbf{u}_h(T) \| \\
 &= \| \exp(-w^2(T-t)) \mathbf{u}(t) - \mathbf{u}_h(T) \| \\
 &\leq \| \mathbf{u}_h(T) - \exp(-w^2(T-t)) \mathbf{u}_h(t) \| + \left| \exp(-w^2(T-t)) \right| \| \mathbf{u}(t) - \mathbf{u}_h(t) \| \\
 &\leq \| \mathbf{u}_h(T) - \exp(-w^2(T-t)) \mathbf{u}_h(t) \| + \| \mathbf{u}(t) - \mathbf{u}_h(t) \|.
 \end{aligned}$$

From Corollary 4.23, the dominant term in $\| \mathbf{u}_h(2n\pi + t) - \exp(-w^2(T-t)) \mathbf{u}_h(t) \|$ is order of $2k+2$. Thus the error $e(T)$ of LDG scheme will not grow until $T = \mathcal{O}\left(\frac{1}{\Delta x^k}\right)$.

4.2.3 Supraconvergence of DG and LDG Schemes

In [120], a so-called supraconvergence property of the DG scheme was studied. It was discovered that the leading term of the local truncation error for the DG scheme is first order accurate when piecewise P^1 polynomial is used with basis functions being Lagrangian interpolant based on uniformly distributed points (i.e. $\phi_{j-\frac{1}{4}}$ and $\phi_{j+\frac{1}{4}}$). In this section, we further study the supraconvergence property of the DG and LDG scheme based on our analysis for the eigen-structure of amplification matrices G .

Firstly, we look into the DG scheme for the model problem (4.1). Without loss of generality, we assume that $a = 1$. Denote D to be the temporal differentiation operator of $\hat{\mathbf{u}}(0)$ with $D\hat{\mathbf{u}}(0) = -i\omega\hat{\mathbf{u}}(0)$. Then the local truncation error denoted as LTE satisfies

$$LTE = D\hat{\mathbf{u}}(0) - G\hat{\mathbf{u}}(0). \quad (4.24)$$

Keeping the notation in equation (4.7), we have

$$\begin{aligned} LTE &= D\hat{\mathbf{u}}(0) - G\hat{\mathbf{u}}(0) \\ &= (-i\omega - \lambda_0)V_0 - \sum_{l=1}^k (\lambda_l + i\omega)V_l. \end{aligned}$$

From our analysis in Section 4.2.1, we have $(-i\omega - \lambda_0)V_0 = \mathcal{O}(\Delta x^{2k+1})$, $(\lambda_l + i\omega) = \mathcal{O}(\Delta x^{-1})$ and $V_l = \mathcal{O}(\Delta x^{k+2})$ for $l \geq 1$ where $k = 1, 2$, and 3 . Thus the LTE based on shifted Radau points is obtained as:

$$LTE = \mathcal{O}(\Delta x^{k+1}). \quad (4.25)$$

It is observed that, due to our estimate for $\lambda_l = \mathcal{O}(\Delta x^{-1})$ with $l \geq 1$, the order of local truncation error is one order lower than the error in (4.13). Note that the local truncation error analyzed based on uniformly distributed points [120] is one order lower than that based on shifted Radau points (4.25). This fact partially explains the supraconvergence property of DG schemes discussed in [120].

Similar results can be derived for LDG schemes.

$$\begin{aligned}
 LTE &= D^2 \hat{\mathbf{u}}(0) - G \hat{\mathbf{u}}(0) \\
 &= (-\omega^2 - \lambda_1) V_1 - \sum_{l=2}^{k+1} (\lambda_l + \omega^2) V_l \\
 &= \mathcal{O}(\Delta x^k),
 \end{aligned} \tag{4.26}$$

since $-\omega^2 - \lambda_0 = \mathcal{O}(\Delta x^{2k+2})$, $(\lambda_l + \omega^2) = \mathcal{O}(\Delta x^{-2})$ and $V_l = \mathcal{O}(\Delta x^{k+2})$ for $l \geq 1$, when $k = 1, 2$, and 3 . The order of local truncation error is two orders lower than the error in equation (4.23) based on shifted Radau points. Similarly, when uniformly distributed Lagrangian basis is used, we have $LTE = \mathcal{O}(\Delta x^{k-1})$.

4.2.4 Fully Discrete RKDG Schemes

An analysis of fully discretized RKDG schemes will be presented below. Without loss of generality, assume $a = 1$ in (2.3).

Let \tilde{u} be the approximation solution of the fully discretized version of (2.3) obtained by using an explicit RK method of order p [43]. Denote $\tilde{\mathbf{u}}_h(T) = \tilde{\mathbf{u}}^n \exp(i\omega x_j)$ to be the point values of the solution \tilde{u} at shifted Radau points on a cell I_j at time

T . Then

$$\tilde{\mathbf{u}}^n = R^n \hat{\mathbf{u}}(0), \quad n = \frac{T}{\Delta t},$$

with

$$R = 1 + \Delta t G + \frac{\Delta t^2}{2!} G^2 + \cdots + \frac{\Delta t^p}{p!} G^p \quad (4.27)$$

for an explicit p^{th} order RK method, where the amplification matrix G is defined in (4.5). Consider the eigen-structure of $G = Q\Lambda Q^{-1}$, where $Q = [V_1, \dots, V_{k+1}]$ is the matrix with its columns being G 's eigenvectors and $\Lambda = \text{diag}(\lambda_1, \dots, \lambda_{k+1})$ where $\lambda_i, i = 1, \dots, k+1$ are G 's eigenvalues, then

$$\tilde{\mathbf{u}}^n = \sum_{l=1}^{k+1} \left(1 + \Delta t \lambda_l + \cdots + \frac{\Delta t^p}{p!} \lambda_l^p \right)^n V_l. \quad (4.28)$$

Proposition 4.25. *With the same assumption as Proposition 4.5. Denote $\tilde{\mathbf{u}}_h$ as the numerical solution of a fully discretized DG scheme with V_h^k as the approximation space and with a p^{th} order RK method. Let $\tilde{\mathbf{e}} = \mathbf{u} - \tilde{\mathbf{u}}_h$. Then for $T > 0$, and under certain linear stability constrain for time step Δt , we have the error estimate*

$$\|\tilde{\mathbf{e}}(T)\| \leq C_1 T \Delta x^{2k+1} + C_2 \Delta x^{k+2} + C_3 T \Delta t^p, \quad (4.29)$$

where C_1, C_2 , and C_3 are positive constants independent of Δx and Δt . Here $\|\cdot\|$ can be any norm for vectors.

Proof.

$$\|\tilde{\mathbf{e}}(T)\| = \|\mathbf{u}(T) - \tilde{\mathbf{u}}_h(T)\| \leq \|\mathbf{u}(T) - \mathbf{u}_h(T)\| + \|\mathbf{u}_h(T) - \tilde{\mathbf{u}}_h(T)\|. \quad (4.30)$$

By Proposition 4.5 with $a = 1$, we have

$$\|\mathbf{u}(T) - \mathbf{u}_h(T)\| \leq C_1 T \Delta x^{2k+1} + C'_2 \Delta x^{k+2}, \quad (4.31)$$

where C_1, C'_2 are positive constants independent of Δx . We only need to estimate the second part on the r.h.s. of (4.30). By Equation (4.7) and (4.28), after Taylor expansion, we have

$$\begin{aligned} \|\mathbf{u}_h(T) - \tilde{\mathbf{u}}_h(T)\| &\leq C \|\hat{\mathbf{u}}(T) - \tilde{\mathbf{u}}(T)\| & (4.32) \\ &\leq C \sum_{l=1}^{k+1} \|\exp(\lambda_l T) V_l - \sum_{l=1}^{k+1} \left(1 + \lambda_l \Delta t + \dots + \frac{\lambda_l^p \Delta t^p}{p!}\right)^{\frac{T}{\Delta t}} V_l\| \\ &\leq C \sum_{l=1}^{k+1} \left| \exp(\lambda_l T) - \left(1 + \lambda_l \Delta t + \dots + \frac{\lambda_l^p \Delta t^p}{p!}\right)^{\frac{T}{\Delta t}} \right| \|V_l\|. \end{aligned}$$

- Firstly, we consider λ_1 .

$$\begin{aligned} &\left| \exp(\lambda_1 n \Delta t) - \left(1 + \Delta t \lambda_1 + \dots + \frac{\Delta t^p \lambda_1^p}{p!}\right)^n \right| \\ &\leq \left| \exp(\lambda_1 \Delta t) - \left(1 + \Delta t \lambda_1 + \dots + \frac{\Delta t^p \lambda_1^p}{p!}\right) \right| \\ &\quad \left| \sum_{m=0}^{n-1} \exp(\lambda_1 m \Delta t) \left(1 + \Delta t \lambda_1 + \dots + \frac{\Delta t^p \lambda_1^p}{p!}\right)^{n-1-m} \right| \\ &\leq C'_3 T \Delta t^p, \end{aligned} \quad (4.33)$$

4.2. EIGEN-STRUCTURES OF G: ERROR ESTIMATE

where the last inequality requires the estimate about λ_1 from Proposition. 4.1,

$$|\exp(\lambda_1 m \Delta t)| < 1, \quad \forall m.$$

We also need

$$\left|1 + \lambda_1 \Delta t + \dots + \frac{(\lambda_1 \Delta t)^p}{p!}\right|^m \leq C'_3, \quad \forall m \leq n - 1. \quad (4.34)$$

To guarantee such inequality, the time step Δt has to be small enough for stability. Below, we only consider a simple case to illustrate how time step restriction is related to the stability, and the equation (4.34). The readers are referred to [116] for more details.

The RKDG scheme with P^1 and RK2. Denote $cf l = \frac{\Delta t}{\Delta x}$ and $\xi = \omega \Delta x$. From Proposition 4.1, we have

$$\begin{aligned} \lambda_0 &= -i\omega - \frac{\omega^4}{72} \Delta x^3 - \frac{i\omega^5}{270} \Delta x^4 + O(\Delta x^5) \\ &= -i\omega - \frac{\xi^4}{72\Delta x} - \frac{i\xi^5}{270\Delta x} + O(\xi^5). \end{aligned}$$

Then

$$\begin{aligned} \left|1 + \lambda_0 \Delta t + \frac{\lambda_0^2 \Delta t^2}{2}\right|^2 &= \left|1 - icfl\xi - \frac{cfl^2}{2}\xi^2 - \frac{cfl}{72}\xi^4 + \mathcal{O}(\xi^5)\right|^2 \\ &= \left|1 + \left(\frac{cfl^4}{4} - \frac{cfl}{36}\right)\xi^4 + \mathcal{O}(\xi^6)\right|. \end{aligned}$$

We require the lead term

$$\frac{cfl^4}{4} - \frac{cfl}{36} < 0,$$

which leads to $cfl < 0.48075$. Note that this is necessary, but not sufficient condition to have (4.34) for all $\xi \in [0, 2\pi]$.

- For $\lambda_l, l \geq 1$. From Proposition 4.1, we have

$$\begin{aligned} & \left| \exp(\lambda_l T) - \left(1 + \lambda_l \Delta t + \dots + \frac{\lambda_l^p \Delta t^p}{p!} \right)^{\frac{T}{\Delta t}} \right| \\ & \leq \left| \exp(\lambda_l T) \right| + \left| \left(1 + \lambda_l \Delta t + \dots + \frac{\lambda_l^p \Delta t^p}{p!} \right)^{\frac{T}{\Delta t}} \right| \\ & \leq C_2^l, \end{aligned} \tag{4.35}$$

if

$$\left| 1 + \lambda_l \Delta t + \dots + \frac{\lambda_l^p \Delta t^p}{p!} \right|^n \leq C_2'. \tag{4.36}$$

This inequality would be valid only for sufficiently small time step as for λ_0 . Below, we use the same example as above to derive a necessary condition for linear stability.

The RKDG scheme with P^1 and SSPRK(2,2). From Proposition 4.1, we have

$$\lambda_1 = -\frac{6}{\Delta x} + \mathcal{O}(1),$$

then

$$\left|1 + \lambda_1 \Delta t + \frac{\lambda_1^2 \Delta t^2}{2}\right| \leq |1 - 6cfl + 18cfl^2| + \mathcal{O}(\xi).$$

To ensure stability, we need

$$|1 - 6cfl + 18cfl^2| < 1, \quad i.e., \quad cfl < \frac{1}{3}.$$

Combining the two time step restrictions, we get $cfl < \frac{1}{3}$, which is a necessary condition for linear stability of the RKDG scheme with P^1 and SSPRK(2,2). Such time step restriction is consistent with the classical results in [32]. Similar analysis can be performed for general RKDG methods.

Now we can finish the proof. Under certain linear stability constrain of time step Δt , we have from the above discussions

$$\|\mathbf{u}_h(T) - \tilde{\mathbf{u}}_h(T)\| \leq C_3 T \Delta t^p + C \sum_l C_2^l \Delta x^{k+2}, \quad (4.37)$$

where $C_3 = CC'_3$ is a positive constant independent of Δx and Δt . Combine (4.30), (4.31) and (4.37), we derive the final error estimate with $C_2 = C'_2 + C \sum_l C_2^l$. ■

Remark 4.26. *In the proof, we do not intend to derive a necessary and sufficient condition of the time step restriction. Related work on this topic can be found [61]. We only assume such time step restriction is satisfied for the linear stability, then the error estimate for the fully discretized scheme can be derived.*

Similarly, an error estimate of a fully discretized LDG scheme with p^{th} order explicit RK method can also be derived.

Proposition 4.27. *With the same assumption as Proposition 4.20. Denote $\tilde{\mathbf{u}}_h$ as the numerical solution of a fully discretized LDG scheme with V_h^k as the approximation space and p^{th} order RK method and $\tilde{\mathbf{e}} = \mathbf{u} - \tilde{\mathbf{u}}_h$. Then for $T > 0$, and under certain linear stability constrain for time step Δt , we have the error estimate*

$$\|\tilde{\mathbf{e}}(T)\| \leq C_1 T \Delta x^{2k+2} + C_2 \Delta x^{k+2} + C_3 T \Delta t^p, \quad (4.38)$$

where C_1 , C_2 , and C_3 are positive constants independent of Δx and Δt . Here that $\|\cdot\|$ can be any norm for vectors.

Proof. The proof is similar to Proposition 4.25.

Remark 4.28. *For a fully discretized RKDG scheme, the results in Remark 3.8 still hold, provided the order of RK method $p \geq 2k + 1$. If a low order of RK method is used with $\Delta t = \mathcal{O}(\Delta x)$, the error will grow linearly with time, see [24] for a detailed numerical comparison. In general, in order to study the superconvergence property of DG and LDG scheme, we use very high order RK method (SSPRK(9,9)) or let $\Delta t = \mathcal{O}(\Delta x^2)$ to reduce temporal errors.*

Numerical Results

Below, we provide a collection of 1-D and 2-D numerical experiments to verify our theoretical analysis. DG schemes for a one-dimensional linear equation based on non-uniform meshes, 1-D nonlinear Burgers' equation, 2-D systems such as wave equations and Maxwell equations are also investigated to explore superconvergence properties of DG methods in a more general setting. We do not report DG errors at

Radau points due to superconvergence properties of physically relevant eigenvectors, as they have been well documented in [124]. In most of our simulations below, we use SSPRK(9,9) for linear problems and use SSPRK(5,4) for nonlinear problems. We also reduce the time step size so that the spatial error from DG schemes is the dominant error. We use Gaussian quadrature rule with $k + 1$ quadrature points to compute the volume integral in the DG formulation, which is exact for linear cases and is of order $2k + 2$ for general variable coefficient cases and nonlinear cases. We use the same quadrature rule to compute the L^2 norm of error functions. We remark that since Gaussian quadrature rule is of order $2k + 2$ for $k + 1$ Gauss points, $(2k + 1)^{th}$ order of convergence will be able to be maintained numerically.

Example 4.29. Consider a 1-D linear advection equation:

$$\begin{cases} u_t + u_x = 0, & x \in [0, 2\pi], \\ u(0, x) = \sin(x), \end{cases} \quad (4.39)$$

with periodic boundary conditions. In order to make the temporal error negligible compared with the spatial error, we adopt SSPRK(9,9) [43] to solve $du/dt = Lu$, where L is the DG discretization operator. Recall that in order to use SSPRK(9,9), L should be a linear operator. In the simulation, we choose CFL=0.3 for P^1 , CFL= 0.2 for P^2 and CFL= 0.1 for P^3 .

In this example, we consider two types of DG errors. One is the regular DG error ($e = u - u_h$), and the other is

$$\bar{e}_n = |u_h(T = 2\pi) - u_h(T = 2(n + 1)\pi)|, \quad n \in \mathbb{N}, \quad (4.40)$$

whose order of convergence is $2k + 1$ as discussed in Corollary 4.9. In Table 4.1 and 4.2 we report the L^2 norm of \bar{e}_1 and \bar{e}_2 , and the order of accuracy for P^1 - P^3 . $(2k + 1)^{th}$ order of accuracy is observed, as expected from Corollary 4.9. It is also observed that $\bar{e}_2 \approx 2\bar{e}_1$ indicating the linear growth rate of the error in time. This is also consistent with the Corollary 4.9. In Figure 4.2, the evolution of L^2 norms of the regular DG errors $e(t)$ and $\bar{e}_n(t)$ in a log-log scale is provided. The magnitude of the regular DG error is observed to be much larger than \bar{e}_n . It is observed that the regular DG error does not grow for a long time period, while linear growth rate of the error \bar{e}_n with respect to time is observed, see Remark 4.10. In Figure 4.3 - 4.5, we plot the regular errors of DG schemes and the errors \bar{e}_1 with $n = 1$ in equation (4.40) in a logarithmic scale for P^1 - P^3 . Highly oscillatory nature of DG errors is observed as in [27]. On the other hand, \bar{e}_1 does not oscillate as much; the magnitude of \bar{e}_1 is much smaller and the order is $2k + 1$.

A non-uniform mesh with two different mesh sizes is used to assess superconvergence properties of DG with non-uniform meshes. We set $\Delta x_{\text{left}}/\Delta x_{\text{right}} = 3/2$, where Δx_{left} and Δx_{right} is the mesh size for the left and right half part of the domain respectively. In Table 4.3, we report the error \bar{e}_1 . An order of $2k + 1$ is observed for P^1 and P^2 cases, but not for the P^3 case. We remark that in [124], with non-uniform meshes, $(2k + 1)^{th}$ order of convergence is not observed well at downwind points for the case of P^3 either. Figure 4.6 shows error \bar{e}_1 in logarithmic scale for P^1 - P^3 when $N = 70$. Non-oscillatory errors are observed for P^1 and P^2 even around the interface of different mesh sizes. For P^3 , the error is observed to be oscillatory near the discontinuity of mesh size.

4.2. EIGEN-STRUCTURES OF G : ERROR ESTIMATE

Table 4.1: Linear advection $u_t + u_x = 0$ with initial condition $u(0, x) = \sin(x)$. The L^2 norm of \bar{e}_1 and the orders of accuracy. Uniform mesh.

mesh	P^1		P^2		P^3	
	L^2 error	order	L^2 error	order	L^2 error	order
20	4.74E-03	–	4.71E-06	–	2.39E-09	–
30	1.41E-03	2.98	6.22E-07	4.99	1.39E-10	7.01
40	5.98E-04	2.99	1.48E-07	5.00	1.86E-11	7.00
50	3.06E-04	3.00	4.84E-08	5.00	3.90E-12	7.00
60	1.77E-04	3.00	1.95E-08	5.00	1.09E-12	7.00

Table 4.2: Linear advection $u_t + u_x = 0$ with initial condition $u(0, x) = \sin(x)$. The L^2 norm of \bar{e}_2 and the orders of accuracy. Uniform mesh.

mesh	P^1		P^2		P^3	
	L^2 error	order	L^2 error	order	L^2 error	order
20	9.47E-03	–	9.42E-06	–	4.76E-09	–
30	2.83E-03	2.98	1.24E-06	4.99	2.79E-10	7.00
40	1.20E-03	2.99	2.96E-07	5.00	3.72E-11	7.00
50	6.13E-04	3.00	9.69E-08	5.00	7.81E-12	7.00
60	3.55E-04	3.00	3.89E-08	5.00	2.18E-12	7.00

Example 4.30. Consider the same advection equation as Example 4.29 but with a different initial condition:

$$u(x, 0) = \exp(\sin(x)). \quad (4.41)$$

Note that the initial condition contains infinite number of Fourier modes. In Table 4.4, we report the L^2 norm of error \bar{e}_1 and the orders of accuracy. $(2k + 1)^{th}$ order of accuracy is observed as expected.

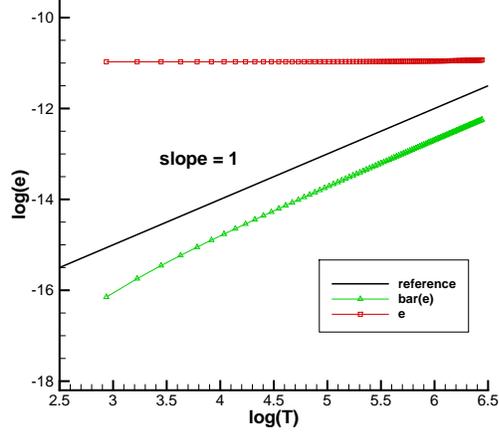


Figure 4.2: DG with P^2 , $u_t + u_x = 0$, the evolution of L^2 norms of regular DG error and error $|\bar{e}_n|$. A reference line with slope 1 is plotted as the reference of linear growth rate of $|\bar{e}_n|$ with respect to time. In the simulation, we use RKDG P^2 with mesh size $\Delta x = \frac{2\pi}{50}$ and CFL= 0.2.

Example 4.31. Consider the following linear variable coefficient equation:

$$\begin{cases} u_t + (a(x)u)_x = b(x, t), & x \in [0, 2\pi], \\ u(x, 0) = \sin(x), \end{cases} \quad (4.42)$$

with periodic boundary conditions. And $a(x)$ and $b(t, x)$ are given by

$$a(x) = \sin(x) + 2, \quad b(t, x) = (\sin(x) + 3) \cos(x + t) + \cos(x) \sin(x + t). \quad (4.43)$$

The exact solution is $u(t, x) = \sin(x + t)$. We use SSPRK(5,4) for the temporal discretization. In the simulation, we let CFL= $\frac{1}{3}$ for P^1 , CFL= $\frac{1}{5}$ for P^2 and CFL= $\frac{1}{7}$ for P^3 , and time step $\Delta t = \text{CFL}\Delta x^2$ to reduce the temporal error. We report the L^2 norm of error \bar{e}_1 as defined in equation (4.40) and numerical orders of accuracy in

4.2. EIGEN-STRUCTURES OF G: ERROR ESTIMATE

Table 4.3: Linear advection $u_t + u_x = 0$ with initial condition $u(0, x) = \sin(x)$. The L^2 norm of \bar{e}_1 and the order of accuracy. Nonuniform mesh with two different mesh sizes. $\Delta x_{\text{left}}/\Delta x_{\text{right}} = 3/2$.

mesh	P^1		P^2		P^3	
	L^2 error	order	L^2 error	order	L^2 error	order
20	3.37E-03	–	5.61E-06	–	2.45E-08	–
30	1.01E-03	2.98	7.42E-07	4.99	1.49E-09	6.90
40	4.27E-04	2.99	1.76E-07	5.00	2.32E-10	6.46
50	2.19E-04	2.99	5.78E-08	5.00	6.00E-11	6.07
60	1.27E-04	3.00	2.32E-08	5.00	2.03E-11	5.93
70	7.98E-05	3.00	1.07E-08	5.00	2.03E-11	5.95

Table 4.4: Linear advection $u_t + u_x = 0$ with initial condition $u(0, x) = \exp(\sin(x))$. The L^2 norm of \bar{e}_1 and the orders of accuracy. Uniform mesh.

mesh	P^1		P^2		P^3	
	L^2 error	order	L^2 error	order	L^2 error	order
30	7.80E-03	–	2.68E-05	–	7.08E-08	–
40	3.42E-03	2.86	6.41E-06	4.96	9.54E-09	6.97
50	1.78E-03	2.93	2.11E-06	4.98	2.01E-09	6.98
60	1.04E-03	2.96	8.51E-07	4.99	5.62E-10	6.99
70	6.57E-04	2.97	3.94E-07	4.99	1.91E-10	6.99

Table 4.5. $(2k + 1)^{\text{th}}$ order of accuracy is observed, although our analysis only works for the constant coefficient problems.

Example 4.32. Consider the following nonlinear Burgers' equation with a source term:

$$\begin{cases} u_t + (u^2)_x = b(t, x) & x \in [0, 2\pi], \\ u(0, x) = \sin(x) + c, \end{cases} \quad (4.44)$$

4.2. EIGEN-STRUCTURES OF G : ERROR ESTIMATE

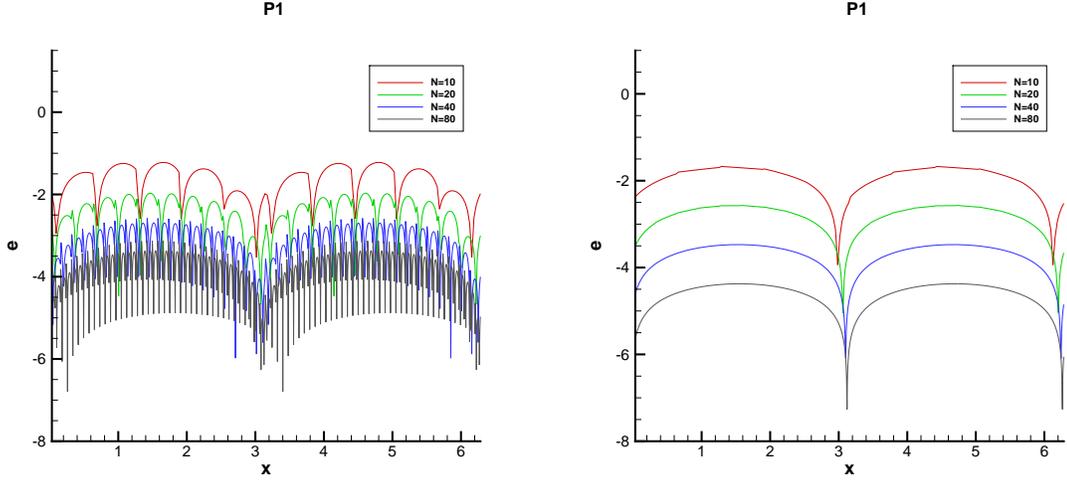


Figure 4.3: DG with P^1 , $u_t + u_x = 0$, regular error $|u(T = 4\pi) - u_h(T = 4\pi)|$ (left); error $|\bar{e}_1|$ (right).

with periodic boundary conditions. Here c is a real number and $b(t, x)$ is given by

$$b(t, x) = \cos(x + t)(2c + 1 + 2 \sin(x + t)). \quad (4.45)$$

The exact solution is $u(t, x) = \sin(x + t) + c$. We use SSPRK(5,4) for the temporal discretization. In the simulation, we let $\text{CFL} = \frac{1}{3}$ for P^1 , $\text{CFL} = \frac{1}{5}$ for P^2 and $\text{CFL} = \frac{1}{7}$

Table 4.5: Linear variable coefficient problem. The L^2 norm of \bar{e}_1 and the orders of accuracy. Uniform mesh.

	P^1		P^2		P^3	
mesh	L^2 error	order	L^2 error	order	L^2 error	order
20	5.00E-04	–	9.43E-07	–	8.70E-08	–
30	1.68E-04	2.68	1.24E-07	5.00	5.66E-09	6.74
40	7.37E-05	2.87	2.95E-08	5.00	7.42E-10	7.06
50	3.83E-05	2.93	9.67E-09	5.00	1.20E-10	8.17
60	2.23E-05	2.96	3.88E-09	5.00	2.26E-11	9.15

4.2. EIGEN-STRUCTURES OF G : ERROR ESTIMATE

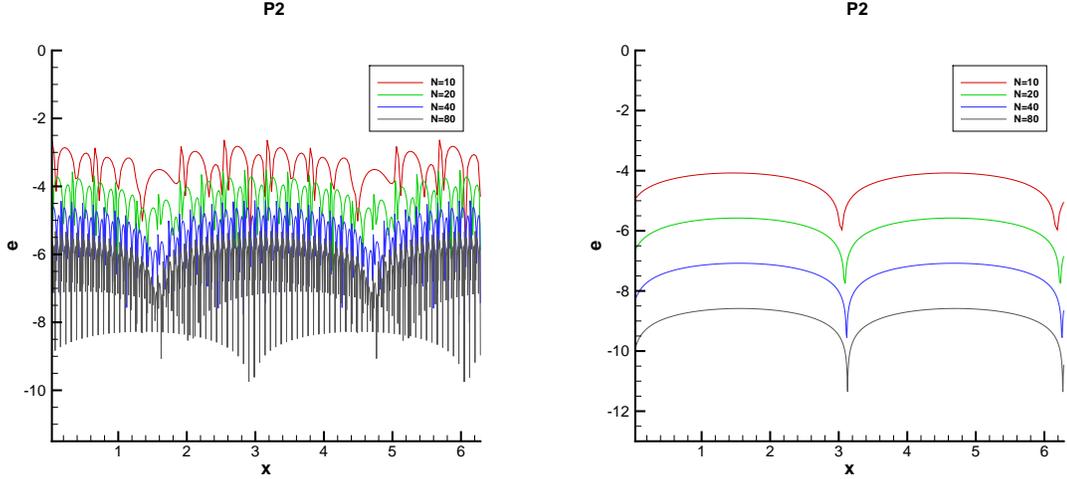


Figure 4.4: DG with P^2 , $u_t + u_x = 0$, regular error $|u(T = 4\pi) - u_h(T = 4\pi)|$ (left); error $|\bar{e}_1|$ (right).

for P^3 , and time step $\Delta t = \text{CFL}\Delta x^2$ to reduce the time errors. Firstly, we let $c = 0$, and compute the error \bar{e}_1 . In Table 4.6, the L^2 error and the orders of accuracy are reported. $(2k + 1)^{\text{th}}$ order of accuracy is not observed. Note that the wave speed is $2u$, and there exist some regions around which wave travels at very slow speed (i.e. the region around which $u = 0$). In these regions, the non-physically relevant eigenvectors are damped very slowly with time, see the second term on the r.h.s. of equation (4.14), see also Remark 4.11. $(2k + 1)^{\text{th}}$ order is not observed numerically. In Figure 4.7, \bar{e}_1 in logarithmic scale for $P^1 - P^3$ cases are plotted when $N = 100$. It is observed that \bar{e}_1 dominates around $x = \pi$. Then, we set $c = 2$. In this case, there is a positive lower bound on the wave speed. We report the L^2 norm of \bar{e}_1 and numerical orders of accuracy in Table 4.7. The $(2k + 1)^{\text{th}}$ order is observed. In Figure 4.8 - 4.10, we plot the regular errors of DG schemes and \bar{e}_1 in logarithmic scale for $P^1 - P^3$ cases. While highly oscillatory nature of regular errors is observed, \bar{e}_1 is observed

4.2. EIGEN-STRUCTURES OF G : ERROR ESTIMATE

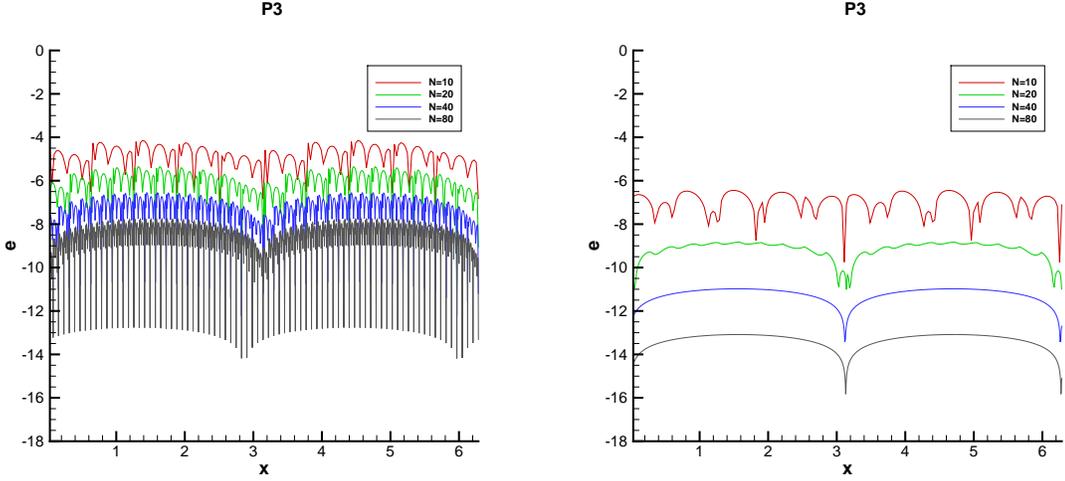


Figure 4.5: DG with P^3 , $u_t + u_x = 0$, regular error $|u(T = 4\pi) - u_h(T = 4\pi)|$ (left); error $|\bar{e}_1|$ (right).

to be much less oscillatory with much smaller magnitude. We remark that although for the $c = 0$ case, $(2k + 1)^{th}$ order of accuracy can't be observed numerically, the long time behavior of errors as commented in Remark 4.8 still holds.

Example 4.33. We consider the following 1-D system:

$$\begin{cases} \begin{pmatrix} u \\ v \end{pmatrix}_t + \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} u \\ v \end{pmatrix}_x = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \\ u(0, x) = \sin x, \\ v(0, x) = \cos x, \end{cases} \quad (4.46)$$

with periodic boundary conditions. Note that this is a 1-D wave equation written as a first order hyperbolic linear system. The upwind flux is used for the DG discretization and SSPRK(9,9) is used for temporal discretization in the simulation. We let $CFL = \frac{1}{3}$

4.2. EIGEN-STRUCTURES OF G: ERROR ESTIMATE

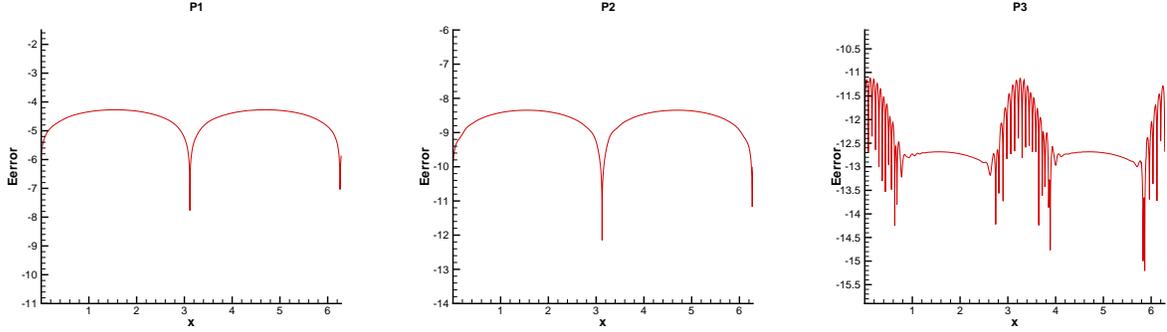


Figure 4.6: DG scheme for $u_t + u_x = 0$, error $|\bar{e}_1|$, non-uniform mesh with two different mesh sizes, $\Delta x_{\text{left}}/\Delta x_{\text{right}} = 3/2$, $N = 70$, P^1 (left), P^2 (middle), P^3 (right).

Table 4.6: Burgers' equation: $u_t + (u^2)_x = b(t, x)$ with initial condition $u(0, x) = \sin(x)$. The L^2 norm of \bar{e}_1 measured from solutions at Gaussian points of each cell and the orders of accuracy. Uniform mesh.

	P^1		P^2		P^3	
mesh	L^2 error	order	L^2 error	order	L^2 error	order
50	1.30E-07	–	8.34E-09	–	1.93E-10	–
60	9.21E-08	1.91	3.96E-09	4.08	1.10E-10	3.09
70	6.79E-08	1.98	1.85E-09	4.95	5.95E-11	3.98
80	5.23E-08	1.95	1.13E-09	3.69	4.00E-11	2.97
90	4.18E-08	1.90	4.76E-10	7.34	2.75E-11	3.19
100	3.43E-08	1.88	3.29E-10	3.48	1.80E-11	4.00

for P^1 , $\text{CFL}=\frac{1}{5}$ for P^2 , and $\text{CFL}=\frac{1}{7}$ for P^3 . We report the L^2 norm of \bar{e}_1 and the orders of accuracy for the u variable in Table 4.8. $(2k + 1)^{\text{th}}$ order is observed.

Example 4.34. We consider the following 1-D heat equation:

$$\begin{cases} u_t = u_{xx}, & x \in [0, 2\pi], \\ u(0, x) = \sin(x), \end{cases} \quad (4.47)$$

4.2. EIGEN-STRUCTURES OF G : ERROR ESTIMATE

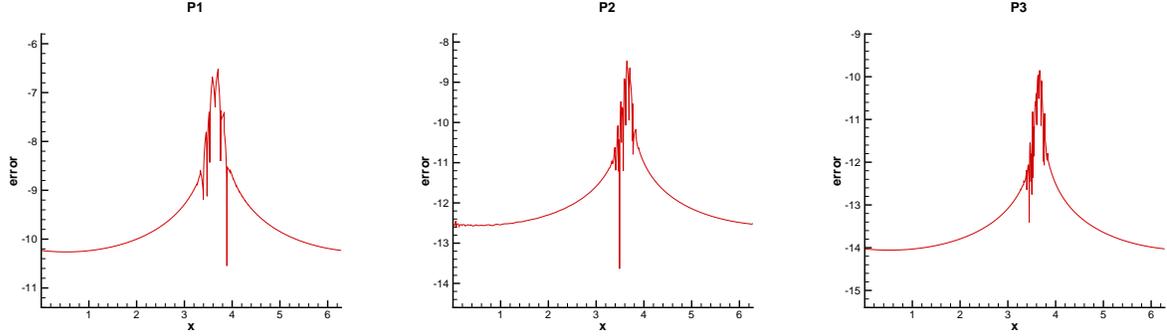


Figure 4.7: DG scheme for Burgers' equation with initial condition $u(0, x) = \sin(x)$, error \bar{e}_1 , $N = 100$, P^1 (left), P^2 (middle), P^3 (right).

Table 4.7: Nonlinear Burgers' equation: $u_t + (u^2)_x = b(t, x)$ with initial condition $u(0, x) = \sin(x) + 2$. The L^2 norm of \bar{e}_1 measured from solutions at Gaussian points of each cell and the orders of accuracy. Uniform mesh.

	P^1		P^2		P^3	
mesh	L^2 error	order	L^2 error	order	L^2 error	order
50	3.64E-05	–	6.61E-09	–	1.76E-12	–
60	2.11E-05	2.98	2.66E-09	5.00	2.53E-13	10.64
70	1.33E-05	2.99	1.23E-09	5.00	7.45E-14	7.92
80	8.93E-06	2.99	6.30E-10	5.00	2.88E-14	7.12
90	6.27E-06	2.99	3.50E-10	5.00	1.26E-14	7.03
100	4.57E-06	3.00	2.06E-10	5.00	6.01E-15	7.02

with periodic boundary conditions. The exact solution of (4.47) is

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

We use a LDG method with SSPRK(9,9) temporal discretization to solve the equation. In the simulation, we use CFL=0.01 and $\Delta t = \text{CFL} \Delta x^2$. We let u_h denote the numerical solution for u and p_h denote that for the derivative u_x . We compare the

4.2. EIGEN-STRUCTURES OF G : ERROR ESTIMATE

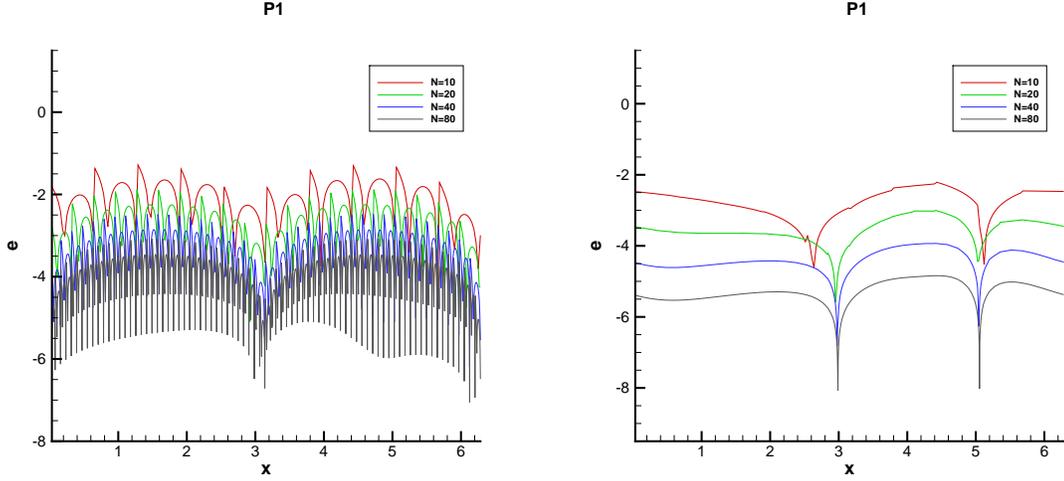


Figure 4.8: DG with P^1 ; Burgers' equation with initial condition $u(0, x) = \sin(x) + 2$; regular error $|u(T = 4\pi) - u_h(T = 4\pi)|$ (left); $|\bar{e}_1|$ (right).

the error

$$\tilde{e}_1 = \exp(-1)u_h(T = 1) - u_h(T = 2), \quad (4.48)$$

and

$$\tilde{e}_2 = \exp(-1)p_h(T = 1) - p_h(T = 2). \quad (4.49)$$

In Table 4.9 and Table 4.10, we report the L^2 norm and orders of accuracy of error

Table 4.8: The linear system (4.46). The L^2 norm of \bar{e}_1 and the orders of accuracy for variable u . Uniform mesh.

mesh	P^1		P^2		P^3	
	L^2 error	order	L^2 error	order	L^2 error	order
30	6.52E-04	–	3.38E-07	–	8.58E-11	–
40	2.80E-04	2.94	8.11E-08	4.86	1.15E-11	6.99
50	1.45E-04	2.96	2.67E-08	4.97	2.42E-12	6.98
60	8.43E-05	2.97	1.08E-08	4.98	6.78E-13	6.98
70	5.33E-05	2.97	5.00E-09	4.98	2.31E-13	6.99

4.2. EIGEN-STRUCTURES OF G : ERROR ESTIMATE

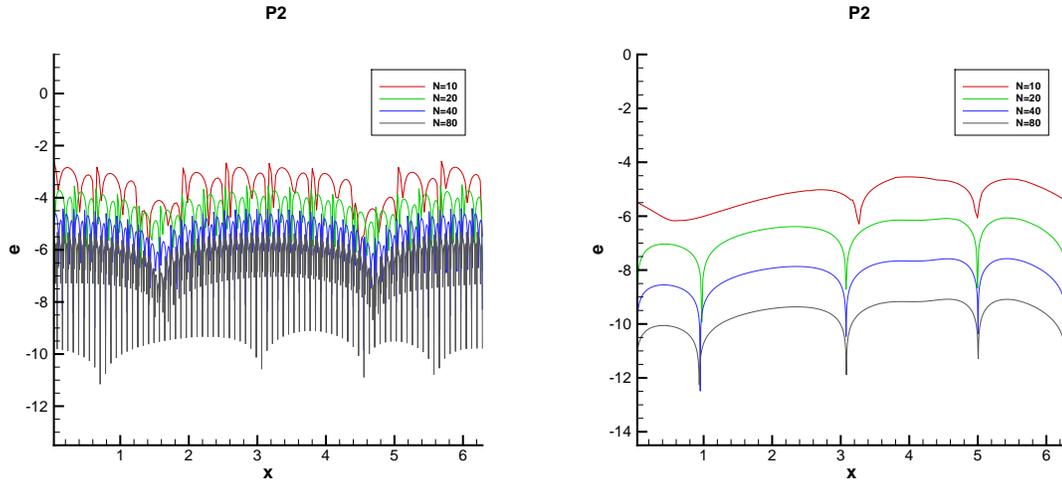


Figure 4.9: DG with P^2 ; Burgers' equation with initial condition $u(0, x) = \sin(x) + 2$, regular error $|u(T = 4\pi) - u_h(T = 4\pi)|$ (left); $|\bar{e}_1|$ (right).

\tilde{e}_1 and error \tilde{e}_2 , respectively. $(2k + 2)^{th}$ order of accuracy is observed as expected from Corollary 4.23. In Figure 4.11 - 4.13, we plot the regular errors of LDG schemes and the errors \tilde{e}_1 in logarithmic scale for P^1 - P^3 . Regular errors are observed to be highly oscillatory, while \tilde{e}_1 is much less oscillatory with much smaller magnitude.

Table 4.9: Heat equation $u_t = u_{xx}$. The L^2 norm and orders of accuracy of \tilde{e}_1 . Uniform mesh.

	P^1		P^2		P^3	
mesh	L^2 error	order	L^2 error	order	L^2 error	order
10	3.99E-05	-	8.15E-08	-	1.04E-10	-
20	2.45E-06	4.02	1.27E-09	6.01	4.08E-13	8.00
30	4.83E-07	4.01	1.11E-10	6.00	1.59E-14	8.00
40	1.53E-07	4.00	1.98E-11	6.00	1.60E-15	8.00
50	6.26E-08	4.00	5.18E-12	6.00	2.68E-16	8.00

4.2. EIGEN-STRUCTURES OF G : ERROR ESTIMATE

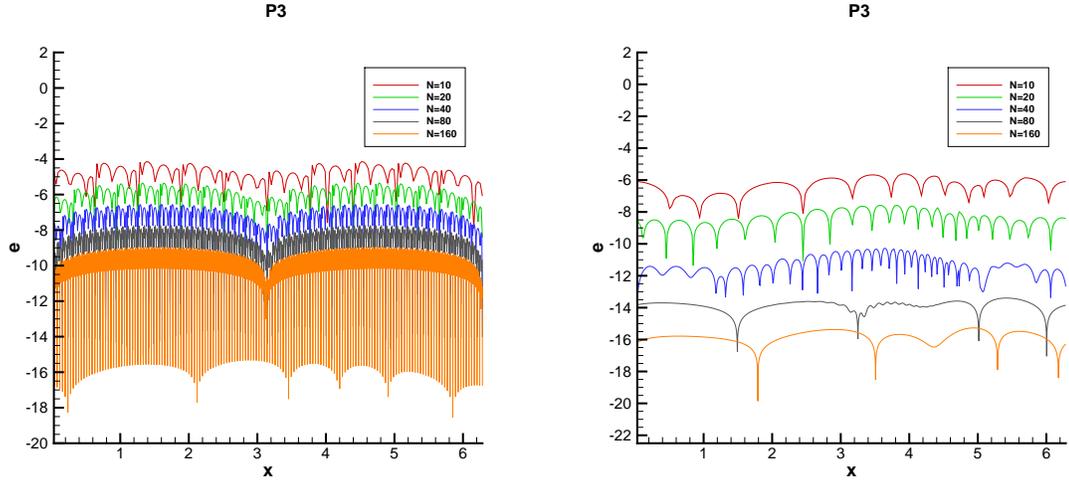


Figure 4.10: DG with P^3 ; Burgers' equation with initial condition $u(0, x) = \sin(x) + 2$; regular error $|u(T = 4\pi) - u_h(T = 4\pi)|$ (left); $|\bar{e}_1|$ (right).

Example 4.35. We consider the following 2-D advection equation:

$$\begin{cases} u_t + u_x + u_y = 0, & (x, y) \in [0, 2\pi] \times [0, 2\pi], \\ u(0, x, y) = \sin(x + y), \end{cases} \quad (4.50)$$

with periodic boundary conditions in both x - and y - directions. The exact solution

Table 4.10: Heat equation $u_t = u_{xx}$. The L^2 norm of \tilde{e}_2 and the orders of accuracy. Uniform mesh.

mesh	P^1		P^2		P^3	
	L^2 error	order	L^2 error	order	L^2 error	order
10	4.08E-05	–	8.15E-08	–	1.04E-10	–
20	2.47E-06	4.05	1.27E-09	6.01	4.08E-13	8.00
30	4.85E-07	4.01	1.11E-10	6.00	1.59E-14	8.00
40	1.53E-07	4.01	1.98E-11	6.00	1.60E-15	8.00
50	6.26E-08	4.00	5.18E-12	6.00	2.68E-16	8.00

4.2. EIGEN-STRUCTURES OF G : ERROR ESTIMATE

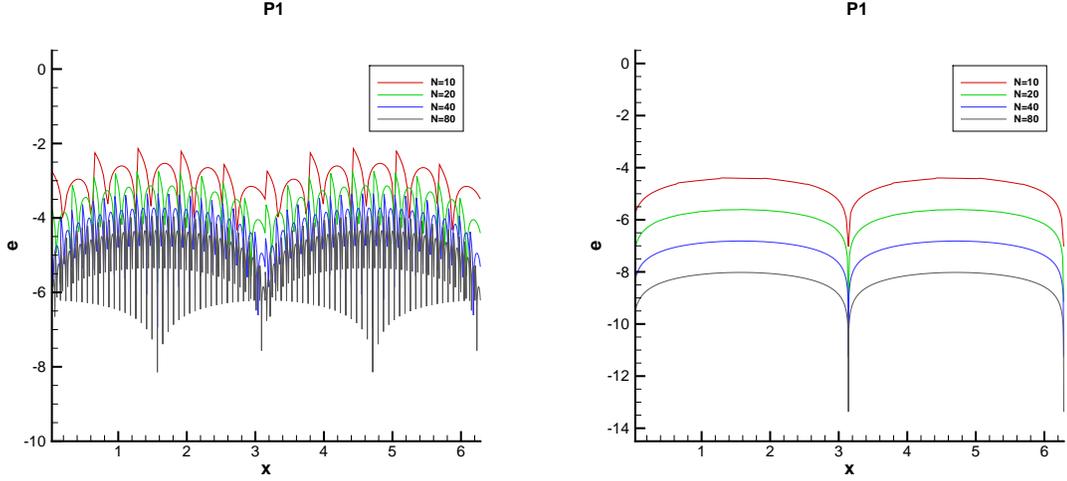


Figure 4.11: LDG with P^1 ; diffusive equation; regular error $|u(T = 2) - u_h(T = 2)|$ (left); $|\tilde{e}_1|$ (right)

of (4.50) is

$$u(t, x, y) = \sin(x + y - 2t).$$

Firstly, we use DG with the Q^k space as the approximation space, see equation (4.16).

We compute the error

$$\bar{e}_1 = u_h(T = \pi) - u_h(T = 2\pi). \quad (4.51)$$

In Table 4.11, we report the L^2 norm of \bar{e}_1 and the orders of accuracy. $(2k + 1)^{th}$ order of accuracy is observed. We choose CFL= 0.1. Then, we use DG with the P^k space, see equation (4.21). We give the L^2 norm of \bar{e}_1 and the orders of accuracy in Table 4.12. Again $(2k + 1)^{th}$ order of accuracy is observed. The magnitude of error appears to be larger than those from DG scheme with a Q^k space. In Figure 4.14, the DG error in 3-D and contour plot are reported.

4.2. EIGEN-STRUCTURES OF G : ERROR ESTIMATE

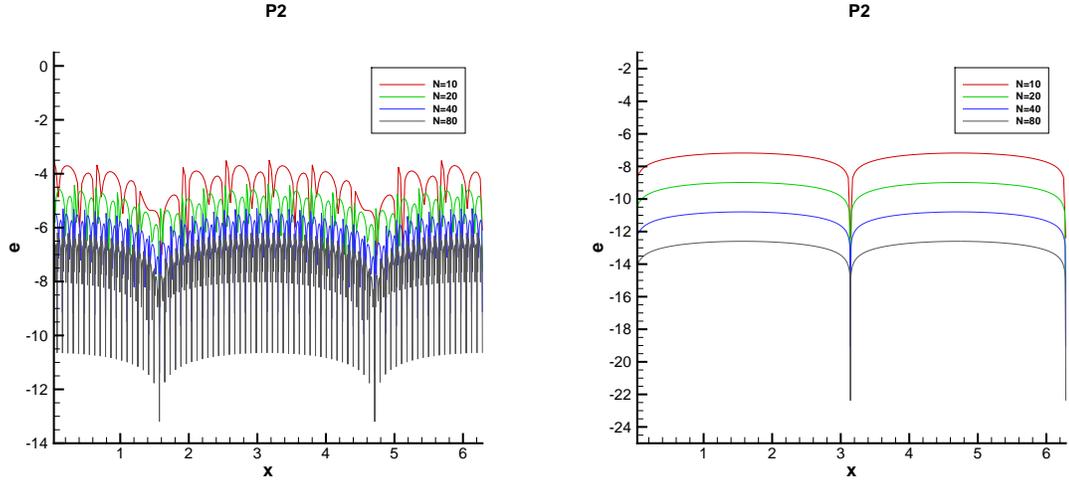


Figure 4.12: LDG with P^2 ; diffusive equation; regular error $|u(T = 2) - u_h(T = 2)|$ (left); $|\tilde{e}_1|$ (right)

Example 4.36. We consider the solid body rotation problem:

$$u_t - yu_x + xu_y = 0, \quad (x, y) \in [-\pi, \pi] \times [-\pi, \pi], \quad (4.52)$$

Table 4.11: 2-D advection equation $u_t + u_x + u_y = 0$. Q^k is used. The L^2 norm of \bar{e}_1 and the orders of accuracy. Uniform mesh.

mesh	Q^1		Q^2		Q^3	
	L^2 error	order	L^2 error	order	L^2 error	order
30×30	3.55E-03	–	1.56E-06	–	4.62E-10	–
40×40	1.50E-03	2.99	3.70E-07	5.00	4.75E-11	7.90
50×50	7.68E-04	3.00	1.21E-07	5.00	9.79E-12	7.08
60×60	4.45E-04	3.00	4.88E-08	5.00	2.73E-12	7.00
70×70	2.80E-04	3.00	2.26E-08	5.00	9.28E-13	7.01

4.2. EIGEN-STRUCTURES OF G : ERROR ESTIMATE

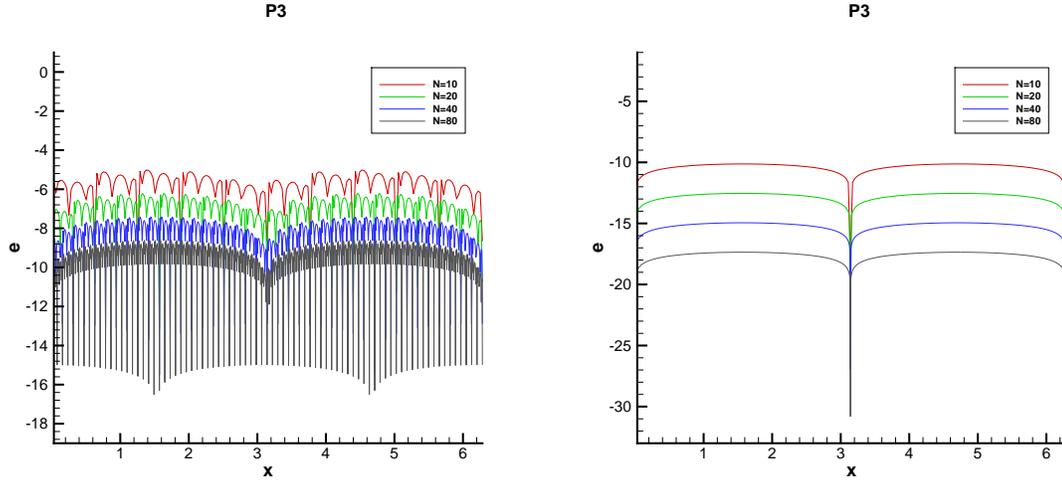


Figure 4.13: LDG with P^3 ; diffusive equation; regular error $|u(T=2) - u_h(T=2)|$ (left); $|\tilde{e}_1|$ (right)

with the following smooth initial condition:

$$u(0, x, y) = \begin{cases} \cos^{12}(r) & r < \frac{\pi}{2}, \\ 0 & \text{otherwise,} \end{cases} \quad (4.53)$$

where $r = \sqrt{(x - \pi/2)^2 + y^2}$. Recall that we use the hybrid scheme to solve the solid

Table 4.12: 2-D advection equation $u_t + u_x + u_y = 0$. P^k is used. The L^2 norm of \bar{e}_1 and the orders of accuracy. Uniform mesh.

mesh	P^1		P^2		P^3	
	L^2 error	order	L^2 error	order	L^2 error	order
30×30	1.40E-02	—	2.22E-05	—	1.71E-08	—
40×40	5.96E-03	2.97	5.29E-06	4.99	2.28E-09	7.01
50×50	3.06E-03	2.98	1.74E-06	4.99	4.77E-10	7.00
60×60	1.77E-03	2.99	6.98E-07	5.00	1.33E-10	7.00
70×70	1.12E-03	2.99	3.23E-07	5.00	4.52E-11	7.01

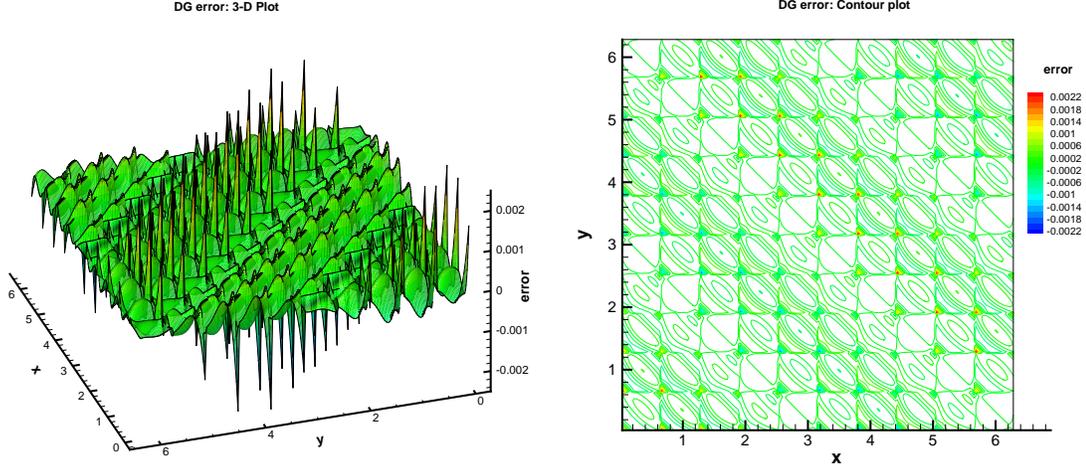


Figure 4.14: DG with P^3 ; 2-D linear advection equation $u_t + u_x + u_y = 0$; $N_x \times N_y = 10 \times 10$; 3-D plot of error $e = u(T = 4\pi) - u_h(T = 4\pi)$ (left); Contour of error $e = u(T = 4\pi) - u_h(T = 4\pi)$ (right)

body rotation problem in Example 3.7. As before, we calculate the error

$$\bar{e}_1 = u_h(T = 2\pi) - u_h(T = 4\pi).$$

In the simulation, we adopt the approximation space P^k for spatial discretization and SSPRK(5,4) for temporal discretization. We choose CFL=0.1. In Table 4.13, we report L^2 norm of \bar{e}_1 and the orders of accuracy. $(2k+1)^{th}$ order is observed. Note that the mesh has to be fine enough to resolve the ‘cosine bell’ in order to observe a clean order of accuracy. Also note that the ‘cosine bell’ is centered at $(\pi/2, 0)$, which is away from the origin. Around the origin, the propagation speed $(-y, x)$ is close to zero. In such situation, the second term on the r.h.s. of Corollary 4.14 might dominate and $(2k+1)^{th}$ order can’t be observed. We omit to present the results for brevity. Again, we remark that the long time behavior of the error as commented in

4.2. EIGEN-STRUCTURES OF G : ERROR ESTIMATE

Remark 4.8 still holds if the ‘cosine bell’ is positioned around the origin.

Table 4.13: Solid body rotation: $u_t - yu_x + xu_y = 0$. P^k is used. The L^2 norm of \bar{e}_1 and the orders of accuracy. Uniform mesh.

	P^1		P^2		P^3	
mesh	L^2 error	order	L^2 error	order	L^2 error	order
100×100	9.98E-03	–	3.77E-05	–	9.33E-08	–
120×120	6.22E-03	2.59	1.53E-05	4.95	2.62E-08	6.97
140×140	4.09E-03	2.72	7.11E-06	4.97	8.94E-09	6.97
160×160	2.81E-03	2.81	3.66E-06	4.98	3.52E-09	6.98
180×180	2.01E-03	2.86	2.04E-06	4.98	1.55E-09	6.99
200×200	1.48E-03	2.89	1.20E-06	4.99	7.40E-10	6.99

Example 4.37. We consider the following 2-D linear system:

$$\left\{ \begin{array}{l} \left(\begin{array}{c} u \\ v \end{array} \right)_t + \left(\begin{array}{cc} -1 & 0 \\ 0 & 1 \end{array} \right) \left(\begin{array}{c} u \\ v \end{array} \right)_x + \left(\begin{array}{cc} 0 & -1 \\ -1 & 0 \end{array} \right) \left(\begin{array}{c} u \\ v \end{array} \right)_y = \left(\begin{array}{c} 0 \\ 0 \end{array} \right), \\ u(0, x, y) = \frac{1}{2\sqrt{2}} \sin(x + y) - \frac{1}{2\sqrt{2}} \cos(x + y), \\ v(0, x, y) = \frac{\sqrt{2}-1}{2\sqrt{2}} \sin(x + y) + \frac{\sqrt{2}+1}{2\sqrt{2}} \cos(x + y), \end{array} \right. \quad (4.54)$$

with periodic boundary conditions in both x - and y - directions. Note that this is the second order wave equation written as a first order linear hyperbolic system and the exact solution is

$$\left\{ \begin{array}{l} u(t, x, y) = \frac{1}{2\sqrt{2}} \sin(x + y + \sqrt{2}t) - \frac{1}{2\sqrt{2}} \cos(x + y - \sqrt{2}t), \\ v(t, x, y) = \frac{\sqrt{2}-1}{2\sqrt{2}} \sin(x + y + \sqrt{2}t) + \frac{\sqrt{2}+1}{2\sqrt{2}} \cos(x + y - \sqrt{2}t). \end{array} \right. \quad (4.55)$$

4.2. EIGEN-STRUCTURES OF G : ERROR ESTIMATE

We remark that the two matrices in equation (4.54) don't commute, therefore the linear system can't be reduced to 2-D scalar problems. In the simulation, the upwind flux and the SSPRK(9,9) scheme is used. We choose CFL=0.1. Note that the period of the solution in time is $\sqrt{2}\pi$, that is $u(t, x, y) = u(t + \sqrt{2}\pi, x, y)$, then we let

$$\bar{e}_1 = u_h(T = 2) - u_h(T = 2 + \sqrt{2}\pi). \quad (4.56)$$

In Table 4.14, we report L^2 norm of \bar{e}_1 and the orders of accuracy. $(2k + 1)^{th}$ order is observed. The error about the v variable is not presented for brevity since it gives almost the same results.

Table 4.14: Two-dimensional linear system (4.54). P^k is used. The L^2 norm of \bar{e}_1 and the orders of accuracy. Uniform mesh.

	P^1		P^2		P^3	
mesh	L^2 error	order	L^2 error	order	L^2 error	order
30×30	1.26E-02	–	1.40E-05	–	2.09E-08	–
40×40	5.34E-03	2.97	3.33E-06	4.99	2.51E-09	7.37
50×50	2.74E-03	2.99	1.09E-06	4.99	4.31E-10	7.91
60×60	1.59E-03	2.99	4.40E-07	4.99	1.14E-10	7.27
70×70	1.00E-03	3.00	2.04E-07	5.00	3.80E-11	7.14

Example 4.38. We consider the 2-D Maxwell equations:

$$\begin{cases} \frac{\partial H_x}{\partial t} + \frac{\partial E_z}{\partial y} = 0, \\ \frac{\partial H_y}{\partial t} - \frac{\partial E_z}{\partial x} = 0, \\ \frac{\partial E_z}{\partial t} - \frac{\partial H_y}{\partial x} + \frac{\partial H_x}{\partial y} = 0. \end{cases} \quad (4.57)$$

4.2. EIGEN-STRUCTURES OF G : ERROR ESTIMATE

Note that the Maxwell equations are linear hyperbolic systems, which can be written as

$$U_t + AU_x + BU_y = 0, \quad (4.58)$$

where

$$A = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & -1 & 0 \end{pmatrix}, \quad B = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix}. \quad (4.59)$$

For any unit vector $\mathbf{n} = (n_1, n_2)$, let

$$D = n_1A + n_2B. \quad (4.60)$$

Note that D always has three eigenvalues 1, -1 and 0 given a unit vector \mathbf{n} . It is easy to check that

$$\begin{cases} H_x(t, x, y) &= -\beta \cos(\alpha x + \beta y + t), \\ H_y(t, x, y) &= \alpha \cos(\alpha x + \beta y + t), \\ E_z(t, x, y) &= \cos(\alpha x + \beta y + t), \end{cases} \quad (4.61)$$

is an exact solution of Maxwell equations for $(x, y) \in [0, 2\pi/\alpha] \times [0, 2\pi/\beta]$, where $\alpha^2 + \beta^2 = 1$. This example is motivated by the work in [25], where the DG error after applying a post-processing technique are enhanced from $(k+1)^{th}$ order to $(2k+1)^{th}$ order. In the simulation, we take $\alpha = \beta = \frac{\sqrt{2}}{2}$. The upwind flux and the SSPRK(9,9)

scheme is used. We choose CFL= 0.1. Let

$$\begin{aligned}\bar{e}_1 &= (E_z)_h(T = 4\pi) - (E_z)_h(T = 2\pi), \\ \bar{e}_2 &= (H_x)_h(T = 4\pi) - (H_x)_h(T = 2\pi), \\ \bar{e}_3 &= (H_y)_h(T = 4\pi) - (H_y)_h(T = 2\pi).\end{aligned}$$

In Table 4.15, we report L^2 norm of \bar{e}_1 and the orders of accuracy. The $(2k+1)^{th}$ order of accuracy is numerically observed for the P^1 and P^2 cases, but not for the P^3 case. In Table 4.16, we report L^2 norm of \bar{e}_2 and the orders of accuracy. $(2k+1)^{th}$ order of accuracy is not observed for the P^2 and P^3 cases. The reason we suspect is that there is one zero eigenvalue in D , along which the non-physically relevant eigenvectors are not damped. In Figure 4.15, contours of \bar{e}_2 and \bar{e}_3 are plotted, for the DG with P^3 space. It is clear that \bar{e}_2 oscillates only in the x -direction and \bar{e}_3 oscillates only in the y -direction. Such observation suggests that the non-physically relevant eigenvectors of $(H_x)_h$ (or $(H_y)_h$) do not damp in x - (or y -) direction properly. We remark that although $(2k+1)^{th}$ order can't be observed for \bar{e}_2 and \bar{e}_3 . The DG solution can still be post-processed to attain $(2k+1)^{th}$ order convergence as in [25]. The long time behavior of the magnitude of errors as commented in Remark 4.8 might still hold.

4.2. EIGEN-STRUCTURES OF G : ERROR ESTIMATE

Table 4.15: Two-dimensional Maxwell equations. P^k is used. The L^2 norm and orders of accuracy of \bar{e}_1 of E_z . Uniform mesh.

mesh	P^1		P^2		P^3	
	L^2 error	order	L^2 error	order	L^2 error	order
20×20	6.82E-02	–	2.49E-04	–	5.49E-07	–
30×30	2.09E-02	2.92	3.34E-05	4.96	4.92E-08	5.95
40×40	8.91E-03	2.96	7.97E-06	4.98	9.65E-09	5.66
50×50	4.58E-03	2.98	2.62E-06	4.98	2.75E-09	5.62
60×60	2.66E-03	2.99	1.06E-06	4.99	9.84E-10	5.65

Table 4.16: Two-dimensional Maxwell equations. P^k is used. The L^2 norm and orders of accuracy of \bar{e}_2 of H_x . Uniform mesh.

mesh	P^1		P^2		P^3	
	L^2 error	order	L^2 error	order	L^2 error	order
20×20	4.88E-02	–	4.00E-04	–	1.02E-05	–
30×30	1.51E-02	2.89	1.08E-04	3.23	1.97E-06	4.06
40×40	6.54E-03	2.92	4.12E-05	3.35	5.69E-07	4.32
50×50	3.40E-03	2.93	1.90E-05	3.47	2.10E-07	4.47
60×60	1.99E-03	2.94	9.93E-06	3.56	9.12E-08	4.57

4.2. EIGEN-STRUCTURES OF G : ERROR ESTIMATE

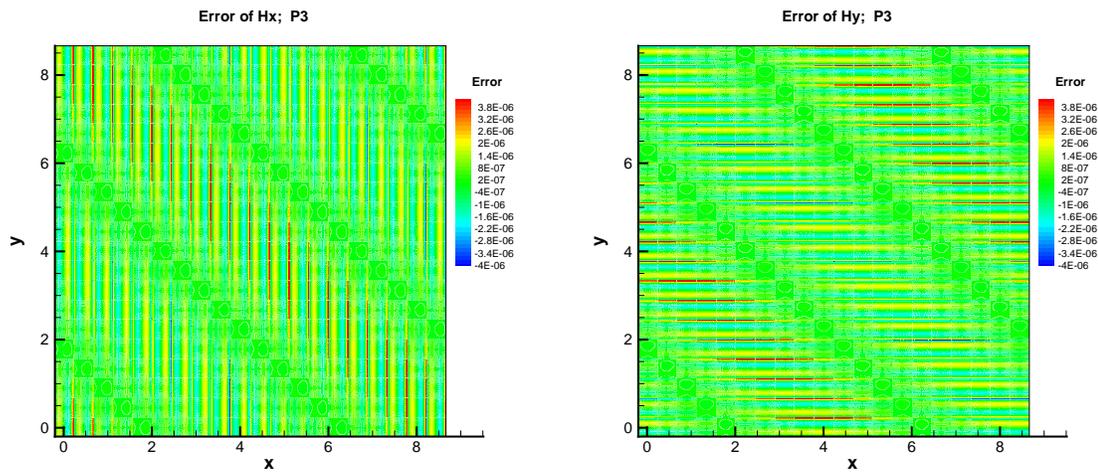


Figure 4.15: DG with P^3 for Maxwell equations; Contour of error $\bar{e}_2 = (H_x)_h(T = 4\pi) - (H_x)_h(T = 2\pi)$ (left); Contour of error $\bar{e}_3 = (H_y)_h(T = 4\pi) - (H_y)_h(T = 2\pi)$ (right)

4.3 A New LWDG Scheme with Superconvergence

As an alternative of using high order SSP RK methods to discretize the time variable, the one-step one-stage high order LW time discretization [63] attracts lots of attentions due to its compactness and low-storage requirement [88, 89]. The LW procedure is known as the Cauchy-Kowalewski type time discretization in the literature, which relies on converting each time derivative in a truncated temporal Taylor expansion (with expected accuracy) of the solution into spatial derivatives by repeatedly using the underlying differential equation and its differentiated forms. The original LWDG scheme is proposed by Qiu et al. in [88] for hyperbolic conservation laws and extended to Hamilton-Jacobi equations in [45]. In this section, we would like to investigate superconvergence properties of the LWDG scheme. The numerical results documented in [27, 15] indicates that the RKDG scheme exhibits superconvergence properties if the spatial error dominates, whereas the results reported below show that such superconvergence can not be observed numerically for the original LWDG schemes in [88].

In order to restore superconvergence in the LWDG framework, we modify the original scheme by borrowing techniques from the LDG scheme. In particular, the high order spatial derivatives in the LW procedure are reconstructed through a LDG formulation, while they are obtained by direct differentiation of the numerical solution for the original LWDG scheme. In addition, for the newly proposed LWDG scheme, flux terms appeared in the LDG framework provide extra freedom to design

appropriate numerical fluxes for numerical stability. Numerical experiments demonstrate that the modified LWDG scheme is stable and high order accurate. Moreover, it is observed that the proposed scheme displays similar superconvergence properties as a RKDG scheme. In order to theoretically reveal superconvergence of the new LWDG scheme, we follow the analysis in Section 4.1 to study the eigen-structures of the amplification matrix in the Fourier analysis framework. In [116], the dispersion and dissipation error analysis for the original LWDG scheme is provided.

4.3.1 Lax-Wendroff Type Time Discretization

The LW time discretization, as an alternative to the method-of-line RK time discretization, was proposed for the DG scheme in [88]. We consider the 1-D hyperbolic conservation law (2.1). The formulation of LW type time discretization starts with a Taylor expansion in time for the solution u . For example, for third order temporal discretization, we have

$$u(x, t + \Delta t) \approx u(x, t) + \Delta t u_t(x, t) + \frac{\Delta t^2}{2} u_{tt}(x, t) + \frac{\Delta t^3}{6} u_{ttt}(x, t). \quad (4.62)$$

The temporal derivatives in (4.62) are then converted into spatial derivatives via the original differential equation (2.1) and its differentiated versions:

$$u_t = -f(u)_x, \quad (4.63)$$

$$u_{tt} = (f'(u)f(u)_x)_x, \quad (4.64)$$

$$u_{ttt} = -\left(f''(u)(f(u)_x)^2 + f'(u)(f'(u)f(u)_x)_x\right)_x. \quad (4.65)$$

Substituting (4.63)-(4.65) into (4.62) gives

$$u(t + \Delta t, x) \approx u(t, x) - \Delta t \left(f(u) - \frac{\Delta t}{2} f'(u) f(u)_x + \frac{\Delta t^2}{6} (f''(u) (f(u)_x)^2 + f'(u) (f'(u) f(u)_x)_x) \right)_x \quad (4.66)$$

$$= u(t, x) - \Delta t F(\Delta t, u)_x, \quad (4.67)$$

where

$$F = f(u) - \frac{\Delta t}{2} f'(u) f(u)_x + \frac{\Delta t^2}{6} (f''(u) (f(u)_x)^2 + f'(u) (f'(u) f(u)_x)_x) \quad (4.68)$$

is a new flux function. The LWDG formulation is defined based on (4.66) as follows:

given $u_h^n \in V_h^k$, find $u_h^{n+1} \in V_h^k$ such that $\forall v \in V_h^k$, we have

$$\int_{I_j} u_h^{n+1} v \, dx = \int_{I_j} u_h^n v \, dx + \Delta t_n \left(\int_{I_j} F(u_h^n, \Delta t_n) v_x \, dx - \hat{F}_{j+\frac{1}{2}} v_{j+\frac{1}{2}}^- + \hat{F}_{j-\frac{1}{2}} v_{j-\frac{1}{2}}^+ \right), \quad (4.69)$$

for $j = 1, \dots, N$ and $n = 0, 1, \dots$. Here u_h^n approximates the solution of (2.1) at time t^n , and $\Delta t_n = t^{n+1} - t^n$. The numerical flux $\hat{F}_{j+\frac{1}{2}}$ is defined as

$$\hat{F}_{j+\frac{1}{2}} = \frac{1}{2} \left(\left(F((u_h^n)_{j+\frac{1}{2}}^+, \Delta t_n) + F((u_h^n)_{j+\frac{1}{2}}^-, \Delta t_n) \right) - \alpha \left((u_h^n)_{j+\frac{1}{2}}^+ - (u_h^n)_{j+\frac{1}{2}}^- \right) \right), \quad (4.70)$$

where $\alpha = \max_u |f'(u)|$ with the maximum taken in the computational domain $[a, b]$.

Here $f(u_h^n)_x$ and $(f'(u_h^n) f(u_h^n)_x)_x$ in F (4.68) are obtained by direct differentiation of

the numerical solution, i.e.,

$$f(u_h^n)_x = f'(u_h^n)(u_h^n)_x, \quad (f'(u_h^n)f(u_h^n)_x)_x = f'(u_h^n)(2f''(u_h^n)(u_h^n)_x^2 + f'(u_h^n)(u_h^n)_{xx}).$$

Note that the numerical flux $\hat{F}_{j+\frac{1}{2}}$ is very similar to the Lax-Friedrichs flux given in Section 2.1.

4.3.2 Numerical Evidences Regarding Superconvergence

In this subsection, we assess the numerical performance of the corresponding properties for the LWDG schemes (4.69) proposed in [88].

Enhanced Accuracy by Post-Processing

It has been proved in [27] that a semi-discrete DG solution for a linear hyperbolic equation is superconvergent in the negative-order norms with order of $2k+1$ assuming that the solution is globally smooth. Because of this, the order convergence can be enhanced from $(k+1)^{th}$ order to $(2k+1)^{th}$ order by applying a post-processor [10], when the mesh is translation invariant. Specifically, such better approximated solution u_h^* is obtained by convolving the numerical solution with a convolution kernel K_h that is a linear combination of B-splines [10, 27],

$$u_h^* = K_h \star u_h. \tag{4.71}$$

Along this line, a similar estimation of the LDG scheme for solving linear convection-diffusion equations was given in [57], and of the DG scheme for solving nonlinear

scalar hyperbolic conservation laws was provided in [58]. The numerical results reported in [27] show that the $(2k + 1)^{th}$ order of convergence is achieved, when a high order SSP RK scheme is used for time discretization. Note that the time step Δt is chosen small enough such that the spatial error dominates.

Now, we study the convergence property of the post-processed solutions for the LWDG scheme (4.69) with numerical flux specified in equation (4.70) for the following linear advection problem:

$$\begin{aligned} u_t + u_x &= 0, \quad x \in [0, 2\pi], \\ u(t = 0, x) &= \sin x, \end{aligned} \tag{4.72}$$

with periodic boundary conditions. In particular, we use the LWDG scheme (4.69) to solve the model problem and post-process the numerical solution at final time step. The numerical mesh is set to be uniform. The numerical solution is computed up to time $T=1$ and we let $\Delta t = CFL\Delta x$ with $CFL=0.01$, thereby making the the spatial error dominant. In Table 4.17, we report the L^2 and L^∞ errors and the orders of accuracy, before and after applying the post-processing procedure. It is observed that the magnitude of numerical errors is reduced by applying the post-processor, however the order of accuracy remains $k + 1$.

Long Time Behavior of Errors

It is discovered that the DG errors do not significantly grow over a very long time period [15]. Such behavior of errors could be explained by the superconvergence of the DG solution in approximating the Radau projection of the exact solution [15, 17, 117].

4.3. A NEW LWDG SCHEME WITH SUPERCONVERGENCE

Table 4.17: Linear advection. LWDG scheme in [88]. T=1. CFL=0.01.

	mesh	Before post-processing				After post-processing			
		L^2 error	order	L^∞ error	order	L^2 error	order	L^∞ error	order
P^1	20	4.21E-03	–	1.34E-02	–	3.77E-04	–	5.37E-04	–
	40	1.06E-03	1.99	3.45E-03	1.96	4.49E-05	3.07	6.27E-05	3.10
	60	4.72E-04	2.00	1.54E-03	1.98	1.38E-05	2.91	1.92E-05	2.92
	80	2.65E-04	2.00	8.71E-04	1.99	6.24E-06	2.76	8.63E-06	2.77
	100	1.70E-04	2.00	5.58E-04	1.99	3.47E-06	2.62	4.79E-06	2.64
P^2	20	1.02E-04	–	3.50E-04	–	4.25E-06	–	6.03E-06	–
	40	1.28E-05	3.00	4.41E-05	2.99	2.67E-07	3.99	3.78E-07	4.00
	60	3.79E-06	3.00	1.31E-05	3.00	7.06E-08	3.28	9.99E-08	3.28
	80	1.60E-06	3.00	5.51E-06	3.00	2.89E-08	3.11	4.09E-08	3.11
	100	8.18E-07	3.00	2.82E-06	3.00	1.46E-08	3.05	2.07E-08	3.05
P^3	20	2.12E-06	–	4.81E-06	–	8.76E-08	–	1.20E-07	–
	40	1.32E-07	4.00	3.48E-07	3.82	2.74E-09	5.00	3.52E-09	5.09
	60	2.59E-08	4.02	6.74E-08	4.02	5.36E-10	4.02	6.88E-10	4.03
	80	8.22E-09	3.99	2.13E-08	4.02	1.70E-10	4.00	2.18E-10	4.00
	100	3.37E-09	4.00	8.75E-09	3.99	6.96E-11	3.99	8.93E-11	3.99

In particular, numerical analysis for solving the linear advection equation (4.72) in [117] shows that the Radau projection of the exact solution $\mathbf{P}u$ satisfies

$$\|\mathbf{P}u - u_h\| \leq C_1 \Delta x^{k+2}.$$

It leads to the following error estimation:

$$\|e\| \leq C_0 \Delta x^{k+1} + \text{TC}_1 \Delta x^{k+2}, \quad (4.73)$$

where C_0 and C_1 are positive constant. Note that the first term, which is independent of time, will dominate the error until $T = \mathcal{O}(1/\Delta x)$. In Section 4.1, an optimal error estimation was obtained via Fourier analysis. If the mesh is assumed to be uniform and the boundary condition is periodic, the numerical error by a DG scheme (when

4.3. A NEW LW DG SCHEME WITH SUPERCONVERGENCE

$k = 1, 2,$ and 3) for the linear advection equation (4.76) can be decomposed into three parts:

$$\|e\| \leq C_0 \Delta x^{k+1} + TC_1 \Delta x^{2k+1} + C_2 \exp\left(-\frac{CT}{\Delta x}\right) \Delta x^{k+1}, \quad (4.74)$$

where $C_0, C_1, C_2,$ and C are positive constant. The claim indicates that the error does not significantly grow until $T = \mathcal{O}(1/\Delta x^k)$. See Proposition 4.5.

We use the model problem (4.76) to compare the LW DG scheme with the RK DG scheme in the context of the long time behavior of numerical errors. We use P^2 as the approximation space, and adopt the third order LW procedure for the LW DG scheme and the SSPRK(3,3) method for the RK DG scheme. Note that both schemes are third order accurate in space and time. Let $N = 50$ and CFL=0.01. We compute the numerical solutions using the two types of DG schemes up to time $T = 500$, and plot the time evolution history of the L^2 errors in Figure 4.16. It is observed that the numerical error by the RK DG scheme does not significantly grow for a long time simulation. In fact, the magnitude of error at $T = 500$ is comparable to that at the very beginning of the simulation. Contrarily, the error by the LW DG scheme begins to noticeably grow around $T=20$, and the growth rate is observed to be linear after some time.

In Section 4.1, the following error is defined and studied for the DG scheme:

$$\bar{e} = u_h(t = 2\pi) - u_h(t = 4\pi). \quad (4.75)$$

It is also shown in Section 4.1 that the error \bar{e} by a RK DG scheme (when $k = 1, 2, 3$)

4.3. A NEW LWDG SCHEME WITH SUPERCONVERGENCE

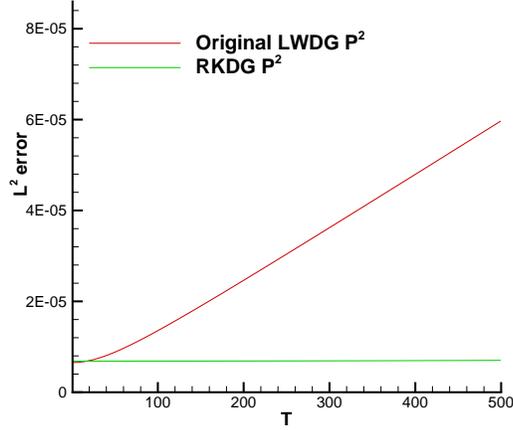


Figure 4.16: Long time behavior of errors. 1-D linear advection. RKDG P^2 scheme versus new LWDG P^2 scheme. CFL=0.01.

is in the order of $2k + 1$ for spatial accuracy. Here, we check the error \bar{e} for the LWDG scheme when solving the model equation (4.72). In Table 4.18, we report the L^2 norms of error \bar{e} and orders of accuracy. Only $(k + 1)^{th}$ order of accuracy is observed for the LWDG scheme.

Table 4.18: Linear advection. Original LWDG scheme in [88]. The L^2 norms of error \bar{e} and the order of accuracy. CFL=0.01.

mesh	P^1		P^2		P^3	
	L^2 error	order	L^2 error	order	L^2 error	order
20	1.94E-03	–	1.32E-05	–	2.74E-07	–
40	2.57E-04	2.91	1.48E-06	3.16	1.71E-08	4.00
60	8.24E-05	2.81	4.29E-07	3.05	3.39E-09	4.00
80	3.80E-05	2.69	1.79E-07	3.03	1.07E-09	3.99
100	2.14E-05	2.57	9.15E-08	3.02	4.36E-10	4.05

In summary, several superconvergence properties of a semi-discrete DG scheme,

including accuracy enhancement by post-processing and long time behaviors of errors, are maintained by the a discrete RKDG scheme, whereas they are not numerically observed when the original LW temporal discretization [88] is used. In the next subsection, we propose a new LWDG scheme, which aims to restore superconvergence properties mentioned above.

4.3.3 A New LWDG Scheme

In the subsection, we formulate a new LWDG scheme for solving linear and nonlinear hyperbolic conservation laws. Some stability issues will be discussed.

Linear Advection Equations

Firstly, we consider the linear advection equation

$$u_t + u_x = 0. \quad (4.76)$$

As the original LWDG scheme, we start with the Taylor expansion of the solution in time as in equation (4.62) for achieving third order temporal accuracy. By repeatedly using the underlying differential equation (4.76) and its differentiation versions, we obtain

$$u(t + \Delta t) \approx u(t) - \Delta t u_x + \frac{\Delta t^2}{2} u_{xx} - \frac{\Delta t^3}{6} u_{xxx}. \quad (4.77)$$

Note that the original LWDG formulation uses direct differentiation of the solution to obtain high order spatial derivatives. However, in the new formulation, we propose

4.3. A NEW LWDG SCHEME WITH SUPERCONVERGENCE

to use the LDG techniques to reconstruct high order derivatives instead. To illustrate the idea of using an LW time discretization procedure in the LDG framework, we consider solving the following PDE by an LDG scheme:

$$u_t = \epsilon_1 u_x + \epsilon_2 u_{xx} + \epsilon_3 u_{xxx}, \quad (4.78)$$

where $\epsilon_1 < 0$, $\epsilon_2 > 0$, and $\epsilon_3 < 0$ are constant. Two auxiliary variables are introduced,

$$p = u_x, \quad q = p_x.$$

The corresponding semi-discrete LDG scheme is to find $u_h, p_h, q_h \in V_h^k$ such that $\forall v, w, s \in V_h^k$, we have

$$\begin{aligned} \frac{d}{dt} \int_{I_j} u_h v \, dx &= - \int_{I_j} (\epsilon_1 u_h + \epsilon_2 p_h + \epsilon_3 q_h) v_x \, dx + (\epsilon_1 (\hat{u}_h)_{j+\frac{1}{2}} + \epsilon_2 (\hat{p}_h)_{j+\frac{1}{2}} \\ &\quad + \epsilon_3 (\hat{q}_h)_{j+\frac{1}{2}}) v_{j+\frac{1}{2}}^- - (\epsilon_1 (\hat{u}_h)_{j-\frac{1}{2}} + \epsilon_2 (\hat{p}_h)_{j-\frac{1}{2}} + \epsilon_3 (\hat{q}_h)_{j-\frac{1}{2}}) v_{j-\frac{1}{2}}^+, \\ \int_{I_j} p_h w \, dx &= - \int_{I_j} u_h w_x \, dx + (\tilde{u}_h)_{j+\frac{1}{2}} w_{j+\frac{1}{2}}^- - (\tilde{u}_h)_{j-\frac{1}{2}} w_{j-\frac{1}{2}}^+, \\ \int_{I_j} q_h s \, dx &= - \int_{I_j} p_h s_x \, dx + (\tilde{p}_h)_{j+\frac{1}{2}} s_{j+\frac{1}{2}}^- - (\tilde{p}_h)_{j-\frac{1}{2}} s_{j-\frac{1}{2}}^+, \end{aligned} \quad (4.79)$$

for $j = 1, \dots, N$. Here u_h, p_h and q_h approximate the solution, and its first and second order spatial derivatives of (4.78), respectively. $\hat{u}_h, \tilde{u}_h, \hat{p}_h, \tilde{p}_h$, and \hat{q}_h are numerical fluxes chosen according to stability consideration as in [30, 115]:

$$\hat{u}_h = u_h^-, \quad \hat{p}_h = p_h^+, \quad \tilde{u}_h = u_h^-, \quad \hat{q}_h = q_h, \quad \tilde{p}_h = p_h \quad (4.80)$$

Similar to the semi-discrete LDG scheme for solving the convection-diffusion equations [30] and KdV-type equations [115], we have the following proposition:

Proposition 4.39. *The semi-discrete LDG scheme (4.79) equipped with the alternating numerical fluxes (4.80) is L^2 stable.*

Proof. The proof is similar to that in [115], thus omitted for brevity.

Based on the semi-discrete LDG scheme (4.79) for equation (4.78), we define the new LWDG formulation as follows: given $u_h^n \in V_h^k$, find p_h^n , q_h^n , and $u_h^{n+1} \in V_h^k$ such that $\forall v$, w , and $s \in V_h^k$, we have

$$\begin{aligned} \int_{I_j} u_h^{n+1} v dx &= \int_{I_j} u_h^n v_x dx + \Delta t \left(\int_{I_j} (u_h^n - \frac{\Delta t}{2} p_h^n + \frac{\Delta t^2}{6} q_h^n) v_x dx \right. \\ &\quad - ((\hat{u}_h^n)_{j+\frac{1}{2}} - \frac{\Delta t}{2} (\hat{p}_h^n)_{j+\frac{1}{2}} + \frac{\Delta t^2}{6} (\hat{q}_h^n)_{j+\frac{1}{2}}) v_{j+\frac{1}{2}}^- \\ &\quad \left. + ((\hat{u}_h^n)_{j-\frac{1}{2}} - \frac{\Delta t}{2} (\hat{p}_h^n)_{j-\frac{1}{2}} + \frac{\Delta t^2}{6} (\hat{q}_h^n)_{j-\frac{1}{2}}) v_{j-\frac{1}{2}}^+ \right), \end{aligned} \quad (4.81)$$

$$\int_{I_j} p_h^n w dx = - \int_{I_j} u_h^n w_x dx + (\tilde{u}_h^n)_{j+\frac{1}{2}} w_{j+\frac{1}{2}}^- - (\tilde{u}_h^n)_{j-\frac{1}{2}} w_{j-\frac{1}{2}}^+, \quad (4.82)$$

$$\int_{I_j} q_h^n s dx = - \int_{I_j} p_h^n s_x dx + (\tilde{p}_h^n)_{j+\frac{1}{2}} s_{j+\frac{1}{2}}^- - (\tilde{p}_h^n)_{j-\frac{1}{2}} s_{j-\frac{1}{2}}^+, \quad (4.83)$$

for $j = 1, \dots, N$ and $n = 0, 1, \dots$. Here u_h^n , p_h^n , and q_h^n approximate the solution, and its first and second order spatial derivatives of (4.76) at time t^n , respectively. \hat{u}_h^n , \tilde{u}_h^n , \hat{p}_h^n , \tilde{p}_h^n , and \hat{q}_h^n are the numerical fluxes chosen as (4.80). Note that the proposed LWDG scheme (4.81) is formulated by applying the forward Euler time discretization to the semi-discrete LDG scheme (4.79) with $\epsilon_1 = -\Delta t$, $\epsilon_2 = \frac{\Delta t^2}{2}$, and $\epsilon_3 = -\frac{\Delta t^3}{6}$.

Nonlinear Scaler Conservation Laws

Now, we consider a 1-D nonlinear scalar hyperbolic conservation law (2.1). Similar to the linear case, we start with the Taylor expansion of the solution, i.e., equation (4.62). Again, in order to formulate a new LWDG scheme, we first consider the following PDE:

$$u_t = \epsilon_1 f(u)_x + \epsilon_2 (f'(u)f(u)_x)_x, \quad (4.84)$$

where $\epsilon_1 < 0$ and $\epsilon_2 > 0$ are constant. By introducing the auxiliary variable

$$p = f(u)_x,$$

we define the corresponding semi-discrete LDG scheme for equation (4.84): find $u_h, p_h \in V_h^k$ such that $\forall v, w \in V_h^k$, we have

$$\begin{aligned} \frac{d}{dt} \int_{I_j} u_h v dx &= - \int_{I_j} (\epsilon_1 f(u_h) + \epsilon_2 f'(u_h) p_h) v_x dx + (\epsilon_1 \hat{f}_{j+\frac{1}{2}} + \epsilon_2 \widehat{f'p}_{j+\frac{1}{2}}) v_{j+\frac{1}{2}}^- \\ &\quad - (\epsilon_1 \hat{f}_{j-\frac{1}{2}} + \epsilon_2 \widehat{f'p}_{j-\frac{1}{2}}) v_{j-\frac{1}{2}}^+, \\ \int_{I_j} p_h w dx &= - \int_{I_j} f(u_h) w_x dx + \tilde{f}_{j+\frac{1}{2}} w_{j+\frac{1}{2}}^- - \tilde{f}_{j-\frac{1}{2}} w_{j-\frac{1}{2}}^+, \end{aligned} \quad (4.85)$$

for $j = 1, \dots, N$. Here $\hat{f}_{j+\frac{1}{2}}$, $\widehat{f'p}_{j+\frac{1}{2}}$, and $\tilde{f}_{j+\frac{1}{2}}$ are the numerical fluxes, which are

carefully chosen for the stability consideration as follows:

$$\begin{aligned}
 \hat{f}_{j+\frac{1}{2}} &: \text{ monotone numerical flux, e.g., the Godunov flux in Section 2.1,} \\
 \widehat{f'p}_{j+\frac{1}{2}} &= \frac{\llbracket f(u_h) \rrbracket_{j+\frac{1}{2}}}{\llbracket u_h \rrbracket_{j+\frac{1}{2}}} (p_h)_{j+\frac{1}{2}}^+, \\
 \tilde{f}_{j+\frac{1}{2}} &= f((u_h)_{j+\frac{1}{2}}^-).
 \end{aligned} \tag{4.86}$$

The resulting semi-discrete scheme features a similar stability property as the linear case. In particular, we have the following proposition.

Proposition 4.40. *The semi-discrete LDG scheme (4.85) with the numerical fluxes (4.86) is L^2 stable.*

Proof. Over cell I_j , let $v = u_h$ and $w = p_h$ in scheme (4.85), we can derive the following cell entropy inequality,

$$\frac{d}{dt} \int_{I_j} u_h^2 dx + \int_{I_j} p_h^2 dx = \hat{H}_{j+\frac{1}{2}} - \hat{H}_{j-\frac{1}{2}} + \Theta_{j-\frac{1}{2}}, \tag{4.87}$$

where $\hat{H}_{j+\frac{1}{2}}$ is the numerical entropy flux defined by

$$\hat{H}_{j+\frac{1}{2}} = \epsilon_1 \left(F((u_h)_{j+\frac{1}{2}}^-) - (u_h)_{j+\frac{1}{2}}^- \hat{f}_{j+\frac{1}{2}} \right) + \epsilon_2 \frac{\llbracket f(u_h) \rrbracket_{j+\frac{1}{2}}}{\llbracket u_h \rrbracket_{j+\frac{1}{2}}} (p_h)_{j+\frac{1}{2}}^+ (u_h)_{j+\frac{1}{2}}^-$$

with $F(u) = \int^u f(s) ds$, and

$$\Theta_{j-\frac{1}{2}} = \epsilon_1 \left(\llbracket F(u_h) \rrbracket_{j-\frac{1}{2}} - \hat{f}_{j-\frac{1}{2}} \llbracket u_h \rrbracket_{j-\frac{1}{2}} \right) \leq 0$$

4.3. A NEW LWDG SCHEME WITH SUPERCONVERGENCE

because of the monotonicity of the numerical flux \hat{f} and $\epsilon_1 < 0$. The L^2 stability follows by summing up (4.87) over j . ■

Similar to the linear case, one can formulate a LWDG scheme for solving (2.1) based on the semi-discrete scheme (4.85) with second order accuracy in time: find $p_h^n, u_h^{n+1} \in V_h^k$ such that, $\forall v, w \in V_h^k$, we have

$$\begin{aligned} \int_{I_j} u_h^{n+1} v \, dx &= \int_{I_j} u_h^n v_x \, dx + \Delta t \left(\int_{I_j} \left(f(u_h^n) - \frac{\Delta t}{2} f'(u_h^n) p_h^n \right) v_x \, dx \right. \\ &\quad \left. - \left(\hat{f}_{j+\frac{1}{2}} - \frac{\Delta t}{2} \widehat{f'p}_{h,j+\frac{1}{2}} \right) v_{j+\frac{1}{2}}^- + \left(\hat{f}_{j-\frac{1}{2}} - \frac{\Delta t}{2} \widehat{f'p}_{h,j-\frac{1}{2}} \right) v_{j-\frac{1}{2}}^+ \right), \end{aligned} \quad (4.88)$$

$$\int_{I_j} p_h^n w \, dx = - \int_{I_j} f(u_h^n) w_x \, dx + \tilde{f}_{j+\frac{1}{2}} w_{j+\frac{1}{2}}^- - \tilde{f}_{j-\frac{1}{2}} w_{j-\frac{1}{2}}^+,$$

for $j = 1, \dots, N$ and $n = 0, 1, \dots$. The numerical fluxes $\hat{f}_{j+\frac{1}{2}}$, $\widehat{f'p}_{j+\frac{1}{2}}$, and $\tilde{f}_{j+\frac{1}{2}}$ are chosen as (4.86). Again, note that the new LWDG scheme (4.88) is defined by the semi-discrete LDG scheme (4.81) coupled with the forward Euler time discretization with $\epsilon_1 = -\Delta t$ and $\epsilon_2 = -\frac{\Delta t^2}{2}$.

The higher temporal order accuracy can be attained by incorporating more time derivatives in the Taylor expansion. For example, the third order temporal derivative can be added to achieve third order accuracy. However, we find it difficult to formulate a semi-discrete LDG scheme, for which the L^2 stability can be proved,

when solving the following PDE:

$$u_t = \epsilon_1 f(u)_x + \epsilon_2 (f'(u)f(u)_x)_x + \epsilon_3 (f''(u)(f(u)_x)^2 + f'(u)(f'(u)f(u)_x)_x)_x. \quad (4.89)$$

Here, $\epsilon_1 < 0$, $\epsilon_2 > 0$, and $\epsilon_3 < 0$ are constant. On the other hand, we can mimic the procedure of the linear case by introducing another auxiliary variable

$$q = (f'(u)p)_x,$$

and obtain

$$f''(u)(f(u)_x)^2 + f'(u)(f'(u)f(u)_x)_x = f''(u)p^2 + f'(u)q.$$

Similarly, to the scheme (4.88), we can develop a LWDG scheme with third order accuracy in time, and the simple alternating fluxes are chosen for the third order terms in the simulations [115, 114]. From our extensive numerical tests, the proposed third order scheme appears to be stable.

Remark 4.41. *The proposed scheme can be directly extended to high dimensional problems as the original LWDG scheme. However, much more auxiliary variables are needed. From this point of view, the proposed scheme is much more involved of implementation than the original LWDG scheme when solving high dimensional problems.*

Hyperbolic Conservation Law System: Euler System

The proposed scheme can be extended for solving hyperbolic conservation law systems. To illustrate the idea, we consider the 1-D Euler system

$$\begin{pmatrix} \rho \\ M \\ E \end{pmatrix}_t + \begin{pmatrix} M \\ vM + P \\ v(E + P) \end{pmatrix}_x = 0, \quad (4.90)$$

where ρ is density, v is velocity, $M = \rho v$ is momentum, E is total energy and P is pressure given by the equation of state $P = (\gamma - 1)(E - \frac{1}{2}vM)$ with $\gamma = 1.4$. Below, we consider a second order accurate LW procedure by repeatedly using the Euler equation (4.90) and its differentiation versions, we have

$$\begin{aligned} \rho(t + \Delta t) &\approx \rho(t) + \Delta t \rho_t + \frac{\Delta t^2}{2} \rho_{tt} \\ &= \rho(t) - \Delta t M_x + \frac{\Delta t^2}{2} (vM + P)_{xx}, \\ M(t + \Delta t) &\approx M(t) + \Delta t M_t + \frac{\Delta t^2}{2} M_{tt} \\ &= M(t) - \Delta t (vM + P)_x + \frac{\Delta t^2}{2} \left((\gamma - 1) (v(E + P))_x \right. \\ &\quad \left. + (3 - \gamma)v(vM + P)_x + \frac{\gamma - 3}{2} v^2 M_x \right)_x, \\ E(t + \Delta t) &\approx E(t) + \Delta t E_t + \frac{\Delta t^2}{2} E_{tt} \\ &= E(t) - \Delta t (v(E + P))_x + \frac{\Delta t^2}{2} \left(\left(\frac{3}{2}(\gamma - 1) + \gamma \frac{E}{P} \right) (vM + P)_x \right. \\ &\quad \left. + \left((\gamma - 1)v^3 - \frac{\gamma E v}{\rho} \right) M_x + \gamma v (v(E + P))_x \right)_x. \end{aligned} \quad (4.91)$$

Similar to the scalar case, define the following three auxiliary variables

$$p = M_x, \quad q = (vM + P)_x, \quad r = (v(E + P))_x. \quad (4.92)$$

Plug (4.92) into (4.91), we obtain

$$\begin{aligned} \rho(t + \Delta t) &\approx \rho(t) - \Delta t M_x + \frac{\Delta t^2}{2} q_x \\ M(t + \Delta t) &\approx M(t) - \Delta t (vM + P)_x + \frac{\Delta t^2}{2} \left(\frac{\gamma - 3}{2} v^2 p + (3 - \gamma) v q + (\gamma - 1) r \right)_x \\ E(t + \Delta t) &\approx E(t) - \Delta t (v(E + P))_x + \frac{\Delta t^2}{2} \left(\left((\gamma - 1) v^3 - \frac{\gamma v E}{\rho} \right) p \right. \\ &\quad \left. + \left(\frac{3}{2} (\gamma - 1) + \gamma \frac{E}{P} \right) q + \gamma v r \right)_x. \end{aligned} \quad (4.93)$$

Then, a second order LWDG method can be formulated based on (4.92) and (4.93). Again, higher order accuracy can be obtained by incorporating more derivatives in the truncated Taylor expansion (4.91) and introducing the corresponding auxiliary variables. In the simulation, we adopt the Godunov flux for the first order spatial derivative and the alternating fluxes for the high order derivatives in the LDG framework. The details of the formulation are omitted for brevity.

4.3.4 Fourier Analysis for the New LWDG Schemes

In this subsection, we apply the classical Fourier analysis to the newly proposed LWDG scheme to study its superconvergence properties. In particular, we follow the

4.3. A NEW LW DG SCHEME WITH SUPERCONVERGENCE

approach in Section 4.1 to analyze the eigen-structure of the amplification matrix for the LW DG schemes via symbolic computations.

We first note that, similar to a RK DG scheme (see equation (4.2) and (4.27)), the original and the new LW DG schemes for solving the model equation (4.1) can both be written in a matrix form

$$\mathbf{u}_j^{n+1} = \sum_{m=-m_0}^{m_1} A_m \mathbf{u}_{j+m}^n, \quad (4.94)$$

where $(j - m_0, \dots, j + m_1)$ is the stencil. For example, denote by $CFL = \frac{\Delta t}{\Delta x}$, we have

$$\begin{aligned} \mathbf{u}_j^{n+1} &= \begin{pmatrix} CFL + 2CFL^2 & \frac{CFL}{2} + CFL^2 \\ -6CFL - 6CFL^2 & -3CFL - 3CFL^2 \end{pmatrix} \mathbf{u}_{j-1}^n \\ &+ \begin{pmatrix} 1 - CFL - 4CFL^2 & -\frac{CFL}{2} - \frac{CFL^2}{2} \\ 6CFL - 6CFL^2 & 1 - 3CFL - 12CFL^2 \end{pmatrix} \mathbf{u}_j^n \\ &+ \begin{pmatrix} 2CFL^2 & -\frac{CFL^2}{2} \\ 12CFL^2 & -3CFL^2 \end{pmatrix} \mathbf{u}_{j+1}^n, \end{aligned}$$

for the proposed LW DG scheme of P^1 DG spatial discretization combined with a second order LW procedure.

Substituting the ansatz (4.3) into the matrix form of the scheme (4.94), we obtain

4.3. A NEW LWDG SCHEME WITH SUPERCONVERGENCE

the following iterative equation for the coefficient vector $\hat{\mathbf{u}}^n$,

$$\hat{\mathbf{u}}^{n+1} = G\hat{\mathbf{u}}^n, \quad (4.95)$$

where G is the amplification matrix, given by

$$G = \sum_{m=-m_0}^{m_1} A_m \exp(im\xi), \quad \xi = \omega\Delta x. \quad (4.96)$$

It is observed from the symbolic computations that G is diagonalizable. Denote the $k+1$ eigenvalues as $\lambda_0, \dots, \lambda_k$. The linear stability requires $|\lambda_l| \leq 1$, $l = 0, \dots, k$. Similar to a RKDG scheme, there is only one physically relevant eigenvalue, denoted by λ_0 , which approximates the analytical eigenvalue $\exp(-i\omega\Delta t)$ for the original or new LWDG scheme, see Proposition 4.1 and Proposition 4.25. Under certain CFL time step restriction, the other k non-physical eigenvalues will be damped exponentially fast as one evolves the numerical solution in time.

Below, we analyze the dispersion and dissipation errors for the physically relevant eigenvalue λ_0 via symbolic computations. Note that such analysis of RKDG schemes and original LWDG schemes are given in Section 4.2.4 and [116], respectively. The comparison between the two LWDG schemes will be drawn based on the dispersion and dissipation errors analysis, which can partially explain why the new LWDG scheme exhibits superconvergence but the original one does not.

We perform symbolic computations on Mathematica. Below is a summary for the two LWDG schemes when $k = 1, 2$. Assume $0 < cfl \leq 1$ and $\omega\Delta x \ll 1$ in the

asymptotic expansions.

- New LWDG scheme

- P^1 case

$$\lambda_0 = 1 - i\omega\Delta t - \frac{\omega^2}{2}\Delta t^2 - \frac{i\omega^4}{72}\Delta t\Delta x^3 + \mathcal{O}(\Delta t\Delta x^4)$$

- P^2 case

$$\lambda_0 = 1 - i\omega\Delta t - \frac{\omega^2}{2}\Delta t^2 + \frac{i\omega^3}{6}\Delta t^3 - \frac{i\omega^6}{7200}\Delta t\Delta x^5 + \mathcal{O}(\Delta t\Delta x^6)$$

- Original LWDG scheme [116]

- P^1 case

$$\lambda_0 = 1 - i\omega\Delta t - \frac{\omega^2}{2}\Delta t^2 - \frac{i\omega^3}{12}\Delta t^2\Delta x + \mathcal{O}(\Delta t\Delta x^3)$$

- P^2 case

$$\lambda_0 = 1 - i\omega\Delta t - \frac{\omega^2}{2}\Delta t^2 + \frac{i\omega^3}{6}\Delta t^3 - \frac{\omega^4}{120} \left(\frac{(20cfl + 5)\Delta t^4 - 2\Delta t^3\Delta x - \Delta t^2\Delta x^2}{1 + cfl} \right) + \mathcal{O}(\Delta t\Delta x^4)$$

4.3. A NEW LWDG SCHEME WITH SUPERCONVERGENCE

The normalized dispersion and dissipation error of the physically relevant eigenvalue λ_0 is defined as

$$e_0 = \frac{1}{\Delta t}(\lambda_0 - \exp(-i\omega\Delta t)). \quad (4.97)$$

In Section 4.1, the error e_0 is analyzed for the RKDG schemes. It is found that e_0 is $(2k + 1)^{th}$ order accurate in space and p^{th} order accurate in time when the approximation space V_h^k and the p^{th} order SSP RK method are used. And such super high order accuracy of e_0 in space contributes to superconvergence properties for the RKDG scheme. We would like to analyze the error e_0 for the two LWDG schemes. Here, we only consider P^1 and P^2 cases, and adopt a LW time discretization procedure such that the resulting LWDG scheme has the same order accuracy in space and time. Denote e_0^N and e_0^O as the normalized dispersion and dissipation errors (4.97) for the new LWDG scheme and the original LWDG scheme, respectively.

By a simple Taylor expansion,

- P^1 case

$$e_0^N = -\frac{\omega^4}{72}\Delta x^3 - \frac{i\omega^3}{6}\Delta t^2 + \mathcal{O}(\Delta x^4 + \Delta t^3), \quad (4.98)$$

$$e_0^O = -\frac{i\omega^3}{12}\Delta t\Delta x - \frac{i\omega^3}{6}\Delta t^2 + \mathcal{O}(\Delta x^3 + \Delta t^3). \quad (4.99)$$

- P^2 case

$$e_0^N = -\frac{i\omega^6}{7200}\Delta x^5 - \frac{\omega^4}{24}\Delta t^3 + \mathcal{O}(\Delta x^6 + \Delta t^4), \quad (4.100)$$

$$e_0^O = -\frac{\omega^4}{cfl+1} \left(\frac{(5cfl+2)}{24}\Delta t^3 - \frac{1}{60}\Delta t^2\Delta x - \frac{1}{120}\Delta t\Delta x^2 \right) + \mathcal{O}(\Delta x^3 + \Delta t^4). \quad (4.101)$$

Discussion: First we note that the order of the leading term of error e_0^N in the asymptotic expansion is the same as that from the RKDG scheme, which is $(2k+1)^{th}$ order in space and $(k+1)^{th}$ order in time for both P^1 and P^2 cases. With the similar argument in Section 4.1, it is implied that the the numerical error by the new LWDG scheme does not significantly grow if the spatial error dominates. In fact, we only need to choose sufficiently small CFL number to reduce the temporal error. In the simulation, we let CFL=0.01 and the superconvergent $(2k+1)^{th}$ order accuracy can be numerically observed when the mesh is relatively coarse. On the other hand, under the assumption $cfl < 1$, error e_0^O by the original LWDG scheme is dominated by

$$e_0^O = \mathcal{O}(\Delta t\Delta x^k),$$

rather than a super high order term $\mathcal{O}(\Delta x^{2k+1})$ as in the new LWDG scheme.

Numerical Results

Below, we provide a collection of 1-D and 2-D numerical experiments to investigate the superconvergence properties of the newly proposed LWDG schemes in different settings.

Example 4.42. Consider the linear advection problem:

$$u_t + u_x = 0, \quad x \in [0, 2\pi], \quad (4.102)$$

with the initial condition

$$u(0, x) = \sin(x), \quad (4.103)$$

and periodic boundary conditions. In the simulation, we choose $\Delta t = \text{CFL} \Delta x$ with $\text{CFL} = 0.01$. In Table 4.19, we report the L^2 and L^∞ errors and the orders of accuracy before and after the post-processing procedure for the proposed LWDG scheme. Unlike the original LWDG scheme, it is clearly observed that the magnitude of numerical errors by the new LWDG schemes is greatly reduced through the post-processing procedure and the orders of accuracy are also enhanced from $k + 1$ to $2k + 1$. The observation indicates that numerical error of the new LWDG scheme is order of $2k + 1$ in space in terms of negative-order norms. A rigorous proof of the claim is subject to future investigation.

Then we test the long time behavior of the L^2 errors by the proposed LWDG scheme in comparison to that by the original LWDG scheme. In Figure 4.17, we report the time evolution of numerical errors by the two types of LWDG P^2 schemes with $N = 50$ up to time $T = 500$. It is observed that the error by the new LWDG scheme does not significantly grow over a long period of time, which is very similar to the result by the RKDG P^2 scheme shown in Figure 4.16. Also note that the error by the original LWDG scheme grows linearly with respect to time.

In Section 4.1, the following error is defined and studied for RKDG schemes:

$$\bar{e} = u_h(t = 2\pi) - u_h(t = 4\pi). \quad (4.104)$$

For a RKDG scheme, the error \bar{e} is in the order of $2k + 1$ in space, which indicates that the numerical error will not significantly grow for a long time period. Here, we also check the error \bar{e} for the proposed LWDG scheme. In Table 4.20, we report the L^2 norms of error \bar{e} and orders of accuracy. Similar to the RKDG scheme, $(2k + 1)^{th}$ order of accuracy is observed, which implies that the numerical error by the new LWDG scheme does not significantly grow for a long time period. The observation is consistent with the results given in Figure 4.17. Also note that the original LWDG scheme does not enjoy such a superconvergence property.

In Figure 4.18, we plot the errors of the numerical solutions before and after post-processing, and errors \bar{e} in absolute value and in logarithmic scale for the new LWDG P^2 scheme and the original LWDG P^2 scheme. It is observed that, for both schemes, the post-processed errors and errors \bar{e} are much less oscillatory and also much smaller in magnitude than the pre-processed errors. Moreover, note that the magnitude of post-processed errors and errors \bar{e} by the new LWDG scheme is smaller than that by the original LWDG scheme, as the former ones are fifth order accurate in space, but the latter ones are only third order.

Example 4.43. Consider 1-D nonlinear Burgers' equation:

$$u_t + \left(\frac{u^2}{2}\right)_x = 0, \quad x \in [0, 2\pi], \quad (4.105)$$

4.3. A NEW LWDG SCHEME WITH SUPERCONVERGENCE

Table 4.19: One-dimensional linear advection. New LWDG scheme. $T=1$. CFL=0.01.

	mesh	Before post-processing				After post-processing			
		L^2 error	order	L^∞ error	order	L^2 error	order	L^∞ error	order
P^1	20	4.22E-03	–	1.37E-02	–	3.73E-04	–	5.41E-04	–
	40	1.06E-03	1.99	3.51E-03	1.96	4.25E-05	3.14	6.09E-05	3.15
	60	4.72E-04	2.00	1.57E-03	1.98	1.22E-05	3.09	1.74E-05	3.09
	80	2.65E-04	2.00	8.85E-04	1.99	5.04E-06	3.06	7.18E-06	3.07
	100	1.70E-04	2.00	5.68E-04	1.99	2.55E-06	3.05	3.63E-06	3.06
P^2	20	1.07E-04	–	3.66E-04	–	2.52E-06	–	3.59E-06	–
	40	1.34E-05	3.00	4.62E-05	2.99	4.47E-08	5.82	6.36E-08	5.82
	60	3.96E-06	3.00	1.37E-05	3.00	4.37E-09	5.74	6.21E-09	5.74
	80	1.67E-06	3.00	5.78E-06	3.00	8.60E-10	5.65	1.22E-09	5.65
	100	8.56E-07	3.00	2.96E-06	3.00	2.48E-10	5.57	3.53E-10	5.57
P^3	20	2.07E-06	–	5.44E-06	–	6.89E-08	–	9.75E-08	–
	40	1.30E-07	3.95	3.85E-07	3.82	2.76E-10	7.97	3.90E-10	7.96
	60	2.55E-08	4.03	7.55E-08	4.02	1.08E-11	7.98	1.53E-11	7.98
	80	8.07E-09	4.00	2.37E-08	4.02	1.09E-12	7.98	1.54E-12	7.98
	100	3.31E-09	4.00	9.74E-09	3.99	1.85E-13	7.96	2.61E-13	7.96

with the initial condition

$$u(0, x) = \sin(x) + 2, \quad (4.106)$$

and periodic boundary conditions. The time step is simply chosen as $\Delta t = \text{CFL}\Delta x$ with CFL= 0.01. In Table 4.21, we report the L^2 errors and the orders of accuracy before and after the post-processing procedure for both the proposed LWDG scheme and the original LWDG scheme. The superconvergent results are clearly observed for the new scheme. However, similar to the linear case, the post-processed error is only order of $k + 1$ by the original LWDG scheme. We remark that using a upwind flux, e.g., the Godunov flux, for the first order derivative term is crucial to obtain superconvergent results in the proposed LWDG formulation. If a general monotone numerical flux such as the Lax-Friedrichs flux is used, the $(2k + 1)^{th}$

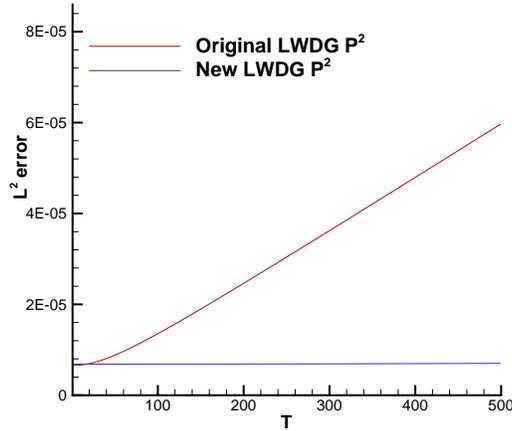


Figure 4.17: Long time behavior of errors. 1-D linear advection. Original LWDG P^2 versus new LWDG P^2 scheme. CFL=0.01.

order superconvergence result may not be observed. In Figure 4.19, we plot errors of the numerical solution before and after post-processing in absolute value and in logarithmic scale for both LWDG P^2 schemes. The highly oscillatory nature of the pre-processed errors is observed for both schemes. The post-processed errors do not oscillate much and the magnitude is also much smaller. Again the magnitude of the post-processed errors by the new LWDG scheme is much smaller than that by the original LWDG scheme.

In order to study error \bar{e} for the Burgers' equation, we add a source term to equation (4.105) such that $\sin(x + t) + 2$ is the exact solution. A similar strategy is used in Example 4.32. We consider the following Burgers' equation with a source term:

$$u_t + \left(\frac{u^2}{2}\right)_x = (\sin(x + t) + 3) \cos(x + t), \quad x \in [0, 2\pi], \quad (4.107)$$

4.3. A NEW LWDG SCHEME WITH SUPERCONVERGENCE

Table 4.20: One-dimensional linear advection. New LWDG scheme. The L^2 norms of error \bar{e} and the orders of accuracy. CFL=0.01.

	P^1		P^2		P^3	
mesh	L^2 error	order	L^2 error	order	L^2 error	order
20	1.89E-03	–	1.88E-06	–	1.40E-09	–
40	2.39E-04	2.99	5.96E-08	4.98	7.42E-12	7.56
60	7.08E-05	3.00	7.98E-09	4.96	4.36E-13	6.99
80	2.99E-05	3.00	1.93E-09	4.93	5.96E-14	6.92
100	1.53E-05	3.00	6.50E-10	4.88	1.34E-14	6.68

with the initial condition

$$u(0, x) = \sin(x) + 2,$$

and periodic boundary condition. We report the L^2 norms of error \bar{e} and the orders of accuracy in Table 4.22 for the proposed LWDG scheme and the original LWDG scheme. $(2k + 1)^{th}$ order of accuracy is clearly observed for the new LWDG scheme. Again, such superconvergence result is not observed for the original LWDG scheme. In Figure 4.20, we plot numerical errors and errors \bar{e} in absolute value and in logarithmic scale for both LWDG P^2 schemes. Note that, the error \bar{e} is much less oscillatory and smaller in magnitude than the pre-processed error for the two schemes, however, the magnitude of the error \bar{e} by the new LWDG scheme is much smaller than that by the original LWDG scheme.

Example 4.44. Consider 1-D Euler system (4.90). Let the initial condition to be

$$\rho(0, x) = 1 + 0.2 \sin(\pi x), \quad v(0, x) = 1 \quad \text{and} \quad P(0, x) = 1,$$

subject to the 2-periodic boundary conditions. The exact solution is $\rho(t, x) = 1 +$

4.3. A NEW LWDG SCHEME WITH SUPERCONVERGENCE

Table 4.21: Burgers' equation (4.105). T=0.2. CFL= 0.01.

	mesh	Before post-processing				After post-processing			
		New LWDG		Original LWDG		New LWDG		Original LWDG	
		L^2 error	order	L^2 error	order	L^2 error	order	L^2 error	order
P^1	20	4.46E-03	–	4.44E-03	–	3.44E-04	–	3.25E-04	–
	40	1.13E-03	1.98	1.13E-03	1.97	3.48E-05	3.30	3.58E-05	3.18
	60	5.05E-04	1.99	5.05E-04	1.99	9.39E-06	3.23	1.16E-05	2.77
	80	2.85E-04	1.99	2.85E-04	1.99	3.74E-06	3.20	5.74E-06	2.45
	100	1.83E-04	1.99	1.83E-04	1.99	1.84E-06	3.18	3.47E-06	2.26
P^2	20	1.37E-04	–	1.28E-04	–	3.35E-05	–	3.41E-05	–
	40	1.75E-05	2.97	1.62E-05	2.98	6.42E-07	5.70	8.02E-07	5.41
	60	5.23E-06	2.98	4.81E-06	2.99	5.90E-08	5.89	1.31E-07	4.47
	80	2.22E-06	2.99	2.04E-06	2.98	1.06E-08	5.96	4.69E-08	3.58
	100	1.14E-06	2.99	1.05E-06	2.99	2.77E-09	6.03	2.28E-08	3.24

$0.2 \sin(\pi(x - t))$, $v(t, x) = 1$ and $P(t, x) = 1$. We compute the numerical solution up to T=2. In Table 4.23, we report the L^2 and L^∞ errors and the orders of accuracy for density ρ before and after applying the post-processing procedure for the new LWDG scheme. Similar superconvergence property is observed as the scalar cases. In the simulation, the Godunov flux is used for the first order derivatives in order to obtain the superconvergence result.

Then we check the error \bar{e} which is defined as

$$\bar{e} = \rho_h(t = 2) - \rho_h(t = 4),$$

since the period of the solution in time is 2. The L^2 norms of error \bar{e} and orders of accuracy are reported in Table 4.24 for P^1 and P^2 . $(2k + 1)^{th}$ order accuracy is observed.

4.3. A NEW LWDG SCHEME WITH SUPERCONVERGENCE

Table 4.22: Burgers' equation (4.107). The L^2 norms of error \bar{e} and the orders of accuracy. CFL= 0.01.

	New LWDG				Original LWDG			
	P^1		P^2		P^1		P^2	
mesh	L^2 error	order	L^2 error	order	L^2 error	order	L^2 error	order
20	5.25E-04	–	8.44E-07	–	6.06E-04	–	4.91E-05	–
40	7.21E-05	2.86	2.67E-08	4.98	9.47E-05	2.68	6.18E-06	2.99
60	2.16E-05	2.97	3.54E-09	4.99	3.34E-05	2.57	1.83E-06	3.00
80	9.14E-06	2.99	8.42E-10	4.99	1.66E-05	2.42	7.74E-07	3.00
100	4.69E-06	3.00	2.77E-10	4.98	9.91E-06	2.31	3.96E-07	3.00

We also use the following benchmark Lax problem, for which discontinuous solution structures will be developed, to test the performance of the proposed LWDG scheme. Consider the Riemann initial condition:

$$(\rho, v, P) = (0.455, 0.689, 3.528) \quad x \leq 0; \quad (\rho, v, P) = (0.5, 0, 0.571) \quad x > 0. \quad (4.108)$$

A robust WENO limiting procedure with the TVB limiter as a troubled cell indicator is used to suppress the spurious oscillations [83]. In Figure 4.21 and Figure 4.22, we plot the numerical solutions of density ρ at $T=1.3$ with different TVB constant M . Comparable numerical results by the proposed LWDG scheme are observed to those by the original LWDG scheme [88].

Example 4.45. Consider the following 2-D linear advection equation

$$u_t + u_x + u_y = 0, \quad x \times y \in [0, 2\pi]^2, \quad (4.109)$$

4.3. A NEW LWDG SCHEME WITH SUPERCONVERGENCE

Table 4.23: 1-D Euler system. New LWDG scheme. L^2 and L^∞ errors of density ρ . T=2. CFL= 0.01.

	mesh	Before post-processing				After post-processing			
		L^2 error	order	L^∞ error	order	L^2 error	order	L^∞ error	order
P^1	20	8.61E-04	–	2.53E-03	–	2.04E-04	–	2.92E-04	–
	40	2.13E-04	2.01	6.77E-04	1.90	2.48E-05	3.04	3.52E-05	3.05
	60	9.45E-05	2.00	3.07E-04	1.95	7.26E-06	3.03	1.03E-05	3.03
	80	5.31E-05	2.00	1.74E-04	1.97	3.05E-06	3.02	4.32E-06	3.02
	100	3.40E-05	2.00	1.12E-04	1.98	1.55E-06	3.02	2.20E-06	3.02
P^2	20	2.14E-05	–	7.35E-05	–	6.33E-07	–	8.98E-07	–
	40	2.67E-06	3.00	9.24E-06	2.99	1.30E-08	5.60	1.85E-08	5.60
	60	7.92E-07	3.00	2.74E-06	3.00	1.42E-09	5.47	2.01E-09	5.47
	80	3.34E-07	3.00	1.16E-06	3.00	3.04E-10	5.36	4.31E-10	5.36
	100	1.71E-07	3.00	5.92E-07	3.00	9.40E-11	5.26	1.33E-10	5.26

Table 4.24: 1-D Euler system. New LWDG scheme. The L^2 norms of error \bar{e} and the orders of accuracy of density ρ . CFL= 0.01.

mesh	P^1		P^2	
	L^2 error	order	L^2 error	order
20	3.79E-04	–	3.77E-07	–
40	4.77E-05	2.99	1.19E-08	4.98
60	1.42E-05	3.00	1.60E-09	4.96
80	5.98E-06	3.00	3.87E-10	4.93
100	3.06E-06	3.00	1.30E-10	4.88

with the initial condition

$$u(0, x, y) = \sin(x + y), \quad (4.110)$$

and periodic boundary conditions in both x - and y -directions. We set CFL=0.01 to make the spatial error dominant in the simulation. In Table 4.25, we report the L^2 and L^∞ errors and the orders of accuracy before and after applying the post-processor for the proposed LWDG scheme. Similar to the results for the 1-D advection problem, $(k + 1)^{th}$ order of accuracy is observed for the pre-processed errors in both L^2 and

4.3. A NEW LWDG SCHEME WITH SUPERCONVERGENCE

L^∞ norms. Moreover, post-processed numerical solutions are superconvergent with the order of $2k + 1$, which implies that numerical error is also order of $2k + 1$ in space in negative-order norms for the 2-D case.

Then, we would like to test the long time behavior of the numerical errors. We set the mesh size as $N_x \times N_y = 50 \times 50$ and report the numerical errors for the two LWDG P^2 scheme at time $T = 1, 20, 50, 100,$ and 200 in Table 4.26. We observe that the numerical error by the new LWDG scheme does not significantly grow for a long time period, which indicates, as the RKDG scheme shown in [15], the new LWDG scheme features similar superconvergence property for the 2-D linear advection problem. However, as the 1-D case, the error by the original LWDG scheme is observed to noticeably grow at the beginning of the simulation.

Finally, we study the error \bar{e} which is defined as

$$\bar{e} = u_h(t = \pi) - u_h(t = 2\pi).$$

As the results reported in Example 4.35, error \bar{e} by the RKDG scheme is order of $2k + 1$ in space for solving the 2-D linear advection problem. In Table 4.27, we report the L^2 norms of error \bar{e} and orders of accuracy. The $(2k + 1)^{th}$ order of accuracy is observed. Such superconvergent behavior of error \bar{e} implies that the numerical error by the proposed LWDG scheme solving the 2-D linear advection problem does not significantly grow for a long time simulation. At last, we want to point out that, similar to the 1-D cases, the original LWDG scheme for the 2-D advection problem does not exhibit any superconvergence discussed above. We omit the numerical

4.3. A NEW LWDG SCHEME WITH SUPERCONVERGENCE

Table 4.25: Two-dimensional linear advection. New LWDG scheme. $T=1$. CFL=0.01.

	mesh	Before post-processing				After post-processing			
		L^2 error	order	L^∞ error	order	L^2 error	order	L^∞ error	order
P^1	20×20	5.40E-02	–	4.28E-02	–	1.61E-02	–	3.65E-03	–
	40×40	1.32E-02	2.03	1.13E-02	1.92	1.97E-03	3.03	4.46E-04	3.03
	60×60	5.86E-03	2.01	5.11E-03	1.96	5.80E-04	3.02	1.31E-04	3.02
	80×80	3.29E-03	2.01	2.89E-03	1.98	2.43E-04	3.02	5.49E-05	3.02
	100×100	2.10E-03	2.00	1.86E-03	1.98	1.24E-04	3.01	2.80E-05	3.02
P^2	20×20	2.99E-03	–	3.40E-03	–	8.14E-05	–	1.83E-05	–
	40×40	3.74E-04	3.00	4.25E-04	3.00	2.13E-06	5.25	4.80E-07	5.26
	60×60	1.11E-04	3.00	1.26E-04	2.99	2.62E-07	5.17	5.90E-08	5.17
	80×80	4.68E-05	3.00	5.33E-05	3.00	6.00E-08	5.13	1.35E-08	5.12
	100×100	2.39E-05	3.00	2.73E-05	3.00	1.92E-08	5.10	4.33E-09	5.10

results from the original LWDG schemes for brevity.

Table 4.26: Long time behavior of errors. 2-D linear advection. LWDG P^2 schemes. The L^2 errors at $T = 1, 20, 50, 100$, and 200 . $N_x \times N_y = 50 \times 50$. CFL=0.01.

Scheme	$T=1$	$T=20$	$T=50$	$T=100$	$T=200$
New LWDG P^2	1.92E-04	1.92E-04	1.94E-04	1.99E-04	2.21E-04
Original LWDG P^2	1.91E-04	1.95E-04	2.16E-04	2.77E-04	4.46E-04

Table 4.27: Two-dimensional linear advection. New LWDG scheme. The L^2 norms of \bar{e} and the orders of accuracy. CFL=0.01.

mesh	P^1		P^2	
	L^2 error	order	L^2 error	order
20×20	4.61E-02	–	1.67E-04	–
40×40	5.96E-03	2.95	5.30E-06	4.98
60×60	1.77E-03	2.99	5.30E-06	4.99
80×80	7.50E-04	2.99	1.66E-07	4.99
100×100	3.84E-04	3.00	5.47E-08	4.99

4.3. A NEW LWDG SCHEME WITH SUPERCONVERGENCE

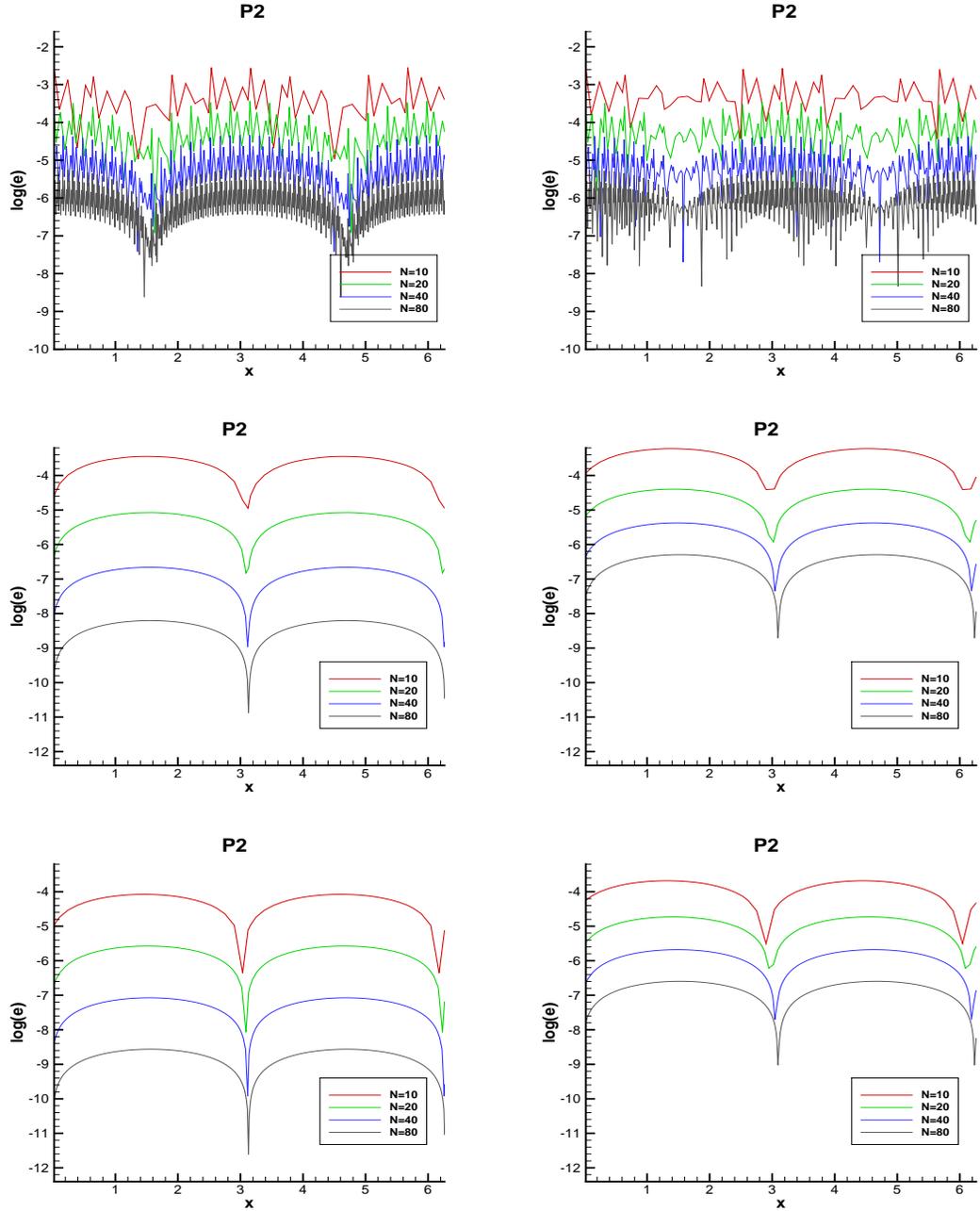


Figure 4.18: Linear advection. The new LWDG P^2 scheme (left column). The original LWDG P^2 scheme (right column). Before post-processing (top). After post-processing (middle). Error \bar{e} (bottom). CFL= 0.01. $T=4\pi$.

4.3. A NEW LWDG SCHEME WITH SUPERCONVERGENCE

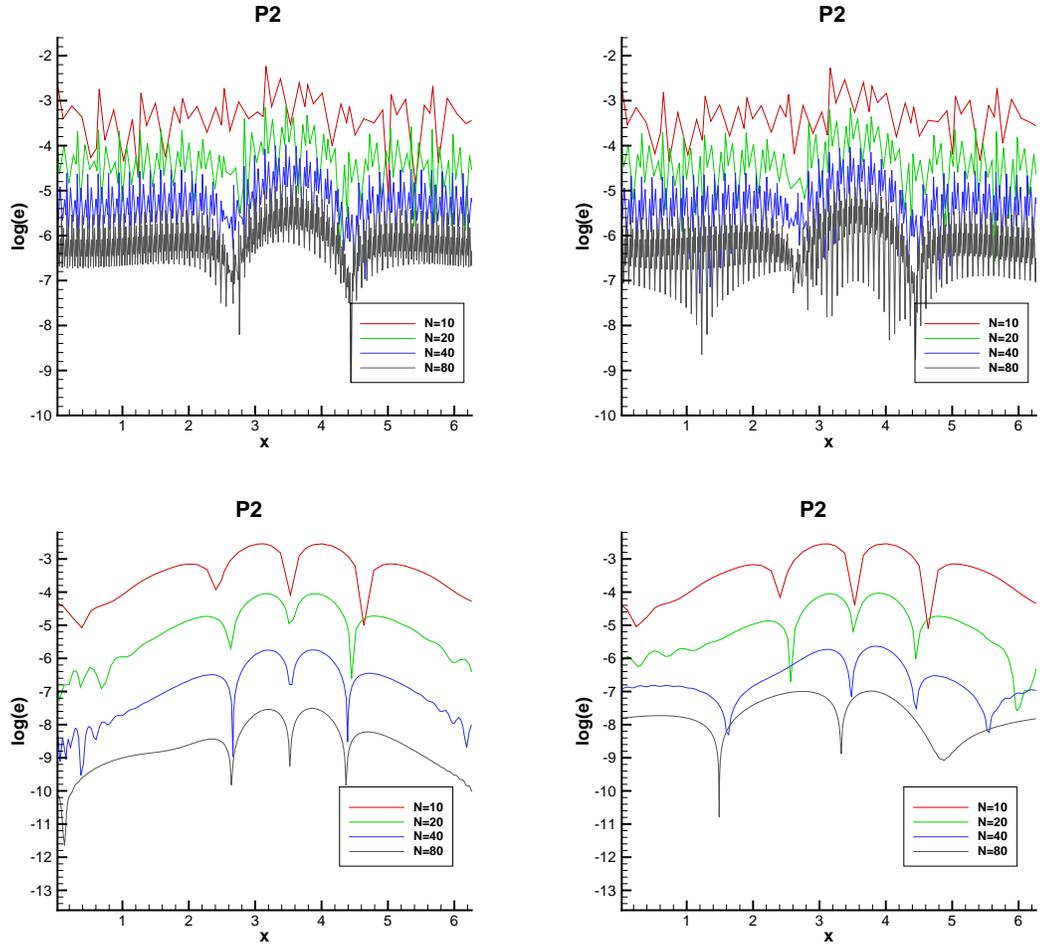


Figure 4.19: Burgers' equation (4.105). New LWDG scheme (left column). Original LWDG scheme (right column). P^2 is used. Before post-processing (top). After post-processing (bottom). CFL=0.01, T=0.2.

4.3. A NEW LWDG SCHEME WITH SUPERCONVERGENCE

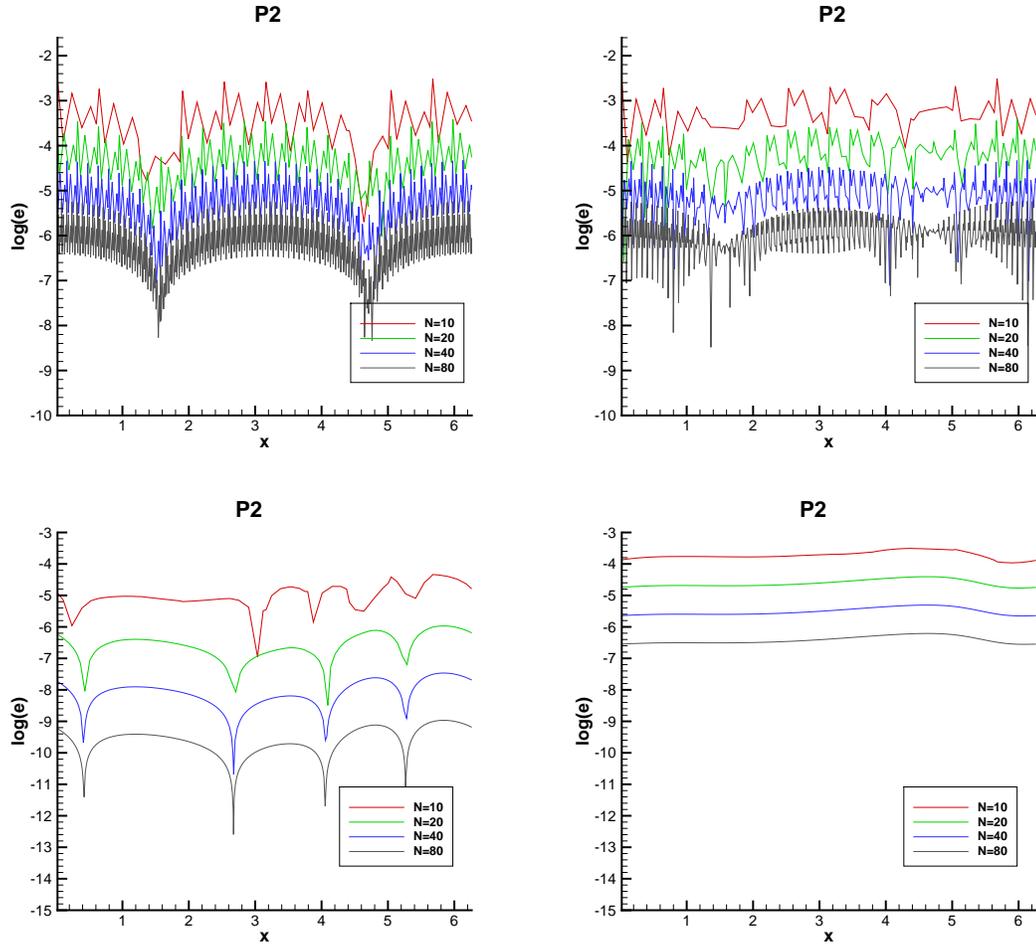


Figure 4.20: Burgers' equation (4.107). New LWDG P^2 scheme (left column). Original LWDG P^2 scheme (right column). Before post-processing (top). Error \bar{e} (bottom). CFL= 0.01. $T=4\pi$.

4.3. A NEW LWDG SCHEME WITH SUPERCONVERGENCE

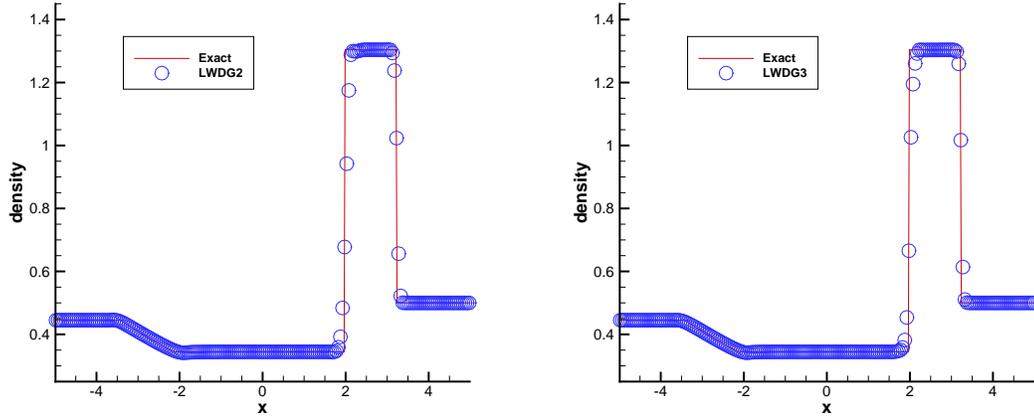


Figure 4.21: One-dimensional Euler system. Lax problem. Density ρ . New LWDG scheme. TVB constant $M=1$. P^1 (left). P^2 (right). $N=200$. $T=1.3$.

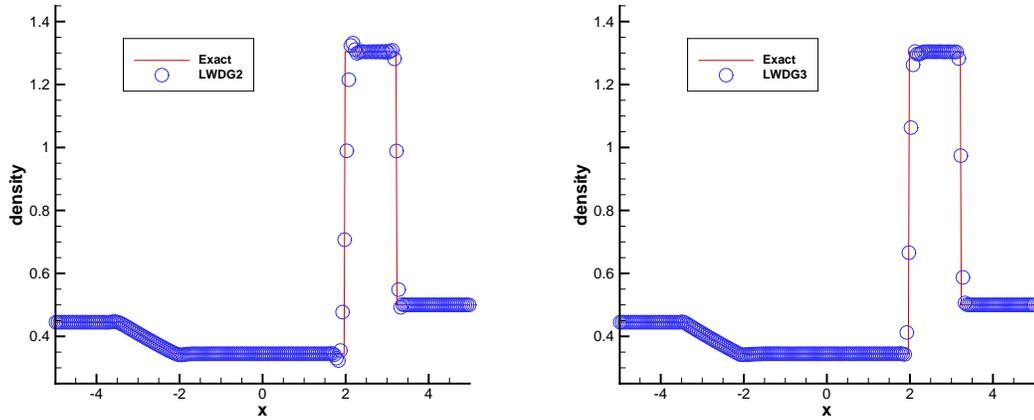


Figure 4.22: One-dimensional Euler system. Lax problem. Density ρ . New LWDG scheme. TVB constant $M=50$. P^1 (left). P^2 (right). $N=200$. $T=1.3$.

CHAPTER 5

Conclusions

In this dissertation, we have systematically studied high order numerical schemes for transport problems with focuses on semi-Lagrangian (SL)-type schemes and super-convergence for discontinuous Galerkin (DG) methods.

First, a family of high order SL methods for linear transport equations were reviewed, formulated, and compared including SLDG methods and SL finite difference weighted essentially non-oscillatory (WENO) schemes. In the SL framework, there is no time step restriction for linear stability, and hence the allowance of taking a extremely large time step leads to great saving of computational cost. Based on these efficient high order numerical solvers, a hybrid SL numerical method has been

designed to solve the Vlasov-Poisson (VP) system. The proposed hybrid method combines the SL/Eulerian Runge-Kutta (RK) DG scheme and the SLWENO scheme for the spacial advection and velocity ac/deceleration, respectively. The DG scheme is adopted for spacial advection due to its compactness. RKDG offers the flexibility to deal with complicated geometries and boundary conditions when multi-dimensional problems are considered. SLWENO is adopted for velocity advection due to its robustness in resolving filamentation solution structures of the VP system. We apply the hybrid methods to several test examples, such as linear advection, solid body rotation, and the VP system to demonstrate the performance. Furthermore, in order to correct the low order splitting error, a method couples the SLWENO scheme with the integral deferred correction (IDC) framework was formulated for Vlasov simulations: we adopt the dimensional splitting SLWENO scheme as a base scheme to get a predicted solution in IDC, and the low order dimensional splitting error is iteratively reduced by solving the error equations again in a dimensional splitting fashion in the correction steps in IDC. We extended the scheme to solve the guiding center Vlasov equation and the two-dimensional incompressible flow in vorticity stream-function formulation. A collection of numerical experiments demonstrate great performance of the proposed high order scheme in resolving solutions structures, even in a long term simulation. However, it is numerically observed that the IDC framework renders some CFL time step restrictions, despite the SL evolution mechanism in the prediction and correction steps of IDC. We quantify such CFL restrictions via Fourier analysis for several high order methods that we used in simulations. Then, the proposed SLDG scheme on Cartesian meshes was extended to the cubed-sphere geometry for

solving global transport equations. The cubed-sphere geometry is known to be free from polar singularities, and well-suited for element-based Galerkin methods, such as DG. A set of benchmark tests were performed to demonstrate the robustness of the proposed SLDG scheme on the sphere. The comparison between the SLDG scheme and another two popular global transport schemes including the RKDG scheme and a conservative semi-Lagrangian scheme (CSLAM) was performed, which shows the advantages of SLDG in different settings.

In the second part of this work, we discussed superconvergence properties of DG and local DG (LDG) methods for linear hyperbolic and parabolic problems via Fourier approach. Especially, superconvergence properties of DG with uniform meshes for linear problems with periodic boundary conditions are discussed in terms of: (1) the dissipation and dispersion error of physically relevant eigenvalue; this part of error is related to the negative-order norm of DG error; (2) the eigenvectors with Lagrangian basis functions based on shifted Radau points; this part of error is related to superconvergence at Radau points; and (3) the long time behavior of the DG error. We conclude that the error of a numerical solution at Radau points will not significantly grow over a long period of time that is on the order of Δx^{-k+1} and Δx^{-k} for DG and LDG, respectively. Moreover, supraconvergence properties of DG and LDG methods are studied based on our understanding of the eigen-structure of the amplification matrix. Extensive numerical examples for scalar and system of equations are demonstrated to verify the analysis and to assess the superconvergence properties of DG and LDG in more general settings. We also discussed the superconvergence properties of DG methods with Lax-Wendroff (LW) time discretization.

Numerical results indicate that the original LWDG scheme does not exhibit several important superconvergence properties including accuracy enhancement by applying a post-processor and long time behaviors of numerical errors. In order to restore the superconvergence in the LW framework, we formulated a new LWDG scheme, in which the techniques borrowed from the local DG (LDG) scheme were adopted to obtain high order spatial derivatives. Again, Fourier analysis via symbolic computations was used to theoretically investigate superconvergence property of the newly proposed scheme.

Future work consists of

- extending the hybrid methods to multi-dimensional VP and Vlasov-Maxwell (VM) systems;
- mitigating or getting rid of the CFL time step restriction in the IDC framework;
- developing a high order asymptotic-preserving SL scheme for the VP system in the quasi-neutral limit;
- developing a high order non-split SL scheme for multi-dimensional transport problems;
- seeking analytical proofs for our symbolic analysis obtained in Chapter 4 and analyzing superconvergence properties of various DG formulations.

Bibliography

- [1] S. Adjerid, K.D. Devine, J.E. Flaherty, and L. Krivodonova. A posteriori error estimation for discontinuous Galerkin solutions of hyperbolic problems. *Comput. Method Appl. M.*, 191(11):1097–1112, 2002.
- [2] S. Adjerid and T.C. Massey. Superconvergence of discontinuous Galerkin solutions for a nonlinear scalar hyperbolic problem. *Comput. Method Appl. M.*, 195(25):3331–3346, 2006.
- [3] M. Ainsworth. Dispersive and dissipative behaviour of high order discontinuous Galerkin finite element methods. *J. Comput. Phys.*, 198(1):106–130, 2004.
- [4] M. Ainsworth, P. Monk, and W. Muniz. Dispersive and dissipative properties of discontinuous Galerkin finite element methods for the second-order wave equation. *J. Sci. Comput.*, 27(1):5–40, 2006.
- [5] F. Bassi and S. Rebay. A high-order accurate discontinuous finite element method for the numerical solution of the compressible Navier–Stokes equations. *J. Comput. Phys.*, 131(2):267–279, 1997.
- [6] ML Begue, A. Ghizzo, P. Bertrand, E. Sonnendrücker, and O. Coulaud. Two-dimensional semi-Lagrangian Vlasov simulations of laser–plasma interaction in the relativistic regime. *J. Plasma Phys.*, 62(04):367–388, 1999.

- [7] N. Besse and E. Sonnendrücker. Semi-Lagrangian schemes for the Vlasov equation on an unstructured mesh of phase space. *J. Comput. Phys.*, 191(2):341–376, 2003.
- [8] C.K. Birdsall and A.B. Langdon. *Plasma Physics Via Computer Simulation*. CRC Press, 2005.
- [9] A. Bourlioux, A.T. Layton, and M.L. Minion. High-order multi-implicit spectral deferred correction methods for problems of reactive flow. *J. Comput. Phys.*, 189(2):651–675, 2003.
- [10] J. Bramble and A. Schatz. Higher order local accuracy by averaging in the finite element method. *Math. Comput.*, 31(137):94–111, 1977.
- [11] W. Cao, Z. Zhang, and Q. Zou. Superconvergence of Discontinuous Galerkin method for linear hyperbolic equations. *arXiv:1311.6938*.
- [12] J.A. Carrillo and F. Vecil. Nonoscillatory interpolation methods applied to Vlasov-based models. *SIAM J. Sci. Comput.*, 29(3):1179–1206, 2007.
- [13] C.-Z. Cheng and G. Knorr. The integration of the Vlasov equation in configuration space. *J. Comput. Phys.*, 22(3):330–351, 1976.
- [14] Y. Cheng, I.M. Gamba, and P.J. Morrison. Study of conservation and recurrence of Runge–Kutta discontinuous Galerkin schemes for Vlasov–Poisson systems. *J. Sci. Comput.*, 56(2):319–349, 2013.
- [15] Y. Cheng and C.-W. Shu. Superconvergence and time evolution of discontinuous Galerkin finite element solutions. *J. Comput. Phys.*, 227(22):9612–9627, 2008.
- [16] Y. Cheng and C.-W. Shu. Superconvergence of local discontinuous Galerkin methods for one-dimensional convection-diffusion equations. *Comput. Struct.*, 87(11-12):630–641, 2009.
- [17] 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 J. Numer. Anal.*, 47(6):4044–4072, 2010.
- [18] P.N. Childs and K.W. Morton. Characteristic Galerkin methods for scalar conservation laws in one dimension. *SIAM J. Numer. Anal.*, 27(3):553–594, 1990.

- [19] A. Christlieb, Y. Liu, and Z. Xu. High order operator splitting methods based on an integral deferred correction framework. *submitted*, 2013.
- [20] A. Christlieb, C. Macdonald, and B. Ong. Parallel high-order integrators. *SIAM J. Sci. Comput.*, 32(2):818–835, 2010.
- [21] A. Christlieb, B. Ong, and J.-M. Qiu. Comments on high order integrators embedded within integral deferred correction methods. *Comm. Appl. Math. Comput. Sci*, 4(1):27–56, 2009.
- [22] A. Christlieb, B. Ong, and J.-M. Qiu. Integral deferred correction methods constructed with high order Runge–Kutta integrators. *Math. Comput.*, 79:761–783, 2010.
- [23] B. Cockburn, S. Hou, and C.-W. Shu. The Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws IV: The multidimensional case. *Math. Comput.*, 54:545–581, 1990.
- [24] B. Cockburn, C. Johnson, C.-W. Shu, and E. Tadmor. *Advanced Numerical Approximation of Nonlinear Hyperbolic Equations*. Springer, New York, 1998.
- [25] B. Cockburn, F. Li, and C.-W. Shu. Locally divergence-free discontinuous Galerkin methods for the Maxwell equations. *J. Comput. Phys.*, 194(2):588–610, 2004.
- [26] B. Cockburn, S.Y. Lin, and C.-W. Shu. TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws III: one-dimensional systems. *J. Comput. Phys.*, 84(1):90–113, 1989.
- [27] B. Cockburn, M. Luskin, C.-W. Shu, and E. Suli. Enhanced accuracy by post-processing for finite element methods for hyperbolic equations. *Math. Comput.*, 72(242):577–606, 2003.
- [28] B. Cockburn and C.-W. Shu. TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws II: general framework. *Math. Comput.*, pages 411–435, 1989.
- [29] B. Cockburn and C.-W. Shu. The Runge-Kutta local projection p^1 -discontinuous Galerkin finite element method for scalar conservation laws. *Math. Model. Numer. Anal.(M²AN)*, 25:337–361, 1991.
- [30] B. Cockburn and C.-W. Shu. The local discontinuous Galerkin method for time-dependent convection-diffusion systems. *SIAM J. Numer. Anal.*, 35(6):2440–2463, 1998.

- [31] B. Cockburn and C.-W. Shu. The Runge-Kutta discontinuous Galerkin method for conservation laws V: multidimensional systems. *J. Comput. Phys.*, 141(2):199–224, 1998.
- [32] B. Cockburn and C.-W. Shu. Runge–Kutta discontinuous Galerkin methods for convection-dominated problems. *J. Sci. Comput.*, 16(3):173–261, 2001.
- [33] R. Courant, K. Friedrichs, and H. Lewy. Über die partiellen Differenzgleichungen der mathematischen Physik. *Math. Ann.*, 100(1):32–74, 1928.
- [34] N. Crouseilles, M. Mehrenberger, and E. Sonnendrücker. Conservative semi-Lagrangian schemes for Vlasov equations. *J. Comput. Phys.*, 229(6):1927–1953, 2010.
- [35] P. Degond, L. Pareschi, and G. Russo. *Modeling and Computational Methods for Kinetic Equations*. Springer, 2004.
- [36] A. Dutt, L. Greengard, and V. Rokhlin. Spectral deferred correction methods for ordinary differential equations. *BIT*, 40(2):241–266, 2000.
- [37] C. Erath, P. H. Lauritzen, J. H. Garcia, and H. M. Tufo. Integrating a scalable and efficient semi-Lagrangian multi-tracer transport scheme in HOMME. *Procedia Comput. Sci.*, 9:994–1003, 2012.
- [38] F. Filbet and E. Sonnendrücker. Comparison of Eulerian Vlasov solvers. *Comput. Phys. Commun.*, 150(3):247–266, 2003.
- [39] F. Filbet, E. Sonnendrücker, and P. Bertrand. Conservative numerical schemes for the Vlasov equation. *J. Comput. Phys.*, 172(1):166–187, 2001.
- [40] R. Frank and C. W. Ueberhuber. Iterated defect correction for differential equations. I. Theoretical results. *Computing*, 20(3):207–228, 1978.
- [41] B.D. Fried and S.D. Conte. The plasma dispersion function. *The Plasma Dispersion Function, New York: Academic Press, 1961*, 1, 1961.
- [42] D. Gottlieb. Strang-type difference schemes for multidimensional problems. *SIAM J. Numer. Anal.*, 9(4):650–661, 1972.
- [43] S. Gottlieb, D.I. Ketcheson, and C.-W. Shu. High order strong stability preserving time discretizations. *J. Sci. Comput.*, 38(3):251–289, 2009.
- [44] S. Gottlieb, C.-W. Shu, and E. Tadmor. Strong stability preserving high order time discretization methods. *SIAM Rev.*, 43:89–112, 2001.

- [45] W. Guo, F. Li, and J.-X. Qiu. Local-structure-preserving discontinuous Galerkin methods with Lax-Wendroff type time discretizations for Hamilton-Jacobi equations. *J. Sci. Comput.*, 47(2):239–257, 2011.
- [46] W. Guo, R.D. Nair, and J.-M. Qiu. A conservative semi-Lagrangian discontinuous Galerkin scheme on the cubed-sphere. *Mon. Wea. Rev.*, 142(1):457–475, 2013.
- [47] E. Hairer, C. Lubich, and G. Wanner. *Geometric numerical integration*, volume 31 of *Springer Series in Computational Mathematics*. Springer-Verlag, Berlin, second edition, 2006. Structure-preserving algorithms for ordinary differential equations.
- [48] D.M. Hall and R.D. Nair. Discontinuous Galerkin transport on the spherical Yin-Yang overset mesh. *Mon. Wea. Rev.*, 142:264–282, 2013.
- [49] E. Hansen and A. Ostermann. High order splitting methods for analytic semi-groups exist. *BIT*, 49(3):527–542, 2009.
- [50] L.M. Harris, P.H. Lauritzen, and R. Mittal. A flux-form version of the conservative semi-Lagrangian multi-tracer transport scheme (CSLAM) on the cubed sphere grid. *J. Comput. Phys.*, 230(4):1215–1237, 2011.
- [51] A. Harten, B. Engquist, S. Osher, and S.R. Chakravarthy. Uniformly high order accurate essentially non-oscillatory schemes, III. *J. Comput. Phys.*, 71(2):231–303, 1987.
- [52] R. Heath, I. Gamba, P. Morrison, and C. Michler. A discontinuous Galerkin method for the Vlasov–Poisson system. *J. Comput. Phys.*, 231(4):1140–1174, 2012.
- [53] R.E. Heath, I.M. Gamba, P.J. Morrison, and C. Michler. A discontinuous Galerkin method for the Vlasov-Poisson system. *J. Comput. Phys.*, 231(4):1140–1174, 2012.
- [54] R.W. Hockney and J.W. Eastwood. *Computer Simulation Using Particles*. CRC Press, 2010.
- [55] F. Hu, M.Y. Hussaini, and P. Rasetarinera. An analysis of the discontinuous Galerkin method for wave propagation problems. *J. Comput. Phys.*, 151(2):921–946, 1999.

- [56] F. Huot, A. Ghizzo, P. Bertrand, E. Sonnendrücker, and O. Coulaud. Instability of the time splitting scheme for the one-dimensional and relativistic Vlasov–Maxwell system. *J. Comput. Phys.*, 185(2):512–531, 2003.
- [57] L. Ji, Y. Xu, and J.K. Ryan. Accuracy-enhancement of discontinuous Galerkin solutions for convection-diffusion equations in multiple-dimensions. *Math. Comput.*, 81(280):1929–1950, 2012.
- [58] L. Ji, Y. Xu, and J.K. Ryan. Negative-order norm estimates for nonlinear hyperbolic conservation laws. *J. Sci. Comput.*, 54(2-3):531–548, 2013.
- [59] G.-S. Jiang and C.-W. Shu. On a cell entropy inequality for discontinuous Galerkin methods. *Math. Comput.*, 62(206):531–538, 1994.
- [60] G.-S. Jiang and C.-W. Shu. Efficient implementation of weighted ENO schemes. *J. Comput. Phys.*, 126(1):202–228, 1996.
- [61] L. Krivodonova and R. Qin. An analysis of the spectrum of the discontinuous Galerkin method. *Appl. Numer. Math.*, 64:1–18, 2013.
- [62] P.H. Lauritzen, R.D. Nair, and P.A. Ullrich. A conservative semi-Lagrangian multi-tracer transport scheme (CSLAM) on the cubed-sphere grid. *J. Comput. Phys.*, 229(5):1401–1424, 2010.
- [63] P. Lax and B. Wendroff. Systems of conservation laws. *Commun. Pur. Appl. Math.*, 13(2):217–237, 1960.
- [64] A.T Layton and M.L Minion. Conservative multi-implicit spectral deferred correction methods for reacting gas dynamics. *J. Comput. Phys.*, 194(2):697–715, 2004.
- [65] B.P. Leonard, A.P. Lock, and M.K. MacVean. Conservative explicit unrestricted-time-step multidimensional constancy-preserving advection schemes. *Mon. Wea. Rev.*, 124(11):2588–2606, 1996.
- [66] R.J. LeVeque. High-resolution conservative algorithms for advection in incompressible flow. *SIAM J. Numer. Anal.*, 33(2):627–665, 1996.
- [67] R.J. LeVeque. *Finite Volume Methods for Hyperbolic Problems*, volume 31. Cambridge University Press, 2002.
- [68] S.J. Lin and R.B. Rood. Multidimensional flux-form semi-Lagrangian transport schemes. *Mon. Wea. Rev.*, 124(9):2046–2070, 1996.

- [69] X.-D. Liu, S. Osher, and T. Chan. Weighted essentially non-oscillatory schemes. *J. Comput. Phys.*, 115(1):200–212, 1994.
- [70] R.B. Lowrie, P.L. Roe, and B. Van Leer. A space-time discontinuous Galerkin method for the time-accurate numerical solution of hyperbolic conservation laws. In *AIAA Computational Fluid Dynamics Conference, 12 th, San Diego, CA*, pages 135–150, 1995.
- [71] D. Majewski, D. Liermann, P. Prohl, B. Ritter, M. Buchhold, T. Hanisch, G. Paul, W. Wergen, and J. Baumgardner. The operational global icosahedral–hexagonal gridpoint model gme: Description and high-resolution tests. *Mon. Wea. Rev.*, 130(2), 2002.
- [72] P.J. Morrison. Hamiltonian description of the ideal fluid. *Rev. Mod. Phys.*, 70(2):467, 1998.
- [73] R.D. Nair and C. Jablonowski. Moving vortices on the sphere: A test case for horizontal advection problems. *Mon. Wea. Rev.*, 136(2):699–711, 2008.
- [74] R.D. Nair and P.H. Lauritzen. A class of deformational flow test cases for linear transport problems on the sphere. *J. Comput. Phys.*, 229(23):8868–8887, 2010.
- [75] R.D. Nair and B. Machenhauer. The mass-conservative cell-integrated semi-Lagrangian advection scheme on the sphere. *Mon. Wea. Rev.*, 130(3), 2002.
- [76] R.D. Nair, S.J. Thomas, and R.D. Loft. A discontinuous Galerkin transport scheme on the cubed sphere. *Mon. Wea. Rev.*, 133(4):814–828, 2005.
- [77] T. Nakamura, R. Tanaka, T. Yabe, and K. Takizawa. Exactly conservative semi-Lagrangian scheme for multi-dimensional hyperbolic equations with directional splitting technique. *J. Comput. Phys.*, 174(1):171–207, 2001.
- [78] T. Nakamura and T. Yabe. Cubic interpolated propagation scheme for solving the hyper-dimensional Vlasov–Poisson equation in phase space. *Comput. Phys. Commun.*, 120(2):122–154, 1999.
- [79] X. Peng, F. Xiao, and K. Takahashi. Conservative constraint for a quasi-uniform overset grid on the sphere. *Q. J. Roy. Meteor. Soc.*, 132(616):979–996, 2006.
- [80] V. Pereyra. Iterated deferred corrections for nonlinear operator equations. *Numer. Math.*, 10:316–323, 1967.

- [81] T.E. Peterson. A note on the convergence of the discontinuous Galerkin method for a scalar hyperbolic equation. *SIAM J. Numer. Anal.*, 28(1):133–140, 1991.
- [82] J.A. Pudykiewicz. On numerical solution of the shallow water equations with chemical reactions on icosahedral geodesic grid. *J. Comput. Phys.*, 230(5):1956–1991, 2011.
- [83] J. Qiu and C.-W. Shu. Runge-Kutta discontinuous Galerkin method using WENO limiters. *SIAM J. Sci. Comput.*, 26(3):907–929, 2005.
- [84] J.-M. Qiu and A. Christlieb. A Conservative high order semi-Lagrangian WENO method for the Vlasov Equation. *J. Comput. Phys.*, 229:1130–1149, 2010.
- [85] J.-M. Qiu and C.-W. Shu. Conservative high order semi-Lagrangian finite difference WENO methods for advection in incompressible flow. *J. Comput. Phys.*, 230(4):863–889, 2011.
- [86] J.-M. Qiu and C.-W. Shu. Conservative semi-Lagrangian finite difference WENO formulations with applications to the Vlasov equation. *Commun. Comput. Phys.*, 10(4):979, 2011.
- [87] J.-M. Qiu and C.-W. Shu. Positivity preserving semi-Lagrangian discontinuous Galerkin formulation: Theoretical analysis and application to the Vlasov–Poisson system. *J. Comput. Phys.*, 230(23):8386–8409, 2011.
- [88] J.-X. Qiu, M. Dumbser, and C.-W. Shu. The discontinuous Galerkin method with Lax–Wendroff type time discretizations. *Comput. Method Appl. M.*, 194(42):4528–4543, 2005.
- [89] J.-X. Qiu and C.-W. Shu. Finite difference WENO schemes with Lax–Wendroff type time discretizations. *SIAM J. Sci. Comput.*, 24(6):2185–2198, 2003.
- [90] M. Rančić. Semi-Lagrangian piecewise biparabolic scheme for two-dimensional horizontal advection of a passive scalar. *Mon. Wea. Rev.*, 120(7):1394–1406, 1992.
- [91] M. Rančić, R. J. Purser, and F. Mesinger. A global shallow-water model using an expanded spherical cube: Gnomonic versus conformal coordinates. *Q. J. Roy. Meteor. Soc.*, 122(532):959–982, 1996.
- [92] W.H. Reed and T.R. Hill. Triangular mesh methods for the neutron transport equation. *Los Alamos Report LA-UR-73-479*, 1973.

BIBLIOGRAPHY

- [93] J.P. René Laprise and A. Plante. A class of semi-Lagrangian integrated-mass (SLM) numerical transport algorithms. *Mon. Wea. Rev.*, 123(2):553–565, 1995.
- [94] M. Restelli, L. Bonaventura, and R. Sacco. A semi-Lagrangian discontinuous Galerkin method for scalar advection by incompressible flows. *J. Comput. Phys.*, 216(1):195–215, 2006.
- [95] C. Ronchi, R. Iacono, and P.S. Paolucci. The cubed sphere: a new method for the solution of partial differential equations in spherical geometry. *J. Comput. Phys.*, 124(1):93–114, 1996.
- [96] J. Rossmannith. A wave propagation method for hyperbolic systems on the sphere. *J. Comput. Phys.*, 213(2):629–658, 2006.
- [97] J. Rossmannith and D. Seal. A positivity-preserving high-order semi-Lagrangian discontinuous Galerkin scheme for the Vlasov-Poisson equations. *J. Comput. Phys.*, 230:6203–6232, 2011.
- [98] T.F. Russell and M.A. Celia. An overview of research on Eulerian-Lagrangian localized adjoint methods (ELLAM). *Adv. Water Resour.*, 25(8):1215–1231, 2002.
- [99] 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 J. Sci. Comput.*, 26(3):821–843, 2005.
- [100] R. Sadourny. Conservative finite-difference approximations of the primitive equations on quasi-uniform spherical grids. *Mon. Wea. Rev.*, 100(2):136–144, 1972.
- [101] D. Sármany, MA Botchev, and J.J.W. van der Vegt. Dispersion and dissipation error in high-order Runge-Kutta discontinuous Galerkin discretisations of the Maxwell equations. *J. Sci. Comput.*, 33(1):47–74, 2007.
- [102] J. Schaeffer. Higher order time splitting for the linear Vlasov equation. *SIAM J. Numer. Anal.*, 47(3):2203–2223, 2009.
- [103] S. Sherwin. Dispersion analysis of the continuous and discontinuous Galerkin formulation. *Lecture Notes in Computational Science and Engineering*, 11:425–432, 2000.
- [104] C.-W. Shu and S. Osher. Efficient implementation of essentially non-oscillatory shock-capturing schemes. *J. Comput. Phys.*, 77(2):439–471, 1988.

BIBLIOGRAPHY

- [105] E. Sonnendruecker, J. Roche, P. Bertrand, and A. Ghizzo. The semi-Lagrangian method for the numerical resolution of the Vlasov equation. *J. Comput. Phys.*, 149(2):201–220, 1999.
- [106] A. Staniforth and J. Cote. Semi-Lagrangian integration schemes for atmospheric models: A review. *Mon. Wea. Rev.*, 119(9):2206–2223, 1991.
- [107] H.J. Stetter. The defect correction principle and discretization methods. *Numer. Math.*, 29(4):425–443, 1978.
- [108] G. Strang. On the construction and comparison of difference schemes. *SIAM J. Numer. Anal.*, 5(3):506–517, 1968.
- [109] M. Taylor, J. Tribbia, and M. Iskandarani. The spectral element method for the shallow water equations on the sphere. *J. Comput. Phys.*, 130(1):92–108, 1997.
- [110] T. Umeda, M. Ashour-Abdalla, and D. Schriver. Comparison of numerical interpolation schemes for one-dimensional electrostatic Vlasov code. *J. Plasma Phys.*, 72(06):1057–1060, 2006.
- [111] J.P. Verboncoeur. Particle simulation of plasmas: review and advances. *Plasma Phys. Contr. F.*, 47(5A):A231–A260, 2005.
- [112] D.L. Williamson, J.B. Drake, J.J. Hack, R. Jakob, and P.N. Swarztrauber. A standard test set for numerical approximations to the shallow water equations in spherical geometry. *J. Comput. Phys.*, 102(1):211–224, 1992.
- [113] Y. Xia, Y. Xu, and C.-W. Shu. Efficient time discretization for local discontinuous Galerkin methods. *Discrete Contin. Dyn. Syst. Ser. B*, 8(3):677–693, 2007.
- [114] Y. Xu and C.W. Shu. Local discontinuous galerkin methods for high-order time-dependent partial differential equations. *Commun. Comput. Phys.*, 7(1):1–46, 2010.
- [115] J. Yan and C.-W. Shu. A local discontinuous Galerkin method for KdV type equations. *SIAM J. Numer. Anal.*, 40(2):769–791, 2002.
- [116] H. Yang, F. Li, and J. Qiu. Dispersion and dissipation errors of two fully discrete discontinuous Galerkin methods. *J. Sci. Comput.*, 55(3):552–574, 2013.

- [117] Y. Yang and C.-W. Shu. Analysis of optimal superconvergence of discontinuous galerkin method for linear hyperbolic equations. *SIAM J. Numer. Anal.*, 50(6):3110–3133, 2012.
- [118] H. Yoshida. Construction of higher order symplectic integrators. *Phys. Lett. A*, 150(5-7):262–268, 1990.
- [119] M. Zhang and C.-W. Shu. An analysis of three different formulations of the discontinuous Galerkin method for diffusion equations. *Math. Mod. Meth. Appl. S.*, 13(3):395–413, 2003.
- [120] M. Zhang and C.-W. Shu. An analysis of and a comparison between the discontinuous Galerkin and the spectral finite volume methods. *Comput. Fluids*, 34(4):581–592, 2005.
- [121] X. Zhang and C.-W. Shu. On maximum-principle-satisfying high order schemes for scalar conservation laws. *J. Comput. Phys.*, 229:3091–3120, 2010.
- [122] X. Zhang and C.-W. Shu. On positivity preserving high order discontinuous Galerkin schemes for compressible Euler equations on rectangular meshes. *J. Comput. Phys.*, 229:8918–8934, 2010.
- [123] Y. Zhang and R.D. Nair. A Nonoscillatory discontinuous Galerkin transport scheme on the cubed-sphere. *Mon. Wea. Rev.*, 140:3106–3126, 2012.
- [124] X. Zhong and C.-W. Shu. Numerical resolution of discontinuous Galerkin methods for time dependent wave equations. *Comput. Method Appl. M.*, 200(41-44):2814–2827, 2011.
- [125] T. Zhou, Y. Guo, and C.-W. Shu. Numerical study on Landau damping. *Physica D*, 157(4):322–333, 2001.