High Order Numerical Methods: Entropy Stability and Deterministic Solvers of Stochastic PDEs

by

Tianheng Chen

B.Sc., Peking University, Beijing, P. R. China, 2014

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

PROVIDENCE, RHODE ISLAND

May 2019

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

Date________________ ____

Chi-Wang Shu, Ph.D., Advisor

Recommended to the Graduate Council

Date________________ ____

Boris Rozovsky, Ph.D., Reader

Date________________ ____

Johnny Guzmán, Ph.D., Reader

Approved by the Graduate Council

Date________________ ____

Andrew G. Campbell, Dean of the Graduate School

Vita

Education

Brown University, Providence, RI, U.S.A.
Ph.D. in Applied Mathematics (expected in May 2019).
Advisor: Chi-Wang Shu.

Peking University, Beijing, China.
B.Sc. in Mathematics and Computational Sciences, 2014.
Advisor: Huazhong Tang.

Publications

1. T. Chen and B. Rozovskii and C.-W. Shu. Numerical solutions of stochastic PDEs driven by arbitrary type of noise. *Stochastics and Partial Differential Equations: Analysis and Computations*, 7:1–39, 2019.
2. T. Chen and C.-W. Shu. Entropy stable high order discontinuous Galerkin methods with suitable quadrature rules for hyperbolic conservation laws. *Journal of Computational Physics*, 345:427–461, 2017.

Professional Experience

Cubist Systematic Strategies LLC, New York, NY, U.S.A.
Research Analyst Intern, summer 2018.

WorldQuant LLC, Greenwich, CT, U.S.A.
Quantitative Researcher Intern, summer 2017.

Professional Service

Referee for *Journal of Scientific Computing.*

Referee for *Journal of Computational Physics.*

Teaching Experience

Teaching Assistant.

Essential Statistics, Brown University, spring 2016.
Introduction to Computational Linear Algebra, Brown University, fall 2015.

Awards & Honors

Academic Excellence Award, Peking University, 2011, 2012 and 2013.
Gold Medal of Chinese Mathematics Olympiad (CMO, top 30), 2009 and 2010.

Acknowledgments

First of all, I would like to thank my advisor, Prof. Chi-Wang Shu, for pointing out the research opportunities and pitfalls ahead of me, for those genuine advice on my career development, and for kindly sharing stories and jokes in the academic community. He is a knowledgeable, insightful and enthusiastic scholar, as well as a considerate and open-minded instructor. I will always benefit from the research experience under Prof. Shu's guidance.

Secondly, I would like to thank Prof. Boris Rozovsky for giving me the chance to participate in the stochastic analysis project. I really enjoy the interdisciplinary nature of this project, which helps me broaden my knowledge.

I would also like to thank Prof. Johnny Guzmán for his taking the time and effort to be part of my dissertation defense committee. There are many other professors who shared their invaluable ideas and expertise with me. To name a few, I would express my gratitude to Prof. Mark Ainsworth, Prof. Jesse Chan, Prof. Michael Tretyakov and Prof. Zhongqiang Zhang, for the help on my research.

I am lucky to be in Prof. Shu's research group, a lively circle full of talented and warm-hearted people. Thanks Guosheng Fu, Yong Liu, Tong Qin and Zheng Sun, for discussions and friendly debates on research projects and random math problems. Thanks Jianfang Lu, for offering rides and helping me practice for the driver's license. Thanks Kunrui Wang and Bingyu Zhao, for organizing our weekly group meeting.

I am especially thankful to my parents, Wenjun Chen and Ping Li. You have always been encouraging me to believe in myself, and to pursue my interest in mathematics. I thank with love to Xuefei Cao, for all the happiness and hardship we face together. The meaning of your support is beyond what I can express here.

Abstract of "High Order Numerical Methods: Entropy Stability and Deterministic Solvers of Stochastic PDEs", by Tianheng Chen, Ph.D., Brown University, May 2019

This thesis consists of two diverse topics on high order numerical methods for time-dependent partial differential equations (PDEs).

In the first part, we develop a unified framework of entropy stable Discontinuous Galerkin (DG) type methods for systems of hyperbolic conservation laws. The well-known cell entropy inequality of classic DG method (Jiang and Shu (1994) [60]) is limited to the square entropy and assumes exact integration. Our framework overcomes such limitation by designing DG method satisfying entropy inequalities for any given single entropy function, through suitable numerical quadrature rules. The one-dimensional methodology is based on Legendre Gauss-Lobatto quadrature. The main ingredients are discrete operators with summation-by-parts (SBP) property, the flux differencing technique, and entropy stable fluxes at cell boundary. We then extend the methodology to higher space dimensions by constructing SBP operators for simplicial meshes with Gauss-Lobatto type quadrature points. The further extension to more general quadrature points is achieved through careful modification of boundary terms. A local discontinuous Galerkin (LDG) type treatment is also incorporated to enable the generalization to convection-diffusion equations. Extensive numerical experiments are performed to validate the accuracy and shock capturing capability of these entropy stable DG methods.

In the second part, we explore the polynomial chaos expansion method for distribution-free stochastic partial differential equations (SPDEs). So far the theory and numerical practice of SPDEs have dealt almost exclusively with Gaussian noise or Lévy noise. Recently, Mikulevicius and Rozovskii (2016) [78] proposed a distribution-free Skorokhod-Malliavin calculus framework that is based on generalized polynomial chaos (gPC) expansion, and is compatible with arbitrary driving noise. We will analyze these newly developed distribution-free SPDEs. We obtain an estimate for the mean square truncation error in the linear case. The convergence rate is exponential with respect to polynomial order and cubic with respect to number of random variables included. Numerical experiments are conducted to exhibit the efficiency of truncated polynomial chaos solutions in approximating moments and distributions. The theoretical convergence rate is also verified by numerical results.

Contents

List of Tables

List of Figures

Introduction

Time-dependent partial differential equations (PDEs), whose evolving functions are of the form $u(t, \mathbf{x})$ with $(t, \mathbf{x}) \in [0, \infty) \times \mathbb{R}^d$, are ubiquitous in science and engineering. Since analytical solutions of these PDEs are rarely available, numerical methods have to be designed to solve them. Most numerical schemes follow the method of lines principle; that is, we first perform spatial discretization with suitable basis functions and test functions, and transform the problem into a system of ordinary differential equations (ODEs). Then standard ODE solvers (e.g. Runge-Kutta time stepping) can be adopted. Numerical methods are usually classified according to the choice of basis functions and test functions. For example, spectral methods use orthogonal polynomials or trigonometric polynomials as basis function [50], and the implementation of spectral methods is often accomplished with Galerkin approach (where test functions are the same as basis functions) or collocation approach (where test functions are Dirac-δ functions at grid points).

Convergence is undoubtedly a major goal for numerical schemes. The numerical solution should approach the exact solution as we refine the resolution. The well-known Lax equivalence theorem [65] states that for a well-posed linear problem, convergence is guaranteed by *consistency* and *stability*. A method is said to stable if the numerical solution is uniformly bounded under certain norm, which is typically related to the well-posedness of the PDE itself. The most widely used type of norm is the L^2 norm. For linear PDEs with periodic boundary condition, Fourier analysis serves as a powerful tool to prove L^2 stability [45]. On the other hand, consistency requires the truncation error, i.e. the error induced by numerical approximation on a smooth solution of the PDE, to converge to zero. The order of method measures the convergence rate of truncation error, mostly in terms of characteristic mesh size h. Given stability (and linear assumption), high order convergence of truncation error implies high order convergence of numerical solution, and proving consistency is in general much easier than proving convergence directly. In this dissertation, we will focus on two diverse topics concerning high order numerical methods:

1. Entropy stable Discontinuous Galerkin (DG) type methods for systems of hyperbolic conservation laws.
2. Polynomial chaos expansion method for stochastic partial differential equations (SPDEs) driven by arbitrary type of noise.

The first topic handles systems of conservation laws, which encompass applications in oceanography (shallow water equations), aerodynamics (Euler equations), plasma physics (MHD equations) and structural mechanics (nonlinear elasticity) [31]. Entropy conditions, where we require the total amount of a set of convex entropy functions to be non-decreasing, play an essential role in the well-posedness of hyperbolic conservation laws. Therefore, it is natural to seek numerical schemes that satisfy a discrete version of entropy conditions, i.e., entropy stable schemes. Discontinuous Galerkin (DG) methods [17, 16, 15, 19], due to their high order accuracy, local conservation, great parallel efficiency and flexibility for dealing with unstructured meshes, are a popular category of numerical schemes for solving hyperbolic conservation laws. Jiang and Shu [60] proved that the semi-discrete DG schemes satisfy a discrete entropy inequality for the square entropy for scalar conservation laws (i.e., L^2 stability), which is extended to symmetric systems by Hou and Liu [55]. However, these results are limited to the square entropy function only, and all integrals in the DG formulation are assumed to be evaluated exactly, which can be costly or even impossible to implement. In practice one often uses quadrature rules and stability might be affected.

In recent years, there have been rapid developments on entropy stable DG type methods directly built upon numerical integration. DG schemes can be recast into the nodal formulation after quadrature [62, 51]. These nodal DG forms are often named as Discontinuous Galerkin Spectral Element methods (DGSEM) in the literature. In [29, 7], Carpenter, Fisher, Nielsen and Frankel established the entropy stable DGSEM framework on Gauss-Lobatto quadrature points for one-dimensional conservation laws. The main ingredients are the summation-by-parts (SBP) property [27] of operators derived by Gauss-Lobatto nodes, flux differencing with Tadmor's entropy conservative fluxes [95, 96], and Tadmor's entropy stable fluxes at cell interfaces. The entropy stable DGSEM methodology is then generalized to higher space dimensions on unstructured meshes by Chen and Shu [13]. Several generalizations to non Gauss-Lobatto type nodes are also recommended in [9, 10, 21, 1]. As we will see in later chapters, these generalizations all have some drawbacks and trade-off, and we are certainly not at the end of story.

The second topic embraces the realm of SPDEs, which essentially describe functions $u(t, \mathbf{x}, \omega)$ with an extra random dimension ω. SPDEs have found a broad type of applications, including mathematical biology, financial engineering and nonlinear filtering, to quantify the intrinsic uncertainty in these models [72]. The most popular numerical approach to solving SPDEs is the Monte Carlo method, which generates independent random sample paths via direct discretization [80, 115] or averaging over characteristic lines [79]. However, it is well known that Monte Carlo simulation suffers from slow rate of convergence and often requires millions of samples to reach a desired accuracy level. Would it be possible to get rid of sampling? The answer lies in the method of lines principle. In the stochastic polynomial chaos expansion approach, we first construct a complete set of L^2 orthogonal basis functions, called polynomial chaos basis functions, for the underlying probability space. By expanding u as a series of polynomial chaos basis functions, we separate the stochastic part and the deterministic part. These expansion coefficients will in turn satisfy a system deterministic PDEs, called *propagator* [69]. Then deterministic numerical solvers can be applied to the propagator system. The polynomial chaos expansion approach is often recognized as stochastic Galerkin method [105] due to its evident resemblance with spectral Galerkin method.

Traditionally, polynomial chaos expansion approach is well suited for SPDEs driven by Wiener process, or Gaussian white noise [54, 25]. Gaussian white noise can be represented by a series of i.i.d standard Gaussian random variables. The corresponding polynomial chaos expansion is also known as *Hermite chaos* or *Wiener chaos* in the literature. Due to multiple Itô integral formula [59], the stochastic integral is equivalent to the Wick product [103] with white noise, which is written in terms of expansion coefficients. Hence we can easily derive the propagator system. The efficiency of Wiener chaos expansion has been validated by the numerical examples in [116, 118, 56, 73]. Such methodology also works for Lévy randomness [54, 25], and Liu [67] presented some numerical results of SPDEs driven by Poisson noise.

Although the stochastic polynomial chaos expansion is mostly restricted to Gaussian and Lévy random noise, the extension towards arbitrary type of noise has been explored over the last two decades. Xiu and Karniadakis [106, 107] constructed the correspondence between types of random variables and orthogonal polynomials in the Askey scheme [2]. Their generalized polynomial chaos (gPC) technique was very successful for problems with random initial/boundary condition and/or random coefficients [107, 108, 105]. However, gPC expansion is not naturally compatible with stochastic integrals due to the lack of some vital connections, e.g., Wick product, Skorokhod integral [92] and Malliavin calculus [74]. Recently, Mikulevicius and Rozovsky built the distribution-free Skorokhod-Malliavin calculus framework [78] upon gPC expansion, giving rise to a new family of SPDEs under their arbitrary noise paradigm. Then Chen, Rozovsky and Shu [12] studied the numerical aspects of these distribution-free SPDEs. The authors proved that for linear problems, the convergence rate of the mean square error is exponential with respect to polynomial order and cubic with respect to number of random variables included, improving the truncation error estimate in [68].

The rest of this dissertation is organized as follows. Part I provides an incremental and unified framework of entropy stable nodal DG schemes for systems of hyperbolic conservation laws. We will discuss the basic one-dimensional methodology in Chapter 1, the generalization to higher space dimensions in 2, and some possible directions towards general set of nodes in Chapter 3. Part II deals with the polynomial chaos expansion method for distribution-free SPDEs. We concentrate on gPC expansion and distribution-free stochastic analysis in Chapter 4, and the truncation error estimate and numerical analysis in Chapter 5. Concluding remarks will be given in the ending part of dissertation. A few technical details can be found in the appendix. We remark that each part will contain its own set of notations. The same symbol might stand for different concepts in two parts.

Part I

Entropy Stable DG Methods for Systems of Hyperbolic Conservation Laws

Chapter One

One-Dimensional Methodology on Gauss-Lobatto Nodes

In this chapter, we will analyze numerical approximations of systems of conservation laws in one space dimension. The equation is of the form

$$\frac{\partial \mathbf{u}}{\partial t} + \frac{\partial \mathbf{f}(\mathbf{u})}{\partial x} = 0, \quad (t,x) \in [0,\infty) \times \mathbb{R}, \tag{1.1}$$

where $\mathbf{u} = [u^1, \cdots, u^p]^T$ denotes the vector of state variables taking values in a convex set $\Omega \in \mathbb{R}^p$, and $\mathbf{f} = [f^1, \cdots, f^p]^T$ is called the flux function. Define the Jacobian matrix

$$\mathbf{f}'(\mathbf{u}) := \{\frac{\partial f^i}{\partial u^j}(\mathbf{u})\}_{1 \le i,j \le p}.$$

Then the system (1.1) is called hyperbolic if $\mathbf{f}'(\mathbf{u})$ has p real eigenvalues and a complete set of eigenvectors for all $\mathbf{u} \in \Omega$. We shall only consider hyperbolic conservation laws from now on. The name conservation law follows from the fact the total amount of $\mathbf{u}$ is conserved. Formally integrating the equation (1.1) in space, and assuming $\mathbf{u}$ is compactly supported, we come up with the identity

$$\frac{d}{dt}\int_{\mathbb{R}} \mathbf{u}(t,x)dx = 0. \tag{1.2}$$

Particularly, for $p = 1$, we have the simple case of scalar conservation law

$$\frac{\partial u}{\partial t} + \frac{\partial f(u)}{\partial x} = 0, \quad (t,x) \in [0,\infty) \times \mathbb{R}. \tag{1.3}$$

Obviously scalar conservation laws are always hyperbolic.

It is well known that shock waves or contact discontinuities might develop at finite time even for smooth initial condition. Hence we have to interpret (1.1) in the sense of distribution and search for weak solutions. However, weak solutions are not necessarily unique. We need some admissibility criterion to single out the "physically relevant" solution among all weak solutions. Entropy conditions turn out to be the adequate criterion.

This chapter consists of following sections. In Section 1.1 and Section 1.2, we introduce entropy analysis for one-dimensional hyperbolic conservation laws, and briefly demonstrate existence and uniqueness of entropy solution, especially in the scalar case. Entropy stability is well developed for first order methods, which will be discussed in Section 1.3. We bring into the concepts of entropy conservative flux and entropy stable flux , proving that for scalar conservation laws, monotone fluxes [20, 48] are entropy stable with respect to all entropy functions, and for systems, Godunov type fluxes [49] are entropy stable. In Section 1.4, we review the classic cell entropy inequality [60, 55] for DG method, and explain why it only works for the square entropy function. Then we move to the construction of one-dimensional entropy stable nodal DG method [29, 7]. In Section 1.5, we present summation-by-parts matrices and the DGSEM formulation on Gauss-Lobatto nodes. In Section 1.6, we describe the flux differencing trick to achieve entropy balance within an element. Just like classic DG method, we can apply a TVD/TVB limiter and / or a bound-preserving limiter to control oscillations and enhance robustness. In Section 1.7, we will see that these limiters do not violate entropy stability. Finally in Section 1.8, we will report numerical results of both smooth accuracy tests and discontinuous shock-capturing tests.

1.1 Entropy function and entropy variables

Let us first define convex entropy functions.

Definition 1.1. *A convex function $U : \Omega \to \mathbb{R}$ is called an entropy function for* (1.1) *if there exists a function $F : \Omega \to \mathbb{R}$, called entropy flux, such that the following* *integrability condition holds*

$$U'(\mathbf{u})\mathbf{f}'(\mathbf{u}) = F'(\mathbf{u}), \tag{1.4}$$

where $U'(\mathbf{u})$ and $F'(\mathbf{u})$ are viewed as row vectors.

In the scalar case, any convex function U is an entropy function, associated with the entropy flux $F(u) = \int^u U'(s)f'(s)ds$. For more general systems, finding entropy function - entropy flux pairs that satisfy (1.4) is much more difficult. The existence of entropy function is a special property of the system. However, in almost all systems we encounter in practice, we are able to find entropy functions with physical meaning.

Define entropy variables $\mathbf{v} := U'(\mathbf{u})^T$. If we further assume that U is strictly convex, $\mathbf{v}'(\mathbf{u}) = U''(\mathbf{u})$ is symmetric positive-definite, and the mapping $\mathbf{u} \mapsto \mathbf{v}$ is one-to-one and can be regarded as a change of variables. Setting $\mathbf{g}(\mathbf{v}) := \mathbf{f}(\mathbf{u}(\mathbf{v}))$, we rewrite (1.1) in terms of entropy variables

$$\mathbf{u}'(\mathbf{v})\frac{\partial \mathbf{v}}{\partial t} + \mathbf{g}'(\mathbf{v})\frac{\partial \mathbf{v}}{\partial x} = 0. \tag{1.5}$$

The following theorem tells us that the existence of entropy function is equivalent to the symmetry of $\mathbf{u}'(\mathbf{v})$ and $\mathbf{g}'(\mathbf{v})$ [41, 81].

Theorem 1.1. *U is a strictly convex entropy function if and only if $\mathbf{u}'(\mathbf{v})$ is symmetric positive-definite, and $\mathbf{g}'(\mathbf{v})$ is symmetric.* (1.5) *is called the symmetrization of* (1.1).

Proof. We only prove the "only if" part. The "if" part is then straightforward as all arguments hold in both directions. By strict convexity, $\mathbf{u}'(\mathbf{v}) = (U''(\mathbf{u}))^{-1}$ is symmetric positive-definite. The integrability condition (1.4) suggests that $U'(\mathbf{u})\mathbf{f}'(\mathbf{u})$ is a gradient, which is equivalent to the symmetry of

$$(U'(\mathbf{u})\mathbf{f}'(\mathbf{u}))' = U''(\mathbf{u})\mathbf{f}'(\mathbf{u}) + U'(\mathbf{u})\mathbf{f}''(\mathbf{u}).$$

The second term, being a linear combination of symmetric matrices $(f^i)''(\mathbf{u}), 1 \leq i \leq p$, is also symmetric. Hence $U''(\mathbf{u})\mathbf{f}'(\mathbf{u})$ is symmetric. Since

$$\mathbf{g}'(\mathbf{v}) = \mathbf{f}'(\mathbf{u})\mathbf{u}'(\mathbf{v}) = \mathbf{f}'(\mathbf{u})(U''(\mathbf{u}))^{-1} = (U''(\mathbf{u}))^{-1}(U''(\mathbf{u})\mathbf{f}'(\mathbf{u}))(U''(\mathbf{u}))^{-1}$$

is congruent with $U''(\mathbf{u})\mathbf{f}'(\mathbf{u})$, we prove the symmetry of $\mathbf{g}'(\mathbf{v})$. □

Remark 1.1. Moreover, $\mathbf{f}'(\mathbf{u}) = \mathbf{g}'(\mathbf{v})\mathbf{v}'(\mathbf{u})$ is similar to $\mathbf{v}'(\mathbf{u})^{\frac{1}{2}}\mathbf{g}'(\mathbf{v})\mathbf{v}'(\mathbf{u})^{\frac{1}{2}}$, which is another symmetric matrix. Therefore all eigenvalues of $\mathbf{f}'(\mathbf{u})$ are real and $\mathbf{f}'(\mathbf{u})$ is diagonalizable. In other words, existence of entropy function implies hyperbolicity.

Now since $\mathbf{u}'(\mathbf{v})$ and $\mathbf{g}'(\mathbf{v})$ are both symmetric, there exist functions $\phi(\mathbf{v})$ and $\psi(\mathbf{v})$, called potential function and potential flux, such that

$$\phi'(\mathbf{v}) = \mathbf{u}(\mathbf{v})^T, \quad \psi'(\mathbf{v}) = \mathbf{g}(\mathbf{v})^T \tag{1.6}$$

It is easy to verify that

$$\phi(\mathbf{v}) = \mathbf{u}(\mathbf{v})^T\mathbf{v} - U(\mathbf{u}(\mathbf{v})), \quad \psi(\mathbf{v}) = \mathbf{g}(\mathbf{v})^T\mathbf{v} - F(\mathbf{u}(\mathbf{v})) \tag{1.7}$$

1.2 Existence and uniqueness of entropy solution

Given entropy function U, in smooth regions, the integrability condition (1.4) leads to an extra conservation law for U

$$\frac{\partial U(\mathbf{u})}{\partial t} + \frac{\partial F(\mathbf{u})}{\partial x} = 0. \tag{1.8}$$

However, at shock waves, we require the entropy to dissipate, which leads to the following definition of entropy solution.

Definition 1.2. *A weak solution* $\mathbf{u}$ *of* (1.1) *is called an entropy solution if for all possible entropy functions* U*, we have*

$$\frac{\partial U(\mathbf{u})}{\partial t} + \frac{\partial F(\mathbf{u})}{\partial x} \leq 0, \tag{1.9}$$

in the sense of distribution.

Formally integrating the entropy condition (1.9) in space, and assuming $\mathbf{u}$ is compactly supported, we obtain the bound

$$\frac{d}{dt} \int_{\mathbb{R}} U(\mathbf{u}) dx \leq 0. \tag{1.10}$$

That is, the total entropy is non-increasing with respect to time. If we further assume that U is uniformly convex, the above bound indeed implies an *a priori* L^2 bound of the entropy solution [49].

The existence of entropy solution follows from the limit of vanishing viscosity approximations. Consider the viscous perturbation of (1.1)

$$\frac{\partial \mathbf{u}_\varepsilon}{\partial t} + \frac{\partial \mathbf{f}(\mathbf{u}_\varepsilon)}{\partial x} = \varepsilon \frac{\partial^2 \mathbf{u}_\varepsilon}{\partial x^2}, \quad \varepsilon > 0. \tag{1.11}$$

By applying $U'(\mathbf{u}_\varepsilon)$ to (1.11),

$$\frac{\partial U(\mathbf{u}_\varepsilon)}{\partial t} + \frac{\partial F(\mathbf{u}_\varepsilon)}{\partial x} = \varepsilon\Big(\frac{\partial^2 U(\mathbf{u}_\varepsilon)}{\partial x^2} - \frac{\partial \mathbf{u}_\varepsilon^T}{\partial x} U''(\mathbf{u}_\varepsilon)\frac{\partial \mathbf{u}_\varepsilon}{\partial x}\Big) \leq \varepsilon \frac{\partial^2 U(\mathbf{u}_\varepsilon)}{\partial x^2}.$$

Sending $\varepsilon \to 0^+$ we recover the entropy condition (1.9). For some physical problems (e.g. compressible Navier-Stokes equations), it is necessary to look at the more general form of viscous perturbation

$$\frac{\partial \mathbf{u}_\varepsilon}{\partial t} + \frac{\partial \mathbf{f}(\mathbf{u}_\varepsilon)}{\partial x} = \varepsilon \frac{\partial}{\partial x}\Big(C(\mathbf{v}_\varepsilon)\frac{\partial \mathbf{v}_\varepsilon}{\partial x}\Big), \tag{1.12}$$

where $\mathbf{v}_\varepsilon := \mathbf{v}(\mathbf{u}_\varepsilon)$ and $C(\mathbf{v}_\varepsilon)$ is a symmetric semi-positive-definite $p \times p$ matrix. Applying $U'(\mathbf{u}_\varepsilon) = \mathbf{v}_\varepsilon^T$ to (1.12) gives us

$$\frac{\partial U(\mathbf{u}_\varepsilon)}{\partial t} + \frac{\partial F(\mathbf{u}_\varepsilon)}{\partial x} = \varepsilon\Big(\frac{\partial}{\partial x}\Big(\mathbf{v}_\varepsilon^T C(\mathbf{v}_\varepsilon)\frac{\partial \mathbf{v}_\varepsilon}{\partial x}\Big) - \frac{\partial \mathbf{v}_\varepsilon^T}{\partial x} C(\mathbf{v}_\varepsilon)\frac{\partial \mathbf{v}_\varepsilon}{\partial x}\Big) \leq \varepsilon \frac{\partial}{\partial x}\Big(\mathbf{v}_\varepsilon^T C(\mathbf{v}_\varepsilon)\frac{\partial \mathbf{v}_\varepsilon}{\partial x}\Big).$$

Then the vanishing viscosity approach works as well. Such heuristic can be made rigorous under smoothness and compactness assumptions. Proof of the following theorem is provided in [39].

Theorem 1.2. *Let $\{\mathbf{u}_\varepsilon\}$ be a sequence of sufficiently smooth solutions of (1.11) such that $\{\mathbf{u}_\varepsilon\}$ converges boundedly and a.e. to some function $\mathbf{u}$ as $\varepsilon \to 0^+$. Then $\mathbf{u}$ is an entropy solution of (1.1).*

Conditions of uniform boundedness and a.e. convergence are satisfied by the scalar case, so that the existence of entropy solution is established. In fact, $\{u_\varepsilon\}$ is bounded in both $L^\infty([0,\infty)\times\mathbb{R})$ and $W^{1,1}_{\text{loc}}([0,\infty)\times\mathbb{R})$, which allows us to extract an a.e convergent subsequence approaching an entropy solution. For general systems, we may only have an L^∞ bound, and additional assumptions are required. Glimm [38] proved the existence result for systems using his random choice method, under the assumption of initial data with sufficiently small total variation. Alternative proof relying on vanishing viscocity approach was revealed by Bianchini and Bressan [4].

Entropy solution is unique for scalar conservation laws, due to the abundance of entropy functions. Kruzhkov [63] proved the uniqueness result using the following family of entropy function - entropy flux pairs,

$$U(u) = |u-k|, \quad F(u) = \operatorname{sgn}(u-k)(f(u)-f(k)), \quad k\in\mathbb{R}. \tag{1.13}$$

We have obtained the well-posedness of entropy solution for scalar conservation laws as a result of existence and uniqueness. In the following theorem, we see that the entropy solution actually behaves well in many aspects [39].

Theorem 1.3. *If $u(0,\cdot)\in L^\infty(\mathbb{R})$, then (1.3) has a unique entropy solution $u\in L^\infty((0,\infty]\times\mathbb{R})$. The solution satisfies*

1. *Maximum principle: $\|u(t,\cdot)\|_{L^\infty}\leq\|u(0,\cdot)\|_{L^\infty}$ for almost all $t\geq 0$.*
2. *Order preservation: if u and v are both entropy solutions, then*

$$u(0,\cdot)\geq v(0,\cdot)\ a.e. \ \Rightarrow\ u(t,\cdot)\geq v(t,\cdot)\ a.e.$$

3. *L^1 contraction: if u and v are both entropy solutions, such that $u(0,\cdot)$ and*

$v(0,\cdot)$ belong to $L^1(\mathbb{R})$, then

$$\|u(t,\cdot)-v(t,\cdot)\|_{L^1} \leq \|u(0,\cdot)-v(0,\cdot)\|_{L^1} \text{ for almost all } t \geq 0.$$

4. *Total variation diminishing (TVD): if we assume $u(0,\cdot)$ has finite total variation, then*

$$TV(u(t,\cdot)) \leq TV(u(0,\cdot)) \text{ for almost all } t \geq 0,$$

where TV is the abbreviation for total variation.

If we restrict ourselves to uniformly convex flux functions, then a single strictly convex entropy function is sufficient to determine the entropy solution [84].

Theorem 1.4. *Consider the scalar conservation law (1.3). Assume that f is uniformly convex. If $u \in L^\infty([0,\infty)\times\mathbb{R})$ is a weak solution satisfying the entropy condition (1.9) with respect to a strictly convex entropy pair (U,F), then u is the entropy solution.*

For systems, we may not have enough entropy functions, and the question of uniqueness is very challenging except for some special cases. We refer the readers to [23, 40] and the references therein for more details on the theory of entropy solutions.

Before proceeding to numerical methods, we present some examples of hyperbolic conservation laws, together with their entropy function - entropy flux pairs and potential function - potential flux pairs.

Example 1.2.1. The linear symmetric system is of the form

$$\frac{\partial \mathbf{u}}{\partial t} + \frac{\partial (A\mathbf{u})}{\partial x} = 0, \tag{1.14}$$

where A is a constant symmetric matrix. The standard energy $U = \frac{1}{2}\mathbf{u}^T\mathbf{u}$ serves as an entropy function. Then $\mathbf{v} = \mathbf{u}$ and

$$F = \frac{1}{2}\mathbf{u}^T A\mathbf{u}, \quad \phi = \frac{1}{2}\mathbf{u}^T\mathbf{u}, \quad \psi = \frac{1}{2}\mathbf{u}^T A\mathbf{u}. \tag{1.15}$$

Example 1.2.2. A prototype of scalar conservation laws is the inviscid Burgers equation

$$\frac{\partial u}{\partial t} + \frac{\partial(u^2/2)}{\partial x} = 0, \tag{1.16}$$

equipped with the square entropy $U = \frac{1}{2}u^2$. Then $v = u$ and

$$F = \frac{1}{3}u^3, \quad \phi = \frac{1}{2}u^2, \quad \psi = \frac{1}{6}u^3. \tag{1.17}$$

Here f is uniformly convex and U is strictly convex. By Theorem 1.4, if a weak solution satisfies the entropy condition with respect to (U, F), it is the entropy solution.

Example 1.2.3. The shallow water equations model water flows with a free surface under the influence of gravity. The governing equations (with flat bottom) are

$$\frac{\partial}{\partial t}\begin{bmatrix} h \\ hw \end{bmatrix} + \frac{\partial}{\partial x}\begin{bmatrix} hw \\ hw^2 + \frac{1}{2}gh^2 \end{bmatrix} = 0. \tag{1.18}$$

Here h and w are the water depth and velocity, and g stands for the gravity acceleration constant. In the absence of dry bed, the water depth is always positive and

$$\Omega = \{\mathbf{u} \in \mathbb{R}^2 : h > 0\}. \tag{1.19}$$

The total (kinetic and potential) energy $U = \frac{1}{2}hw^2 + \frac{1}{2}gh^2$ is a convex function of

$\mathbf{u} \in \Omega$ and serves as an entropy function with

$$\mathbf{v} = \begin{bmatrix} gh - \frac{1}{2}w^2 \\ \\ w \end{bmatrix}, \quad F = \frac{1}{2}hw^3 + gh^2w, \quad \phi = \frac{1}{2}gh^2, \quad \psi = \frac{1}{2}gh^2w. \tag{1.20}$$

Example 1.2.4. The Euler equations of gas dynamics are

$$\frac{\partial}{\partial t}\begin{bmatrix} \rho \\ \\ \rho w \\ \\ E \end{bmatrix} + \frac{\partial}{\partial x}\begin{bmatrix} \rho w \\ \\ \rho w^2 + p \\ \\ w(E+p) \end{bmatrix} = 0. \tag{1.21}$$

Here ρ, w and p are the density, velocity and pressure of the gas. E is the total energy. In the case of polytropic ideal gas, the equation of state is

$$E = \frac{1}{2}\rho w^2 + \frac{p}{\gamma - 1}, \tag{1.22}$$

where γ is ratio of specific heats. $\gamma = 5/3$ for monatomic gas and $\gamma = 7/5$ corresponds to diatomic molecules. Assume that there is no vacuum. Then density and pressure need to be positive and

$$\Omega = \{\mathbf{u} \in \mathbb{R}^3 : \rho > 0, p > 0\} = \{\mathbf{u} \in \mathbb{R}^3 : \rho > 0, (\gamma - 1)(E - \frac{(\rho w)^2}{2\rho}) > 0\}. \tag{1.23}$$

We can verify that Ω is a convex set and (1.21) is hyperbolic in Ω. The physical specific entropy is $s = \log(p\rho^{-\gamma})$. Harten [46] proved that there exists a family of entropy pairs that are related to s:

$$U = -\rho h(s), \quad F = -\rho w h(s). \tag{1.24}$$

To make U convex, h can be any function such that

$$h'(s) - \gamma h''(s) > 0, \quad h'(s) > 0. \tag{1.25}$$

However, by regarding Euler equations as the vanishing viscosity limit of compressible Navier-Stokes equations, we need to make sure that $C(\mathbf{v}_\varepsilon)$ in (1.12) is symmetric semi-positive-definite. The only choice of entropy pair is (see [57])

$$U = -\frac{\rho s}{\gamma - 1}, \quad F = -\frac{\rho w s}{\gamma - 1}. \tag{1.26}$$

The corresponding entropy variables and potential function-potential flux pair are

$$\mathbf{v} = \begin{bmatrix} \frac{\gamma - s}{\gamma - 1} - \frac{\rho w^2}{2p} \\ \frac{\rho w}{p} \\ -\frac{\rho}{p} \end{bmatrix}, \quad \phi = \rho, \quad \psi = \rho w. \tag{1.27}$$

1.3 Review of first order methods

In this section, we start to look into the numerical aspects of one-dimensional conservation laws. We will mostly conduct semi-discrete analysis, i.e., we investigate the system of ODEs derived via method of lines principle, and temporal discretization is not taken into account. For spatial discretization, suppose that we have a computational interval $\Gamma = [x_L, x_R]$, equipped with periodic boundary condition, and divided into cells $\{I_i\}_{i=1}^N$ such that

$$x_L = x_{1/2} < x_{3/2} < \cdots < x_{N+1/2} = x_R, \quad I_i = [x_{i-1/2}, x_{1+1/2}], \quad \Delta x_i := x_{i+1/2} - x_{i-1/2},$$

and $h := \max_{1\leq i\leq N}$ is the characteristic mesh size. The first order (finite volume) method simulates the evolution of cell averages. Let $\mathbf{u}_i(t) := \frac{1}{\Delta x_i}\int_{I_i} u(t,x)dx$. Integrating (1.1) in I_i, we have the integral form

$$\frac{d\mathbf{u}_i(t)}{dt} + \frac{1}{\Delta x_i}(\mathbf{f}(\mathbf{u}(t,x_{i+1/2})) - \mathbf{f}(\mathbf{u}(t,x_{i-1/2}))) = 0. \tag{1.28}$$

Then the first order scheme reads

$$\frac{d\mathbf{u}_i}{dt} + \frac{1}{\Delta x_i}(\widehat{\mathbf{f}}_{i+1/2} - \widehat{\mathbf{f}}_{i-1/2}) = 0, \tag{1.29}$$

where $\widehat{\mathbf{f}}_{i+1/2} = \widehat{\mathbf{f}}(\mathbf{u}_i, \mathbf{u}_{i+1})$, and $\widehat{\mathbf{f}}(\mathbf{u}_L, \mathbf{u}_R)$ is some consistent two-point numerical flux function such that

$$\widehat{\mathbf{f}}(\mathbf{u},\mathbf{u}) = \mathbf{f}(\mathbf{u}). \tag{1.30}$$

Scheme (1.29) is conservative, in the sense that if we define $\mathbf{u}_h(t,x) := \sum_{i=1}^N \mathbf{u}_i(t)1_{I_i}$,

$$\frac{d}{dt}\int_\Gamma \mathbf{u}_h(t,x)dx = \frac{d}{dt}\Big(\sum_{i=1}^N \Delta x_i \mathbf{u}_i\Big) = \sum_{i=1}^N(\widehat{\mathbf{f}}_{i-1/2} - \widehat{\mathbf{f}}_{i+1/2}) = 0.$$

Entropy stability of (1.29) is thoroughly studied by Tadmor in [95, 96]. For an entropy function U, the rate of change of the total entropy is

$$\frac{d}{dt}\int_\Gamma U(\mathbf{u}_h(t,x))dx = \frac{d}{dt}\Big(\sum_{i=1}^N \Delta x_i U_i\Big) = \sum_{i=1}^N \mathbf{v}_i^T(\widehat{\mathbf{f}}_{i-1/2} - \widehat{\mathbf{f}}_{i+1/2}) = \sum_{i=1}^N (\mathbf{v}_{i+1} - \mathbf{v}_i)^T\widehat{\mathbf{f}}_{i+1/2}, \tag{1.31}$$

where we use the short hand notation $U_i := U(\mathbf{u}_i)$ and $\mathbf{v}_i := \mathbf{v}(\mathbf{u}_i)$. This motivates us to define the concepts of entropy conservative flux and entropy stable flux.

Definition 1.3. *A bivariate numerical flux function $\mathbf{f}_S(\mathbf{u}_L, \mathbf{u}_R)$ is called entropy conservative with respect to some entropy function U if it is consistent, symmetric,* *and*

$$(\mathbf{v}_R - \mathbf{v}_L)^T \mathbf{f}_S(\mathbf{u}_L, \mathbf{u}_R) = \psi_R - \psi_L, \tag{1.32}$$

where we again set $\mathbf{v}_{L,R} := \mathbf{v}(\mathbf{u}_{L,R})$, $\psi_{L,R} := \psi(\mathbf{v}_{L,R})$, and ψ is the potential flux defined in (1.7).

Definition 1.4. *A bivariate numerical flux function $\widehat{\mathbf{f}}(\mathbf{u}_L, \mathbf{u}_R)$ is called entropy stable with respect to some entropy function U if it is consistent, and*

$$(\mathbf{v}_R - \mathbf{v}_L)^T \widehat{\mathbf{f}}(\mathbf{u}_L, \mathbf{u}_R) \leq (\psi_R - \psi_L). \tag{1.33}$$

Recall (1.31). If $\widehat{\mathbf{f}}$ is entropy stable,

$$\frac{d}{dt} \int_\Gamma U(\mathbf{u}_h(t,x))dx \leq \sum_{i=1}^{N} (\psi_{i+1} - \psi_i) = 0$$

We accordingly say that (1.29) is entropy stable with respect to U. In addition, we can define the numerical entropy flux

$$\widehat{F}(\mathbf{u}_L, \mathbf{u}_R) := \frac{1}{2}(\mathbf{v}_L + \mathbf{v}_R)^T \widehat{\mathbf{f}}(\mathbf{u}_L, \mathbf{u}_R) - \frac{1}{2}(\psi_L + \psi_R). \tag{1.34}$$

It is consistent as $\widehat{F}(\mathbf{u}, \mathbf{u}) = \mathbf{v}(\mathbf{u})^T \mathbf{f}(\mathbf{u}) - \psi(\mathbf{v}) = F(\mathbf{u})$. Let $\widehat{F}_{i+1/2} = \widehat{F}(\mathbf{u}_i, \mathbf{u}_{i+1})$. We get the cell entropy inequality

$$\begin{aligned} &\frac{dU_i}{dt} + \frac{1}{\Delta x_i}(\widehat{F}_{i+1/2} - \widehat{F}_{i-1/2}) \\ =&\frac{1}{2}((\mathbf{v}_{i+1} - \mathbf{v}_i)^T \widehat{\mathbf{f}}_{i+1/2} + (\mathbf{v}_i - \mathbf{v}_{i-1})^T \widehat{\mathbf{f}}_{i-1/2} - (\psi_{i+1} - \psi_{i-1})) \leq 0, \end{aligned} \tag{1.35}$$

which corresponds to the integral form of entropy condition (1.9). Similarly, if $\widehat{\mathbf{f}}$ is entropy conservative, the total entropy does not change and the scheme is said to be entropy conservative. All inequalities above become equalities.

The importance of conservation and entropy stability resides in the well-known Lax-Wendroff [64] theorem. Proof is a direct application of dominated convergence theorem. Notice its evident resemblance with Theorem 1.2. Actually taking the limit of numerical methods is another way to prove the existence of entropy solution.

Theorem 1.5. *If $\{\mathbf{u}_h(t,x)\}$ converges boundedly and a.e. to some function $u(t,x)$ as $h \to 0^+$, then $\mathbf{u}$ is a weak solution of (1.1). Furthermore, if $\widehat{\mathbf{f}}$ is entropy stable with respect to all entropy functions, $\mathbf{u}$ is an entropy solution.*

In the scalar case, the entropy conservative flux is uniquely determined

$$f_S(u_L, u_R) = \begin{cases} \frac{\psi_R - \psi_L}{v_R - v_L} & u_L \neq u_R \\ f(u_L) & u_L = u_R \end{cases}. \tag{1.36}$$

For systems, (1.32) is underdetermined and $\mathbf{f}_S(\mathbf{u}_L, \mathbf{u}_R)$ is not unique. A generic choice of entropy conservative flux is the following path integration [96].

$$\mathbf{f}_S(\mathbf{u}_L, \mathbf{u}_R) = \int_0^1 \mathbf{g}(\mathbf{v}_L + \lambda(\mathbf{v}_R - \mathbf{v}_L))d\lambda, \tag{1.37}$$

which may not have an explicit formula and can be computationally expensive. Fortunately, for many systems we are able to derive explicit entropy conservative fluxes that are easy to compute. Let us revisit the examples in Section 1.2.

Example 1.3.1. For linear symmetric system (1.14), the entropy conservative flux is simply the arithmetic mean

$$\mathbf{f}_S(\mathbf{u}_L, \mathbf{u}_R) = \frac{1}{2}(A\mathbf{u}_L + A\mathbf{u}_R). \tag{1.38}$$

Example 1.3.2. For Burgers equation (1.16),

$$f_S(u_L, u_R) = \frac{1}{6}(u_L^2 + u_L u_R + u_R^2). \tag{1.39}$$

is conservative with respect to the square entropy function.

Example 1.3.3. For shallow water equations (1.18), an explicit entropy conservative flux is

$$\mathbf{f}_S(\mathbf{u}_L, \mathbf{u}_R) = \begin{bmatrix} \frac{1}{2}(h_L w_L + h_R w_R) \\ \frac{1}{4}(h_L w_L + h_R w_R)(w_L + w_R) + \frac{1}{2} g h_L h_R \end{bmatrix}. \tag{1.40}$$

Example 1.3.4. For Euler equations (1.21), Ismail and Roe [58] suggested the following affordable entropy conservative flux

$$\begin{aligned} f_S^1 &= \overline{z^2}(\overline{z^3})^{\log}, \\ f_S^2 &= \frac{\overline{z^3}}{\overline{z^1}} + \frac{\overline{z^2}}{\overline{z^1}} f_S^1, \\ f_S^3 &= \frac{1}{2}\frac{\overline{z^2}}{\overline{z^1}}\left(\frac{\gamma+1}{\gamma-1}\frac{(\overline{z^3})^{\log}}{(\overline{z^1})^{\log}} + f_S^2\right), \end{aligned} \tag{1.41}$$

where

$$\mathbf{z} := \begin{bmatrix} z^1 \\ z^2 \\ z^3 \end{bmatrix} = \sqrt{\frac{\rho}{p}} \begin{bmatrix} 1 \\ w \\ p \end{bmatrix}.$$

$\overline{z^s}$ and $(\overline{z^s})^{\log}$ are the arithmetic mean and the logarithmic mean

$$\overline{z^s} := \frac{1}{2}(z_L^s + z_R^s), \quad (\overline{z^s})^{\log} := \frac{z_R^s - z_L^s}{\log z_R^s - \log z_L^s}, \quad s = 1, 2, 3.$$

Another entropy conservative flux, which also preserves kinetic energy, was recommended by Chandrashekar in [11]:

$$
\begin{aligned}
f_S^1 &= (\overline{\rho})^{\log}\overline{w},\\
f_S^2 &= \frac{\overline{\rho}}{2\overline{\overline{\beta}}} + \overline{w} f_S^1,\\
f_S^3 &= \Big(\frac{1}{2(\gamma-1)(\overline{\beta})^{\log}} - \frac{1}{2}\overline{w^2}\Big) f_S^1 + \overline{w} f_S^2,
\end{aligned}
\tag{1.42}
$$

where $\beta := \frac{\rho}{2p}$.

The construction of entropy stable fluxes can be divided into two categories. In [58, 11, 7, 31], the authors build $\widehat{\mathbf{f}}$ by adding some numerical dissipation operators, of Lax-Friedrichs type or Roe type, to the entropy conservative flux, so that the amount of entropy dissipation can be precisely determined. On the other hand, it has been known for decades that the widely used upwind numerical fluxes, including monotone fluxes for scalar conservation laws and Godunov-type fluxes for general systems, are entropy stable. Here we will follow the latter approach because of other desirable properties of upwind fluxes (e.g. maximum principle and TVD).

Most popular numerical fluxes rely on Riemann solvers, which exactly compute or approximate the solution of the Riemann problem

$$
\begin{cases}
\frac{\partial \mathbf{u}}{\partial t} + \frac{\partial \mathbf{f}(\mathbf{u})}{\partial x} = 0\\
\mathbf{u}(x,0) = \begin{cases} \mathbf{u}_L & x \le 0\\ \mathbf{u}_R & x > 0 \end{cases}
\end{cases}
.
\tag{1.43}
$$

The solution of the Riemann problem is self-similar. We assume that our Riemann solver also has self-similar structure and is denoted by $\mathbf{q}(x/t; \mathbf{u}_L, \mathbf{u}_R)$. Let λ_L and

λ_R be the leftmost and rightmost wave speed such that

$$\mathbf{q}(r;\mathbf{u}_L,\mathbf{u}_R) = \begin{cases} \mathbf{u}_L & r \leq \lambda_L \\ \\ \mathbf{u}_R & r \geq \lambda_R \end{cases}. \tag{1.44}$$

The Riemann solver should maintain conservation. For any $S_L \leq \min\{\lambda_L, 0\}$ and $S_R \geq \max\{\lambda_R, 0\}$, integrating along the rectangle $[S_L, S_R] \times [0,1]$ yields

$$\int_{S_L}^{S_R} \mathbf{q}(r;\mathbf{u}_L,\mathbf{u}_R)dr - (S_R\mathbf{u}_R - S_L\mathbf{u}_L) + (\mathbf{f}_R - \mathbf{f}_L) = 0. \tag{1.45}$$

Definition 1.5. *Let $\mathbf{q}(x/t;\mathbf{u}_L,\mathbf{u}_R)$ be a self-similar Riemann solver that satisfies* (1.45)*. $\widehat{\mathbf{u}}(\mathbf{u}_L,\mathbf{u}_R)$ is called a Godunov-type flux [49] if*

$$\widehat{\mathbf{f}}(\mathbf{u}_L,\mathbf{u}_R) = \mathbf{f}_R + \int_0^{S_R} \mathbf{q}(r;\mathbf{u}_L,\mathbf{u}_R)dr - S_R\mathbf{u}_R = \mathbf{f}_L - \int_{S_L}^0 \mathbf{q}(r;\mathbf{u}_L,\mathbf{u}_R)dr - S_L\mathbf{u}_L. \tag{1.46}$$

The idea follows from integration along $[0, S_R] \times [0,1]$ or $[S_L, 0] \times [0,1]$.

Theorem 1.6. *For an entropy function U, assume that the Riemann solver also satisfies the entropy condition such that for any $S_L \leq \min\{\lambda_L, 0\}$ and $S_R \geq \max\{\lambda_R, 0\}$,*

$$\int_{S_L}^{S_R} U(\mathbf{q}(r;\mathbf{u}_L,\mathbf{u}_R))dr - (S_RU_R - S_LU_L) + F_R - F_L \leq 0 \tag{1.47}$$

Then the corresponding Godunov-type flux is entropy stable with respect to U.

Proof. By (1.46) and Jensen's inequality,

$$\int_0^{S_R} U(\mathbf{q}(r;\mathbf{u}_L,\mathbf{u}_R))dr \geq S_RU(\frac{1}{S_R}\int_0^{S_R} \mathbf{q}(r;\mathbf{u}_L,\mathbf{u}_R)dr) = S_RU(\mathbf{u}_R + \frac{1}{S_R}(\widehat{\mathbf{f}}(\mathbf{u}_L,\mathbf{u}_R) - \mathbf{f}_R)),$$

$$\int_{S_L}^0 U(\mathbf{q}(r;\mathbf{u}_L,\mathbf{u}_R))dr \geq -S_LU(-\frac{1}{S_L}\int_{S_L}^0 \mathbf{q}(r;\mathbf{u}_L,\mathbf{u}_R)dr) = -S_LU(\mathbf{u}_L + \frac{1}{S_L}(\widehat{\mathbf{f}}(\mathbf{u}_L,\mathbf{u}_R) - \mathbf{f}_L)).$$

Summing them up and applying (1.47) gives

$$S_R(U(\mathbf{u}_R+\frac{1}{S_R}(\widehat{\mathbf{f}}(\mathbf{u}_L,\mathbf{u}_R)-\mathbf{f}_R))-U_R)-S_L(U(\mathbf{u}_L+\frac{1}{S_L}(\widehat{\mathbf{f}}(\mathbf{u}_L,\mathbf{u}_R)-\mathbf{f}_L))-U_L)+F_R-F_L \leq 0.$$

We send $S_R \to \infty$ and $S_L \to -\infty$. The first term converges to $\mathbf{v}_R^T(\widehat{\mathbf{f}}(\mathbf{u}_L,\mathbf{u}_R)-\mathbf{f}_R)$ and the second term converges to $\mathbf{v}_L^T(\widehat{\mathbf{f}}(\mathbf{u}_L,\mathbf{u}_R)-\mathbf{f}_L)$. The inequality above simplifies to

$$\mathbf{v}_R^T(\widehat{\mathbf{f}}(\mathbf{u}_L,\mathbf{u}_R)-\mathbf{f}_R)-\mathbf{v}_L^T(\widehat{\mathbf{f}}(\mathbf{u}_L,\mathbf{u}_R)-\mathbf{f}_L)+F_R-F_L = (\mathbf{v}_R-\mathbf{v}_L)^T\widehat{\mathbf{f}}(\mathbf{u}_L,\mathbf{u}_R)-(\psi_R-\psi_L) \leq 0,$$

which is exactly the condition (1.33). □

Example 1.3.5 (Godunov flux)**.** The Riemann problem can be solved exactly for scalar problems, as well as shallow water equations and Euler equations. The resulting numerical flux is called Godunov flux. Since the exact solutions satisfy all entropy conditions, Godunov flux is entropy stable with respect to all entropy functions.

Example 1.3.6 (HLL flux)**.** The computation of exact Riemann solver often requires several Newton-Raphson iteration steps. Practically we resort to approximate Riemann solvers to reduce computational cost. A commonly used approximate Riemann solver is the HLL Riemann solver [46], which assumes a constant middle state. We first prescribe values of λ_L and λ_R. Then

$$\mathbf{q}(r;\mathbf{u}_L,\mathbf{u}_R) = \begin{cases} \mathbf{u}_L & r \leq \lambda_L \\ \mathbf{u}_R & r \geq \lambda_R \\ \frac{\lambda_R\mathbf{u}_R-\lambda_L\mathbf{u}_L-(\mathbf{f}_R-\mathbf{f}_L)}{\lambda_R-\lambda_L} & \lambda_L < r < \lambda_R \end{cases}. \tag{1.48}$$

Inserting (1.46), we obtain the HLL flux

$$\widehat{\mathbf{f}}(\mathbf{u}_L, \mathbf{u}_R) = \begin{cases} \mathbf{f}_L & \lambda_L \geq 0 \\ \mathbf{f}_R & \lambda_R \leq 0 \\ \frac{\lambda_R \mathbf{f}_L - \lambda_L \mathbf{f}_R + \lambda_L \lambda_R (\mathbf{u}_R - \mathbf{u}_L)}{\lambda_R - \lambda_L} & \lambda_L < 0 < \lambda_R \end{cases} . \tag{1.49}$$

The HLL flux in entropy stable provided we approximate λ_L and λ_R properly.

Corollary 1.1. *If λ_L is not larger than the true leftmost wave speed and λ_R is not smaller than the true rightmost wave speed, the HLL flux is entropy stable with respect to all entropy functions.*

Proof. It suffices to prove that for all entropy functions, (1.47) is satisfied by the HLL Riemann solver. Since the approximate wave fan is larger than the true wave fan and the middle state is constant, the HLL Riemann solver is simply an average of the exact Riemann solver. Another application of Jensen's inequality completes the proof. □

Example 1.3.7 (Lax-Friedrichs flux)**.** The local Lax-Friedrichs flux is written as

$$\widehat{\mathbf{f}}(\mathbf{u}_L, \mathbf{u}_R) = \frac{1}{2}(\mathbf{f}_L + \mathbf{f}_R) - \frac{\lambda}{2}(\mathbf{u}_R - \mathbf{u}_L), \tag{1.50}$$

for some $\lambda > 0$. It is a special case of HLL flux with $\lambda_L = -\lambda$ and $\lambda_R = \lambda$. Hence local Lax-Friedrichs flux is entropy stable with respect to all entropy functions if λ is not smaller than the true maximum wave speed (in terms of absolute vale).

The computation of λ_L and λ_R is, however, not trivial. Simplistic approximation usually fails to bound the true wave speeds. Toro [98, 99] recommends the two-rarefaction approximation, and Guermond and Popov [44] prove that the two-rarefaction approximated wave speeds indeed provide the correct bounds for Euler equations with $1 \leq \gamma \leq 5/3$. We can also prove the similar result for shallow water equations. The details of two-rarefaction approximation will be given in Appendix A.

In the scalar case, it is more convenient to examine the monotonicty of flux.

Definition 1.6. *A consistent numerical flux function $\widehat{f}(u_L, u_R)$ is called monotone if it is a a non-decreasing function of its first argument and a non-increasing function of its second argument.*

Theorem 1.7. *Monotone fluxes are entropy stable with respect to all entropy functions.*

Proof. For any entropy function U and its entropy variable v, since $\psi'(v) = g(v) = f(u)$, there exists $\widetilde{v}$ between v_L and v_R such that

$$\psi_R - \psi_L = (v_R - v_L)g(\widetilde{v}) = (v_R - v_L)f(u(\widetilde{v})).$$

Due to the convexity of U, $u(v)$ is an increasing function, and $u(\widetilde{v})$ is also between u_L and u_R. By the monotonicity of $\widehat{f}$ we have

$$(u_R - u_L)(\widehat{f}(u_L, u_R) - f(u(\widetilde{v}))) \leq 0. \tag{1.51}$$

Consequently

$$(\psi_R - \psi_L) - \widehat{f}(u_L, u_R)(v_R - v_L) = (v_R - v_L)(f(u(\widetilde{v})) - \widehat{f}(u_L, u_R)) \geq 0.$$

We remark that (1.51) is exactly the characterization of the E-flux by Osher [82]. □

Example 1.3.8 (Godunov flux, revisited)**.** The Godunov flux in the scalar case has an explicit expression

$$\widehat{f}(u_L, u_R) = \begin{cases} \min\limits_{u \in [u_L, u_R]} f(u) & \text{if } u_L \le u_R \\ \\ \max\limits_{u \in [u_R, u_L]} f(u) & \text{if } u_R \le u_L \end{cases}. \tag{1.52}$$

It is obviously monotone.

Example 1.3.9 (Lax-Friedrichs flux, revisited)**.** Recall the local Lax-Friedrichs flux

$$\widehat{f}(u_L, u_R) = \frac{1}{2}(f_L + f_R) - \frac{\lambda}{2}(u_R - u_L). \tag{1.53}$$

It is monotone provided that $\lambda \geq \max\{|f'(u_L)|, |f'(u_R)|\}$, i.e., not smaller than the maximum wave speed.

Example 1.3.10 (Engquist-Osher flux)**.** The numerical flux built by Eugquist and Osher [26] is

$$\widehat{f}(u_L, u_R) = f(0) + f^+(u_L) + f^-(u_R), \tag{1.54}$$

where

$$f^+(u) := \int_0^u \max\{f'(s), 0\} ds, \quad f^-(u) := \int_0^u \min\{f'(s), 0\} ds. \tag{1.55}$$

$\widehat{f}$ is monotone as f^+ is non-decreasing and f^- is non-increasing.

Besides entropy stability, numerical solutions of first order method (1.29) with monotone flux also share the discrete version of the well-posedness properties of the entropy solution in Theorem 1.3.

Theorem 1.8. *Suppose that $\{u_i(t)\}_{i=1}^N$ and $\{v_i(t)\}_{i=1}^N$ are both numerical solutions of the semi-discrete scheme (1.29) for some scalar conservation law, where $\widehat{f}$ is* *monotone and Lipschitz continuous of both arguments. Then we have the following properties:*

1. *Maximum principle:* $\max\limits_{1\leq i\leq N}\{u_i(t)\} \leq \max\limits_{1\leq i\leq N}\{u_i(0)\}$.
2. *Order preservation:* $u_i(0) \geq v_i(0), \forall 1 \leq i \leq N \quad \Rightarrow \quad u_i(t) \geq v_i(t), \forall 1 \leq i \leq N$
3. L^1 *contraction:* $\sum\limits_{i=1}^{N} \Delta x_i |u_i(t) - v_i(t)| \leq \sum\limits_{i=1}^{N} \Delta x_i |u_i(0) - v_i(0)|$.
4. *Total variation diminishing (TVD):* $\sum\limits_{i=1}^{N} |u_{i+1}(t) - u_i(t)| \leq \sum\limits_{i=1}^{N} |u_{i+1}(0) - u_i(0)|$.

We recheck the conditions in the Lax-Wendroff theorem. Maximum principle implies uniform boundedness, and TVD property leads to a a.e. convergent subsequence of $\{u_h(t,x)\}$. Since monotone flux is entropy stable with respect to all entropy functions, the subsequence converges to the entropy solution. Moreover, by the uniqueness of entropy solution and a sub-subsequence argument (every subsequence has a sub-subsequence approaching the entropy solution), the whole sequence $\{u_h(t,x)\}$ converges to the entropy solution. We obtain the "ultimate theorem" for first order methods.

Theorem 1.9. *Let* $u_h(t,x) = \sum_{i=1}^N u_i(t) 1_{I_i}$. *If* $\widehat{f}$ *is monotone and globally Lipschitz continuous of both arguments, then* $\{u_h(t,x)\}$ *converges a.e. to the unique entropy solution.*

Remark 1.2. All theorems in the chapter are still valid for the fully discrete first order scheme, where we use Euler forward time discretization (see e.g. Chapter 3 of [39]).

1.4 Review of classic DG method

Although first order methods have a bunch of desirable properties including provable entropy stability, the convergence rate is quite slow. There are mainly two directions towards high order methods for systems of conservation laws. In high order finite volume methods, we maintain cell averages, and the numerical flux depends on a wider stencil of cells with some reconstruction technique; while in discontinuous Galerkin (DG) methods, we keep locality and the two-point numerical flux, and evolve high order polynomials in cells. When designing high order methods, we hope to recover some (if not all) of the properties in Theorem 1.8 without affecting high order accuracy. People have developed various types of modifications, such as TVD/TVB limiters [88], bound-preseving limiter [112, 113], ENO method [47], and WENO method [90], to achieve that goal. However, in general entropy stability for all entropy functions can not be accomplished in high order schemes. It is shown in [48, 82] that both monotone fluxes and E-fluxes are at most first order accurate, and Osher and Tadmor proved [83] that E-flux is in fact necessary for stability with respect to all entropy functions. Therefore we have to make a compromise. In the literature, one usually seeks entropy stability with respect to a *single* entropy function. Let us remark that Bouchut, Bourdaris and Perthame [5] gave a second order scheme that satisfies all entropy inequalities. It does not contradict the argument by Osher and Tadmor since their scheme was not written in the standard conservative form.

In the realm of finite volume methods, a major result of entropy stable high order method is the *TeCNO* scheme, proposed by Fjordholm, Mishra and Tadmor [31] as a version of ENO schemes. The authors used the high order linear combinations of entropy conservative fluxes in [66], along with the sign property of ENO reconstruction [32]. The TeCNO scheme is only stable with respect to a given entropy function, as entropy conservative fluxes are specific to entropy functions. A second order generalization to higher dimensional unstructured meshes is presented in [87].

The story is similar for DG methods. Let us start with the classic DG method developed by Cockburn and Shu in their series of papers [17, 16, 15, 19]. We still assume periodic boundary condition and domain decomposition $\{I_i\}_{i=1}^N$. Given polynomial degree $k \geq 0$, we define the DG space of piecewise polynomials

$$\mathbf{V}_h^k = \{\mathbf{w}_h : \mathbf{w}_h|_{I_i} \in [\mathcal{P}^k(I_i)]^p, 1 \leq i \leq N\}. \tag{1.56}$$

We seek $\mathbf{u}_h \in \mathbf{V}_h^k$ such that for each $\mathbf{w}_h \in \mathbf{V}_h^k$ and $1 \leq i \leq N$,

$$\int_{I_i} \frac{\partial \mathbf{u}_h^T}{\partial t} \mathbf{w}_h dx - \int_{I_i} \mathbf{f}(\mathbf{u}_h)^T \frac{d\mathbf{w}_h}{dx} dx = -\widehat{\mathbf{f}}_{i+1/2}^T \mathbf{w}_h(x_{i+1/2}^-) + \widehat{\mathbf{f}}_{i-1/2}^T \mathbf{w}_h(x_{i-1/2}^+), \tag{1.57}$$

where

$$\widehat{\mathbf{f}}_{i+1/2} = \widehat{\mathbf{f}}(\mathbf{u}_h(x_{i+1/2}^-), \mathbf{u}_h(x_{i+1/2}^+)) \tag{1.58}$$

for some consistent two-point numerical flux function $\widehat{\mathbf{f}}$. (1.57) is usually called the *weak form* of DG method as it approximates the weak problem

$$\int_{\mathbb{R}} \frac{\partial \mathbf{u}(t,x)^T}{\partial t} \mathbf{w}(x) dx - \int_{\mathbb{R}} \mathbf{u}(t,x)^T \frac{d\mathbf{w}(x)}{dx} dx = 0, \tag{1.59}$$

for all smooth and compactly supported $\mathbf{w}$. In the case that $k = 0$, the DG space contains piecewise constants and (1.57) reduces to the first order method (1.29). The *strong form* of DG method is obtained after a simple integration by parts

$$\begin{aligned}\int_{I_i} \Big(\frac{\partial \mathbf{u}_h}{\partial t} + \frac{\mathbf{f}(\mathbf{u}_h)}{\partial x}\Big)^T \mathbf{w}_h dx = {} & (\mathbf{f}(\mathbf{u}_h(x_{i+1/2}^-)) - \widehat{\mathbf{f}}_{i+1/2})^T \mathbf{w}_h(x_{i+1/2}^-) \\ & - (\mathbf{f}(\mathbf{u}_h(x_{i-1/2}^+)) - \widehat{\mathbf{f}}_{i-1/2})^T \mathbf{w}_h(x_{i-1/2}^+),\end{aligned} \tag{1.60}$$

which corresponds to the equation (1.1) itself. The classic DG method is conservative (by taking $\mathbf{w}_h = 1$), high order accurate, and L^2 stable if we have a square entropy function, e.g. in scalar problems and linear symmetric systems.

Theorem 1.10. *If $U = \frac{1}{2}\mathbf{u}^T\mathbf{u}$ is an entropy function of (1.1), and $\widehat{\mathbf{f}}$ is entropy stable with respect to U, then the DG scheme (1.57) and (1.60) is L^2 stable in the sense that*

$$\frac{d}{dt}\int_\Gamma U(\mathbf{u}_h)dx = \frac{d}{dt}\Big(\frac{1}{2}\|\mathbf{u}_h\|_{L^2}^2\Big) \leq 0. \tag{1.61}$$

Proof. Since $U = \frac{1}{2}\mathbf{u}^T\mathbf{u}$, $\mathbf{v} = \mathbf{u}$, $\psi = \mathbf{u}^T\mathbf{f} - F$, and $\psi'(\mathbf{u}) = \mathbf{f}(\mathbf{u})$. We set $\mathbf{w}_h = \mathbf{u}_h$ in (1.57) and get

$$\begin{aligned}\frac{d}{dt}\Big(\frac{1}{2}\|\mathbf{u}_h\|_{L^2}^2\Big) &= \sum_{i=1}^N \int_{I_i} \frac{\partial \mathbf{u}_h^T}{\partial t}\mathbf{u}_h dx = \sum_{i=1}^N \Big(\int_{I_i} \mathbf{f}(\mathbf{u}_h)^T \frac{\partial \mathbf{u}_h}{\partial x} dx - \widehat{\mathbf{f}}_{i+1/2}^T \mathbf{u}_{i+1/2}^- + \widehat{\mathbf{f}}_{i-1/2}^T \mathbf{u}_{i-1/2}^+\Big)\\ &= \sum_{i=1}^N (\psi_{i+1/2}^- - \psi_{i-1/2}^+ - \widehat{\mathbf{f}}_{i+1/2}^T \mathbf{u}_{i+1/2}^- + \widehat{\mathbf{f}}_{i-1/2}^T \mathbf{u}_{i-1/2}^+)\\ &= \sum_{i=1}^N \Big(\widehat{\mathbf{f}}_{i+1/2}^T(\mathbf{u}_{i+1/2}^+ - \mathbf{u}_{i+1/2}^-) - (\psi_{i+1/2}^+ - \psi_{i+1/2}^-)\Big) \leq 0,\end{aligned}$$

where we use the short hand notation $\mathbf{u}_{i+1/2}^\pm := \mathbf{u}_h(x_{i+1/2}^\pm)$ and $\psi_{i+1/2}^\pm := \psi(\mathbf{u}_{i+1/2}^\pm)$. The last inequality results from the entropy stability of $\widehat{\mathbf{f}}$. □

Remark 1.3. In [60, 55], the L^2 stability is characterized by the cell entropy inequality

$$\frac{d}{dt}\int_{I_i} U(\mathbf{u}_h)dx + \widehat{F}_{i+1/2} - \widehat{F}_{i-1/2} \leq 0, \tag{1.62}$$

where $\widehat{F}_{i+1/2} := \frac{1}{2}(\mathbf{u}_{i+1/2}^- + \mathbf{u}_{i+1/2}^+)^T\widehat{\mathbf{f}}_{i+1/2} - \frac{1}{2}(\psi_{i+1/2}^- + \psi_{i+1/2}^+)$.

The stability result is limited to the square entropy function. For a general entropy U, the mapping $\mathbf{u} \mapsto \mathbf{v}$ is nonlinear, and $\mathbf{v}(\mathbf{u}_h)$ does not live in the piecewise polynomial space $\mathbf{V}_h^k$. We can not use $\mathbf{v}(\mathbf{u}_h)$ as the test function. One possible remedy is to approximate $\mathbf{v}$ directly. We evolve $\mathbf{v}_h \in \mathbf{V}_h^k$ such that for each $\mathbf{w}_h \in \mathbf{V}_h^k$ and $1 \leq i \leq N$,

$$\int_{I_i} \frac{\partial \mathbf{u}(\mathbf{v}_h)^T}{\partial t} \mathbf{w}_h dx - \int_{I_i} \mathbf{g}(\mathbf{v}_h)^T \frac{d\mathbf{w}_h}{dx} dx = -\widehat{\mathbf{f}}_{i+1/2}^T \mathbf{w}_h(x_{i+1/2}^-) + \widehat{\mathbf{f}}_{i-1/2}^T \mathbf{w}_h(x_{i-1/2}^+), \quad (1.63)$$

This approach, initiated by Hughes, Franca and Mallet [57], is entropy stable for any given entropy function. The proof is exactly the same as Theorem 1.10. It has the drawback that nonlinear solvers are required at each time step, even for explicit time discretization. Hence, people are in favor of space-time DG formulation [3, 53].

The entropy stable DG type method we are going to discuss does not incur nonlinear solvers. It is based on quadrature points and nodal formulation, so that we can perform nonlinear mapping freely. Actually, quadrature rules are necessary for the implementation of DG method. If the flux function $\mathbf{f}$ has a convoluted form (e.g. in Euler equations), it is costly or even impossible to evaluate the second integral in (1.57) exactly. There are two technical challenges related to the nodal form. We need discrete versions of integration by parts and chain rule, which are crucial to the proof of Theorem 1.10. In the next two sections, we will address the following ideas:

1. The summation-by-parts property on Gauss-Lobatto nodes is the discrete analogue of integration by parts (see Section 1.5).
2. Identity (1.32) satisfied by entropy conservative fluxes is the discrete analogue of chain rule $\mathbf{f}(\mathbf{u})\frac{\partial \mathbf{v}}{\partial x} = \frac{\partial \psi(\mathbf{v})}{\partial x}$. The flux differencing technique, which can be thought as linear combination of entropy conservative fluxes (but different than the construction in [66]), is the key to entropy balance within cells (see Section 1.6).
3. The numerical flux $\widehat{\mathbf{f}}$ at cell interfaces should be entropy stable (see Section

1.6).

1.5 Gauss-Lobatto quadrature and summation-by-parts

The discontinuous Galerkin spectral element method (DGSEM) is developed by applying quadrature rules to the two integrals in (1.57), and evolving nodal values at these quadrature points. We are going to choose the Legendre-Gauss-Lobatto quadrature rule. Consider the reference element $I = [-1, 1]$ associated with Gauss-Lobatto quadrature points

$$-1 = \xi_0 < \xi_1 < \cdots < \xi_k = 1,$$

and quadrature weights $\{\omega_j\}_{j=0}^{k}$. Let $(\cdot,\cdot)$ and $(\cdot,\cdot)_\omega$ denote the continuous and discrete inner product

$$(\mathbf{u},\mathbf{v}) := \int_{-1}^{1} \mathbf{u}^T \mathbf{v} d\xi, \quad (\mathbf{u},\mathbf{v})_\omega := \sum_{j=0}^{k} \omega_j \mathbf{u}(\xi_j)^T \mathbf{v}(\xi_j) \quad \text{for } \mathbf{u},\mathbf{v} \in (L^2(I))^p,$$

and define the Lagrangian (nodal) basis polynomials

$$L_j(\xi) = \prod_{\substack{l=0 \\ l\neq j}}^{k} \frac{\xi - \xi_l}{\xi_j - \xi_l},$$

such that $L_j(\xi_l) = \delta_{jl}$.

Now we bring forth the vector notation of nodal functions, and some discrete operators (matrices) of nodal functions. For a function u on I,

$$\overrightarrow{u} := \begin{bmatrix} u(\xi_0) & \cdots & u(\xi_k) \end{bmatrix}^T$$

represents the vector of values of u on quadrature points. The mass matrix M and boundary matrix B are set to be

$$M := \operatorname{diag}\{\omega_0, \cdots, \omega_k\}, \tag{1.64}$$

$$B := \operatorname{diag}\{\tau_0, \cdots, \tau_k\} = \{-1, 0, \cdots, 0, 1\}. \tag{1.65}$$

B corresponds to a (zero-dimensional exact) quadrature rule on the boundary such that for $u, v \in L^2(I)$,

$$\overrightarrow{u}^T M \overrightarrow{v} = \sum_{j=0}^{k} \omega_j u_i v_j = (u, v)_\omega, \quad \overrightarrow{u}^T B \overrightarrow{v} = u(1)v(1) - u(-1)v(-1) = uv \mid_{-1}^{1}.$$

The indication of B will be more clear in higher space dimensions. The difference matrix is set to be

$$D := \{L_l'(\xi_j)\}_{0 \le j, l \le k} \tag{1.66}$$

Then D is exact for polynomial functions. If $u \in \mathcal{P}^k(I)$, $u(\xi) = \sum\limits_{l=0}^{k} u(\xi_l) L_l(\xi)$, and

$$(D\overrightarrow{u})_j = \sum_{l=0}^{k} L_l'(\xi_j) u(\xi_l) = u'(\xi_j).$$

In particular, since the derivative of constant function vanishes,

$$D\overrightarrow{1} = \overrightarrow{0}, \tag{1.67}$$

where $\overrightarrow{0}$ and $\overrightarrow{1}$ are the vector of 0s and 1s.

Theorem 1.11 (summation-by-parts property)**.** *Set the stiffness matrix $S := MD$. Then we have*

$$B = S + S^T = MD + D^T M, \tag{1.68}$$

which is a discrete analogue of integration by parts.

Proof. The stiffness matrix satisfies

$$S_{jl} = \omega_j L_l'(\xi_j) = \sum_{s=0}^{k} \omega_s L_j(\xi_s) L_l'(\xi_s) = (L_j, L_l')_\omega = (L_j, L_l'), \tag{1.69}$$

where we use the fact that Gauss-Lobatto quadrature is exact for polynomials of degree $2k-1$. Hence

$$S_{jl} + S_{lj} = (L_j, L_l') + (L_l, L_j') = L_j L_l \mid_{-1}^{1} = \delta_{kj}\delta_{kl} - \delta_{0j}\delta_{0l} = B_{ij}.$$

□

Remark 1.4. Recall (1.67). We immediately deduce that

$$S\overrightarrow{1} = MD\overrightarrow{1} = \overrightarrow{0}, \quad S^T\overrightarrow{1} = B\overrightarrow{1} = \begin{bmatrix} -1 & 0 & \cdots & 0 & 1 \end{bmatrix}^T. \tag{1.70}$$

In order to incorporate vector-valued functions, we introduce the extended vector of nodal values

$$\overrightarrow{\mathbf{u}} := \begin{bmatrix} \mathbf{u}(\xi_0) \\ \vdots \\ \mathbf{u}(\xi_k) \end{bmatrix},$$

and the kronecker products

$$\mathbf{M} = M \otimes I_p, \quad \mathbf{D} = D \otimes I_p, \quad \mathbf{S} = S \otimes I_p, \quad \mathbf{B} = B \otimes I_p.$$

Then

$$\overrightarrow{\mathbf{u}}^T\mathbf{M}\overrightarrow{\mathbf{v}} = \langle \mathbf{u}, \mathbf{v} \rangle_\omega, \quad \overrightarrow{\mathbf{u}}^T\mathbf{B}\overrightarrow{\mathbf{v}} = \mathbf{u}^T\mathbf{v} \,|_{-1}^{1} .$$

We still have the SBP property

$$\mathbf{B} = \mathbf{S} + \mathbf{S}^T = \mathbf{MD} + \mathbf{D}^T\mathbf{M}, \tag{1.71}$$

and the Kronecker product versions of (1.67) and (1.70)

$$\mathbf{D}\overrightarrow{\mathbf{1}} = \mathbf{S}\overrightarrow{\mathbf{1}} = \overrightarrow{\mathbf{0}}, \quad \mathbf{S}^T\overrightarrow{\mathbf{1}} = \mathbf{B}\overrightarrow{\mathbf{1}}. \tag{1.72}$$

On a local cell I_i, we also need to consider the change of variables between I_i and the reference element $I = [-1, 1]$,

$$x_i(\xi) = \frac{1}{2}(x_{i-1/2} + x_{i+1/2}) + J_i\xi,$$

where $J_i = \frac{\Delta x_i}{2}$ is the Jacobian factor of mapping. The local quadrature points are $\{x_i(\xi_j)\}_{j=0}^k$, and the local discrete operators are scaled as

$$\mathbf{M}_i = J_i\mathbf{M}, \quad \mathbf{D}_i = \frac{1}{J_i}\mathbf{D}, \quad \mathbf{S}_i = \mathbf{S}, \quad \mathbf{B}_i = \mathbf{B}$$

Specific to the DG form (1.57), let $\overrightarrow{\mathbf{u}_i}$ and $\overrightarrow{\mathbf{w}_i}$ denote the values of $\mathbf{u}_h$ and $\mathbf{w}_h$ at Gauss-Lobatto points

$$\overrightarrow{\mathbf{u}_i} := \begin{bmatrix} \mathbf{u}_h(x_i(\xi_0)) \\ \vdots \\ \mathbf{u}_h(x_i(\xi_k)) \end{bmatrix}, \quad \overrightarrow{\mathbf{w}_i} := \begin{bmatrix} \mathbf{w}_h(x_i(\xi_0)) \\ \vdots \\ \mathbf{w}_h(x_i(\xi_k)) \end{bmatrix},$$

and $\overrightarrow{\mathbf{v}_i}$ be the short hand notation of the nodal values of $\mathbf{v}(\mathbf{u}_h)$. Likewise we can define $\overrightarrow{\mathbf{f}_i}$ and $\overrightarrow{U_i}$. We also put the numerical fluxes into a vector

$$\overrightarrow{\mathbf{f}_i^*} := \begin{bmatrix} \widehat{\mathbf{f}}_{i-1/2} \\ \mathbf{0} \\ \vdots \\ \mathbf{0} \\ \widehat{\mathbf{f}}_{i+1/2} \end{bmatrix}.$$

Using the vector notations and matrices above, we are able to derive the DGSEM in a compact matrix vector formulation. After applying Gauss-Lobatto quadrature the both integrals of (1.57), we have the approximation

$$\overrightarrow{\mathbf{w}_i}^T \mathbf{M}_i \frac{d\overrightarrow{\mathbf{u}_i}}{dt} - (\mathbf{D}_i \overrightarrow{\mathbf{w}_i})^T \mathbf{M}_i \overrightarrow{\mathbf{f}_i} = -\overrightarrow{\mathbf{w}_i}^T \mathbf{B}_i \overrightarrow{\mathbf{f}_i^*}.$$

Since $\overrightarrow{\mathbf{w}_i}$ can be arbitrary, we arrive at the weak DGSEM formulation [62, 51]

$$\mathbf{M}_i \frac{d\overrightarrow{\mathbf{u}_i}}{dt} - \mathbf{S}_i^T \overrightarrow{\mathbf{f}_i} = -\mathbf{B}_i \overrightarrow{\mathbf{f}_i^*}. \tag{1.73}$$

Using the SBP property (1.68), we can deduce another equivalent characterization, corresponding to the strong form (1.60).

$$\mathbf{M}_i \frac{d\overrightarrow{\mathbf{u}_i}}{dt} + \mathbf{S}_i \overrightarrow{\mathbf{f}_i} = \mathbf{B}_i (\overrightarrow{\mathbf{f}_i} - \overrightarrow{\mathbf{f}_i^*}),$$

i.e.,

$$\frac{d\overrightarrow{\mathbf{u}_i}}{dt} + \mathbf{D}_i \overrightarrow{\mathbf{f}_i} = \mathbf{M}_i^{-1} \mathbf{B}_i (\overrightarrow{\mathbf{f}_i} - \overrightarrow{\mathbf{f}_i^*}). \tag{1.74}$$

It is closely related to the spectral collocation method with penalty type boundary treatment in [50]. We can also use global operators to describe (1.73) and (1.74)

$$J_i \mathbf{M} \frac{d\overrightarrow{\mathbf{u}_i}}{dt} - \mathbf{S}^T \overrightarrow{\mathbf{f}_i} = -\mathbf{B}\overrightarrow{\mathbf{f}_i^*}, \tag{1.75}$$

$$J_i \frac{d\overrightarrow{\mathbf{u}_i}}{dt} + \mathbf{D}\overrightarrow{\mathbf{f}_i} = \mathbf{M}^{-1}\mathbf{B}(\overrightarrow{\mathbf{f}_i} - \overrightarrow{\mathbf{f}_i^*}). \tag{1.76}$$

These DGSEM forms do not satisfy any entropy condition (even L^2 stability that is satisfied by the classic DG method) due to the lack of chain rule. In the next section, we will make them entropy stable through the flux differencing trick.

Remark 1.5. Since the algebraic degree of accuracy is $2k-1$, the Gauss-Lobatto quadrature is not exact for the first integral (1.57). Such technique is typically termed *mass lumping*. On the other hand, it is exact for the second integral if $\mathbf{f}$ is linear.

1.6 Flux differencing

In the flux differencing technique, we replace the difference term in (1.76) with difference operation on entropy conservative fluxes. The modified DGSEM reads

$$J_i \frac{d\overrightarrow{\mathbf{u}_i}}{dt} + 2\mathbf{D} \circ \mathbf{F}_S(\overrightarrow{\mathbf{u}_i}, \overrightarrow{\mathbf{u}_i})\overrightarrow{\mathbf{1}} = \mathbf{M}^{-1}\mathbf{B}(\overrightarrow{\mathbf{f}_i} - \overrightarrow{\mathbf{f}_i^*}), \tag{1.77}$$

where $\circ$ denotes the Hadamard (pointwise) product of vectors and matrices, and $\mathbf{F}_S(\cdot,\cdot)$ is the matrix of entropy conservative fluxes

$$\mathbf{F}_S(\overrightarrow{\mathbf{u}_L}, \overrightarrow{\mathbf{u}_R}) := \begin{bmatrix} \operatorname{diag}(\mathbf{f}_S(\mathbf{u}_{L,0}, \mathbf{u}_{R,0})) & \cdots & \operatorname{diag}(\mathbf{f}_S(\mathbf{u}_{L,0}, \mathbf{u}_{R,k})) \\ \vdots & \ddots & \vdots \\ \operatorname{diag}(\mathbf{f}_S(\mathbf{u}_{L,k}, \mathbf{u}_{R,0})) & \cdots & \operatorname{diag}(\mathbf{f}_S(\mathbf{u}_{L,k}, \mathbf{u}_{R,k})) \end{bmatrix}.$$

We can demystify involved flux differencing term the by writing down the evolution of nodal values

$$J_i \frac{d\mathbf{u}_{i,j}}{dt} + 2\sum_{l=0}^{k} D_{jl}\mathbf{f}_S(\mathbf{u}_{i,j}, \mathbf{u}_{i,l}) = \frac{\tau_j}{\omega_j}(\mathbf{f}_{i,j} - \mathbf{f}^*_{i,j}). \tag{1.78}$$

Before proving the main result of this chapter, we first give a lemma indicating local conservation and local entropy balance of (1.77).

Lemma 1.1. *If $\mathbf{f}_S$ is consistent and symmetric, then for the DGSEM form (1.77),*

$$\frac{d}{dt}(J_i \overrightarrow{\mathbf{1}}^T \mathbf{M}\overrightarrow{\mathbf{u}_i}) + \widehat{\mathbf{f}}_{i+1/2} - \widehat{\mathbf{f}}_{i-1/2} = 0. \tag{1.79}$$

If we further assume that $\mathbf{f}_S$ is entropy conservative with respect to some entropy function U, then

$$\frac{d}{dt}(J_i \overrightarrow{1}^T M\overrightarrow{U_i}) + (\mathbf{v}^T_{i,k}\widehat{\mathbf{f}}_{i+1/2} - \psi_{i,k}) - (\mathbf{v}^T_{i,0}\widehat{\mathbf{f}}_{i-1/2} - \psi_{i,0}) = 0. \tag{1.80}$$

Proof. Since $\mathbf{M}$ is diagonal, $\mathbf{M}(\mathbf{D} \circ \mathbf{F}_S(\overrightarrow{\mathbf{u}_i})) = \mathbf{S} \circ \mathbf{F}_S(\overrightarrow{\mathbf{u}_i})$, and by symmetry of $\mathbf{f}_S$, $\mathbf{F}_S(\overrightarrow{\mathbf{u}_i}, \overrightarrow{\mathbf{u}_i})$ is a symmetric matrix. Then

$$\begin{aligned}
\frac{d}{dt}(J_i \overrightarrow{\mathbf{1}}^T \mathbf{M}\overrightarrow{\mathbf{u}_i}) &= -2\overrightarrow{\mathbf{1}}^T \mathbf{S} \circ \mathbf{F}_S(\overrightarrow{\mathbf{u}_i}, \overrightarrow{\mathbf{u}_i})\overrightarrow{\mathbf{1}} + \overrightarrow{\mathbf{1}}^T \mathbf{B}(\overrightarrow{\mathbf{f}_i} - \overrightarrow{\mathbf{f}^*_i}) \\
&= -\overrightarrow{\mathbf{1}}^T(\mathbf{S} + \mathbf{S}^T) \circ \mathbf{F}_S(\overrightarrow{\mathbf{u}_i}, \overrightarrow{\mathbf{u}_i})\overrightarrow{\mathbf{1}} + \overrightarrow{\mathbf{1}}^T \mathbf{B}(\overrightarrow{\mathbf{f}_i} - \overrightarrow{\mathbf{f}^*_i}) \quad \text{(by symmetry of } \mathbf{f}_S\text{)} \\
&= -\overrightarrow{\mathbf{1}}^T \mathbf{B} \circ \mathbf{F}_S(\overrightarrow{\mathbf{u}_i}, \overrightarrow{\mathbf{u}_i})\overrightarrow{\mathbf{1}} + \overrightarrow{\mathbf{1}}^T \mathbf{B}(\overrightarrow{\mathbf{f}_i} - \overrightarrow{\mathbf{f}^*_i}) \quad \text{(by SBP property)} \\
&= -\overrightarrow{\mathbf{1}}^T \mathbf{B}\overrightarrow{\mathbf{f}^*_i} = -(\widehat{\mathbf{f}}_{i+1/2} - \widehat{\mathbf{f}}_{i-1/2}),
\end{aligned}$$

where we use the identity that

$$\mathbf{B}\circ\mathbf{F}_S(\overrightarrow{\mathbf{u}_i},\overrightarrow{\mathbf{u}_i})\overrightarrow{\mathbf{1}} = \begin{bmatrix} -\mathbf{f}_S(\mathbf{u}_{i,0},\mathbf{u}_{i,0}) \\ 0 \\ \vdots \\ 0 \\ \mathbf{f}_S(\mathbf{u}_{i,k},\mathbf{u}_{i,k}) \end{bmatrix} = \begin{bmatrix} -\mathbf{f}(\mathbf{u}_{i,0}) \\ 0 \\ \vdots \\ 0 \\ \mathbf{f}(\mathbf{u}_{i,k}) \end{bmatrix} = \mathbf{B}\overrightarrow{\mathbf{f}_i}. \tag{1.81}$$

As for internal entropy balance,

$$\begin{aligned} \frac{d}{dt}(J_i\overrightarrow{1}^T M\overrightarrow{U_i}) &= J_i\frac{d}{dt}\Big(\sum_{j=0}^{k}\omega_j U_{i,j}\Big) = J_i\Big(\sum_{j=0}^{k}\omega_j\mathbf{v}_{i,j}^T\frac{d\mathbf{u}_{i,j}}{dt}\Big) = \overrightarrow{\mathbf{v}_i}^T\mathbf{M}\frac{d\overrightarrow{\mathbf{u}_i}}{dt} \\ &= -2\overrightarrow{\mathbf{v}_i}^T\mathbf{S}\circ\mathbf{F}_S(\overrightarrow{\mathbf{u}_i},\overrightarrow{\mathbf{u}_i})\overrightarrow{\mathbf{1}} + \overrightarrow{\mathbf{v}_i}^T\mathbf{B}(\overrightarrow{\mathbf{f}_i} - \overrightarrow{\mathbf{f}_i^*}) \\ &= \overrightarrow{\mathbf{v}_i}^T(\mathbf{S}^T - \mathbf{S} - \mathbf{B})\circ\mathbf{F}_S(\overrightarrow{\mathbf{u}_i},\overrightarrow{\mathbf{u}_i})\overrightarrow{\mathbf{1}} + \overrightarrow{\mathbf{v}_i}^T\mathbf{B}(\overrightarrow{\mathbf{f}_i} - \overrightarrow{\mathbf{f}_i^*}) \quad \text{(by SBP property)} \\ &= \overrightarrow{\mathbf{v}_i}^T(\mathbf{S}^T - \mathbf{S})\circ\mathbf{F}_S(\overrightarrow{\mathbf{u}_i},\overrightarrow{\mathbf{u}_i})\overrightarrow{\mathbf{1}} - \overrightarrow{\mathbf{v}_i}^T\mathbf{B}\overrightarrow{\mathbf{f}_i^*} \quad \text{(by (1.81))}. \end{aligned}$$

The second terms equals $\mathbf{v}_{i,k}^T\widehat{\mathbf{f}}_{i+1/2} - \mathbf{v}_{i,0}^T\widehat{\mathbf{f}}_{i-1/2}$, and we simplify the first term

$$\begin{aligned} \overrightarrow{\mathbf{v}_i}^T(\mathbf{S}^T - \mathbf{S})\mathbf{F}_S(\overrightarrow{\mathbf{u}_i},\overrightarrow{\mathbf{u}_i})\overrightarrow{\mathbf{1}} &= \sum_{j=0}^{k}\sum_{l=0}^{k}\mathbf{v}_{i,j}^T(S_{lj} - S_{jl})\mathbf{f}_S(\mathbf{u}_{i,j},\mathbf{u}_{i,l}) \\ &= \sum_{j=0}^{k}\sum_{l=0}^{k}S_{lj}(\mathbf{v}_{i,j} - \mathbf{v}_{i,l})^T\mathbf{f}_S(\mathbf{u}_{i,j},\mathbf{u}_{i,l}) \quad \text{(by symmetry of } \mathbf{f}_S) \\ &= \sum_{j=0}^{k}\sum_{l=0}^{k}S_{lj}(\psi_{i,j} - \psi_{i,l}) \quad \text{(by entropy conservation of } \mathbf{f}_S) \\ &= \overrightarrow{\psi_i}^T(S^T - S)\overrightarrow{1} = \overrightarrow{\psi_i}^T B\overrightarrow{1} = \psi_{i,k} - \psi_{i,0} \quad \text{(by (1.70))}. \end{aligned}$$

Notice that the entropy conservation of $\mathbf{f}_S$ plays a similar role to the chain rule. □

We are ready to provide the main theorem, which states that the scheme (1.77) is conservative and entropy stable, and maintains high order accuracy.

Theorem 1.12. *Assume that all mappings and bivariate fluxes (such as $\mathbf{v}(\mathbf{u})$, $\psi(\mathbf{v})$, $\mathbf{f}_S(\mathbf{u}_L, \mathbf{u}_R)$, $\widehat{\mathbf{f}}(\mathbf{u}_L, \mathbf{u}_R)$, etc) are smooth and Lipschitz continuous. If $\mathbf{f}_S$ is consistent and symmetric and $\widehat{\mathbf{f}}$ is consistent, then (1.77) is conservative in the sense that*

$$\frac{d}{dt}\Big(\sum_{i=1}^{N} J_i \overrightarrow{\mathbf{1}}^T \mathbf{M} \overrightarrow{\mathbf{u}_i}\Big) = 0, \tag{1.82}$$

and high order accurate in the sense that for all i, j and smooth solution $\mathbf{u}$ of (1.1), the local truncation error

$$\frac{d\mathbf{u}_{i,j}}{dt} + 2\sum_{l=0}^{k} D_{i,jl}\mathbf{f}_S(\mathbf{u}_{i,j}, \mathbf{u}_{i,l}) - \frac{\tau_j}{J_i\omega_j}(\mathbf{f}_{i,j} - \mathbf{f}^*_{i,j}) = \mathcal{O}(h^k). \tag{1.83}$$

If we further assume that $\mathbf{f}_S$ is entropy conservative and $\widehat{\mathbf{f}}$ is entropy stable with respect to some entropy function U, then (1.77) is also entropy stable with respect to U in the sense that

$$\frac{d}{dt}\Big(\sum_{i=1}^{N} J_i \overrightarrow{1}^T M \overrightarrow{U_i}\Big) \leq 0. \tag{1.84}$$

Proof. Conservation: given Lemma 1.1,

$$\frac{d}{dt}\Big(\sum_{i=1}^{N} J_i \overrightarrow{\mathbf{1}}^T \mathbf{M} \overrightarrow{\mathbf{u}_i}\Big) = \sum_{i=0}^{N}(\widehat{\mathbf{f}}_{i-1/2} - \widehat{\mathbf{f}}_{i+1/2}) = 0.$$

Accuracy: $\mathbf{u}$ is single-valued at cell interfaces. By the consistency of $\widehat{\mathbf{f}}$, $\widehat{\mathbf{f}}_{i+1/2} = \mathbf{f}_{i,k} = \mathbf{f}_{i+1,0}$, and the last term in (1.83) vanishes. Since $\mathbf{u}$ is a smooth solution of (1.1), it suffices to prove that

$$2\sum_{l=0}^{k} D_{i,jl}\mathbf{f}_S(\mathbf{u}_{i,j}, \mathbf{u}_{i,l}) - \frac{\partial \mathbf{f}(\mathbf{u})}{\partial x}(x_i(\xi_j)) = \mathcal{O}(h^k).$$

Let $\widetilde{\mathbf{f}_S}(x, y) := \mathbf{f}_S(\mathbf{u}(x), \mathbf{u}(y))$ and $\widetilde{\mathbf{f}}(x) := \mathbf{f}(\mathbf{u}(x))$. Then $\widetilde{\mathbf{f}_S}$ is also symmetric and consistent such that $\widetilde{\mathbf{f}}_S(x,x) = \widetilde{\mathbf{f}}(x)$. Therefore

$$\frac{\partial \widetilde{\mathbf{f}}}{\partial x}(x) = \frac{\partial \widetilde{\mathbf{f}}_S}{\partial x}(x,x) + \frac{\partial \widetilde{\mathbf{f}}_S}{\partial y}(x,x) = 2\frac{\partial \widetilde{\mathbf{f}}_S}{\partial y}(x,x).$$

Due to the approximation property of local difference matrix D_i,

$$\begin{aligned} 2\sum_{l=0}^{k} D_{i,jl}\mathbf{f}_S(\mathbf{u}_{i,j},\mathbf{u}_{i,l}) &= 2\sum_{l=0}^{k} D_{i,jl}\widetilde{\mathbf{f}}_S(x_i(\xi_j),x_i(\xi_l)) = 2\frac{\partial \widetilde{\mathbf{f}}_S}{\partial y}(x_i(\xi_j),x_i(\xi_j)) + \mathcal{O}(h^k) \\ &= \frac{\partial \widetilde{\mathbf{f}}}{\partial x}(x_i(\xi_j)) + \mathcal{O}(h^k) = \frac{\partial \mathbf{f}(\mathbf{u})}{\partial x}(x_i(\xi_j)). \end{aligned}$$

Entropy stability: again by Lemma 1.1 and entropy stability of $\widehat{\mathbf{f}}$,

$$\begin{aligned} \frac{d}{dt}\Big(\sum_{i=1}^{N} J_i \overrightarrow{1}^T M \overrightarrow{U_i}\Big) &= \sum_{i=1}^{N}\Big((\mathbf{v}_{i,0}^T\widehat{\mathbf{f}}_{i-1/2} - \psi_{i,0}) - (\mathbf{v}_{i,k}^T\widehat{\mathbf{f}}_{i+1/2} - \psi_{i,k})\Big) \\ &= \sum_{i=1}^{N}\Big((\mathbf{v}_{i+1,0} - \mathbf{v}_{i,k})^T\widehat{\mathbf{f}}_{i+1/2} - (\psi_{i+1,0} - \psi_{i,k})\Big) \leq 0. \end{aligned}$$

□

Remark 1.6. Along the lines of [29], the entropy stable DGSEM (1.78) can be written in the finite volume manner

$$\frac{d\mathbf{u}_{i,j}}{dt} + \frac{1}{J_i\omega_j}(\mathbf{f}_{i,j+1/2} - \mathbf{f}_{i,j-1/2}) = 0, \tag{1.85}$$

where

$$\mathbf{f}_{i,j+1/2} = \begin{cases} \widehat{\mathbf{f}}_{i-1/2} & j = -1 \\ \widehat{\mathbf{f}}_{i+1/2} & j = k \\ 2\sum\limits_{l=0}^{j}\sum\limits_{r=j+1}^{k} S_{lr}\mathbf{f}_S(\mathbf{u}_{i,l},\mathbf{u}_{i,r}) & 0 \leq j \leq k-1 \end{cases}. \tag{1.86}$$

The entropy stability is also transformed into the local entropy inequality

$$\frac{dU_{i,j}}{dt} + \frac{1}{J_i\omega_j}(F_{i,j+1/2} - F_{i,j-1/2}) \leq 0, \tag{1.87}$$

where

$$F^i_{j+1/2} = \begin{cases} \frac{1}{2}(\mathbf{v}_{i-1,k} + \mathbf{v}_{i,0})^T \widehat{\mathbf{f}}_{i-1/2} - \frac{1}{2}(\psi_{i-1,k} + \psi_{i,0}) & j = -1 \\ \frac{1}{2}(\mathbf{v}_{i,k} + \mathbf{v}_{i+1,0})^T \widehat{\mathbf{f}}_{i+1/2} - \frac{1}{2}(\psi_{i,k} + \psi_{i+1,0}) & j = k \\ \sum\limits_{l=0}^{j} \sum\limits_{r=j+1}^{k} S_{lr}((\mathbf{v}_{i,l} + \mathbf{v}_{i,r})^T \mathbf{f}_S(\mathbf{u}_{i,l}, \mathbf{u}_{i,r}) - (\psi_{i,l} + \psi_{i,r})) & 0 \leq j \leq k-1 \end{cases} . \tag{1.88}$$

The Lax-Wendroff type argument (Theorem 1.5) will yield that, if the sequence of numerical solutions converges boundedly and a.e. to some function $\mathbf{u}$, then $\mathbf{u}$ is a weak solution of (1.1) supporting the entropy condition with respect to U. According to Theorem 1.4, such single entropy condition is enough to determine the entropy solution of scalar conservation laws with uniformly convex flux function.

We finish this section by examining the entropy stable DGSEM (1.77) for examples of conservation laws and their corresponding entropy functions. We will find that flux differencing is equivalent to splitting in some cases.

Example 1.6.1. For linear symmetric system (1.14), $\mathbf{f}_S(\mathbf{u}_{i,j}, \mathbf{u}_{i,l}) = \frac{1}{2}(\mathbf{f}(\mathbf{u}_{i,j}) + \mathbf{f}(\mathbf{u}_{i,l}))$ is the arithmetic mean. Then

$$2\mathbf{D} \circ \mathbf{F}_S(\overrightarrow{\mathbf{u}_i}, \overrightarrow{\mathbf{u}_i})\overrightarrow{\mathbf{1}} = (\mathbf{D}\overrightarrow{\mathbf{1}}) \circ \overrightarrow{\mathbf{f}_i} + \mathbf{D}\overrightarrow{\mathbf{f}_i} = \mathbf{D}\overrightarrow{\mathbf{f}_i} \quad \text{(by (1.72))},$$

and the scheme (1.77) reduces to the original DGSEM (1.76).

Example 1.6.2. For Burgers equation (1.16) with square entropy function, inserting the entropy conservative flux $f_S(u_L, u_R) = \frac{1}{6}(u_L^2 + u_L u_R + u_R^2)$ into the flux differencing term yields

$$2D \circ F_S(\overrightarrow{u_i}, \overrightarrow{u_i})\overrightarrow{1} = \frac{2}{3}D\overrightarrow{f_i} + \frac{1}{3}(D\overrightarrow{u_i}) \circ \overrightarrow{u_i} + \frac{2}{3}(D\overrightarrow{1}) \circ \overrightarrow{f_i} = \frac{2}{3}D\overrightarrow{f_i} + \frac{1}{3}\overrightarrow{u_i} \circ (D\overrightarrow{u_i}).$$

The method (1.77) turns into

$$J_i\frac{d\overrightarrow{u_i}}{dt} + \frac{2}{3}D\overrightarrow{f_i} + \frac{1}{3}(D\overrightarrow{u_i}) \circ \overrightarrow{u_i} = M^{-1}B(\overrightarrow{f_i} - \overrightarrow{f_i^*}).$$

This is the DGSEM discretization of the skew-symmetric form [30, 34]

$$\frac{\partial u}{\partial t} + \frac{1}{3}\frac{\partial(u^2)}{\partial x} + \frac{1}{3}u\frac{\partial u}{\partial x} = 0, \tag{1.89}$$

a well-known splitting technique for Burgers equation to improve stability. One may check [94] for the link between entropy stability and skew-symmetric splitting.

Example 1.6.3. For shallow water equations (1.18), the entropy conservative flux (1.40) also leads to a flux difference term equivalent to the DGSEM discretization of

$$\frac{\partial h}{\partial t} + \frac{\partial(hw)}{\partial x} = 0, \tag{1.90a}$$

$$\frac{\partial(hw)}{\partial t} + \frac{1}{2}\frac{\partial(hw^2)}{\partial x} + \frac{1}{2}w\frac{\partial(hw)}{\partial x} + \frac{1}{2}hw\frac{\partial w}{\partial x} + gh\frac{\partial h}{\partial x} = 0. \tag{1.90b}$$

We obtain the skew-symmetric form of (1.90b) by subtracting (1.90a) multiplied by $w/2$.

$$\frac{1}{2}(\frac{\partial(hw)}{\partial t} + h\frac{\partial w}{\partial t}) + \frac{1}{2}(\frac{\partial(hw^2)}{\partial x} + hw\frac{\partial w}{\partial x}) + gh\frac{\partial h}{\partial x} = 0, \tag{1.91}$$

which corresponds to the splitting procedure in [36].

Example 1.6.4. For Euler equations (1.21), both the Ismail-Roe entropy conservative flux (1.41) and the Chandrashekar entropy conservative flux (1.42) include logarithmic mean. Then the flux differencing is no longer equivalent to any kind of splitting.

Remark 1.7. Semidiscrete analysis is a crucial assumption. Fully discrete entropy stability analysis is available for implicit time integration [66] and space-time DG methods. The entropy stability of high-order schemes with explicit time integration, such as strong stability preserving (SSP) Runge-Kutta methods [42, 90], is still an open problem. There are positive results for the L^2 stability of the Runge-Kutta DG discretization of linear advection equation [110], but the nonlinear (in the sense of both flux function and entropy function) analogue is difficult to prove.

1.7 Compatibility with limiters

As in the classic DG method, it is possible to design TVD/TVB limiter and / or bound-preserving limiter as an extra stabilizing mechanism. Limiters tend to squeeze the data towards the cell average, and hence make total entropy smaller. We formulate such intuition in the following lemma.

Lemma 1.2. *Suppose $\alpha_j > 0, \mathbf{u}_j \in \Omega$ for $0 \leq j \leq k$ with $\sum\limits_{j=0}^{k} \alpha_j = 1$. Define the average $\overline{\mathbf{u}} := \sum\limits_{j=0}^{k} \alpha_j \mathbf{u}_j$. We modify these values without changing the average. Let*

$$\widetilde{\mathbf{u}}_j := \overline{\mathbf{u}} + \theta_j(\mathbf{u}_j - \overline{\mathbf{u}}), \quad 0 \leq \theta_j \leq 1,$$

such that $\sum\limits_{j=0}^{k} \alpha_j \widetilde{\mathbf{u}}_j = \overline{\mathbf{u}}$. Then for any convex entropy function U, we have

$$\sum_{j=0}^{k} \alpha_j U(\widetilde{\mathbf{u}}_j) \leq \sum_{j=0}^{k} \alpha_j U(\mathbf{u}_j). \tag{1.92}$$

Proof. Since $\sum\limits_{j=0}^{k} \alpha_j \widetilde{\mathbf{u}}_j = \sum\limits_{j=0}^{k} \alpha_j (\overline{\mathbf{u}} + \theta_j(\mathbf{u}_j - \overline{\mathbf{u}})) = \overline{\mathbf{u}}$,

$$\sum_{j=0}^{k} \alpha_j (1-\theta_j)\mathbf{u}_j = (\sum_{j=0}^{k} \alpha_j(1-\theta_j))\overline{\mathbf{u}}.$$

By the convexity of U,

$$U(\widetilde{\mathbf{u}}_j) \leq \theta_j U(\mathbf{u}_j) + (1-\theta_j)U(\overline{\mathbf{u}}), \quad (\sum_{j=0}^{k} \alpha_j(1-\theta_j))U(\overline{\mathbf{u}}) \leq \sum_{j=0}^{k} \alpha_j(1-\theta_j)U(\mathbf{u}_j).$$

Hence

$$\begin{aligned}\sum_{j=0}^{k} \alpha_j U(\widetilde{\mathbf{u}}_j) \leq \sum_{j=0}^{k} \alpha_j(\theta_j U(\mathbf{u}_j) + (1-\theta_j)U(\overline{\mathbf{u}})) &= \sum_{j=0}^{k} \alpha_j \theta_j U(\mathbf{u}_j) + (\sum_{j=0}^{k} \alpha_j(1-\theta_j))U(\overline{\mathbf{u}}) \\ \leq \sum_{j=0}^{k} \alpha_j \theta_j U(\mathbf{u}_j) + \sum_{j=0}^{k} \alpha_j(1-\theta_j)U(\mathbf{u}_j) &= \sum_{j=0}^{k} \alpha_j U(\mathbf{u}_j).\end{aligned}$$

□

The bound-preserving limiter was developed by Zhang and Shu in [112, 113] to maintain the physical bound Ω of numerical approximations, such as the maximum principle for scalar conservation laws and positivity of density and pressure for Euler equations. This technique is constructed on Gauss-Lobatto nodes, so that it perfectly matches our nodal DG scheme. We will clarify the theoretical issues of bound-preserving limiter in Appendix B. In a nutshell, we compute the cell average $\overline{\mathbf{u}}_i := \sum\limits_{j=0}^{k} \frac{\omega_j}{2}\mathbf{u}_{i,j}$ and perform a simple linear limiting procedure with some $0 \leq \theta_i \leq 1$ such that $\widetilde{\mathbf{u}}_{i,j} := \overline{\mathbf{u}}_i + \theta(\mathbf{u}_{i,j} - \overline{\mathbf{u}}_i) \in \Omega$. Clearly, we have the following entropy stability result due to Lemma 1.2.

Theorem 1.13. *Bound-preserving limiter does not increase entropy.*

Bound-preserving limiter helps enhance robustness, but the solution profile may still contain oscillations. The TVD/TVB limiter is well suited for damping oscillations. For scalar conservation laws, the TVD type limiting procedure can be defined as

$$\widetilde{u}_{i,0} := \overline{u}_i - m(\overline{u}_i - u_{i,0}, \overline{u}_{i+1} - \overline{u}_i, \overline{u}_i - \overline{u}_{i-1}), \quad \widetilde{u}_{i,k} := \overline{u}_i + m(u_{i,k} - \overline{u}_i, \overline{u}_{i+1} - \overline{u}_i, \overline{u}_i - \overline{u}_{i-1}),$$

$$\widetilde{u}_{i,j} := \overline{u}_i + \theta(u_{i,j} - \overline{u}_i) \text{ for } 1 \le j \le k-1, \text{ where } \theta := \frac{(\widetilde{u}_{i,0} - \overline{u}_i) + (\widetilde{u}_{i,k} - \overline{u}_i)}{(u_{i,0} - \overline{u}_i) + (u_{i,k} - \overline{u}_i)}.$$

We set θ such that cell average does not change. The minmod function m is

$$m(a,b,c) := \begin{cases} s \min\{|a|, |b|, |c|\} & \text{if } s = \text{sign}(a) = \text{sign}(b) = \text{sign}(c) \\ 0 & \text{otherwise} \end{cases}.$$

The TVB (total variation bounded) limiter is devised by replacing m with the modified minmod function $\widetilde{m}$ [88].

$$\widetilde{m}(a,b,c) = \begin{cases} a & \text{if } |a| \le Mh^2 \\ \text{sign}(a) \max\{|m(a,b,c)|, Mh^2\} & \text{if } |a| > Mh^2 \end{cases},$$

where M is a parameter that has to be tuned appropriately.

Theorem 1.14. *For scalar conservation laws, the TVD/TVB limiter mentioned above does not increase entropy.*

Proof. We only focus on the TVD limiter. The proof for the TVB limiter is exactly the same. Without loss of generality we assume that $\overline{u}_i = 0$. According to Lemma 1.2, we only need to show that $0 \le \widetilde{u}_{i,j}/u_{i,j} \le 1$ for each $0 \le j \le k$. By the definition of minmod function,

$$\frac{\widetilde{u}_{i,0}}{u_{i,0}} = -\frac{m(-u_{i,0}, -\overline{u}_{i-1}, \overline{u}_{i+1})}{u_{i,0}} \in [0,1], \quad \frac{\widetilde{u}_{i,k}}{u_{i,k}} = \frac{m(u_{i,k}, -\overline{u}_{i-1}, \overline{u}_{i+1})}{u_{i,k}} \in [0,1].$$

It remains to prove that $0 \leq \theta \leq 1$. If $u_{i,0}$ and $u_{i,k}$ have the same sign, it is obvious. Otherwise we assume that $u_{i,0} < 0$, $u_{i,k} > 0$ and $u_{i,0} + u_{i,k} \geq 0$. Then $\widetilde{u}_{i,0} = -\min\{-u_{i,0}, \overline{u}^{-}_{i-1}, \overline{u}^{+}_{i+1}\}$ and $\widetilde{u}_{i,k} = \min\{u_{i,k}, \overline{u}^{-}_{i-1}, \overline{u}^{+}_{i+1}\}$. It is easy to verify that $0 \leq \widetilde{u}_{i,0} + \widetilde{u}_{i,k} \leq u_{i,0} + u_{i,k}$. Other cases can be proved in a similar fashion. □

Remark 1.8. In general the TVD/TVB limiter for systems is not guaranteed to be entropy stable. The reason is that different components or characteristics are limited independently, which does not satisfy the assumption of Lemma 1.2 and the influence on total entropy is undecided. Certainly we could come up with a limiter that squeeze all components to the same degree, but it might be too restrictive.

Remark 1.9. There is a still a gap in our result: entropy stability relies on semi-discrete analysis, while limiters can only be applied to fully discrete schemes. If we assume the fully discrete version of (1.77) is entropy stable, since limiters do not increase total entropy, the scheme modified by limiters is also entropy stable.

1.8 Numerical experiments

In this section, we test the performance of the entropy stable DGSEM (1.77) for one-dimensional systems of conservation laws. All tests are performed on uniform grids. The schemes are integrated in time with third order SSP Runge-Kutta method (given in Appendix B). Godunov flux will be employed at cell interfaces. For Euler equations, the ratio of specific heat γ is taken to be 7/5, and the entropy conservative flux (1.42) will be used.

We first test problems with smooth solutions to validate the accuracy of the method. We would like to compute with $k = 2, 3, 4$. If $k = 2$, we set the CFL number to be 0.15; otherwise we will let $\Delta t = \text{CFL} \cdot h^{(k+1)/3}$, so that time error will be dominated by space error.

Example 1.8.1. We solve the linear advection equation

$$\frac{\partial u}{\partial x} + \frac{\partial u}{\partial t} = 0, \quad x \in [0, 2\pi],$$

with periodic boundary condition and initial data $u(0, x) = \sin^4(x)$. The exact solution is $u(t, x) = \sin^4(x - t)$. The entropy function in this case is the exponential function $U = e^u$, and the entropy conservative flux is given by

$$f_S(u_L, u_R) = \frac{(u_R - 1)e^{u_R} - (u_L - 1)e^{u_L}}{e^{u_R} - e^{u_L}}, \quad \text{if } u_L \neq u_R.$$

When $|u_L - u_R|$ is small, such formula suffers from round-off effect. Instead, we should use Taylor's expansion to approximate the numerator and the denominator. Numerical errors and orders of convergence of the entropy stable DGSEM with $k = 2, 3, 4$ are listed in Table 1.1. The scheme is evolved up to $t = 2\pi$. We observe optimal $(k + 1)$-th order convergence for all values of k.

Example 1.8.2. Next we consider the Burgers equation

$$\frac{\partial u}{\partial t} + \frac{\partial (u^2/2)}{\partial x} = 0, \quad x \in [0, 2\pi],$$

with periodic boundary condition and initial data $u(0, x) = 0.5 + \sin x$. The exact solution can be obtained by tracing back characteristic lines. We choose square entropy function $U = u^2/2$. Then the entropy stable DGSEM is equivalent to the skew-symmetric splitting. In Table 1.2, we present the errors at $t = 0.5$ when the solution is still smooth. It is evident that the convergence rate is below optimal,

Table 1.1: Example 1.8.1: accuracy test of the linear advection equation associated with initial data $u(x,0) = \sin^4(x)$ and exponential entropy function at $t = 2\pi$.

k	N	L^1 error	order	L^2 error	order	L^∞ error	order
2	20	7.030e-2	-	3.347e-2	-	2.688e-2	-
	40	5.363e-3	3.712	2.669e-3	3.649	2.340e-3	3.522
	80	4.575e-4	3.551	2.205e-4	3.598	1.846e-4	3.664
	160	4.414e-5	3.374	2.230e-5	3.305	2.582e-5	2.838
	320	4.745e-6	3.218	2.595e-6	3.103	3.626e-6	2.832
	640	5.485e-7	3.113	3.181e-7	3.028	4.794e-7	2.919
3	20	3.097e-3	-	1.514e-3	-	1.890e-3	-
	40	1.675e-4	4.208	8.672e-5	4.126	1.359e-4	3.798
	80	1.053e-5	3.993	5.372e-6	4.013	8.928e-6	3.928
	160	6.571e-7	4.002	3.354e-7	4.001	5.664e-7	3.978
	320	4.107e-8	4.000	2.096e-8	4.000	3.553e-8	3.995
4	10	2.608e-2	-	1.178e-2	-	8.580e-3	-
	20	8.325e-4	4.969	3.763e-4	4.969	3.497e-4	4.617
	40	2.623e-5	4.988	1.179e-5	4.997	9.860e-6	5.149
	80	8.170e-7	5.004	3.683e-7	5.000	3.084e-7	4.999
	160	2.553e-8	5.000	1.151e-8	5.000	9.454e-9	5.028

especially for the L^∞ error and even values of k.

Table 1.2: Example 1.8.2: accuracy test of the Burgers equation associated with initial data $u(0,x) = 0.5 + \sin x$ and square entropy function at $t = 0.5$.

k	N	L^1 error	order	L^2 error	order	L^∞ error	order
2	40	1.320e-3	-	1.178e-3	-	3.269e-3	-
	80	2.071e-4	2.672	2.284e-4	2.366	7.923e-4	2.045
	160	3.162e-5	2.711	4.316e-5	2.404	2.078e-4	1.931
	320	4.724e-6	2.743	7.979e-6	2.435	5.100e-5	2.026
	640	6.911e-7	2.773	1.450e-6	2.460	1.290e-5	1.983
	1280	9.930e-8	2.799	2.606e-7	2.477	3.209e-6	2.008
3	40	4.344e-5	-	4.566e-5	-	1.658e-4	-
	80	3.348e-6	3.698	3.703e-6	3.624	1.610e-5	3.364
	160	2.344e-7	3.836	2.771e-7	3.740	1.306e-6	3.624
	320	1.577e-8	3.894	1.950e-8	3.829	9.301e-8	3.812
	640	1.036e-9	3.928	1.336e-9	3.868	6.252e-9	3.895
4	20	6.782e-5	-	6.319e-5	-	1.525e-4	-
	40	2.630e-6	4.688	2.849e-6	4.471	1.126e-5	3.760
	80	1.067e-7	4.624	1.374e-7	4.375	7.149e-7	3.977
	160	4.203e-9	4.666	6.385e-9	4.427	4.342e-8	4.041
	320	1.576e-10	4.737	2.858e-10	4.481	2.620e-9	4.050

Remark 1.10. The suboptimal convergence is probably due to the fact that the Gauss-Lobatto quadrature is exact for polynomials of degree only up to $2k-1$. In order to maintain optimal convergence, the algebraic degree of accuracy should be at least $2k$ (see [15]). In linear problems, the Gauss-Lobatto quadrature is exact for the linear convective part, and we still get optimal convergence.

Example 1.8.3. We solve Euler equations with initial condition

$$\rho(0,x) = 1 - 0.5\sin x, \quad w(0,x) = p(0,x) = 1, \quad x \in [0, 2\pi],$$

with periodic boundary condition. The exact solution is

$$\rho(t,x) = 1 - 0.5\sin(x-t), \quad w(t,x) = p(t,x) = 1.$$

Errors and orders of convergence of the density variable at $t = 1$ are given in Table 1.3. Here the convergence rate of entropy stable DGSEM is also optimal. It should be related to the linear behavior of the exact solution.

Then we provide discontinuous test problems to illustrate shock-capturing capability. We will only show the numerical solutions with $k = 2$ and CFL number 0.15. The bound-preserving limiter can be added to make the entropy stable DGSEM more robust. However, for Euler equations, due to the lack of entropy stable TVD/TVB limiter, there are still some oscillations in test results.

Example 1.8.4. We consider the following Riemann problem of Buckley-Leverett equation

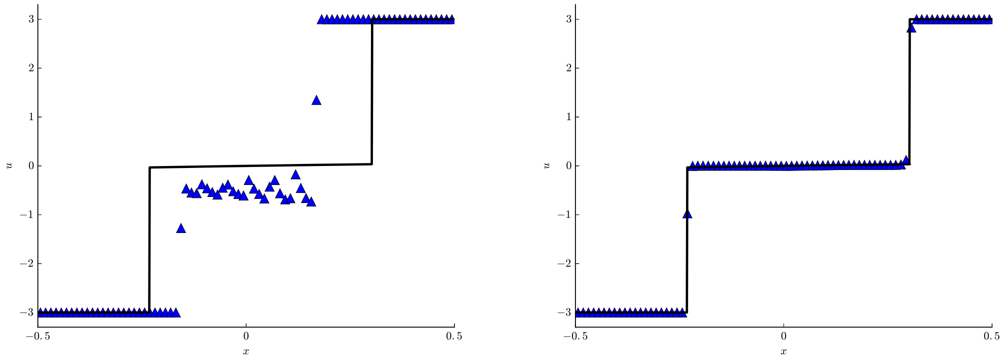
$$\frac{\partial u}{\partial t} + \frac{\partial}{\partial x}\Big(\frac{4u^2}{4u^2+(1-u)^2}\Big) = 0, \quad u(0,x) = \begin{cases} -3 & \text{if } x < 0 \\ 3 & \text{if } x \geq 0 \end{cases}.$$

Table 1.3: Example 1.8.3: accuracy test of one-dimensional Euler equations associated with initial data $u(0,x) = 0.5 + \sin x$ at $t = 1$. Results of density are tabulated.

k	N	L^1 error	order	L^2 error	order	L^∞ error	order
2	20	9.019e-04		5.216e-04		6.653e-04	
	40	1.105e-04	3.029	6.537e-05	2.996	8.253e-05	3.011
	80	1.363e-05	3.018	8.175e-06	2.999	1.028e-05	3.006
	160	1.694e-06	3.009	1.022e-06	3.000	1.282e-06	3.003
	320	2.110e-07	3.005	1.278e-07	3.000	1.601e-07	3.002
	640	2.633e-08	3.002	1.597e-08	3.000	2.000e-08	3.001
3	20	1.560e-05		9.882e-06		2.517e-05	
	40	9.951e-07	3.970	6.280e-07	3.976	1.718e-06	3.873
	80	6.223e-08	3.999	3.928e-08	3.999	1.084e-07	3.987
	160	3.890e-09	4.000	2.455e-09	4.000	6.772e-09	4.000
	320	2.432e-10	4.000	1.534e-10	4.000	4.233e-10	4.000
4	10	1.097e-05		9.013e-06		2.279e-05	
	20	2.980e-07	5.203	2.384e-07	5.240	7.351e-07	4.954
	40	9.620e-09	4.953	7.880e-09	4.919	2.297e-08	5.000
	80	2.871e-10	5.066	2.488e-10	4.985	7.669e-10	4.905
	160	8.885e-12	5.014	7.692e-12	5.015	2.402e-11	4.997

The exact entropy solution contains two shock waves connected by a flat rarefaction wave that is close to 0. For such a non-convex flux function, the choice of entropy function will affect the performance of numerical method substantially. We first test the scheme with square entropy function $U = u^2/2$. The computational domain is $[-0.5, 0.5]$ and the end time $t = 1$. We also apply the bound-preserving limiter with $\Omega = [-3, 3]$. The numerical solution on 80 cells is plotted in the left panel of Figure 1.1. Evidently it does not agree with the entropy solution. Then we try an ad hoc entropy function $U = \int \arctan(20u)du$. The entropy variable $v = \arctan(20u)$, which emphasizes the states near $u = 0$. In fact it can be viewed as a mollified version of the Kruzhkov's entropy function $U = |u|$ (see (1.13)). The numerical solution with the same setting is depicted in the right panel of Figure 1.1. The result is quite satisfactory thanks to the carefully chosen entropy function.

Example 1.8.5 (Sod's shock tube)**.** It is a classical Riemann problem of Euler



(a) $U = u^2/2$

(b) $U = \int \arctan(20u)du$

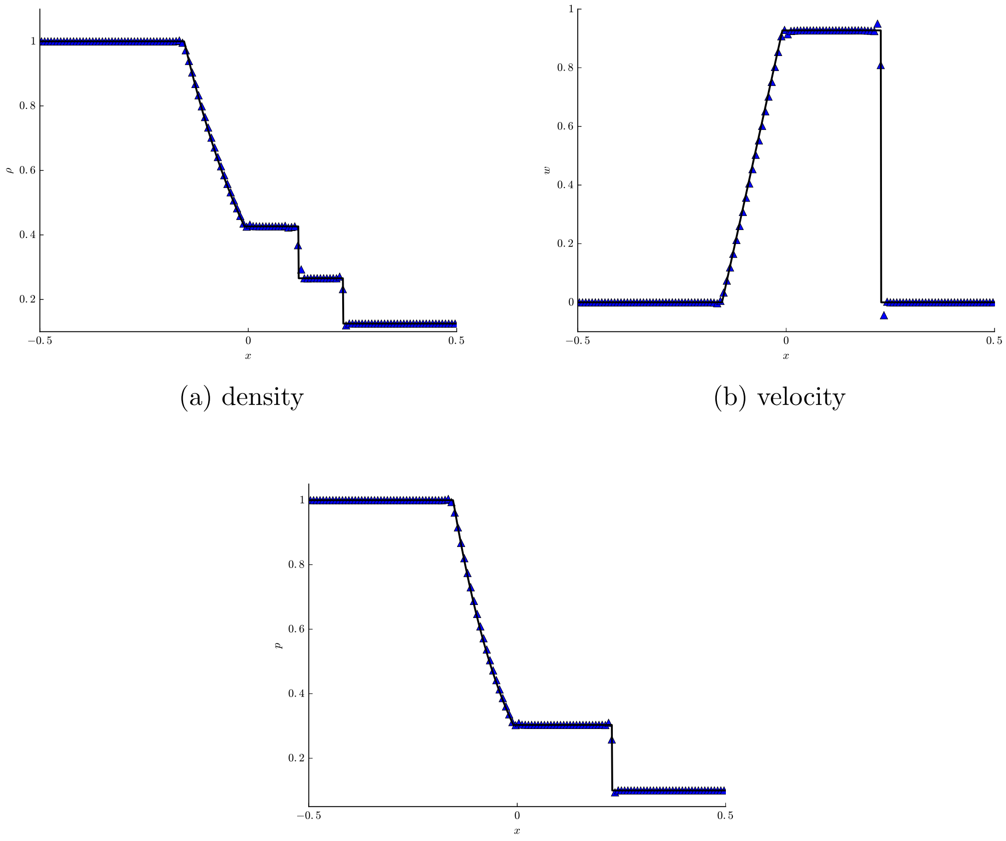
Figure 1.1: Example 1.8.4. Numerical solution of the Riemann problem of Buckley-Leverett equation at $t = 1$ with the square entropy function and an ad hoc entropy function. Computational domain $[-0.5, 0.5]$ is decomposed into $N = 80$ cells. Bound-preserving limiter is used. The solid line represents the exact entropy solution and the triangle symbols are cell averages.

equations. The computational domain is $[-0.5, 0.5]$ and the initial condition is

$$\begin{bmatrix} \rho & w & p \end{bmatrix}^T = \begin{cases} \begin{bmatrix} 1 & 0 & 1 \end{bmatrix}^T & \text{if } x < 0 \\ \begin{bmatrix} 0.125 & 0 & 0.1 \end{bmatrix}^T & \text{if } x \geq 0 \end{cases}.$$

The exact solution contains a left rarefaction wave, a right shock wave and a middle contact discontinuity. The classic DG method tends to blow up due to emergence of negative density or negative pressure unless we apply bound-preserving limiter or TVD/TVB limiter. The entropy stable DGSEM, on the other hand, can be evolved without any limiter. Figure 1.2 illustrates the profiles of density, velocity and pressure at $t = 0.13$ with 130 cells. All waves are resolved correctly despite some slight oscillations at the right shock wave. Entropy stability contributes to a more robust scheme for this test problem.

Example 1.8.6 (Sine-shock interaction)**.** This benchmark test problem of Euler equations was given by Shu and Osher in [91]. The solution has complicated structure



(a) density

(b) velocity

(c) pressure

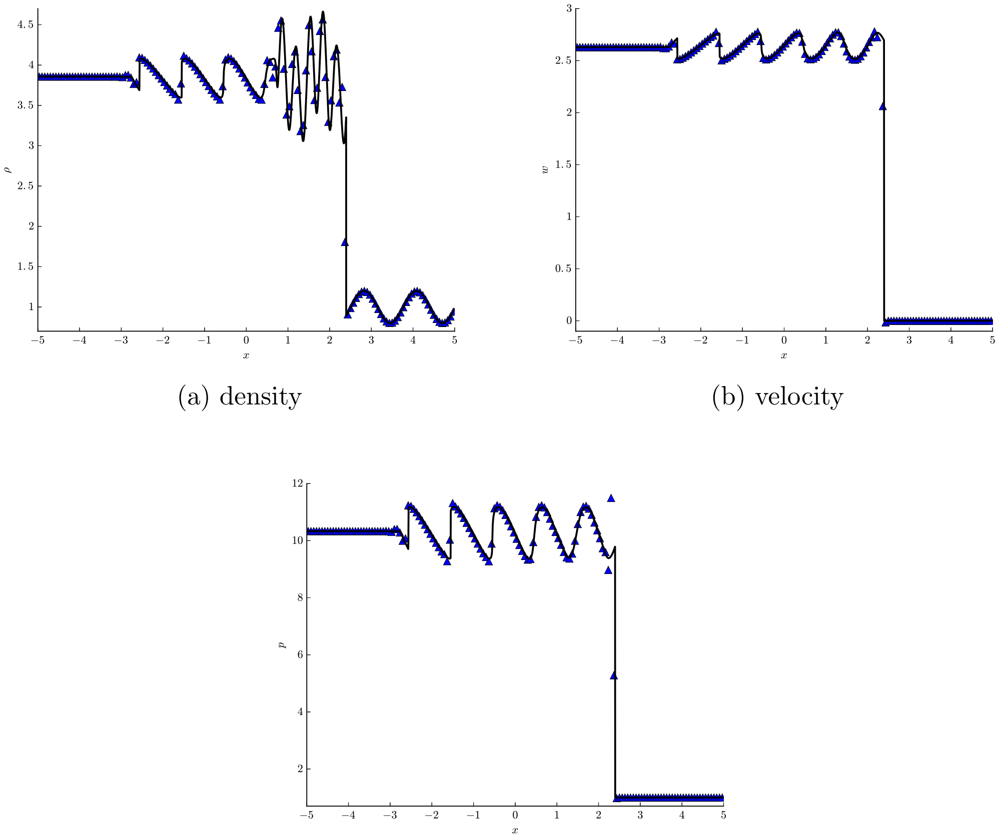
Figure 1.2: Example 1.8.5: Numerical solution of Sod's shock tube problem at $t = 0.13$ with 130 cells. We do not apply any limiter. The solid line represents the exact entropy solution and the triangle symbols are cell averages.

in that it contains both strong and weak shock waves and highly oscillatory smooth waves. The computational domain is $[-5, 5]$ and the initial condition is

$$\begin{bmatrix}\rho & w & p\end{bmatrix}^T = \begin{cases}\begin{bmatrix}3.857143 & 2.629369 & 10.3333\end{bmatrix}^T & \text{if } x < -4\\ \begin{bmatrix}1 + 0.2\sin(5x) & 0 & 1\end{bmatrix}^T & \text{if } x \geq -4\end{cases}.$$

We compute the reference solution using a first order scheme on a very fine mesh with 80000 cells. Once again the classic DG method suffers from negative pressure or negative density, while the entropy stable DGSEM works without any limiter.

The plots of density, velocity and pressure at $t = 1.8$ with 150 cells are displayed in Figure 1.3. The scheme performs well despite some minor oscillations.



(a) density

(b) velocity

(c) pressure

Figure 1.3: Example 1.8.6: Numerical solution of sine-shock interaction test problem at $t = 1.8$ with 150 cells. We do not apply any limiter. The solid line represents the reference solution computed with 80000 cells and the triangle symbols are cell averages.

Chapter Two

Generalization to Higher Space Dimensions

In this chapter, we move on to systems of conservation laws in arbitrary space dimensions

$$\frac{\partial \mathbf{u}}{\partial t}+\sum_{m=1}^{d}\frac{\partial \mathbf{f}_m(\mathbf{u})}{\partial x_m}=0,\quad (t,\mathbf{x})\in[0,\infty)\times\mathbb{R}^d \tag{2.1}$$

For a normal vector $\mathbf{n}\in\mathbb{R}^d$, let $\mathbf{f}_{\mathbf{n}}(\mathbf{u}):=\sum\limits_{m=1}^{d}n_m\mathbf{f}_m(\mathbf{u})$. (2.1) is called hyperbolic if $\mathbf{f}_{\mathbf{n}}'(\mathbf{u})$ has p real eigenvalues and a complete set of eigenvectors, for any $\mathbf{u}\in\Omega$ and $\mathbf{n}\in\mathbb{R}^d$. We always assume hyperbolicity.

The one-dimensional entropy stable DGSEM framework in Chapter 1 can be directly applied to multi-dimensional Cartesian meshes through tensor product. However, for problems with complex geometry, it is often desired to consider numerical methods that work on unstructured meshes. Inspired by [52], we construct of multi-dimensional discrete operators with summation-by-parts property. With the SBP operators and flux differencing technique at hand, we are able to develop the entropy stable DG type method on unstructured meshes. We remark that another way to discretize domains with complex geometry is to use curvilinear Cartesian meshes. See [35] for the survey of entropy stable DGSEM on curvilinear meshes.

This chapter consists of the following sections. In Section 2.1, we briefly review the entropy analysis and numerical discretization of systems of hyperbolic conservation laws. Most materials are essentially the same as the one-dimensional counterpart in Section 1.1 – Section 1.4. We only go through some key concepts. In Section 2.2, we design the multi-dimensional SBP operators, mainly on simplicial elements, but the general idea works for any polygonal element. In Section 2.3, we introduce the multi-dimensional high order entropy stable DGSEM. In Section 2.4, we discuss a specific topic of practical importance, i.e., the entropy stability of wall boundary condition for two-dimensional Euler equations. In Section 2.5, we consider convection-diffusion equations, for which an entropy stable local discontinuous Galerkin (LDG) type method [18, 8] will be included to handle the diffusive term. Finally in Section 2.6, we will perform numerical experiments on some two-dimensional test cases.

2.1 Preliminaries

Similar to one-dimensional problems, we define the convex entropy function and entropy solution for multi-dimensional systems of conservation laws.

Definition 2.1. *A convex function $U : \Omega \to \mathbb{R}$ is called an entropy function for (2.1) if there exist functions $\{F_m(\mathbf{u})\}_{m=1}^{d}$, called entropy fluxes, such that the following integrability condition holds*

$$U'(\mathbf{u})\mathbf{f}_m'(\mathbf{u}) = F_m'(\mathbf{u}), \quad 1 \le m \le d. \tag{2.2}$$

Definition 2.2. *A weak solution $\mathbf{u}$ of (2.1) is called an entropy solution if for all possible entropy functions U, we have*

$$\frac{\partial U(\mathbf{u})}{\partial t} + \sum_{m=1}^{d} \frac{\partial F_m(\mathbf{u})}{\partial x_m} \le 0, \tag{2.3}$$

in the sense of distribution.

For an entropy function U, let $\mathbf{v} := U'(\mathbf{u})$ be the entropy variables and $\mathbf{g}_m(\mathbf{v}) := \mathbf{f}(\mathbf{u}(\mathbf{v}))$. Then the symmetrization of (2.1) is

$$\mathbf{u}'(\mathbf{v})\frac{\partial \mathbf{v}}{\partial t} + \sum_{m=1}^{d} \mathbf{g}_m'(\mathbf{v})\frac{\partial \mathbf{v}}{\partial x_m} = 0. \tag{2.4}$$

We also define the potential function and potential fluxes

$$\phi(\mathbf{v}) := \mathbf{u}(\mathbf{v})^T\mathbf{v} - U(\mathbf{u}(\mathbf{v})), \quad \psi_m(\mathbf{v}) := \mathbf{g}_m(\mathbf{v})^T\mathbf{v} - F_m(\mathbf{u}(\mathbf{v})), \quad 1 \le m \le d, \tag{2.5}$$

so that $\phi'(\mathbf{v}) = \mathbf{u}(\mathbf{v})^T$ and $\psi'_m(\mathbf{v}) = \mathbf{g}_m(\mathbf{v})^T$. For a normal vector $\mathbf{n} \in \mathbb{R}^d$, we set $F_{\mathbf{n}} := \sum\limits_{m=1}^{d} n_m F_m$ and $\psi_{\mathbf{n}} := \sum\limits_{m=1}^{d} n_m \psi_m$.

Theorem 2.1. *U is a strictly convex entropy function if and only if $\mathbf{u}'(\mathbf{v})$ is symmetric positive-definite, and $\mathbf{g}'_m(\mathbf{v})$ is symmetric for each $1 \le m \le d$. Moreover, if U is a strictly convex entropy function, then* (2.1) *is hyperbolic.*

In the scalar case, any convex function is an entropy function, and there exists a unique entropy solution satisfying the well-posedness properties in Theorem 1.3. However, for general systems, there are very few global existence results, and Chiodaroli, De Lellis and Kreml [14] even found an counterexample showing the non-uniqueness of entropy solution. It is conjectured that measure-valued solutions might be the correct paradigm to describe multi-dimensional systems [33].

We now turn to numerical discretization. Suppose that we have a polygonal computational domain $\Gamma \in \mathbb{R}^d$ with periodic boundary condition. Let $\mathcal{K}_h := \{K_i\}_{i=1}^N$ be some domain decomposition of Γ, and h is the characteristic length of $\mathcal{K}_h$. For the sake of simplicity, we assume that all elements are simplices, and there is no hanging node in the mesh. The first order finite volume method is given by

$$\frac{d\mathbf{u}_i}{dt} + \frac{1}{|K_i|}\Big(\sum_{\gamma \in \partial K_i} |\gamma| \widehat{\mathbf{f}}_{\mathbf{n}}(\mathbf{u}_i, \mathbf{u}_i^{\gamma,\mathrm{out}})\Big) = 0, \tag{2.6}$$

where $\mathbf{n}$ is the outer normal vector, $\widehat{\mathbf{f}}_{\mathbf{n}}$ is some directional numerical flux function corresponding to the directional Riemann solver with $\mathbf{f}_{\mathbf{n}}$, and $\mathbf{u}_i^{\gamma,\mathrm{out}}$ denotes the numerical solution from the opposite that of γ. The finite volume method approximates the following integral form of (2.1):

$$\frac{d}{dt}\Big(\int_{K_i} \mathbf{u} d\mathbf{x}\Big) + \int_{\partial K} \mathbf{f}_{\mathbf{n}}(\mathbf{u}) dS = 0. \tag{2.7}$$

Entropy stability of (2.6) is again specified by entropy conservative fluxes and (directional) entropy stable fluxes.

Definition 2.3. *For $1 \leq m \leq d$, a numerical flux function $\mathbf{f}_{m,S}(\mathbf{u}_L, \mathbf{u}_R)$ in the m-th space dimension is called entropy conservative with respect to some entropy function U if it satisfies the following conditions:*

1. *Consistency:* $\mathbf{f}_{m,S}(\mathbf{u}, \mathbf{u}) = \mathbf{f}_m(\mathbf{u})$.
2. *Symmetry:* $\mathbf{f}_{m,S}(\mathbf{u}_L, \mathbf{u}_R) = \mathbf{f}_{m,S}(\mathbf{u}_R, \mathbf{u}_L)$.
3. *Entropy conservation:* $(\mathbf{v}_R - \mathbf{v}_L)^T \mathbf{f}_{m,S}(\mathbf{u}_L, \mathbf{u}_R) = \psi_{m,R} - \psi_{m,L}$.

Given entropy conservative fluxes in all space dimensions, we also set

$$\mathbf{f}_{\mathbf{n},S}(\mathbf{u}_L, \mathbf{u}_R) := \sum_{m=1}^{d} n_m \mathbf{f}_{m,S}(\mathbf{u}_L, \mathbf{u}_R)$$

Definition 2.4. *A directional numerical flux function $\widehat{\mathbf{f}}_{\mathbf{n}}(\mathbf{u}, \mathbf{u}^{out})$ is called entropy stable with respect to some entropy function U if it satisfies the following conditions:*

1. *Consistency:* $\widehat{\mathbf{f}}_{\mathbf{n}}(\mathbf{u}, \mathbf{u}) = \mathbf{f}_{\mathbf{n}}(\mathbf{u})$.
2. *Conservation (single-valuedness):* $\widehat{\mathbf{f}}_{\mathbf{n}}(\mathbf{u}, \mathbf{u}^{out}) = -\widehat{\mathbf{f}}_{-\mathbf{n}}(\mathbf{u}^{out}, \mathbf{u})$.
3. *Entropy stability:* $(\mathbf{v}^{out} - \mathbf{v})^T \widehat{\mathbf{f}}_{\mathbf{n}}(\mathbf{u}, \mathbf{u}^{out}) \leq \psi_{\mathbf{n}}^{out} - \psi_{\mathbf{n}}$.

Since entropy conservative are specific to each space dimension, and entropy conservative fluxes are specific to each normal vector, they can be derived in the same manner as in Section 1.3. As a consequence, the upwind numerical fluxes (monotone fluxes for scalar problems and Godunov type fluxes for general systems) are still entropy stable. In the scalar case, the finite volume method (2.6) with monotone fluxes at element interfaces also satisfies the well-posedness properties in Theorem 1.8, and the sequence of numerical solutions will converge to the unique entropy solution.

Example 2.1.1. Consider the two-dimensional Euler equations

$$\frac{\partial}{\partial t}\begin{bmatrix}\rho\\ \rho w_1\\ \rho w_2\\ E\end{bmatrix}+\frac{\partial}{\partial x_1}\begin{bmatrix}\rho w_1\\ \rho w_1^2+p\\ \rho w_1 w_2\\ w_1(E+p)\end{bmatrix}+\frac{\partial}{\partial x_2}\begin{bmatrix}\rho w_2\\ \rho w_1 w_2\\ \rho w_2^2+p\\ w_2(E+p)\end{bmatrix}=0. \tag{2.8}$$

Here, $\mathbf{w}=\begin{bmatrix}w_1 & w_2\end{bmatrix}^T$ is the velocity field. The equation of state is

$$E=\frac{1}{2}\rho(w_1^2+w_2^2)+\frac{p}{\gamma-1}. \tag{2.9}$$

The Euler equations are rotationally invariant. Under the change of coordinates $(x_1,x_2)\mapsto(x_{\mathbf{n}},x_{\mathbf{n}^\perp})$, where $\mathbf{n}\in\mathbb{R}^2$ is a normal vector, $x_{\mathbf{n}}:=n_1x_1+n_2x_2$, and $x_{\mathbf{n}^\perp}:=n_2x_1-n_1x_2$, (2.8) are converted into

$$\frac{\partial}{\partial t}\begin{bmatrix}\rho\\ \rho w_{\mathbf{n}}\\ \rho w_{\mathbf{n}^\perp}\\ E\end{bmatrix}+\frac{\partial}{\partial x_{\mathbf{n}}}\begin{bmatrix}\rho w_{\mathbf{n}}\\ \rho w_{\mathbf{n}}^2+p\\ \rho w_{\mathbf{n}} w_{\mathbf{n}^\perp}\\ w_{\mathbf{n}}(E+p)\end{bmatrix}+\frac{\partial}{\partial x_{\mathbf{n}^\perp}}\begin{bmatrix}\rho w_{\mathbf{n}^\perp}\\ \rho w_{\mathbf{n}} w_{\mathbf{n}^\perp}\\ \rho w_{\mathbf{n}^\perp}^2+p\\ w_{\mathbf{n}^\perp}(E+p)\end{bmatrix}=0. \tag{2.10}$$

We also set $w_{\mathbf{n}}:=n_1w_1+n_2w_2$ and $w_{\mathbf{n}^\perp}:=n_2w_1-n_1w_2$. With the physical specific entropy is $s=\log(p\rho^{-\gamma})$, $U=-\frac{\rho s}{\gamma-1}$ is still an entropy function of (2.8), such that the entropy fluxes are $F_1 = -\frac{\rho w_1 s}{\gamma-1}$ and $F_2 = -\frac{\rho w_2 s}{\gamma-1}$. The entropy variables, potential function and potential fluxes are given by

$$\mathbf{v} = \begin{bmatrix} \frac{\gamma-s}{\gamma-1} - \frac{\rho(w_1^2+w_2^2)}{2p} \\ \frac{\rho w_1}{p} \\ \frac{\rho w_2}{p} \\ -\frac{\rho}{p} \end{bmatrix}, \quad \phi = \rho, \quad \psi_1 = \rho w_1, \quad \psi_2 = \rho w_2. \tag{2.11}$$

Both the Ismail-Roe entropy conservative flux and the Chandrashekar entropy conservative flux have the two-dimensional version. The Chandrashekar's construction is

$$\begin{aligned}
&f^1_{1,S} = (\overline{\rho})^{\log}\overline{w_1}, \quad f^1_{2,S} = (\overline{\rho})^{\log}\overline{w_2}, \\
&f^2_{1,S} = \frac{\overline{\rho}}{2\overline{\beta}} + \overline{w_1} f^1_{1,S}, \quad f^2_{2,S} = \overline{w_1} f^1_{2,S}, \\
&f^3_{1,S} = \overline{w_2} f^1_{1,S}, \quad f^3_{2,S} = \frac{\overline{\rho}}{2\overline{\beta}} + \overline{w_2} f^1_{2,S}, \\
&f^4_{1,S} = \Big(\frac{1}{2(\gamma-1)(\overline{\beta})^{\log}} - \frac{\overline{w_1^2} + \overline{w_2^2}}{2}\Big) f^1_{1,S} + \overline{w_1} f^2_{1,S} + \overline{w_2} f^3_{1,S}, \\
&f^4_{2,S} = \Big(\frac{1}{2(\gamma-1)(\overline{\beta})^{\log}} - \frac{\overline{w_1^2} + \overline{w_2^2}}{2}\Big) f^1_{2,S} + \overline{w_1} f^2_{2,S} + \overline{w_2} f^3_{2,S}.
\end{aligned} \tag{2.12}$$

By rotational invariance, the Godunov flux and HLL flux for one-dimensional Euler equations can be directly used to build directional entropy stable fluxes.

The L^2 stability result of classic DG method can also be extended to higher space dimensions. In the DG method with polynomial degree k, numerical solutions and test functions both live in the space

$$\mathbf{V}_h^k = \{\mathbf{w}_h : \mathbf{w}_h|_{K_i} \in [\mathcal{P}^k(K_i)]^p, 1 \le i \le N\}. \tag{2.13}$$

We seek $\mathbf{u}_h \in \mathbf{V}_h^k$ such that for each $\mathbf{w}_h \in \mathbf{V}_h^k$ and $1 \leq i \leq N$,

$$\int_{K_i} \frac{\partial \mathbf{u}_h^T}{\partial t} \mathbf{w}_h d\mathbf{x} - \sum_{m=1}^{d} \int_{K_i} \mathbf{f}_m(\mathbf{u}_h)^T \frac{d\mathbf{w}_h}{dx_m} d\mathbf{x} = -\int_{\partial K_i} \widehat{\mathbf{f}_\mathbf{n}}(\mathbf{u}_h, \mathbf{u}_h^{\text{out}})^T \mathbf{w}_h dS. \tag{2.14}$$

The strong DG form is

$$\int_{K_i} \Big(\frac{\partial \mathbf{u}_h}{\partial t} + \sum_{m=1}^{d} \frac{\partial \mathbf{f}_m(\mathbf{u}_h)}{\partial x_m} \Big)^T \mathbf{w}_h d\mathbf{x} = \int_{\partial K_i} (\mathbf{f}_\mathbf{n}(\mathbf{u}_h) - \widehat{\mathbf{f}_\mathbf{n}}(\mathbf{u}_h, \mathbf{u}_h^{\text{out}}))^T \mathbf{w}_h dS. \tag{2.15}$$

Theorem 2.2. *If $U = \frac{1}{2}\mathbf{u}^T\mathbf{u}$ is an entropy function of (2.1), and $\widehat{\mathbf{f}_\mathbf{n}}$ is entropy stable with respect to U, then the DG scheme (2.14) and (2.15) is L^2 stable in the sense that*

$$\frac{d}{dt}\int_{\Gamma} U(\mathbf{u}_h) d\mathbf{x} = \frac{d}{dt}\Big(\frac{1}{2}\|\mathbf{u}_h\|_{L^2}^2\Big) \leq 0. \tag{2.16}$$

2.2 Multi-dimensional summation-by-parts operators

The summation-by-parts operators on simplicial meshes can be established in two steps. We first need to find a special Gauss-Lobatto type quadrature rule that also contains some boundary quadrature points. The algebraic degree of accuracy is at least $2k-1$ in the element, and at least $2k$ over the boundary. Then the difference matrices have to be carefully designed to achieve the SBP property. Without loss of generality, we only need to work on the reference simplex K with global coordinates $\boldsymbol{\xi} \in \mathbb{R}^d$, such that K_i is the image of K under some affine mapping $\boldsymbol{\xi} \mapsto \mathbf{x}_i(\boldsymbol{\xi})$. One example of the global reference simplex is

$$K = \Big\{ \boldsymbol{\xi} : \xi_m \geq 0 \text{ for each } 1 \leq m \leq d, \quad \sum_{m=1}^{d} \xi_m \leq 1 \Big\}, \tag{2.17}$$

Given polynomial degree k, the dimension of $\mathcal{P}^k(K)$ is

$$\mathcal{N}_{P,k} := \binom{k+d}{d}.$$

Suppose that there is a degree $2k-1$ quadrature rule on K, associated with $\mathcal{N}_{Q,k} \geq \mathcal{N}_{P,k}$ nodes $\{\boldsymbol{\xi}_j\}_{j=1}^{\mathcal{N}_{Q,k}}$, and positive weights $\{\omega_j\}_{j=1}^{\mathcal{N}_{Q,k}}$. Some quadrature points should be on ∂K, and these points form a degree $2k$ quadrature rule over the boundary, with positive weights $\{\tau_j\}_{j=1}^{\mathcal{N}_{Q,k}}$. We also let $\mathbf{n}_j$ be the outer normal vector at $\boldsymbol{\xi}_j$. If $\boldsymbol{\xi}_j \notin \partial T$ is an interior point, we require that $\tau_j = \mathbf{n}_j = 0$. The continuous and discrete inner product on K and ∂K are defined as

$$(\mathbf{u}, \mathbf{v}) := \int_K \mathbf{u}^T \mathbf{v} d\mathbf{x}, \quad (\mathbf{u}, \mathbf{v})_\omega := \sum_{j=1}^{\mathcal{N}_{Q,k}} \omega_j \mathbf{u}(\boldsymbol{\xi}_j)^T \mathbf{v}(\boldsymbol{\xi}_j) \tag{2.18}$$

$$\langle \mathbf{u}, \mathbf{v} \rangle := \int_{\partial K} \mathbf{u}^T \mathbf{v} dS, \quad \langle \mathbf{u}, \mathbf{v} \rangle_\tau := \sum_{j=1}^{\mathcal{N}_{Q,k}} \tau_j \mathbf{u}(\boldsymbol{\xi}_j)^T \mathbf{v}(\boldsymbol{\xi}_j) \tag{2.19}$$

Discrete operators are based on nodes $\{\boldsymbol{\xi}_j\}_{j=1}^{\mathcal{N}_{Q,k}}$. The vector notation of nodal function is again adopted. The restriction of function u on quadrature points is denoted by

$$\overrightarrow{u} := \begin{bmatrix} u(\boldsymbol{\xi}_1) & \cdots & u(\boldsymbol{\xi}_{\mathcal{N}_{Q,k}}) \end{bmatrix}^T.$$

We define mass matrix M and boundary matrix B

$$M := \operatorname{diag}\{\omega_1, \cdots, \omega_{\mathcal{N}_{Q,k}}\}, \quad B := \operatorname{diag}\{\tau_1, \cdots, \tau_{\mathcal{N}_{Q,k}}\}, \tag{2.20}$$

such that

$$\overrightarrow{u}^T M \overrightarrow{v} = (u, v)_\omega, \quad \overrightarrow{u}^T B \overrightarrow{v} = \langle u, v \rangle_\tau,$$

and the diagonal matrices of outer normal vectors

$$N_m := \operatorname{diag}\{n_{1,m}, \cdots, n_{\mathcal{N}_{Q,k},m}\}, \quad 1 \leq m \leq d. \tag{2.21}$$

Let $\{p_l(\boldsymbol{\xi})\}_{l=1}^{\mathcal{N}_{P,k}}$ be a set of basis functions of $\mathcal{P}^k(K)$. The $\mathcal{N}_{Q,k} \times \mathcal{N}_{P,k}$ Vandermonde matrix V consists of columns of nodal values of basis functions

$$V := \{p_l(\boldsymbol{\xi}_j)\}_{1 \leq j \leq \mathcal{N}_{Q,k}, 1 \leq l \leq \mathcal{N}_{P,k}}. \tag{2.22}$$

Since $\mathcal{N}_{Q,k} \geq \mathcal{N}_{P,k}$, the Vandermonde matrix is not always invertible. We still have its pseudo-inverse under norm M, i.e., the quadrature-based projection matrix into $\mathcal{P}^k(K)$

$$P := (V^T M V)^{-1} V^T M. \tag{2.23}$$

Derivatives of polynomials in $\mathcal{P}^k(K)$ still belong to $\mathcal{P}^k(K)$. We set the (modal) $\mathcal{N}_{P,k} \times \mathcal{N}_{P,k}$ differentiation matrices $\widehat{D}_m$ for $1 \leq m \leq d$, such that

$$\frac{\partial p_l}{\partial \xi_m}(\boldsymbol{\xi}) = \sum_{r=1}^{\mathcal{N}_{P,k}} \widehat{D}_{m,rl} p_r(\boldsymbol{\xi}).$$

Hence

$$(V\widehat{D}_m)_{jl} = \sum_{r=1}^{\mathcal{N}_{P,k}} p_r(\boldsymbol{\xi}_j) \widehat{D}_{m,rl} = \frac{\partial p_l}{\partial \xi_m}(\boldsymbol{\xi}_j).$$

In other words, $V\widehat{D}_m$ is the Vandermonde matrix of the m-th partial derivative of basis functions. The existence of nodal difference matrices with the SBP property will be clarified in the following theorem.

Theorem 2.3. *We compute the $\mathcal{N}_{Q,k} \times \mathcal{N}_{Q,k}$ difference matrices D_m for each $1 \leq m \leq d$, using the formula*

$$D_m := \frac{1}{2} M^{-1} (I + VP)^T B N_m (I - VP) + V\widehat{D}_m P. \tag{2.24}$$

Then these difference matrices satisfy the two properties below:

1. *Exactness: if* $u \in \mathcal{P}^k(K)$*,* $D_m \overrightarrow{u}$ *contains nodal values of* $\frac{\partial u}{\partial \xi_m}$*; that is,*

$$D_m V = V \widehat{D}_m. \tag{2.25}$$

2. *Summation-by-parts: set the stiffness matrix* $S_m := M D_m$*. Then we have*

$$BN_m = S_m + S_m^T = M D_m + D_m^T M. \tag{2.26}$$

Remark 2.1. By the exactness property and SBP property,

$$S_m \overrightarrow{1} = D_m \overrightarrow{1} = \overrightarrow{0}, \quad S_m^T \overrightarrow{1} = BN_m \overrightarrow{1}. \tag{2.27}$$

Proof. Since $PV = I$, $(I - VP)V = 0$, and

$$D_m V = V \widehat{D}_m = V \widehat{D}_m P V = V \widehat{D}_m.$$

As for the SBP property,

$$\begin{aligned} S_m = M D_m &= \frac{1}{2}(I + VP)^T BN_m (I - VP) + MV\widehat{D}_m P \\ &= \frac{1}{2} BN_m + \frac{1}{2}(P^T V^T BN_m - BN_m VP) + \left(MV\widehat{D}_m P - \frac{1}{2} P^T V^T BN_m VP \right) \end{aligned}$$

Let us check the entries of V^TBN_mV:

$$\begin{aligned}(V^TBN_mV)_{lr} &= \sum_{j=1}^{\mathcal{N}_{Q,k}} \tau_j n_{j,m} p_l(\boldsymbol{\xi}_j) p_r(\boldsymbol{\xi}_j) = \langle p_l, p_r n_m\rangle_\tau = \langle p_l, p_r n_m\rangle \\ &= \Big(p_l, \frac{\partial p_r}{\partial \xi_m}\Big) + \Big(\frac{\partial p_l}{\partial \xi_m}, p_r\Big) = \Big(p_l, \frac{\partial p_r}{\partial \xi_m}\Big)_\omega + \Big(\frac{\partial p_l}{\partial \xi_m}, p_r\Big)_\omega \\ &= \sum_{j=1}^{\mathcal{N}_{Q,k}} \omega_j p_l(\boldsymbol{\xi}_j) \frac{\partial p_r}{\partial \xi_m}(\boldsymbol{\xi}_j) + \sum_{j=1}^{\mathcal{N}_{Q,k}} \omega_j \frac{\partial p_l}{\partial \xi_m}(\boldsymbol{\xi}_j) p_r(\boldsymbol{\xi}_j) \\ &= (V^TMV\widehat{D}_m + \widehat{D}_m^T V^T MV)_{lr},\end{aligned}$$

where we use integration by parts, and the algebraic degree of accuracy of $(\cdot,\cdot)$ and $\langle\cdot,\cdot\rangle$. Therefore

$$\begin{aligned}S_m &= \frac{1}{2}BN_m + \frac{1}{2}(P^TV^TBN_m - BN_mVP) + \Big(MV\widehat{D}_mP - \frac{1}{2}P^T(V^TMV\widehat{D}_m + \widehat{D}_m^TV^TMV)P\Big) \\ &= \frac{1}{2}BN_m + \frac{1}{2}(P^TV^TBN_m - BN_mVP) + \frac{1}{2}(MV\widehat{D}_mP - P^T\widehat{D}_m^TV^TM),\end{aligned}$$

due to the identity $(V^TMV)P = V^TM$. Summing S_m and its transpose, the first term becomes BN_m, and the second and third term will vanish. This completes our proof. □

Remark 2.2. For one-dimensional Gauss-Lobatto quadrature rule, $\mathcal{N}_{P,k} = \mathcal{N}_{Q,k} = k+1$ and V is invertible. We simply take $D_m = V^{-1}V\widehat{D}_m$. In general we need more than $\mathcal{N}_{P,k}$ nodes to accomplish the quadrature rule, which complicates the derivation of difference matrices.

Discrete operators for nodal values of vector-valued functions are again understood as Kronecker products

$$\mathbf{M} = M \otimes I_p, \quad \mathbf{D}_m = D_m \otimes I_p, \quad \mathbf{S}_m = S_m \otimes I_p, \quad \mathbf{B} = B \otimes I_p, \quad \mathbf{N}_m = N_m \otimes I_p.$$

For a local simplex element $T_i \in \mathcal{T}_h$, let $J_i := \det(\mathbf{x}_i'(\boldsymbol{\xi}))$ be the global-to-local

Jacobian factor, and $G_i := \boldsymbol{\xi}'(\mathbf{x}_i)$ be the inverse of Jacobian matrix. $J^b_{i,j}$ denotes the Jacobian factor of face mapping at $\boldsymbol{\xi}_j$ such that $J^b_{i,j} = 0$ for $\boldsymbol{\xi}_j \notin \partial K$, and

$$\mathbf{J}^b_i := \mathrm{diag}\{J^b_{i,1}, \cdots, J^b_{i,\mathcal{N}_{Q,k}}\} \otimes I_p.$$

Then the local discrete operators are

$$\mathbf{M}_i = J_i\mathbf{M}, \quad \mathbf{D}_{i,m} = \sum_{r=1}^{d} G_{i,rm}\mathbf{D}_r, \quad \mathbf{B}_i = \mathbf{J}^b_i\mathbf{B}. \tag{2.28}$$

These geometric factors satisfy the following identity [9].

$$J_i\Big(\sum_{r=1}^{d} G_{i,rm}n_{j,r}\Big) = J^b_{i,j}n_{i,j,m}, \tag{2.29}$$

where $\mathbf{n}_{i,j}$ is the j-th outer normal vector on K_i. Hence we still have the local SBP property

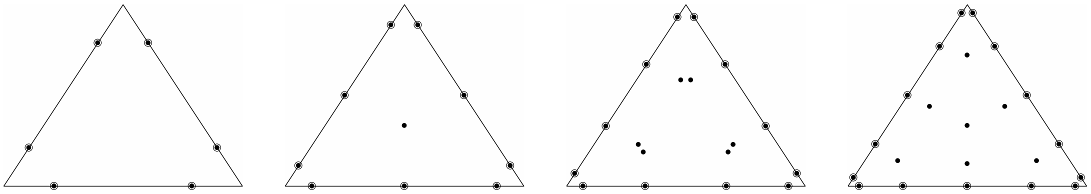
$$\mathbf{B}_i\mathbf{N}_{i,m} = \mathbf{S}_{i,m} + \mathbf{S}^T_{i,m} = \mathbf{M}_i\mathbf{D}_{i,m} + \mathbf{D}^T_{i,m}\mathbf{M}_i, \quad 1 \le m \le d. \tag{2.30}$$

Remark 2.3. Conceptually, the SBP framework can be further generalized to arbitrary polygonal meshes without any difficulty. We stick to simplicial meshes for practical purposes. We only need to store one set of discrete operators, and the local operators are acquired through the global-to-local mapping. This is efficient in terms of space complexity, especially for meshes with a large number of elements.

The remaining part of this section is devoted to the implementation of SBP operators on two-dimensional triangular meshes. We need to find the two-dimensional quadrature rule that achieves interior and boundary accuracy simultaneously. For boundary accuracy, we put $k+1$ Legendre-Gauss points along each edge. Let us summarize the prerequisites of the quadrature rule:

1. It is symmetric so that adjacent elements can be glued together.
2. The quadrature weights should be positive to make M positive-definite.
3. It is exact for polynomials up to degree $2k-1$.
4. The quadrature points include $k+1$ Legendre-Gauss points on each edge.

Quadrature rules that meet these requirements are investigated in the literature. We use the software presented in [109] to obtain the rules of order $k = 1, 2, 3$ and 4[1]. The distribution of quadrature points are illustrated in Figure 2.1. We call them *B-type* quadrature rules. The letter B indicates the restriction that these quadrature rules must contain boundary quadrature points. For reference, we also list the coordinates of quadrature points and their weights in Appendix C.



(a) $k = 1, \mathcal{N}_{Q,k} = 6$ (b) $k = 2, \mathcal{N}_{Q,k} = 10$ (c) $k = 3, \mathcal{N}_{Q,k} = 18$ (d) $k = 4, \mathcal{N}_{Q,k} = 22$

Figure 2.1: B-type quadrature rules on triangles with $k = 1, 2, 3, 4$. We use equilateral triangles to emphasize symmetry. Dots are quadrature points for the triangle, and circles are quadrature points for the edges. The symbols overlap because boundary nodes play both roles.

Remark 2.4. The same requirements also arise in [114] where the authors tried to implement bound-preserving limiter on triangular meshes. They proposed a generic quadrature rule based on three warped transformation from a unit square to triangles. However, $\mathcal{N}_{Q,k} = 3k(k+1)$ for such technique, which is unnecessarily large.

Then we compute the entries of difference matrices according to 2.24. We use the orthonormal set of polynomials on triangles [61] as basis functions. It is not an

[1] http://lsec.cc.ac.cn/phg/download/quadrule.tar.bz2

orthonormal basis under $(\cdot,\cdot)_\omega$ as the quadrature rule is not exact for polynomials of degree $2k$. However, the condition number of the Vandermonde matrix will be small enough to prevent large rounding error. For $k = 1, 2$, it is possible to use symbolic computation software to obtain the exact values of difference matrices.

2.3 Multi-dimensional entropy stable DG method

For clarity of notations, we explain the entropy stable DGSEM on the reference element and omit the subscript i. Numerical solution collocated at the $\mathcal{N}_{Q,k}$ quadrature points will be evolved. $\overrightarrow{\mathbf{u}}$ denotes the numerical solution, and $\overrightarrow{\mathbf{f}^*_{\mathbf{n}}}$ stands for the vector of directional fluxes on element interfaces such that

$$\mathbf{f}^*_{\mathbf{n},j} = \begin{cases} \widehat{\mathbf{f}}_{\mathbf{n}}(\mathbf{u}_j, \mathbf{u}_j^{\text{out}}) & \boldsymbol{\xi}_j \in \partial T \\ 0 & \boldsymbol{\xi}_j \notin \partial T \end{cases}.$$

We develop the multi-dimensional entropy stable DGSEM by doing flux differencing in each space dimension:

$$\frac{d\overrightarrow{\mathbf{u}}}{dt} + 2\sum_{m=1}^{d} \mathbf{D}_m \circ \mathbf{F}_{m,S}(\overrightarrow{\mathbf{u}}, \overrightarrow{\mathbf{u}})\overrightarrow{\mathbf{1}} = \mathbf{M}^{-1}\mathbf{B}(\overrightarrow{\mathbf{f}_{\mathbf{n}}} - \overrightarrow{\mathbf{f}^*_{\mathbf{n}}}), \tag{2.31}$$

where

$$\mathbf{F}_{m,S}(\overrightarrow{\mathbf{u}_L}, \overrightarrow{\mathbf{u}_R}) := \begin{bmatrix} \text{diag}(\mathbf{f}_{m,S}(\mathbf{u}_{L,1}, \mathbf{u}_{R,1})) & \cdots & \text{diag}(\mathbf{f}_{m,S}(\mathbf{u}_{L,1}, \mathbf{u}_{R,\mathcal{N}_{Q,k}})) \\ \vdots & \ddots & \vdots \\ \text{diag}(\mathbf{f}_{m,S}(\mathbf{u}_{L,\mathcal{N}_{Q,k}}, \mathbf{u}_{R,1})) & \cdots & \text{diag}(\mathbf{f}_{m,S}(\mathbf{u}_{L,\mathcal{N}_{Q,k}}, \mathbf{u}_{R,\mathcal{N}_{Q,k}})) \end{bmatrix}.$$

The component-wise form of (2.31) is

$$\frac{d\mathbf{u}_j}{dt} + 2\sum_{m=1}^{d}\sum_{l=1}^{\mathcal{N}_{Q,k}} D_{m,jl}\mathbf{f}_{m,S}(\mathbf{u}_j, \mathbf{u}_l) = \frac{\tau_j}{\omega_j}(\mathbf{f}_{\mathbf{n},j} - \mathbf{f}^*_{\mathbf{n},j}) \tag{2.32}$$

The main properties of the (2.31) are outlined in the following theorem.

Theorem 2.4. *Assume that the sequence of meshes $\{\mathcal{T}_h\}$, parameterized by h, is uniform, and all mappings are bivariate fluxes are smooth and Lipschitz continuous. If $\mathbf{f}_{m,S}$ is entropy conservative for each $1 \leq m \leq d$, and $\widehat{\mathbf{f}}_{\mathbf{n}}$ is entropy stable, then the scheme (2.31) is high order accurate in the sense that for all i, j and smooth solution $\mathbf{u}$ of (2.1), the local truncation error*

$$\frac{d\mathbf{u}_{i,j}}{dt} + 2\sum_{m=1}^{d}\sum_{l=0}^{k} D_{i,m,jl}\mathbf{f}_{m,S}(\mathbf{u}_{i,j}, \mathbf{u}_{i,l}) - \frac{J^b_{i,j}\tau_j}{J_i\omega_j}(\mathbf{f}_{i,\mathbf{n},j} - \mathbf{f}^*_{i,\mathbf{n},j}) = \mathcal{O}(h^k), \tag{2.33}$$

and conservative and entropy stable in the sense that

$$\frac{d}{dt}\Big(\sum_{i=1}^{N} \vec{\mathbf{1}}^T\mathbf{M}_i\vec{\mathbf{u}}_i\Big) = 0, \quad \frac{d}{dt}\Big(\sum_{i=1}^{N} \vec{1}^T M_i\vec{U}_i\Big) \leq 0. \tag{2.34}$$

Proof. We only present the sketch of proof as it is almost the same as the proof of Theorem 1.12. For accuracy, we use the approximation property on uniform meshes to show that

$$2\sum_{l=0}^{k} D_{i,m,jl}\mathbf{f}_{m,S}(\mathbf{u}_{i,j}, \mathbf{u}_{i,l}) - \frac{\partial\mathbf{f}_m(\mathbf{u})}{\partial x_m}(\mathbf{x}_i(\boldsymbol{\xi}_j)) = \mathcal{O}(h^k).$$

As for conservation and entropy stability, we have the same local conservation and entropy balance result as in Lemma 1.1

$$\frac{d}{dt}(\vec{\mathbf{1}}^T\mathbf{M}\vec{\mathbf{u}}) = -\vec{\mathbf{1}}^T\mathbf{B}\vec{\mathbf{f}}^*_{\mathbf{n}} = -\sum_{j=1}^{\mathcal{N}_{Q,k}} \tau_j\widehat{\mathbf{f}}_{\mathbf{n}}(\mathbf{u}_j, \mathbf{u}_j^{\text{out}}), \tag{2.35}$$

$$\frac{d}{dt}(\overrightarrow{1}^T M \overrightarrow{U}) = \overrightarrow{\psi_{\mathbf{n}}}^T B \overrightarrow{1} - \overrightarrow{\mathbf{v}} \mathbf{M} \overrightarrow{\mathbf{f}_{\mathbf{n}}^*} = \sum_{j=1}^{\mathcal{N}_{Q,k}} \tau_j (\psi_{\mathbf{n},j} - \mathbf{v}_j^T \widehat{\mathbf{f}}_{\mathbf{n}}(\mathbf{u}_j, \mathbf{u}_j^{\text{out}})). \qquad (2.36)$$

Then at a boundary node $\boldsymbol{\xi}_j \in \partial T$, since $\widehat{\mathbf{f}}_{\mathbf{n}}(\mathbf{u}_j, \mathbf{u}_j^{\text{out}})$ and $\widehat{\mathbf{f}}_{-\mathbf{n}}(\mathbf{u}_j^{\text{out}}, \mathbf{u}_j)$ cancel out, we prove the global conservation. The entropy production rate at $\boldsymbol{\xi}_j$ is

$$\begin{aligned} &\tau_j(\psi_{\mathbf{n},j} + \psi_{-\mathbf{n},j}^{\text{out}} - \mathbf{v}_j^T \widehat{\mathbf{f}}_{\mathbf{n}}(\mathbf{u}_j, \mathbf{u}_j^{\text{out}}) - (\mathbf{v}_j^{\text{out}})^T \widehat{\mathbf{f}}_{-\mathbf{n}}(\mathbf{u}_j^{\text{out}}, \mathbf{u}_j)) \\ =&\tau_j(\psi_{\mathbf{n},j} - \psi_{\mathbf{n},j}^{\text{out}} - (\mathbf{v}_j - \mathbf{v}_j^{\text{out}})^T \widehat{\mathbf{f}}_{\mathbf{n}}(\mathbf{u}_j, \mathbf{u}_j^{\text{out}})) \leq 0. \end{aligned}$$

Therefore we also have entropy dissipation. □

Remark 2.5. The multi-dimensional bound-preserving limiter is again a simple linear scaling procedure $\widetilde{\mathbf{u}}_j := \overline{\mathbf{u}} + \theta(\mathbf{u}_j - \overline{\mathbf{u}})$. It can be imposed naturally without affecting entropy stability. However, it is very challenging to design entropy stable TVD/TVB limiters.

Remark 2.6. The link between the entropy stable DGSEM and the classic DG method looks vague due to the fact that the degree of freedom ($\mathcal{N}_{Q,k}$) is larger than the dimension of the underlying polynomial space ($\mathcal{N}_{P,k}$). These issues will be addressed in Section 3.2 and in Section 3.7.

2.4 Entropy stability of wall boundary condition

So far we have always assumed periodic boundary condition. There is a need to investigate the solid wall boundary condition of two-dimensional Euler equations (2.8). We will prove that the commonly used mirror state treatment is entropy stable. This section extends the one-dimensional analysis in [93].

At the wall boundary, we prescribe the no penetration condition; that is,

$$w_{\mathbf{n}} = w_1 n_1 + w_2 n_2 = 0 \tag{2.37}$$

Suppose that we have a numerical state $\mathbf{u}$ on the solid wall. In order to weakly impose the no penetration condition, we have to provide an artificial state $\mathbf{u}^{\text{out}}$ on the other side of the interface, and compute the numerical flux $\widehat{\mathbf{f}}_{\mathbf{n}}(\mathbf{u}, \mathbf{u}^{\text{out}})$. The reflecting technique introduces a mirror state such that

$$\rho^{\text{out}} = \rho, \quad p^{\text{out}} = p, \quad w_{\mathbf{n}}^{\text{out}} = -w_{\mathbf{n}}, \quad w_{\mathbf{n}^\perp}^{\text{out}} = w_{\mathbf{n}^\perp}. \tag{2.38}$$

The following theorem affirms the entropy stability of the reflecting technique.

Theorem 2.5. *If $\widehat{\mathbf{f}}_{\mathbf{n}}(\mathbf{u}, \mathbf{u}^{out})$ is Godunov flux or HLL flux where $\mathbf{u}^{out}$ is taken to be the mirror state (2.38), then such boundary treatment is entropy stable.*

Proof. According to (2.36), we need to prove that the entropy production rate at the interface

$$\psi_{\mathbf{n}} - \mathbf{v}^T \widehat{\mathbf{f}}_{\mathbf{n}}(\mathbf{u}, \mathbf{u}^{\text{out}})$$

is non-positive. By rotational symmetry, it it enough to consider the vertical wall $x_1 = 0$. Then $\mathbf{n} = \begin{bmatrix} 1 & 0 \end{bmatrix}^T$ and

$$\mathbf{u} = \begin{bmatrix} \rho & \rho w_1 & \rho w_2 & E \end{bmatrix}^T, \quad \mathbf{u}^{\text{out}} = \begin{bmatrix} \rho & -\rho w_1 & \rho w_2 & E \end{bmatrix}^T.$$

The numerical flux simply solves the Riemann problem in the first dimension. The exact Riemann solver will give a middle state $\mathbf{u}^*$ such that $w_1^* = 0$. Hence the Godunov flux is

$$\widehat{\mathbf{f}}_{\mathbf{n}}(\mathbf{u}, \mathbf{u}^{\text{out}}) = \mathbf{f}_1(\mathbf{u}^*) = \begin{bmatrix} 0 & p^* & 0 & 0 \end{bmatrix}^T. \tag{2.39}$$

For the HLL Riemann solver, the two-rarefaction approximation yields $\lambda_L = -\lambda$ and $\lambda_R = \lambda$. Then we actually have the local Lax-Friedrichs flux

$$\widehat{\mathbf{f}}_{\mathbf{n}}(\mathbf{u}, \mathbf{u}^{\text{out}}) = \frac{1}{2}(\mathbf{f}_1(\mathbf{u}) + \mathbf{f}_1(\mathbf{u}^{\text{out}})) - \frac{\lambda}{2}(\mathbf{u}^{\text{out}} - \mathbf{u}) = \begin{bmatrix} 0 & p + \lambda\rho w_1 & 0 & 0 \end{bmatrix}^T. \quad (2.40)$$

In both cases only the second component of $\widehat{\mathbf{f}}_{\mathbf{n}}$ is nonzero. On the other hand, since

$$\mathbf{v} = \begin{bmatrix} \frac{\gamma-s}{\gamma-1} - \frac{\rho(w_1^2+w_2^2)}{2p} & \frac{\rho w_1}{p} & \frac{\rho w_2}{p} & -\frac{\rho}{p} \end{bmatrix}^T, \quad \psi_{\mathbf{n}} = \rho w_1,$$

$$\mathbf{v}^{\text{out}} = \begin{bmatrix} \frac{\gamma-s}{\gamma-1} - \frac{\rho(w_1^2+w_2^2)}{2p} & -\frac{\rho w_1}{p} & \frac{\rho w_2}{p} & -\frac{\rho}{p} \end{bmatrix}^T, \quad \psi_{\mathbf{n}}^{\text{out}} = -\rho w_1,$$

we can easily verify that

$$\psi_{\mathbf{n}} - \mathbf{v}^T \widehat{\mathbf{f}}(\mathbf{u}, \mathbf{u}^{\text{out}}, \mathbf{n}) = (\mathbf{v}^{\text{out}})^T \widehat{\mathbf{f}}(\mathbf{u}, \mathbf{u}^{\text{out}}, \mathbf{n}) - \psi_{\mathbf{n}}^{\text{out}} = \frac{1}{2}((\mathbf{v}^{\text{out}} - \mathbf{v})^T \widehat{\mathbf{f}}(\mathbf{u}, \mathbf{u}^{\text{out}}, \mathbf{n}) - (\psi_{\mathbf{n}}^{\text{out}} - \psi_{\mathbf{n}})).$$

It is non-positive due to the entropy stability of Godunov flux and HLL flux. □

2.5 Convection-diffusion equations

In this section, we add viscous diffusive terms to the hyperbolic conservation law (2.1). The convection-diffusion equations are given by:

$$\frac{\partial \mathbf{u}}{\partial t} + \sum_{m=1}^{d} \frac{\partial}{\partial x_m}(\mathbf{f}_m(\mathbf{u}) - \sum_{r=1}^{d} C_{mr}(\mathbf{v}) \frac{\partial \mathbf{v}}{\partial \mathbf{x}_r}) = 0, \quad (2.41)$$

where $\mathbf{v}$ is the entropy variable of some entropy function U, and $C_{mr}(\mathbf{v})$ are $p \times p$ matrix-valued functions. One typical examples is the compressible Navier-Stokes equations. Recall the vanishing viscosity approach in Section 1.2. We assume that the matrix

$$\begin{bmatrix} C_{11}(\mathbf{v}) & \cdots & C_{1d}(\mathbf{v}) \\ \vdots & \ddots & \vdots \\ C_{d1}(\mathbf{v}) & \cdots & C_{dd}(\mathbf{v}) \end{bmatrix}$$

is symmetric semi-positive-definite. Then (2.41) supports the entropy condition with respect to U. The convective part will be handled by (2.31). For the diffusive part, we present a nodal version of the LDG method of Cockburn and Shu [18], with provable entropy stability. We recast (2.41) into the mixed formulation

$$\frac{\partial \mathbf{u}}{\partial t} + \sum_{m=1}^{d} \frac{\partial}{\partial x_m}(\mathbf{f}_m(\mathbf{u}) - \mathbf{q}_m) = 0, \quad \mathbf{q}_m = \sum_{r=1}^{d} C_{mr}(\mathbf{v})\boldsymbol{\theta}_r, \quad \boldsymbol{\theta}_r = \frac{\partial \mathbf{v}}{\partial x_r}. \tag{2.42}$$

The LDG type method evolves the nodal discretization of $\mathbf{u}$ and $\{\boldsymbol{\theta}_r\}_{r=1}^d$ simultaneously. The coupling between adjoining elements are achieved by $\widehat{\mathbf{f}}_{\mathbf{n}}(\mathbf{u}, \mathbf{u}^{\text{out}})$ and single-valued numerical fluxes of $\mathbf{v}$ and $\mathbf{q}_{\mathbf{n}}$:

$$\widehat{\mathbf{v}} := \widehat{\mathbf{v}}(\mathbf{v}, \mathbf{v}^{\text{out}}), \quad \widehat{\mathbf{q}}_{\mathbf{n}} := \widehat{\mathbf{q}}_{\mathbf{n}}(\mathbf{v}, \mathbf{v}^{\text{out}}, \mathbf{q}_{\mathbf{n}}, \mathbf{q}_{\mathbf{n}}^{\text{out}}). \tag{2.43}$$

Once again $\overrightarrow{\mathbf{u}}$ and $\overrightarrow{\boldsymbol{\theta}}_r$ denote the nodal values of numerical solutions in the reference element. We further define

$$\mathbf{C}_{mr} := \text{diag}\{C_{mr}(\mathbf{v}_1), \cdots, C_{mr}(\mathbf{v}_{\mathcal{N}_{Q,k}})\}, \quad \overrightarrow{\mathbf{q}_m} := \sum_{r=1}^{d} \mathbf{C}_{mr}\overrightarrow{\boldsymbol{\theta}_r}.$$

Additionally, we also let $\overrightarrow{\mathbf{v}^*}$ and $\overrightarrow{\mathbf{q}_{\mathbf{n}}^*}$ describe the vectors of corresponding numerical fluxes. Then the LDG method is

$$\frac{d\overrightarrow{\mathbf{u}}}{dt} + 2\sum_{m=1}^{d}\Big(\mathbf{D}_m \circ \mathbf{F}_{m,S}(\overrightarrow{\mathbf{u}}, \overrightarrow{\mathbf{u}})\overrightarrow{\mathbf{1}} - \mathbf{D}_m\overrightarrow{\mathbf{q}_m}\Big) = \mathbf{M}^{-1}\mathbf{B}\Big((\overrightarrow{\mathbf{f}_{\mathbf{n}}} - \overrightarrow{\mathbf{f}_{\mathbf{n}}^*}) - (\overrightarrow{\mathbf{q}_{\mathbf{n}}} - \overrightarrow{\mathbf{q}_{\mathbf{n}}^*})\Big), \tag{2.44a}$$

$$\overrightarrow{\boldsymbol{\theta}_r} - \mathbf{D}_r\overrightarrow{\mathbf{v}} = -\mathbf{M}^{-1}\mathbf{B}\mathbf{N}_r(\overrightarrow{\mathbf{v}} - \overrightarrow{\mathbf{v}^*}), \quad 1 \le r \le d. \tag{2.44b}$$

Such scheme is entropy stable for carefully chosen fluxes of $\mathbf{v}$ and $\mathbf{q_n}$.

Theorem 2.6. *Given parameters $\alpha \ge 0$ and $\beta \in \mathbb{R}$, if we use the LDG fluxes*

$$\begin{aligned} &\widehat{\mathbf{v}}(\mathbf{v}, \mathbf{v}^{out}) = \frac{1}{2}(\mathbf{v} + \mathbf{v}^{out}) + \beta(\mathbf{v} - \mathbf{v}^{out}), \\ &\widehat{\mathbf{q}}_\mathbf{n}(\mathbf{v}, \mathbf{v}^{out}, \mathbf{q_n}, \mathbf{q}_\mathbf{n}^{out}) = \frac{1}{2}(\mathbf{q_n} + \mathbf{q}_\mathbf{n}^{out}) - \beta(\mathbf{q_n} - \mathbf{q}_\mathbf{n}^{out}) - \alpha(\mathbf{v}_j - \mathbf{v}_j^{out}), \end{aligned} \tag{2.45}$$

then (2.44) *is entropy stable.*

Proof. We left multiply (2.44a) by $\overrightarrow{\mathbf{v}}^T\mathbf{M}$ and (2.44b) by $\overrightarrow{\mathbf{q}_r}^T\mathbf{M}$, and sum them up. The convective part is already entropy stable according to Theorem 2.4. The remaining terms are

$$\begin{aligned} &\sum_{r=1}^{d}\Big(-\overrightarrow{\mathbf{q}_r}^T\mathbf{M}\overrightarrow{\boldsymbol{\theta}_r} + \overrightarrow{\mathbf{v}}^T\mathbf{M}\mathbf{D}_r\overrightarrow{\mathbf{q}_r} + \overrightarrow{\mathbf{q}_r}^T\mathbf{M}\mathbf{D}_r\overrightarrow{\mathbf{v}}\Big) - \overrightarrow{\mathbf{v}}^T\mathbf{B}(\overrightarrow{\mathbf{q_n}} - \overrightarrow{\mathbf{q_n}}^*) - \overrightarrow{\mathbf{q_n}}^T\mathbf{B}(\overrightarrow{\mathbf{v}} - \overrightarrow{\mathbf{v}^*}) \\ = &-\sum_{r=1}^{d}\overrightarrow{\mathbf{q}_r}^T\mathbf{M}\overrightarrow{\boldsymbol{\theta}_r} + (\overrightarrow{\mathbf{v}}^T\mathbf{B}\overrightarrow{\mathbf{q}_\mathbf{n}^*} + \overrightarrow{\mathbf{q_n}}^T\mathbf{B}\overrightarrow{\mathbf{v}^*} - \overrightarrow{\mathbf{v}}^T\mathbf{B}\overrightarrow{\mathbf{q_n}}). \end{aligned}$$

The first sum is the interior contribution, it is non-positive since

$$-\sum_{r=1}^{d}\overrightarrow{\mathbf{q}_r}^T\mathbf{M}\overrightarrow{\boldsymbol{\theta}_r} = -\sum_{j=1}^{\mathcal{N}_{Q,k}}\omega_j\Big(\sum_{r=1}^{d}\mathbf{q}_{r,j}^T\boldsymbol{\theta}_{r,j}\Big) = -\sum_{j=1}^{\mathcal{N}_{Q,k}}\omega_j\Big(\sum_{m=1}^{d}\sum_{r=1}^{d}\boldsymbol{\theta}_{m,j}^T C_{mr}(\mathbf{v}_j)\boldsymbol{\theta}_{r,j}\Big) \le 0.$$

The boundary contribution reduces to

$$\sum_{j=1}^{\mathcal{N}_{Q,k}}\tau_j\Big(\mathbf{v}_j^T\widehat{\mathbf{q}}_\mathbf{n}(\mathbf{v}_j, \mathbf{v}_j^{\text{out}}, \mathbf{q}_{\mathbf{n},j}, \mathbf{q}_{\mathbf{n},j}^{\text{out}}) + \mathbf{q}_{\mathbf{n},j}^T\widehat{\mathbf{v}}(\mathbf{v}, \mathbf{v}^{\text{out}}) - \mathbf{v}_j^T\mathbf{q}_{\mathbf{n},j}\Big) := \sum_{j=1}^{\mathcal{N}_{Q,k}}\tau_j A_j.$$

If $\boldsymbol{\xi}_j \in \partial T$, we add the corresponding terms from the other side of the interface. The contribution at $\boldsymbol{\xi}_j$ is

$$\begin{aligned} A_j + A_j^{\text{out}} &= (\mathbf{v}_j - \mathbf{v}_j^{\text{out}})^T \widehat{\mathbf{q}}_{\mathbf{n}}(\mathbf{v}_j, \mathbf{v}_j^{\text{out}}, \mathbf{q}_{\mathbf{n},j}, \mathbf{q}_{\mathbf{n},j}^{\text{out}}) \\ &\qquad + (\mathbf{q}_{\mathbf{n},j} - \mathbf{q}_{\mathbf{n},j}^{\text{out}})^T \widehat{\mathbf{v}}(\mathbf{v}_j, \mathbf{v}_j^{\text{out}}) - (\mathbf{v}_j^T \mathbf{q}_{\mathbf{n},j} - (\mathbf{v}_j^{\text{out}})^T \mathbf{q}_{\mathbf{n},j}^{\text{out}}). \end{aligned}$$

Due to the identity

$$\mathbf{v}_j^T \mathbf{q}_{\mathbf{n},j} - (\mathbf{v}_j^{\text{out}})^T \mathbf{q}_{\mathbf{n},j}^{\text{out}} = \frac{1}{2}(\mathbf{v}_j + \mathbf{v}_j^{\text{out}})^T (\mathbf{q}_{\mathbf{n},j} - \mathbf{q}_{\mathbf{n},j}^{\text{out}}) + \frac{1}{2}(\mathbf{v}_j - \mathbf{v}_j^{\text{out}})^T (\mathbf{q}_{\mathbf{n},j} + \mathbf{q}_{\mathbf{n},j}^{\text{out}}),$$

plugging (2.45) yields

$$A_j + A_j^{\text{out}} = -\alpha(\mathbf{v}_j - \mathbf{v}_j^{\text{out}})^T (\mathbf{v}_j - \mathbf{v}_j^{\text{out}}) \leq 0.$$

Hence the boundary contribution is also non-positive and our nodal LDG method is entropy stable. □

Remark 2.7. Both α and β may depend on j. We can also replace α by a symmetric positive-definite $p \times p$ matrix.

2.6 Numerical experiments

We will perform numerical tests for two-dimensional systems of conservation laws on unstructured triangular meshes generated by Gmsh[2] [37]. The discrete SBP operators are built on the B-type quadrature points in Section 2.2, with $k = 1, 2, 3, 4$. Basic settings are the same as in one-dimensional numerical experiments. We take $k = 2, 3, 4$ for smooth test problems, and only $k = 2$ for discontinuous test problems. For Euler equations, we still use the Chandrashekar's entropy conservative flux (2.12),

[2] http://gmsh.info/

and the local Lax-Friedrichs flux will be considered at boundary for problems with strong shocks, due to the reason that the exact Riemann solver might contain vacuum state.

Example 2.6.1. The first smooth test example is the two-dimensional linear advection equation

$$\frac{\partial u}{\partial t} + \frac{\partial u}{\partial x_1} + \frac{\partial u}{\partial x_2} = 0, \quad \mathbf{x} \in [0,1]^2,$$

with periodic boundary condition and initial data $u(0,\mathbf{x}) = \sin(2\pi x_1)\sin(2\pi x_2)$, and square entropy function $U = u^2/2$. The exact solution is $u(t,\mathbf{x}) = u(0, x_1 - t, x_2 - t)$. We test the two-dimensional entropy stable DGSEM on a hierarchy of unstructured triangular meshes. Errors and orders of convergence at $t = 0.2$ are shown in Table 2.1. Same as the one-dimensional linear advection equation, we obtain optimal convergence.

Table 2.1: Example 2.6.1: accuracy test of the two-dimensional linear advection equation associated with initial data $u(0,\mathbf{x}) = \sin(2\pi x_1)\sin(2\pi x_2)$ and square entropy function at $t = 0.2$.

k	h	L^1 error	order	L^2 error	order	L^∞ error	order
2	1/8	3.380e-3	-	6.000e-3	-	6.890e-2	-
	1/16	5.032e-4	2.748	9.868e-4	2.604	1.809e-2	1.930
	1/32	6.170e-5	3.028	1.213e-4	3.024	2.292e-3	2.981
	1/64	7.916e-6	2.962	1.551e-5	2.967	3.387e-4	2.758
	1/128	9.890e-7	3.001	1.926e-6	3.010	4.419e-5	2.938
	1/256	1.244e-7	2.991	2.414e-7	2.996	5.929e-6	2.898
3	1/8	2.329e-4	-	4.375e-4	-	8.752e-3	-
	1/16	2.114e-5	3.461	3.536e-5	3.629	8.228e-4	3.411
	1/32	1.790e-6	3.562	2.810e-6	3.654	6.194e-5	3.731
	1/64	1.429e-7	3.647	2.210e-7	3.668	4.310e-6	3.845
	1/128	1.063e-8	3.748	1.658e-8	3.737	3.183e-7	3.759
	1/256	7.341e-10	3.856	1.160e-9	3.838	2.194e-8	3.859
4	1/8	1.295e-5	-	2.230e-5	-	6.184e-4	-
	1/16	4.534e-7	4.837	9.969e-7	4.483	6.627e-5	3.222
	1/32	1.528e-8	4.891	2.824e-8	5.141	1.401e-6	5.564
	1/64	4.923e-10	4.956	8.940e-10	4.982	6.046e-8	4.535
	1/128	1.547e-11	4.992	2.773e-11	5.011	1.897e-9	4.994

Example 2.6.2. We consider the two-dimensional Burgers equation

$$\frac{\partial u}{\partial t} + \frac{\partial u^2}{\partial x_1} + \frac{\partial u^2}{\partial x_2} = 0, \quad \mathbf{x} \in [0,1]^2,$$

with periodic boundary condition and initial data $u(0,\mathbf{x}) = 0.5\sin(2\pi(x_1 + x_2))$, and square entropy function $U = u^2/2$. Exact solution follows from the solution of one-dimensional Burgers equation in the direction $\eta = x_1 + x_2$. The entropy stable DGSEM is evolved up to $t = 0.05$ when the solution is still smooth. Errors and orders of convergence are displayed in Table 2.2. The results are similar to its one-dimensional counterpart. Convergence rate is below optimal.

Table 2.2: Example 2.6.2: accuracy test of the two-dimensional Burgers equation associated with initial data $u(0,\mathbf{x}) = 0.5\sin(2\pi(x_1{+}x_2))$ and square entropy function at $t = 0.05$.

k	h	L^1 error	order	L^2 error	order	L^∞ error	order
2	1/16	1.354e-3	-	3.275e-3	-	5.954e-2	-
	1/32	2.394e-4	2.500	7.046e-4	2.217	1.646e-2	1.855
	1/64	3.900e-5	2.618	1.406e-4	2.325	4.894e-3	1.750
	1/128	5.773e-6	2.756	2.456e-5	2.518	1.269e-3	1.948
	1/256	8.431e-7	2.776	4.109e-6	2.579	2.413e-4	2.394
3	1/16	1.890e-4	-	6.252e-4	-	1.968e-2	-
	1/32	2.482e-5	2.929	1.058e-4	2.563	4.859e-3	2.018
	1/64	2.327e-6	3.415	1.106e-5	3.258	7.311e-4	2.733
	1/128	2.065e-7	3.494	1.158e-6	3.255	1.195e-4	2.613
	1/256	1.898e-8	3.444	1.236e-7	3.229	1.299e-5	3.202
4	1/16	3.740e-5	-	1.454e-4	-	6.039e-3	-
	1/32	2.787e-6	3.746	1.427e-5	3.349	1.068e-3	2.500
	1/64	1.348e-7	4.370	7.651e-7	4.221	8.839e-5	3.595
	1/128	5.566e-9	4.598	3.722e-8	4.362	6.398e-6	3.788
	1/256	2.293e-10	4.602	1.696e-9	4.456	3.059e-7	4.387

Example 2.6.3 (Isentropic vortex)**.** The last smooth test case is the isentropic vortex advection problem for the two-dimensional Euler equations, taken from Shu [89]. The computational domains is $[0,10]^2$ and the initial condition is given by

$$w_1(0,\mathbf{x}) = 1 - (x_2 - y_2)\phi(r), \quad w_2(0,\mathbf{x}) = 1 + (x_1 - y_1)\phi(r),$$

$$T(0,\mathbf{x}) = 1 - \frac{\gamma-1}{2\gamma}\phi(r)^2, \quad \rho(0,\mathbf{x}) = T^{\frac{1}{\gamma-1}}, \quad p(0,\mathbf{x}) = T^{\frac{\gamma}{\gamma-1}},$$

where (y_1, y_2) is the initial center of the vortex and

$$\phi(r) = \varepsilon e^{\alpha(1-r^2)}, \quad r = \sqrt{(x_1-y_1)^2 + (x_2-y_2)^2}.$$

The parameters are $\varepsilon = \frac{5}{2\pi}$, $\alpha = 0.5$ and $(y_1, y_2) = (5, 5)$. The vortex will be advected in the diagonal direction and the exact solution is $\mathbf{u}(t,\mathbf{x}) = \mathbf{u}(0, x_1 - t, x_2 - t)$. We use the exact solution to prescribe boundary conditions. Table 2.3 summarizes errors and orders of convergence of the density at $t = 1$. Here the convergence rate is also slightly below optimal, but better than Burgers equation.

Table 2.3: Example 2.6.3: accuracy test of isentropic vortex problem for two-dimensional Euler equations at $t = 1$. Results of the density are tabulated.

k	h	L^1 error	order	L^2 error	order	L^∞ error	order
2	10/8	2.299e-1	-	6.053e-2	-	8.735e-2	-
	10/16	4.204e-2	2.451	1.223e-2	2.307	2.957e-2	1.563
	10/32	6.598e-3	2.671	1.918e-3	2.673	5.162e-3	2.518
	10/64	9.330e-4	2.822	2.688e-4	2.835	1.064e-3	2.279
	10/128	1.273e-4	2.873	3.609e-5	2.897	1.717e-4	2.631
	10/256	1.652e-5	2.947	4.779e-6	2.917	2.280e-5	2.913
3	10/8	4.344e-2		1.160e-2		2.960e-2	
	10/16	3.976e-3	3.450	1.155e-3	3.327	4.271e-3	2.793
	10/32	3.632e-4	3.453	1.030e-4	3.487	2.652e-4	4.009
	10/64	3.041e-5	3.578	8.538e-6	3.593	4.557e-5	2.541
	10/128	2.536e-6	3.584	7.148e-7	3.578	3.793e-6	3.587
	10/256	1.990e-7	3.672	5.670e-8	3.656	2.720e-7	3.802
4	10/8	7.754e-3	-	2.136e-3	-	8.836e-3	-
	10/16	3.941e-4	4.298	1.308e-4	4.030	5.582e-4	3.985
	10/32	1.546e-5	4.672	4.858e-6	4.750	2.549e-5	4.452
	10/64	5.620e-7	4.782	1.806e-7	4.749	1.680e-6	3.923
	10/128	2.020e-8	4.798	6.433e-9	4.812	8.998e-8	4.223

Example 2.6.4 (two-dimensional Riemann problem)**.** We solve the Riemann prob-

lem of the two-dimensional Burgers equation

$$\frac{\partial u}{\partial t} + \frac{\partial u^2}{\partial x_1} + \frac{\partial u^2}{\partial x_2} = 0, \quad \mathbf{x} \in [0,1]^2,$$

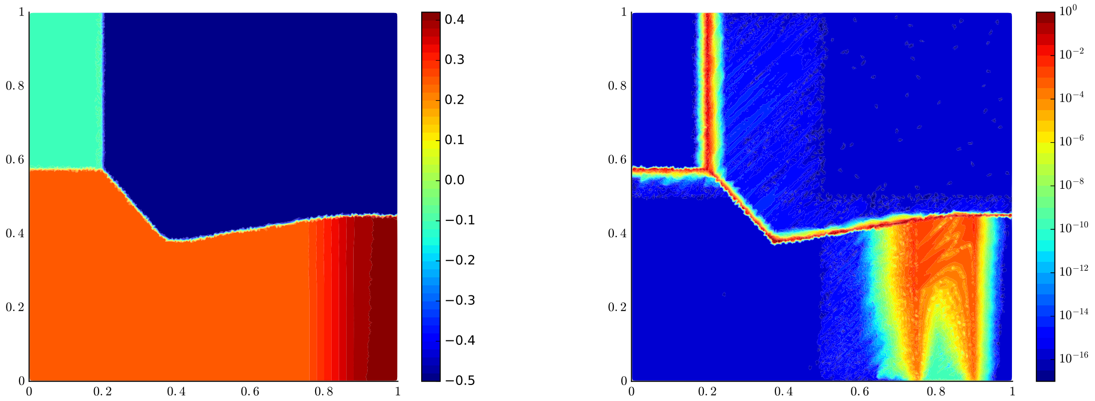
subject to the initial condition

$$u(\mathbf{x},0) = \begin{cases} 0.25 & \text{if } x_1 < 0.5 \text{ and } x_2 < 0.5 \\ -0.1 & \text{if } x_1 < 0.5 \text{ and } x_2 \geq 0.5 \\ 0.4 & \text{if } x_1 \geq 0.5 \text{ and } x_2 < 0.5 \\ -0.5 & \text{if } x_1 \geq 0.5 \text{ and } x_2 \geq 0.5 \end{cases}.$$

The exact solution for $t > 0$ is as follows [101, 43]

$$u(\mathbf{x},t) = \begin{cases} 0.25 & \text{if } x_1 < \frac{1}{2} - \frac{3t}{5} \text{ and } x_2 < \frac{1}{2} + \frac{t}{30} \\ -0.1 & \text{if } x_1 < \frac{1}{2} - \frac{3t}{5} \text{ and } x_2 \geq \frac{1}{2} + \frac{t}{30} \\ 0.25 & \text{if } \frac{1}{2} - \frac{3t}{5} \leq x_1 < \frac{1}{2} - \frac{t}{4} \text{ and } x_2 < \frac{-8x_1}{7} + \frac{15}{14} - \frac{15t}{28} \\ -0.5 & \text{if } \frac{1}{2} - \frac{3t}{5} \leq x_1 < \frac{1}{2} - \frac{t}{4} \text{ and } x_2 \geq \frac{-8x_1}{7} + \frac{15}{14} - \frac{15t}{28} \\ 0.25 & \text{if } \frac{1}{2} - \frac{t}{4} \leq x_1 < \frac{1}{2} + \frac{t}{2} \text{ and } x_2 < \frac{x_1}{6} + \frac{5}{12} - \frac{5t}{24} \\ -0.5 & \text{if } \frac{1}{2} - \frac{t}{4} \leq x_1 < \frac{1}{2} + \frac{t}{2} \text{ and } x_2 \geq \frac{x_1}{6} + \frac{5}{12} - \frac{5t}{24} \\ \frac{2x_1-1}{4t} & \text{if } \frac{1}{2} + \frac{t}{2} \leq x_1 < \frac{1}{2} + \frac{4t}{5} \text{ and } x_2 < x_1 - \frac{5}{18t}(x_1 + t - \frac{1}{2})^2 \\ -0.5 & \text{if } \frac{1}{2} + \frac{t}{2} \leq x_1 < \frac{1}{2} + \frac{4t}{5} \text{ and } x_2 \geq x_1 - \frac{5}{18t}(x_1 + t - \frac{1}{2})^2 \\ 0.4 & \text{if } x_1 \geq \frac{1}{2} + \frac{4t}{5} \text{ and } x_2 < \frac{1}{2} - \frac{t}{10} \\ -0.5 & \text{if } x_1 \geq \frac{1}{2} + \frac{4t}{5} \text{ and } x_2 \geq \frac{1}{2} - \frac{t}{10} \end{cases}.$$

We choose the square entropy function $U = u^2/2$ and run the entropy stable DGSEM up to $t = 0.5$ on a triangular mesh with $h = 1/128$. The bound-preserving limiter with $\Omega = [-0.5, 0.4]$ is also imposed. The numerical result is shown in the left panel of Figure 2.2, and the absolute value error is also plotted in the right panel where we use logarithmic scale and values less than 10^{-16} are transformed to 10^{-16}. The scheme successfully captures the correct profile. Error is very small unless near shock waves.



(a) numerical solution

(b) absolute value of error

Figure 2.2: Example 2.6.4: Numerical solution and error of a Riemann problem of two-dimensional Burgers equation at $t = 0.5$ on a mesh with $h = 1/128$. Entropy function is $U = u^2/2$ and bound-preserving limiter is used. Error is shown in logarithmic scale.

Example 2.6.5 (double Mach reflection)**.** This famous test problem of two-dimensional Euler equations was proposed by Woodward and Colella in [104] and has been intensively studied in the last few decades. It involves a Mach 10 shock which makes a 60° angle with a reflecting wall. The undisturbed air ahead of the shock has a density of 1.4 and a pressure of 1. Usually people solve the problem with rectangular computational domain and horizontal wall. Here we use the flexibility of unstructured triangular mesh to consider the original physical problem with a horizontally moving shock and a wall inclined with a 30° angle (e.g. [97]).We illustrate the computational domain and the unstructured mesh with $h = 1/20$ in Figure 2.3. Initially the shock is positioned at $x_1 = 0$. Inflow/outflow boundary conditions are prescribed for the left and right boundaries, and at the top boundary the flow values are set to describe the exact motion of shock.

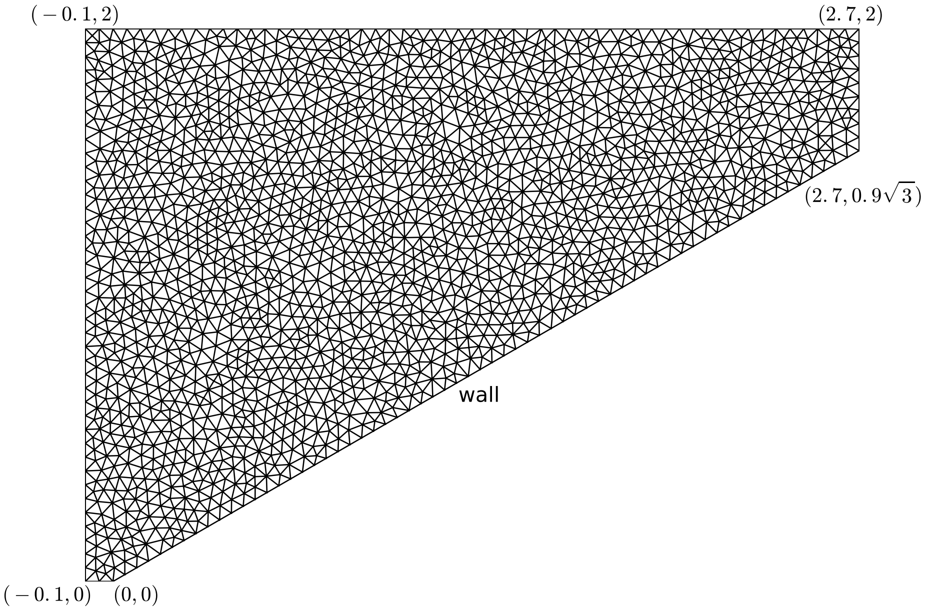
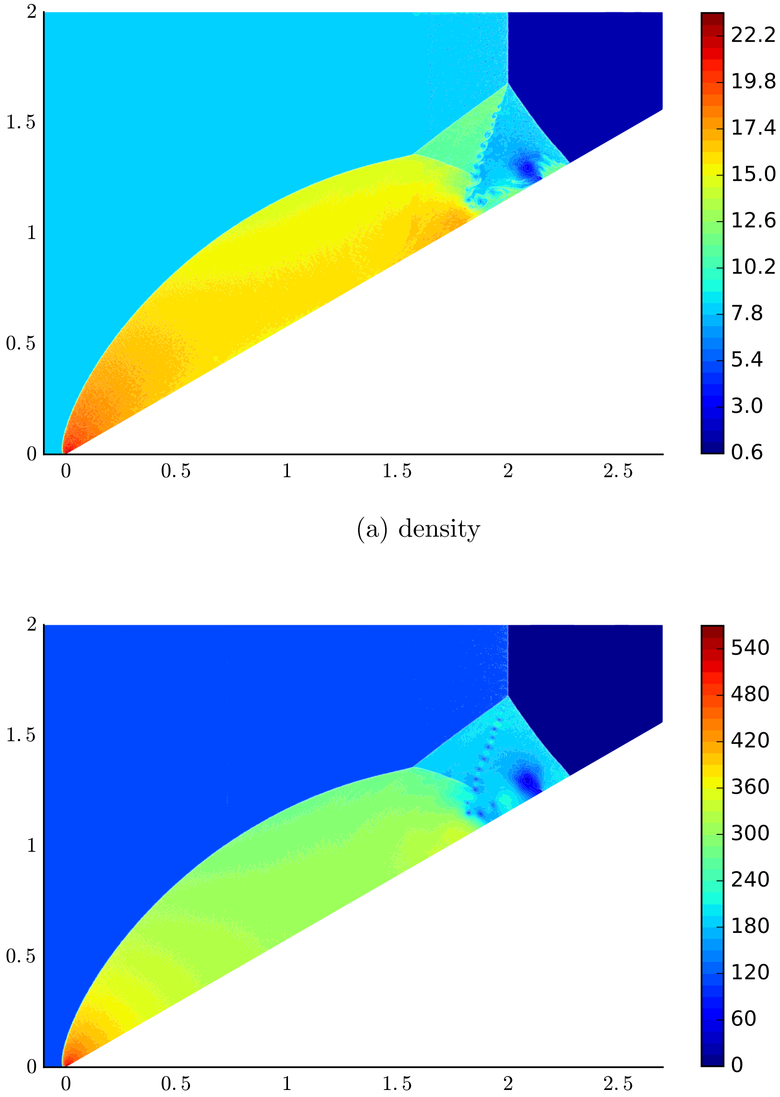


Figure 2.3: Example 2.6.5: illustration of the computational domain and the unstructured mesh with $h = 1/20$.

The entropy stable DGSEM is implemented with bound-preserving limiter (called positivity-preserving limiter as the density and pressure are kept positive) and local Lax-Friedrichs flux. The plots of density and pressure at $t = 0.2$ with mesh size $h = 1/240$ are given in Figure 2.4. Similar to the observations in [111], the solution is more oscillatory than results obtained via WENO scheme or DG scheme with TVD/TVB limiter, but it also catches some interesting features such as the small roll-ups due to Kelvin-Helmholtz instability, which indicates low numerical dissipation of our scheme.

Example 2.6.6 (shock diffraction)**.** A shock wave diffracting at a sharp corner is another popular test problem for two-dimensional Euler equations. In [19, 113] the results of a Mach 5.09 shock diffracting at a 90° edge are presented. Here we would like to study a Mach 10 shock diffracting at a 120° degree [114]. The computational



(a) density

(b) pressure

Figure 2.4: Example 2.6.5: profiles of density and pressure at $t = 0.2$ on a mesh with $h = 1/240$. 40 equally spaced contour levels are used for both plots.

domain and the triangular mesh with $h = 1/4$ are demonstrated in Figure 2.5. The shock is initially located at $x_1 = 3.4$ and $6 \leq x_2 \leq 11$, moving into undisturbed air with a density of 1.4 and a pressure of 1. Boundary conditions are inflow at the left/top boundary (in accordance with the exact shock motion), and outflow at the right/bottom boundary.

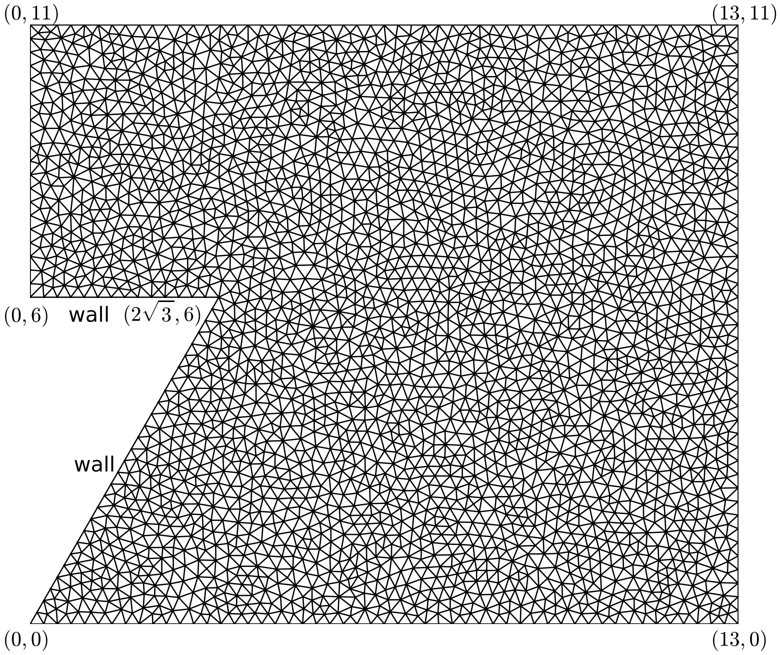
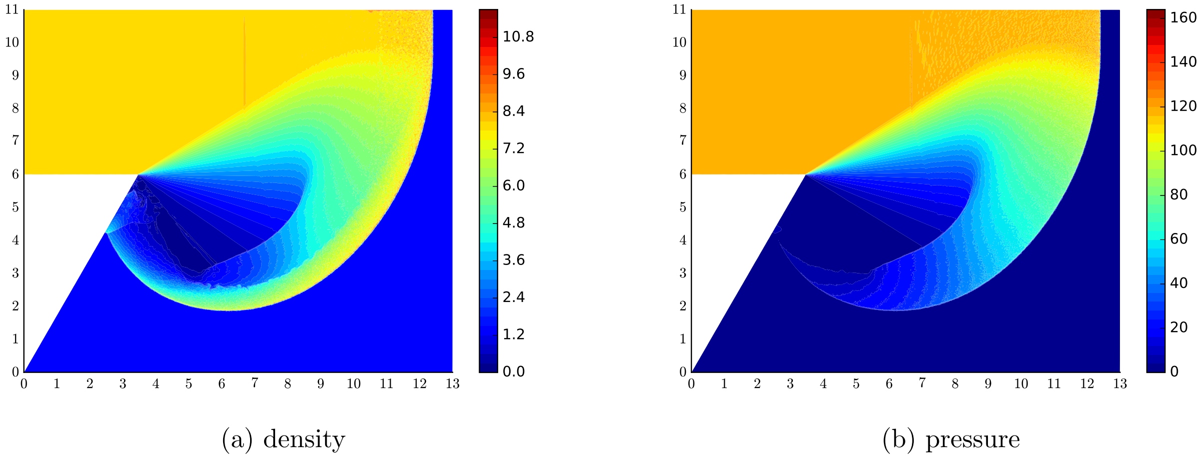


Figure 2.5: Example 2.6.6: illustration of the computational domain and the unstructured mesh with $h = 1/4$.

We still use positivity-preserving limiter and local Lax-Friedrichs interface flux. The contour plots of density and pressure at $t = 0.9$ with mesh size $h = 1/40$ are depicted in Figure 2.6. The result is comparable to the one in [114] despite some oscillations and overshoots near the shock wave.



(a) density

(b) pressure

Figure 2.6: Example 2.6.6: profiles of density and pressure at $t = 0.9$ on a mesh with $h = 1/20$. 40 equally spaced contour levels are used for both plots.

Chapter Three

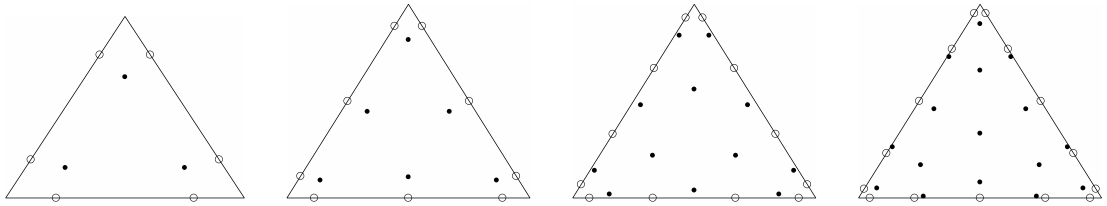
General Set of Nodes

The entropy stable DGSEM established in previous chapters depends on Gauss-Lobatto type quadrature points, i.e., the internal quadrature and the boundary quadrature share the same set of nodes. However, we also notice some drawbacks related to these Gauss-Lobatto type nodes:

1. The internal quadrature rule is only of degree $2k-1$. We detect suboptimal convergence rate in nonlinear test problems.
2. In higher space dimensions, the local degree of freedom ($\mathcal{N}_{Q,k}$) is much larger than the dimension of polynomials ($\mathcal{N}_{P,k}$), which makes the scheme more expensive than the classic DG method, in terms of both space complexity and time complexity.

Therefore, it is worthwhile to consider more general set of nodes, i.e., the internal quadrature and the boundary quadrature have separate sets of quadrature points. In one space dimension, the Legendre-Gauss quadrature rule is of degree $2k+1$. For two-dimensional triangular element, quadrature rules of degree $2k$ with $k = 1, 2, 3, 4$ are illustrated in Figure 3.1. We call them *A-type* quadrature rules, where the letter A means arbitrary distribution of quadrature points. The coordinates of quadrature points and the quadrature weights are also given in Appendix C. Compared to the B-type quadrature rules in Figure 2.1, the A-type quadrature rules achieve better algebraic accuracy with fewer number of nodes. This is the benefit of removing Gauss-Lobatto type constraint.

We would like to extend the entropy stable DGSEM to general set of nodes. Flux differencing can still be used to ensure internal entropy balance, while the main difficulty lies in boundary treatment. Under the generalized summation-by-parts paradigm [27, 28] for general set of nodes, the boundary matrices are dense, and identities such as (1.81) are no longer valid. Then we have to alter the boundary



(a) $k = 1, \mathcal{N}_{Q,k} = 3$ (b) $k = 2, \mathcal{N}_{Q,k} = 6$ (c) $k = 3, \mathcal{N}_{Q,k} = 12$ (d) $k = 4, \mathcal{N}_{Q,k} = 16$

Figure 3.1: A-type quadrature rules on triangles with $k = 1, 2, 3, 4$. Dots are quadrature points for the triangle, and circles are quadrature points for the edges.

penalty term (called simultaneous approximation term in the SBP community). Several approaches have been proposed to overcome this issue, and we will dig into them in this chapter.

This chapter consists of the following sections. In Section 3.1, we develop discrete operators with the generalized SBP property. The main new ingredient the extrapolation matrix that maps data to boundary nodes. In Section 3.2, we apply quadrature rules to the classic DG method (2.14) and deduce DGSEM formulations. Because of the mismatch of $\mathcal{N}_{P,k}$ and $\mathcal{N}_{Q,k}$, there are two related but non-equivalent formulations, i.e., the modal formulation (evolving polynomial coefficients) and the nodal formulation (evolving point values). In Section 3.3, we reinterpret the entropy stable DGSEM by Chan in [9, 10]. The key trick is a skew-symmetric boundary correction term. In Section 3.4, we review an alternative entropy stable boundary treatment by Crean et al in [21]. The authors replaced the bivariate interface flux with an extrapolation of entropy conservative fluxes. The resulting scheme is entropy conservative, and additional entropy dissipation can be added to element interfaces. In Section 3.5, we cover the “brutal force” type approach by Abgrall in [1]. We analyze the original DGSEM, which is not entropy stable due to aliasing error. Then a simple linear correction procedure will help eliminate the aliasing error. In Section 3.6, we will see that for convection-diffusion equations, the LDG discretization in Section 2.5 also works perfectly on general set of nodes. In Section 3.7, we try to derive modal versions of entropy stable DGSEM formulations. Finally in Section 3.8, we will examine the accuracy (i.e., whether optimal convergence is recovered) of these schemes on two-dimensional Burgers equation.

3.1 Generalized summation-by-parts operators

We still consider the reference simplex $K \subset \mathbb{R}^d$. Suppose that we have a degree $2k-1$ internal quadrature rule with $\mathcal{N}_{Q,k} \geq \mathcal{N}_{P,k}$ nodes $\{\boldsymbol{\xi}_j\}_{j=1}^{\mathcal{N}_{Q,k}}$ and positive weights $\{\omega_j\}_{j=1}^{\mathcal{N}_{Q,k}}$, and a degree $2k$ boundary quadrature rule with $\mathcal{N}_{B,k}$ nodes $\{\boldsymbol{\xi}_s^b\}_{s=1}^{\mathcal{N}_{B,k}}$ and positive weights $\{\tau_s\}_{s=1}^{\mathcal{N}_{B,k}}$. Let $\{\mathbf{n}_s\}_{s=1}^{\mathcal{N}_{B,k}}$ be the collection of outer normal vectors at boundary nodes. We define

$$M := \operatorname{diag}\{\omega_1, \cdots, \omega_{\mathcal{N}_{Q,k}}\}, \quad B := \operatorname{diag}\{\tau_1, \cdots, \tau_{\mathcal{N}_{B,k}}\}, \tag{3.1}$$

and

$$N_m := \operatorname{diag}\{n_{1,m}, \cdots, n_{\mathcal{N}_{B,k},m}\}, \quad 1 \leq m \leq d. \tag{3.2}$$

We also set the Vandermonde matrix and the boundary Vandermonde matrix:

$$V := \{p_l(\boldsymbol{\xi}_j)\}_{1\leq j\leq \mathcal{N}_{Q,k}, 1\leq l\leq \mathcal{N}_{P,k}}, \quad V^b := \{p_l(\boldsymbol{\xi}_s^b)\}_{1\leq s\leq \mathcal{N}_{B,k}, 1\leq l\leq \mathcal{N}_{P,k}}, \tag{3.3}$$

as well as the projection matrix

$$P := (V^T M V)^{-1} V^T M. \tag{3.4}$$

Since the internal nodes and the boundary nodes do not overlap, we need some communication mechanism between them. We construct an $\mathcal{N}_{B,k} \times \mathcal{N}_{Q,k}$ extrapolation matrix R that is exact for all polynomials in $\mathcal{P}^k(T)$. In other words,

$$RV = V^b. \tag{3.5}$$

A simple choice of the extrapolation matrix is $R = V^b P$. For a function u, the vectors of internal nodal values and boundary nodal values are defined as:

$$\overrightarrow{u} := \begin{bmatrix} u(\boldsymbol{\xi}_1) & \cdots & u(\boldsymbol{\xi}_{\mathcal{N}_{Q,k}}) \end{bmatrix}^T, \quad \overrightarrow{u^s} := R\overrightarrow{u}.$$

Difference matrices with generalized SBP property will be provided in the following theorem.

Theorem 3.1. *Assume that we have an extrapolation matrix R with the exactness property* (3.5)*. Then the difference matrices, given by the formula*

$$D_m := \frac{1}{2} M^{-1} (R + V^b P)^T B N_m (R - V^b P) + V \widehat{D}_m P, \tag{3.6}$$

satisfy the two properties below:

1. *Exactness: $D_m V = V \widehat{D}_m$.*
2. *Generalized summation-by-parts: $E_m = S_m + S_m^T$, where $S_m := MD_m$ and $E_m := R^T B N_m R$.*

Remark 3.1. By exactness of R and D_m, and the generalized SBP property,

$$R\overrightarrow{1} = \overrightarrow{1^b}, \quad S_m\overrightarrow{1} = D_m\overrightarrow{1} = \overrightarrow{0}, \quad S_m^T\overrightarrow{1} = E_m\overrightarrow{1} = R^T B N_m \overrightarrow{1^b}, \tag{3.7}$$

where $\overrightarrow{1^b}$ is the length-$\mathcal{N}_{B,k}$ vector of 1s.

The proof is the same as Theorem 2.3. Here E_m denotes the generalized boundary matrix, such that for functions u and v,

$$\overrightarrow{u}^T E_m \overrightarrow{v} = \left(\overrightarrow{u^b}\right)^T B N_m \overrightarrow{v^b} = \sum_{s=1}^{\mathcal{N}_{B,k}} \tau_s n_{s,m} u_s^b v_s^b$$

still approximates the boundary integral $\int_{\partial K} uvn_m dS$. Notice that E_m is dense, which makes boundary treatment more involved in entropy stable DG type methods.

Similar to Section 2.2, we can use Kroncker products to generate discrete operators for vector-valued functions, and Jacobian factors to compute discrete operators on local elements.

Remark 3.2. We would like to highlight some special cases of extrapolation matrix and difference matrices.

1. If $\mathcal{N}_{P,k} = \mathcal{N}_{Q,k}$ (e.g. one-dimensional Legendre-Gauss quadrature and Gauss-Lobatto quadrature), $R = V_b V^{-1}$ and $D_m = V\widehat{D}_m V^{-1}$.
2. If $R = V_b P$, the first term of (3.6) vanishes, and $D_m = V\widehat{D}_m P$.
3. In Gauss-Lobatto type quadrature, $\boldsymbol{\xi}_s^b = \boldsymbol{\xi}_s$ for each $1 \leq s \leq \mathcal{N}_{B,k}$, and we can choose $R = \begin{bmatrix} I_{\mathcal{N}_{B,k}} & 0 \end{bmatrix}$. Then we recover the difference matrices in (2.24).

3.2 DGSEM: from modal formulation to nodal formulation

Recall the classic DG method for systems of conservation laws (on the reference element K):

$$\int_K \frac{\partial \mathbf{u}_h^T}{\partial t} \mathbf{w}_h d\boldsymbol{\xi} - \sum_{m=1}^{d} \int_K \mathbf{f}_m(\mathbf{u}_h)^T \frac{d\mathbf{w}_h}{d\xi_m} d\boldsymbol{\xi} = -\int_{\partial K} \widehat{\mathbf{f}_\mathbf{n}}(\mathbf{u}_h, \mathbf{u}_h^{\text{out}})^T \mathbf{w}_h dS. \tag{3.8}$$

We expand $\mathbf{u}_h$ and $\mathbf{w}_h$ under the basis $\{p_l(\boldsymbol{\xi})\}_{l=1}^{\mathcal{N}_{P,k}}$:

$$\mathbf{u}_h = \sum_{l=1}^{\mathcal{N}_{P,k}} \widehat{\mathbf{u}}_l p_l(\boldsymbol{\xi}), \quad \mathbf{w}_h = \sum_{l=1}^{\mathcal{N}_{P,k}} \widehat{\mathbf{w}}_l p_l(\boldsymbol{\xi}).$$

Define

$$\overrightarrow{\widehat{\mathbf{u}}} := \begin{bmatrix} \widehat{\mathbf{u}}_1 \\ \vdots \\ \widehat{\mathbf{u}}_{\mathcal{N}_{P,k}} \end{bmatrix}, \quad \overrightarrow{\widehat{\mathbf{w}}} := \begin{bmatrix} \widehat{\mathbf{w}}_1 \\ \vdots \\ \widehat{\mathbf{w}}_{\mathcal{N}_{P,k}} \end{bmatrix}.$$

Then $\overrightarrow{\mathbf{u}} = \mathbf{V}\overrightarrow{\widehat{\mathbf{u}}}$, $\overrightarrow{\mathbf{u}^b} = \mathbf{V}^b\overrightarrow{\widehat{\mathbf{u}}}$. We use the internal quadrature rule to approximate the left hand side of (3.8), and the boundary quadrature rule to approximate the right hand side. The resulting scheme is

$$\left(\mathbf{V}\overrightarrow{\widehat{\mathbf{w}}}\right)^T \mathbf{M}\left(\mathbf{V}\frac{d\overrightarrow{\widehat{\mathbf{u}}}}{dt}\right) - \sum_{m=1}^{d}\left(\mathbf{V}\widehat{\mathbf{D}}_m\overrightarrow{\widehat{\mathbf{w}}}\right)^T \mathbf{M}\overrightarrow{\mathbf{f}_m} = -\left(\mathbf{V}^b\overrightarrow{\widehat{\mathbf{w}}}\right)^T \mathbf{B}\overrightarrow{\mathbf{f}^*_{\mathbf{n}}},$$

where $\mathbf{f}^*_{\mathbf{n},s} = \widehat{\mathbf{f}}_{\mathbf{n}}(\mathbf{u}^b_s, \mathbf{u}^{b,\text{out}}_s)$. Since $\overrightarrow{\widehat{\mathbf{w}}}$ can be arbitrary, we obtain

$$(\mathbf{V}^T\mathbf{M}\mathbf{V})\frac{d\overrightarrow{\widehat{\mathbf{u}}}}{dt} - \sum_{m=1}^{d}(\mathbf{V}\widehat{\mathbf{D}}_m)^T\mathbf{M}\overrightarrow{\mathbf{f}_m} = -(\mathbf{V}^b)^T\mathbf{B}\overrightarrow{\mathbf{f}^*_{\mathbf{n}}}, \tag{3.9}$$

i.e.,

$$\frac{d\overrightarrow{\widehat{\mathbf{u}}}}{dt} - \sum_{m=1}^{d}(\mathbf{V}^T\mathbf{M}\mathbf{V})^{-1}(\mathbf{V}\widehat{\mathbf{D}}_m)^T\mathbf{M}\overrightarrow{\mathbf{f}_m} = -(\mathbf{V}^T\mathbf{M}\mathbf{V})^{-1}(\mathbf{V}^b)^T\mathbf{B}\overrightarrow{\mathbf{f}^*_{\mathbf{n}}}. \tag{3.10}$$

This is called modal DGSEM formulation as we evolve the vector of polynomial expansion coefficients. Applying the Vandermonde matrix $\mathbf{V}$ to (3.10), we come up with the nodal formulation that describes the evolution of $\overrightarrow{\mathbf{u}}$:

$$\frac{d\overrightarrow{\mathbf{u}}}{dt} - \sum_{m=1}^{d}\mathbf{M}^{-1}(\mathbf{V}\widehat{\mathbf{D}}_m\mathbf{P})^T\mathbf{M}\overrightarrow{\mathbf{f}_m} = -\mathbf{M}^{-1}(\mathbf{V}^b\mathbf{P})^T\mathbf{B}\overrightarrow{\mathbf{f}^*_{\mathbf{n}}}. \tag{3.11}$$

Here we use the relation $\mathbf{V}(\mathbf{V}^T\mathbf{M}\mathbf{V})^{-1} = \mathbf{M}^{-1}\mathbf{P}^T$. It is a special case of the more general nodal DGSEM formulation

$$\frac{d\overrightarrow{\mathbf{u}}}{dt} - \sum_{m=1}^{d} \mathbf{M}^{-1}\mathbf{S}_m^T\overrightarrow{\mathbf{f}_m} = -\mathbf{M}^{-1}\mathbf{R}^T\mathbf{B}\overrightarrow{\mathbf{f}_{\mathbf{n}}^*}, \tag{3.12}$$

by choosing $\mathbf{R} = \mathbf{V}^b\mathbf{P}$ and $\mathbf{D}_m = \mathbf{V}\widehat{\mathbf{D}}_m\mathbf{P}$. According to generalized SBP property, we also deduce the strong version of (3.12):

$$\begin{aligned}\frac{d\overrightarrow{\mathbf{u}}}{dt} + \sum_{m=1}^{d} \mathbf{D}_m\overrightarrow{\mathbf{f}_m} &= \mathbf{M}^{-1}\Big(\sum_{m=1}^{D} \mathbf{E}_m\overrightarrow{\mathbf{f}_m} - \mathbf{R}^T\mathbf{B}\overrightarrow{\mathbf{f}_{\mathbf{n}}^*}\Big)\\ &= \mathbf{M}^{-1}\mathbf{R}^T\mathbf{B}\Big(\sum_{m=1}^{d} \mathbf{N}_m\overrightarrow{\mathbf{f}_m^b} - \overrightarrow{\mathbf{f}_{\mathbf{n}}^*}\Big) = \mathbf{M}^{-1}\mathbf{R}^T\mathbf{B}\Big(\overrightarrow{\mathbf{f}_{\mathbf{n}}^b} - \overrightarrow{\mathbf{f}_{\mathbf{n}}^*}\Big).\end{aligned} \tag{3.13}$$

On the other hand, we can recover the modal formulation by applying projection to (3.12), and setting $\frac{d\overrightarrow{\widehat{\mathbf{u}}}}{dt} = \mathbf{P}\frac{\overrightarrow{\mathbf{u}}}{dt}$:

$$\frac{d\overrightarrow{\widehat{\mathbf{u}}}}{dt} - \sum_{m=1}^{d} \mathbf{P}\mathbf{M}^{-1}\mathbf{S}_m^T\overrightarrow{\mathbf{f}_m} = -\mathbf{P}\mathbf{M}^{-1}\mathbf{R}^T\mathbf{B}\overrightarrow{\mathbf{f}_{\mathbf{n}}^*}.$$

This reduces to (3.10) due to the exactness properties:

$$\mathbf{P}\mathbf{M}^{-1}\mathbf{D}_m^T = (\mathbf{V}^T\mathbf{M}\mathbf{V})^{-1}(\mathbf{D}_m\mathbf{V})^T = (\mathbf{V}^T\mathbf{M}\mathbf{V})^{-1}(\mathbf{V}\widehat{\mathbf{D}}_m)^T,$$

and

$$\mathbf{P}\mathbf{M}^{-1}\mathbf{R}^T = (\mathbf{V}^T\mathbf{M}\mathbf{V})^{-1}(\mathbf{R}\mathbf{V})^T = (\mathbf{V}^T\mathbf{M}\mathbf{V})^{-1}(\mathbf{V}^b)^T.$$

To conclude, by taking interpolation, the modal formulation implies a specific nodal formulation (with particular choices of $\mathbf{R}$ and $\mathbf{D}_m$). However, by taking projection, all nodal formulations (with any $\mathbf{R}$ and $\mathbf{D}_m$ satisfying exactness properties and generalized SBP property) lead to the modal formulation. The reason for such asymmetric relation is the fact that $\mathcal{N}_{Q,k} \geq \mathcal{N}_{P,k}$.

3.3 Approach 1: skew-symmetric boundary correction

We start to analyze the entropy stable DGSEM in [9, 10] by Chan. We still modify (3.13) by replacing $\mathbf{D}_m\overrightarrow{\mathbf{f}_m}$ with the flux differencing term $2\mathbf{D}_m \circ \mathbf{F}_{m,S}(\overrightarrow{\mathbf{u}}, \overrightarrow{\mathbf{u}})\overrightarrow{\mathbf{1}}$. Using the proof of Lemma 1.1, we can show that

$$\overrightarrow{\mathbf{1}}^T\mathbf{M}\Big(2\mathbf{D}_m \circ \mathbf{F}_{m,S}(\overrightarrow{\mathbf{u}}, \overrightarrow{\mathbf{u}})\overrightarrow{\mathbf{1}}\Big) = \overrightarrow{\mathbf{1}}^T\mathbf{E}_m \circ \mathbf{F}_{m,S}(\overrightarrow{\mathbf{u}}, \overrightarrow{\mathbf{u}})\overrightarrow{\mathbf{1}},$$

and

$$\overrightarrow{\mathbf{v}}^T\mathbf{M}\Big(2\mathbf{D}_m \circ \mathbf{F}_{m,S}(\overrightarrow{\mathbf{u}}, \overrightarrow{\mathbf{u}})\overrightarrow{\mathbf{1}}\Big) = \overrightarrow{\mathbf{v}}^T\mathbf{E}_m \circ \mathbf{F}_{m,S}(\overrightarrow{\mathbf{u}}, \overrightarrow{\mathbf{u}})\overrightarrow{\mathbf{1}} - \overrightarrow{\psi}^T E_m \overrightarrow{1}.$$

Since $\mathbf{E}_m$ is dense, it is difficult characterize $\mathbf{E}_m \circ \mathbf{F}_{m,S}(\overrightarrow{\mathbf{u}}, \overrightarrow{\mathbf{u}})\overrightarrow{\mathbf{1}}$. A natural step is to replace $\mathbf{M}^{-1}\mathbf{E}_m\overrightarrow{\mathbf{f}_m}$ with $\mathbf{M}^{-1}\mathbf{E}_m \circ \mathbf{F}_{m,S}(\overrightarrow{\mathbf{u}}, \overrightarrow{\mathbf{u}})\overrightarrow{\mathbf{1}}$, which gives us the scheme

$$\frac{d\overrightarrow{\mathbf{u}}}{dt} + 2\sum_{m=1}^{d} \mathbf{D}_m \circ \mathbf{F}_{m,S}(\overrightarrow{\mathbf{u}}, \overrightarrow{\mathbf{u}})\overrightarrow{\mathbf{1}} = \mathbf{M}^{-1}\Big(\sum_{m=1}^{d} \mathbf{E}_m \circ \mathbf{F}_{m,S}(\overrightarrow{\mathbf{u}}, \overrightarrow{\mathbf{u}})\overrightarrow{\mathbf{1}} - \mathbf{R}^T\mathbf{B}\overrightarrow{\mathbf{f}^*_{\mathbf{n}}}\Big). \tag{3.14}$$

Then (3.14) will achieve local conservation and entropy balance:

$$\frac{d}{dt}(\overrightarrow{\mathbf{1}}^T\mathbf{M}\overrightarrow{\mathbf{u}}) = \left(\overrightarrow{\mathbf{1}^b}\right)^T\mathbf{B}\overrightarrow{\mathbf{f}^*_{\mathbf{n}}}, \quad \frac{d}{dt}(\overrightarrow{1}^T B\overrightarrow{U}) = \left(\overrightarrow{\psi^b_{\mathbf{n}}}\right)^T M\overrightarrow{1^b} - \left(\overrightarrow{\mathbf{v}^b}\right)^T\mathbf{B}\overrightarrow{\mathbf{f}^*_{\mathbf{n}}}. \tag{3.15}$$

We also have global conservation due to the cancellation of numerical fluxes from both sides of element interface. However, unlike flux differencing, the boundary modification in (3.14) violates accuracy. Besides, the entropy production rate at $\boldsymbol{\xi}^b_s$ is

$$(\mathbf{v}^{b,\text{out}}_s - \mathbf{v}^b_s)^T\widehat{\mathbf{f}_{\mathbf{n}}}(\mathbf{u}^b_s, \mathbf{u}^{b,\text{out}}_s) - (\psi^{b,\text{out}}_{\mathbf{n},s} - \psi^b_{\mathbf{n},s}).$$

The sign is undecided as $\mathbf{u}_s^b \neq \mathbf{u}(\mathbf{v}_s^b)$ and $\psi_s^b \neq \psi(\mathbf{v}_s^b)$. i.e., extrapolation does not commute with function evaluation. We solve the latter issue by resorting to entropy extrapolations. Set $\overrightarrow{\widetilde{\mathbf{u}}^b}$ and $\overrightarrow{\widetilde{\psi}^b}$ such that $\widetilde{\mathbf{u}}_s^b = \mathbf{u}(\mathbf{v}_s^b)$, $\widetilde{\psi}_{m,s}^b = \psi_m(\mathbf{v}_s^b)$. The interface flux also depends on entropy-extrapolated values, such that $\mathbf{f}^*_{\mathbf{n},s} = \widehat{\mathbf{f}}_{\mathbf{n}}(\widetilde{\mathbf{u}}_s^b, \widetilde{\mathbf{u}}_s^{b,\text{out}})$. In order to maintain high order accuracy, we add a skew-symmetric correction term, and build the following scheme:

$$
\begin{aligned}
\frac{d\overrightarrow{\mathbf{u}}}{dt} + 2\sum_{m=1}^{d} \mathbf{D}_m \circ \mathbf{F}_{m,S}(\overrightarrow{\mathbf{u}}, \overrightarrow{\mathbf{u}})\overrightarrow{\mathbf{1}} &= \mathbf{M}^{-1}\Big(\sum_{m=1}^{d} \mathbf{E}_m \circ \mathbf{F}_{m,S}(\overrightarrow{\mathbf{u}}, \overrightarrow{\mathbf{u}})\overrightarrow{\mathbf{1}} - \mathbf{R}^T\mathbf{B}\overrightarrow{\mathbf{f}^*_{\mathbf{n}}} \\
&+ \sum_{m=1}^{d}\Big(\mathbf{R}^T\mathbf{B}\mathbf{N}_m\Big(\mathbf{R}\circ\mathbf{F}_{m,S}\Big(\overrightarrow{\widetilde{\mathbf{u}}^b}, \overrightarrow{\mathbf{u}}\Big)\Big)\overrightarrow{\mathbf{1}} - \mathbf{R}^T \circ \mathbf{F}_{m,S}\Big(\overrightarrow{\widetilde{\mathbf{u}}^b}, \overrightarrow{\mathbf{u}}\Big)^T \mathbf{B}\mathbf{N}_{\mathbf{m}}\overrightarrow{\mathbf{1}^b}\Big)\Big),
\end{aligned}
\tag{3.16}
$$

where

$$
\mathbf{F}_{m,S}\Big(\overrightarrow{\widetilde{\mathbf{u}}^b}, \overrightarrow{\mathbf{u}}\Big) = \begin{bmatrix} \operatorname{diag}(\mathbf{f}_{m,S}(\widetilde{\mathbf{u}}_1^b, \mathbf{u}_1)) & \cdots & \operatorname{diag}(\mathbf{f}_{m,S}(\widetilde{\mathbf{u}}_1^b, \mathbf{u}_{\mathcal{N}_{Q,k}})) \\ \vdots & \ddots & \vdots \\ \operatorname{diag}(\mathbf{f}_{m,S}(\widetilde{\mathbf{u}}_{\mathcal{N}_{B,k}}^b, \mathbf{u}_1)) & \cdots & \operatorname{diag}(\mathbf{f}_{m,S}(\widetilde{\mathbf{u}}_{\mathcal{N}_{B,k}}^b, \mathbf{u}_{\mathcal{N}_{Q,k}})) \end{bmatrix}.
$$

The component-wise form of (3.16) is

$$
\begin{aligned}
\frac{d\mathbf{u}_j}{dt} + 2\sum_{m=1}^{d}\sum_{l=1}^{\mathcal{N}_{Q,k}} D_{m,jl}\mathbf{f}_{m,S}(\mathbf{u}_j, \mathbf{u}_l) &= \sum_{s=1}^{\mathcal{N}_{B,k}} R_{sj}\frac{\tau_s}{\omega_j}\Big(\sum_{l=1}^{\mathcal{N}_{Q,k}} R_{sl}\mathbf{f}_{\mathbf{n},S}(\mathbf{u}_j, \mathbf{u}_l) \\
&- \widehat{\mathbf{f}}_{\mathbf{n}}(\widetilde{\mathbf{u}}_s^b, \widetilde{\mathbf{u}}_s^{b,\text{out}}) + \sum_{l=1}^{\mathcal{N}_{Q,k}} R_{sl}\mathbf{f}_{\mathbf{n},S}(\widetilde{\mathbf{u}}_s^b, \mathbf{u}_l) - \mathbf{f}_{\mathbf{n},S}(\widetilde{\mathbf{u}}_s^b, \mathbf{u}_j)\Big).
\end{aligned}
\tag{3.17}
$$

Theorem 3.2. *Assume that the sequence of meshes $\{\mathcal{T}_h\}$ is uniform, and that all mappings and bivariate fluxes are smooth and Lipschitz continuous. If $\mathbf{f}_{m,S}$ is entropy conservative for each $1 \leq m \leq d$, and $\widehat{\mathbf{f}}_{\mathbf{n}}$ is entropy stable, then the scheme* (3.16) *is* *high order accurate in the sense that for all i, j and smooth solution $\mathbf{u}$ of (2.1),*

$$\begin{aligned}\frac{d\mathbf{u}_{i,j}}{dt} + 2\sum_{m=1}^{d}\sum_{l=1}^{\mathcal{N}_{Q,k}} D_{i,m,jl}\mathbf{f}_{m,S}(\mathbf{u}_{i,j},\mathbf{u}_{i,l}) - \sum_{s=1}^{\mathcal{N}_{B,k}} R_{sj}\frac{J^b_{i,s}\tau_s}{J_i\omega_j}\Big(\sum_{l=1}^{\mathcal{N}_{Q,k}} R_{sl}\mathbf{f}_{\mathbf{n},S}(\mathbf{u}_{i,j},\mathbf{u}_{i,l})\\ - \widehat{\mathbf{f}_{\mathbf{n}}}(\widetilde{\mathbf{u}}^b_{i,s},\widetilde{\mathbf{u}}^{b,out}_{i,s}) + \sum_{l=1}^{\mathcal{N}_{Q,k}} R_{sl}\mathbf{f}_{\mathbf{n},S}(\widetilde{\mathbf{u}}^b_{i,s},\mathbf{u}_{i,l}) - \mathbf{f}_{\mathbf{n},S}(\widetilde{\mathbf{u}}^b_{i,s},\mathbf{u}_{i,j})\Big) = \mathcal{O}(h^k),\end{aligned} \tag{3.18}$$

and conservative and entropy stable in the sense that

$$\frac{d}{dt}\Big(\sum_{i=1}^{N}\overrightarrow{\mathbf{1}}^T\mathbf{M}_i\overrightarrow{\mathbf{u}_i}\Big) = 0, \quad \frac{d}{dt}\Big(\sum_{i=1}^{N}\overrightarrow{1}^T M_i\overrightarrow{U_i}\Big) \leq 0. \tag{3.19}$$

Proof. Since the flux differencing is high order accurate, and the Jacobian factors have the scales $J_i = \Theta(h^d)$ and $J^b_{i,s} = \Theta(h^{d-1})$ (as a result of uniform mesh), it suffices to show that

$$\sum_{l=1}^{\mathcal{N}_{Q,k}} R_{sl}\mathbf{f}_{\mathbf{n},S}(\mathbf{u}_{i,j},\mathbf{u}_{i,l}) - \widehat{\mathbf{f}_{\mathbf{n}}}(\widetilde{\mathbf{u}}^b_{i,s},\widetilde{\mathbf{u}}^{b,\text{out}}_{i,s}) + \sum_{l=1}^{\mathcal{N}_{Q,k}} R_{sl}\mathbf{f}_{\mathbf{n},S}(\widetilde{\mathbf{u}}^b_{i,s},\mathbf{u}_{i,l}) - \mathbf{f}_{\mathbf{n},S}(\widetilde{\mathbf{u}}^b_{i,s},\mathbf{u}_{i,j}) = \mathcal{O}(h^{k+1}).$$

By the approximation property of extrapolation and Lipschitz continuity,

$$\mathbf{v}^b_{i,s} - \mathbf{v}(\mathbf{x}_i(\boldsymbol{\xi}^b_s)) = \mathcal{O}(h^{k+1}), \quad \widetilde{\mathbf{u}}^b_{i,s} - \mathbf{u}(\mathbf{x}_i(\boldsymbol{\xi}^b_s)) = \mathcal{O}(h^{k+1}).$$

We check each component separately:

$$\sum_{l=1}^{\mathcal{N}_{Q,k}} R_{sl}\mathbf{f}_{\mathbf{n},S}(\mathbf{u}_{i,j},\mathbf{u}_{i,l}) = \mathbf{f}_{\mathbf{n},S}(\mathbf{u}_{i,j},\mathbf{u}(\mathbf{x}_i(\boldsymbol{\xi}^b_s)) + \mathcal{O}(h^{k+1}),$$

$$\widehat{\mathbf{f}_{\mathbf{n}}}(\widetilde{\mathbf{u}}^b_{i,s},\widetilde{\mathbf{u}}^{b,\text{out}}_{i,s}) = \mathbf{f}_{\mathbf{n}}(\mathbf{u}(\mathbf{x}_i(\boldsymbol{\xi}^b_s)) + \mathcal{O}(h^{k+1}),$$

$$\sum_{l=1}^{\mathcal{N}_{Q,k}} R_{sl}\mathbf{f}_{\mathbf{n},S}(\widetilde{\mathbf{u}}^b_{i,s},\mathbf{u}_{i,l}) = \mathbf{f}_{\mathbf{n},S}(\widetilde{\mathbf{u}}^b_{i,s},\mathbf{u}(\mathbf{x}_i(\boldsymbol{\xi}^b_s)) + \mathcal{O}(h^{k+1}) = \mathbf{f}_{\mathbf{n}}(\mathbf{u}(\mathbf{x}_i(\boldsymbol{\xi}^b_s)) + \mathcal{O}(h^{k+1}),$$

$$\mathbf{f}_{\mathbf{n},S}(\widetilde{\mathbf{u}}^b_{i,s},\mathbf{u}_{i,j}) = \mathbf{f}_{\mathbf{n},S}(\mathbf{u}(\mathbf{x}_i(\boldsymbol{\xi}^b_s),\mathbf{u}_{i,j}) + \mathcal{O}(h^{k+1}).$$

Hence the truncation error of boundary terms is also of high order. For conservation and entropy stability, we identify the effect of the skew-symmetric correction term:

$$\begin{aligned}
&\overrightarrow{\mathbf{1}}^T\left(\mathbf{R}^T\mathbf{B}\mathbf{N}_m\left(\mathbf{R}\circ\mathbf{F}_{m,S}\left(\overrightarrow{\widetilde{\mathbf{u}}^b},\overrightarrow{\mathbf{u}}\right)\right)\overrightarrow{\mathbf{1}}-\mathbf{R}^T\circ\mathbf{F}_{m,S}\left(\overrightarrow{\widetilde{\mathbf{u}}^b},\overrightarrow{\mathbf{u}}\right)^T\mathbf{B}\mathbf{N}_{\mathbf{m}}\overrightarrow{\mathbf{1}^b}\right)\\
=&\left(\overrightarrow{\mathbf{1}^b}\right)^T\mathbf{B}\mathbf{N}_m\left(\mathbf{R}\circ\mathbf{F}_{m,S}\left(\overrightarrow{\widetilde{\mathbf{u}}^b},\overrightarrow{\mathbf{u}}\right)\right)\overrightarrow{\mathbf{1}}-\overrightarrow{\mathbf{1}}^T\mathbf{R}^T\circ\mathbf{F}_{m,S}\left(\overrightarrow{\widetilde{\mathbf{u}}^b},\overrightarrow{\mathbf{u}}\right)^T\mathbf{B}\mathbf{N}_{\mathbf{m}}\overrightarrow{\mathbf{1}^b}=0,
\end{aligned}$$

and

$$\begin{aligned}
&\overrightarrow{\mathbf{v}}^T\left(\mathbf{R}^T\mathbf{B}\mathbf{N}_m\left(\mathbf{R}\circ\mathbf{F}_{m,S}\left(\overrightarrow{\widetilde{\mathbf{u}}^b},\overrightarrow{\mathbf{u}}\right)\right)\overrightarrow{\mathbf{1}}-\mathbf{R}^T\circ\mathbf{F}_{m,S}\left(\overrightarrow{\widetilde{\mathbf{u}}^b},\overrightarrow{\mathbf{u}}\right)^T\mathbf{B}\mathbf{N}_{\mathbf{m}}\overrightarrow{\mathbf{1}^b}\right)\\
=&\left(\overrightarrow{\mathbf{v}^b}\right)^T\mathbf{B}\mathbf{N}_m\left(\mathbf{R}\circ\mathbf{F}_{m,S}\left(\overrightarrow{\widetilde{\mathbf{u}}^b},\overrightarrow{\mathbf{u}}\right)\right)\overrightarrow{\mathbf{1}}-\overrightarrow{\mathbf{v}}^T\mathbf{R}^T\circ\mathbf{F}_{m,S}\left(\overrightarrow{\widetilde{\mathbf{u}}^b},\overrightarrow{\mathbf{u}}\right)^T\mathbf{B}\mathbf{N}_{\mathbf{m}}\overrightarrow{\mathbf{1}^b}\\
=&\sum_{j=1}^{\mathcal{N}_{Q,k}}\sum_{s=1}^{\mathcal{N}_{B,k}}\tau_s n_{s,m}R_{sj}(\mathbf{v}_s^b-\mathbf{v}_j)\mathbf{f}_{m,S}(\widetilde{\mathbf{u}}_s^b,\mathbf{u}_j)=\sum_{j=1}^{\mathcal{N}_{Q,k}}\sum_{s=1}^{\mathcal{N}_{B,k}}\tau_s n_{s,m}R_{sj}(\widetilde{\psi}_{m,s}^b-\psi_{m,j})\\
=&\left(\overrightarrow{\widetilde{\psi}_m^b}\right)^T BN_mR\overrightarrow{\mathbf{1}}-\overrightarrow{\psi_m}^TR^TBN_m\overrightarrow{\mathbf{1}^b}=\left(\overrightarrow{\widetilde{\psi}_m^b}-\overrightarrow{\psi_m^b}\right)^TBN_m\overrightarrow{\mathbf{1}^b}.
\end{aligned}$$

Recall (3.15). We see that

$$\frac{d}{dt}(\overrightarrow{\mathbf{1}}^T\mathbf{M}\overrightarrow{\mathbf{u}})=\left(\overrightarrow{\mathbf{1}^b}\right)^T\mathbf{B}\overrightarrow{\mathbf{f}_{\mathbf{n}}^*},\quad \frac{d}{dt}(\overrightarrow{1}^TM\overrightarrow{U})=\left(\overrightarrow{\widetilde{\psi}_{\mathbf{n}}^b}\right)^TB\overrightarrow{1^b}-\left(\overrightarrow{\mathbf{v}^b}\right)^T\mathbf{B}\overrightarrow{\mathbf{f}_{\mathbf{n}}^*}. \tag{3.20}$$

The skew-symmetric correction term reduces to zero when multiplied by constant vector, and contributes to entropy-extrapolated values of $\psi_{\mathbf{n}}$ when multiplied by $\overrightarrow{\mathbf{v}}$. As a result, conservation is obvious, and the entropy production rate at $\boldsymbol{\xi}_s^b$ is:

$$(\mathbf{v}_s^{b,\mathrm{out}}-\mathbf{v}_s^b)^T\widehat{\mathbf{f}_{\mathbf{n}}}(\widetilde{\mathbf{u}}_s^b,\widetilde{\mathbf{u}}_s^{b,\mathrm{out}})-(\widetilde{\psi}_{\mathbf{n},s}^{b,\mathrm{out}}-\widetilde{\psi}_{\mathbf{n},s}^b)\leq 0,$$

which clinches entropy stability. □

Remark 3.3. Constructing bound-preserving limiter and TVD/TVB limiter is highly non-trivial due to the nonlinear entropy extrapolation. We also have such difficulty in the other two schemes in this chapter. Compatibility with limiters is a notable advantage of Gauss-Lobatto type nodes.

Although the boundary penalty term in (3.16) looks very complicated, it can be greatly simplified in some particular situations. In the following examples, we will study the effect of boundary treatment.

Example 3.3.1. For Gauss-Lobatto type nodes with $R = \begin{bmatrix} I_{\mathcal{N}_{B,k}} & 0 \end{bmatrix}$, $\mathbf{E}_m$ is diagonal, and $\overrightarrow{\widetilde{\mathbf{u}}^b} = \overrightarrow{\mathbf{u}^b}$. Then $\mathbf{E}_m \circ \mathbf{F}_{m,S}(\overrightarrow{\mathbf{u}}, \overrightarrow{\mathbf{u}})\overrightarrow{\mathbf{1}} = \mathbf{E}_m\overrightarrow{\mathbf{f}_m}$, and

$$\mathbf{R}^T\mathbf{B}\mathbf{N}_m\Big(\mathbf{R} \circ \mathbf{F}_{m,S}\Big(\overrightarrow{\widetilde{\mathbf{u}}^b}, \overrightarrow{\mathbf{u}}\Big)\Big)\overrightarrow{\mathbf{1}} = \mathbf{R}^T\mathbf{B}\mathbf{N}_m\mathbf{R}\overrightarrow{\mathbf{f}_m} = \mathbf{E}_m\overrightarrow{\mathbf{f}_m},$$

$$\begin{aligned}\mathbf{R}^T \circ \mathbf{F}_{m,S}\Big(\overrightarrow{\widetilde{\mathbf{u}}^b}, \overrightarrow{\mathbf{u}}\Big)^T\mathbf{B}\mathbf{N}_{\mathbf{m}}\overrightarrow{\mathbf{1}^b} &= (\mathbf{R}^T\mathbf{B}\mathbf{N}_{\mathbf{m}}) \circ \mathbf{F}_{m,S}\Big(\overrightarrow{\widetilde{\mathbf{u}}^b}, \overrightarrow{\mathbf{u}}\Big)^T\overrightarrow{\mathbf{1}^b} \\ &= \mathbf{R}^T\mathbf{B}\mathbf{N}_{\mathbf{m}}\overrightarrow{\mathbf{f}^b_m} = \mathbf{E}_m\overrightarrow{\mathbf{f}_m}.\end{aligned}$$

We use the identity $\mathbf{X} \circ (\mathbf{Y}\mathbf{Z})^T\overrightarrow{\mathbf{1}} = \mathrm{diag}(\mathbf{X}\mathbf{Y}\mathbf{Z}) = (\mathbf{X}\mathbf{Y}) \circ \mathbf{Z}^T\overrightarrow{\mathbf{1}}$ herein. Notice that all three components equal $\mathbf{E}_m\overrightarrow{\mathbf{f}_m}$, and (3.16) reduces to

$$\frac{d\overrightarrow{\mathbf{u}}}{dt} + 2\sum_{m=1}^{d}\mathbf{D}_m \circ \mathbf{F}_{m,S}(\overrightarrow{\mathbf{u}}, \overrightarrow{\mathbf{u}})\overrightarrow{\mathbf{1}} = \mathbf{M}^{-1}\Big(\sum_{m=1}^{d}\mathbf{E}_m\overrightarrow{\mathbf{f}_m} - \mathbf{R}^T\mathbf{B}\overrightarrow{\mathbf{f}^*_{\mathbf{n}}}\Big) = \mathbf{M}^{-1}\mathbf{R}^T\mathbf{B}\Big(\overrightarrow{\mathbf{f}_{\mathbf{n}}} - \overrightarrow{\mathbf{f}^*_{\mathbf{n}}}\Big).$$

Therefore, the entropy stable DGSEM (2.31) is a special case of (3.16).

Example 3.3.2. For the linear symmetric system (1.14) in one space dimension with $\mathbf{f}_S(\mathbf{u}_L, \mathbf{u}_R) = \frac{1}{2}(\mathbf{f}(\mathbf{u}_L) + \mathbf{f}(\mathbf{u}_R))$, the one-dimensional version of (3.16) is

$$\begin{aligned}\frac{d\overrightarrow{\mathbf{u}}}{dt} + 2\mathbf{D} \circ \mathbf{F}_S(\overrightarrow{\mathbf{u}}, \overrightarrow{\mathbf{u}})\overrightarrow{\mathbf{1}} = \mathbf{M}^{-1}\Big(&\mathbf{E} \circ \mathbf{F}_S(\overrightarrow{\mathbf{u}}, \overrightarrow{\mathbf{u}})\overrightarrow{\mathbf{1}} - \mathbf{R}^T\mathbf{B}\overrightarrow{\mathbf{f}^*} \\ &+ \mathbf{R}^T\mathbf{B}\Big(\mathbf{R} \circ \mathbf{F}_S\Big(\overrightarrow{\widetilde{\mathbf{u}}^b}, \overrightarrow{\mathbf{u}}\Big)\Big)\overrightarrow{\mathbf{1}} - \mathbf{R}^T \circ \mathbf{F}_S\Big(\overrightarrow{\widetilde{\mathbf{u}}^b}, \overrightarrow{\mathbf{u}}\Big)^T\mathbf{B}\overrightarrow{\mathbf{1}^b}\Big),\end{aligned} \tag{3.21}$$

where $B = \mathrm{diag}\{-1, 1\}$. We have already shown in Section 1.6 that flux differencing term equals $\mathbf{D}\overrightarrow{\mathbf{f}}$, i.e., the unmodified difference term in (3.13). The boundary components are recast into:

$$\mathbf{E} \circ \mathbf{F}_S(\overrightarrow{\mathbf{u}}, \overrightarrow{\mathbf{u}})\overrightarrow{\mathbf{1}} = \frac{1}{2}\mathbf{E}\overrightarrow{\mathbf{f}} + \frac{1}{2}(\mathbf{E}\overrightarrow{\mathbf{1}}) \circ \overrightarrow{\mathbf{f}} = \frac{1}{2}\mathbf{R}^T\mathbf{B}\overrightarrow{\mathbf{f}^b} + \frac{1}{2}\Big(\mathbf{R}^T\mathbf{B}\overrightarrow{\mathbf{1}^b}\Big) \circ \overrightarrow{\mathbf{f}},$$

$$\begin{aligned}\mathbf{R}^T\mathbf{B}\Big(\mathbf{R}\circ\mathbf{F}_S\Big(\overrightarrow{\widetilde{\mathbf{u}}^b},\overrightarrow{\mathbf{u}}\Big)\Big)\overrightarrow{\mathbf{1}} &= \frac{1}{2}\mathbf{R}^T\mathbf{B}\mathbf{R}\overrightarrow{\mathbf{f}} + \frac{1}{2}\mathbf{R}^T\mathbf{B}\Big((\mathbf{R}\overrightarrow{\mathbf{1}})\circ\overrightarrow{\widetilde{\mathbf{f}}^b}\Big)\\ &= \frac{1}{2}\mathbf{R}^T\mathbf{B}\overrightarrow{\mathbf{f}^b} + \frac{1}{2}\mathbf{R}^T\mathbf{B}\overrightarrow{\widetilde{\mathbf{f}}^b},\end{aligned}$$

$$\mathbf{R}^T\circ\mathbf{F}_S\Big(\overrightarrow{\widetilde{\mathbf{u}}^b},\overrightarrow{\mathbf{u}}\Big)^T\mathbf{B}\overrightarrow{\mathbf{1}^b} = (\mathbf{R}^T\mathbf{B})\circ\mathbf{F}_S\Big(\overrightarrow{\widetilde{\mathbf{u}}^b},\overrightarrow{\mathbf{u}}\Big)^T\overrightarrow{\mathbf{1}^b} = \frac{1}{2}\mathbf{R}^T\mathbf{B}\overrightarrow{\widetilde{\mathbf{f}}^b} + \frac{1}{2}\Big(\mathbf{R}^T\mathbf{B}\overrightarrow{\mathbf{1}^b}\Big)\circ\overrightarrow{\mathbf{f}}.$$

The resulting boundary penalty term is $\mathbf{M}^{-1}\mathbf{R}^T\mathbf{B}\Big(\overrightarrow{\mathbf{f}^b}-\overrightarrow{\mathbf{f}^*}\Big)$. We also recover the unmodified boundary term in (3.13).

Example 3.3.3. For the Burgers equation (1.16) with $f_S(u_L,u_R)=\frac{1}{6}(u_L^2+u_Lu_R+u_R^2)$. Since we have square entropy function, $u=v$ and $\overrightarrow{\widetilde{u}^b}=\overrightarrow{u^b}$. In Section 1.6 we revealed that flux differencing is equivalent to the skew-symmetric splitting. As for the boundary penalty term,

$$\begin{aligned}EF_S(\overrightarrow{u},\overrightarrow{u})\overrightarrow{1} &= \frac{1}{3}E\overrightarrow{f} + \frac{1}{6}(E\overrightarrow{u})\circ\overrightarrow{u} + \frac{1}{3}(E\overrightarrow{1})\circ\overrightarrow{f}\\ &= \frac{1}{3}R^TB\overrightarrow{f^b} + \frac{1}{6}\Big(R^TB\overrightarrow{u^b}\Big)\circ\overrightarrow{u} + \frac{1}{3}\Big(R^TB\overrightarrow{1^b}\Big)\circ\overrightarrow{f},\end{aligned}$$

$$\begin{aligned}R^TB\Big(R\circ F_S\Big(\overrightarrow{\widetilde{u}^b},\overrightarrow{u}\Big)\Big)\overrightarrow{1} &= \frac{1}{3}R^TBR\overrightarrow{f} + \frac{1}{6}R^TB\Big((R\overrightarrow{u})\circ\overrightarrow{u^b}\Big) + \frac{1}{3}R^TB\Big((R\overrightarrow{1})\circ\overrightarrow{\widetilde{f}^b}\Big)\\ &= \frac{1}{3}R^TB\overrightarrow{f^b} + \frac{2}{3}R^TB\overrightarrow{\widetilde{f}^b},\ \text{where } \overrightarrow{\widetilde{f}^b} = \frac{1}{2}\overrightarrow{u^b}\circ\overrightarrow{u^b},\end{aligned}$$

$$\begin{aligned}R^T\circ F_S\Big(\overrightarrow{\widetilde{u}^b},\overrightarrow{u}\Big)^TB\overrightarrow{1^b} &= (R^TB)\circ F_S\Big(\overrightarrow{\widetilde{u}^b},\overrightarrow{u}\Big)^T\overrightarrow{1^b}\\ &= \frac{1}{3}R^TB\overrightarrow{\widetilde{f}^b} + \frac{1}{6}\Big(R^TB\overrightarrow{u^b}\Big)\circ\overrightarrow{u} + \frac{1}{3}\Big(R^TB\overrightarrow{1^b}\Big)\circ\overrightarrow{f}.\end{aligned}$$

After summing them up, the method (3.21) turns into

$$\frac{d\overrightarrow{u}}{dt} + \frac{2}{3}D\overrightarrow{f} + \frac{1}{3}\overrightarrow{u}\circ(D\overrightarrow{u}) = M^{-1}R^TB\Big(\frac{2}{3}\overrightarrow{f^b} + \frac{1}{3}\overrightarrow{\widetilde{f}^b} - \overrightarrow{f^*}\Big).$$

Hence the boundary adjustment also corresponds to some splitting procedure (this is exactly the entropy stable DGSEM in [86]).

3.4 Approach 2: replacing bivariate interface flux

The second entropy stable DGSEM was found by Crean et al in [21]. We still try to modify the scheme (3.14). Instead of plugging correction terms, we implement element coupling in a different way by replacing vector $\overrightarrow{\mathbf{f}_{\mathbf{n}}^*}$. Suppose that K has $\mathcal{N}_E$ faces, and $\{\gamma_e\}_{e=1}^{\mathcal{N}_E}$ is the collection of faces of K. There are $\mathcal{N}_{F,k}$ boundary quadrature points on each face so that $\mathcal{N}_{B,k} = \mathcal{N}_E \mathcal{N}_{F,k}$. Furthermore,

$$\mathbf{B} = \mathrm{diag}\{\mathbf{B}^1, \cdots, \mathbf{B}^{\mathcal{N}_E}\}, \quad \mathbf{N}_m = \mathrm{diag}\{\mathbf{N}_m^1, \cdots, \mathbf{N}_m^{\mathcal{N}_E}\},$$

where $\mathbf{B}^e$ and $\mathbf{N}_m^e$ are $\mathcal{N}_{F,k} \times \mathcal{N}_{F,k}$ blocks corresponding to the face γ_e. The extrapolation matrix is also decomposed as

$$\mathbf{R} = \begin{bmatrix} \mathbf{R}^1 \\ \vdots \\ \mathbf{R}^{\mathcal{N}_E} \end{bmatrix}.$$

For each face γ_e, let $K^{e,\mathrm{out}}$ be the adjacent element on the other side of γ_e, and $\overrightarrow{\mathbf{u}^{e,\mathrm{out}}}$ be the solution vector at $K^{e,\mathrm{out}}$. Define $1 \leq \sigma(e) \leq \mathcal{N}_E$ such that γ_e is the $\sigma(e)$-th face of $K^{e,\mathrm{out}}$ (by considering the affine mapping between K and $K^{e,\mathrm{out}}$). The extrapolated values on both sides of γ_e are given by

$$\overrightarrow{\mathbf{u}^{b,e}} := \mathbf{R}^e \overrightarrow{\mathbf{u}}, \quad \overrightarrow{\mathbf{u}^{b,e,\mathrm{out}}} := \mathbf{R}^{\sigma(e)} \overrightarrow{\mathbf{u}^{e,\mathrm{out}}}.$$

Then $\overrightarrow{\mathbf{u}^b}$ and $\overrightarrow{\mathbf{u}^{b,\mathrm{out}}}$ are concatenations of these block vectors:

$$\overrightarrow{\mathbf{u}^b} = \begin{bmatrix} \overrightarrow{\mathbf{u}^{b,1}} \\ \vdots \\ \overrightarrow{\mathbf{u}^{b,\mathcal{N}_E}} \end{bmatrix}, \quad \overrightarrow{\mathbf{u}^{b,\mathrm{out}}} = \begin{bmatrix} \overrightarrow{\mathbf{u}^{b,1,\mathrm{out}}} \\ \vdots \\ \overrightarrow{\mathbf{u}^{b,\mathcal{N}_E,\mathrm{out}}} \end{bmatrix}.$$

We also set

$$\mathbf{E}_m^e := (\mathbf{R}^e)^T\mathbf{B}^e\mathbf{N}_m^e\mathbf{R}^e, \quad \mathbf{E}_m^{e,\text{out}} := (\mathbf{R}^e)^T\mathbf{B}^e\mathbf{N}_m^e\mathbf{R}^{\sigma(e)}, \quad 1 \le e \le \mathcal{N}_E. \tag{3.22}$$

Clearly $\mathbf{E}_m = \sum_{e=1}^{\mathcal{N}_E}\mathbf{E}_m^e$. By designing a new coupling term $\mathbf{E}_m^{e,\text{out}} \circ \mathbf{F}_{m,S}\Big(\overrightarrow{\mathbf{u}}, \overrightarrow{\mathbf{u}^{e,\text{out}}}\Big)\overrightarrow{\mathbf{1}}$, we produce the following entropy *conservative* DGSEM:

$$\begin{aligned}&\frac{d\overrightarrow{\mathbf{u}}}{dt} + 2\sum_{m=1}^{d}\mathbf{D}_m \circ \mathbf{F}_{m,S}(\overrightarrow{\mathbf{u}}, \overrightarrow{\mathbf{u}})\overrightarrow{\mathbf{1}}\\ &\quad = \mathbf{M}^{-1}\Big(\sum_{m=1}^{d}\sum_{e=1}^{\mathcal{N}_E}\Big(\mathbf{E}_m^e \circ \mathbf{F}_{m,S}(\overrightarrow{\mathbf{u}}, \overrightarrow{\mathbf{u}})\overrightarrow{\mathbf{1}} - \mathbf{E}_m^{e,\text{out}} \circ \mathbf{F}_{m,S}\Big(\overrightarrow{\mathbf{u}}, \overrightarrow{\mathbf{u}^{e,\text{out}}}\Big)\overrightarrow{\mathbf{1}}\Big)\Big),\end{aligned} \tag{3.23}$$

with the component-wise representation

$$\begin{aligned}&\frac{d\mathbf{u}_j}{dt} + 2\sum_{m=1}^{d}\sum_{l=1}^{\mathcal{N}_{Q,k}} D_{m,jl}\mathbf{f}_{m,S}(\mathbf{u}_j, \mathbf{u}_l)\\ &\quad = \sum_{e=1}^{\mathcal{N}_E}\sum_{r=1}^{\mathcal{N}_{F,k}} R_{sj}\frac{\tau_s}{\omega_j}\Big(\sum_{l=1}^{\mathcal{N}_{Q,k}} R_{sl}\mathbf{f}_{\mathbf{n},S}(\mathbf{u}_j, \mathbf{u}_l) - \sum_{l=1}^{\mathcal{N}_{Q,k}} R_{rl}^{\sigma(e)}\mathbf{f}_{\mathbf{n},S}(\mathbf{u}_j, \mathbf{u}_l^{e,\text{out}})\Big)\end{aligned} \tag{3.24}$$

where $s = (e-1)\mathcal{N}_{F,k} + r$.

Theorem 3.3. *Under the same assumptions as in Theorem 3.2, scheme (3.23) is high order accurate, conservative and entropy conservative.*

Proof. For accuracy, we only need to prove that for all i, j and smooth solution $\mathbf{u}$,

$$\sum_{l=1}^{\mathcal{N}_{Q,k}} R_{sl}\mathbf{f}_{\mathbf{n},S}(\mathbf{u}_{i,j}, \mathbf{u}_{i,l}) - \sum_{l=1}^{\mathcal{N}_{Q,k}} R_{rl}^{\sigma_i(e)}\mathbf{f}_{\mathbf{n},S}(\mathbf{u}_{i,j}, \mathbf{u}_{i,l}^{e,\text{out}}) = \mathcal{O}(h^{k+1}).$$

By the approximation property of extrapolation,

$$\sum_{l=1}^{\mathcal{N}_{Q,k}} R_{sl}\mathbf{f}_{\mathbf{n},S}(\mathbf{u}_{i,j}, \mathbf{u}_{i,l}) = \mathbf{f}_{\mathbf{n},S}(\mathbf{u}_{i,j}, \mathbf{u}(\mathbf{x}_i(\boldsymbol{\xi}_s^b))) + \mathcal{O}(h^{k+1}),$$

$$\sum_{l=1}^{\mathcal{N}_{Q,k}} R_{rl}^{\sigma_i(e)} \mathbf{f}_{\mathbf{n},S}(\mathbf{u}_{i,j}, \mathbf{u}_{i,l}^{e,\text{out}}) = \mathbf{f}_{\mathbf{n},S}(\mathbf{u}_{i,j}, \mathbf{u}(\mathbf{x}_i(\boldsymbol{\xi}_s^b))) + \mathcal{O}(h^{k+1}).$$

Then the boundary truncation error is of high order. For conservation and entropy conservation, $\mathbf{D}_m \circ \mathbf{F}_{m,S}(\overrightarrow{\mathbf{u}}, \overrightarrow{\mathbf{u}})\overrightarrow{\mathbf{1}}$ and $\mathbf{E}_m \circ \mathbf{F}_{m,S}(\overrightarrow{\mathbf{u}}, \overrightarrow{\mathbf{u}})\overrightarrow{\mathbf{1}}$ cancel with each other, we are left with

$$\frac{d}{dt}(\overrightarrow{\mathbf{1}}^T \mathbf{M} \overrightarrow{\mathbf{u}}) = -\sum_{m=1}^{d} \sum_{e=1}^{\mathcal{N}_E} \overrightarrow{\mathbf{1}}^T \mathbf{E}_m^{e,\text{out}} \circ \mathbf{F}_{m,S}\left(\overrightarrow{\mathbf{u}}, \overrightarrow{\mathbf{u}^{e,\text{out}}}\right) \overrightarrow{\mathbf{1}},$$

$$\frac{d}{dt}(\overrightarrow{1}^T M \overrightarrow{U}) = \sum_{e=1}^{\mathcal{N}_E} \left(\overrightarrow{\psi_{\mathbf{n}}^{b,e}}\right)^T B^e \overrightarrow{1^{b,e}} - \sum_{m=1}^{d} \sum_{e=1}^{\mathcal{N}_E} \overrightarrow{\mathbf{v}}^T \mathbf{E}_m^{e,\text{out}} \circ \mathbf{F}_{m,S}\left(\overrightarrow{\mathbf{u}}, \overrightarrow{\mathbf{u}^{e,\text{out}}}\right) \overrightarrow{\mathbf{1}}.$$

On a face $\gamma(e)$, the corresponding contributions from $K^{e,\text{out}}$ are

$$-\sum_{m=1}^{d} \overrightarrow{\mathbf{1}}^T (\mathbf{R}^{\sigma(e)} \mathbf{B}^e (-\mathbf{N}_m^e) \mathbf{R}^e) \circ \mathbf{F}_{m,S}\left(\overrightarrow{\mathbf{u}^{e,\text{out}}}, \overrightarrow{\mathbf{u}}\right) \overrightarrow{\mathbf{1}} = \sum_{m=1}^{d} \overrightarrow{\mathbf{1}}^T (\mathbf{E}^{e,\text{out}})^T \circ \mathbf{F}_{m,S}\left(\overrightarrow{\mathbf{u}}, \overrightarrow{\mathbf{u}^{e,\text{out}}}\right)^T \overrightarrow{\mathbf{1}},$$

and

$$-\left(\overrightarrow{\psi_{\mathbf{n}}^{b,e,\text{out}}}\right)^T B^e \overrightarrow{1^{b,e}} + \sum_{m=1}^{d} \left(\overrightarrow{\mathbf{v}^{e,\text{out}}}\right)^T (\mathbf{E}^{e,\text{out}})^T \circ \mathbf{F}_{m,S}\left(\overrightarrow{\mathbf{u}}, \overrightarrow{\mathbf{u}^{e,\text{out}}}\right)^T \overrightarrow{\mathbf{1}}.$$

Conservation is obvious, and entropy conservation results from

$$\begin{aligned}
&\sum_{m=1}^{d} \left(\left(\overrightarrow{\mathbf{v}^{e,\text{out}}}\right)^T (\mathbf{E}^{e,\text{out}})^T \circ \mathbf{F}_{m,S}\left(\overrightarrow{\mathbf{u}}, \overrightarrow{\mathbf{u}^{e,\text{out}}}\right)^T \overrightarrow{\mathbf{1}} - \overrightarrow{\mathbf{v}}^T \mathbf{E}_m^{e,\text{out}} \circ \mathbf{F}_{m,S}\left(\overrightarrow{\mathbf{u}}, \overrightarrow{\mathbf{u}^{e,\text{out}}}\right) \overrightarrow{\mathbf{1}} \right) \\
=& \sum_{m=1}^{d} \sum_{j=1}^{\mathcal{N}_{Q,k}} \sum_{l=1}^{\mathcal{N}_{Q,k}} E_{m,jl}^{e,\text{out}} (\mathbf{v}_l^{e,\text{out}} - \mathbf{v}_j)^T \mathbf{f}_{m,S}(\mathbf{u}_j, \mathbf{u}_l^{e,\text{out}}) = \sum_{m=1}^{d} \sum_{j=1}^{\mathcal{N}_{Q,k}} \sum_{l=1}^{\mathcal{N}_{Q,k}} E_{m,jl}^{e,\text{out}} (\psi_l^{e,\text{out}} - \psi_j) \\
=& \sum_{m=1}^{d} \left(\overrightarrow{\psi}^T E_m^{e,\text{out}} \overrightarrow{1} - \overrightarrow{1}^T E_m^{e,\text{out}} \overrightarrow{\psi^{e,\text{out}}} \right) = \left(\overrightarrow{\psi_{\mathbf{n}}^{b,e,\text{out}}} - \overrightarrow{\psi_{\mathbf{n}}^{b,e}} \right)^T B^e \overrightarrow{1^{b,e}}.
\end{aligned}$$

□

Remark 3.4. The coupling term $\mathbf{E}_m^{e,\text{out}} \circ \mathbf{F}_{m,S}\left(\overrightarrow{\mathbf{u}}, \overrightarrow{\mathbf{u}^{e,\text{out}}}\right) \overrightarrow{\mathbf{1}}$ requires the nodal values of all neighboring elements. This affects the locality of DG type formulation, and makes the implementation of non-periodic boundary conditions (inflow, outflow, solid wall, etc) very difficult. We can not simply prescribe values of $\overrightarrow{\mathbf{u}^{b,e,\text{out}}}$.

Remark 3.5. For one-dimensional linear symmetric system (1.14),

$$\mathbf{E}^e \circ \mathbf{F}_S(\overrightarrow{\mathbf{u}}, \overrightarrow{\mathbf{u}})\overrightarrow{\mathbf{1}} = \frac{1}{2}(\mathbf{R}^e)^T\mathbf{B}^e\overrightarrow{\mathbf{f}^{b,e}} + \frac{1}{2}\Big((\mathbf{R}^e)^T\mathbf{B}^e\overrightarrow{\mathbf{1}^{b,e}}\Big)\circ\overrightarrow{\mathbf{f}},$$

$$\mathbf{E}^{e,\text{out}} \circ \mathbf{F}_S(\overrightarrow{\mathbf{u}}, \overrightarrow{\mathbf{u}^{e,\text{out}}})\overrightarrow{\mathbf{1}} = \frac{1}{2}(\mathbf{R}^e)^T\mathbf{B}^e\overrightarrow{\mathbf{f}^{b,e,\text{out}}} + \frac{1}{2}\Big((\mathbf{R}^e)^T\mathbf{B}^e\overrightarrow{\mathbf{1}^{b,e}}\Big)\circ\overrightarrow{\mathbf{f}},$$

The boundary penalty term is

$$\sum_{e=1}^{\mathcal{N}_E}\frac{1}{2}(\mathbf{R}^e)^T\mathbf{B}^e\Big(\overrightarrow{\mathbf{f}^{b,e}} - \overrightarrow{\mathbf{f}^{b,e,\text{out}}}\Big) = \frac{1}{2}\mathbf{R}^T\mathbf{B}\Big(\overrightarrow{\mathbf{f}^{b}} - \overrightarrow{\mathbf{f}^{b,\text{out}}}\Big) = \mathbf{R}^T\mathbf{B}\Big(\overrightarrow{\mathbf{f}^{e}} - \overrightarrow{\mathbf{f}^{*}}\Big),$$

where $\overrightarrow{\mathbf{f}^{*}} = \frac{1}{2}\Big(\overrightarrow{\mathbf{f}^{e}} + \overrightarrow{\mathbf{f}^{e,\text{out}}}\Big)$. Hence (3.23) also reduces to the unmodified DGSEM (3.13), by using the entropy conservative interface flux:

$$\widehat{\mathbf{f}}(\mathbf{u}_L, \mathbf{u}_R) = \mathbf{f}_S(\mathbf{u}_L, \mathbf{u}_R) = \frac{1}{2}(\mathbf{f}(\mathbf{u}_L) + \mathbf{f}(\mathbf{u}_R)).$$

In order to make (3.23) an entropy stable scheme, we define the entropy stable dissipation function $\widehat{\mathbf{d}}_{\mathbf{n}}(\mathbf{u}, \mathbf{u}^{\text{out}})$. It is essentially the dissipative part of entropy stable flux $\widehat{\mathbf{f}}_{\mathbf{n}}(\mathbf{u}, \mathbf{u}^{\text{out}})$.

Definition 3.1. *A bivariate function $\widehat{\mathbf{d}}_{\mathbf{n}}(\mathbf{u}, \mathbf{u}^{out})$ serves as an entropy stable dissipation function with respect to an entropy function U if it satisfies the following conditions:*

1. *Consistency:* $\widehat{\mathbf{d}}_{\mathbf{n}}(\mathbf{u}, \mathbf{u}) = 0$.
2. *Conservation:* $\widehat{\mathbf{d}}_{\mathbf{n}}(\mathbf{u}, \mathbf{u}^{out}) = -\widehat{\mathbf{d}}_{-\mathbf{n}}(\mathbf{u}^{out}, \mathbf{u})$.
3. *Entropy stability:* $(\mathbf{v}^{out} - \mathbf{v})^T\widehat{\mathbf{d}}_{\mathbf{n}}(\mathbf{u}, \mathbf{u}^{out}) \leq 0$.

For example, the Lax-Friedrichs type dissipation function is commonly used:

$$\widehat{\mathbf{d}}_{\mathbf{n}}(\mathbf{u},\mathbf{u}^{\text{out}}) = \lambda_{\mathbf{n}}(\mathbf{u},\mathbf{u}^{\text{out}})(\mathbf{u}^{\text{out}}-\mathbf{u}). \tag{3.25}$$

$\lambda_{\mathbf{n}}(\mathbf{u},\mathbf{u}^{\text{out}}) \geq 0$ is the maximum wave speed. It can be the largest absolute eigenvalue in $\mathbf{f}'_{\mathbf{n}}(\mathbf{u})$ and $\mathbf{f}'_{\mathbf{n}}(\mathbf{u}^{\text{out}})$, or the two-rarefaction approximated wave speed in Appendix A. Lax-Friedrichs type dissipation function is entropy stable in that

$$\lambda_{\mathbf{n}}(\mathbf{u},\mathbf{u}^{\text{out}})(\mathbf{v}-\mathbf{v}^{\text{out}})^T(\mathbf{u}^{\text{out}}-\mathbf{u}) = \lambda_{\mathbf{n}}(\mathbf{u},\mathbf{u}^{\text{out}})(\mathbf{u}-\mathbf{u}^{\text{out}})^T\mathbf{v}'(\widetilde{\mathbf{u}})(\mathbf{u}^{\text{out}}-\mathbf{u}) \geq 0,$$

where $\widetilde{\mathbf{u}}$ is some value on the line segment connecting $\mathbf{u}$ and $\mathbf{u}^{\text{out}}$. We generate entropy stable DGSEM by adding entropy dissipation to (3.23). Let $\overrightarrow{\mathbf{d}^*_{\mathbf{n}}}$ be the vector of entropy dissipation terms, such that the arguments are entropy-extrapolated values:

$$\overrightarrow{\mathbf{d}^*_{\mathbf{n}}} := \begin{bmatrix} \widehat{\mathbf{d}}_{\mathbf{n}}(\widetilde{\mathbf{u}}^b_1, \widetilde{\mathbf{u}}^{b,\text{out}}_1) \\ \vdots \\ \widehat{\mathbf{d}}_{\mathbf{n}}(\widetilde{\mathbf{u}}^b_{\mathcal{N}_{B,k}}, \widetilde{\mathbf{u}}^{b,\text{out}}_{\mathcal{N}_{B,k}}) \end{bmatrix}.$$

The entropy stable scheme reads

$$\begin{aligned} \frac{d\overrightarrow{\mathbf{u}}}{dt} + 2\sum_{m=1}^{d} \mathbf{D}_m \circ \mathbf{F}_{m,S}(\overrightarrow{\mathbf{u}},\overrightarrow{\mathbf{u}})\overrightarrow{\mathbf{1}} = \mathbf{M}^{-1}\Bigg(& \sum_{m=1}^{d}\sum_{e=1}^{\mathcal{N}_E}\Big(\mathbf{E}^e_m \circ \mathbf{F}_{m,S}(\overrightarrow{\mathbf{u}},\overrightarrow{\mathbf{u}})\overrightarrow{\mathbf{1}} \\ & - \mathbf{E}^{e,\text{out}}_m \circ \mathbf{F}_{m,S}\Big(\overrightarrow{\mathbf{u}},\overrightarrow{\mathbf{u}^{e,\text{out}}}\Big)\overrightarrow{\mathbf{1}}\Big) - \mathbf{R}^T\mathbf{B}\overrightarrow{\mathbf{d}^*_{\mathbf{n}}}\Bigg). \end{aligned} \tag{3.26}$$

The proof of high order accuracy, conservation, entropy stability follows respectively from the consistency, conservation and entropy stability of $\widehat{\mathbf{d}}_{\mathbf{n}}$.

Corollary 3.1. *Under the same assumptions as in Theorem 3.2, if* $\widehat{\mathbf{d}}_{\mathbf{n}}$ *is an entropy stable dissipation function, then the scheme (3.26) is high order accurate, conservative and entropy stable.*

3.5 Approach 3: eliminating aliasing error

In [1], Abgrall recommended a simple "brute force" type approach that enforces entropy stability. It is written in the residual distribution framework. We will the apply the idea to the DGSEM formulations. We start with the original DGSEM (3.13), using entropy conservative flux at element interface:

$$\mathbf{f}^*_{\mathbf{n},s} = \mathbf{f}_{\mathbf{n},S}(\widetilde{\mathbf{u}}^b_s, \widetilde{\mathbf{u}}^{b,\text{out}}_s), \quad 1 \leq s \leq \mathcal{N}_{B,k}.$$

Notice that the arguments are again entropy-extrapolated values. We also set $\overrightarrow{F^*_{\mathbf{n}}}$ to be the vector of bivariate entropy flux functions:

$$F^*_{\mathbf{n},s} = F_{\mathbf{n},S}(\widetilde{\mathbf{u}}^b_s, \widetilde{\mathbf{u}}^{b,\text{out}}_s) := \frac{1}{2}(\mathbf{v}^b_s + \mathbf{v}^{b,\text{out}}_s)^T \mathbf{f}_{\mathbf{n},S}(\widetilde{\mathbf{u}}^b_s, \widetilde{\mathbf{u}}^{b,\text{out}}_s) - \frac{1}{2}\left(\widetilde{\psi}^b_{\mathbf{n},s} + \widetilde{\psi}^{b,\text{out}}_{\mathbf{n},s}\right).$$

The aliasing error of (3.13) is defined as:

$$\begin{aligned}\mathcal{E} :=& \frac{d}{dt}(\overrightarrow{1}^T M \overrightarrow{U}) + \left(\overrightarrow{1^b}\right)^T B \overrightarrow{F^*_{\mathbf{n}}} = \overrightarrow{\mathbf{v}}^T\Big(- \sum_{m=1}^{d} \mathbf{S}_m \overrightarrow{\mathbf{f}_m} + \mathbf{R}^T\mathbf{B}\Big(\overrightarrow{\mathbf{f}^b_{\mathbf{n}}} - \overrightarrow{\mathbf{f}^*_{\mathbf{n}}}\Big)\Big) + \left(\overrightarrow{1^b}\right)^T B \overrightarrow{F^*_{\mathbf{n}}} \\ =& \sum_{m=1}^{d} \overrightarrow{\mathbf{v}}^T \mathbf{S}^T_m \overrightarrow{\mathbf{f}_m} - \left(\overrightarrow{\mathbf{v}^b}\right)^T \mathbf{B}\overrightarrow{\mathbf{f}^*_{\mathbf{n}}} + \left(\overrightarrow{1^b}\right)^T B \overrightarrow{F^*_{\mathbf{n}}}.\end{aligned}$$

Since

$$(\mathbf{v}^b_s)^T \mathbf{f}_{\mathbf{n}}(\widetilde{\mathbf{u}}^b_s, \widetilde{\mathbf{u}}^{b,\text{out}}_s) - F_{\mathbf{n}}(\widetilde{\mathbf{u}}^b_s, \widetilde{\mathbf{u}}^{b,\text{out}}_s) = \frac{1}{2}(\mathbf{v}^b_s - \mathbf{v}^{b,\text{out}}_s)^T \mathbf{f}_{\mathbf{n},S}(\widetilde{\mathbf{u}}^b_s, \widetilde{\mathbf{u}}^{b,\text{out}}_s) + \frac{1}{2}\left(\widetilde{\psi}^b_{\mathbf{n},s} + \widetilde{\psi}^{b,\text{out}}_{\mathbf{n},s}\right) = \widetilde{\psi}^b_{\mathbf{n},s},$$

We obtain

$$\mathcal{E} = \sum_{m=1}^{d} \overrightarrow{\mathbf{v}}^T \mathbf{S}^T_m \overrightarrow{\mathbf{f}_m} - \left(\overrightarrow{1^b}\right)^T B \overrightarrow{\widetilde{\psi}^b_{\mathbf{n}}}. \tag{3.27}$$

We will demonstrate that for smooth solutions, the aliasing error is of high order.

Theorem 3.4. *Assume that the sequence of meshes $\{\mathcal{T}_h\}$ is uniform, and that all* *mappings and bivariate fluxes are smooth and Lipschitz continuous. Then for each* $1 \leq i \leq N$ *and smooth solution* $\mathbf{u}$*, the local aliasing error* $\mathcal{E}_i = \mathcal{O}(h^{k+d})$*.*

Proof. (3.27) describes the discretization error of Green's formula:

$$\int_{K_i} \sum_{m=1}^{d} \frac{\partial \mathbf{v}^T}{\partial x_m} \mathbf{f}_m(\mathbf{u}) d\mathbf{x} = \int_{K_i} \sum_{m=1}^{d} \frac{\partial \mathbf{v}^T}{\partial x_m} \mathbf{g}_m(\mathbf{v}) d\mathbf{x} = \int_{\partial K_i} \psi_{\mathbf{n}}(\mathbf{v}) dS.$$

By the approximation property of difference and extrapolation matrix, and algebraic accuracy of internal and boundary quadrature rule,

$$\begin{aligned}\int_{K_i} \sum_{m=1}^{d} \frac{\partial \mathbf{v}^T}{\partial x_m} \mathbf{f}_m(\mathbf{u}) d\mathbf{x} &= \sum_{m=1}^{d} \sum_{j=1}^{\mathcal{N}_{Q,k}} J_i \omega_j \frac{\partial \mathbf{v}^T}{\partial x_m} (\mathbf{x}_i(\boldsymbol{\xi}_j)) \mathbf{f}_{i,m,j} + \mathcal{O}(h^{2k+d}) \\ &= \sum_{m=1}^{d} (\mathbf{D}_i \overrightarrow{\mathbf{v}_i})^T \mathbf{M}_i \overrightarrow{\mathbf{f}_{i,m}} + \mathcal{O}(h^{k+d}) = \sum_{m=1}^{d} \overrightarrow{\mathbf{v}_i}^T \mathbf{S}_{i,m}^T \overrightarrow{\mathbf{f}_{i,m}} + \mathcal{O}(h^{k+d}),\end{aligned}$$

$$\begin{aligned}\int_{\partial K_i} \psi_{\mathbf{n}}(\mathbf{v}) dS &= \sum_{s=1}^{\mathcal{N}_{B,k}} J_{i,s}^b \tau_s \psi_{\mathbf{n}}(\mathbf{v}(\mathbf{x}_i(\boldsymbol{\xi}_s^b))) + \mathcal{O}(h^{2k+d}) = \sum_{s=1}^{\mathcal{N}_{B,k}} J_{i,s}^b \tau_s \psi_{\mathbf{n}}(\mathbf{v}_{i,s}^b) + \mathcal{O}(h^{k+d}) \\ &= \left(\overrightarrow{1^b}\right)^T B_i \overrightarrow{\widetilde{\psi}_{i,\mathbf{n}}^b} + \mathcal{O}(h^{k+d}).\end{aligned}$$

Hence $\mathcal{E}_i = \sum_{m=1}^{d} \overrightarrow{\mathbf{v}_i}^T \mathbf{S}_{i,m}^T \overrightarrow{\mathbf{f}_{i,m}} - \left(\overrightarrow{1^b}\right)^T B_i \overrightarrow{\widetilde{\psi}_{i,\mathbf{n}}^b} = \mathcal{O}(h^{k+d})$. □

To neutralize the aliasing error, a simple linear correction term will be introduced to (3.13), resulting in the following entropy conservative DGSEM.

$$\frac{d\overrightarrow{\mathbf{u}}}{dt} + \sum_{m=1}^{d} \mathbf{D}_m \overrightarrow{\mathbf{f}_m} = \mathbf{M}^{-1} \mathbf{R}^T \left(\overrightarrow{\mathbf{f}_{\mathbf{n}}^b} - \overrightarrow{\mathbf{f}_{\mathbf{n}}^*}\right) - \alpha \mathbf{M}^{-1} \overrightarrow{\mathbf{v}^b}, \tag{3.28}$$

where

$$\alpha = \frac{\mathcal{E}}{\left(\overrightarrow{\mathbf{v}^b}\right)^T \overrightarrow{\mathbf{v}^b}},$$

and $\overrightarrow{\mathbf{v}^o}$ is the vector of normalized values:

$$\mathbf{v}_j^o = \mathbf{v}_j - \overline{\mathbf{v}}, \quad \overline{\mathbf{v}} := \frac{1}{\mathcal{N}_{Q,k}} \sum_{j=1}^{\mathcal{N}_{Q,k}} \mathbf{v}_j.$$

Theorem 3.5. *If* $\mathbf{f}_{m,S}$ *is entropy conservative for each* $1 \leq m \leq d$*, then the scheme* (3.28) *is conservative and entropy conservative.*

Proof. For conservation,

$$\frac{d}{dt}(\overrightarrow{\mathbf{1}}^T \mathbf{M} \overrightarrow{\mathbf{u}}) = -\left(\overrightarrow{\mathbf{1}^b}\right)^T \mathbf{B} \overrightarrow{\mathbf{f}_{\mathbf{n}}^*} - \alpha\, \overrightarrow{\mathbf{1}}^T \overrightarrow{\mathbf{v}^o} = -\left(\overrightarrow{\mathbf{1}^b}\right)^T \mathbf{B} \overrightarrow{\mathbf{f}_{\mathbf{n}}^*}.$$

as $\overrightarrow{\mathbf{1}}^T \overrightarrow{\mathbf{v}^o} = 0$. For entropy conservation, by the definition of $\mathcal{E}$,

$$\frac{d}{dt}(\overrightarrow{1}^T M \overrightarrow{U}) = -\left(\overrightarrow{1^b}\right)^T B \overrightarrow{F_{\mathbf{n}}^*} + \mathcal{E} - \alpha\, \overrightarrow{\mathbf{v}}^T \overrightarrow{\mathbf{v}^o} = -\left(\overrightarrow{1^b}\right)^T B \overrightarrow{F_{\mathbf{n}}^*}.$$

We use the relation

$$\alpha \overrightarrow{\mathbf{v}}^T \overrightarrow{\mathbf{v}^o} = \mathcal{E} \frac{\sum_{j=1}^{\mathcal{N}_{Q,k}} \mathbf{v}_j^T (\mathbf{v}_j - \overline{\mathbf{v}})}{\sum_{j=1}^{\mathcal{N}_{Q,k}} (\mathbf{v}_j - \overline{\mathbf{v}})^T (\mathbf{v}_j - \overline{\mathbf{v}})} = \mathcal{E} \frac{\sum_{j=1}^{\mathcal{N}_{Q,k}} \mathbf{v}_j^T \mathbf{v}_j - \mathcal{N}_{Q,k} \overline{\mathbf{v}}^T \overline{\mathbf{v}}}{\sum_{j=1}^{\mathcal{N}_{Q,k}} \mathbf{v}_j^T \mathbf{v}_j - \mathcal{N}_{Q,k} \overline{\mathbf{v}}^T \overline{\mathbf{v}}} = \mathcal{E}.$$

Due to the symmetry of $\mathbf{f}_{\mathbf{n},S}$ and $F_{\mathbf{n},S}$,

$$\mathbf{f}_{-\mathbf{n},S}(\widetilde{\mathbf{u}}_s^{b,\text{out}}, \widetilde{\mathbf{u}}_s^b) = -\mathbf{f}_{\mathbf{n},S}(\widetilde{\mathbf{u}}_s^b, \widetilde{\mathbf{u}}_s^{b,\text{out}}), \quad F_{-\mathbf{n},S}(\widetilde{\mathbf{u}}_s^{b,\text{out}}, \widetilde{\mathbf{u}}_s^b) = -F_{\mathbf{n},S}(\widetilde{\mathbf{u}}_s^b, \widetilde{\mathbf{u}}_s^{b,\text{out}}),$$

which implies conservation and entropy conservation. □

Remark 3.6. In numerical implementation, we should take

$$\alpha = \frac{\mathcal{E}}{\left(\overrightarrow{\mathbf{v}^o}\right)^T \overrightarrow{\mathbf{v}^o} + \varepsilon}$$

to avoid division by zero, where ε is positive and close to machine precision.

Remark 3.7. Although we have proved that $\mathcal{E}_i$ is of high order, this does not guarantee high order accuracy of (3.28). The main reason is that $\overrightarrow{\mathbf{v}_i^b} = \mathcal{O}(h)$, and we are not able to control the order of $\alpha_i = \mathcal{O}(h^{k+d})/\mathcal{O}(h^2)$.

Similar to Section 3.4, we make (3.28) entropy stable by attaching the dissipation vector $\overrightarrow{\mathbf{d}_{\mathbf{n}}^*}$:

$$\frac{d\overrightarrow{\mathbf{u}}}{dt} + \sum_{m=1}^{d} \mathbf{D}_m \overrightarrow{\mathbf{f}_m} = \mathbf{M}^{-1}\mathbf{R}^T\left(\overrightarrow{\mathbf{f}_{\mathbf{n}}^b} - \overrightarrow{\mathbf{f}_{\mathbf{n}}^*} - \overrightarrow{\mathbf{d}_{\mathbf{n}}^*}\right) - \alpha\mathbf{M}^{-1}\overrightarrow{\mathbf{v}^b}, \tag{3.29}$$

Corollary 3.2. *If $\mathbf{f}_{m,S}$ is entropy conservative for each $1 \le m \le d$, and $\widehat{\mathbf{d}}_{\mathbf{n}}$ is an entropy stable dissipation function, then the scheme (3.29) is conservative and entropy stable.*

3.6 Convection-diffusion equations

Recall the mixed form of convection-diffusion equations

$$\frac{\partial \mathbf{u}}{\partial t} + \sum_{m=1}^{d} \frac{\partial}{\partial x_m}(\mathbf{f}_m(\mathbf{u}) - \mathbf{q}_m) = 0, \quad \mathbf{q}_m = \sum_{r=1}^{d} C_{mr}(\mathbf{v})\boldsymbol{\theta}_r, \quad \boldsymbol{\theta}_r = \frac{\partial \mathbf{v}}{\partial x_r}.$$

We use $\overrightarrow{\mathbf{L}}\left(\overrightarrow{\mathbf{u}}; \left\{\overrightarrow{\mathbf{u}^{e,\text{out}}}\right\}_{e=1}^{\mathcal{N}_E}\right)$ to represent the generic form of entropy stable discretization of the convective part, including (3.16), (3.26) and (3.29). For the diffusive part, the extension to general set of nodes is much easier. We just keep the LDG type formulation:

$$\frac{d\overrightarrow{\mathbf{u}}}{dt} + \overrightarrow{\mathbf{L}}\left(\overrightarrow{\mathbf{u}}; \left\{\overrightarrow{\mathbf{u}^{e,\text{out}}}\right\}_{e=1}^{\mathcal{N}_E}\right) - \sum_{m=1}^{d} \mathbf{D}_m \overrightarrow{\mathbf{q}_m} = -\mathbf{M}^{-1}\mathbf{R}^T\mathbf{B}\left(\overrightarrow{\mathbf{q}_{\mathbf{n}}^b} - \overrightarrow{\mathbf{q}_{\mathbf{n}}^*}\right), \tag{3.30a}$$

$$\overrightarrow{\boldsymbol{\theta}_r} - \mathbf{D}_r\overrightarrow{\mathbf{v}} = -\mathbf{M}^{-1}\mathbf{R}^T\mathbf{B}\mathbf{N}_r\left(\overrightarrow{\mathbf{v}^b} - \overrightarrow{\mathbf{v}^*}\right), \quad 1 \le r \le d, \tag{3.30b}$$

where $\mathbf{v}_s^* = \widehat{\mathbf{v}}(\mathbf{v}_s^b, \mathbf{v}_s^{b,\text{out}})$ and $\mathbf{q}_{\mathbf{n},s}^* = \widehat{\mathbf{q}}_{\mathbf{n}}(\mathbf{v}_s^b, \mathbf{v}_s^{b,\text{out}}, \mathbf{q}_{\mathbf{n},s}^b, \mathbf{q}_{\mathbf{n},s}^{b,\text{out}})$. We can still establish the entropy stability of (3.30) with LDG fluxes. The proof is exactly the same as in Theorem 2.6.

Theorem 3.6. *Given parameters $\alpha \geq 0$ and $\beta \in \mathbb{R}$, if we the LDG fluxes*

$$\begin{aligned}
&\widehat{\mathbf{v}}(\mathbf{v}, \mathbf{v}^{out}) = \frac{1}{2}(\mathbf{v} + \mathbf{v}^{out}) + \beta(\mathbf{v} - \mathbf{v}^{out}),\\
&\widehat{\mathbf{q}}_{\mathbf{n}}(\mathbf{v}, \mathbf{v}^{out}, \mathbf{q}_{\mathbf{n}}, \mathbf{q}_{\mathbf{n}}^{out}) = \frac{1}{2}(\mathbf{q}_{\mathbf{n}} + \mathbf{q}_{\mathbf{n}}^{out}) - \beta(\mathbf{q}_{\mathbf{n}} - \mathbf{q}_{\mathbf{n}}^{out}) - \alpha(\mathbf{v}_j - \mathbf{v}_j^{out}),
\end{aligned}$$

(3.30) *is entropy stable.*

3.7 Back to modal formulation

We have only considered nodal entropy stable DGSEM formulations up to now. In this section, we will recover the corresponding modal formulations in a straightforward manner. Here in order to maintain entropy stability, we have to make sure that the nodal values of $\mathbf{v}$ live in the polynomial space, which brings us the idea of entropy projection.

On the reference element K, suppose that $\mathbf{u}_h(\boldsymbol{\xi}) := \sum_{l=1}^{\mathcal{N}_{Q,k}} \widehat{\mathbf{u}}_l p_l(\boldsymbol{\xi})$ is the numerical solution function, $\overrightarrow{\widehat{\mathbf{u}}}$ is the vector of polynomial coefficients, and $\overrightarrow{\mathbf{u}} = \mathbf{V}\overrightarrow{\widehat{\mathbf{u}}}$ is the vector of nodal values. For entropy variables $\mathbf{v}$, we define the projected polynomial:

$$\overrightarrow{\widehat{\mathbf{v}}} := \mathbf{P}\overrightarrow{\mathbf{v}}, \quad \mathbf{v}_h(\boldsymbol{\xi}) := \sum_{l=1}^{\mathcal{N}_{Q,k}} \widehat{\mathbf{v}}_l p_l(\boldsymbol{\xi}),$$

as well as the entropy-projected values $\overrightarrow{\widetilde{\mathbf{v}}}$ and $\overrightarrow{\widetilde{\mathbf{u}}}$, such that

$$\overrightarrow{\widetilde{\mathbf{v}}} := \mathbf{V}\overrightarrow{\widehat{\mathbf{v}}} = \mathbf{VP}\overrightarrow{\mathbf{v}}, \quad \widetilde{\mathbf{u}}_j = \mathbf{v}(\widetilde{\mathbf{v}}_j), \quad 1 \leq j \leq \mathcal{N}_{Q,k}.$$

Now given the generic nodal DGSEM formulation

$$\frac{d\overrightarrow{\mathbf{u}}}{dt} + \overrightarrow{\mathbf{L}}\left(\overrightarrow{\mathbf{u}}; \left\{\overrightarrow{\mathbf{u}^{e,\text{out}}}\right\}_{e=1}^{\mathcal{N}_E}\right) = 0, \tag{3.31}$$

the modal formulation is derived through projection (recall Section 3.2) and inserting entropy-projected values:

$$\frac{d\overrightarrow{\widehat{\mathbf{u}}}}{dt} + \mathbf{P}\overrightarrow{\mathbf{L}}\left(\overrightarrow{\widetilde{\mathbf{u}}}; \left\{\overrightarrow{\widetilde{\mathbf{u}}^{e,\text{out}}}\right\}_{e=1}^{\mathcal{N}_E}\right) = 0. \tag{3.32}$$

Theorem 3.7. *If* (3.31) *is conservative and entropy stable, then the modal formulation* (3.32) *is also conservative and entropy stable, in the sense that*

$$\frac{d}{dt}\Big(\sum_{i=1}^{N}\int_{K_i}\mathbf{u}_h(\mathbf{x})d\mathbf{x}\Big) = \frac{d}{dt}\Big(\sum_{i=1}^{N}\overrightarrow{\mathbf{1}}^T\mathbf{M}_i\overrightarrow{\mathbf{u}_i}\Big) = 0, \quad \frac{d}{dt}\Big(\sum_{i=1}^{N}\overrightarrow{1}^T M_i\overrightarrow{U_i}\Big) \leq 0. \tag{3.33}$$

Proof. The evolution of nodal values is

$$\frac{d\overrightarrow{\mathbf{u}}}{dt} + \mathbf{V}\mathbf{P}\overrightarrow{\mathbf{L}}\left(\overrightarrow{\widetilde{\mathbf{u}}}; \left\{\overrightarrow{\widetilde{\mathbf{u}}^{e,\text{out}}}\right\}_{e=1}^{\mathcal{N}_E}\right) = 0.$$

Since $\mathbf{MVP} = \mathbf{MV}(\mathbf{V}^T\mathbf{MV})^{-1}\mathbf{V}^T\mathbf{M} = \mathbf{P}^T\mathbf{V}^T\mathbf{M}$,

$$\begin{aligned}\frac{d}{dt}(\overrightarrow{\mathbf{1}}^T\mathbf{M}\overrightarrow{\mathbf{u}}) &= -\overrightarrow{\mathbf{1}}^T\mathbf{MVP}\overrightarrow{\mathbf{L}}\left(\overrightarrow{\widetilde{\mathbf{u}}}; \left\{\overrightarrow{\widetilde{\mathbf{u}}^{e,\text{out}}}\right\}_{e=1}^{\mathcal{N}_E}\right) - (\mathbf{VP}\overrightarrow{\mathbf{1}})^T\mathbf{M}\overrightarrow{\mathbf{L}}\left(\overrightarrow{\widetilde{\mathbf{u}}}; \left\{\overrightarrow{\widetilde{\mathbf{u}}^{e,\text{out}}}\right\}_{e=1}^{\mathcal{N}_E}\right)\\ &= -\overrightarrow{\mathbf{1}}^T\mathbf{M}\overrightarrow{\mathbf{L}}\left(\overrightarrow{\widetilde{\mathbf{u}}}; \left\{\overrightarrow{\widetilde{\mathbf{u}}^{e,\text{out}}}\right\}_{e=1}^{\mathcal{N}_E}\right),\end{aligned}$$

and

$$\begin{aligned}\frac{d}{dt}(\overrightarrow{1}^T M\overrightarrow{U}) &= -\overrightarrow{\mathbf{v}}^T\mathbf{MVP}\overrightarrow{\mathbf{L}}\left(\overrightarrow{\widetilde{\mathbf{u}}}; \left\{\overrightarrow{\widetilde{\mathbf{u}}^{e,\text{out}}}\right\}_{e=1}^{\mathcal{N}_E}\right) = -(\mathbf{VP}\overrightarrow{\mathbf{v}})^T\mathbf{M}\overrightarrow{\mathbf{L}}\left(\overrightarrow{\widetilde{\mathbf{u}}}; \left\{\overrightarrow{\widetilde{\mathbf{u}}^{e,\text{out}}}\right\}_{e=1}^{\mathcal{N}_E}\right)\\ &= -\overrightarrow{\widetilde{\mathbf{v}}}^T\mathbf{M}\overrightarrow{\mathbf{L}}\left(\overrightarrow{\widetilde{\mathbf{u}}}; \left\{\overrightarrow{\widetilde{\mathbf{u}}^{e,\text{out}}}\right\}_{e=1}^{\mathcal{N}_E}\right).\end{aligned}$$

Then from the conservation and entropy stability of (3.31), we see that (3.32) is also conservative and entropy stable. □

Entropy projection will not affect high order accuracy. For a smooth solution $\mathbf{u}$ and $1 \le i \le N$, the projection error $\overrightarrow{\widetilde{\mathbf{v}}}_i - \overrightarrow{\mathbf{v}}_i = \mathcal{O}(h^{k+1})$, and $\overrightarrow{\widetilde{\mathbf{u}}}_i - \overrightarrow{\mathbf{u}}_i = \mathcal{O}(h^{k+1})$. We can easily prove that for both (3.16) and (3.26),

$$\overrightarrow{\mathbf{L}}_i\left(\overrightarrow{\widetilde{\mathbf{u}}}_i; \left\{\overrightarrow{\widetilde{\mathbf{u}}_i^{e,\text{out}}}\right\}_{e=1}^{\mathcal{N}_E}\right) - \overrightarrow{\mathbf{L}}_i\left(\overrightarrow{\mathbf{u}}_i; \left\{\overrightarrow{\mathbf{u}_i^{e,\text{out}}}\right\}_{e=1}^{\mathcal{N}_E}\right) = \mathcal{O}(h^k).$$

Hence the local truncation error of the modal formulation is also $\mathcal{O}(h^k)$. Moreover, by using entropy-projected values, we achieve a better estimate for the local aliasing error in Section 3.5.

Theorem 3.8. *Under the same assumptions as in Theorem 3.4, if we replace* $\overrightarrow{\mathbf{v}}$ *with entropy-projected values* $\overrightarrow{\widetilde{\mathbf{v}}}$, *the local aliasing error*

$$\mathcal{E}_i = \sum_{m=1}^{d} \overrightarrow{\widetilde{\mathbf{v}}}_i^{\,T} \mathbf{S}_{i,m}^T \overrightarrow{\widetilde{\mathbf{f}}_{i,m}} - \left(\overrightarrow{1^b}\right)^T B_i \overrightarrow{\widetilde{\psi}_{i,\mathbf{n}}^b} = \mathcal{O}(h^{2k+d}). \tag{3.34}$$

Proof. Since the extrapolation is exact for polynomials, $\mathbf{v}_{i,s}^b = \mathbf{v}_h(\mathbf{x}_i(\boldsymbol{\xi}_s^b))$. We can consider the Green's formula for $\mathbf{v}_h$:

$$\int_{K_i} \sum_{m=1}^{d} \frac{\partial \mathbf{v}_h^T}{\partial x_m} \mathbf{g}_m(\mathbf{v}_h) d\mathbf{x} = \int_{\partial K_i} \psi_{\mathbf{n}}(\mathbf{v}_h) dS,$$

such that the discretization error only comes from quadrature:

$$\int_{K_i} \sum_{m=1}^{d} \frac{\partial \mathbf{v}_h^T}{\partial x_m} \mathbf{g}_m(\mathbf{v}_h) d\mathbf{x} = \sum_{m=1}^{d} \overrightarrow{\widetilde{\mathbf{v}}}_i^{\,T} \mathbf{S}_{i,m}^T \overrightarrow{\widetilde{\mathbf{f}}_{i,m}} + \mathcal{O}(h^{2k+d}),$$

$$\int_{\partial K_i} \psi_{\mathbf{n}}(\mathbf{v}_h) dS = \left(\overrightarrow{1^b}\right)^T B_i \overrightarrow{\widetilde{\psi}_{i,\mathbf{n}}^b} + \mathcal{O}(h^{2k+d}).$$

Hence $\mathcal{E}_i = \mathcal{O}(h^{2k+d})$. □

3.8 Accuracy test

We test the numerical convergence rates of entropy stable nodal DGSEM (3.16), (3.26) and (3.29) on the two-dimensional Burgers equation. Discrete operators are built on the A-type quadrature points with $t = 1, 2, 3, 4$. The settings will be the same as in Example 2.6.2. All three schemes are evolved up to $t = 0.05$. Here we consider two different entropy functions: the square entropy function $U = \frac{u^2}{2}$ and the hyperbolic entropy function $U = \cosh(u)$.

Numerical errors and orders of convergence of scheme (3.16) are presented in Table 3.1 (for square entropy function) and Table 3.2 (for hyperbolic cosine entropy function). The corresponding numerical results of scheme (3.26) are listed in Table 3.3 and 3.4, and the results of scheme scheme (3.29) are listed in Table 3.5 and 3.6. Since the internal quadrature rule is of degree $2k$ and the boundary quadrature rule is of degree $2k + 1$, it might be possible to recover the optimal $(k + 1)$-th order convergence (recall Remark 1.10). We do achieve optimal convergence in (3.16) and (3.29), despite the fact that the truncation error of (3.29) is not fully understood. However, the convergence is still below optimal in scheme (3.26). The change of entropy function has relatively little impact. For a given scheme, and the convergence orders with two entropy functions are almost the same, and the numerical error with hyperbolic cosine entropy function is slightly smaller.

Table 3.1: Accuracy test of the two-dimensional Burgers equation at $t = 0.05$: scheme (3.16) and square entropy function.

k	h	L^1 error	order	L^2 error	order	L^∞ error	order
2	1/16	2.388e-04	-	4.924e-04	-	6.618e-03	-
	1/32	3.683e-05	2.697	8.852e-05	2.476	1.431e-03	2.210
	1/64	4.821e-06	2.933	1.182e-05	2.905	1.873e-04	2.934
	1/128	6.340e-07	2.927	1.726e-06	2.775	3.270e-05	2.518
	1/256	8.354e-08	2.924	2.483e-07	2.797	6.024e-06	2.440
3	1/16	5.966e-05	-	2.064e-04	-	3.278e-03	-
	1/32	5.006e-06	3.575	2.081e-05	3.310	5.757e-04	2.510
	1/64	3.397e-07	3.881	1.508e-06	3.787	3.352e-05	4.102
	1/128	2.187e-08	3.957	1.027e-07	3.876	3.931e-06	3.092
	1/256	1.459e-09	3.906	6.922e-09	3.891	3.300e-07	3.574
4	1/8	1.617e-04	-	5.139e-04	-	6.596e-03	-
	1/16	1.209e-05	3.741	4.902e-05	3.390	1.231e-03	2.422
	1/32	5.545e-07	4.446	2.877e-06	4.091	1.243e-04	3.308
	1/64	1.741e-08	4.993	8.684e-08	5.050	5.782e-06	4.426
	1/128	5.561e-10	4.969	2.819e-09	4.945	1.766e-07	5.033

Table 3.2: Accuracy test of the two-dimensional Burgers equation at $t = 0.05$: scheme (3.16) and hyperbolic cosine entropy function.

k	h	L^1 error	order	L^2 error	order	L^∞ error	order
2	1/16	2.244e-04	-	4.597e-04	-	6.226e-03	-
	1/32	3.391e-05	2.726	8.188e-05	2.489	1.367e-03	2.187
	1/64	4.386e-06	2.951	1.074e-05	2.931	1.793e-04	2.931
	1/128	5.717e-07	2.939	1.557e-06	2.786	2.992e-05	2.583
	1/256	7.511e-08	2.928	2.238e-07	2.798	5.511e-06	2.440
3	1/16	5.445e-05	-	1.913e-04	-	3.067e-03	-
	1/32	4.526e-06	3.589	1.921e-05	3.316	5.347e-04	2.520
	1/64	3.019e-07	3.906	1.365e-06	3.815	3.096e-05	4.110
	1/128	1.920e-08	3.975	9.184e-08	3.894	3.538e-06	3.130
	1/256	1.275e-09	3.913	6.148e-09	3.901	2.941e-07	3.589
4	1/8	1.476e-04	-	4.757e-04	-	6.202e-03	-
	1/16	1.092e-05	3.757	4.482e-05	3.408	1.132e-03	2.454
	1/32	4.984e-07	4.454	2.637e-06	4.087	1.150e-04	3.300
	1/64	1.528e-08	5.028	7.728e-08	5.093	5.355e-06	4.424
	1/128	4.818e-10	4.987	2.472e-09	4.966	1.641e-07	5.028

Table 3.3: Accuracy test of the two-dimensional Burgers equation at $t = 0.05$: scheme (3.26) and square entropy function.

k	h	L^1 error	order	L^2 error	order	L^∞ error	order
2	1/16	5.589e-04	-	1.417e-03	-	1.480e-02	-
	1/32	7.520e-05	2.894	2.264e-04	2.646	2.942e-03	2.331
	1/64	9.540e-06	2.979	3.179e-05	2.832	4.968e-04	2.566
	1/128	1.159e-06	3.042	4.063e-06	2.968	9.024e-05	2.461
	1/256	1.449e-07	2.999	5.328e-07	2.931	1.510e-05	2.579
3	1/16	8.852e-05	-	2.776e-04	-	4.435e-03	-
	1/32	9.359e-06	3.242	3.815e-05	2.863	9.120e-04	2.282
	1/64	8.005e-07	3.547	3.835e-06	3.314	1.521e-04	2.584
	1/128	6.294e-08	3.669	3.702e-07	3.373	2.543e-05	2.581
	1/256	5.182e-09	3.602	3.689e-08	3.327	3.160e-06	3.008
4	1/8	2.135e-04	-	6.078e-04	-	6.544e-03	-
	1/16	1.920e-05	3.475	7.180e-05	3.082	1.661e-03	1.978
	1/32	1.216e-06	3.981	5.972e-06	3.588	2.215e-04	2.906
	1/64	4.846e-08	4.649	2.767e-07	4.432	1.829e-05	3.598
	1/128	1.670e-09	4.859	9.917e-09	4.802	1.061e-06	4.108

Table 3.4: Accuracy test of the two-dimensional Burgers equation at $t = 0.05$: scheme (3.26) and hyperbolic cosine entropy function.

k	h	L^1 error	order	L^2 error	order	L^∞ error	order
2	1/16	5.300e-04	-	1.367e-03	-	1.451e-02	-
	1/32	7.057e-05	2.909	2.154e-04	2.666	2.813e-03	2.366
	1/64	8.973e-06	2.975	3.028e-05	2.831	4.761e-04	2.563
	1/128	1.089e-06	3.042	3.872e-06	2.967	8.605e-05	2.468
	1/256	1.363e-07	2.998	5.082e-07	2.930	1.447e-05	2.572
3	1/16	8.324e-05	-	2.611e-04	-	4.398e-03	-
	1/32	8.824e-06	3.238	3.621e-05	2.850	8.565e-04	2.361
	1/64	7.546e-07	3.548	3.655e-06	3.308	1.463e-04	2.550
	1/128	5.944e-08	3.666	3.537e-07	3.370	2.444e-05	2.581
	1/256	4.908e-09	3.598	3.530e-08	3.325	3.024e-06	3.015
4	1/8	1.945e-04	-	5.587e-04	-	6.280e-03	-
	1/16	1.768e-05	3.459	6.770e-05	3.045	1.607e-03	1.966
	1/32	1.123e-06	3.977	5.577e-06	3.602	2.089e-04	2.943
	1/64	4.482e-08	4.648	2.583e-07	4.432	1.709e-05	3.612
	1/128	1.549e-09	4.855	9.283e-09	4.798	9.867e-07	4.114

Table 3.5: Accuracy test of the two-dimensional Burgers equation at $t = 0.05$: scheme (3.29) and square entropy function.

k	h	L^1 error	order	L^2 error	order	L^∞ error	order
2	1/16	2.423e-04	-	4.972e-04	-	6.534e-03	-
	1/32	3.695e-05	2.713	8.831e-05	2.493	1.421e-03	2.201
	1/64	4.829e-06	2.936	1.181e-05	2.903	1.865e-04	2.930
	1/128	6.345e-07	2.928	1.725e-06	2.775	3.245e-05	2.523
	1/256	8.357e-08	2.925	2.482e-07	2.797	5.993e-06	2.437
3	1/16	6.369e-05	-	2.222e-04	-	3.553e-03	-
	1/32	5.322e-06	3.581	2.234e-05	3.314	6.373e-04	2.479
	1/64	3.541e-07	3.910	1.587e-06	3.816	3.607e-05	4.143
	1/128	2.251e-08	3.975	1.066e-07	3.896	4.092e-06	3.140
	1/256	1.491e-09	3.916	7.121e-09	3.904	3.434e-07	3.575
4	1/8	1.821e-04	-	5.807e-04	-	7.067e-03	-
	1/16	1.336e-05	3.768	5.422e-05	3.421	1.358e-03	2.379
	1/32	5.991e-07	4.480	3.120e-06	4.119	1.348e-04	3.332
	1/64	1.834e-08	5.030	9.209e-08	5.082	6.217e-06	4.439
	1/128	5.771e-10	4.990	2.945e-09	4.967	1.901e-07	5.032

Table 3.6: Accuracy test of the two-dimensional Burgers equation at $t = 0.05$: scheme (3.29) and hyperbolic cosine entropy function.

k	h	L^1 error	order	L^2 error	order	L^∞ error	order
2	1/16	2.321e-04	-	4.779e-04	-	6.456e-03	-
	1/32	3.456e-05	2.748	8.407e-05	2.507	1.431e-03	2.173
	1/64	4.432e-06	2.963	1.088e-05	2.950	1.888e-04	2.923
	1/128	5.748e-07	2.947	1.566e-06	2.797	3.105e-05	2.604
	1/256	7.532e-08	2.932	2.244e-07	2.803	5.511e-06	2.494
3	1/16	6.136e-05	-	2.201e-04	-	3.608e-03	-
	1/32	5.050e-06	3.603	2.199e-05	3.323	6.367e-04	2.503
	1/64	3.266e-07	3.951	1.513e-06	3.861	3.610e-05	4.141
	1/128	2.031e-08	4.007	9.928e-08	3.929	3.837e-06	3.234
	1/256	1.330e-09	3.933	6.533e-09	3.926	3.184e-07	3.591
4	1/8	1.787e-04	-	5.778e-04	-	7.072e-03	-
	1/16	1.298e-05	3.783	5.361e-05	3.430	1.356e-03	2.382
	1/32	5.751e-07	4.497	3.081e-06	4.121	1.346e-04	3.334
	1/64	1.693e-08	5.086	8.712e-08	5.144	6.191e-06	4.442
	1/128	5.198e-10	5.026	2.711e-09	5.006	1.902e-07	5.024

Part II

Polynomial Chaos Expansion Method for Distribution-free SPDEs

Chapter Four

Distribution-free Stochastic Analysis

This chapter is a brief tutorial of the distribution-free stochastic analysis in [78]. We first describe the generalized polynomial chaos (gPC) expansion for an uncorrelated sequence of random variables in Section 4.1. Then we bring into the ideas of driving noise, Wick product and Skorokhod integral in Section 4.2 and 4.3. These are basically generalizations of Gaussian white noise and Itô integral.

4.1 Generalized polynomial chaos expansion

Suppose that $\Xi = \{\xi_k\}_{k=1}^{\infty}$ is a sequence of uncorrelated random variables within some probability space $(\Omega, \mathcal{F}, \mathbb{P})$, such that $\mathbb{E}[\xi_k] = 0$ and $\mathbb{E}[\xi_k^2] = 1$ for each k. We also assume $\mathcal{F} = \sigma(\Xi)$ is the σ-algebra generated by $\{\xi_k\}_{k=1}^{\infty}$.The term *distribution-free* arises from the fact that each random variable can be of any distribution. They are not required to be identically distributed or independent. We aim to construct an orthogonal basis of $L^2(\Omega)$, under the notation of multi-indices.

Definition 4.1. *Let $\alpha = (\alpha_1, \alpha_2, \cdots)$ be an multi-index whose length is denoted by $|\alpha| := \sum_{k=1}^{\infty} \alpha_k$. $\mathcal{J}$ stands for the set of multi-indices of finite length:*

$$\mathcal{J} = \{\alpha = (\alpha_1, \alpha_2, \cdots) : \ \ \alpha_k \geq 0 \textit{ for each } k, \ \ |\alpha| < \infty\}.$$

The polynomials and factorials of multi-indices are defined as:

$$\xi^{\alpha} := \prod_{k=1}^{\infty} \xi_k^{\alpha_k}, \quad \alpha! := \prod_{k=1}^{\infty} \alpha_k!.$$

Moreover, ε_0 is the multi-index whose entries are all zero, and ε_k is the multi-index such that its k-th entry is 1 and all other entries are zero, for each $k \geq 1$.

Each multi-index α with $|\alpha| = n$ can be uniquely identified by its characteristic set $I_\alpha = (i_\alpha^1, i_\alpha^2, \cdots, i_\alpha^n)$, which is a vector of length n and given by

$$i_\alpha^m = k \text{ if and only if } \sum_{j=1}^{k-1} \alpha_j < m \leq \sum_{j=1}^{k} \alpha_j, \text{ for each } 1 \leq m \leq n.$$

For instance, if $\alpha = (0, 1, 0, 2, 3, 0, \cdots)$, $I_\alpha = (2, 4, 4, 5, 5, 5)$. Particularly, we let $d(\alpha) := i_\alpha^n$, the position of the rightmost nonzero entry in α. We impose the following two assumptions on Ξ.

A1. For each finite dimensional random vector $(\xi_{i_1}, \xi_{i_2}, \cdots, \xi_{i_d})$, the moment generating function $\mathbb{E}[\exp(\theta_1 \xi_{i_1} + \theta_2 \xi_{i_2} + \cdots + \theta_d \xi_{i_d})]$ exists for all $(\theta_1, \theta_2, \cdots, \theta_d)$ in some neighborhood of $0 \in \mathbb{R}^d$.

A2. We have an orthogonal set of polynomials $\{K_\alpha, \alpha \in \mathcal{J}\}$ such that for each $n \geq 1$,

$$\text{span}\{K_\beta, |\beta| \leq n\} = \text{span}\{\xi^\beta : |\beta| \leq n\} := \mathcal{P}^n,$$

and for each $|\alpha| = n + 1$,

$$K_\alpha = \xi^\alpha - \text{projection}_{\mathcal{P}^n} \xi^\alpha.$$

The generalized polynomial chaos basis functions $\{\Phi_\alpha, |\alpha| \in \mathcal{J}\}$ are scaled versions of $\{K_\alpha, |\alpha| \in \mathcal{J}\}$:

$$\Phi_\alpha := c_\alpha K_\alpha, \text{ so that } \mathbb{E}[\Phi_\alpha \Phi_\beta] = \delta_{\alpha\beta}(\alpha!) \tag{4.1}$$

Obviously $\Phi_{\varepsilon_0} = 1$ and $\Phi_{\varepsilon_k} = \xi_k$. Under assumptions **A1** and **A2**, $\{\Phi_\alpha, \alpha \in \mathcal{J}\}$ (and hence $\{K_\alpha, \alpha \in \mathcal{J}\}$) forms a complete Cameron-Martin [6] type orthogonal basis.

The following theorem is proved in [78].

Theorem 4.1. *Assume* ***A1*** *and* ***A2*** *hold. Then $\{\Phi_\alpha, \alpha \in \mathcal{J}\}$ is a complete set of orthogonal basis functions of $L^2(\Omega)$. For each $\eta \in L^2(\Omega)$, its polynomial chaos expansion is*

$$\eta = \sum_{\alpha \in \mathcal{J}} \eta_\alpha \Phi_\alpha, \quad \eta_\alpha = \frac{\mathbb{E}[\eta \Phi_\alpha]}{\alpha!},$$

and the Parseval's identity holds:

$$\mathbb{E}[\eta^2] = \sum_{\alpha \in \mathcal{J}} (\alpha!) \eta_\alpha^2.$$

In this way we separate the stochastic part (Φ_α) and the deterministic part (η_α).

In the special case where $\{\xi_k\}_{k=1}^\infty$ are independent identically distributed (i.i.d.) random variables, **A1** and **A2** can be simplified into the two assumptions below.

B1. The moment generating function $\mathbb{E}[\exp(\theta \xi_k)]$ exists for all θ in some neighborhood of 0.

B2. There exists an orthogonal set of univariate polynomials $\{\varphi_n(\xi)\}_{n=0}^\infty$ such that $\mathbb{E}[\varphi_n(\xi_k)\varphi_m(\xi_k)] = \delta_{mn} n!$, and the gPC basis functions are simply tensor products of $\{\varphi_n(\xi)\}_{n=0}^\infty$:

$$\Phi_\alpha = \prod_{k=1}^\infty \varphi_{\alpha_k}(\xi_k). \tag{4.2}$$

Corollary 4.1. *Suppose that $\{\xi_k\}_{k=1}^\infty$ is a sequence of i.i.d random variables with zero mean and unit variance, and assumptions B1 and B2 hold. Then $\{\Phi_\alpha, \alpha \in \mathcal{J}\}$ given by (4.2) is a complete set of orthogonal basis functions of $L^2(\Omega)$.*

Table 4.1 shows the orthogonal polynomials for some common random distributions (see e.g. [106]). It may be necessary to shift and scale the distribution to achieve zero mean and unit variance.

Table 4.1: Correspondence between random distribution and orthogonal polynomials for an i.i.d. sequence of random variables.

	Distribution of ξ_k	Orthogonal polynomials
Continuous	Gaussian	Hermite
	Gamma	Lagurre
	Beta	Jacobi
	Uniform	Legendre
Discrete	Poisson	Charlier
	Binomial	Krawtchouk
	Negative binomial	Meixner
	Hypergeometric	Hahn

Remark 4.1. We use the weaker assumption of uncorrelated random variables to incorporate Lévy randomness, whose gPC basis functions are not polynomials of simple random variables. We will always consider i.i.d. random variables in the numerical tests in Section 5.3.

4.2 Driving noise

Now we take the time variable into account. Let $[0, T]$ be some time interval and $H := L^2([0, T])$ .We define the following driving noise $\dot{\mathcal{N}}(t)$:

$$\dot{\mathcal{N}}(t) = \sum_{k=1}^{\infty} m_k(t)\xi_k = \sum_{k=1}^{\infty} m_k(t)\Phi_{\varepsilon_k}, \tag{4.3}$$

and the stochastic process

$$\mathcal{N}(t) = \int_0^t \dot{\mathcal{N}}(s)ds = \sum_{k=1}^{\infty} \Big(\int_0^t m_k(s)ds\Big)\xi_k, \tag{4.4}$$

where $\{m_k(t)\}_{k=1}^{\infty}$ is a complete orthonormal basis of H. Then $\mathcal{N}(t)$ is a process with zero mean, and the covariance function

$$\mathbb{E}[\mathcal{N}(t_1)\mathcal{N}(t_2)] = \sum_{k=1}^{\infty}\Big(\int_0^{t_1} m_k(s)ds\Big)\Big(\int_0^{t_2} m_k(s)ds\Big) = \sum_{k=1}^{\infty}(1_{[0,t_1]}, m_k)_H(1_{[0,t_2]}, m_k)_H$$
$$= (1_{[0,t_1]}, 1_{[0,t_2]})_H = \min\{t_1, t_2\}.$$

In other words, $\mathcal{N}(t)$ has uncorrelated increments, such that for $0 \leq t_1 \leq t_2$,

$$\mathbb{E}[(\mathcal{N}(t_2) - \mathcal{N}(t_1))\mathcal{N}(t_1)] = \mathbb{E}[\mathcal{N}(t_2)\mathcal{N}(t_1) - \mathcal{N}(t_1)^2] = t_1 - t_1 = 0.$$

Two specific examples of $\dot{\mathcal{N}}(t)$ and $\mathcal{N}(t)$ are provided below.

Example 4.2.1. If $\{\xi_k\}_{k=1}^{\infty}$ are i.i.d. standard Gaussian random variables, $\mathcal{N}(t)$ is a Gaussian process with independent increments (zero correlation implies independence for jointly Gaussian random variables). One can easily show that

$$\mathbb{E}[(\mathcal{N}(t_2) - \mathcal{N}(t_1))^4] = 3(t_2 - t_1)^2$$

Then by Kolmogorov continuity theorem, we can find a version of $\mathcal{N}(t)$ with continuous path. This is indeed standard Wiener process $W(t)$, and the driving noise $\dot{\mathcal{N}}(t)$ is the Gaussian white noise $\dot{W}(t)$. As for gPC basis functions, $\{\varphi_n(\xi)\}_{n=0}^{\infty}$ are probabilists' Hermite polynomials $\{He_n(x)\}_{n=0}^{\infty}$.

Example 4.2.2. If $\{\xi_k\}_{k=1}^{\infty}$ are i.i.d. uniformly distributed on $[-\sqrt{3}, \sqrt{3}]$, the driving stochastic process $\mathcal{N}(t)$ is non-Gaussian as its characteristic function is

$$\mathbb{E}[i\theta\mathcal{N}(t)] = \mathbb{E}\Big[\exp\Big(i\theta\sum_{k=1}^{\infty}\Big(\int_0^t m_k(s)ds\Big)\xi_k\Big)\Big] = \prod_{k=1}^{\infty}\frac{\sin\Big(\sqrt{3}\theta\Big(\int_0^t m_k(s)ds\Big)\Big)}{\sqrt{3}\theta\Big(\int_0^t m_k(s)ds\Big)}.$$

As for gPC basis functions, $\{\varphi_n(\xi)\}_{n=0}^{\infty}$ are the scaled versions of Legendre polynomials $\{L_n(x)\}_{n=0}^{\infty}$:

$$\varphi_n(\xi) = \sqrt{(2n+1)n!}L_n\Big(\frac{\xi}{\sqrt{3}}\Big), \tag{4.5}$$

such that $\mathbb{E}[\varphi_n(\xi_k)^2] = n!$.

4.3 Wick product and Skorokhod integral

In this section briefly explains the languages to construct distribution-free stochastic integrals. For the opposite direction, i.e. the stochastic (Malliavin) derivative, we refer interested readers to [78] for more details. Let E be some given Hilbert space. We will work on the space of generalized random variables written as formal chaos expansion series:

$$\mathcal{D}'(E) := \Big\{u = \sum_{\alpha\in\mathcal{J}} u_\alpha\Phi_\alpha : u_\alpha \in E\Big\},$$

and the space of square integrable general random variables:

$$\mathcal{D}(E) := \Big\{u = \sum_{\alpha\in\mathcal{J}} u_\alpha\Phi_\alpha : u_\alpha \in E,\ \ \mathbb{E}[\|u\|_E^2] = \sum_{\alpha\in\mathcal{J}} \alpha!\|u_\alpha\|_E^2 < \infty\Big\},$$

For instance, if $E = H = L^2([0,T])$, $\mathcal{D}'(H)$ consists of generalized stochastic processes $u = u(t) = \sum\limits_{\alpha\in\mathcal{J}} u_\alpha(t)\Phi_\alpha$ such that each $u_\alpha \in L^2([0,T])$; while $\mathcal{D}(H) = L^2([0,T]\times\Omega)$, the subspace of square integrable stochastic processes in $\mathcal{D}'(H)$.

We first introduce Wick product $\diamond$, a convolution type binary operator on chaos expansion coefficients:

$$\Phi_\alpha \diamond \Phi_\beta = \Phi_{\alpha+\beta}, \quad u \diamond v = \sum_{\alpha\in\mathcal{J}}\sum_{\beta\in\mathcal{J}} u_\alpha v_\beta \Phi_{\alpha+\beta} \text{ for } u, v \in \mathcal{D}'(\mathbb{R}).$$

Then for $u = u(t) \in \mathcal{D}'(H)$, its Skorokhod integral $\delta(u) \in \mathcal{D}'(\mathbb{R})$ is denoted by

$$\delta(u) := \int_0^T u(t) \diamond \dot{\mathcal{N}}(t)dt = \sum_{\alpha \in \mathcal{J}} \sum_{k=1}^{\infty} (u_\alpha, m_k)_H \Phi_{\alpha+\varepsilon_k} = \sum_{\alpha \in \mathcal{J}} \Big(\sum_{\varepsilon_k \leq \alpha} (u_{\alpha-\varepsilon_k}, m_k)_H \Big) \Phi_\alpha. \tag{4.6}$$

The Skorokhod integral can be better characterized in terms of multiple integrals. For $n \geq 0$, let $H^n := L^2([0,T]^n)$ and $\widetilde{H^n}$ be the family of symmetric functions in H^n. We use $t^{(n)}$ as the short hand notation of $(t_1, t_2, \cdots, t_n)$. For each multi-index α with $|\alpha| = n$, we set

$$E_\alpha(t^{(n)}) = \sum_{\sigma \in \mathcal{G}_n} m_{i_\alpha^1}(t_{\sigma(1)}) m_{i_\alpha^2}(t_{\sigma(2)}) \cdots m_{i_\alpha^n}(t_{\sigma(n)}), \tag{4.7}$$

where $\mathcal{G}_n$ is the permutation group on $\{1, 2, \cdots, n\}$. Then $\{E_\alpha, |\alpha| = n\}$ is a complete orthogonal basis of $\widetilde{H^n}$, and $\|E_\alpha\|_{H^n}^2 = n!\alpha!$. For each $f \in \widetilde{H^n}$, we have the expansion and the Parseval's identity

$$f = \sum_{|\alpha|=n} f_\alpha E_\alpha, \quad f_\alpha = \frac{(f, E_\alpha)_{H_n}}{n!\alpha!}, \quad \|f\|_{H_n}^2 = \sum_{|\alpha|=n} n!\alpha! f_\alpha^2.$$

The multiple integral I_n is a linear operator from $\widetilde{H^n}$ to $L^2(\Omega)$, such that

$$I_n(f) := n! \sum_{|\alpha|=n} f_\alpha \Phi_\alpha, \quad f \in \widetilde{H^n} \tag{4.8}$$

for each $f \in \widetilde{H^n}$, and

$$\mathbb{E}[I_n(f)^2] = \sum_{|\alpha|=n} \alpha!(n! f_\alpha)^2 = n! \|f\|_{H^n}^2$$

Therefore, I_n (to be more rigorous, $I_n/\sqrt{n!}$) defines an isometric embedding. The connection between Skorokhod integral and multiple integral is pointed out in the following theorem.

Theorem 4.2. *Suppose that $u = u(t) = I_n(f(t, t^{(n)})) \in L^2([0,T] \times \Omega)$, and $f(t,\cdot) \in \widetilde{H^n}$ for each $t \in [0,T]$. Then $\delta(u) = I_{n+1}(\widetilde{f}) \in L^2(\Omega)$. Here for $g \in H^n$, $\widetilde{g}$ is standard symmetrization of g:*

$$\widetilde{g}(t^{(n)}) := \frac{1}{n!} \sum_{\sigma \in \mathcal{G}_n} g(t_{\sigma(1)}, t_{\sigma(2)}, \cdots, t_{\sigma(n)}). \tag{4.9}$$

Proof. We denote the expansion of $f(t,\cdot)$ by:

$$f(t, t^{(n)}) = \sum_{|\alpha|=n} f_\alpha(t) E_\alpha(t^{(n)}).$$

By the definition of I_n and δ,

$$u(t) = n! \sum_{|\alpha|=n} f_\alpha(t) \Phi_\alpha, \quad \delta(u) = n! \sum_{|\alpha|=n} \sum_{k=1}^{\infty} (f_\alpha, m_k)_H \Phi_{\alpha+\varepsilon_k}.$$

Since $u \in L^2([0,T] \times \Omega)$,

$$\|f\|^2_{H^{n+1}} = \int_0^T \|f(t,\cdot)\|^2_{H^n} = \frac{1}{n!} \int_0^T \mathbb{E}[u(t)^2] dt < \infty.$$

Hence $f \in H^{n+1}$, and $\widetilde{f} \in \widetilde{H^{n+1}}$, with expansion:

$$\widetilde{f} = \sum_{|\alpha|=n} \widetilde{f_\alpha E_\alpha} = \sum_{|\alpha|=n} \sum_{k=1}^{\infty} (f_\alpha, m_k)_H \widetilde{m_k E_\alpha} = \frac{1}{n+1} \sum_{|\alpha|=n} \sum_{k=1}^{\infty} (f_\alpha, m_k)_H E_{\alpha+\varepsilon_k}.$$

Comparing the expansions of $\delta(u)$ and $\widetilde{f}$, we draw the conclusion that $\delta(u) = I_{n+1}(\widetilde{f}) \in L^2(\Omega)$. □

As a special case, if $n = 0$ and $f = f(t) \in H$, $I_0(f) = f$ is deterministic, and the

Skorokohd integral of f is:

$$\delta(f) = \sum_{k=1}^{\infty}(f, m_k)_H \xi_k = \sum_{k=1}^{\infty}(f, E_{\varepsilon_k})_H \Phi_{\varepsilon_k} = I_1(f) = I_1(\widetilde{f}). \tag{4.10}$$

In other words, $\delta(f) = I_1(f)$ is an isometric embedding from H to $L^2(\Omega)$. If we also assume that f is continuous, $\delta(f)$ is limit of discrete sums in Itô's sense. Hence Skorokhod can be regarded as the generalization of classic Itô integral.

Theorem 4.3. *Suppose $f = f(t) \in C([0,T])$. Consider the partition of $[0,T]$:*

$$\Delta = \{[t_{i-1}, t_i] : 1 \leq i \leq N_\Delta, \;\; t_0 = 0, \;\; t_{N_\Delta} = T\}, \quad |\Delta| := \min_{1 \leq i \leq N_\Delta}(t_i - t_{i-1}).$$

As $\|\Delta\| \to= 0$, the Itô type discrete sum $\sum\limits_{i=1}^{N_\Delta} f(t_{i-1})(\mathcal{N}(t_i) - \mathcal{N}_{i-1})$ converges to $\delta(f)$ in $L^2(\Omega)$.

Proof. Since

$$\mathcal{N}(t) = \sum_{k=1}^{\infty}\Big(\int_0^t m_k(s)ds\Big)\xi_k = \sum_{k=1}^{\infty}(1_{[0,t]}, m_k)_H \xi_k = \delta(1_{[0,t]}),$$

the discrete sum equals

$$\sum_{i=1}^{N_\Delta} f(t_{i-1})(\mathcal{N}(t_i) - \mathcal{N}_{i-1}) = \delta(f_\Delta), \quad f_\Delta(t) := \sum_{i=1}^{N_\Delta} f(t_{i-1}) 1_{[t_{i-1}, t_i)}(t).$$

By uniform continuity of f and isometric property of δ, as $|\Delta| \to 0$, f_Δ converges uniformly to f (which implies convergence in H), and thus $\delta(f_\Delta)$ converges to $\delta(f)$ in $L^2(\Omega)$. □

Furthermore, in the case that $\{\xi_k\}_{k=1}^{\infty}$ are i.i.d. standard Gaussian variables, we have seen in Example 4.2.1 that $\dot{\mathcal{N}}(t) = \dot{W}(t)$ is the Gaussian white noise, and we will demonstrate that Skorokhod integral is equivalent to Itô integral for adapted processes. Let us first provide the definition of adapted processes.

Definition 4.2. *For $u = u(t) \in L^2([0,T]\times\Omega)$, we can write down its chaos expansion with regard to multiple integrals:*

$$u(t) = \sum_{n=0}^{\infty} I_n(f_n(t,t^{(n)})), \quad f_n(t,t^{(n)}) := \frac{1}{n!}\sum_{|\alpha|=n} u_\alpha(t)E_\alpha(t^{(n)}).$$

u is called adapted if

$$\textit{supp } f_n(t,\cdot) \in [0,t]^n, \quad \textit{for each } t \in [0,T].$$

Theorem 4.4. *Suppose that $u = u(t) \in L^2([0,T]\times\Omega)$ is adapted. Then $\delta(u) \in L^2(\Omega)$, and we have the Itô-Skorokhod isometry:*

$$\mathbb{E}[\delta(u)^2] = \int_0^T \mathbb{E}[u(t)^2]dt. \tag{4.11}$$

If we further assume that $\{\xi_k\}_{k=1}^{\infty}$ are i.i.d. standard Gaussian variables, $\delta(u)$ coincides with the Itô integral:

$$\delta(u) = \int_0^T u(t)\diamond \dot{W}(t)dt = \int_0^T u(t)dW(t). \tag{4.12}$$

Proof. By Theorem 4.2, for $u = u(t) = \sum\limits_{n=0}^{\infty} f_n(t,t^{(n)})$,

$$\delta(u) = \sum_{n=0}^{\infty} I_{n+1}(\widetilde{f}_n), \quad \mathbb{E}[\delta(u)^2] = \sum_{n=0}^{\infty}(n+1)!\|\widetilde{f}_n\|^2_{H^{n+1}}.$$

Since $f_n(t,t^{(n)})$ is symmetric with respect to $t^{(n)}$, $\widetilde{f}_n = \frac{1}{n+1}\sum\limits_{i=0}^{n} f_{n,i}(t,t^{(n)})$, where

$$f_{n,0}(t,t^{(n)}) = f_n(t,t^{(n)}), \quad f_{n,i}(t,t^{(n)}) := f_n(t_i,t_1,\cdots,t_{i-1},t,t_{i+1},\cdots,t_n) \text{ for } 1 \le i \le n.$$

From the definition of adaptedness, $\{f_{n,i}\}_{i=0}^n$ have disjoint supports, which gives us

$$\|\widetilde{f}_n\|^2_{H^{n+1}} = \frac{1}{(n+1)^2}\sum_{i=0}^n \|f_{n,i}\|^2_{H^{n+1}} = \frac{1}{n+1}\|f_n\|^2_{H^{n+1}}.$$

Then we are able to prove the Itô-Skorokhod isometry:

$$\mathbb{E}[\delta(u)^2] = \sum_{n=0}^\infty n!\|f_n\|^2_{H^{n+1}} = \int_0^T \Big(\sum_{n=0}^\infty n!\|f_n(t,\cdot)\|^2_{H^n}\Big)dt = \int_0^T \mathbb{E}[u(t)^2]dt.$$

If $\{\xi_k\}_{k=1}^\infty$ are i.i.d. standard Gaussian variables, for each $|\alpha| = n$, we have the multiple Itô integral formula (see [25]):

$$n!\int_0^T\int_0^{t_n}\cdots\int_0^{t_2} E_\alpha(t^{(n)})dW(t_1)\cdots dW(t_{n-1})dW(t_n) = n!\Phi_\alpha = I_n(E_\alpha). \quad (4.13)$$

Hence

$$u(t) = \sum_{n=0}^\infty I_n(f_n(t,t^{(n)})) = \sum_{n=0}^\infty n!\int_0^T\int_0^{t_n}\cdots\int_0^{t_2} f_n(t,t^{(n)})dW(t_1)\cdots dW(t_{n-1})dW(t_n),$$

and

$$\begin{aligned}
&\int_0^T u(t)dW(t)\\
=&\sum_{n=0}^\infty n!\int_0^T \Big(\int_0^T\int_0^{t_n}\cdots\int_0^{t_2} f_n(t,t^{(n)})dW(t_1)\cdots dW(t_{n-1})dW(t_n)\Big)dW(t)\\
=&\sum_{n=0}^\infty n!\int_0^T\int_0^t\int_0^{t_n}\cdots\int_0^{t_2} f_n(t,t^{(n)})dW(t_1)\cdots dW(t_{n-1})dW(t_n)dW(t)\\
=&\sum_{n=0}^\infty (n+1)!\int_0^T\int_0^t\int_0^{t_n}\cdots\int_0^{t_2} \widetilde{f}_n(t,t^{(n)})dW(t_1)\cdots dW(t_{n-1})dW(t_n)dW(t)\\
=&\sum_{n=0}^\infty I_{n+1}(\widetilde{f}_n) = \delta(u).
\end{aligned}$$

The last two equalities follow from adaptedness. We use the fact that if $f_n(t,t^{(n)})$ is nonzero, then $t \geq t_i$ for each $1 \leq i \leq n$, and $\widetilde{f}_n(t,t^{(n)}) = \frac{1}{n+1}f_n(t,t^{(n)})$. □

Chapter Five

Error Estimate and Numerical Results

In this chapter, we will look into the polynomials chaos expansion approach to distribution-free SPDEs. Let $\Gamma \in \mathbb{R}^d$ denote some smooth finite domain. The stochastic solution function $u(t,\mathbf{x})$ lives in the space $L^2([0,T]\times\Gamma\times\Omega)$, represented as gPC expansion

$$u(t,\mathbf{x}) = \sum_{\alpha\in\mathcal{J}} u_\alpha(t,\mathbf{x})\Phi_\alpha, \quad u_\alpha \in L^2([0,T]\times\Gamma).$$

Then we are left with the propagator system, i.e. the system of deterministic PDEs satisfied by $\{u_\alpha(t,\mathbf{x}), \alpha\in\mathcal{J}\}$. For linear SPDEs, the propagator system has lower triangular and sparse dependency [75, 68, 76, 69], and is independent of the type of noise involved. However, for nonlinear problems, the propagator system is fully coupled [56, 73], and varies from one kind of noise to another. The Wick-Malliavin approximation was proposed in [77] as a decoupling technique. Numerical simulations of Wick-Malliavin approximation can be found in [102, 100].

In practice, a finite truncation of $\{u_\alpha(t,\mathbf{x}), \alpha\in\mathcal{J}\}$ is always necessary. For $K, N \geq 0$, define the truncated multi-index set

$$\mathcal{J}_{N,K} := \{\alpha\in\mathcal{J} : |\alpha|\leq N,\ \ d(\alpha)\leq K\}, \quad \#(\mathcal{J}_{N,K}) = \binom{N+K}{N}.$$

That is, $\mathcal{J}_{N,K}$ contains multi-indices whose polynomial order is no more than N, and number of random variables is no more than K. The size of $\mathcal{J}_{N,K}$ grows exponentially with respect to both N and K. Then we compute the truncated solution

$$u_{N,K}(t,\mathbf{x}) := \sum_{\alpha\in\mathcal{J}_{N,K}} \widehat{u}_\alpha(t,\mathbf{x})\Phi_\alpha,$$

where $\{\widehat{u}_\alpha, \alpha\in\mathcal{J}_{N,K}\}$ satisfies some truncated propagator system. Notice that due to aliasing error, $\widehat{u}_\alpha$ might not be the same as u_α.

This chapter consists of the following sections. In Section 5.1, we derive the propagator systems for two model problems. We will take a linear parabolic SPDE as the linear model problem, and stochastic Burgers equation as the nonlinear model problem. In Section 5.2, we analyze the approximation error induced by the truncation of index set, proving that for linear problems, the convergence rate of mean square error is actually exponential with respect to N, and cubic with respect to K. In Section 5.3, we carry out numerical experiments on linear and nonlinear SPDEs. We will study numerical orders of convergence, and the error estimate in Section 5.2 is verified.

5.1 SPDE model problems

Example 5.1.1. Consider the following homogeneous linear parabolic SPDE:

$$\begin{aligned} &\frac{\partial u}{\partial t}(t,\mathbf{x}) = \mathcal{L}u + \mathcal{M}u \diamond \dot{\mathcal{N}}(t), \quad (t,\mathbf{x}) \in (0,T] \times \Gamma, \\ &u(0,\mathbf{x}) = u_0(\mathbf{x}), \quad x \in \Gamma, \end{aligned} \tag{5.1}$$

where

$$\mathcal{L} = \sum_{i=1}^{d}\sum_{j=1}^{d} a_{ij}(\mathbf{x})\frac{\partial^2}{\partial x_j \partial x_j} + \sum_{i=1}^{d} b_i(\mathbf{x})\frac{\partial}{\partial x_i} + c(\mathbf{x}), \quad \mathcal{M} = \sum_{i=1}^{d} \alpha_i(\mathbf{x})\frac{\partial}{\partial x_i} + \beta(\mathbf{x}). \tag{5.2}$$

If $\{\xi_k\}_{k=1}^{\infty}$ are i.i.d. standard Gaussian variables, according to Theorem 4.4, (5.1) is equivalent to the Itô type SPDE

$$du(t,\mathbf{x}) = \mathcal{L}udt + \mathcal{M}udW(t), \tag{5.3}$$

and the Stratonovich type SPDE

$$du(t,\mathbf{x}) = \widetilde{\mathcal{L}}udt + \mathcal{M}u \circ dW(t), \tag{5.4}$$

where $\widetilde{\mathcal{L}} = \mathcal{L} - \frac{1}{2}\mathcal{M}\mathcal{M}$. We assume that the coefficients in $\mathcal{L}$ and $\mathcal{M}$ are smooth and bounded, $\widetilde{\mathcal{L}}$ is uniformly elliptic, and the initial condition $u_0(\mathbf{x})$ is deterministic and bounded. These assumptions are sufficient for a unique square integrable solution $u \in L^2([0,T] \times \Gamma \times \Omega)$ (see [76, 70]). As we will see later, the propagator system is independent of the type of noise. Therefore these well-posedness requirements remain the same in the distribution-free setting.

Recall the definition of the Skorokhod integral (4.6). We come up with the propagator system by comparing the expansion coefficients on both sides of (5.1):

$$\frac{\partial u_\alpha}{\partial t}(t,\mathbf{x}) = \mathcal{L}u_\alpha + \sum_{\varepsilon_k \le \alpha} \mathcal{M}u_{\alpha-\varepsilon_k} m_k(t), \quad u_\alpha(0,\mathbf{x}) = u_0(\mathbf{x})1_{\{\alpha=\varepsilon_0\}}. \tag{5.5}$$

It is a system of linear parabolic deterministic PDEs, with a lower-triangular and sparse structure, i.e., a multi-index of order n only talks to itself and multi-indices of order $n-1$. As a result, the system is not affected by the truncation of multi-index set, and the truncated solution is

$$u_{N,K}(t,\mathbf{x}) = \sum_{\alpha \in \mathcal{J}_{N,K}} u_\alpha(t,\mathbf{x})\Phi_\alpha.$$

Parallelization is also possible as coefficients of the same order can be updated simultaneously. Moreover, the propagator system does not depend on the type of randomness involved. It is solved once and for all, the computational overhead from changes of noise is almost negligible.

From numerical perspective, we follow the method of lines principle to discretize

(5.1) and (5.5). After some suitable spatial discretization with M degrees of freedom, $\mathcal{L}$ and $\mathcal{M}$ turn into $M \times M$ difference matrices A and B, and (5.1) reduces to a linear system of SODEs

$$\begin{aligned} &\mathbf{u}'(t) = A\mathbf{u}(t) + B\mathbf{u}(t) \diamond \dot{\mathcal{N}}(t), \quad t \in (0, T] \\ &\mathbf{u}(0) = \mathbf{u}_0, \end{aligned} \tag{5.6}$$

where $\mathbf{u}(t) \in \mathbb{R}^M$ is the vector $u(t, \mathbf{x})$ evaluated at those degrees of freedom. The corresponding propagator system is

$$\mathbf{u}'_\alpha(t) = A\mathbf{u}_\alpha(t) + \sum_{\varepsilon_k \leq \alpha} m_k(t) B\mathbf{u}_{\alpha - \varepsilon_k}(t), \quad \mathbf{u}_\alpha(0) = \mathbf{u_0} 1_{\{\alpha = \varepsilon_0\}}, \tag{5.7}$$

and the truncated solution of (5.7) is

$$\mathbf{u}_{N,K}(t) = \sum_{\alpha \in \mathcal{J}_{N,K}} \mathbf{u}_\alpha(t) \Phi_\alpha.$$

Example 5.1.2. Consider the one-dimensional stochastic Burgers equation with additive noise and periodic boundary condition [56]:

$$\begin{aligned} &\frac{\partial u}{\partial t}(t, x) + \frac{1}{2}\frac{\partial u^2}{\partial x} = \mu \frac{\partial^2 u}{\partial x^2} + \sigma(x)\dot{\mathcal{N}}(t), \quad (t, x) \in (0, T] \times \Gamma, \\ &u(0, x) = u_0(x), \quad x \in \Gamma, \end{aligned} \tag{5.8}$$

where μ is a positive constant and $\sigma(x)$ is a periodic forcing function. The corresponding Itô type SPDE in the case of i.i.d standard Gaussian noise is

$$du(t, x) = \Big(\mu \frac{\partial^2 u}{\partial x^2} - \frac{1}{2}\frac{\partial u^2}{\partial x}\Big) dt + \sigma(x) dW(t). \tag{5.9}$$

By assuming that $u_0(x)$ is deterministic and $\sigma, u_0 \in L^2(\Gamma)$, we make sure that (5.8) has a unique square integrable solution (see [22]). However, such result does not generalize to the distribution-free setting as the propagator system varies for different driving noises.

In order to figure out the propagator system, we have to expand u^2 into gPC series:

$$u^2 = \sum_{\alpha \in \mathcal{J}} \Big(\sum_{\beta \in \mathcal{J}} \sum_{p \in \mathcal{J}} B(\alpha, \beta, p) u_\beta u_p \Big) \Phi_\alpha, \tag{5.10}$$

where

$$B(\alpha, \beta, p) = \frac{\mathbb{E}[\Phi_\alpha \Phi_\beta \Phi_p]}{\mathbb{E}[\Phi_\alpha^2]} = \frac{\mathbb{E}[\Phi_\alpha \Phi_\beta \Phi_p]}{\alpha!} \tag{5.11}$$

are interaction coefficients. Hence the propagator equations are

$$\begin{aligned} &\frac{\partial u_\alpha}{\partial t}(t,x) + \frac{1}{2} \sum_{\beta \in \mathcal{J}} \sum_{p \in \mathcal{J}} B(\alpha, \beta, p) \frac{\partial (u_\beta u_p)}{\partial x} = \mu \frac{\partial^2 u_\alpha}{\partial x^2} + \sigma(x) \sum_{k=1}^{\infty} 1_{\{\alpha = \varepsilon_k\}} m_k(t), \\ &u_\alpha(0,x) = u_0(x) 1_{\{\alpha = \varepsilon_0\}}. \end{aligned} \tag{5.12}$$

It is a fully coupled system of nonlinear PDEs, whose interaction coefficients $B(\alpha, \beta, p)$ depend on the type of driving noise. Compared with the linear case, (5.12) lacks sparsity, and must be recalculated each time we change distribution. Both features make the nonlinear problem much more expensive to simulate. For a truncated multi-index set $\mathcal{J}_{N,K}$, we need to solve the truncated propagator system

$$\frac{\partial \widehat{u}_\alpha}{\partial t}(t,x) + \frac{1}{2} \sum_{\beta \in \mathcal{J}_{N,K}} \sum_{p \in \mathcal{J}_{N,K}} B(\alpha, \beta, p) \frac{\partial (\widehat{u}_\beta \widehat{u}_p)}{\partial x} = \mu \frac{\partial^2 \widehat{u}_\alpha}{\partial x^2} + \sigma(x) \sum_{k=1}^{\infty} 1_{\{\alpha = \varepsilon_k\}} m_k(t). \tag{5.13}$$

Notice that $\widehat{u}_\alpha$ is not u_α, because the evolution of u_α depends on multi-indices that do not belong to $\mathcal{J}_{N,K}$. In Appendix D we will present the generic procedure to calculate interaction coefficients, as well as explicit formulas for some special types of distribution.

Remark 5.1. In principle, the propagator system can be determined explicitly as long as we only have polynomial nonlinearity. We expand power functions as tensor products in a way that is similar to (5.10). The expansion of nonpolynomial functions is much more challenging. Several methods are presented in [24] to perform general function evaluations on polynomial chaos series.

5.2 Error estimate

For the sake of simplicity, we will focus on the truncation error analysis of the linear SODE system (5.6), and its propagator system (5.7):

$$\mathbf{u}'_\alpha(t) = A\mathbf{u}_\alpha(t) + \sum_{\varepsilon_k \le \alpha} m_k(t) B \mathbf{u}_{\alpha-\varepsilon_k}(t), \quad t \in [0,T]$$

$$\mathbf{u}_\alpha(0) = \mathbf{u_0} 1_{\{\alpha=\varepsilon_0\}},$$

where $\mathbf{u} \in (L^2([0,T] \times \Omega))^M$, A and B are constant $M \times M$ matrices. The reason for such simplification is twofold. Since (5.6) is the spatial discretization of (5.1), it is the equation we are actually dealing with in numerical simulations. Besides, all arguments in this section can be generalized to (5.1) with more technical considerations. We simply replace the Euclidean norm with appropriate Sobolev norms and impose regularity assumptions on $\mathcal{L}$ and $\mathcal{M}$ (see [118]).

Theorem 5.1. *Suppose that $\{m_k\}_{k=1}^{\infty}$ is the trigonometric basis*

$$m_1(t) = \sqrt{\frac{1}{T}}, \quad m_k(t) = \sqrt{\frac{2}{T}} \cos\left(\frac{(k-1)\pi t}{T}\right), \quad k \ge 2. \tag{5.14}$$

Let $\lambda_A := \|A\|_2$ and $\lambda_B := \|B\|_2$ be the matrix norms. Then the mean square error *of the truncated solution $\mathbf{u}_{N,K}(T)$ is bounded by the estimate*

$$\mathbb{E}[|\mathbf{u}_{N,K}(T)-\mathbf{u}(T)|^2] \leq e^{(2\lambda_A+\lambda_B^2)T}\Big(\frac{(\lambda_B^2 T)^{N+1}}{(N+1)!}+\frac{16\lambda_A^2\lambda_B^2T^3}{\pi^4(K-\frac{1}{2})^3}(5+3\lambda_A^2T^2+6\lambda_B^4T^2)\Big)|\mathbf{u}_0|^2. \tag{5.15}$$

Remark 5.2. From (5.15), we conclude that the mean square truncation error converges at an exponential rate with respect to N, and at a cubic rate with respect to K. We improve the error estimate in [68] where the authors only proved linear rate with respect to K. Cubic convergence result can also be found in [56] for a special example of stochastic Burgers equation. However, the approximation error increases exponentially in time. Long time simulation might be impractical. At least more expansion coefficients are required to compensate error growth.

The proof is primarily along the lines in [68]. We first prove a lemma to extract the analytical solution of (5.7).

Lemma 5.1. *Suppose $\{\mathbf{u}_\alpha(t), \alpha \in \mathcal{J}\}$ solves the propagator system (5.7). For each $n \geq 0$ and $\alpha \in \mathcal{J}$ with $|\alpha| = n$, the explicit formula of $\mathbf{u}_\alpha(t)$ is*

$$\mathbf{u}_\alpha(t) = \frac{1}{\alpha!}\int^{(t,n)} \mathbf{F}_n(t,t^{(n)})E_\alpha(t^{(n)})dt^{(n)}, \tag{5.16}$$

where $E_\alpha(t^{(n)})$ is from (4.7) and

$$\mathbf{F}_n(t,t^{(n)}) := e^{(t-t_n)A}Be^{(t_n-t_{n-1})A}B\cdots Be^{t_1A}\mathbf{u}_0,$$

$$\int^{(t,n)} g(t^{(n)})dt^{(n)} := \int_0^t\int_0^{t_n}\cdots\int_0^{t_2} g(t^{(n)})dt_1\cdots dt_{n-1}dt_n.$$

Proof. We prove by induction on n. If $n = 0$, $\mathbf{u}_{\varepsilon_0}(t) = e^{tA}\mathbf{u}_0$. (5.16) is obviously correct. Now for $n \geq 1$ and $|\alpha| = n$, we assume that (5.16) holds for all $\beta \in \mathcal{J}$ with

$|\beta| < n$. By Duhamel's principle,

$$\begin{aligned}
\mathbf{u}_\alpha(t) &= \int_0^t e^{(t-s)A}\Big(\sum_{\varepsilon_k\le\alpha} m_k(s)B\mathbf{u}_{\alpha-\varepsilon_k}(s)\Big)ds \\
&= \frac{1}{\alpha!}\int_0^t e^{(t-s)A}\Big(\sum_{\varepsilon_k\le\alpha}\alpha_k m_k(s)B\int^{(s,n-1)}\mathbf{F}_{n-1}(s,t^{(n-1)})E_{\alpha-\varepsilon_k}(t^{(n-1)})dt^{(n-1)}\Big)ds \\
&= \frac{1}{\alpha!}\int^{(t,n)}\mathbf{F}_n(t,t^{(n)})\Big(\sum_{\varepsilon_k\le\alpha}\alpha_k m_k(t_n)E_{\alpha-\varepsilon_k}(t^{(n-1)})\Big)dt^{(n)} \\
&= \frac{1}{\alpha!}\int^{(t,n)}\mathbf{F}_n(t,t^{(n)})E_\alpha(t^{(n)})dt^{(n)},
\end{aligned}$$

where we use the identity

$$E_\alpha(t^{(n)}) = \sum_{\varepsilon_k\le\alpha}\alpha_k m_k(t_n)E_{\alpha-\varepsilon_k}(t^{(n-1)}).$$

Hence (5.16) is satisfied by any α. □

Proof of Theorem 5.1. According to Parseval's identity, we decompose the truncation error as

$$\mathbb{E}[|\mathbf{u}_{N,K}(T)-\mathbf{u}(T)|^2] = \sum_{n=N+1}^{\infty}\sum_{|\alpha|=n}\alpha!|\mathbf{u}_\alpha(T)|^2 + \sum_{k=K+1}^{\infty}\sum_{n=1}^{N}\sum_{\substack{|\alpha|=n\\ d(\alpha)=k}}\alpha!|\mathbf{u}_\alpha(T)|^2.$$

We only need to show the two inequalities below:

$$\sum_{n=N+1}^{\infty}\sum_{|\alpha|=n}\alpha!|\mathbf{u}_\alpha(T)|^2 \le e^{(2\lambda_A+\lambda_B^2)T}\frac{(\lambda_B^2 T)^{N+1}}{(N+1)!}|\mathbf{u}_0|^2, \tag{5.17}$$

$$\sum_{k=K+1}^{\infty}\sum_{n=1}^{N}\sum_{\substack{|\alpha|=n\\ d(\alpha)=k}}\alpha!|\mathbf{u}_\alpha(T)|^2 \le e^{(2\lambda_A+\lambda_B^2)T}\frac{16\lambda_A^2\lambda_B^2T^3}{\pi^4(K-\frac{1}{2})^3}(5+3\lambda_A^2T^2+6\lambda_B^4T^2)|\mathbf{u}_0|^2. \tag{5.18}$$

As for (5.17), setting $\widetilde{\mathbf{F}}_n(T,\cdot)$ to be the standard symmetrization of $\mathbf{F}_n(T,\cdot)$ ($\mathbf{F}_n$ is extended with zero value outside the simplex $\{t^{(n)} : 0 \leq t_1 \leq \cdots \leq t_n \leq T\}$), we have

$$\mathbf{u}_\alpha(T) = \frac{1}{\alpha!}\int^{(T,n)} \mathbf{F}_n(T, t^{(n)})E_\alpha(t^{(n)})dt^{(n)} = \frac{1}{\alpha!}\int_{[0,T]^n} \widetilde{\mathbf{F}}_n(T, t^{(n)})E_\alpha(t^{(n)})dt^{(n)}.$$

Since $\{E_\alpha, |\alpha| = n\}$ is an orthogonal basis of $\widetilde{H^n}$,

$$\begin{aligned}\sum_{|\alpha|=n} \alpha!|\mathbf{u}_\alpha(T)|^2 &= \sum_{|\alpha|=n} \frac{1}{\alpha!}\Big|\int_{[0,T]^n} \widetilde{\mathbf{F}}_n(T, t^{(n)})E_\alpha(t^{(n)})dt^{(n)}\Big|^2 \\ &= n!\|\widetilde{\mathbf{F}}_n(T, \cdot)\|^2_{H^n} = (n!)^2 \int^{(T,n)} |\widetilde{\mathbf{F}}_n(T, t^{(n)})|^2 dt^{(n)} \\ &= \int^{(T,n)} |\mathbf{F}_n(T, t^{(n)})|^2 dt^{(n)}.\end{aligned} \tag{5.19}$$

For any given $t^{(n)}$,

$$|\mathbf{F}_n(T, t^{(n)})| \leq e^{\lambda_A(T-t_n)}\lambda_B e^{\lambda_A(t_n-t_{n-1})} \cdots \lambda_B e^{\lambda_A t_1}|\mathbf{u}_0| = e^{\lambda_A T}\lambda_B^n|\mathbf{u}_0|.$$

Plugging this into (5.19) yields

$$\sum_{n=N+1}^{\infty}\sum_{|\alpha|=n} \alpha!|\mathbf{u}_\alpha(T)|^2 \leq \Big(\sum_{n=N+1}^{\infty} \frac{e^{2\lambda_A T}(\lambda_B^2 T)^n}{n!}\Big)|\mathbf{u}_0|^2 \leq e^{(2\lambda_A+\lambda_B^2)T}\frac{(\lambda_B^2 T)^{N+1}}{(N+1)!}|\mathbf{u}_0|^2,$$

which exactly recovers (5.17). Here we use the mean-value form of the remainder term of Taylor's expansion:

$$\sum_{n=N+1}^{\infty} \frac{x^n}{n!} = e^{\theta x}\frac{x^{N+1}}{(N+1)!} \text{ for some } \theta \in [0, 1].$$

Proof of (5.18) is more involved. For any α with $|\alpha| = n$ and $d(\alpha) = k$,

$$\begin{aligned} &\int^{(T,n)} \mathbf{F}_n(T, t^{(n)}) E_\alpha(t^{(n)}) dt^{(n)} \\ =&\sum_{j=1}^{n} \int^{(T,n-1)} \Big(\int_{t_{j-1}}^{t_{j+1}} \mathbf{F}_n(T, t^{(n)}) m_k(t_j) dt_j \Big) E_{\alpha-\varepsilon_k}(t^{(n\backslash j)}) dt^{(n\backslash j)} \\ =&\sum_{j=1}^{n} \int^{(T,n-1)} \Big(\int_{t_{j-1}}^{t_j} \mathbf{F}_n(T, t^{(n\backslash j,s)}) m_k(s) ds \Big) E_{\alpha-\varepsilon_k}(t^{(n-1)}) dt^{(n-1)}, \end{aligned} \tag{5.20}$$

where $t^{(n\backslash j)}$ is the short hand notation of $(t_1, \cdots, t_{j-1}, t_{j+1}, \cdots, t_n)$ and $t^{(n\backslash j,s)}$ is the short hand notation of $(t_1, \cdots, t_{j-1}, s, t_j, \cdots, t_{n-1})$. We also adopt the convention $t_0 = 0, t_{n+1} = T$. Define

$$\begin{aligned} M_k^1(t) :=& \int_0^t m_k(s) ds = \frac{\sqrt{2T}}{(k-1)\pi} \sin\Big(\frac{(k-1)\pi t}{T}\Big), \\ M_k^2(t) :=& \int_0^t M_k^1(s) ds = \frac{\sqrt{2T^3}}{(k-1)^2\pi^2} \Big(1 - \cos\Big(\frac{(k-1)\pi t}{T}\Big)\Big), \end{aligned}$$

and

$$\begin{aligned} &\mathbf{F}_n^j(T, t^{(n)}) := \frac{\partial \mathbf{F}_n}{\partial t_j}(T, t^{(n)}) \\ =&e^{(T-t_n)A} B \cdots e^{(t_{j+1}-t_j)A}(BA - AB) e^{(t_j - t_{j-1})A} \cdots B e^{t_1 A} \mathbf{u}_0, \\ &\mathbf{F}_n^{jj}(T, t^{(n)}) = \frac{\partial^2 \mathbf{F}_n}{\partial t_j^2}(T, t^{(n)}) \\ =&e^{(T-t_n)A} B \cdots e^{(t_{j+1}-t_j)A}(A^2 B + BA^2 - 2ABA) e^{(t_j - t_{j-1})A} \cdots B e^{t_1 A} \mathbf{u}_0. \end{aligned}$$

The following estimates are right at hand:

$$|\mathbf{F}_n^j(T, t^{(n\backslash j,s)})| \le 2e^{\lambda_A T} \lambda_A \lambda_B^n |\mathbf{u}_0|, \quad |\mathbf{F}_n^{jj}(T, t^{(n\backslash j,s)})| \le 4e^{\lambda_A T} \lambda_A^2 \lambda_B^n |\mathbf{u}_0|, \tag{5.21}$$

$$M_k^1(T) = 0 \quad , |M_k^2(T)| \le \frac{\sqrt{8T^3}}{(k-1)^2\pi^2}, \quad \int_0^T (M_k^2(t))^2 dt = \frac{3T^4}{(k-1)^4\pi^4}. \tag{5.22}$$

Then we perform integration-by-parts twice on the inner integral of (5.20) and obtain

$$\int^{(T,n)} \mathbf{F}_n(T,t^{(n)})E_\alpha(t^{(n)})dt^{(n)} := \int^{(T,n-1)} \mathbf{G}_{n,k}(T,t^{(n-1)})E_{\alpha-\varepsilon_k}(t^{(n-1)})dt^{(n-1)}.$$

where

$$\mathbf{G}_{n,k}(T,t^{(n-1)}) := \mathbf{G}^1_{n,k}(T,t^{(n-1)}) + \mathbf{G}^2_{n,k}(T,t^{(n-1)}) + \mathbf{G}^3_{n,k}(T,t^{(n-1)}),$$

and

$$\begin{aligned}
\mathbf{G}^1_{n,k}(T,t^{(n-1)}) &= \sum_{j=1}^{n} \Big(\mathbf{F}_n(T,t^{(n\backslash j,s)})M^1_k(s)\Big|_{s=t_{j-1}}^{s=t_j} \Big),\\
\mathbf{G}^2_{n,k}(T,t^{(n-1)}) &= -\sum_{j=1}^{n} \Big(\mathbf{F}^j_n(T,t^{(n\backslash j,s)})M^2_k(s)\Big|_{s=t_{j-1}}^{s=t_j} \Big),\\
\mathbf{G}^3_{n,k}(T,t^{(n-1)}) &= \sum_{j=1}^{n} \Big(\int_{t_{j-1}}^{t_j} \mathbf{F}^{jj}_n(T,t^{(n\backslash j,s)})M^2_k(s)ds \Big).
\end{aligned}$$

Since $M^1_k(T) = 0$ and $\mathbf{F}_n(T,t^{(n\backslash j,s)}; s = t_j) = \mathbf{F}_n(T,t^{(n\backslash (j+1),s)}; s = t_j)$ for $1 \le j \le n-1$, $\mathbf{G}^1_{n,k}(T,t^{(n-1)}) = 0$. By (5.21) and (5.22), the other two terms are bounded by

$$\begin{aligned}
|\mathbf{G}^2_{n,k}(T,t^{(n-1)})| &\le 2e^{\lambda_A T}\lambda_A\lambda_B^n|\mathbf{u}_0|\Big(2\sum_{j=1}^{n-1}|M^2_k(t_j)| + |M^2_k(T)|\Big)\\
&\le 4e^{\lambda_A T}\lambda_A\lambda_B^n|\mathbf{u}_0|\Big(\sum_{j=1}^{n-1}|M^2_k(t_j)| + \frac{\sqrt{2T^3}}{(k-1)^2\pi^2}\Big),\\
|\mathbf{G}^3_{n,k}(T,t^{(n-1)})| &\le 4e^{\lambda_A T}\lambda_A^2\lambda_B^n|\mathbf{u}_0|\Big(\int_0^T |M^2_k(t)|dt\Big)\\
&\le 4e^{\lambda_A T}\lambda_A^2\lambda_B^n|\mathbf{u}_0|\sqrt{T\int_0^T (M^2_k(t))^2dt}\\
&= 4e^{\lambda_A T}\lambda_A^2\lambda_B^n|\mathbf{u}_0|\frac{\sqrt{3T^5}}{(k-1)^2\pi^2}.
\end{aligned}$$

Similar to the idea in (5.19),

$$
\begin{aligned}
&\sum_{|\alpha|=n,d(\alpha)=k} \alpha!|\mathbf{u}_\alpha(T)|^2 = \sum_{|\alpha|=n,d(\alpha)=k} \frac{1}{\alpha!}\Big|\int^{(T,n)} \mathbf{F}_n(T,t^{(n)})E_\alpha(t^{(n)})dt^{(n)}\Big|^2 \\
=&\sum_{|\alpha|=n,d(\alpha)=k} \frac{1}{\alpha!}\Big|\int^{(T,n-1)} \mathbf{G}_{n,k}(T,t^{(n-1)})E_{\alpha-\varepsilon_k}(t^{(n-1)})dt^{(n-1)}\Big|^2 \\
\leq&\sum_{|\beta|=n-1} \frac{1}{\beta!}\Big|\int^{(T,n-1)} \mathbf{G}_{n,k}(T,t^{(n-1)})E_{\beta}(t^{(n-1)})dt^{(n-1)}\Big|^2 \\
=&\int^{(T,n-1)} |\mathbf{G}_{n,k}(T,t^{(n-1)})|^2 dt^{(n-1)} \leq \int^{(T,n-1)} \Big(|\mathbf{G}^2_{n,k}(T,t^{(n-1)})| + |\mathbf{G}^3_{n,k}(T,t^{(n-1)})|\Big)^2 dt^{(n-1)} \\
\leq& 16e^{2\lambda_A T}\lambda_A^2\lambda_B^{2n}\int^{(T,n-1)} \Big(\sum_{j=1}^{n-1}|M_k^2(t_j)| + \frac{\sqrt{2T^3}}{(k-1)^2\pi^2} + \lambda_A\frac{\sqrt{3T^5}}{(k-1)^2\pi^2}\Big)^2 dt^{(n-1)} \\
\leq& 48e^{2\lambda_A T}\lambda_A^2\lambda_B^{2n}\int^{(T,n-1)} \Big((n-1)\Big(\sum_{j=1}^{n-1}(M_k^2(t_j))^2\Big) + \frac{2T^3}{(k-1)^4\pi^4} + \lambda_A^2\frac{3T^5}{(k-1)^4\pi^4}\Big)dt^{(n-1)}.
\end{aligned}
$$

The remaining part of proof is clear. Since $\sum_{j=1}^{n-1}(M_k^2(t_j))^2$ is a symmetric function,

$$
\begin{aligned}
\int^{(T,n-1)} \Big(\sum_{j=1}^{n-1}(M_k^2(t_j))^2\Big)dt^{(n-1)} &= \frac{1}{(n-1)!}\int_{[0,T]^{n-1}} \Big(\sum_{j=1}^{n-1}(M_k^2(t_j))^2\Big)dt^{(n-1)} \\
&= \frac{n-1}{(n-1)!}\frac{3T^{n+2}}{(k-1)^4\pi^4}.
\end{aligned}
$$

Therefore

$$
\sum_{|\alpha|=n,d(\alpha)=k} \alpha!|\mathbf{u}_\alpha(T)|^2 \leq \frac{48e^{2\lambda_A T}\lambda_A^2\lambda_B^{2n}}{(n-1)!(k-1)^4\pi^4}\Big(3(n-1)^2T^{n+2} + 2T^{n+2} + 3\lambda_A^2 T^{n+4}\Big).
$$

Summing over n and k yields:

$$\begin{aligned}
&\sum_{k=K+1}^{\infty}\sum_{n=1}^{N}\sum_{|\alpha|=n,d(\alpha)=k}\alpha!|\mathbf{u}_\alpha(T)|^2\\
\le& e^{2\lambda_A T}\frac{48\lambda_A^2}{\pi^4}\Big(\sum_{k=K+1}^{\infty}\frac{1}{(k-1)^4}\Big)\Big(\sum_{n=1}^{N}\frac{\lambda_B^{2n}(3(n-1)^2T^{n+2}+2T^{n+2}+3\lambda_A^2T^{n+4})}{(n-1)!}\Big)|\mathbf{u}_0|^2\\
\le& e^{(2\lambda_A+\lambda_B^2)T}\frac{16\lambda_A^2\lambda_B^2T^3}{\pi^4(K-\frac{1}{2})^3}(5+3\lambda_A^2T^2+6\lambda_B^4T^2)|\mathbf{u}_0|^2,
\end{aligned}$$

where we use the inequalities

$$\begin{aligned}
&\sum_{k=K+1}^{\infty}\frac{1}{(k-1)^4}\le\sum_{k=K}^{\infty}\int_{k-\frac{1}{2}}^{k+\frac{1}{2}}\frac{1}{x^4}dx=\int_{K-\frac{1}{2}}^{\infty}\frac{1}{x^4}dx=\frac{1}{3(K-\frac{1}{2})^3},\\
&\sum_{n=1}^{N}\frac{x^{n-1}}{(n-1)!}\le\sum_{n=0}^{\infty}\frac{x^n}{n!}=e^x,\\
&\sum_{n=1}^{N}\frac{(n-1)^2x^{n-1}}{(n-1)!}=x+\sum_{n=0}^{N-3}\frac{(n+2)x^{n+2}}{(n+1)!}\le x+\sum_{n=0}^{\infty}\frac{2x^{n+2}}{n!}\le e^x(1+2x^2).
\end{aligned}$$

We have finished the proofs of (5.17) and (5.18). Then (5.15) immediately follows.

□

Remark 5.3. The proof of (5.17) is independent of the choice of $\{m_k\}_{k=1}^{\infty}$, the convergence rate with respect to N is always exponential. The proof of (5.18) relies on the trigonometric basis assumption. The crucial property is (5.22), which enables cubic convergence. In fact, the proof will work for any orthonormal basis such that

$$M_k^1(T)=0,\quad M_k^2(T)=O(k^{-2}),\quad \|M_k^2\|_H=O(k^{-2}),\quad \forall k\ge 2.$$

For example, consider the scaled Legendre basis

$$m_k(t)=\sqrt{\frac{2k-1}{T}}L_{k-1}\Big(\frac{2t}{T}-1\Big). \tag{5.23}$$

We are able to show that

$$M_k^1(t) = \frac{1}{2}\sqrt{\frac{T}{2k-1}}\Big(L_k\Big(\frac{2t}{T}-1\Big) - L_{k-2}\Big(\frac{2t}{T}-1\Big)\Big),$$

$$\begin{aligned} M_k^2(t) = &\frac{1}{4(2k+1)}\sqrt{\frac{T^3}{2k-1}}\Big(L_{k+1}\Big(\frac{2t}{T}-1\Big) - L_{k-1}\Big(\frac{2t}{T}-1\Big)\Big) \\ &- \frac{1}{4(2k-3)}\sqrt{\frac{T^3}{2k-1}}\Big(L_{k-1}\Big(\frac{2t}{T}-1\Big) - L_{k-3}\Big(\frac{2t}{T}-1\Big)\Big), \end{aligned}$$

where L_k is taken to be 0 for negative k. Then $M_k^1(T) = 0$ for any $k \geq 2$ and $M_k^2(T) = 0$ for any $k \geq 3$. We also have $\|M_k^2\|_H = O(k^{-2})$. Therefore for Legendre basis, the convergence rate with respect to K is still cubic.

Remark 5.4. If A and B commute such that $AB = BA$, then

$$\mathbf{F}_n(T, t^{(n)}) = e^{(T-t_n)A}e^{(t_n-t_{n-1})A}\cdots e^{t_1A}B^n\mathbf{u}_0 = e^{TA}B^n\mathbf{u}_0$$

is a constant vector that does not depend on $t^{(n)}$. Consequently, for any $|\alpha| = n$,

$$\begin{aligned} \mathbf{u}_\alpha(T) &= \frac{e^{TA}B^n\mathbf{u}_0}{\alpha!}\int^{(T,n)} E_\alpha(t^{(n)})dt^{(n)} = \frac{e^{TA}B^n\mathbf{u}_0}{n!\alpha!}\int_{[0,T]^n} E_\alpha(t^{(n)})dt^{(n)} \\ &= \frac{e^{TA}B^n\mathbf{u}_0}{\alpha!}\int_{[0,T]^n} m_{i_\alpha^1}(t_1)\cdots m_{i_\alpha^n}(t_n)dt^{(n)} = \frac{e^{TA}B^n\mathbf{u}_0}{\alpha!}M_{i_\alpha^1}^1(T)\cdots M_{i_\alpha^n}^1(T). \end{aligned}$$

For trigonometric basis (and Legendre basis), $M_k^1(T) = 0$ for any $k \geq 2$. That is, $u_\alpha(T) = 0$ whenever $d(\alpha) = i_\alpha^n \geq 2$. It is enough to fix $K = 1$ and only consider the truncation on N. The resulting error estimate is simply

$$\mathbb{E}[|\mathbf{u}_{N,1}(T) - \mathbf{u}(T)|^2] \leq e^{(2\lambda_A+\lambda_B^2)T}\frac{(\lambda_B^2T)^{N+1}}{(N+1)!}|\mathbf{u}_0|^2. \tag{5.24}$$

5.3 Numerical experiments

We first introduce some post-processing techniques for the numerical solution written as polynomial chaos expansion:

$$u_{N,K}(t,\mathbf{x}) = \sum_{\alpha\in\mathcal{J}_{N,K}} \widehat{u}_\alpha(t,\mathbf{x})\Phi_\alpha.$$

Moments can be computed directly, the first and second moments are

$$\mathbb{E}[u_{N,K}] = u_{\varepsilon_0}, \quad \mathbb{E}[u_{N,K}^2] = \sum_{\alpha\in\mathcal{J}_{N,K}} \alpha!\widehat{u}_\alpha^2,$$

and The third and fourth moments are given by

$$\begin{aligned}\mathbb{E}[u_{N,K}^3] &= \sum_{\alpha\in\mathcal{J}_{N,K}} \alpha!\widehat{u}_\alpha\Big(\sum_{\beta\in\mathcal{J}_{N,K}}\sum_{p\in\mathcal{J}_{N,K}} B(\alpha,\beta,p)\widehat{u}_\beta\widehat{u}_p\Big),\\ \mathbb{E}[u_{N,K}^4] &= \sum_{\alpha\in\mathcal{J}_{2N,K}} \alpha!(\Big(\sum_{\beta\in\mathcal{J}_{N,K}}\sum_{p\in\mathcal{J}_{N,K}} B(\alpha,\beta,p)\widehat{u}_\beta\widehat{u}_p\Big)^2.\end{aligned}$$

In the computation of fourth moment, we use the fact that the expansion order of $u_{N,K}^2(t,x)$ is at most $2N$. For linear problems, the first two moments remain the same for all kinds of noises, while higher moments always depend on the type of randomness (due to the emergence of $B(\alpha,\beta,p)$). Other statistics can be computed via random sampling. We simply generate L i.i.d. realizations of $(\xi_1,\cdots,\xi_K)$, denoted by $\{\xi_1^{(l)},\cdots,\xi_K^{(l)}\}$ for each $1\le l\le L$. The sample points of $u_{N,K}(t,\mathbf{x})$ are:

$$u_{N,K}^{(l)} := \sum_{J\in\mathcal{J}_{N,K}} \widehat{u}_\alpha\Phi_\alpha(\xi_1^{(l)},\cdots,\xi_K^{(l)}).$$

Then for any function f, the expectation $\mathbb{E}[f(u_{N,K})]$ is approximated by a sample mean. We can also plot the normalized histogram of these sample points to visualize empirical distribution.

In this section, we always assume that $\{\xi_k\}_{k=1}^{\infty}$ are i.i.d. random variables. To be more specific, we will test three types of randomness: Gaussian noise with Hermite chaos (Example 4.2.1), uniform noise with Legendre chaos (Example 4.2.2), and Beta($\frac{1}{2}, \frac{1}{2}$) noise with Chebyshev chaos. In the last situation, $\{\xi_k\}_{k=1}^{\infty}$ are supported on $[-\sqrt{2}, \sqrt{2}]$ with probability density function

$$\rho(\xi) = \frac{\sqrt{2-\xi_1^2}}{2\pi}, \quad \xi \in [-\sqrt{2}, \sqrt{2}].$$

The univariate gPC basis functions are scaled Chebyshev polynomials $\{T_n(x)\}_{n=0}^{\infty}$:

$$\varphi_0(\xi) = 1, \quad \varphi_n(\xi) = \sqrt{2n!}T_n\Big(\frac{\xi}{\sqrt{2}}\Big), \;\; n \geq 1. \tag{5.25}$$

Now we proceed to solve distribution-free linear and nonlinear SPDEs numerically. Propagator systems are integrated in time with a fourth order Runge-Kutta method. Time step size is small enough so that error from temporal discretization is negligible. Here $\{m_k(t)\}_{k=1}^{\infty}$ is taken to be the trigonometric basis (5.14). The results of the scaled Legendre basis (5.23) are almost indistinguishable from trigonometric basis and will not be reported. We will conduct comparisons with reference solutions, and among different types of driving noise. Several techniques are adopted to compute reference solutions.

1. Moment equations: the ODE of first few moments, available to linear Itô type SPDEs.
2. Fokker-Planck equation: the PDE of probability density function, available to low-dimensional Itô type SODEs.
3. Monte Carlo simulation: the most commonly used and least restrictive ap-

proach, available to additive noise and/or Itô type SPDEs. The main bottleneck is high computational cost and low accuracy.

The first two methods are free from sampling error, and will be selected whenever possible. Monte Carlo simulation serves as a backup option.

Example 5.3.1 (Linear SODE)**.** Suppose that $u = u(t) \in L^2([0,T] \times \Omega)$ satisfies the following linear SODE

$$\begin{aligned} & u'(t) = u(t) + 1 + u(t) \diamond \dot{\mathcal{N}}(t), \quad t \in [0,T], \\ & u(0) = 1. \end{aligned} \tag{5.26}$$

For Gaussian noise, it is equivalent to the Itô type SODE

$$du(t) = (u(t)+1)dt + u(t)dW(t). \tag{5.27}$$

By Itô's formula, we are able to derive its moment equations

$$\begin{aligned} \frac{d\mathbb{E}[u(t)]}{dt} &= \mathbb{E}[u(t)] + 1, \\ \frac{d\mathbb{E}[u^2(t)]}{dt} &= 3\mathbb{E}[u^2(t)] + 2\mathbb{E}[u(t)], \\ \frac{d\mathbb{E}[u^3(t)]}{dt} &= 6\mathbb{E}[u^3(t)] + 3\mathbb{E}[u^2(t)], \\ \frac{d\mathbb{E}[u^4(t)]}{dt} &= 10\mathbb{E}[u^4(t)] + 6\mathbb{E}[u^3(t)]. \end{aligned} \tag{5.28}$$

The analytical solution to (5.28) is

$$\begin{aligned} \mathbb{E}[u(t)] &= 2e^t - 1, \\ \mathbb{E}[u^2(t)] &= \frac{7}{3}e^{3t} - 2e^t + \frac{2}{3}, \\ \mathbb{E}[u^3(t)] &= \frac{37}{15}e^{6t} - \frac{7}{3}e^{3t} + \frac{6}{5}e^t - \frac{1}{3}, \\ \mathbb{E}[u^4(t)] &= \frac{38}{15}e^{10t} - \frac{37}{15}e^{6t} + \frac{4}{3}e^{3t} - \frac{8}{15}e^t + \frac{2}{15}. \end{aligned} \tag{5.29}$$

We can also find out the probability density function of $u(t)$, denoted by $\rho(u,t)$. The governing PDE of $\rho(t,u)$, i.e. Fokker-Planck equation is:

$$\begin{aligned}&\partial_t\rho(t,u) = -\partial_u((u+1)\rho) + \partial_u^2\Big(\frac{u^2}{2}\rho\Big), \quad (t,u)\in(0,T]\times(0,\infty),\\&\rho(0,u) = \delta(u-1), \quad u\in(0,\infty),\end{aligned} \tag{5.30}$$

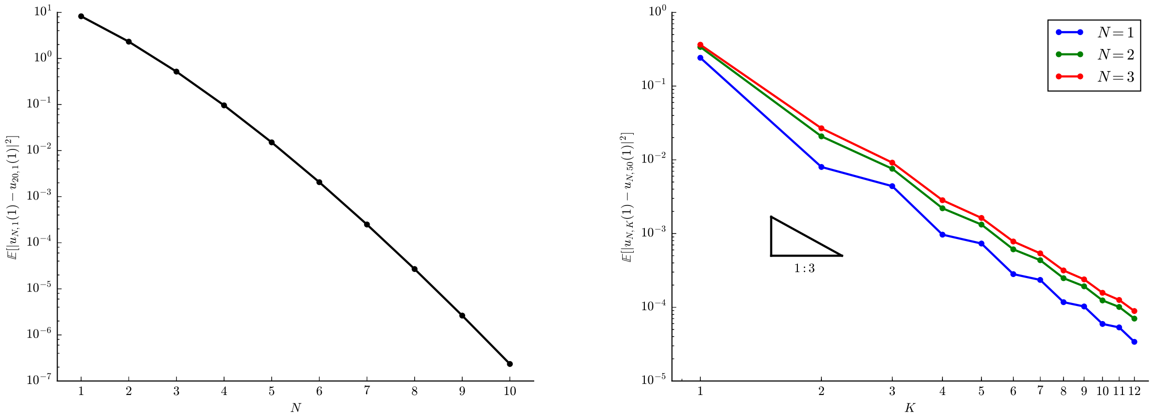
where δ is the Dirac delta function. Substituting $v = \log u$, we simplify (5.30) and get

$$\partial_t\rho(t,v) = \Big(\frac{1}{2} - e^{-v}\Big)\partial_v\rho + \frac{1}{2}\partial_v^2\rho, \quad \rho(0,v) = \delta(v). \tag{5.31}$$

(5.31) is a standard convection-diffusion equation, which can be solved by the LDG method. The computational domain is $[-7,7]$ with zero boundary condition, and divided into 501 quadratic elements. For uniform and Beta noise, we do not have derive moment equations or Fokker-Planck equation. The first and second moments will be the same, as a result of linearity.

We evolve the propagator ODE system up to end time $T = 1$ with time step size $\delta t = 10^{-4}$. In order to examine the convergence rates of mean square truncation error, ideally we should compute $\mathbb{E}[|u_{N,K} - u_{\infty,K}|^2]$ to single out the error induced by N, and $\mathbb{E}[|u_{N,K} - u_{N,\infty}|^2]$ to single out the error induced by K. In practice we use $u_{20,K}$ to approximate $u_{\infty,K}$, and $u_{N,50}$ to approximate $u_{N,\infty}$. Figure 5.1 contains the semi-log plot of $\mathbb{E}[|u_{N,K}(1) - u_{20,K}(1)|^2]$ versus N with $K = 1$, and the log-log plot of $\mathbb{E}[|u_{N,K}(1) - u_{N,50}(1)|^2]$ versus K with $N = 1, 2, 3$. The numerical rate of convergence with respect to N is evidently exponential. The plot with respect to K has a zigzag shape (especially for $N = 1$), but the average slope is close to 3. The cubic convergence rate is more clearly seen in Table 5.1, where we only compare even values of K to average out the zig-zag profile.

Next we compute moments of the truncated solution. Table 5.2 lists the first



(a) Convergence with respect to N

(b) Convergence with respect to K

Figure 5.1: Example 5.3.1: Plots of mean square truncation error with respect to N and K. Left panel shows the semi-log plot of $\mathbb{E}[|u_{N,1}(1) - u_{20,1}(1)|^2]$ versus N for $N = 1, \cdots, 10$. Right panel shows the log-log plot of $\mathbb{E}[|u_{N,K}(1) - u_{N,50}(1)|^2]$ versus K for $N = 1, 2, 3$ and $K = 1, \cdots, 12$.

Table 5.1: Example 5.3.1: Values and numerical convergence orders of $e_{N,K} := \mathbb{E}[|u_{N,K}(1) - u_{N,50}(1)|^2]$ for $N = 1, 2, 3$ and $K = 2, 4, \cdots, 12$. The orders are given by $\log(\frac{e_{N,K}}{e_{N,K+2}}) / \log(\frac{K+2}{K})$.

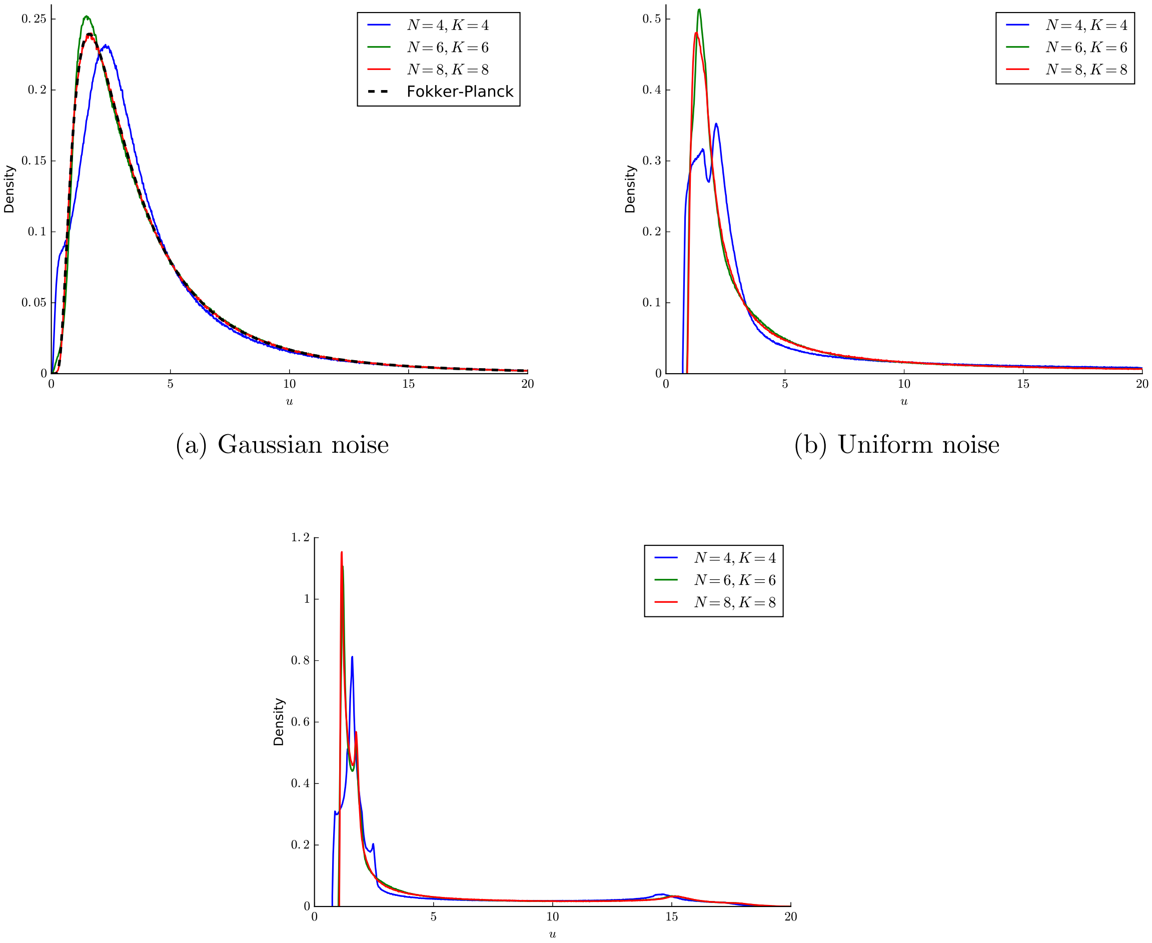
K	$N = 1$		$N = 2$		$N = 3$	
	error	order	error	order	error	order
2	7.997e-3	-	2.080e-2	-	2.677e-2	-
4	9.659e-4	3.049	2.201e-3	3.241	2.831e-3	3.241
6	2.816e-4	3.040	6.102e-4	3.164	7.799e-4	3.180
8	1.173e-4	3.044	2.480e-4	3.129	3.158e-4	3.143
10	5.939e-5	3.050	1.238e-4	3.114	1.572e-4	3.126
12	3.398e-5	3.062	7.018e-5	3.113	8.895e-5	3.123

four central moments of $u_{N,K}(1)$ with all three types of randomness, by taking $N = K = 4$, $N = K = 6$ and $N = K = 8$. For Gaussian noise, moments of $u(1)$ are also included according to (5.2). Two conclusions can be drawn from the table. Comparing the central moments of $u_{N,K}(1)$ and $u(1)$, we see that the variance can be approximated well with relatively few chaos expansion terms, but more terms are needed to resolve higher moments. Comparing among types of driving noise, we notice the large discrepancy in third and fourth moments, despite the fact that they share the same mean and variance. It is probably related to the kurtosis of different distributions. The fourth moment of ξ_1 is 3 for standard Gaussian distribution, $\frac{9}{5}$ for uniform distribution, and $\frac{3}{2}$ for Beta$(\frac{1}{2}, \frac{1}{2})$ distribution. Higher kurtosis in $\{\xi_k\}_{k=1}^{\infty}$ leads to higher kurtosis in $u_{N,K}$.

Table 5.2: Example 5.3.1: Comparison of central moments of $u_{N,K}(1)$ and $u(1)$. We take $N = K = 4$, $N = K = 6$ and $N = K = 8$. Higher moments of $u(1)$ are only available to Gaussian noise.

Type of noise	Type of moment	$u(1)$	$u_{4,4}(1)$	$u_{6,6}(1)$	$u_{8,8}(1)$
Gaussian	Variance	22.413	22.313	22.410	22.413
	Third central moment	565.548	487.838	558.223	565.138
	Fourth central moment	41759.97	22914.36	37479.48	41233.50
Uniform	Variance	22.413	22.313	22.410	22.413
	Third central moment	-	208.415	220.604	221.203
	Fourth central moment	-	3080.52	3446.20	3474.01
Beta	Variance	22.413	22.313	22.410	22.413
	Third central moment	-	150.739	159.566	160.011
	Fourth central moment	-	1789.49	1957.34	1970.63

Empirical distribution is a more intuitive way to describe random variables. Figure 5.2 demonstrates the empirical probability densities of $u_{4,4}(1)$, $u_{6,6}(1)$ and $u_{8,8}(1)$ for all three types of randomness. All densities are estimated by normalized histograms with 10^3 bins out of 10^7 i.i.d samples of $u_{N,K}(1)$. For Gaussian noise, the numerical solution of Fokker-Planck equation at $t = 1$ is also displayed in Figure 5.2. The empirical densities of $u_{4,4}(1)$ and $u_{6,6}(1)$ slightly deviates from the Fokker-Planck solution, and the empirical density of $u_{8,8}(1)$ agrees with the Fokker-Planck solution very well. As for the comparison among three types of noise, their density patterns are qualitatively different. The distributions with Gaussian noise spread out and have long tails. The density profiles with Beta$(\frac{1}{2}, \frac{1}{2})$ noise are mostly constrained in a narrow region, and the density profiles with uniform noise lie somewhere in between. This also explains the difference of higher moments in Table 5.2.



(a) Gaussian noise

(b) Uniform noise

(c) Beta($\frac{1}{2}$, $\frac{1}{2}$) noise

Figure 5.2: Example 5.3.1: Normalized histograms of $u_{N,K}(1)$ out of 10^7 i.i.d samples. We take $N = K = 4$, $N = K = 6$ and $N = K = 8$. For Gaussian distribution, the black dashed line represents the numerical solution of Fokker-Planck equation (5.31). Values larger than 20 are discarded. Number of bins is 10^3.

In summary, in this example we study a very simple linear SODE so that reference solutions of moments and density function can be acquired without much effort. By linearity, we only need to solve the propagator once and save the expansion coefficients for post-processing. The mean square truncation error results in Figure 5.1 and Table 5.1 indicate exponential convergence with respect to N and cubic convergence with respect to K, as predicted by Theorem 5.1. Table 5.2 and Figure 5.2 highlight the contrast of higher moments and empirical distributions with different noises. We also observe that although the mean square error converges rapidly, high order chaos expansion terms are beneficial to the approximation of higher moments and distribution.

Example 5.3.2 (Linear parabolic PDE)**.** We solve the linear parabolic PDE in Example 5.1.1. Consider the one-dimensional space region $\Gamma = [0, 2\pi]$ with periodic boundary condition. The initial data is $u_0(x) = \cos x$. Differential operators $\mathcal{L}$ and $\mathcal{M}$ are set to be [117, 118]

$$\mathcal{L} = 0.145\frac{\partial^2}{\partial x^2} + 0.1\sin x\frac{\partial}{\partial x}, \quad \mathcal{M}u = 0.5\frac{\partial}{\partial x}.$$

Fourier collocation method with $M = 32$ collocation points is used for spatial discretization. Let $\{x_i\}_{i=1}^M$ be the set of equidistant collocation points such that $x_i = \frac{2\pi(i-1)}{M}$. Then $u(t,\cdot)$ is identified by the length-M vector $\mathbf{u}(t)$. We only need to work on the linear SODE system (5.6). For Gaussian noise, it is possible to obtain moment equations of (5.6). Direct application of Itô's formula yields

$$\frac{d\mathbb{E}[u_i u_j]}{dt} = \sum_{l=1}^{M}(A_{il}\mathbb{E}[u_j u_l] + A_{jl}\mathbb{E}[u_i u_l]) + \sum_{l=1}^{M}\sum_{r=1}^{M} B_{il}B_{jr}\mathbb{E}[u_l u_r], \quad 1 \le i, j \le M. \tag{5.32}$$

It is a M^2-dimensional ODE system describing the evolution of covariance matrix. Higher moment equations are written in a similar fashion, but they are computationally formidable in that for d-th moment, we have to tackle the full tensor product system with M^d dimensions. Solving Fokker-Planck equation is also not feasible, as it depends on M spatial variables. We will use Monte Carlo simulation to approximate higher moments and density function. Let $\mathbf{u}_p$ be the sample of $\mathbf{u}$ at p-th time step. The update rule of the second order weak scheme is (see Chapter 2 of [80])

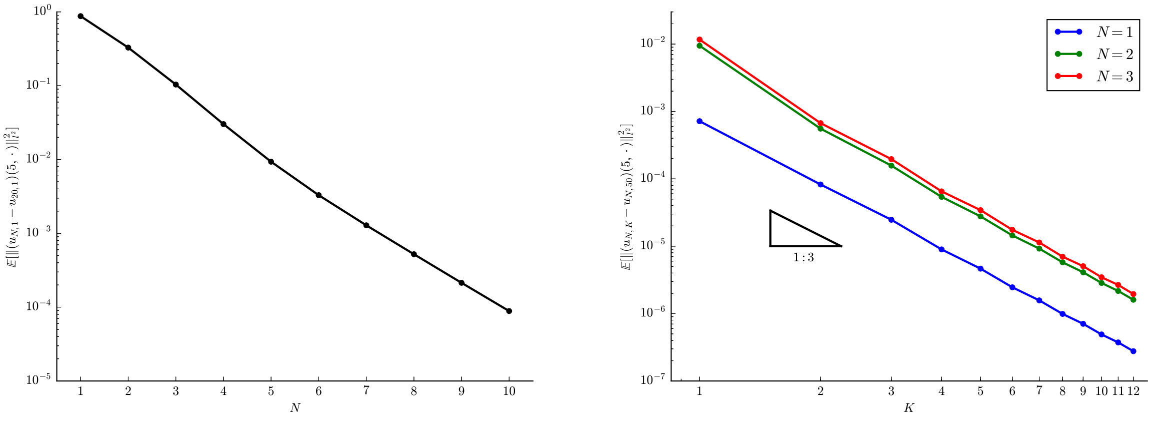
$$\mathbf{u}_{p+1} = \mathbf{u}_p + \delta t A\mathbf{u}_p + \frac{\delta t^2}{2}A^2\mathbf{u}_p + \sqrt{\delta t}\zeta_p B\mathbf{u}_p + \frac{\delta t}{2}(\zeta_p^2 - 1)B^2\mathbf{u}_p + \frac{\sqrt{\delta t^3}}{2}\zeta_p(AB + BA)\mathbf{u}_p,$$

where ζ_p are i.i.d. standard Gaussian random variables.

The propagator system is run up to $T = 5$ with Runge-Kutta time step size $\delta t = 10^{-3}$. Figure 5.3 depicts the mean square truncation error with respect to N and K. Same as in the previous example, we compute $\mathbb{E}[\|u_{N,K}(5,\cdot) - u_{20,K}(5,\cdot)\|_{l^2}^2]$ as the proxy of error induced by N, and $\mathbb{E}[\|u_{N,K}(5,\cdot) - u_{N,50}(5,\cdot)\|_{l^2}^2]$ as the proxy of error induced by K, where the discrete L^2 norm for $v \in H$ is defined by

$$\|v\|_{l^2}^2 := \frac{2\pi}{M} \sum_{i=1}^{M} v(x_i)^2.$$

Once again, the N-version convergence shows exponential rate, and the K-version convergence shows cubic rate.

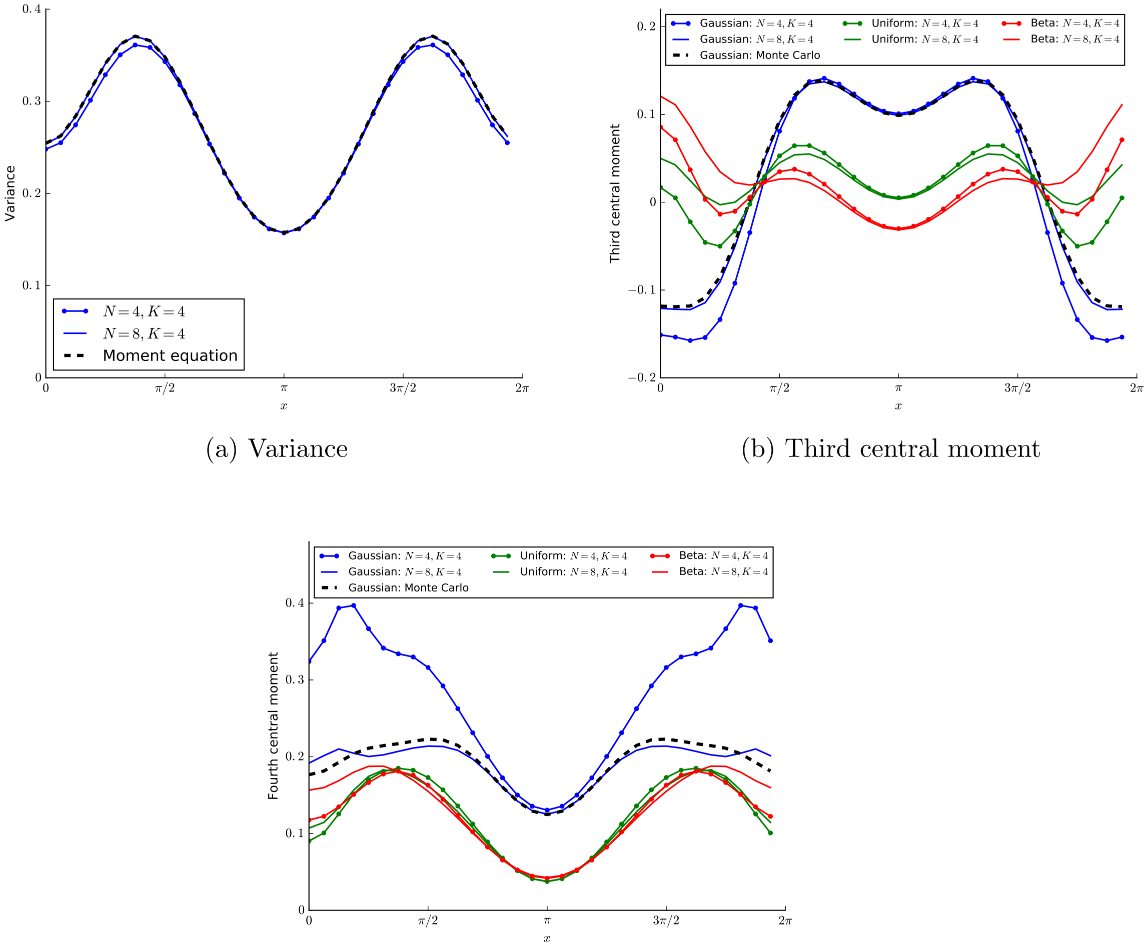


(a) Convergence with respect to N

(b) Convergence with respect to K

Figure 5.3: Example 5.3.2: Plots of mean square truncation error in discrete l^2 norm with respect to N and K. Left panel shows the semi-log plot of $\mathbb{E}[\|u_{N,1}(5,\cdot) - u_{20,1}(5,\cdot)\|_{l^2}^2]$ versus N for $N = 1, \cdots, 10$. Right panel shows the log-log plot of $\mathbb{E}[\|u_{N,K}(5,\cdot) - u_{N,50}(5,\cdot)\|_{l^2}^2]$ versus K for $N = 1, 2, 3$ and $K = 1, \cdots, 12$.

The first four central moments of $u_{N,K}(5,\cdot)$ with all three noises are plotted in Figure 5.4. We consider $N = 4, 8$ and $K = 4$. Variance of $u(5,\cdot)$ is also exhibited as reference by solving the moment equation (5.32) through fourth order Runge-Kutta method with time step size $\delta t = 10^{-3}$. For Gaussian noise, third and fourth central moments of the Monte Carlo solution with 10^6 sample paths and time step size $\delta t = 10^{-3}$ are also shown in Figure 5.4. We observe that both $u_{4,4}$ and $u_{8,4}$ predict the variance sufficiently well, but only $u_{8,4}$ succeeds in resolving higher moments. As for the comparison among three types of noise, their third central moments have different structures. Fourth central moments look similar in shape but different in magnitude.



(a) Variance

(b) Third central moment

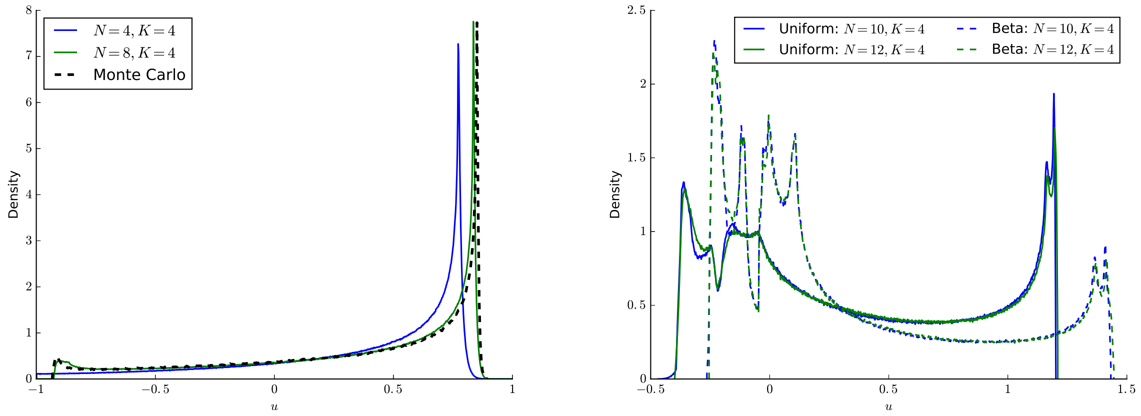
(c) Fourth central moment

Figure 5.4: Example 5.3.2: Central moments of $u_{K,N}(5,\cdot)$. We take $N = 4, 8$ and $K = 4$. Black dashed lines are reference solutions. Variance is computed via moment equation (5.32) (for all types of noise), and higher moments are approximated by Monte Carlo method with 10^6 samples (only for Gaussian noise).

Empirical distributions at the second collocation point x_2 are illustrated in Figure 5.5. For Gaussian noise, we plot normalized histograms of $u_{4,4}(5, x_2)$ and $u_{8,4}(5, x_2)$ out of 10^7 samples in the left panel, as well as the histogram of 10^6 Monte Carlo samples for reference. Number of bins is 10^3. We underline that the distribution of $u(5, x_2)$ is supported in $[-1, 1]$, as a result of averaging over characteristic lines [79]. The empirical distribution of Monte Carlo sampling is indeed inside $[-1, 1]$ and highly rightly skewed. Almost all samples of $u_{4,4}(5, x_2)$ and $u_{8,4}(5, x_2)$ fall into $[-1, 1]$ as well. We discard the few outlier samples in the figure. The empirical density function of $u_{4,4}$ underestimates the position of right peak, and the empirical density function $u_{8,4}$ is almost on top of the reference solution. For uniform noise and Beta$(\frac{1}{2}, \frac{1}{2})$ noise, we plot the empirical density functions of $u_{10,4}(5, x_2)$ and $u_{12,4}(5, x_2)$ in the right panel. The profiles of $u_{10,4}$ and $u_{12,4}$ nearly coincide with each other, which suggests that we achieve reasonable approximations of the true density functions. These density patterns look dramatically different from the Gaussian noise case, in that they are neither supported in $[-1, 1]$ nor rightly skewed. Such distinction is consistent with Figure 5.4, where the skewness (third central moment) at x_2 is negative for Gaussian noise, and positive for other two noises.

In this example, we analyze a one-dimensional linear parabolic SPDE with relatively long evolution time. Here solving moment equations is only practical for the second moment. We resort to Monte Carlo simulation to produce other reference solutions. Our observations are roughly parallel to the previous example. The truncation error of the second moment converges at rates predicted by Theorem 5.1. Higher moments and empirical distributions are more difficult to characterize, and highly depend on the type of underlying randomness.

Example 5.3.3 (Passive scalar equation)**.** We move on to two-dimensional linear transport type SPDE. Consider the following distribution-free passive scalar equa-



(a) Gaussian noise

(b) Uniform and Beta($\frac{1}{2}, \frac{1}{2}$) noise

Figure 5.5: Example 5.3.2: Normalized histograms of $u_{N,K}(5, x_2)$ out of 10^7 i.i.d samples. In the left panel, we take $N = 4, 8$ and $K = 4$. Values outside $[-1, 1]$ are discarded, and the black dashed line represents the normalized histogram of Monte Carlo simulation with 10^6 samples. In the right panel, we take $N = 10, 12$ and $K = 4$. Number of bins is 10^3.

tion, driven by two independent noises $\dot{\mathcal{N}}_1(t)$ and $\dot{\mathcal{N}}_2(t)$, and equipped with periodic boundary condition on $\Gamma = [0, 2\pi]^2$.

$$\begin{aligned} \frac{\partial u}{\partial t}(t, x, y) &= \frac{1}{2}(\mathcal{M}_1^2 + \mathcal{M}_2^2)u + \mathcal{M}_1 u \diamond \dot{\mathcal{N}}_1(t) + \mathcal{M}_2 u \diamond \dot{\mathcal{N}}_2(t), \\ u(0, x, y) &= \sin(2x)\sin(y), \end{aligned} \tag{5.33}$$

where

$$\mathcal{M}_1 = \cos(x + y)\Big(\frac{\partial}{\partial x} - \frac{\partial}{\partial y}\Big), \quad \mathcal{M}_2 = \sin(x + y)\Big(\frac{\partial}{\partial x} - \frac{\partial}{\partial y}\Big).$$

For Gaussian noise, (5.33) reduces to the passive scalar equation in Stratonovich version [71, 116].

$$du(t, x, y) = \mathcal{M}_1 u \circ dW_1(t) + \mathcal{M}_2 u \circ dW_2(t). \tag{5.34}$$

Since there are two driving noises, we need to introduce the Cartesian product of multi-index sets:

$$\mathcal{J}^2 := \{\alpha = (\alpha^1, \alpha^2) : \alpha^1, \alpha^2 \in \mathcal{J}\}, \quad \mathcal{J}^2_{N,K} := \{\alpha = (\alpha^1, \alpha^2) : \alpha^1, \alpha^2 \in \mathcal{J}_{N,K}\}.$$

The polynomial chaos basis functions are

$$\Phi_\alpha := \prod_{k=1}^{\infty} \varphi_{\alpha^1_k}(\xi^1_k) \prod_{k=1}^{\infty} \varphi_{\alpha^2_k}(\xi^2_k).$$

Under the extended nomenclature, the gPC expansion and the truncated solution are still defined as

$$u = \sum_{\alpha \in \mathcal{J}^2} u_\alpha \Phi_\alpha, \quad u_{N,K} := \sum_{\alpha \in \mathcal{J}^2_{N,K}} u_\alpha \Phi_\alpha.$$

It is easy to verify that $\mathcal{M}_1$ and $\mathcal{M}_2$ commute with each other, so that they also commute with $\frac{1}{2}(\mathcal{M}_1^2 + \mathcal{M}_2^2)$. By Remark 5.4, $K = 1$ is enough for the truncation, and we can only adjust the value of N. In fact, for Gaussian noise, we are able to work out the analytical solution based on tracing back characteristic lines [79] of (5.34). $u(T, x, y)$ has the following representation

$$u(T, x, y) = u_0(X_{x,y}(0), Y_{x,y}(0)), \tag{5.35}$$

where $X_{x,y}(t)$ and $Y_{x,y}(t)$ satisfy the system of backward (characteristic) SODEs

$$\begin{aligned} dX_{x,y}(t) &= \cos(X_{x,y} + Y_{x,y})\overleftarrow{dW_1}(t) + \sin(X_{x,y} + Y_{x,y})\overleftarrow{dW_2}(t), \quad t \in [0, T] \\ dY_{x,y}(t) &= -\cos(X_{x,y} + Y_{x,y})\overleftarrow{dW_1}(t) - \sin(X_{x,y} + Y_{x,y})\overleftarrow{dW_2}(t), \quad t \in [0, T] \\ X_{x,y}(T) &= x, \quad Y_{x,y}(T) = y. \end{aligned} \tag{5.36}$$

The definition of backward Itô integral $\overleftarrow{dW}(t)$ can also be found in [79]. Summing the two equations, we realize that $X_{x,y}(t) + Y_{x,y}(t)$ is constant over time. Therefore

$$\begin{aligned} X_{x,y}(0) &= x - \cos(x+y)W_1(T) - \sin(x+y)W_2(T), \\ Y_{x,y}(0) &= y + \cos(x+y)W_1(T) + \sin(x+y)W_2(T). \end{aligned}$$

Then the analytical solution is

$$\begin{aligned} u(T,x,y) = &\sin(2(x - \cos(x+y)W_1(T) - \sin(x+y)W_2(T))) \\ &\cos(y + \cos(x+y)W_1(T) + \sin(x+y)W_2(T)). \end{aligned} \tag{5.37}$$

Notice that the distribution of u is again supported in $[-1,1]$, and u just depends on $W_1(T) = \sqrt{T}\xi_1^1$ and $W_2(T) = \sqrt{T}\xi_1^2$. Monte Carlo sampling of (5.37) is trivial. Moments of $u(T,x,y)$ can be computed very accurately through Gauss-Hermite quadrature rule. We pick 50 quadrature points in each dimension.

We employ Fourier collocation method with $M = 64$ collocation points in each dimension for the spatial discretization of the propagator system. Equidistant collocation points are denoted by $x_i = y_i = \frac{2\pi(i-1)}{M}$. The propagator system is then computed up to $T = 0.2$ with time step size $\delta t = 5 \times 10^{-4}$. The N-version convergence of mean square truncation error is displayed in Figure 5.6. We plot values of $\mathbb{E}[\|u_{N,1}(0.2,\cdot,\cdot) - u(0.2,\cdot,\cdot)\|_{l^2}^2]$ in logarithm scale for $N = 1, \cdots, 10$, where the discrete L^2 norm is

$$\|v\|_{l^2}^2 := \frac{4\pi^2}{M^2}\sum_{i=1}^{M}\sum_{j=1}^{M}(v(x_i,y_j))^2.$$

The convergence rate is clearly exponential.

Next we fix $N = 8$ and pay attention to higher moments. Figure 5.7 presents contour plots for the third and fourth central moments of $u_{8,1}(0,2,\cdot,\cdot)$. For Gaussian noise, we also provide third and fourth central moments of $u(0.2,\cdot,\cdot)$ using

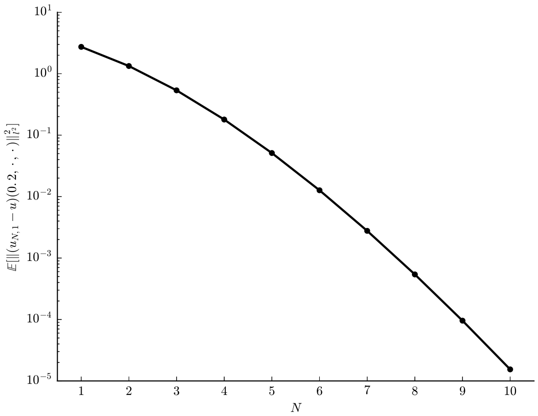
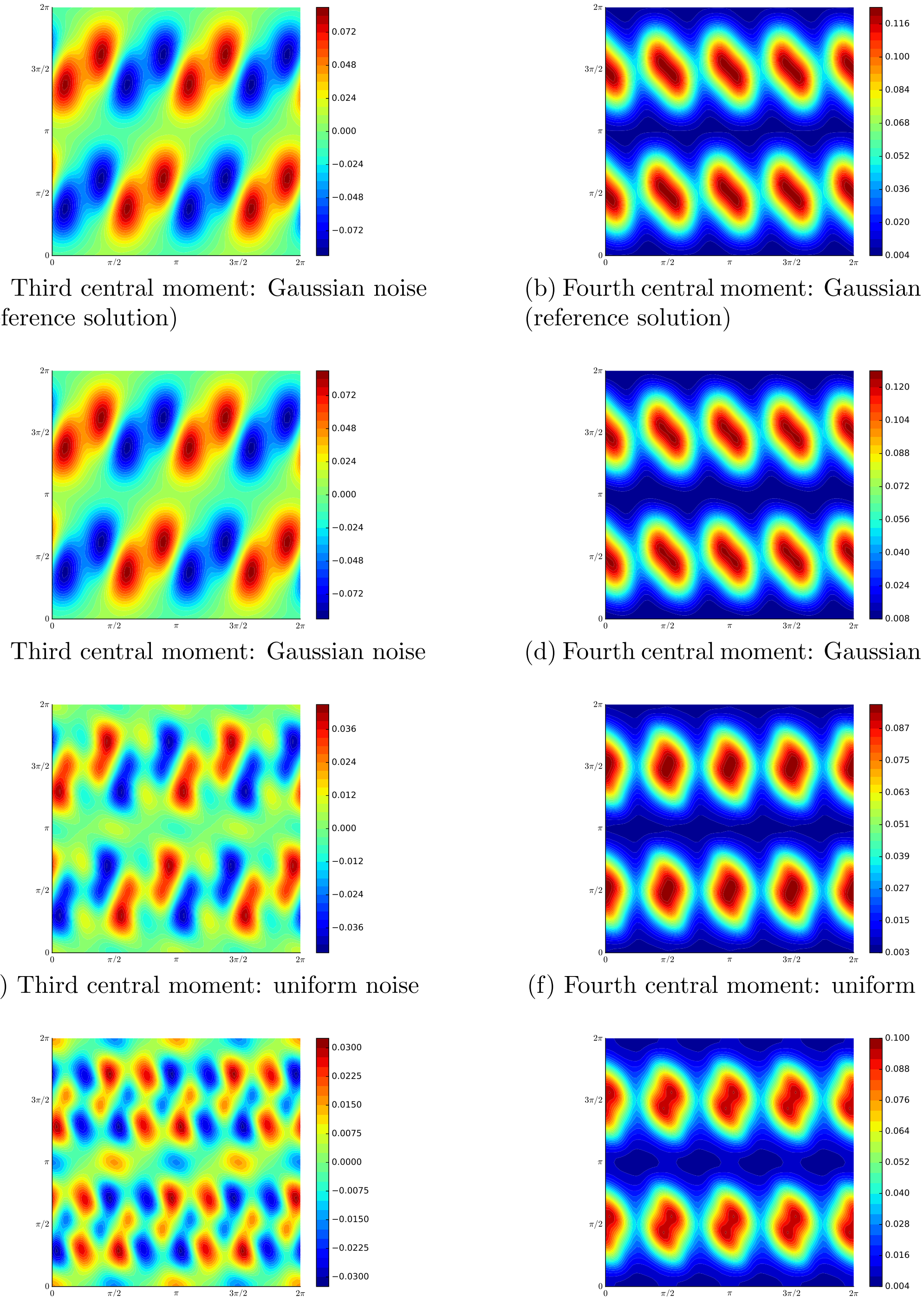


Figure 5.6: Example 5.3.3: Semi-log plot of mean square truncation error $\mathbb{E}[\|u_{N,1}(0.2,\cdot,\cdot)-u(0.2,\cdot,\cdot)\|_{l^2}^2]$ for $N=1,\cdots,10$.

Gauss-Hermite quadrature. The agreement between the truncated solution and the reference solution is quite satisfactory. For uniform noise and Beta$(\frac{1}{2},\frac{1}{2})$ noise, the corresponding contour plots have different patterns, especially for the third central moment.

We also detect the impact of driving noise by checking empirical distributions. In Figure 5.8 we demonstrate normalized histograms of $u_{N,1}$ at the collocation point (x_6,y_6) out of 10^7 samples. For Gaussian noise, we choose $N=4,8$, together with the reference distribution generated by 10^7 samples of (5.37). Samples outside $[-1,1]$ are discarded. Similar to Figure 5.8, $u_{8,1}$ outperforms $u_{4,1}$ in approximating the highly rightly skewed true distribution. For uniform noise and Beta$(\frac{1}{2},\frac{1}{2})$ noise, we consider $N=8,10$. The empirical density functions of $u_{8,1}$ and $u_{10,1}$ are mostly overlapping, so that they can be thought as credible approximations. Once again we notice the fact that different driving noises lead to strikingly different empirical density profiles.

Example 5.3.4 (Stochastic Burgers equation)**.** We consider the stochastic Burgers equation in Example 5.1.2. The space region is $D=[0,1]$ with periodic boundary



(a) Third central moment: Gaussian noise (reference solution)

(b) Fourth central moment: Gaussian noise (reference solution)

(c) Third central moment: Gaussian noise

(d) Fourth central moment: Gaussian noise

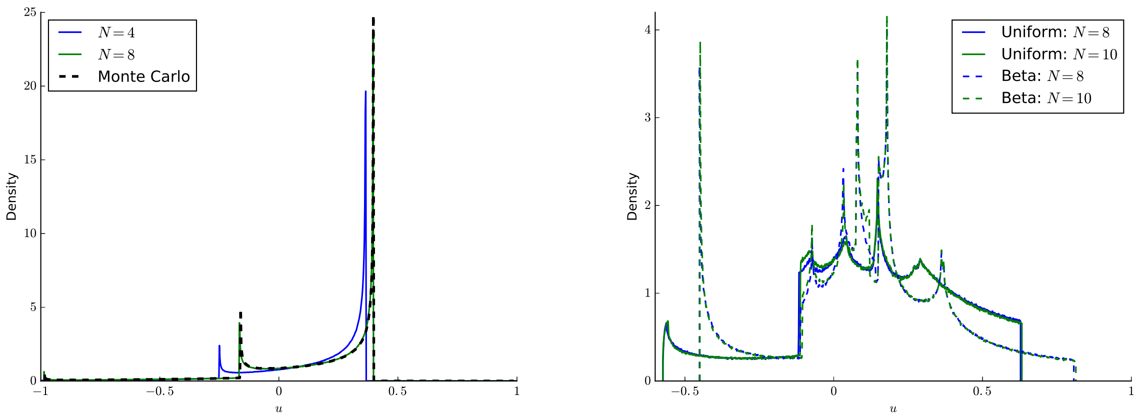
(e) Third central moment: uniform noise

(f) Fourth central moment: uniform noise

(g) Third central moment: Beta noise

(h) Fourth central moment: Beta noise

Figure 5.7: Example 5.3.3: Third and fourth central moments of $u_{8,1}(0.2, \cdot, \cdot)$. First two plots are reference solutions with Gaussian noise. 30 equally spaced contour levels are used for all plots.



(a) Gaussian noise

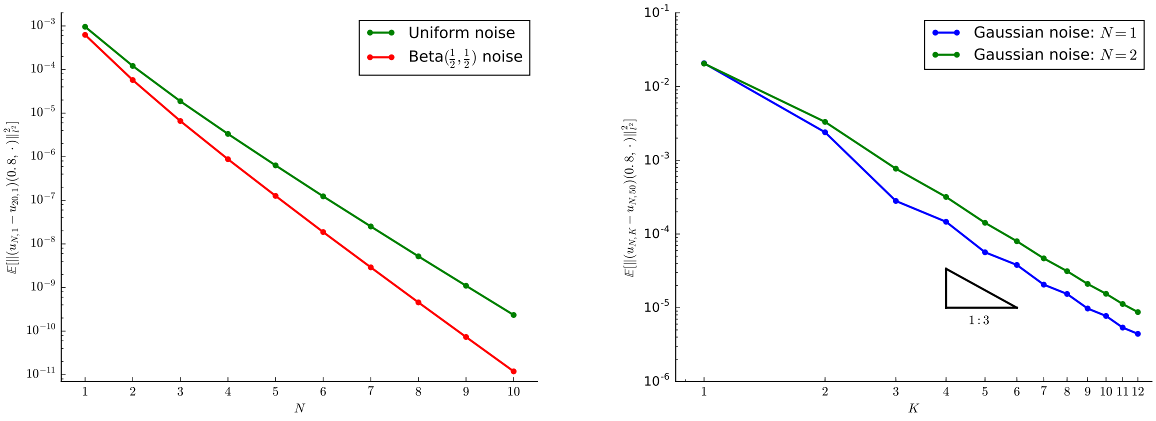
(b) Uniform and Beta$(\frac{1}{2}, \frac{1}{2})$ noise

Figure 5.8: Example 5.3.3: Normalized histograms of $u_{N,1}(0.2, x_6, y_6)$ out of 10^7 i.i.d samples. In the left panel, we take $N = 4, 8$. Values outside $[-1, 1]$ are discarded, and the black dashed line represents the normalized histogram of 10^7 Monte Carlo samples of (5.37). In the right panel, we take $N = 8, 10$. Number of bins is 10^3.

condition. The parameters are $\mu = 0.005$ and $\sigma(x) = \frac{1}{2}\cos(4\pi x)$, and the initial data is $u_0(x) = \frac{1}{2}(e^{(2\pi x)} - 1.5)\sin(2\pi(x + 0.37))$. We apply the Fourier collocation method with $M = 128$ collocation points for spatial discretization. The end time is set to be $T = 0.8$ with Runge-Kutta time step size $\delta t = 10^{-3}$.

For such additive noise, Monte Carlo simulation is well suited for any type of distribution. We generate sample paths of $\dot{\mathcal{N}}(t)$ by truncating the infinite sum up to $K = 50$. For a fixed sample path, we solve the resulting deterministic Burgers equation using Fourier collocation method and fourth order Runge-Kutta time stepping, with $M = 128$ collocation points and time step size $\delta t = 10^{-3}$. Moments and empirical distributions of Monte Carlo samples will be chosen as reference solutions. We take 10^6 sample paths.

The convergence of mean square truncation error is given in Figure 5.9. We again plot $\mathbb{E}[\|u_{N,K}(0.8, \cdot) - u_{20,K}(0.8, \cdot)\|_{l^2}^2]$ to represent N-version convergence, and $\mathbb{E}[\|u_{N,K}(0.8, \cdot) - u_{N,50}(0.8, \cdot)\|_{l^2}^2]$ to represent K-version convergence. Due to nonlinearity, these truncation errors rely on the underlying randomness. For the N-version convergence, we only plot results with uniform and Beta$(\frac{1}{2}, \frac{1}{2})$ noise as the interaction coefficients of Hermite polynomials grow exponentially with respect to N, causing the numerical computation to blow up for large N. For the K-version convergence, the plots are nearly identical for three noises, so that we only present the plot with Gaussian noise. We emphasize that the numerical convergence rate is still exponential with respect to N and cubic with respect to K, even though Theorem 5.1 is only proved for the linear case.

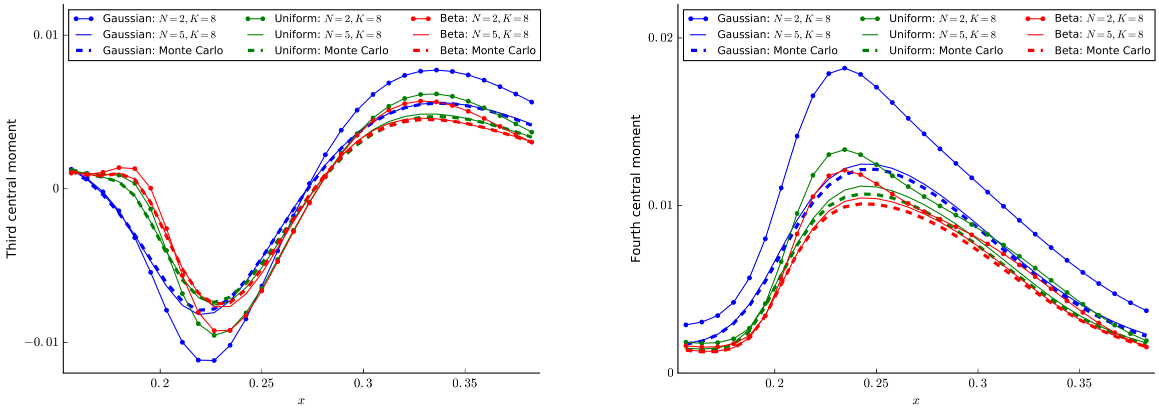


(a) Convergence with respect to N

(b) Convergence with respect to K

Figure 5.9: Example 5.3.4: Plots of mean square truncation error in discrete l^2 norm with respect to N and K. Left panel shows the semi-log plot of $\mathbb{E}[\|u_{N,1}(0.8,\cdot) - u_{20,1}(0.8,\cdot)\|_{l^2}^2]$ versus N with uniform and Beta$(\frac{1}{2}, \frac{1}{2})$ noise for $N = 1, \cdots, 10$. Right panel shows the log-log plot of $\mathbb{E}[\|u_{N,K}(5,\cdot) - u_{N,50}(5,\cdot)\|_{l^2}^2]$ versus K with Gaussian noise for $N = 1, 2$ and $K = 1, \cdots, 12$.

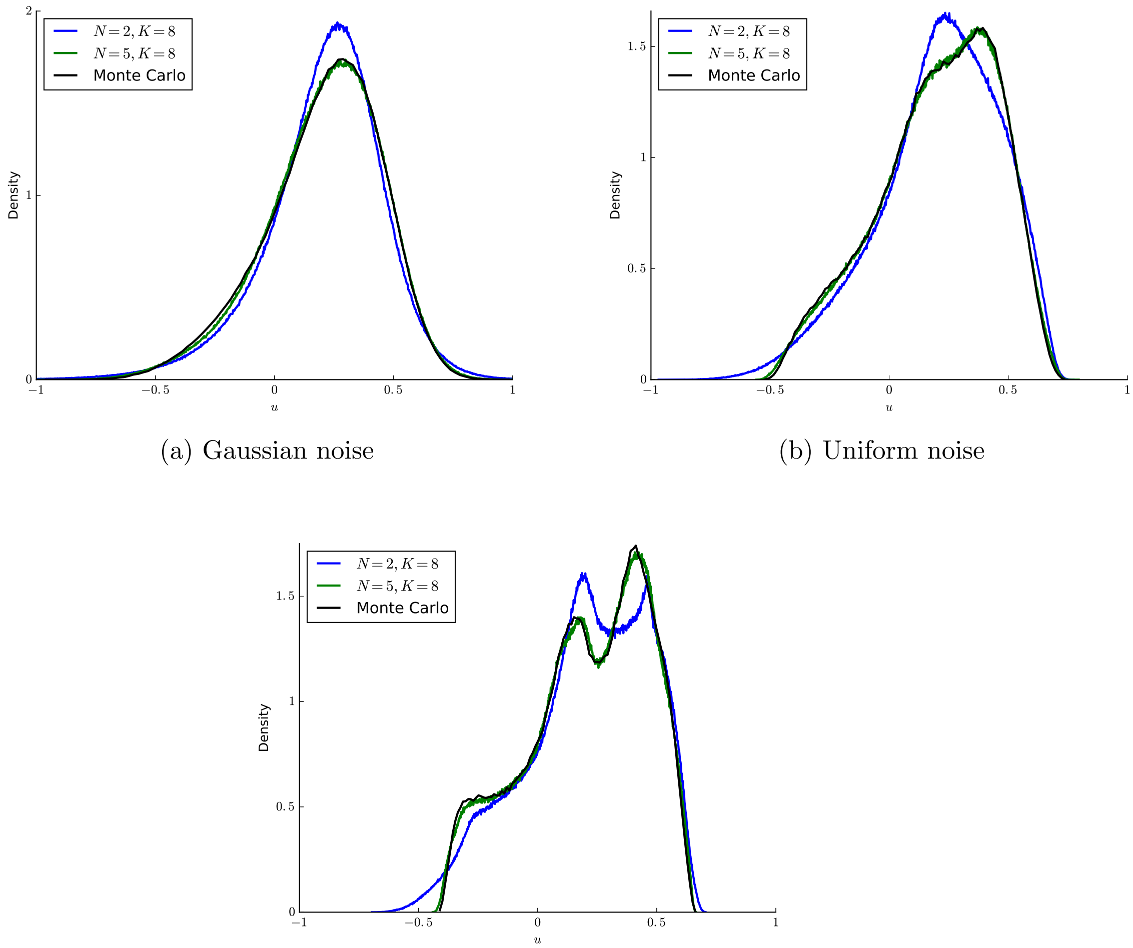
Then we fix $K = 8$. Third and fourth central moments of $u_{2,8}(0.8,\cdot)$ and $u_{5,8}(0.8,\cdot)$ are drawn in Figure 5.10. As the profiles with different noises are close to each other, we only show the zoomed-in view between the 21-st collocation point and the 50-th collocation point. Central moments of Monte Carlo solution are also plotted for comparison. We note that for all noises, $u_{2,8}$ results in inaccurate approximations, and $u_{5,8}$ leads to much better performance. The plots of empirical density functions of $u_{2,8}$ and $u_{5,8}$ out of 10^7 samples at x_{30} are provided in Figure 5.11. From the figure we can also see how $u_{5,8}$ is superior to $u_{2,8}$ in agreeing with the reference distributions.



(a) Third central moment

(b) Fourth central moment

Figure 5.10: Example 5.3.4: Third and fourth central moments of $u_{N,K}(0.8, \cdot)$ between x_{21} and x_{50}. We take $N = 2, 5$ and $K = 8$. Dashed lines are reference solutions computed by Monte Carlo simulation with 10^6 samples.



(a) Gaussian noise

(b) Uniform noise

(c) Beta($\frac{1}{2}, \frac{1}{2}$) noise

Figure 5.11: Example 5.3.4: Normalized histograms of $u_{N,K}(0.8, x_{30})$ out of 10^7 i.i.d samples. We take $N = 2, 5$ and $K = 8$. Black dashed line represents the normalized histogram of Monte Carlo simulation with 10^6 samples. Number of bins is 10^3.

Conclusion

This dissertation is mainly based on the work in [13, 12]. In Part I, we construct high order entropy stable DG type methods for systems of hyperbolic conservation laws. These DG methods can be stable with respect to an arbitrary entropy function. Therefore we circumvent the limitations of the Jiang-Shu L^2-stability result for the classic DG method. The methodology is based on quadrature rules, and applies from one-dimensional Gauss-Lobatto nodes all the way to general set of nodes on multi-dimensional simplicial meshes. The entropy stability is guaranteed by three main ingredients:

1. Discrete operators with the summation-by-parts property. This is straightforward if $\mathcal{N}_{P,k}$ (dimension of modal space) is the same as $\mathcal{N}_{Q,k}$ (dimension of nodal space). In the general case, we need to work out the difference matrices, which are given in equations (2.24) and (3.6).
2. Flux differencing technique , i.e., a high order linear combination of entropy conservative fluxes that ensures entropy balance within an element. The same goal can also be achieved by the "brute force" type method in Section 3.5.
3. Entropy stable boundary penalty term. For Gauss-Lobatto type nodes, simply inserting entropy stable fluxes at element interfaces is enough. For general set of nodes, some extra effort is required, and two possible boundary treatment approaches are established in Section 3.3 and 3.4.

Our entropy stable DG framework has great flexibility in that it enables reflecting wall boundary condition (Section 2.4), generalization to convection-diffusion equations (Section 2.5 and Section 3.6), and the transformation between nodal and modal formulation (Section 3.7). For Gauss-Lobatto type nodes, we can also impose entropy stable bound-preserving limiter and (one-dimensional scalar) TVD/TVB limiter. On the other hand, the main advantage non Gauss-Lobatto type nodes is their better algebraic accuracy with smaller degrees of freedom.

We perform a large number of numerical tests whose results are comparable to other existing schemes. In smooth test problems, the numerical orders of convergence are usually suboptimal on Gauss-Lobatto type nodes. We are able to recover optimal convergence by using the method on general set of nodes. In discontinuous test problems, our scheme shows the potential of better robustness (Example 1.8.5 and Examp 1.8.6) and computing physically correct solution (Example 1.8.4). However, for problems with strong shocks (Example 2.6.5 and Example 2.6.6), only entropy stabilization is not enough. The numerical solution profiles contain evident oscillations. In our future study, we would like to research further on the combination of entropy stabilization and other types of stabilization mechanism, such as bound-preserving limiters on general set of nodes, TVD/TVB limiters for systems, and the introduction of artificial viscosity.

In Part II, we explore the polynomial chaos expansion approach method for distribution-free SPDEs. We generalize the definition of Wick product and Skorokhod integral to arbitrary driving noise. Then the resulting SPDEs are not limited to Gaussian or Lévy randomness. More importantly, for linear SPDEs, the propagator system, and even the first two moments or the solution, are the same for different noises. The computational burden of solving the propagator system is purely offline. The only on-line work is post-processing. However, the propagator system of nonlinear SPDEs changes with noise as interaction terms come into play.

Analysis of the mean square truncation error is carried out for linear problems. We prove exponential convergence with respect to polynomial order, and cubic convergence with respect to the number of random variables. The cubic rate arises from repeated integration-by-parts and special properties of the orthonormal basis

$\{m_k(t)\}_{k=1}^{\infty}$. We need to assume trigonometric basis or Legendre basis.

We conduct systematic investigation on numerical results of linear and nonlinear SPDEs with different driving noises. Numerical rates of convergence are consistent with our theoretical analysis. Higher moments and density function can also be approximated effectively with sufficiently many expansion terms. However, we recognize some drawbacks and unsolved problems, which gives hints on our future research.

1. To the best of our knowledge, the limiting procedure of distribution-free Skorokohd integral is unclear. Theorem 4.3 is only for deterministic processes, and Theorem 4.4 is only for Gaussian (and Lévy) noise. Further work is required for better understanding of the distribution-free stochastic analysis.
2. We do not focus on long time integration in this paper, but the exponential growth of error with respect to time is seen both theoretically and numerically. Proper techniques should be devised to mitigate the impact of time evolution. We remark that direct generalization of the multi-stage methodology in [68, 67, 118] is incorrect as the driving process $\mathcal{N}(t)$ may not have independent increments.
3. The propagator system usually consists of PDEs of the same type but with different data. It can be expected that for a large number of expansion terms, the application of reduced basis method [85] may significantly reduce the computational cost while maintaining desired accuracy.

Appendix

Appendix A

Two-rarefaction Approximation

For Euler equations, the solution of the Riemann problem consists of three characteristic waves. The left wave and right wave are either rarefaction fans or shocks, and the middle wave is a contact discontinuity. The pressure is continuous across the contact discontinuity and thus constant in the middle region, denoted by p^*. We find the exact value of p^* by solving the following equation.

$$\varphi(p^*, p_L, \rho_L) + \varphi(p^*, p_R, \rho_R) + w_R - w_L = 0 \tag{A.1}$$

where

$$\varphi(p^*, p, \rho) = \begin{cases} \varphi_r(p^*, p, \rho) = \frac{2a}{\gamma-1}((\frac{p^*}{p})^{(\gamma-1)/2\gamma} - 1) & \text{if } p^* \leq p \text{ (rarefaction wave)} \\ \varphi_s(p^*, p, \rho) = \frac{p^*-p}{\sqrt{(\rho((\gamma-1)p^*+(\gamma+1)p)/2}} & \text{if } p^* > p \text{ (shock wave)} \end{cases} \tag{A.2}$$

and $a = \sqrt{\gamma p/\rho}$ is the sound speed. φ is a continuous, strictly increasing and concave function of p^* (see [99]), so that we can use Newton-Raphson iteration to find the unique root. Once we have p^*, the leftmost and rightmost wave speeds are given by

$$\lambda_L = w_L - a_L q(p^*, p_L), \quad \lambda_R = w_R + a_R q(p^*, p_R) \tag{A.3}$$

such that

$$q(p^*, p) = \begin{cases} 1 & \text{if } p^* \leq p \\ \sqrt{1 + \frac{\gamma+1}{2\gamma}(\frac{p^*}{p} - 1)} & \text{if } p^* > p \end{cases} \tag{A.4}$$

The following inequality is proved in [44].

Theorem A.1. *If $1 < \gamma \leq 5/3$, $\varphi_s(p^*, p, \rho) \geq \varphi_r(p^*, p, \rho)$ for $p^* > p$.*

Proof. Substitute $x = (p^*/p)^{(\gamma-1)/2\gamma}$. Then

$$\varphi_r(p^*, p, \rho) = \frac{2a}{\gamma - 1}(x - 1), \quad \varphi_s(p^*, p, \rho) = \frac{a}{\gamma}\frac{x^{2\gamma/(\gamma-1)} - 1}{\sqrt{(\gamma-1)/2\gamma + ((\gamma+1)/2\gamma)x^{2\gamma/(\gamma-1)}}}$$

Let $\alpha = 2\gamma/(\gamma - 1) \in [5, \infty)$. We need to show that

$$(\frac{x^\alpha - 1}{x - 1})^2 \geq \alpha + \alpha(\alpha - 1)x^\alpha, \quad \text{for } x > 1$$

Rearranging the term yields

$$(\frac{x^\alpha - 1}{x - 1} - \frac{1}{2}\alpha(\alpha - 1)(x - 1))^2 \geq \alpha^2 + \frac{1}{4}\alpha^2(\alpha - 1)^2(x - 1)^2 \tag{A.5}$$

By Taylor's expansion

$$\frac{x^\alpha - 1}{x - 1} \geq \alpha + \frac{1}{2}\alpha(\alpha - 1)(x - 1) + \frac{1}{6}\alpha(\alpha - 1)(\alpha - 2)(x - 1)^2$$

Inserting this inequality, we have

$$\begin{aligned}(\frac{x^\alpha - 1}{x - 1} - \frac{1}{2}\alpha(\alpha - 1)(x - 1))^2 &\geq (\alpha + \frac{1}{6}\alpha(\alpha - 1)(\alpha - 2)(x - 1)^2)^2 \\ &\geq \alpha^2 + \frac{1}{3}\alpha^2(\alpha - 1)(\alpha - 2)(x - 1)^2\end{aligned}$$

Since $\alpha \geq 5$, $(\alpha - 2)/3 \geq (\alpha - 1)/4$ and so (A.5) is valid. We note that in most physical applications γ does fall into the range $(1, 5/3]$ ($5/3$ for monatomic gas and $7/5$ for diatomic gas). □

Invoking Newton-Raphson iteration during all flux computations can be time-consuming. The two-rarefaction approximation assumes that the left wave and the right wave are both rarefaction waves, and provides an explicit formula of p^*, λ_L and λ_R. Thanks to Theorem A.1, the approximated wave speeds bound the true wave speeds. Then we can take these wave speeds to construct entropy stable HLL flux (or local Lax-Friedrichs flux).

Theorem A.2. *The two-rarefaction approximation solves the equation*

$$\varphi_r(p^*_{tr}, p_L, \rho_L) + \varphi_r(p^*_{tr}, p_R, \rho_R) + w_R - w_L = 0 \tag{A.6}$$

The explicit solution is

$$p^*_{tr} = \left(\frac{a_L + a_R + (\gamma-1)(w_L - w_R)/2}{a_L/p_L^{(\gamma-1)/2\gamma} + a_R/p_R^{(\gamma-1)/2\gamma}}\right)^{2\gamma/(\gamma-1)} \tag{A.7}$$

The approximated wave speeds are

$$\lambda_{tr,L} = w_L - a_L q(p^*_{tr}, p_L), \quad \lambda_{tr,R} = w_R + a_R q(p^*_{tr}, p_R) \tag{A.8}$$

Then $q^*_{tr} \geq q^*$, $\lambda_{tr,L} \leq \lambda_L$ and $\lambda_{tr,R} \geq \lambda_R$.

Proof. By Theorem A.1, $\varphi_r \leq \varphi$ for all $p > 0$. As both φ and φ_r are strictly increasing, $p^*_{tr} \geq p^*$. q is also an increasing function of p^*. Hence $\lambda_{tr,L} \leq \lambda_L$ and $\lambda_{tr,R} \geq \lambda_R$. □

The same argument also works for shallow water equations. We will omit the details, only giving the key inequality without proof. The exact Riemann solver reduces to the equation

$$\varphi(h^*, h_L) + \varphi(h^*, h_R) + w_R - w_L = 0 \tag{A.9}$$

where

$$\varphi(h^*, h) = \begin{cases} \varphi_r(h^*, h) = 2(\sqrt{gh^*} - \sqrt{gh}) & \text{if } h^* \leq h \\ \varphi_s(h^*, h) = (h^* - h)\sqrt{\frac{1}{2}g\frac{h^*+h}{h^*h}} & \text{if } h^* > h \end{cases} \tag{A.10}$$

When $h^* > h$, it is easy to prove that

$$\varphi_s(h^*, h) \geq \varphi_r(h^*, h) \tag{A.11}$$

Therefore two-rarefaction approximation will produce proper wave speeds.

Appendix B

Bound-preserving Limiter

The bound-preserving limiter is designed for fully discrete schemes. For the sake of simplicity we shall assume uniform grid size Δx and uniform time step Δt. Let $\lambda := \Delta t/\Delta x$. We start with the first order method for one-dimensional conservation laws, using first order finite volume spatial discretization (1.29), and Euler forward time stepping.

$$\mathbf{u}_i^{n+1} = H(\mathbf{u}_{i-1}^n, \mathbf{u}_i^n, \mathbf{u}_{i+1}^n; \lambda) := \mathbf{u}_i^n - \lambda(\widehat{\mathbf{f}}(\mathbf{u}_i^n, \mathbf{u}_{i+1}^n) - \widehat{\mathbf{f}}(\mathbf{u}_{i-1}^n, \mathbf{u}_i^n)). \tag{B.1}$$

Definition B.1. *Scheme* (B.1) *is called bound-preserving if for some* $\lambda_0 > 0$, $\mathbf{u}_{i-1}^n, \mathbf{u}_i^n, \mathbf{u}_{i+1}^n \in \Omega$ *and* $\lambda \le \lambda_0$ *will imply* $\mathbf{u}_i^{n+1} = H(\mathbf{u}_{i-1}^n, \mathbf{u}_i^n, \mathbf{u}_{i+1}^n; \lambda) \in \Omega$.

Theorem B.1. *In the scalar case, if* $\widehat{f}$ *is monotone and Lipschitz continuous of both arguments, and* $\Omega = [m, M]$ *for* $m, M \in \mathbb{R}$, *then* (B.1) *is bound-preserving (usually called maximum-principle-preserving).*

Proof. Since $\widehat{f}$ is monotone, H is non-decreasing with respect to u_i^{n-1} and u_i^{n+1}. Let L be the Lipschitz constant of $\widehat{f}$. Then H is also a non-decreasing function of u_i^n provided that $\lambda \le \frac{1}{2L}$. Now if $u_{i-1}^n, u_i^n, u_{i+1}^n \in [m, M]$,

$$u_i^{n+1} \ge H(m, m, m; \lambda) = m, \quad u_i^{n+1} \le H(M, M, M; \lambda) = M.$$

We see that H is bound-preserving with $\lambda_0 = \frac{1}{2L}$. □

Theorem B.2. *For general systems, if the exact Riemann solver is bound-preserving (e.g. no dry bed for shallow water equations or no vacuum for Euler equations), and* $\widehat{\mathbf{f}}$ *is Godunov flux or HLL flux with suitable wave speed estimates, then* (B.1) *is bound-preserving.*

Proof. Since Ω is a convex set, the HLL Riemann solver, as an average of the exact

Riemann solver, is also bound-preserving. When λ is small enough such that waves of Riemann solvers at different cell interfaces do not intersect, the first order method (B.1) can be regarded as another averaging procedure of Riemann solvers. Hence it is also bound-preserving. □

High order spatial discretization is generally not bound-preserving in the sense of Definition B.1. However, we can still prove a weaker argument, i.e. the cell average at next time step is in Ω. The next theorem paves the path for high order bound-preserving limiter. It can be formulated in a more general manner, but we stay within the context of entropy stable DGSEM.

Theorem B.3. *For the fully discrete version entropy stable DGSEM* (1.77)

$$\overrightarrow{\mathbf{u}_i^{n+1}} = \overrightarrow{\mathbf{u}_i^{n}} - 2\lambda\Big(2\mathbf{D}\circ\mathbf{F}_S(\overrightarrow{\mathbf{u}_i^{n}},\overrightarrow{\mathbf{u}_i^{n}})\,\overrightarrow{\mathbf{1}} - \mathbf{M}^{-1}\mathbf{B}(\overrightarrow{\mathbf{f}_i^{n}} - \overrightarrow{\mathbf{f}_i^{n,*}})\Big) \tag{B.2}$$

Assume that the underlying first order method is bound preserving under the CFL condition $\lambda \le \lambda_0$. If $\mathbf{u}_{i,j}^n \in \Omega$ for each $1 \le i \le N$ and $0 \le j \le k$, then $\overline{\mathbf{u}}_i^{n+1} \in \Omega$ provided that $\lambda \le \frac{\omega_0}{2}\lambda_0$.

Proof. By local conservation (1.79),

$$\begin{aligned}
\overline{\mathbf{u}}_i^{n+1} &= \overline{\mathbf{u}}_i^{n} - \lambda(\widehat{\mathbf{f}}(\mathbf{u}_{i,k}^n,\mathbf{u}_{i+1,0}^n) - \widehat{\mathbf{f}}(\mathbf{u}_{i-1,k}^n,\mathbf{u}_{i,0}^n)) \\
&= \sum_{j=0}^{k}\frac{\omega_j}{2}\mathbf{u}_{i,j}^n - \lambda(\widehat{\mathbf{f}}(\mathbf{u}_{i,k}^n,\mathbf{u}_{i+1,0}^n) - \widehat{\mathbf{f}}(\mathbf{u}_{i-1,k}^n,\mathbf{u}_{i,0}^n)) \\
&= \sum_{j=1}^{k-1}\frac{\omega_j}{2}\mathbf{u}_{i,j}^n + \frac{\omega_0}{2}H\Big(\mathbf{u}_{i-1,k}^n,\mathbf{u}_{i,0}^n,\mathbf{u}_{i,k}^n;\frac{2\lambda}{\omega_0}\Big) + \frac{\omega_0}{2}H\Big(\mathbf{u}_{i,0}^n,\mathbf{u}_{i,k}^n,\mathbf{u}_{i+1,0}^n;\frac{2\lambda}{\omega_0}\Big)
\end{aligned}$$

If $\lambda \le \frac{\omega_0}{2}\lambda_0$, the last two terms are in Ω. Then $\overline{\mathbf{u}}_i^{n+1} \in \Omega$ as it is a convex combination of elements in Ω. □

The bound-preserving limiter is a simple linear scaling procedure $\widetilde{\mathbf{u}}_{i,j}^n = \overline{\mathbf{u}}_i^n + \theta_i^n(\mathbf{u}_{i,j}^n - \overline{\mathbf{u}}_i^n)$ to enforce $\widetilde{\mathbf{u}}_{i,j}^n \in \Omega$. It can be enforced as long as $\overline{\mathbf{u}}_i^n \in \Omega$. Roughly speaking, for each $0 \leq j \leq k$ we compute

$$\theta_{i,j}^n := \max\{s \in [0,1] : \overline{\mathbf{u}}_i^n + s(\mathbf{u}_{i,j}^n - \overline{\mathbf{u}}_i^n) \in \Omega\}$$

Then we simply let $\theta_i^n := \min\limits_{0\leq j\leq k} \theta_{i,j}^n$. A combination of mathematical induction and Theorem B.3 tells us that we can apply such limiter at each time step, leading to a robust scheme whose numerical solution never goes out of Ω. For implementation details and the proof that bound-preserving limiter is genuinely high order accurate, one may check the papers by Zhang and Shu [112, 113]. The positivity-preserving limiter is generalized to higher space dimensions in [114].

Finally, the magic of SSP time discretization enables us to go beyond Euler forward time stepping. In this paper, we use the third order SSP Runge-Kutta method. For an ODE system $\frac{d\mathbf{u}}{dt} = L\mathbf{u}$, the three stages at the n-th time step are

$$\mathbf{u}^{(n,1)} = \mathbf{u}^n + \Delta t L(\mathbf{u}^n) \tag{B.3a}$$

$$\mathbf{u}^{(n,2)} = \frac{3}{4}\mathbf{u}^n + \frac{1}{4}(\mathbf{u}^{(n,1)} + \Delta t L(\mathbf{u}^{(n,1)})) \tag{B.3b}$$

$$\mathbf{u}^{n+1} = \frac{1}{3}\mathbf{u}^n + \frac{2}{3}(\mathbf{u}^{(n,2)} + \Delta t L(\mathbf{u}^{(n,2)})) \tag{B.3c}$$

Since it is a convex combination of Euler forward steps, all the previous analyses are still valid.

Appendix C

Quadrature Rules on a Triangle

As indicated in [109], we group the quadrature points into symmetry orbits. The orbit S_3 only includes one point, the barycenter of the triangle. The three points in S_{21} are determined by a single abscissa, and the six points in S_{111} are determined by two abscissas. Table C.1 shows the idea of symmetry orbits. The B-type quadrature rules in Chapter 2 are listed in Table C.2, and the A-type quadrature rules in Chapter 3 are provided in Table C.3. In both tables, the quadrature weights are scaled so that the sum of them is 1.

Table C.1: Symmetry orbits on a triangle.

orbit	barycentric coordinates	# of points
$S_3(\frac{1}{3})$	$(\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$	1
$S_{21}(\alpha)$	permutation of $(\alpha, \alpha, 1-2\alpha)$	3
$S_{111}(\alpha, \beta)$	permutation of $(\alpha, \beta, 1-\alpha-\beta)$	6

Table C.2: B-type quadrature rules on a triangle with $k = 1, 2, 3, 4$.

(a) $k = 1, \mathcal{N}_{Q,k} = 6$

orbit	abscissas	weight
S_{111}	$(0, 0.211324865405187)$	$\frac{1}{6}$

(b) $k = 2, \mathcal{N}_{Q,k} = 10$

orbit	abscissas	weight
S_3	$\frac{1}{3}$	$\frac{9}{20}$
S_{21}	$\frac{1}{2}$	$\frac{1}{10}$
S_{111}	$(0, 0.112701665379258)$	$\frac{1}{24}$

(c) $k = 3, \mathcal{N}_{Q,k} = 18$

orbit	abscissas	weight
S_{111}	$(0, 0.330009478207572)$	0.04045654068298998
S_{111}	$(0, 0.0694318442029737)$	0.0150990148725656
S_{111}	$(0.1870738791912771, 0.5841571139756569)$	$\frac{1}{9}$

(d) $k = 4, \mathcal{N}_{Q,k} = 22$

orbit	abscissas	weight
S_3	$\frac{1}{3}$	0.09109991119771334
S_{21}	$\frac{1}{2}$	0.01853708483394977
S_{21}	0.4384239524408185	0.1247367322897736
S_{21}	0.1394337314154536	0.1054293296208443
S_{111}	$(0, 0.230765344947159)$	0.02053045968042896
S_{111}	$(0, 0.046910077030668)$	0.006601315081001616

Table C.3: A-type quadrature rules on a triangle with $k = 1, 2, 3, 4$.

(a) $k = 1, \mathcal{N}_{Q,k} = 3$

orbit	abscissas	weight
S_{21}	$\frac{1}{6}$	$\frac{1}{3}$

(b) $k = 2, \mathcal{N}_{Q,k} = 6$

orbit	abscissas	weight
S_{21}	0.09157621350977073	0.1099517436553219
S_{21}	0.4459484909159649	0.2233815896780115

(c) $k = 3, \mathcal{N}_{Q,k} = 12$

orbit	abscissas	weight
S_{21}	0.219429982549783	0.171333124152981
S_{21}	0.480137964112215	0.08073108959303098
S_{111}	$(0.1416190159239681, 0.01937172436124079)$	0.04063455979366066

(d) $k = 4, \mathcal{N}_{Q,k} = 16$

orbit	abscissas	weight
S_3	$\frac{1}{3}$	0.1443156076777872
S_{21}	0.1705693077517602	0.1032173705347182
S_{21}	0.4592925882927231	0.09509163426728465
S_{21}	0.05054722831703097	0.03245849762319808
S_{111}	$(0.2631128296346381, 0.008394777409957609)$	0.027230314174435

Appendix D

Interaction Coefficients $B(\alpha, \beta, p)$

We assume that $\{\xi_k\}_{k=1}^{\infty}$ are i.i.d. random variables. Then the interaction coefficient $B(\alpha, \beta, p)$ can be decomposed into

$$B(\alpha, \beta, p) = \frac{\mathbb{E}[\Phi_\alpha \Phi_\beta \Phi_p]}{\alpha!} = \prod_{k=1}^{\infty} \frac{\mathbb{E}[\varphi_{\alpha_k}(\xi_k)\varphi_{\beta_k}(\xi_k)\varphi_{p_k}(\xi_k)]}{\alpha_k!} := \prod_{k=1}^{\infty} b(\alpha_k, \beta_k, p_k). \quad \text{(D.1)}$$

It suffices to compute $b(i, j, l)$ for any $i, j, l \geq 0$. According to orthogonality,

$$\varphi_j(\xi)\varphi_l(\xi) = \sum_{i=0}^{\infty} \frac{\mathbb{E}[\varphi_i(\xi)\varphi_j(\xi)\varphi_l(\xi)]}{i!} \varphi_i(\xi) = \sum_{i=0}^{\infty} b(i, j, l)\varphi_i(\xi). \quad \text{(D.2)}$$

Hence $b(i, j, l)$ is the i-th expansion coefficient of $\varphi_j(\xi)\varphi_l(\xi)$ in terms of $\{\varphi_n(\xi)\}_{n=0}^{\infty}$. In particular, for the three types of noises and corresponding orthogonal polynomials considered in Section 5.3, there are explicit formulas for these expansion coefficients.

Example D.1. For Gaussian noise and Hermite chaos. $\varphi_n(\xi) = He_n(\xi)$. Since

$$He_j(x)He_l(x) = \sum_{r=0}^{\min\{j,l\}} \frac{j!l!}{(j-r)!(l-r)!r!} He_{j+l-2r}(x), \quad \text{(D.3)}$$

we have

$$b(i, j, l) = \begin{cases} \frac{j!l!}{(j-r)!(l-r)!r!} & \text{if } i = j + l - 2r \text{ and } r \leq \min\{i, j\} \\ 0 & \text{otherwise} \end{cases}. \quad \text{(D.4)}$$

Example D.2. For uniform noise and Legendre chaos, $\varphi_n(\xi) = \sqrt{(2n+1)n!}L_n(\xi/\sqrt{3})$. Define

$$\lambda_n := \frac{\Gamma(n + 1/2)}{n!\Gamma(1/2)} = \frac{\prod_{m=0}^{n-1}(m + 1/2)}{n!}.$$

Then the expansion of $L_j(x)L_l(x)$ is

$$L_j(x)L_l(x) = \sum_{r=0}^{\min\{j,l\}} \frac{2(j + l - 2r) + 1}{2(j + l - r) + 1} \frac{\lambda_r \lambda_{i-r} \lambda_{j-r}}{\lambda_{i+j-r}} L_{j+l-2r}(x). \quad \text{(D.5)}$$

Thus

$$b(i,j,l) = \begin{cases} \frac{\sqrt{(2i+1)(2j+1)(2l+1)}}{2(j+l-r)+1}\sqrt{\frac{j!l!}{i!}}\frac{\lambda_r\lambda_{i-r}\lambda_{j-r}}{\lambda_{i+j-r}} & \text{if } i = j+l-2r \text{ and } r \leq \min\{i,j\} \\ 0 & \text{otherwise} \end{cases}. \tag{D.6}$$

Example D.3. For Beta($\frac{1}{2}, \frac{1}{2}$) noise and Chebyshev chaos, $\varphi_n(\xi) = \sqrt{c_n n!}T_n(\xi/\sqrt{2})$ where $c_0 = 1$ and $c_n = 2$ for $n \geq 1$. Since Chebyshev polynomials are essentially cosine functions,

$$T_j(x)T_l(x) = \frac{1}{2}T_{j+l}(x) + \frac{1}{2}T_{|j-l|}(x). \tag{D.7}$$

Thus

$$b(i,j,l) = \begin{cases} 1 & \text{if } i = j, l = 0 \text{ or } i = l, j = 0 \\ \frac{1}{2}\sqrt{\frac{c_j c_l}{c_i}}\sqrt{\frac{j!l!}{i!}} & \text{if } j,l > 0 \text{ and } i = j+l \text{ or } i = |j-l| \\ 0 & \text{otherwise} \end{cases}. \tag{D.8}$$

Here the expansion coefficients have a sparse pattern. For fixed j and l, there are at most two values of i such that $b(i,j,l)$ is nonzero.

In general, we compute $b(i,j,l)$ by matching the monomial coefficients on the both sides of (D.2) (see [119]). Suppose that

$$\varphi_n(\xi) = \sum_{m=0}^{n} P_{m,n}\xi^m.$$

According to (D.2), for $i > j+l$, $b(i,j,l) = 0$, and $\{b(i,j,l) : 0 \leq i \leq j+l\}$ satisfies the following linear system

$$\sum_{i=0}^{j+l} b(i,j,l)P_{m,i} = \sum_{r=\max\{0,i-l\}}^{\min\{i,j\}} P_{r,j}P_{i-r,l}. \tag{D.9}$$

It is easy to solve (D.9) directly as $\{P_{m,n}\}_{m,n=0}^{j+l}$ is a upper triangular matrix. This procedure is applicable to any set of orthogonal polynomials.

Bibliography

[1] R. Abgrall, *A general framework to construct schemes satisfying additional conservation relations. Application to entropy conservative and entropy dissipative schemes*, Journal of Computational Physics, 372 (2018), pp. 640–666.

[2] R. Askey and J. A. Wilson, *Some basic hypergeometric orthogonal polynomials that generalize Jacobi polynomials*, vol. 319, American Mathematical Soc., 1985.

[3] T. J. Barth, *Numerical methods for gasdynamic systems on unstructured meshes*, in An introduction to recent developments in theory and numerics for conservation laws, Springer, 1999, pp. 195–285.

[4] S. Bianchini and A. Bressan, *Vanishing viscosity solutions of nonlinear hyperbolic systems*, Annals of Mathematics, (2005), pp. 223–342.

[5] F. Bouchut, C. Bourdarias, and B. Perthame, *A MUSCL method satisfying all the numerical entropy inequalities*, Mathematics of Computation of the American Mathematical Society, 65 (1996), pp. 1439–1461.

[6] R. H. Cameron and W. T. Martin, *The orthogonal development of non-linear functionals in series of fourier-hermite functionals*, Annals of Mathematics, (1947), pp. 385–392.

[7] M. H. Carpenter, T. C. Fisher, E. J. Nielsen, and S. H. Frankel, *Entropy stable spectral collocation schemes for the Navier–Stokes equations: Discontinuous interfaces*, SIAM Journal on Scientific Computing, 36 (2014), pp. B835–B867.

[8] P. Castillo, B. Cockburn, I. Perugia, and D. Schötzau, *An a priori error analysis of the local discontinuous Galerkin method for elliptic problems*, SIAM Journal on Numerical Analysis, 38 (2000), pp. 1676–1706.

[9] J. Chan, *On discretely entropy conservative and entropy stable discontinuous Galerkin methods*, Journal of Computational Physics, 362 (2018), pp. 346–374.

[10] J. Chan, D. C. Fernandez, and M. H. Carpenter, *Efficient entropy stable Gauss collocation methods*, arXiv preprint arXiv:1809.01178, (2018).

[11] P. Chandrashekar, *Kinetic energy preserving and entropy stable finite volume schemes for compressible Euler and Navier-Stokes equations*, Communications in Computational Physics, 14 (2013), pp. 1252–1286.

[12] T. Chen, B. Rozovskii, and C.-W. Shu, *Numerical solutions of stochastic pdes driven by arbitrary type of noise*, Stochastics and Partial Differential Equations: Analysis and Computations, 7 (2019), pp. 1–39.

[13] T. Chen and C.-W. Shu, *Entropy stable high order discontinuous Galerkin methods with suitable quadrature rules for hyperbolic conservation laws*, Journal of Computational Physics, 345 (2017), pp. 427–461.

[14] E. Chiodaroli, C. De Lellis, and O. Kreml, *Global ill-posedness of the isentropic system of gas dynamics*, Communications on Pure and Applied Mathematics, 68 (2015), pp. 1157–1190.

[15] B. Cockburn, S. Hou, and C.-W. Shu, *The Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws. IV. the multidimensional case*, Mathematics of Computation, 54 (1990), pp. 545–581.

[16] 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*, Journal of Computational Physics, 84 (1989), pp. 90–113.

[17] B. Cockburn and C.-W. Shu, *TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws. II. general framework*, Mathematics of computation, 52 (1989), pp. 411–435.

[18] ——, *The local discontinuous Galerkin method for time-dependent convection-diffusion systems*, SIAM Journal on Numerical Analysis, 35 (1998), pp. 2440–2463.

[19] ——, *The Runge–Kutta discontinuous Galerkin method for conservation laws V: multidimensional systems*, Journal of Computational Physics, 141 (1998), pp. 199–224.

[20] M. G. Crandall and A. Majda, *Monotone difference approximations for scalar conservation laws*, Mathematics of Computation, 34 (1980), pp. 1–21.

[21] J. Crean, J. E. Hicken, D. C. D. R. Fernández, D. W. Zingg, and M. H. Carpenter, *Entropy-stable summation-by-parts discretization of the Euler equations on general curved elements*, Journal of Computational Physics, 356 (2018), pp. 410–438.

[22] G. Da Prato, A. Debussche, and R. Temam, *Stochastic Burgers' equation*, Nonlinear Differential Equations and Applications NoDEA, 1 (1994), pp. 389–402.

[23] C. M. Dafermos, *Hyperbolic conservation laws in continuum physics*, vol. 325, Springer, 2010.

[24] B. J. Debusschere, H. N. Najm, P. P. Pébay, O. M. Knio, R. G. Ghanem, and O. P. Le Mai tre, *Numerical challenges in the use of polynomial chaos representations for stochastic processes*, SIAM journal on scientific computing, 26 (2004), pp. 698–719.

[25] G. Di Nunno, B. K. Øksendal, and F. Proske, *Malliavin calculus for Lévy processes with applications to finance*, vol. 2, Springer, 2009.

[26] B. Engquist and S. Osher, *Stable and entropy satisfying approximations for transonic flow calculations*, Mathematics of Computation, 34 (1980), pp. 45–75.

[27] D. C. D. R. Fernández, P. D. Boom, and D. W. Zingg, *A generalized framework for nodal first derivative summation-by-parts operators*, Journal of Computational Physics, 266 (2014), pp. 214–239.

[28] D. C. D. R. Fernández, J. E. Hicken, and D. W. Zingg, *Simultaneous approximation terms for multi-dimensional summation-by-parts operators*, Journal of Scientific Computing, 75 (2018), pp. 83–110.

[29] T. C. Fisher and M. H. Carpenter, *High-order entropy stable finite difference schemes for nonlinear conservation laws: Finite domains*, Journal of Computational Physics, 252 (2013), pp. 518–557.

[30] T. C. Fisher, M. H. Carpenter, J. Nordström, N. K. Yamaleev, and C. Swanson, *Discretely conservative finite-difference formulations for nonlinear conservation laws in split form: Theory and boundary conditions*, Journal of Computational Physics, 234 (2013), pp. 353–375.

[31] U. S. Fjordholm, S. Mishra, and E. Tadmor, *Arbitrarily high-order accurate entropy stable essentially nonoscillatory schemes for systems of conservation laws*, SIAM Journal on Numerical Analysis, 50 (2012), pp. 544–573.

[32] ——, *ENO reconstruction and ENO interpolation are stable*, Foundations of Computational Mathematics, 13 (2013), pp. 139–159.

[33] ——, *On the computation of measure-valued solutions*, Acta Numerica, 25 (2016), pp. 567–679.

[34] G. J. Gassner, *A skew-symmetric discontinuous Galerkin spectral element discretization and its relation to SBP-SAT finite difference methods*, SIAM Journal on Scientific Computing, 35 (2013), pp. A1233–A1253.

[35] G. J. Gassner, A. R. Winters, F. J. Hindenlang, and D. A. Kopriva, *The BR1 scheme is stable for the compressible Navier–Stokes equations*, Journal of Scientific Computing, 77 (2018), pp. 154–200.

[36] G. J. Gassner, A. R. Winters, and D. A. Kopriva, *A well balanced and entropy conservative discontinuous Galerkin spectral element method for the shallow water equations*, Applied Mathematics and Computation, 272 (2016), pp. 291–308.

[37] C. Geuzaine and J.-F. Remacle, *Gmsh: A 3-D finite element mesh generator with built-in pre-and post-processing facilities*, International Journal for Numerical Methods in Engineering, 79 (2009), pp. 1309–1331.

[38] J. Glimm, *Solutions in the large for nonlinear hyperbolic systems of equations*, Communications on pure and applied mathematics, 18 (1965), pp. 697–715.

[39] E. Godlewski and P.-A. Raviart, *Hyperbolic systems of conservation laws*, Mathématiques & applications, Ellipses, 1991.

[40] ——, *Numerical approximation of hyperbolic systems of conservation laws*, vol. 118, Springer, 2013.

[41] S. K. Godunov, *An interesting class of quasilinear systems*, in Dokl. Akad. Nauk SSSR, vol. 139, 1961, pp. 521–523.

[42] S. Gottlieb, C.-W. Shu, and E. Tadmor, *Strong stability-preserving high-order time discretization methods*, SIAM review, 43 (2001), pp. 89–112.

[43] J.-L. Guermond, R. Pasquetti, and B. Popov, *Entropy viscosity method for nonlinear conservation laws*, Journal of Computational Physics, 230 (2011), pp. 4248–4267.

[44] J.-L. Guermond and B. Popov, *Fast estimation from above of the maximum wave speed in the Riemann problem for the Euler equations*, Journal of Computational Physics, 321 (2016), pp. 908–926.

[45] B. Gustafsson, H.-O. Kreiss, and J. Oliger, *Time dependent problems and difference methods*, vol. 24, John Wiley & Sons, 1995.

[46] A. Harten, *On the symmetric form of systems of conservation laws with entropy*, Journal of computational physics, 49 (1983), pp. 151–164.

[47] A. Harten, B. Engquist, S. Osher, and S. R. Chakravarthy, *Uniformly high order accurate essentially non-oscillatory schemes, III*, in Upwind and high-resolution schemes, Springer, 1987, pp. 218–290.

[48] A. Harten, J. M. Hyman, P. D. Lax, and B. Keyfitz, *On finite-difference approximations and entropy conditions for shocks*, Communications on pure and applied mathematics, 29 (1976), pp. 297–322.

[49] A. Harten, P. D. Lax, and B. Van Leer, *On upstream differencing and Godunov-type schemes for hyperbolic conservation laws*, SIAM Review, 25 (1983), pp. 35–61.

[50] J. S. Hesthaven, S. Gottlieb, and D. Gottlieb, *Spectral methods for time-dependent problems*, vol. 21, Cambridge University Press, 2007.

[51] J. S. Hesthaven and T. Warburton, *Nodal Discontinuous Galerkin methods: algorithms, analysis, and applications*, Springer, 2007.

[52] J. E. Hicken, D. C. Fernández, and D. W. Zingg, *Multidimensional summation-by-parts operators: General theory and application to simplex elements*, arXiv preprint arXiv:1505.03125, (2015).

[53] A. Hiltebrand and S. Mishra, *Entropy stable shock capturing space–time discontinuous Galerkin schemes for systems of conservation laws*, Numerische Mathematik, 126 (2014), pp. 103–151.

[54] H. Holden, B. Øksendal, J. Ubøe, and T. Zhang, *Stochastic partial differential equations*, Springer, 1996.

[55] S. Hou and X.-D. Liu, *Solutions of multi-dimensional hyperbolic systems of conservation laws by square entropy condition satisfying discontinuous Galerkin method*, Journal of Scientific Computing, 31 (2007), pp. 127–151.

[56] T. Y. Hou, W. Luo, B. Rozovskii, and H.-M. Zhou, *Wiener chaos expansions and numerical solutions of randomly forced equations of fluid mechanics*, Journal of Computational Physics, 216 (2006), pp. 687–706.

[57] T. J. Hughes, L. Franca, and M. Mallet, *A new finite element formulation for computational fluid dynamics: I. symmetric forms of the compressible Euler and Navier-Stokes equations and the second law of thermodynamics*, Computer Methods in Applied Mechanics and Engineering, 54 (1986), pp. 223–234.

[58] F. Ismail and P. L. Roe, *Affordable, entropy-consistent Euler flux functions II: Entropy production at shocks*, Journal of Computational Physics, 228 (2009), pp. 5410–5436.

[59] K. Itô, *Multiple wiener integral*, Journal of the Mathematical Society of Japan, 3 (1951), pp. 157–169.

[60] G. S. Jiang and C.-W. Shu, *On a cell entropy inequality for discontinuous Galerkin methods*, Mathematics of Computation, 62 (1994), pp. 531–538.

[61] G. Karniadakis and S. Sherwin, *Spectral/hp element methods for computational fluid dynamics*, Oxford University Press, 2013.

[62] D. A. Kopriva and G. Gassner, *On the quadrature and weak form choices in collocation type discontinuous Galerkin spectral element methods*, Journal of Scientific Computing, 44 (2010), pp. 136–155.

[63] S. N. Kruzhkov, *First order quasilinear equations in several independent variables*, Matematicheskii Sbornik, 123 (1970), pp. 228–255.

[64] P. Lax and B. Wendroff, *Systems of conservation laws*, Communications on Pure and Applied mathematics, 13 (1960), pp. 217–237.

[65] P. D. Lax and R. D. Richtmyer, *Survey of the stability of linear finite difference equations*, Communications on pure and applied mathematics, 9 (1956), pp. 267–293.

[66] P. G. Lefloch, J.-M. Mercier, and C. Rohde, *Fully discrete, entropy conservative schemes of arbitrary order*, SIAM Journal on Numerical Analysis, 40 (2002), pp. 1968–1992.

[67] H. Liu, *On spectral approximations of stochastic partial differential equations driven by Poisson noise*, PhD thesis, University of Southern California, 2007.

[68] S. Lototsky, R. Mikulevicius, and B. L. Rozovskii, *Nonlinear filtering revisited: a spectral approach*, SIAM Journal on Control and Optimization, 35 (1997), pp. 435–461.

[69] S. Lototsky and B. Rozovskii, *Stochastic differential equations: a Wiener chaos approach*, From stochastic calculus to mathematical finance, (2006), pp. 433–506.

[70] S. Lototsky and B. Rozovskii, *Wiener chaos solutions of linear stochastic evolution equations*, The Annals of Probability, (2006), pp. 638–662.

[71] S. Lototsky and B. L. Rozovskii, *Passive scalar equation in a turbulent incompressible Gaussian velocity field*, Russian Mathematical Surveys, 59 (2004), p. 297.

[72] S. V. Lototsky and B. L. Rozovskii, *Stochastic partial differential equations*, Springer, 2017.

[73] W. Luo, *Wiener chaos expansion and numerical solutions of stochastic partial differential equations*, PhD thesis, California Institute of Technology, 2006.

[74] P. Malliavin, *Stochastic calculus of variation and hypoelliptic operators*, in Proc. Intern. Symp. SDE Kyoto 1976, Kinokuniya, 1978, pp. 195–263.

[75] R. Mikulevicius and B. Rozovskii, *Separation of observations and parameters in nonlinear filtering*, in Decision and Control, 1993., Proceedings of the 32nd IEEE Conference on, IEEE, 1993, pp. 1564–1569.

[76] R. Mikulevicius and B. Rozovskii, *Linear parabolic stochastic PDE and Wiener chaos*, SIAM journal on mathematical analysis, 29 (1998), pp. 452–480.

[77] R. Mikulevicius and B. Rozovskii, *On unbiased stochastic Navier–Stokes equations*, Probability Theory and Related Fields, 154 (2012), pp. 787–834.

[78] R. Mikulevicius and B. Rozovskii, *On distribution free Skorokhod–Malliavin calculus*, Stochastics and Partial Differential Equations: Analysis and Computations, 4 (2016), pp. 319–360.

[79] G. Milstein and M. Tretyakov, *Solving parabolic stochastic partial differential equations via averaging over characteristics*, Mathematics of computation, 78 (2009), pp. 2075–2106.

[80] G. N. Milstein and M. V. Tretyakov, *Stochastic numerics for mathematical physics*, Springer Science & Business Media, 2013.

[81] M. S. Mock, *Systems of conservation laws of mixed type*, Journal of Differential equations, 37 (1980), pp. 70–88.

[82] S. Osher, *Riemann solvers, the entropy condition, and difference*, SIAM Journal on Numerical Analysis, 21 (1984), pp. 217–235.

[83] S. Osher and E. Tadmor, *On the convergence of difference approximations to scalar conservation laws*, Mathematics of Computation, 50 (1988), pp. 19–51.

[84] E. Y. Panov, *Uniqueness of the solution of the Cauchy problem for a first order quasilinear equation with one admissible strictly convex entropy*, Mathematical Notes, 55 (1994), pp. 517–525.

[85] A. Quarteroni, A. Manzoni, and F. Negri, *Reduced basis methods for partial differential equations: an introduction*, vol. 92, Springer, 2015.

[86] H. Ranocha, P. Öffner, and T. Sonar, *Extended skew-symmetric form for summation-by-parts operators and varying Jacobians*, Journal of Computational Physics, 342 (2017), pp. 13–28.

[87] D. Ray, P. Chandrashekar, U. S. Fjordholm, and S. Mishra, *Entropy stable scheme on two-dimensional unstructured grids for Euler equations*, Communications in Computational Physics, 19 (2016), pp. 1111–1140.

[88] C.-W. Shu, *TVB uniformly high-order schemes for conservation laws*, Mathematics of Computation, 49 (1987), pp. 105–121.

[89] ——, *High order ENO and WENO schemes for computational fluid dynamics*, in High-order methods for computational physics, Springer, 1999, pp. 439–582.

[90] C.-W. Shu and S. Osher, *Efficient implementation of essentially non-oscillatory shock-capturing schemes*, Journal of Computational Physics, 77 (1988), pp. 439–471.

[91] ——, *Efficient implementation of essentially non-oscillatory shock-capturing schemes, II*, in Upwind and High-Resolution Schemes, Springer, 1989, pp. 328–374.

[92] A. V. Skorokhod, *On a generalization of a stochastic integral*, Theory of Probability & Its Applications, 20 (1976), pp. 219–233.

[93] M. Svärd and H. Özcan, *Entropy-stable schemes for the Euler equations with far-field and wall boundary conditions*, Journal of Scientific Computing, 58 (2014), pp. 61–89.

[94] E. Tadmor, *Skew-selfadjoint form for systems of conservation laws*, Journal of Mathematical Analysis and Applications, 103 (1984), pp. 428–442.

[95] ——, *The numerical viscosity of entropy stable schemes for systems of conservation laws. I*, Mathematics of Computation, 49 (1987), pp. 91–103.

[96] ——, *Entropy stability theory for difference approximations of nonlinear conservation laws and related time-dependent problems*, Acta Numerica, 12 (2003), pp. 451–512.

[97] S. Tan and C.-W. Shu, *Inverse Lax-Wendroff procedure for numerical boundary conditions of conservation laws*, Journal of Computational Physics, 229 (2010), pp. 8144–8166.

[98] E. F. Toro, *Shock-capturing methods for free-surface shallow flows*, John Wiley, 2001.

[99] ——, *Riemann solvers and numerical methods for fluid dynamics: a practical introduction*, Springer, 2013.

[100] D. Venturi, X. Wan, R. Mikulevicius, B. Rozovskii, and G. Karniadakis, *Wick–Malliavin approximation to nonlinear stochastic partial differential equations: analysis and simulations*, in Proc. R. Soc. A, vol. 469, The Royal Society, 2013, pp. 1–20.

[101] D. H. Wagner, *The Riemann problem in two space dimensions for a single conservation law*, SIAM Journal on Mathematical Analysis, 14 (1983), pp. 534–559.

[102] X. Wan and B. L. Rozovskii, *The Wick–Malliavin approximation of elliptic problems with log-normal random coefficients*, SIAM Journal on Scientific Computing, 35 (2013), pp. A2370–A2392.

[103] G.-C. Wick, *The evaluation of the collision matrix*, Physical review, 80 (1950), p. 268.

[104] P. Woodward and P. Colella, *The numerical simulation of two-dimensional fluid flow with strong shocks*, Journal of computational physics, 54 (1984), pp. 115–173.

[105] D. Xiu, *Numerical methods for stochastic computations: a spectral method approach*, Princeton university press, 2010.

[106] D. Xiu and G. E. Karniadakis, *The Wiener–Askey polynomial chaos for stochastic differential equations*, SIAM journal on scientific computing, 24 (2002), pp. 619–644.

[107] ——, *Modeling uncertainty in flow simulations via generalized polynomial chaos*, Journal of computational physics, 187 (2003), pp. 137–167.

[108] ——, *Supersensitivity due to uncertain boundary conditions*, International journal for numerical methods in engineering, 61 (2004), pp. 2114–2138.

[109] L. Zhang, T. Cui, and H. Liu, *A set of symmetric quadrature rules on triangles and tetrahedra*, Journal of Computational Mathematics, (2009), pp. 89–96.

[110] Q. Zhang and C.-W. Shu, *Stability analysis and a priori error estimates of the third order explicit Runge-Kutta discontinuous Galerkin method for scalar conservation laws*, SIAM Journal on Numerical Analysis, 48 (2010), pp. 1038–1063.

[111] X. Zhang, *On positivity-preserving high order discontinuous Galerkin schemes for compressible Navier–Stokes equations*, Journal of Computational Physics, 328 (2017), pp. 301–343.

[112] X. Zhang and C.-W. Shu, *On maximum-principle-satisfying high order schemes for scalar conservation laws*, Journal of Computational Physics, 229 (2010), pp. 3091–3120.

[113] ——, *On positivity-preserving high order discontinuous Galerkin schemes for compressible Euler equations on rectangular meshes*, Journal of Computational Physics, 229 (2010), pp. 8918–8934.

[114] X. Zhang, Y. Xia, and C.-W. Shu, *Maximum-principle-satisfying and positivity-preserving high order discontinuous Galerkin schemes for conservation laws on triangular meshes*, Journal of Scientific Computing, 50 (2012), pp. 29–62.

[115] Z. Zhang and G. Karniadakis, *Numerical methods for stochastic partial differential equations with white noise*, vol. 196, Springer, 2017.

[116] Z. Zhang, B. Rozovskii, M. V. Tretyakov, and G. E. Karniadakis, *A multistage Wiener chaos expansion method for stochastic advection-diffusion-reaction equations*, SIAM Journal on Scientific Computing, 34 (2012), pp. A914–A936.

[117] Z. Zhang, M. V. Tretyakov, B. Rozovskii, and G. E. Karniadakis, *A recursive sparse grid collocation method for differential equations with white noise*, SIAM Journal on Scientific Computing, 36 (2014), pp. A1652–A1677.

[118] ——, *Wiener chaos versus stochastic collocation methods for linear advection-diffusion-reaction equations with multiplicative white noise*, SIAM Journal on Numerical Analysis, 53 (2015), pp. 153–183.

[119] M. Zheng, B. Rozovsky, and G. E. Karniadakis, *Adaptive Wick–Malliavin approximation to nonlinear SPDEs with discrete random variables*, SIAM Journal on Scientific Computing, 37 (2015), pp. A1872–A1890.