# Stony Brook University

# Weighted-Least-Squares Based Essentially Non-Oscillatory Schemes on Unstructured Meshes

A Dissertation Presented

by

**Hongxu Liu**

to

The Graduate School

in Partial Fulfillment of the Requirements

for the Degree of

**Doctor of Philosophy**

in

**Applied Mathematics and Statistics**

Stony Brook University

December 2016

**Stony Brook University**

The Graduate School

Hongxu Liu

We, the dissertation committe for the above candidate for the

Doctor of Philosophy degree, hereby recommend

acceptance of this dissertation

**Xiangmin Jiao**
**Associate Professor, Department of Applied Mathematics and Statistics**

**James Glimm**
**Professor, Department of Applied Mathematics and Statistics**

**Roman Samulyak**
**Professor, Department of Applied Mathematics and Statistics**

**Alan C. Calder**
**Associate Professor, Department of Physics and Astronomy**

This dissertation is accepted by the Graduate School

Charles Taber
Dean of the Graduate School

Abstract of the Dissertation

# Weighted-Least-Squares Based Essentially Non-Oscillatory Schemes on Unstructured Meshes

by

**Hongxu Liu**

**Doctor of Philosophy**

in

**Applied Mathematics and Statistics**

Stony Brook University

2016

Essentially non-oscillatory schemes and their variants, such as ENO and WENO, are widely used high-order schemes for solving partial differential equations (PDEs), especially hyperbolic conservation laws with piecewise smooth solutions. For structured meshes, these techniques can achieve high order accuracy for smooth functions while being non-oscillatory near discontinuities. For unstructured meshes, which are needed for complex geometries, similar schemes are required but they are much more challenging to design, especially for finite different schemes. We propose a new family of non-oscillatory schemes, called WLS-ENO, in the context of solving hyperbolic conservation laws using both finite volume and finite difference methods over unstructured meshes. WLS-ENO is derived based on Taylor series expansion and solved using a weighed least squares formulation. Unlike other non-oscillatory schemes, the WLS-ENO does not require constructing sub-stencils, and hence it provides a more flexible framework and is less sensitive to mesh quality. We present both finite difference and finite volume schemes under the same framework and analyze the accuracy and stability. We show that finite volume WLS-ENO schemes can achieve better accuracy and stability than WENO finite volume schemes, and WLS-ENO

finite difference schemes are accurate, stable and more efficient than finite volume schemes. We present numerical results in 1-D, 2-D and 3-D for a number of benchmark problems and also report some comparisons against WENO if applicable.

# Contents

## Acknowledgements

First, I would like to thank Dr. Jiao for all his guidance, help, and support for the last 5 years. His organization, patience, and enthusiasm make him an excellent mentor. I have learned so much from him.

Next, I would like to thank all the present and past members of my research group, including Cao Lu, Xinglin Zhao, Rebecca Conley, Tristan Delaney, Xuebin Wang, Oliver Yang, Aditi Ghai, Xiongfei Wei and Cai Xue, for their help preparing this thesis. Thank you for all the help and encouragement. It has been great to work with all of you.

Lastly, I would like to thank my family and friends: Mom, Dad, Zheng Zhi, Fei Wang, Zhiqiang He, Wenxiang Ding, Lizhou Nie, Howard Fei, Peng Zhang, Kun Lan, Hao Huang, Xuyang Song, Runzhi Zhou and too many others to name. Thank all of you for your sincerely encouragement and support.

I acknowledge the sponsorship of part of my research from Argonne National Laboratory under Contract DE-AC02-06CH11357 for the SciDAC program funded by the Office of Science, Advanced Scientific Computing Research of the U.S. Department of Energy, as well as from DoD-ARO under contract number W911NF131-0249.

# List of Figures

# Chapter 1

# Introduction

Many physical phenomena, such as waves, heat conduction, electrodynamics, elasticity, etc., can be modeled by partial differential equations. With the development of computer technology, many numerical methods have been designed to solve these kinds of problems over the past decades. Among these there are finite difference methods and their generalizations, finite volume methods, and finite element methods.

In thesis, we consider the problem of reconstructing a piecewise smooth function, in the context of finite volume methods for hyperbolic conservation laws. Given a geometric domain $\Omega \subseteq \mathbb{R}^d$, suppose $u$ is a time-dependent piecewise smooth function over $\Omega$, such as a density function. For any connected region $\tau$, the $d$-dimensional conservation law can be written in the form

$$\int_\tau \frac{\partial u(\boldsymbol{x}, t)}{\partial t} d\boldsymbol{x} = -\int_{\partial\tau} \boldsymbol{F}(u) \cdot d\boldsymbol{a}, \tag{1.0.1}$$

where $\partial\tau$ is the boundary of $\tau$, and $\boldsymbol{F}$ is a function of $u$, corresponding to the flux.

A finite volume method solves the problem by decomposing the domain $\Omega$ into cells $\{\tau_i \mid i = 1, \ldots, N\}$. Let $|\tau_i|$ denote the volume of $\tau_i$ and $\overline{u}_i(t) = \frac{1}{|\tau_i|}\int_{\tau_i} u(\boldsymbol{x}, t)\, d\boldsymbol{x}$, the average of $u$ over $\tau_i$. We obtain an equation

$$\frac{d\overline{u}_i(t)}{\partial t} = -|\tau_i| \int_{\partial\tau_i} \boldsymbol{F}(u) \cdot d\boldsymbol{a}, \tag{1.0.2}$$

for each $\tau_i$. The boundary integral requires using numerical quadrature for the flux. The integration of the flux requires reconstructing $u$ from the cell averages $\overline{u}(t)$ in an accurate and stable fashion, and then evaluating the reconstruction at the quadrature points along the cell boundaries. For stability, $\boldsymbol{F}\cdot\boldsymbol{n}$ is typically replaced by a numerical flux, such as the Lax-Friedrichs flux,

$$\boldsymbol{F}\cdot\boldsymbol{n} = \frac{1}{2}\left[\left(\boldsymbol{F}\left(u^-\right) + \boldsymbol{F}\left(u^+\right)\right)\cdot\boldsymbol{n} - \alpha\left(u^+ - u^-\right)\right], \tag{1.0.3}$$

where $u^-$ and $u^+$ are the values of $u$ inside and outside the cell $\tau_i$. The parameter $\alpha$ is a constant, and it should be an upper bound of the eigenvalues of the Jacobian of $u$ in the normal direction.

In this context, we formulate the mathematical problem addressed in this paper as follows: Given the cell averages $\overline{u}_i$ of a piecewise smooth function $u(\boldsymbol{x})$ for cell $\tau_1, \tau_2, \ldots, \tau_N$, let $h_i$ be some length measure of cell $\tau_i$. Find a polynomial approximation $\widetilde{u}_i(\boldsymbol{x})$ of degree at most $p-1$ over $\tau_i$, such that

$$\|\widetilde{u}_i(\boldsymbol{x}) - u_i(\boldsymbol{x})\| = \mathcal{O}(h_i^p), \quad \boldsymbol{x} \in \tau_i. \tag{1.0.4}$$

In other words, $\widetilde{u}_i(\boldsymbol{x})$ is a $p$th order accurate approximation to $u(\boldsymbol{x})$ inside $\tau_i$. In the context of hyperbolic conservation laws, $u(\boldsymbol{x})$ in (1.0.4) is equal to $u(\boldsymbol{x}, t)$ in

(1.0.1) at a given $t$. For the facet between two cells, these reconstructions give us two values $u^-$ and $u^+$, which can then be substituted into (1.0.3) to calculate the numerical flux. These reconstructions must be accurate, and also must lead to stable discretization of the hyperbolic conservation laws when coupled with some appropriate time integration schemes, such as TVD Runge-Kutta schemes [12].

This reconstruction problem is decidedly challenging, because hyperbolic conservation laws can produce non-smooth solutions. An approximation scheme for smooth functions may lead to oscillations that do not diminish as the mesh is refined, analogous to the Gibbs phenomena. Such oscillations would undermine the convergence of the solutions. The ENO (Essentially Non-Oscillatory) and WENO (Weighted Essentially Non-Oscillatory) schemes [16, 32, 34] have been successful in solving this problem. In a nutshell, the WENO schemes use a convex combination of polynomials constructed over some neighboring cells, with higher weights for cells with smoother solutions and lower weights for cells near discontinuities. As a result, these methods can achieve high-order accuracy at smooth regions while being non-oscillatory near discontinuities. These reconstructions can be integrated into both finite volume and finite difference methods. With years of development, finite volume WENO schemes have been applied to both structured and unstructured meshes and higher dimensions [15, 20, 22, 31, 35, 37]. Various attempts have been applied to improve the weights for WENO reconstruction [1, 14, 29, 45]. Also, they have used WENO schemes in many applications, such as shock vortex interaction [13], incompressible flow problems [44], Hamilton-Jacobi equations [17], shallow water equations [23], etc.

Along the path of applying WENO schemes on unstructured meshes, tremendous effort has been made to improve the robustness of the schemes. Early attempts

[15] work well for most unstructured meshes, but some point distributions may lead to negative weights and in turn make the schemes unstable. An extension was proposed in [31] to mitigate the issue, but it still had limited success over complicated geometries due to inevitably large condition numbers of their local linear systems. More recently, several different partition techniques were proposed to improve stability, such as [26], which uses a hybrid of two different reconstruction strategies to achieve better results. The technique was adopted in [4, 7, 48] for further development.

Finite difference method is another type methods that widely used for solving conservation laws, which can be expressed in the differential form as

$$\frac{\partial \boldsymbol{u}(\boldsymbol{x},t)}{\partial t} + \boldsymbol{\nabla} \cdot \boldsymbol{F} = 0, \tag{1.0.5}$$

where $\boldsymbol{u}$ can be scalar or vector, $\boldsymbol{F}$ is a flux vector, and $\boldsymbol{\nabla}\cdot$ denotes the divergence operator. The finite difference methods were traditionally limited to structured meshes. WENO finite difference schemes are claimed to be applicable for only structured and smooth varying meshes [32]. Otherwise, it can be proven that no schemes higher than 2nd order can be conservative on unstructured meshes [32]. Spectral difference [25, 38] is another type of finite difference method which can be used on unstructured meshes. It is conservative due to the proper location of the solution points and flux points. It can also deliver high order accuracy and high efficiency. However, the initial setting requires that the solution points follow the same pattern in each element. Thus we cannot apply this method on unstructured meshes with initial condition available only on the nodes.

Since unstructured meshes are often necessary for complex geometries in solv-

ing PDEs, a finite difference method that can truly be applied on unstructured mesh and started with simple initial setting (such as given function values on nodes) is becoming a demanding task. Another motivation for developing new finite difference type schemes for conservation law is that when comparing finite difference schemes and finite volume schemes, we observed, as well as many researchers [4], that finite difference schemes are more efficient for a given accuracy, or in another words they are more accurate per degree of freedom, especially in higher dimensions. This is because on a given mesh in two and three dimensions, there are more elements than vertices. This makes finite volume approach more accurate, but at the expense of higher computational cost. Thus, node-based finite difference scheme is more preferable than finite volume scheme.

In this thesis, we propose two new families of schemes over unstructured meshes. We refer to the schemes as *WLS-ENO*, or *Weighted-Least-Squares based Essentially Non-Oscillatory* schemes for both finite volume and finite difference methods. Unlike the WENO scheme, our approach uses a generalized finite difference (GFD) formulation based on weighted least squares, rather than a weighted averaging of traditional finite differences. The GFD method is derived rigorously from Taylor series, and hence can deliver the same order of accuracy as traditional finite differences. In WLS-ENO, the convexity requirement is satisfied automatically, since the weights are specified *a priori*. These properties enable a more systematic way to construct non-oscillatory schemes. We also adapt the weights based on the functions so that they can deliver accurate and stable results for smooth conditions, as well as maintain non-oscillatory properties near discontinuities. Unlike WENO schemes for finite volume methods, our schemes work well on unstructured meshes and does not depend on mesh quality. They can be easily designed for high or-

der schemes in higher dimensions. Unlike WENO finite difference schemes, our schemes can also be applied on unstructured meshes, which makes them more suitable for real world problems where unstructured meshes are needed and discontinuities are presented in a node based fashion, such as underexpanded supersonic jet and free shear layer [6], Hamilton-Jacobi Equations [2], and shallow water problems [43].

The remainder of the thesis is organized as follows. In Chapter 2, we review background knowledge about WENO schemes for finite volume and finite difference methods and spectral difference. Chapter 3 introduces the general framework of WLS-ENO schemes for both finite volume and generalized finite difference method in 1D, 2D and 3D. In Chapter 3, we analyze the accuracy and stability of the proposed schemes and compares them against WENO and its previous generalization to unstructured meshes. Chapter 5 presents some numerical experiments for both structured and unstructured meshes. Finally, Chapter 6 concludes the paper with a discussion about the future work.

# Chapter 2

# Background and Related Work

In this chapter, we will review some of the previous finite difference and finite volume schemes used to solve hyperbolic conservation laws, including WENO schemes for finite difference and finite volume, spectral difference methods and some numerical methods for hyperbolic conservation laws. These methods will be used to design our new framework and schemes for both finite volume and finite difference methods.

## 2.1 ENO and WENO Schemes

### 2.1.1 WENO Reconstructions in 1-D

In the context of finite volume methods, the basic idea of WENO is to first construct several stencils for each cell and local polynomials over these stencils, so that the cell averages of these polynomials are the same as the given values. Then, a WENO scheme uses a convex combination of these polynomials to obtain a reconstruction of the function, where the weights for each stencil are controlled by a smoothness

indicator. We briefly describe the WENO scheme on a uniform 1-D grid below, and refer readers to [32] for more detail.

Given a 1-D domain $[a, b]$, suppose we have a uniform grid with nodes

$$a = x_{\frac{1}{2}} < x_{\frac{3}{2}} < x_{\frac{5}{2}} < \cdots < x_{N-\frac{1}{2}} < x_{N+\frac{1}{2}} = b. \tag{2.1.1}$$

We denote $i$th cell $\left[x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}\right]$ as $\tau_i$ for $i = 1, 2, \ldots, N$. Its cell center is $x_i = \frac{1}{2}\left(x_{i-\frac{1}{2}} + x_{i+\frac{1}{2}}\right)$, and its cell size is $h_i = x_{i+\frac{1}{2}} - x_{i-\frac{1}{2}}$. The cell average of a function $u(x)$ over $\tau_i$ is then

$$\overline{u}_i = \frac{1}{\Delta x_i} \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} u(x)\, dx, \quad i = 1, 2, \ldots, N. \tag{2.1.2}$$

For each cell $\tau_i$, our goal is to reconstruct a piecewise polynomial approximation $\widetilde{u}_i(x)$ of degree at most $p-1$, such that it approximates $u(x)$ to $p$th order accuracy within $\tau_i$, i.e.,

$$\widetilde{u}_i(x) = u(x) + \mathcal{O}(h^p), \quad x \in \tau_i, \quad i = 1, \ldots, N, \tag{2.1.3}$$

where $h = \min\{h_i \mid 1 \le i \le N\}$.

To find such a polynomial, a WENO scheme first selects $p$ sub-stencils about $\tau_i$, each containing $p$ cells. Consider a particular sub-stencil

$$S_j(i) = \{\tau_{i-j}, \ldots, \tau_{i-j+p-1}\}, \tag{2.1.4}$$

and let $\phi_{i,j}(x)$ be a polynomial approximation of $u$ over $S_j(i)$, obtained by requiring the integral of $\phi_{i,j}(x)$ over each cell in the sub-stencil to be equal to that of $u_i(x)$.

If the $p$th derivative of $u$ is bounded over the sub-stencil $S_j(i)$, then $\phi_{i,j}(x)$ satisfies (2.1.3). However, if $u(x)$ has discontinuities within the sub-stencil, then $\phi_{i,j}(x)$ may be oscillatory. The WENO scheme then constructs a non-oscillatory approximation by taking a convex combination of $\phi_{i,j}(x)$

$$\widetilde{u}_i(x) = \sum_j \omega_j \phi_{i,j}(x), \tag{2.1.5}$$

where $\omega_j = \alpha_j / \sum_{k=0}^{p-1} \alpha_k$ and is chosen such that $\omega_j$ approaches zero for sub-stencils with discontinuities. A typical choice of $\alpha_j$ is

$$\alpha_j = d_j / \left(\epsilon + \beta_j\right)^2, \tag{2.1.6}$$

where $d_j$ is a nonnegative coefficient such that

$$\widetilde{u}_i(x) = \sum_{j=0}^{p-1} d_j \phi_{i,j}(x) = u\left(x_{i+\frac{1}{2}}\right) + \mathcal{O}(h^{2p-1}). \tag{2.1.7}$$

The parameter $\epsilon$ is a small parameter, such as $\epsilon = 10^{-6}$, introduced to avoid instability due to division by zero or too small a number. The non-negativity of $d_i$ is important for stability purposes. The $\beta_j$ is the *smoothness indicator*. If $u(x)$ is smooth over the sub-stencil $S_j(i)$, then $\beta_j = \mathcal{O}(h^2)$; otherwise, $\beta_j = \mathcal{O}(1)$. A typical choice of $\beta_j$, as introduced in [32], is

$$\beta_j = \sum_{k=1}^{p-1} \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} h^{2k-1} \left(\frac{\partial^k \phi_j(x)}{\partial x^k}\right)^2 dx, \tag{2.1.8}$$

where the $h^{2k-1}$ term is introduced to make $\beta_j$ independent of the grid resolution.

For example, in the simplest case where $p = 2$,

$$\begin{aligned} \beta_0 &= \left(\overline{u}_{i+1} - \overline{u}_i\right)^2, \\ \beta_1 &= \left(\overline{u}_i - \overline{u}_{i-1}\right)^2. \end{aligned} \tag{2.1.9}$$

Alternative smoothness indicators have been proposed in [1, 3, 14, 30, 46].

### 2.1.2 WENO Schemes on 2-D and 3-D Structured Meshes

Originally developed in 1-D, the WENO schemes can be generalized to structured meshes in 2-D and 3-D. Here, we give a brief overview of the reconstructions in 2-D, which generalize to 3-D in a relatively straightforward manner.

Consider an $N_x$-by-$N_y$ structured grid, and let $\tau_{ij}$ denote the cell $(i, j)$ in the grid, where $i = 1, 2, \ldots, N_x$ and $j = 1, 2, .., N_y$. Suppose the cell averages of a function $u(x, y)$,

$$\overline{u}_{ij} = \frac{1}{\Delta x_i \Delta y_j} \int_{y_{j-\frac{1}{2}}}^{y_{j+\frac{1}{2}}} \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} u\left(x, y\right) \, dxdy, \tag{2.1.10}$$

are given. We would like to find

$$\widetilde{u}_{ij}(x, y) = \sum_{r=0}^{p-1} \sum_{s=0}^{p-1} a_{rs} x^r y^s, \tag{2.1.11}$$

where $a_{rs}$ are the coefficients to be determined so that $\widetilde{u}_{ij}$ approximates $u$ to $p$th order accuracy over $\tau_{ij}$.

Similar to the 1-D case, a WENO scheme first constructs polynomial approximations over a selection of sub-stencils and then computes a convex combination

of these approximations. Consider a particular sub-stencil

$$S_{lm}(i,j) = \{\tau_{IJ} : i-l \le I \le i-l+p-1, j-m \le J \le j-m+p-1\}. \tag{2.1.12}$$

Let $\phi_{ijlm}(x,y)$ denote a polynomial reconstruction of $u$ over $S_{lm}(i,j)$, whose integration over each cell in $S_{lm}(i,j)$ is equal to the given cell average. The function $\phi_{ijlm}(x,y)$ approximates $u(x,y)$ over $\tau_{ij}$ to $p$th order accuracy for smooth functions. To obtain a non-oscillatory reconstruction for non-smooth functions, the WENO scheme computes $\widetilde{u}_{ij}$ as

$$\widetilde{u}_{ij}(x,y) = \sum_l \sum_m \omega_{lm} \phi_{ijlm}(x,y). \tag{2.1.13}$$

The weights $\omega_{lm}$ are chosen so that the order of accuracy of $\widetilde{u}_{ij}$ is maximized for smooth functions, and then further augmented based on similar smoothness indicators as in 1-D, so that the weights would approach zero for the sub-stencils with discontinuities. For stability, it is important that the weights are nonnegative, which imposes some constraints to the selection of stencils. For more detail, see [22, 32, 41].

### 2.1.3 ENO and WENO Schemes on Unstructured Meshes

Besides structured meshes, WENO can also be generalized to unstructured meshes. Typically, these schemes are also constructed from some convex combination of lower-order schemes on sub-stencils. However, compared to structured meshes, it is much more challenging to construct stable WENO schemes on unstructured meshes, because it is difficult to satisfy the convexity requirement. Some WENO

schemes have been proposed for 2-D [15, 22, 31, 41] and 3-D [20, 35, 37, 44, 47]. Below, we briefly review these generalizations, focusing on three different types.

The first type of WENO reconstruction, as proposed in [15], uses a combination of high-order polynomials computed from low-order polynomials over sub-stencils. For example, to obtain a third-order reconstruction, the scheme would first construct linear approximations over several sub-stencils by requiring the cell averages of the polynomials to be equal to the given cell averages, and then compute a quadratic polynomial from a weighted average of these linear polynomials. This technique works for unstructured meshes, even for meshes with mixed types of elements. However, depending on the mesh, the weighted average may not form a convex combination, and the linear system for calculating the weights may be ill-conditioned.

The second type of WENO reconstruction is similar to the first type, except that it compromises the order of accuracy of the convex combination, by allowing convex combination to be the same degree polynomials as those for the sub-stencils. Compared to the first type, this construction is less sensitive to mesh quality than the first type at the cost of lower accuracy. Therefore, it is often used as a fallback of the first type for robustness [26].

The third type of WENO reconstruction builds the reconstructions in a hierarchical fashion. An example is the approach in [41], which first finds the smoothest linear reconstructions over the first-layer three-cell stencils, and then use these linear reconstructions to build quadratic reconstructions over the second-layer stencils. This approach can be applied iteratively to construct higher-order reconstructions. However, as shown in [41], the accuracy of the reconstruction may not improve as the degree of the polynomial increases, especially near boundaries.

Besides the above WENO schemes, we also note some recent development of the Central ENO (CENO) schemes [4, 5], based on least squares approximations. CENO schemes require limiters for linear reconstructions to preserve monotonicity. Our proposed WLS-ENO schemes differ from the WENO and CENO reconstructions, in that the WLS-ENO schemes utilize a weighed least squares formulation, do not require limiters, and are insensitive to mesh quality due to adaptive stencils.

## 2.2 WENO and Spectral Difference Schemes

### 2.2.1 WENO Schemes for Finite Difference Method

WENO finite difference schemes have been successful for many decades. Based on the simple framework, we are able to use WENO finite difference schemes for a lot of problems on structured grid with both high accuracy and efficiency. Suppose we have a structured mesh in 1D domain $[a, b]$,

$$a = x_1 < x_2 < x_3 < \cdots < x_{N-1} < x_N = b, \tag{2.2.1}$$

and we define $x_{i+\frac{1}{2}} = (x_i + x_{i+1})/2$, $u_i = u(x_i)$. The conservation law in 1-D has the following form

$$\frac{du_i}{dt} + \frac{df(u_i)}{dx} = 0, \quad i = 1, 2, ..., N \tag{2.2.2}$$

The derivative of function $f(x)$ is approximated in a conservative form and $f(x)$ is considered as the cell average of another function $g$. What we need is to approximate $g$ to high order accuracy at cell boundary $x_{i+\frac{1}{2}}$ and $x_{i-\frac{1}{2}}$. This is where

WENO schemes are used to maintain high accuracy and non-oscillatory property.

The key idea for WENO finite difference schemes is as follows. The scheme is composed of weighted average of several sub-stencils, and the weights are properly computed so that they approach to a constant if the function is smooth within sub-stencils, and approach to zero if discontinuities occur. In this way, we can maximize the use of the function values for smooth points to achieve high order accuracy and avoid as much as possible the use of discontinuous points. For example, suppose we have three approximated values $g_i^{(k)}, \quad k = 1, 2, 3$ for $x_{i+\frac{1}{2}}$ from three sub-stencils. The fifth order WENO finite difference schemes have the following form

$$\widehat{g_i} = \sum_{k=1}^{3} w_k g_i^{(k)} \tag{2.2.3}$$

with

$$w_k = \frac{a_k}{\sum_{j=1}^{3} a_j}, \quad a_k = \frac{d_k}{(\epsilon + \beta_k)^2}, \quad k = 1, 2, 3 \tag{2.2.4}$$

where $d_k$ is a constant such that the weighted sum of the $g_i^{(k)}$ is fifth order accuracy, $\beta_k$ is the smoothness indicator, which measures the smoothness within each sub-stencils, and $\epsilon$ is a small constant to avoid the denominator to become zero. One advantage of this non-linear weight is that it can capture the discontinuities within sub-stencils and diminish the influence of discontinuities so that non-oscillatory solutions can be achieved. For more details, please refer to [28, 33].

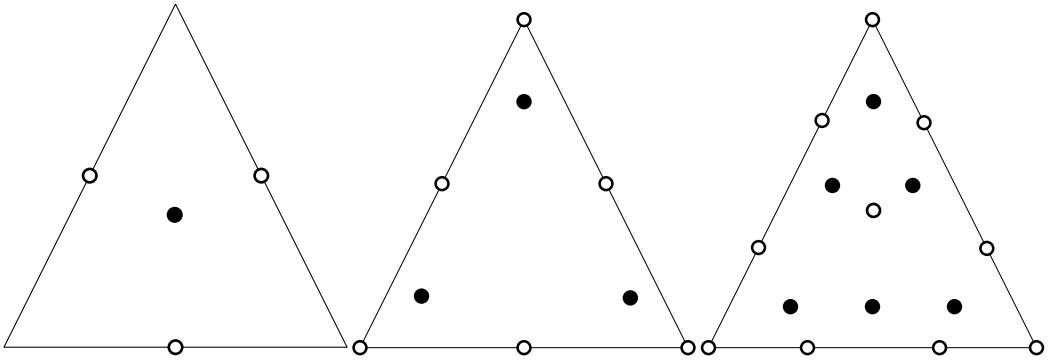
### 2.2.2 Spectral Difference Method

The spectral difference method has been around for a few decades [25, 38]. It works well on both structured and unstructured grid. It can achieve high order

accuracy with high efficiency. The basic idea is that it places solution points in every element of unstructured mesh in the same pattern, so that the scheme looks the same for every element if converted to a reference element. In this way, high efficiency can be achieved. Also, the order of accuracy depends on the number of solution points placed in each element and the conservation is preserved because of the proper positions of these solution points. Thus, for computations starting with proper placement of solution points, it may be by far the best method we can use. However, for real world problem, the solutions points may not be properly placed and spectral difference method may not be an option for these problems.

Figure 2.1 showed the placement of solution points and flux points. The solution points are represented as black dots and flux points are circles. The basic framework for spectral difference method is as follows. Suppose we are given function values at the solution points. We first build a polynomial for each element by interpolating the function values at solution points within the element. Then, we evaluate the polynomial at flux points, where we need to compute the fluxes. The fluxes are calculated through normal and tangential vectors of the edge of the element. This is because the normal component of the flux vector on each edge should be the same for two elements sharing the same edge. After obtaining the fluxes, we use the fluxes on each edge to compute the derivatives. Finally, we use TVD time stepping, such as TVD Runge-Kutta method.

To deal with discontinuities, the spectral difference method exploited minmod limiters to suppress oscillations. If the evaluated values on the flux points are beyond certain range, the values are limited. Similar technique has been adopted by the discontinuous Galerkin method [8] and the spectral volume method [39, 40]. Also, it can be proved that for the configuration of the solution points and flux



(a) 1st order. (b) 2nd order. (c) 3rd order.

Figure 2.1: Placement of solution points and flux points.

points, the schemes are conservative. For more details, please refer to [25, 38].

# Chapter 3

# Weighed-Least-Squares Based ENO Schemes

In this chapter, we present the general framework for solving conservation laws with discontinuities and propose two classes of essentially non-oscillatory schemes based on the framework, referred to as *WLS-ENO finite volume schemes and WLS-ENO finite difference schemes.* In the context of finite volume methods, the schemes reconstruct a function $u(\boldsymbol{x})$ over each cell, given the cell averages of $u$, denoted by $\overline{u}$, for all the cells. For each point along cell boundaries, the reconstructions then provide two values, $u^-$ and $u^+$, which can then be used in (1.0.3) to calculate fluxes. Unlike WENO schemes, the WLS-ENO finite volume scheme does not use weighted averaging of functional approximations over sub-stencils. Instead, it computes the reconstruction over each cell based on weighted least squares with an adaptive stencil. It can achieve optimal accuracy for smooth functions, stability around discontinuities, and insensitivity to mesh quality.

WLS-ENO finite difference schemes are designed solving the conservative form

of hyperbolic conservation laws. They are based on the same weighted least squares framework and can be applied on complex geometries with unstructured meshes. The accuracy, stability and efficiency can be achieved by the following steps. Suppose we are given a mesh with function values at the nodes. We first define a control volume for each node. The control volumes of all nodes cover the entire computational domain. For each control volume, we define flux points, which are the points where we compute the fluxes. For different schemes, we may use different patterns of flux points. The function values are computed at flux points using stencils for each node in a weighed least squares sense. Then, we apply numerical fluxes like Lax-Friedrichs flux to obtain the fluxes. Next, we compute the flux derivates by the fluxes we just obtained. Finally, TVD time stepping is applied so we can get the function values at the next time step.

We will present the derivations as well as the implementation of WLS-ENO finite volume schemes in 1-D,2-D and 3-D over structured and unstructured meshes. The derivations of WLS-ENO finite difference schemes will also be introduced in 1-D and 2-D.

## 3.1 Weighted-Least-Squares Based ENO Schemes for Finite Volume Method

### 3.1.1 WLS-ENO Schemes for Finite Volume in 1-D

We first derive the WLS-ENO schemes in 1-D. Suppose we are given a grid

$$a = x_{\frac{1}{2}} < x_{\frac{3}{2}} < x_{\frac{5}{2}} < \cdots < x_{N-\frac{1}{2}} < x_{N+\frac{1}{2}} = b, \tag{3.1.1}$$

and the cell averages $\overline{u}_i$ of a function $u(x)$ over each cell $\tau_i$, $i = 1, 2, \ldots, N$. For generality, we assume the grid is non-uniform, with varying cell sizes. Below, we first describe how to reconstruct $u$ from $\overline{u}_i$ for smooth functions, and then augment the method to handle discontinuities.

**WLS-Based Reconstruction for Smooth Functions**

Without a loss of generality, let us consider the reconstruction of $u$ over $\tau_i$ at its boundary point $x_{i+\frac{1}{2}}$. To achieve $p$th order accuracy, we need to construct a polynomial approximation of degree at least $p-1$. We choose a stencil with $n$ cells to perform the reconstruction, where $n \geq p$.

Suppose there are $l$ cells to the left of $\tau_i$ in the stencil. The full stencil is given by the set

$$S(i) = \{\tau_{i-l}, \ldots, \tau_{i-l+n-1}\}. \tag{3.1.2}$$

From Taylor series expansion, we can approximate function $u(x)$ at point $x_{i+\frac{1}{2}}$ to $p$th order accuracy by

$$u(x) = \sum_{k=0}^{p-1} \frac{u^{(k)}(x_{i+\frac{1}{2}})}{k!} \left(x - x_{i+\frac{1}{2}}\right)^k + \mathcal{O}(h^p), \tag{3.1.3}$$

where $h$ denotes the average edge lengths. The cell average over $\tau_j$ in the stencil can be approximated by

$$\begin{aligned}
\overline{u}_j &= \sum_{k=0}^{p-1} \frac{u^{(k)}(x_{i+\frac{1}{2}})}{k!\left(x_{j+\frac{1}{2}} - x_{j-\frac{1}{2}}\right)} \int_{x_{j-\frac{1}{2}}}^{x_{j+\frac{1}{2}}} \left(x - x_{i+\frac{1}{2}}\right)^k dx + \mathcal{O}(h^p) \\
&= \sum_{k=0}^{p-1} \frac{u^{(k)}(x_{i+\frac{1}{2}})}{(k+1)!} \left[\left(x_{j+\frac{1}{2}} - x_{i+\frac{1}{2}}\right)^{k+1} - \left(x_{j-\frac{1}{2}} - x_{i+\frac{1}{2}}\right)^{k+1}\right] + \mathcal{O}(h^p).
\end{aligned}$$

Given $\overline{u}_j$ at the $j$th cells in $S(i)$, we then construct an $n \times p$ linear system

$$\boldsymbol{A}\boldsymbol{v} \approx \overline{\boldsymbol{u}}, \tag{3.1.4}$$

where

$$a_{JK} = \frac{1}{K!}\left[\left(x_{i-l+J-\frac{1}{2}} - x_{i+\frac{1}{2}}\right)^K - \left(x_{i-l+J-\frac{3}{2}} - x_{i+\frac{1}{2}}\right)^K\right] \tag{3.1.5}$$

for $J \equiv j + l + 1 - i \in [1, n]$ and $K \equiv k + 1 \in [1, p]$, $\overline{\boldsymbol{u}}$ is composed of the cell averages $\overline{u}_j$, and $\boldsymbol{v}$ is composed of the derivative of function $u$ at $x_{i+\frac{1}{2}}$, i.e., $v_K = u^{(k)}(x_{i+\frac{1}{2}})$.

Eq. (3.1.4) in general is a rectangular linear system, and we can solve it using a weighted least squares formulation. In particular, we assign a different weight to each cell. Let $\boldsymbol{W}$ denote a diagonal matrix containing these weights. The problem can be written in matrix form as

$$\min \|\boldsymbol{W}\boldsymbol{A}\boldsymbol{v} - \boldsymbol{W}\overline{\boldsymbol{u}}\|_2. \tag{3.1.6}$$

The weights allow us to assign different priorities to different cells. For example, we may give higher weights to the cells closer to $I_i$. We solve this weighted least squares problem using QR factorization with column pivoting, as we will describe in more detail in Section 3.3. Since the method is derived based on Taylor series expansions directly, this WLS-based reconstruction can deliver the same order of accuracy as interpolation-based schemes for smooth functions, as we demonstrate in Section 4.1.2. For discontinuous functions, these weights can also allow us to suppress the influence of cells close to discontinuities, as we discuss next.

**WLS-ENO for Discontinuous Functions**

To apply WLS-ENO schemes to discontinuous functions, we modify the weighting matrix $\boldsymbol{W}$ in (3.1.6). The main idea is to assess the smoothness of the function within each cell of the stencil, and then define the weights correspondingly. By letting the weights be far smaller for the cells near discontinuities than those away from discontinuities, we can then effectively suppress oscillations.

We first construct a *non-smoothness indicator* of the function, analogous to those used in WENO schemes. Specifically, for the $j$th cell in the stencil for $I_i$, with $j = i - l + J$, we can define the indicator for cell $I_j$ as

$$\beta_j = \begin{cases} (\overline{u}_j - \overline{u}_i)^2 + \epsilon h^2 & j \neq i \\ \min\{\beta_{j-1}, \beta_{j+1}\} & j = i \end{cases}, \tag{3.1.7}$$

where $\epsilon$ is a small constant, such as $\epsilon = 10^{-2}$, introduced to avoid the indicator being too close to zero, and $h$ is some measure of average edge length. Note that $\beta_j = \mathcal{O}(h^2)$ if $u$ is smooth at $\tau_j$ and $\beta_j = \mathcal{O}(1)$ near discontinuities. Therefore, it captures the non-smoothness of the function. We therefore refer to $\beta$ as a non-smoothness indicator, although its counterpart in WENO is called the "smoothness indicator."

We then define the weights based on $\beta_j$. To suppress oscillations, it is desirable to use smaller weights for cells at discontinuities. Therefore, we make the weights in $\boldsymbol{W}$ to be inversely proportional to $\beta_j$ when $j \neq i$, and make the value larger for

$\beta_i$. Specifically, we choose

$$\omega_j = \begin{cases} 1/\beta_j & j \neq i \\ \alpha/\beta_j & j = i \end{cases}, \tag{3.1.8}$$

where $\alpha > 1$, such as $\alpha = 1.5$. It is easy to see that $\omega_j$ is $\mathcal{O}(1/h^2)$ if the function is smooth around cell $i$ and $\omega_q = \mathcal{O}(1)$ if the function is discontinuous in the cell. After computing the weights, we substitute them into (3.1.6) to compute the reconstruction. Note that unlike the weights in WENO, we do not need to normalize the weights by dividing them by the sum of the weights. As we will demonstrate in Section 4.1.2, this approach effectively suppresses the oscillations near discontinuities, similar to WENO schemes.

### 3.1.2 Generalization of WLS-ENO Schemes to 2-D and 3-D

The WLS-ENO reconstruction can be generalized to 2-D and 3-D, for evaluating the values at quadrature points along the cell boundaries. Similar to 1-D, we derive the higher-dimensional version of the linear system (3.1.4) over each cell based on Taylor series expansion, and then solve it based on weighted least squares.

Let us first consider the scheme in 2-D. First, we choose $n$ cells with index $i_1, i_2, \ldots, i_n$ as the stencil for cell $\tau_i$. Let $(x_i, y_i)$ denote its centroid. From the 2-D Taylor series expansion, we can approximate $u(x, y)$ about $(x_i, y_i)$ to $p$th order accuracy by

$$u(x,y) = \sum_{q=0}^{p-1} \sum_{k,l \geq 0}^{k+l=q} \frac{\partial^q u\left(x_i, y_i\right)}{\partial x^k \partial y^l} \frac{\left(x - x_i\right)^k \left(y - y_i\right)^l}{k!l!} + \mathcal{O}\left(\|\delta\|^p\right), \tag{3.1.9}$$

where $\delta = \max\{|x - x_i|, |y - y_i|\}$. Thus, the cell averages over the $j$th cell $\tau_j$ can be approximated by

$$\overline{u}_j = \frac{1}{k!l!\,|\tau_j|} \sum_{q=0}^{p-1} \sum_{k,l \geq 0}^{k+l=q} \frac{\partial^q u(x_i, y_i)}{\partial x^k \partial y^l} \iint_{\tau_j} (x - x_i)^k (y - y_i)^l \, dxdy + \mathcal{O}(\|\delta\|^p) \tag{3.1.10}$$

for $j = i_1, \ldots, i_n$, where $|\tau_j|$ denotes the area of $\tau_j$. Therefore, we obtain $n$ equations from the $n$ cells in the stencil about $\tau_i$, which can then be solved using the weighted least squares formulation (3.1.6).

The construction in 3-D is based on the 3-D Taylor series expansion about a centroid $(x_i, y_i, z_i)$ of the $i$th cell $\tau_i$,

$$u(x, y, z) = \sum_{q=0}^{p-1} \sum_{k,l,m \geq 0}^{k+l+m=q} \frac{\partial^q u(x_i, y_i, z_i)}{\partial x^k \partial y^l \partial z^m} \frac{(x - x_i)^k (y - y_i)^l (z - z_i)^m}{k!l!m!} + \mathcal{O}(\|\delta\|^p), \tag{3.1.11}$$

where $\delta = \max\{|x - x_i|, |y - y_i|, |z - z_i|\}$. Then the cell averages over the $j$th cell $\tau_j$ in the stencil for $\tau_i$ can be approximated by

$$\overline{u}_j = \frac{1}{k!l!m!\,|\tau_j|} \sum_{q=0}^{p-1} \sum_{k,l,m \geq 0}^{k+l+m=q} \frac{\partial^q u(x_i, y_i, z_i)}{\partial x^k \partial y^l \partial z^m} \iiint_{\tau_j} (x - x_i)^k (y - y_i)^l (z - z_i)^m \, dxdydz + \mathcal{O}(\|\delta\|^p) \tag{3.1.12}$$

for $j = i_1, \ldots, i_n$, where $|\tau_j|$ denotes the volume of $\tau_i$. We can then solve the resulting least squares problem using weighted least squares.

To determine the weights in (3.1.6), we define the non-smoothness indicator $\beta_j$ similar to (3.1.7), and then define the weights $\omega_j$ as in (3.1.8). As in 1-D, it is easy to show that $\beta_j = \mathcal{O}(h^2)$ if $u$ is smooth at $\tau_j$ and $\beta_j = \mathcal{O}(1)$ near discontinuities, and therefore the weights can effectively suppress the effect of cells near discontinuities. After solving the linear system and obtaining the polynomial approximation, we can

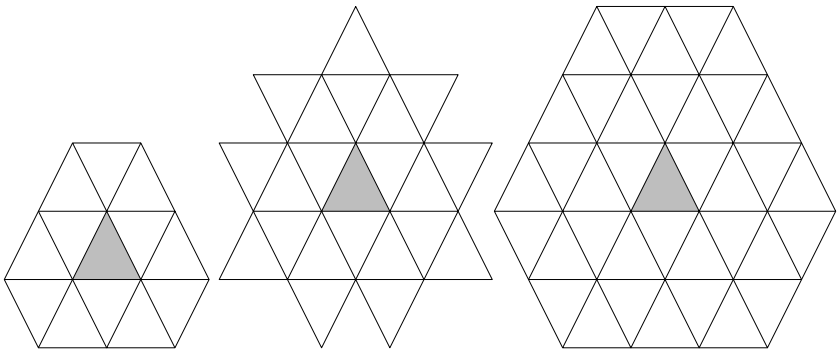


Figure 3.1: Examples of 1-ring, 1.5-ring and 2-ring neighborhood of a triangle.

then evaluate $u(x, y)$ at the quadrature points along the boundary for computing the numerical fluxes.

One remaining question is the selection of the stencils. For the reconstruction over triangular meshes in 2-D, we adopt the strategy in [18] to define $k$-ring neighbor cells, with $1/2$-ring increments:

- The $1/3$-ring neighbor cells are the cells that share an edge with a cell.
- The 1-ring neighbor cells of a cell are those that share at least one vertex with the center cell.
- For any positive integer $k$, the $(k+1)$-ring neighborhood of a cell is the union of $k$-ring neighborhood and 1-ring neighbors of its $k$-ring neighbor cells. The $(k + 1/2)$-ring neighborhood is the union of $k$-ring neighborhood and $1/2$-ring neighbors of the $k$-ring neighbor cells.

Figure 3.1 illustrates the neighborhood definitions up to 2 rings. The $1/2$-ring increment allows finer granularity in the increment of the stencil sizes.

For the reconstruction over tetrahedral meshes in 3-D, the standard $k$-ring neighbors grow very rapidly. To allow finer granularity, we define $k$-ring neighbor cells

with $1/3$-ring increments, similar to those defined in [9]:

- The $1/3$-ring neighbor cells of a cell are the cells that share at least one face with the center cell.
- The $2/3$-ring neighbor cells of a cell are the cells that share at least one edge with the center cell.
- The 1-ring neighbor cells of a cell are the cells that share at least one vertex with the center cell.
- For any positive integer $k$, the $(k + 1/3)$-ring neighborhood of a cell is the union of the $k$-ring neighborhood and the $1/3$-ring neighbors of its $k$-ring neighbor cells. The $(k + 2/3)$-ring neighborhood is the union of the $k$-ring neighborhood and the $2/3$-ring neighbors of the $k$-ring neighbor cells.

With the above definitions, we adaptively choose the $k$-ring neighborhood, so that the number of cells in a stencil is approximately equal to 1.5 to 2 times the number of coefficients in the Taylor polynomial. This adaptive strategy allows the WLS-ENO scheme to be less dependent on mesh quality than WENO schemes.

The above strategy works well in practice for most cases. However, on poor-quality meshes, some $k$-ring neighborhoods may be nearly one-sided, which may cause the least-squares approximation to be closer to extrapolation along some directions. This may cause the reconstructed value to fall beyond the maximum and minimum values of the cell averages in the neighborhood, and in turn lead to oscillations. This rarely happens in 1-D or 2-D, but we do observe it in practice for some 3-D meshes. This issue could be mitigated by using limiters, analogous to the approach in [27] for multidimensional reconstructions on unstructured meshes.

However, we resolve the issue by adapting the stencil as follows. First, after the reconstruction, we check whether the reconstructed values are between the maximum and minimum values in the neighborhood. If not, we compute the plane that passes through the centroid $\boldsymbol{x}_i$ of $\tau_i$ and is orthogonal to $\nabla u$ at $\boldsymbol{x}_i$. Next, we select a subset of the cells from an enlarged stencil to ensure the new stencil is well balanced on the two sides of the plane. Because of the smoothing nature of least squares, we find that the new polynomial approximation typically falls within the range on balanced stencils.

## 3.2 Weighted-Least-Squares ENO Finite Difference Schemes

In this section we will describe the basic framework of weighted-least-squares based ENO schemes for finite difference. This framework allows us to work on complex geometries with unstructured meshes. The schemes utilize the Taylor series expansions and weighted-least-squares approximations in a generalized finite difference setting, which are accurate, stable and efficient.

Suppose we are given a mesh with function values at the nodes. We first define a control volume for each node. The control volumes of all nodes cover the entire computational domain. For each control volume, we define flux points, which are the points where we compute the fluxes. For different schemes, we may use different patterns of flux points. The procedure for WLS-ENO schemes is as follows. We first compute the function values at flux points using stencils for each node in a weighed least squares sense. Second, we apply numerical fluxes like Lax-Friedrichs

flux to obtain the fluxes. Then we compute the flux derivates by the fluxes we just obtained. Finally, TVD time stepping is applied so we can get the function values at the next time step. In the following, we provide more details of various schemes for in both 1D and 2D cases.

### 3.2.1 WLS-ENO Schemes in 1-D

Suppose we are given a grid

$$a = x_1 < x_2 < x_3 < \cdots < x_{N-1} < x_N = b, \tag{3.2.1}$$

and the function $u(x)$ at each node $x_i$, $i = 1, 2, ..., N$. The mesh can be either structured or unstructured(i.e. not equally spaced). We first define the control volume for each node. For any point not on the boundary $x_i$, $i = 2, 3, ..., N-1$, we define the control volume as

$$V_i = \left[x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}\right], \tag{3.2.2}$$

where $x_{i-\frac{1}{2}} = (x_{i-1} + x_i)/2$.

#### WLS-Based Reconstruction for Smooth Functions

Here for each node, we are aiming at approximating the function values within its control volume to certain order of accuracy. Without loss of generality, suppose we want to achieve $p$th order accuracy. To do this, we need to choose a stencil with $n$ points and construct a polynomial of degree at least $p-1$. We require that $n \geq p$ and the stencil is chosen symmetrically with respect to the node. For example, for

degree 2, we choose a five-point stencil which has the following form

$$S\left(i\right)=\left\{x_{i-2},x_{i-1},x_i,x_{i+1,}x_{i+2}\right\}, \tag{3.2.3}$$

For degree 3 and degree 4 schemes, the stencil we choose has seven points,

$$S\left(i\right)=\left\{x_{i-3},x_{i-2},x_{i-1},x_i,x_{i+1,}x_{i+2},x_{i+3}\right\}, \tag{3.2.4}$$

For degree 5, we choose 9 points as our stencil,

$$S\left(i\right)=\left\{x_{i-4},x_{i-3},x_{i-2},x_{i-1},x_i,x_{i+1,}x_{i+2},x_{i+3},x_{i+4}\right\}, \tag{3.2.5}$$

Suppose there are $l$ points to the left of $x_i$ in the stencil, we have the following form,

$$S\left(i\right)=\left\{x_{i-l},x_{i-l+1},...,x_{i-l+n-1}\right\}. \tag{3.2.6}$$

The polynomial we need is computed by Taylor series expansion and weighted least squares formulation. In detail, we can approximate a function $u(x)$ at point $x_i$ to $p$th order accuracy by

$$u(x)=\sum_{k=0}^{p-1}\frac{u^{(k)}(x_{i+\frac{1}{2}})}{k!}\left(x-x_i\right)^k+\mathcal{O}(h^p). \tag{3.2.7}$$

Substituting each function value within the stencil, we can form a linear system

$$\boldsymbol{Av}\approx\boldsymbol{u}, \tag{3.2.8}$$

where $\boldsymbol{A}$ is a Vandermonde matrix, $\boldsymbol{v}$ is composed of the derivatives of the function

$u(x)$ at $x_{i+\frac{1}{2}}$, divided by the factorials, and $\boldsymbol{u}$ contains all the function values in the stencil. Furthermore, by solving this linear system, we are able to get all the function derivatives, which will allow us to evaluate the function value at any point in the control volume to high order accuracy.

To make our schemes stable, we need to apply weights for the rows of the linear system. For smooth functions, we define the weights as follows

$$\overline{\omega}_j = \begin{cases} 1/\overline{\beta}_j & j \neq i \\ \alpha/\overline{\beta}_j & j = i \end{cases}, \quad \text{with} \quad \overline{\beta}_j = \begin{cases} (x_j - x_i)^2 & j \neq i \\ \min\left\{\overline{\beta}_{j-1}, \overline{\beta}_{j+1}\right\} & j = i \end{cases}, \tag{3.2.9}$$

where $\alpha$ is a constant and added here for stability reason. For degree 2 and 5 schemes, $\alpha$ is chosen as 1.5. For degree 3 and 4, $\alpha = 0.25$.

In general, let $\boldsymbol{W}$ denote a diagonal matrix containing these weights, we solve the weighted least squares formulation, which has the form

$$\min \|\boldsymbol{W}\boldsymbol{A}\boldsymbol{v} - \boldsymbol{W}\boldsymbol{u}\|_2. \tag{3.2.10}$$

This problem can be solved by QR factorization with column pivoting, as we will describe in more detail in Section 3.3. After solving this problem, we obtain the derivatives of the function value at $x_i$, which can be used to approximate any point within the control volume. One thing to note is that once we solved the linear system, we can compute the coefficients of each node in the stencil so that we can save them for the same process we need later during our computation and improve the efficiency greatly. In detail, suppose we calculated the pseduinverse of the linear

system, then

$$u(x) \approx \left[1, x - x_i, ..., (x - x_i)^{p-1}\right] \boldsymbol{v} \tag{3.2.11}$$

$$\approx \left[1, x - x_i, ..., (x - x_i)^{p-1}\right] (\boldsymbol{W}\boldsymbol{A})^{+} \boldsymbol{W}\boldsymbol{u} \tag{3.2.12}$$

$$\approx \boldsymbol{c}\boldsymbol{u} \tag{3.2.13}$$

where **c** is the coefficient vector for the nodes in the stencil.

**WLS-ENO for Flux Derivatives**

After obtaining the function values and their derivatives, we need to compute the fluxes at flux points. For degree 2 scheme, we define flux point on the boundary of the control volume (3.2.2). For the schemes higher than 2, the flux points are the Gaussian-Lobatto points. Since the control volumes are connected, for flux points on the boundary of each control volume, we may have two different approximate values from left and right sides of the boundary point. We define $u^-$ and $u^+$ to be the values calculated from left and right respectively. We use numerical flux to make our schemes stable. In this paper, Lax-Fridriches scheme is adopted, i.e.,

$$f = \frac{1}{2}\left[f\left(u^-\right) + f\left(u^+\right) - \alpha\left(u^+ - u^-\right)\right], \tag{3.2.14}$$

where $\alpha = \max_u |f'(u)|$. Once we have all the fluxes ready for each control volume, we solve the linear system as what we did before except that the Taylor series expansion is done for all the flux points within control volumes and we do not apply any weights. By solving the linear system, we can get flux derivatives. Again, this can be accelerated if we save the coefficients for each flux point within the stencil

so that we do not need to compute the coefficients over and over again for smooth conditions.

### WLS-ENO for Discontinuous Functions in 1-D

The above technique works well for smooth conditions. For functions with discontinuities, we need to make some modifications to maintain non-oscillatory properties. First, for stencils near discontinuities. we gradually lower the degree of polynomials down to linear polynomials in our approximation to suppress oscillation. Also, we need to control the weights across discontinuities so that we maximize the use of smooth points and avoid the influence of discontinuous points. In detail, we apply the following weights to each row of the linear system (3.2.8)

$$\omega_j = \frac{1}{\beta_j}, \quad \text{with} \quad \beta_j = \begin{cases} (f_j - f_i)^2 + \epsilon & j \neq i \\ \min\{\beta_{j-1}, \beta_{j+1}\} & j = i \end{cases}, \tag{3.2.15}$$

where $\epsilon$ is a small number to avoid the denominator to be zero. In this paper, $\epsilon = 10^{-6}$. In fact, this set of weights give us more influence at the smooth points and less influence at discontinuous points, which makes the scheme stable and non-oscillatory, similar to WENO.

To obtain flux derivatives, we follow a similar technique. We use the same polynomial to construct a linear system as before except that we do not apply any weights. Each row of the linear system corresponds to a function value at flux points. The linear system is solved at each node to get flux derivatives.

To apply our strategies to various situations, we need some indicator to detect discontinuities. Here we design the non-differential indicator for each point of the computational domain. It can indicate whether a function is discontinuous at a

particular point or not. The non-differential indicator has the form

$$\alpha_{\mathrm{i}} = \begin{cases} 1, & \text{if} \quad \left|(f_i - f_{i-1})^2 \left(\frac{x_{i+1}-x_i}{x_i-x_{i-1}}\right)^2 - (f_{i+1} - f_i)^2\right| > \min\left\{(f_i - f_{i-1})^2, (f_{i+1} - f_i)^2\right\} \\ 0, & \text{otherwise} \end{cases} . \tag{3.2.16}$$

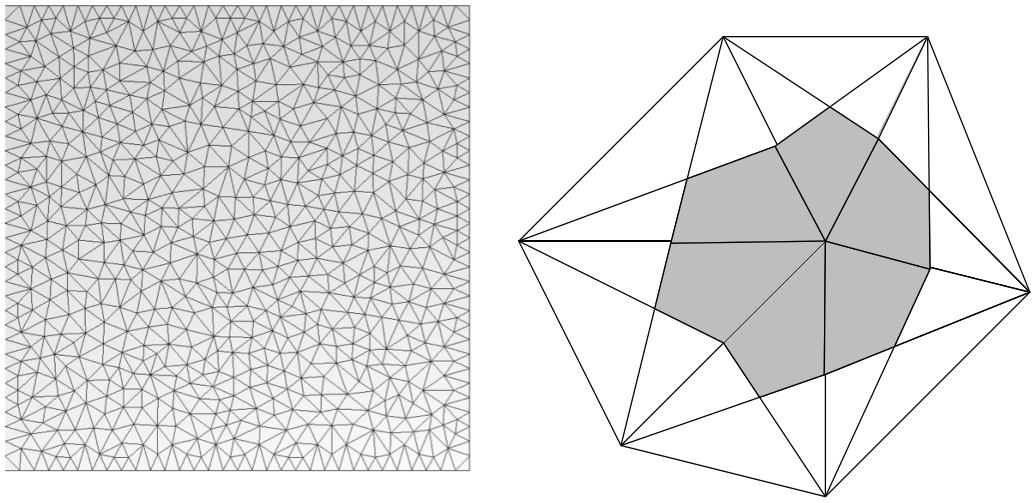
In fact, it is easy to see that if the function is smooth at point $x_i$, the left-hand side of the inequation is a higher order term than either term of the right-hand side. If the function is smooth on every point in the stencil, we use the technique in section 3.2.1 for reconstruction. Otherwise, we modify the weights for the linear system (3.2.8) to capture discontinuities

### 3.2.2 Generalization of WLS-ENO Schemes to 2-D

Our 1-D scheme can be generalized to unstructured meshes in 2-D. First, we introduce the concept of control volumes for each node on the mesh. Figure 3.2 shows an example of an unstructured mesh and the control volume of a node. The boundary of the control column consists of edges that connect the centroids of triangles and middle points of the edges linking to the central node. All control volumes should cover the whole computational domain.

The stencil we choose is adopted from [18]. Basically, we choose $k$-ring neighborhood with $1/2$-ring increments:

- The 1-ring neighbor faces of a vertex $v$ are the faces incident on $v$, and the 1-ring neighbor vertices are the vertices of these faces.
- The 1.5-ring neighbor faces are the faces that share an edge with a 1-ring neighbor face, and the 1.5-ring neighbor vertices are the vertices of these faces.



(a) An example of unstructured mesh. (b) An example of control volume

Figure 3.2: An example of an unstructured mesh and a control volume.

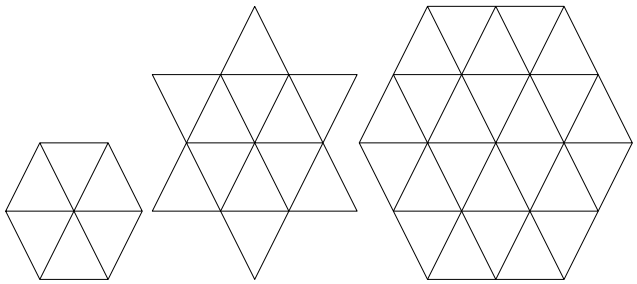


Figure 3.3: Examples of 1-ring, 1.5-ring and 2-ring neighborhood of a vertex.

- For an integer $k \geq 1$, the ($k$ + 1)-ring neighborhood of a vertex is the union of the 1-ring neighbors of its $k$-ring neighbor vertices, and the ($k$ + 1.5)-ring neighborhood is the union of the 1.5-ring neighbors of the $k$-ring neighbor vertices.

Figure 3.3 illustrates the neighborhood definitions up to 2 rings.

After selecting our stencil, we are able to build the linear system similarly as in

1-D case. The Taylor series expansion in 2-D has the form

$$u(x,y) = \sum_{q=0}^{p-1} \sum_{k,l \geq 0}^{k+l=q} \frac{\partial^q u\left(x_i, y_i\right)}{\partial x^k \partial y^l} \frac{\left(x - x_i\right)^k \left(y - y_i\right)^l}{k!l!} + \mathcal{O}\left(\|\delta\|^p\right), \tag{3.2.17}$$

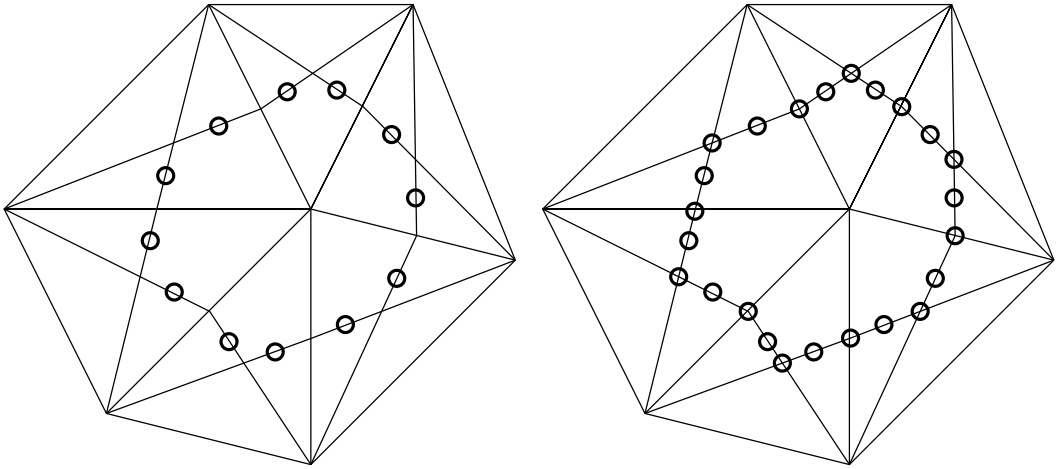
where $\boldsymbol{X}_i = (x_i, y_i)$ is the central node, $\delta = \max\left\{\left|x - x_i\right|, \left|y - y_i\right|\right\}$. $p$ is the order of accuracy. Substituting all the vertices in the equation (3.2.17), we can obtain a linear system in the form of

$$\boldsymbol{B}\boldsymbol{v} \approx \boldsymbol{u} \tag{3.2.18}$$

where $\boldsymbol{B}$ is the generalized Vandermonde matrix in 2-D. Again, we need to apply the weights for each row of the linear system to make our scheme stable. In this paper, we demonstrate the weights degree 2 and 3 schemes. Basically, we assign different weights for different rings. We define $R_0$ as the set of central node, $R_1$ as the set of 1-ring vertices, $R_i, i \geq 2$ as the set of vertices on the $i$th ring. The weights for smooth condition have the form as follows

$$\overline{\omega}_j = \begin{cases} 2, & \boldsymbol{X}_j \in R_0 \\ 1.5, & \boldsymbol{X}_j \in R_1 \\ 0.1, & \boldsymbol{X}_j \in R_2 \cup R_3 \end{cases}, \tag{3.2.19}$$

Again, we use the same technique as in 1-D case to solve this weighted least squares problems and obtain the function derivatives at the central node. We save the coefficients for each node in the stencil for efficiency.



(a) Flux points for degree 2 scheme. (b) Flux points for degree 3 scheme

Figure 3.4: Flux points for degree 2 and 3 schemes.

**WLS-ENO for Flux Derivatives in 2-D**

After building the polynomials, we focus on the computation of fluxes. The fluxes are computed on flux points within the control volume. Here, we explain the placement of flux points for degree 2 and 3 schemes. For degree-2 scheme, the flux points are placed in the middle of each boundary edge of control volume. For degree 3 scheme, we also add the corner points on the boundaries.

Figure 3.4 demonstrates the position points for degree 2 and 3 schemes. Obviously, when computing the function values on flux points, we have at least two different approximations from two control volumes sharing the same edge. At the corner, we have actually three different values. We employ a one-dimensional Riemann solver such as Lax-Friedrichs flux in the normal direction of each edge and require that the normal component of the flux vector across each edge are identical for two control volumes sharing the same edge. Suppose we define $u_L$ and $u_R$ as

two values from the current control volume and its neighbor receptively, we have

$$\boldsymbol{F} \cdot \boldsymbol{n} = \frac{1}{2} \left\{ \left[ \boldsymbol{F} \left( u_L \right) + \boldsymbol{F} \left( u_R \right) \right] \cdot \boldsymbol{n} - \alpha \left( u_R - u_L \right) \right\} \tag{3.2.20}$$

where $\boldsymbol{n}$ is the normal direction, $\alpha$ is taken as an upper bound for the eigenvalues of the Jacobian. For tangential direction $\boldsymbol{l}$, since it does not affect conservation property, we determine its component of flux vector as

$$\boldsymbol{F} \cdot \boldsymbol{l} = \frac{1}{2} \left\{ \left[ \boldsymbol{F} \left( u_L \right) + \boldsymbol{F} \left( u_R \right) \right] \cdot \boldsymbol{l} \right\} \tag{3.2.21}$$

For flux points in the middle of edges of the control volume, we combine equation (3.2.20) and (3.2.21) to solve for fluxes. For corner points where we may have two different normal directions $\boldsymbol{n}_1$ and $\boldsymbol{n}_2$, we use equation (3.2.20) with two different normals to get the fluxes.

Now we are ready to compute the flux derivatives on vertices. As in 1-D case, we build our linear system for all the fluxes within the control volume and solve the least squares problem to get the first order derivatives. The degree of the polynomial we use depends on the order of accuracy of function values from the previous step. Basically, we approximate the flux on the central vertex to the same order accuracy in Taylor series as we can obtain from reconstruction.

**WLS-ENO for Discontinuous Functions in 2-D**

Above strategy works well for smooth functions. For discontinuous functions, however, we need to design similar technique to suppress oscillations. To deal with

discontinuity, we solve the linear system (3.2.18) with different weights as follows

$$\omega_j = \frac{1}{\beta_j}, \quad \text{with} \quad \beta_j = \begin{cases} (u_{ij} - u_i)^2 + \|\boldsymbol{X}_i - \boldsymbol{X}_{ij}\|^2 & j \neq i \\ \min\{\beta_j, j \neq i\} & j = i \end{cases}, \tag{3.2.22}$$

Also, we lower our approximated polynomials according to the number of discontinuous points in the stencil. Typically, the number of continuous points should be 1.5 to 2 times bigger than the number of unknowns in the system. This can help us suppress oscillation and achieve good results.

In order to detect discontinuities, we need to define the non-differential indicator. For any vertex $\boldsymbol{X}_i$ in the domain and the vertices in its one-ring neighborhood $S_i = \{\boldsymbol{X}_{ij}, j = 1, 2, ..., k\}$, we have the form

$$(u_i - u_{ij})^2 = \left(\frac{\partial u}{\partial x}(x_{ij} - x_i) + \frac{\partial u}{\partial y}(y_{ij} - y_i)\right)^2 + \mathcal{O}\left(\|\delta\|^3\right). \tag{3.2.23}$$

To design a non-differential indicator, we need to eliminate the second order terms in (3.2.23). Therefore, suppose we assign each vertex in the one-ring neighborhood a coefficient $C_{ij}, j = 1, 2, ..., k$, they need to satisfy

$$\boldsymbol{D}\boldsymbol{c} = 0 \tag{3.2.24}$$

where $\boldsymbol{D}$ is a 3-by-$k$ matrix and $\boldsymbol{D}_{1j} = (x_{ij} - x_i)^2$,$\boldsymbol{D}_{2j} = (x_{ij} - x_i)(y_{ij} - y_i)$, $\boldsymbol{D}_{3j} = (y_{ij} - y_i)^2$, $\boldsymbol{c} = [C_{i1}, C_{i2}, ..., C_{ik}]^T$. In this way, we can expect that

$$\sum_{j=1}^{k} C_{ij}(u_i - u_{ij})^2 \tag{3.2.25}$$

is a higher order term than any terms in the lefthand side of (3.2.23). Thus we define our non-differential indicator as

$$\alpha_{\mathrm{i}} = \begin{cases} 1, & \text{if} \quad \sum_{j=1}^{k} C_{ij} \left(u_i - u_{ij}\right)^2 > \min\left\{err_1, err_2, ..., err_k\right\} \\ 0, & \text{otherwise} \end{cases}, \quad (3.2.26)$$

where $err_j = \left(u_i - u_{ij}\right)^2 + \left\|\boldsymbol{X}_i - \boldsymbol{X}_{ij}\right\|^2, j = 1, 2, ..., k$. To obtain these coefficients, we solve the following problem

$$\min_{C} \sum_{j=2}^{k} \left(C_{ij} - C_{i1}\right)^2 \quad (3.2.27)$$

such that equation (3.2.24) is satisfied and

$$\sum_{j=1}^{k} |C_{ij}| = 1. \quad (3.2.28)$$

This can help us identify discontinuities. The problem need to be solved only once for efficiency.

For time stepping, we use TVD Runge-Kutta method. TVD Runge-Kutta method is a typical method for solving hyperbolic conservation laws explicitly. Basically, an $n$-stage Runge-Kutta method for the ODE $u_t = L(u)$ has the general form of

$$k_0 = u(t), \quad (3.2.29)$$

$$k_i = \sum_{j=0}^{i-1} \left(\alpha_{ij} k_j + \beta_{ij} \Delta t L(k_j)\right), \qquad i = 1, \ldots, n, \quad (3.2.30)$$

where $k_i$ is the intermediate solution after the $i$th stage, and $u(t + \Delta t) = k_n$. A total

variation diminishing (TVD) Runge-Kutta method should satisfy the condition that all the coefficients $\alpha_{ij}$ and $\beta_{ij}$ are nonnegative. The CFL coefficient of such a scheme is given by

$$c = \min_{i,k}\{\alpha_{ik}/\beta_{ik}\}. \tag{3.2.31}$$

Specifically, what we use in this section and Chapter 5 is the third-order TVD Runge-Kutta scheme, given by

$$k_1 = u + \Delta t L\left(u\right), \tag{3.2.32}$$

$$k_2 = \frac{3}{4}u + \frac{1}{4}k_1 + \frac{1}{4}\Delta t L\left(k_1\right), \tag{3.2.33}$$

$$k_3 = \frac{1}{3}u + \frac{2}{3}k_2 + \frac{2}{3}\Delta t L(k_2), \tag{3.2.34}$$

for which the CFL coefficient is $c = 1$.

### 3.2.3 WLS-ENO Schemes for Euler System

For Euler system, the reconstruction step can be performed in either component by component fashion or characteristic decomposition fashion. The first technique has the advantage of efficiency but may not detect discontinuities as good as characteristic decomposition. However, the computational cost of characteristic decomposition is much higher than component-wise fashion. Throughout our experiments, we only use component-wise reconstruction for efficiency purpose.

## 3.3 Implementation Details

In this section we describe some of implementation details needed for our framework, including data structure and the solution for the weighted least squares problems.

### 3.3.1 Data Structure for Neighborhood Search

To support our framework and the construction of stencils in 2-D and 3-D, we use the data structured introduced in [11], which is called Array-based Half-Facet (AHF) data structure, to store the mesh information. In AHF, the term facet represents lower dimensional mesh entities. For example, in 2-D the facets are the edges, and in 3-D the facets are the faces. Every facet in a manifold mesh consists of two half-facets oriented in opposite direction. These two half-facets are called sibling half facets. On the boundary, half-facets have no siblings. We refer half-facets as half-edges and half-faces in 2-D and 3-D respectively. In order to store mesh information, we identify each half-facet by a two tuple: the element ID and a local facet ID within the element. For example in 2-D, we store element connectivity, sibling half-edges, and a mapping from each node to an incident half-edge. In 3-D, we store element connectivity, sibling half-faces, and a mapping from each node to an incident half-face. In this way, we can do neighborhood queries for a node in constant time. For more details, please refer to [11].

### 3.3.2 Solution of Weighted Least Squares Problems

In this paper, we solve weighted least squares problems by OR factorization with column pivoting. The basic idea is the following. Suppose we are aiming at finding

the solution that minimize

$$\min_{d} \|\boldsymbol{W}\boldsymbol{A}\boldsymbol{v} - \boldsymbol{W}\boldsymbol{u}\|_2 . \tag{3.3.1}$$

where $\boldsymbol{W}$ is the weighting matrix in (3.2.18). To improve the condition number of $\boldsymbol{A}$, we scale the columns of the matrix $\boldsymbol{W}\boldsymbol{A}$ ([18, 19]), and solve the following problem instead

$$\min_{d} \|\boldsymbol{W}\boldsymbol{A}\boldsymbol{S}\boldsymbol{d} - \boldsymbol{W}\boldsymbol{u}\|_2 . \tag{3.3.2}$$

where $\boldsymbol{d} \equiv \boldsymbol{S}^{-1}\boldsymbol{x}$. $\boldsymbol{S} = \operatorname{diag}\left(1/\left\|\tilde{\boldsymbol{a}}_1\right\|_2, 1/\left\|\tilde{\boldsymbol{a}}_2\right\|_2, \ldots, 1/\left\|\tilde{\boldsymbol{a}}_n\right\|_2\right)$, where $\tilde{\boldsymbol{a}}_i$ is the $i$th column vector of $\boldsymbol{W}\boldsymbol{A}$. We perform reduced QR factorization with column pivoting to the matrix $\boldsymbol{W}\boldsymbol{A}\boldsymbol{S}$:

$$\boldsymbol{W}\boldsymbol{A}\boldsymbol{S}\boldsymbol{E} = \boldsymbol{Q}\boldsymbol{R}. \tag{3.3.3}$$

Here $\boldsymbol{E}$ is chosen so that the diagonal of $\boldsymbol{R}$ is in decreasing order. If $\boldsymbol{W}\boldsymbol{A}\boldsymbol{S}$ has full rank, then its pseudoinverse is

$$(\boldsymbol{W}\boldsymbol{A}\boldsymbol{S})^{+} = \boldsymbol{E}\boldsymbol{R}^{-1}\boldsymbol{Q}^{T}. \tag{3.3.4}$$

Otherwise, the pseudoinverse is computed as

$$(\boldsymbol{W}\boldsymbol{A}\boldsymbol{S})^{+} = \boldsymbol{E}_{1:k,1:r}\boldsymbol{R}^{-1}_{1:r,1:r}(\boldsymbol{Q}_{1:m,1:r})^{T}, \tag{3.3.5}$$

where $r$ the numerical rank of $\boldsymbol{R}$. In this way, we can truncate the higher order terms in $\boldsymbol{W}\boldsymbol{A}\boldsymbol{S}$ for best-possible accuracy whenever possible.

# Chapter 4

# Accuracy and Stability Analysis of WLS-ENO

In this chapter, we will analyze the accuracy as well as the stability of WLS-ENO for both finite volume and finite difference methods by solving hyperbolic conservation laws.

## 4.1 Accuracy and Stability of WLS-ENO

### 4.1.1 Accuracy of WLS-ENO for Finite Volume Method

First, we analyze the WLS-ENO schemes and show that they can achieve the expected order of accuracy for smooth functions.

**Theorem 1.** *Given a mesh with a smooth function $f$. Let $\boldsymbol{W}$ be a diagonal matrix containing all the weights for the cells. $\boldsymbol{A}$ and $\boldsymbol{S}$ are the matrices in (3.3.2). Suppose the cell average of $f$ is approximated with an error $\mathcal{O}(h^p)$ and the matrix*

*$\boldsymbol{WAS}$ has a bounded condition number. The degree-$(p-1)$ cell average weighed least squares fitting approximates $q$th order derivatives of function $f$ to $\mathcal{O}(h^{p-q})$.*

For simplicity, we only prove the theorem in 2-D. The analysis also applies to 1-D and 3-D.

*Proof.* The 2-D Taylor series expansion about the point $(x_i, y_i)$ reads

$$f(x,y) = \sum_{q=0}^{p-1} \sum_{j,k\geq 0}^{j+k=q} \frac{f_{jk}}{j!k!}(x-x_i)^j (y-y_i)^k + \mathcal{O}\left(\|\delta \boldsymbol{x}\|^p\right), \tag{4.1.1}$$

where $\delta \boldsymbol{x} = [x - x_i, y - y_i]^T$. The cell average of $f(x,y)$ over some cell $\tau_i$ can be written as

$$\frac{1}{|\tau_i|}\iint_{\tau_i} f(x,y)\,dxdy = \frac{1}{|\tau_i|}\sum_{q=0}^{p-1}\sum_{j,k\geq 0}^{j+k=q} \frac{f_{jk}}{j!k!}\iint_{\tau_i}(x-x_i)^j(y-y_i)^k\,dxdy + \mathcal{O}\left(\|\delta \boldsymbol{x}\|^p\right). \tag{4.1.2}$$

Let $\boldsymbol{v}$ denote the exact derivatives of function $f$, $\tilde{\boldsymbol{v}}$ the numerical solution from the WLS fitting. Let $\boldsymbol{r} = \overline{\boldsymbol{u}} - \boldsymbol{A}\boldsymbol{v}$. By assumption, each component of $\boldsymbol{r}$ is $\mathcal{O}\left(\|\delta \boldsymbol{x}\|^p\right)$. The error of coefficients has the relationship $\boldsymbol{WA}\left(\tilde{\boldsymbol{v}} - \boldsymbol{v}\right) \approx \boldsymbol{W}\boldsymbol{r}$. The error of $\boldsymbol{d}$ can then be written as

$$\boldsymbol{WAS}\delta \boldsymbol{d} \approx \boldsymbol{W}\boldsymbol{r}. \tag{4.1.3}$$

By solving this least squares problem, we have $\delta \boldsymbol{d} = \left(\boldsymbol{WAS}\right)^+ \boldsymbol{W}\boldsymbol{r}$. Since the function $f$ is smooth, all the diagonal entries in $\boldsymbol{W}$ are $\mathcal{O}(1/\left\|\delta \boldsymbol{x}\right\|^2)$. Under the assumption that $\boldsymbol{WAS}$ has a bounded condition number $\kappa$, all the component of $\delta \boldsymbol{d}$ are $\mathcal{O}(\kappa \left\|\delta \boldsymbol{x}\right\|^{p-2})$. For a $q$th order partial derivative of function $f$, the corresponding column in $\boldsymbol{WA}$ is $\mathcal{O}(\|\delta \boldsymbol{x}\|^{q-2})$, so is the 2-norm of the column. Therefore, the $q$th order derivatives of function $f$ are approximated to $\mathcal{O}(h^{p-q})$. □

From this theorem, we can conclude that the degree-$(p-1)$ WLS-ENO reconstruction delivers a $p$th order accurate reconstruction for smooth functions.

### 4.1.2 Accuracy of WLS-ENO for Finite Difference Method

The accuracy of WLS-ENO schemes are guaranteed by Taylor series expansion and weighted least squares approximations. We prove the following theorem to show that they can achieve the expected order of accuracy for smooth functions.

**Theorem 2.** *Given a mesh with a smooth function $f$. Let $\boldsymbol{W}$ be a diagonal matrix containing all the weights for the vertices. $\boldsymbol{A}$ and $\boldsymbol{S}$ are the matrices in (3.3.2). Suppose $f$ is approximated with an error $\mathcal{O}(h^p)$ and the matrix $\boldsymbol{WAS}$ has a bounded condition number. The degree-($p-1$) weighed least squares fitting approximates $q$th order derivatives of function $f$ to $\mathcal{O}(h^{p-q})$.*

This theorem is similar to the proposition 1 in [19]. The idea is that since the residue of the linear system is high order term, and $\boldsymbol{WAS}$ has a bounded condition number, the difference between numerical and exact solution must also be a high order term. By this observation we can conclude that all the derivatives can be approximated to expected order of accuracy. We refer to readers to [19] for the proof. From this theorem, we can conclude that the degree-$(p-1)$ WLS-ENO reconstruction delivers a $p$th order accurate reconstruction for smooth functions. We also emphasize that this theorem does not limit to 2 dimensional space and it works for 1D and 3D.

## 4.2 Stability Analysis of WLS-ENO

For the WLS-ENO to be practically useful, it is important that it enables a stable discretization for hyperbolic conservation laws, when coupled with a proper time-integration scheme. In the following, we analyze two fifth-order WLS-ENO finite volume schemes and degree 3 and 5 WLS-ENO finite difference schemes on structured mesh for a model problem in 1-D, based on a modified von Neumann stability analysis. We show that the WLS-ENO finite volume and finite difference schemes have larger stability regions than the WENO scheme on structured meshes. We will further demonstrate its stability for higher dimension problems in Chapter 5.

### 4.2.1 von Neumann Stability Analysis in 1-D

We consider one dimensional wave equation

$$u_t + u_x = 0, \quad x \in [0,1], \quad t > 0 \tag{4.2.1}$$

with the periodic boundary condition

$$u(x,0) = u^0(x), \quad x \in [0,1]. \tag{4.2.2}$$

Suppose we have a structured grid $0 = x_0 < x_1 < \cdots < x_N = 1$ with $x_i = i\Delta x$ , $x_{i+\frac{1}{2}} = (x_i + x_{i+1})/2$ and $\Delta x = 1/N$. For different type of schemes, we use different formulation for flux derivatives. For the finite volume method, we integrate the above wave equation and divide it by the length of the cell, and obtain

$$\frac{d\overline{u}(x_i,t)}{dt} = -\frac{1}{\Delta x_i}\left(f\left(u\left(x_{i+\frac{1}{2}},t\right)\right) - f\left(u\left(x_{i-\frac{1}{2}},t\right)\right)\right), \tag{4.2.3}$$

where

$$\overline{u}\left(x_i,t\right)=\frac{1}{\Delta x_i}\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}}u(x,t)\,dx. \tag{4.2.4}$$

We approximate (4.2.3) by the following conservative scheme

$$\frac{d\overline{u}(x_i,t)}{dt}=-\frac{1}{\Delta x_i}\left(\hat{f}_{i+\frac{1}{2}}-\hat{f}_{i-\frac{1}{2}}\right), \tag{4.2.5}$$

where the numerical flux $\hat{f}_{i+\frac{1}{2}}$ is replaced by the Lax-Friedrichs flux (1.0.3), with $\alpha=\max_u\left|f'(u)\right|$. For this particular problem, $\hat{f}_{i+\frac{1}{2}}-\hat{f}_{i-\frac{1}{2}}=u^-_{i+\frac{1}{2}}-u^-_{i-\frac{1}{2}}$. For finite different schemes, We discretize the wave equation and obtain the semi-discretization

$$\frac{d}{dt}u_j\left(t\right)=-\frac{1}{\triangle x}L\left(u_{j-r},...,u_{j+s}\right). \tag{4.2.6}$$

Next, we explain von Neumann stability analysis with finite difference method. The analysis for finite volume schemes can be similarly derived. By von Neumann stability analysis, the semi-discrete solution can be written in a discrete Fourier series form

$$u_j(t)=\sum_{k=-N/2}^{N/2}\hat{u}_k(t)e^{i\omega_k j\Delta x},\quad \omega_k\in R. \tag{4.2.7}$$

According to the superposition principle, we can use only one term in the series for analysis

$$u_j(t)=\hat{u}_k(t)e^{ij\theta_k},\quad \theta_k=\omega_k\Delta x, \tag{4.2.8}$$

where $k=-N/2,\ldots,N/2$. We assume that the numerical flux can be expressed by the following form

$$L\left(u_{j-r},...,u_{j+s}\right)=z\left(\theta_k\right)u_i, \tag{4.2.9}$$

where the complex function $z\left(\theta_k\right)$ is the Fourier symbol.

Let $u_i^n = u_i(t^n)$ be the numerical solution at time level $t^n = n\Delta t$. We denote the amplification factor $g$ by substituting (4.2.8) into the fully-discrete system and obtain

$$u_i^{n+1} = g(\hat{z}_k)u_i^n, \quad \hat{z}_k = -\sigma z\left(\theta_k\right), \quad k = -N/2 \ldots N/2, \tag{4.2.10}$$

where $\sigma = \Delta t/\Delta x$. The linear stability domain of an explicit time-stepping scheme is thus $S_t = \{\hat{z} : |g\left(\hat{z}\right)| \leq 1\}$. Also, the discrete spectrum $S$ of a spatial discretization scheme can be defined as

$$S = \left\{-z\left(\theta_k\right) : \theta_k \in 0, \Delta\theta, 2\Delta\theta, \ldots, 2\pi\right\}, \quad \Delta\theta = 2\pi\Delta x. \tag{4.2.11}$$

The stability limit is thus the largest CFL number $\widetilde{\sigma}$ such that the rescaled spectrum $\widetilde{\sigma}S$ lies inside the stability domain

$$\widetilde{\sigma}S \in S_t. \tag{4.2.12}$$

For the third-order Runge-Kutta scheme, the amplification factor is given by

$$g\left(\widetilde{z}\right) = 1 + \widetilde{z} + \frac{1}{2}\widetilde{z}^2 + \frac{1}{6}\widetilde{z}^3. \tag{4.2.13}$$

The boundary of the stability domain $\partial S_t = \{\widetilde{z} : |g(\widetilde{z})| = 1\}$ can be determined by setting $g(\widetilde{z}) = e^{i\phi}$ and solving the following equation

$$\widetilde{z}^3 + 3\widetilde{z}^2 + 6\widetilde{z} + 6 - 6e^{i\phi} = 0. \tag{4.2.14}$$

Figure 4.1 showed the stability domain in solid blue curves.

### 4.2.2 Stability Analysis for WLS-ENO Finite Volume Schemes

#### Fifth-Order WLS-ENO Scheme with Five Cells

Let us first consider a fifth-order scheme for a cell $\tau_i$, with five cells

$$S_i = \left\{\tau_{i-2}, \tau_{i-1}, \tau_i, \tau_{i+1}, \tau_{i+2}\right\}, \tag{4.2.15}$$

where $\tau_i$ is the interval $\left[x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}\right]$. For this stencil, the coefficient matrix in (3.1.4) is given by

$$\boldsymbol{A} = \begin{pmatrix} 1 & -\frac{5\Delta x}{2} & \frac{19\Delta x^2}{6} & -\frac{65\Delta x^3}{24} & \frac{211\Delta x^4}{120} \\ 1 & -\frac{3\Delta x}{2} & \frac{7\triangle x^2}{6} & -\frac{15\Delta x^3}{24} & \frac{31\Delta x^4}{120} \\ 1 & -\frac{\Delta x}{2} & \frac{\Delta x^2}{6} & -\frac{\Delta x^3}{24} & \frac{\Delta x^4}{120} \\ 1 & \frac{\Delta x}{2} & \frac{\Delta x^2}{6} & \frac{\Delta x^3}{24} & \frac{\Delta x^4}{120} \\ 1 & \frac{3\Delta x}{2} & \frac{7\Delta x^2}{6} & \frac{15\Delta x^3}{24} & \frac{31\Delta x^4}{120} \end{pmatrix}. \tag{4.2.16}$$

Since $\boldsymbol{A}$ is nonsingular, the weights do not affect the solution. The solution is given by

$$u^-_{i+\frac{1}{2}} = \frac{2}{60}\overline{u}_{i-2} - \frac{13}{60}\overline{u}_{i-1} + \frac{47}{60}\overline{u}_i + \frac{27}{60}\overline{u}_{i+1} - \frac{3}{60}\overline{u}_{i+2}. \tag{4.2.17}$$

This is mathematically equivalent to the fifth-order WENO scheme without the nonlinear weights. Also, the flux reads

$$u^-_{i+\frac{1}{2}} - u^-_{i-\frac{1}{2}} = -\frac{2}{60}\overline{u}_{i-3} + \frac{15}{60}\overline{u}_{i-2} - \frac{60}{60}\overline{u}_{i-1} + \frac{20}{60}\overline{u}_i + \frac{30}{60}\overline{u}_{i+1} - \frac{3}{60}\overline{u}_{i+2}. \tag{4.2.18}$$

Substituting it into (4.2.8), we get

$$z\left(\theta_k\right)=\frac{16}{15}\sin^6\left(\frac{\theta_k}{2}\right)+i\left(-\frac{1}{6}\sin\left(2\theta_k\right)+\frac{4}{3}\sin\left(\theta_k\right)+\frac{16}{15}\sin^5\left(\frac{\theta_k}{2}\right)\cos\left(\frac{\theta_k}{2}\right)\right). \tag{4.2.19}$$

The discrete spectrum is shown in Figure 4.1(left).

Given the spectrum and the stability domain, the CFL number of this scheme can be computed by finding the largest rescaling parameter $\sigma$, so that the rescaled spectrum still lies in the stability domain. Using interval bisection, we find that the CFL number is $\sigma = 1.44$ for the fifth-order five-cell scheme.

**Fifth-Order WLS-ENO Scheme with Seven Cells**

Next, let us consider a least-squares fitting for $\tau_i$ using seven cells

$$S_i=\left\{\tau_{i-3},\tau_{i-2},\tau_{i-1},\tau_i,\tau_{i+1},\tau_{i+2},\tau_{i+3}\right\}. \tag{4.2.20}$$

The coefficient matrix in (3.1.4) is given by

$$\boldsymbol{A}=\begin{pmatrix} 1 & -\frac{7\Delta x}{2} & \frac{37\Delta x^2}{6} & -\frac{175\Delta x^3}{24} & \frac{781\Delta x^4}{120} \\ 1 & -\frac{5\Delta x}{2} & \frac{19\Delta x^2}{6} & -\frac{65\Delta x^3}{24} & \frac{211\Delta x^4}{120} \\ 1 & -\frac{3\Delta x}{2} & \frac{7\triangle x^2}{6} & -\frac{15\Delta x^3}{24} & \frac{31\Delta x^4}{120} \\ 1 & -\frac{\Delta x}{2} & \frac{\Delta x^2}{6} & -\frac{\Delta x^3}{24} & \frac{\Delta x^4}{120} \\ 1 & \frac{\Delta x}{2} & \frac{\Delta x^2}{6} & \frac{\Delta x^3}{24} & \frac{\Delta x^4}{120} \\ 1 & \frac{3\Delta x}{2} & \frac{7\Delta x^2}{6} & \frac{15\Delta x^3}{24} & \frac{31\Delta x^4}{120} \\ 1 & \frac{5\Delta x}{2} & \frac{19\Delta x^2}{6} & \frac{65\Delta x^3}{24} & \frac{211\Delta x^4}{120} \end{pmatrix}. \tag{4.2.21}$$

If $\epsilon = 0$, from (3.1.7) and (3.1.8), we obtain the weights

$$\begin{aligned} &w_1 \approx \tfrac{1}{9\Delta x^2}, \quad w_2 \approx \tfrac{1}{4\Delta x^2}, \quad w_3 \approx \tfrac{1}{\Delta x^2}, \quad w_4 \approx \tfrac{1.5}{\Delta x^2}, \quad w_5 \approx \tfrac{1}{\Delta x^2} \\ &w_6 \approx \tfrac{1}{4\Delta x^2}, \quad w_7 \approx \tfrac{1}{9\Delta x^2} \end{aligned} \tag{4.2.22}$$

Solving the weighted least squares system, we obtain the following scheme

$$\begin{aligned} u^-_{i+\frac{1}{2}} =& \frac{1226983}{9489680}\overline{u}_{i-3} - \frac{963431}{47447340}\overline{u}_{i-4} - \frac{13515169}{94894680}\overline{u}_{i-1} + \frac{66771}{87380}\overline{u}_i + \\ & \frac{38388551}{94894680}\overline{u}_{i+1} - \frac{93404}{11861835}\overline{u}_{i+2} - \frac{348299}{31631560}\overline{u}_{i+3}. \end{aligned}$$

Therefore, the flux reads

$$\begin{aligned} u^-_{i+\frac{1}{2}} - u^-_{i-\frac{1}{2}} =& -\frac{1226983}{94894680}\overline{u}_{i-4} + \frac{630769}{18978936}\overline{u}_{i-3} + \frac{3862769}{31631560}\overline{u}_{i-2} \\ & - \frac{17205695}{18978936}\overline{u}_{i-1} + \frac{6824951}{18978936}\overline{u}_i + \frac{13045261}{31631560}\overline{u}_{i+1} \\ & + \frac{59533}{18978936}\overline{u}_{i+2} - \frac{348299}{31631560}\overline{u}_{i+3.} \end{aligned}$$

Substituting it into (4.2.8), we get

$$z\left(\theta_k\right) = p\left(\theta_k\right) + iq\left(\theta_k\right), \tag{4.2.23}$$

where

$$\begin{aligned} p\left(\theta_k\right) = \quad & -0.0129\cos\left(4\theta_k\right) + 0.0222\cos\left(3\theta_k\right) + 0.1253\cos\left(2\theta_k\right) \\ & -0.4942\cos\left(\theta_k\right) + 0.3596, \end{aligned} \tag{4.2.24}$$

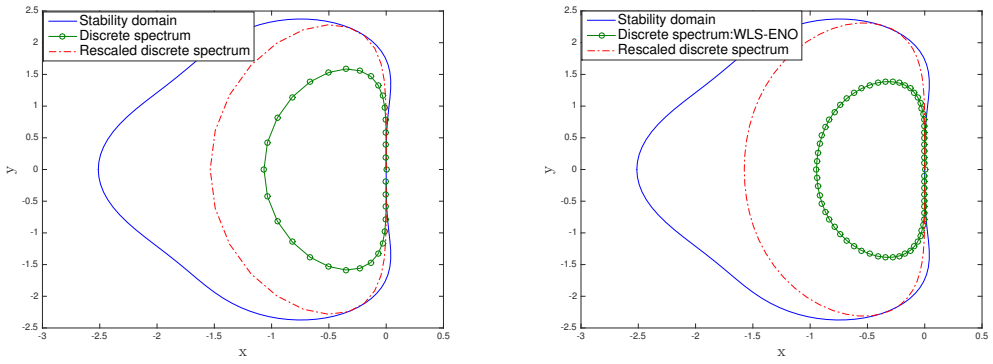



Figure 4.1: Rescaled spectra and stability domains of fifth-order WLS-ENO with five (left) and seven (right) cells.

and

$$q\left(\theta_k\right) = 0.0129\sin\left(4\theta_k\right) - 0.0442\sin\left(3\theta_k\right) - 0.1190\sin\left(2\theta_k\right) + 1.3190\sin\left(\theta_k\right). \tag{4.2.25}$$

The discrete spectrum is shown in Figure 4.1(right). The CFL number is computed in the same way as the above scheme. For this scheme with seven cells, we obtain $\sigma = 1.67$. Therefore, the seven-cell fifth-order WLS-ENO scheme has a larger stability region than the five-cell counterpart, which is mathematically equivalent to the fifth-order WENO scheme without nonlinear weights.

### 4.2.3 Stability Analysis for WLS-ENO Finite Difference Schemes

#### Degree 3 WLS-ENO scheme

Next, let us consider degree 3 WLS-ENO scheme with the stencil

$$S_i = \left\{x_{i-3}, x_{i-2}, x_{i-1}, x_i, x_{i+1}, x_{i+2}, x_{i+3}\right\}, \tag{4.2.26}$$

The linear system is given by

$$\boldsymbol{A} = \begin{pmatrix} 1 & -3\triangle x & 9\triangle x^2 & -27\triangle x^3 \\ 1 & -2\triangle x & 4\triangle x^2 & -8\triangle x^3 \\ 1 & -\triangle x & \triangle x^2 & -\triangle x^3 \\ 1 & 0 & 0 & 0 \\ 1 & \triangle x & \triangle x^2 & \triangle x^3 \\ 1 & 2\triangle x & 4\triangle x^2 & 8\triangle x^3 \\ 1 & 3\triangle x & 9\triangle x^2 & 27\triangle x^3 \end{pmatrix}. \tag{4.2.27}$$

Again, we solve this linear system with the weights (3.2.9) and compute the function values on flux points

$$\begin{aligned} u^-_{i+\frac{1}{2}} = &- \frac{441573}{22003808}u_{i-3} - \frac{241521}{5500952}u_{i-2} + \frac{6030169}{22003808}u_{i-1} + \frac{3447}{60784}u_i \\ &+ \frac{17617119}{22003808}u_{i+1} - \frac{304489}{11001904}u_{i+2} - \frac{874659}{22003808}u_{i+3}. \end{aligned}$$

Suppose we define $u_1^*$ and $u_2^*$ to be the approximated value at flux point $-\sqrt{\frac{1}{5}}$ and $\sqrt{\frac{1}{5}}$ respectively, we have

$$\begin{aligned} u_1^* = &- 0.0423u_{i-3} - 0.0459u_{i-2} + 0.6761u_{i-1} + 0.0625u_i \\ &+ 0.4334u_{i+1} - 0.0518u_{i+2} - 0.0320u_{i+3}. \end{aligned}$$

$$u_2^* = -0.0320u_{i-3} - 0.0518u_{i-2} + 0.4334u_{i-1} + 0.0625u_i$$
$$+ 0.6761u_{i+1} - 0.0459u_{i+2} - 0.0423u_{i+3}.$$

Therefore, the flux reads

$$\begin{aligned} L =& 0.25u_{i-\frac{1}{2}} - 0.25u_{i+\frac{1}{2}} - 2.7951u_1^* + 2.7951u_2^* \\ =& -0.0050u_{i-4} + 0.0229u_{i-3} + 0.0630u_{i-2} - 0.7329u_{i-1} \\ &+ 0.1860u_i + 0.4715u_{i+1} + 0.0135u_{i+2} - 0.0189u_{i+3.} \end{aligned}$$

Substituting it into (4.2.8), we get

$$\begin{aligned} \mathrm{Re}\left(z\left(\theta_k\right)\right) &= -0.0401\cos\left(x\right)^4 + 0.0159\cos\left(x\right)^3 + 0.1931\cos\left(x\right)^2 \\ &\quad -0.25\cos\left(x\right) + 0.1045 \end{aligned}$$

$$\mathrm{Im}\left(z\left(\theta_k\right)\right) = 1.2045\sin\left(x\right) - 0.0495\sin\left(2x\right) - 0.0418\sin\left(3x\right) + 0.005\sin\left(4x\right) \tag{4.2.28}$$

The stability region and its discrete spectrum is shown in Figure 4.2 (left). The CFL number is computed in the same way as the above scheme. For this scheme, we obtain $\sigma = 1.65$. As a comparison, the CFL number for 3rd order WENO finite difference scheme is 1.63. We can see that degree 3 WLS-ENO scheme has a comparable CFL number.

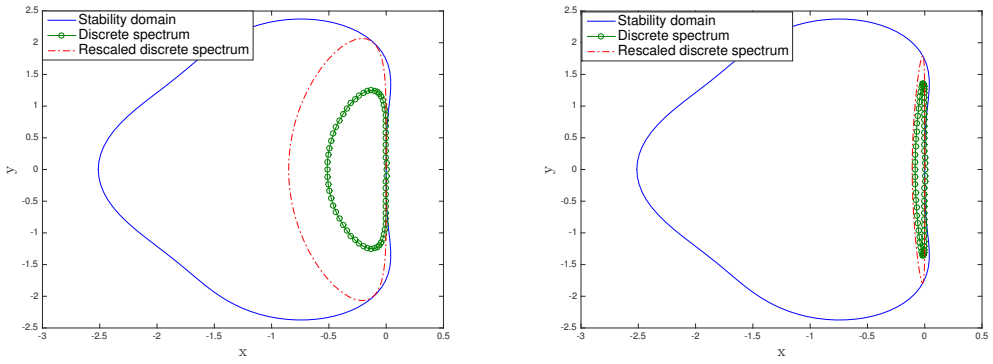



Figure 4.2: Rescaled spectrum and stability domains of degree 3 WLS-ENO scheme (left) and degree 5 WLS-ENO scheme (right).

**Degree 5 WLS-ENO Scheme**

Let us first consider a degree 5 WLS-ENO scheme for a point $x_i$, with its stencil

$$S(i) = \{x_{i-4}, x_{i-3}, x_{i-2}, x_{i-1}, x_i, x_{i+1}, x_{i+2}, x_{i+3}, x_{i+4}\}, \tag{4.2.29}$$

According to (3.2.8) , the system has the following form

$$\boldsymbol{A} = \begin{pmatrix} 1 & -4\Delta x & 16\Delta x^2 & -64\Delta x^3 & 256\Delta x^4 & -1024\Delta x^5 \\ 1 & -3\Delta x & 9\Delta x^2 & -27\Delta x^3 & 81\Delta x^4 & -243\Delta x^5 \\ 1 & -2\Delta x & 4\Delta x^2 & -8\Delta x^3 & 16\Delta x^4 & -32\Delta x^5 \\ 1 & -\Delta x & \Delta x^2 & -\Delta x^3 & \Delta x^4 & -\Delta x^5 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & \Delta x & \Delta x^2 & \Delta x^3 & \Delta x^4 & \Delta x^5 \\ 1 & 2\Delta x & 4\Delta x^2 & 8\Delta x^3 & 16\Delta x^4 & 32\Delta x^5 \\ 1 & 3\Delta x & 9\Delta x^2 & 27\Delta x^3 & 81\Delta x^4 & 243\Delta x^5 \\ 1 & 4\Delta x & 16\Delta x^2 & 64\Delta x^3 & 256\Delta x^4 & 1024\Delta x^5 \end{pmatrix}. \tag{4.2.30}$$

Suppose the function is smooth within the stencil. By applying the constant weights (3.2.9), we can approximate the function values on the flux points of the boundary

$$\begin{aligned} u^-_{i+\frac{1}{2}} =& -0.0021u_{i-4} + 0.0092u_{i-3} + 0.0042u_{i-2} - 0.1291u_{i-1} + 0.6803u_i \\ &+ 0.4689u_{i+1} - 0.0196u_{i+2} - 0.0158u_{i+3} + 0.0040u_{i+4}. \end{aligned}$$

We define $u^*_i, i = 1, 2, 3, 4$ to be the flux points in the control volume, we can obtain the flux

$$\begin{aligned} L =& -0.125u_{i-\frac{1}{2}} + 0.125u_{i+\frac{1}{2}} + 0.509u^*_1 - 4.433u^*_2 + 4.433u^*_3 - 0.509u^*_4 \\ =& 0.0003u_{i-5} - 0.009u_{i-4} + 0.0290u_{i-3} + 0.0513u_{i-2} - 0.6641u_{i-1} \\ &+ 0.0264u_i + 0.6240u_{i+1} + 0.0351u_{i+2} - 0.0308u_{i+3} + 0.0078u_{i+4}. \end{aligned}$$

Substituting it into (4.2.8), we get

$$\begin{aligned} \mathrm{Re}\left(z\left(\theta_k\right)\right) =& 0.0041\cos^5\left(x\right) - 0.0072\cos^4\left(x\right) - 0.0126\cos^3\left(x\right) + 0.0395\cos^2\left(x\right) \\ &- 0.0332\cos\left(x\right) + 0.0093. \end{aligned}$$

$$\begin{aligned} \mathrm{Im}\left(z\left(\theta_k\right)\right) =& -2.5832 \times 10^{-4}\sin\left(5x\right) + 0.0164\sin\left(4x\right) - 0.598\sin\left(3x\right) \\ &- 0.0865\sin\left(2x\right) + 1.2880\sin\left(x\right). \end{aligned}$$

After obtaining the spectrum and the stability domain, we rescale the spectrum so that it touches the stability domain and the rescaling factor is 1.3145. In com-

parison, the CFL number for 5th order WENO scheme is 1.43. Figure 4.2 (right) shows the stability region and the discrete spectrum.

Here we omit the stability analysis of degree 2 and 4 schemes because they follow the same procedure. We just emphasize that they both can enable a relatively large stability domain.

# Chapter 5

# Numerical Experiments

In this chapter, we present some numerical experiments of both WLS-ENO finite volume and finite difference schemes in 1-D, 2-D and 3-D, and compare it against WENO schemes on both structured and unstructured meshes when applicable. For all the numerical test, we use third-order TVD Runge-Kutta for time integration.

## 5.1 Numerical Experiments for WLS-ENO Finite Volume Schemes

### 5.1.1 1-D Results

We first show some results in 1-D, for the reconstruction of a piecewise smooth function as well as the solutions of PDEs, including a linear wave equation, Burgers' equation, and the Euler equations.

**Reconstruction of Discontinuous Functions**

We first test WLS-ENO for the reconstruction of a 1-D discontinuous function, given by

$$v(x) = \begin{cases} \sin(\pi x) & 0 \le x \le 0.6 \\ \cos(\pi x) & 0.6 < x \le 1 \end{cases} . \tag{5.1.1}$$

This function is discontinuous at $x = 0.6$ but smooth everywhere else within the interval $[0, 1]$. We performed grid convergence study under grid refinement, starting from an equidistant grid with 32 grid cells. For the WLS-ENO, we used degree-four polynomials over seven-cell stencil, which according to our theory should deliver fifth-order accuracy in smooth regions and fourth-order accuracy near discontinuities. As a point of reference, we also perform the reconstruction using the fifth-order WENO scheme, which is fifth-order accurate in smooth regions and third-order accurate near discontinuities. Figure 5.1 shows the $L_\infty$-norm error for the reconstructed values at the grid points that are one cell away from the discontinuity. It can be seen that both WLS-ENO and WENO delivered fifth-order accuracy, but WLS-ENO is more accurate. When including the grid points near discontinuities, as can be seen in Figure 5.2, WLS-ENO achieved fourth-order convergence, whereas the fifth-order WENO reduced to third order, as predicted by their respective theoretical analyses.

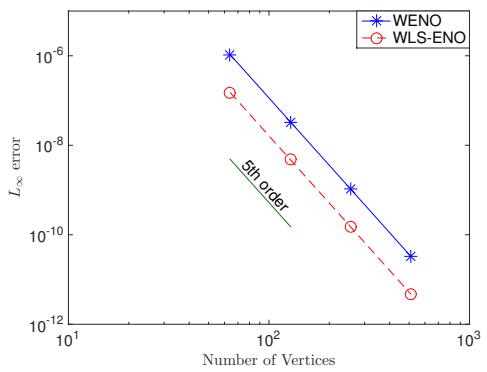



Figure 5.1: Convergence of fifth-order WENO and WLS-ENO away from discontinuity.

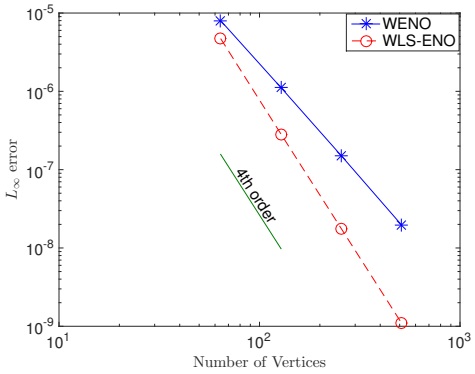



Figure 5.2: Convergence of fifth-order WENO and WLS-ENO near discontinuity.

### 1-D Wave Equation

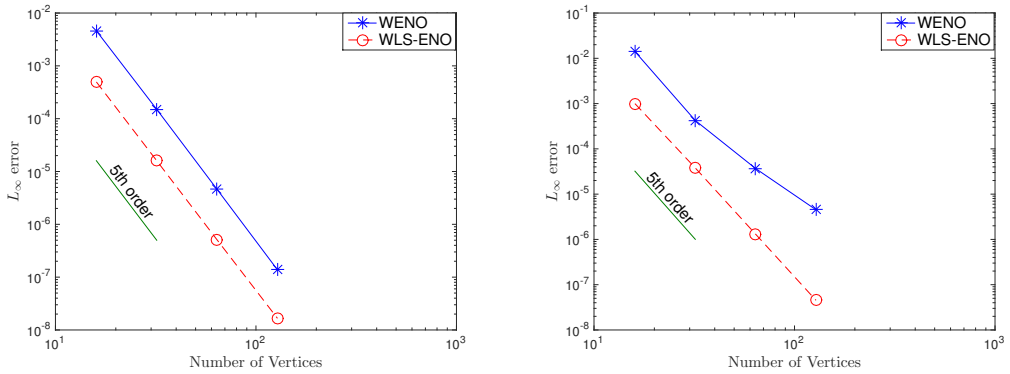
To test the effectiveness of WLS-ENO for solving hyperbolic PDEs, we first solve a simple linear wave equation

$$u_t + u_x = 0, \qquad -1 \leq x \leq 1, \tag{5.1.2}$$

with periodic boundary conditions. Similar to the reconstruction problem, we use WLS-ENO with degree-four polynomials over seven cells. To assess the accuracy for smooth solutions, consider the smooth initial condition

$$u(x, 0) = \sin(\pi x), \tag{5.1.3}$$

for which the solution remains smooth over time. We assess the order of accuracy of the solutions at $t = 1$ under grid refinement, and compare the errors against the fifth-order WENO. Figure 5.3 shows the results for uniform and non-uniform grids. For




(a) Errors on uniform grids.

(b) Errors on non-uniform grids.

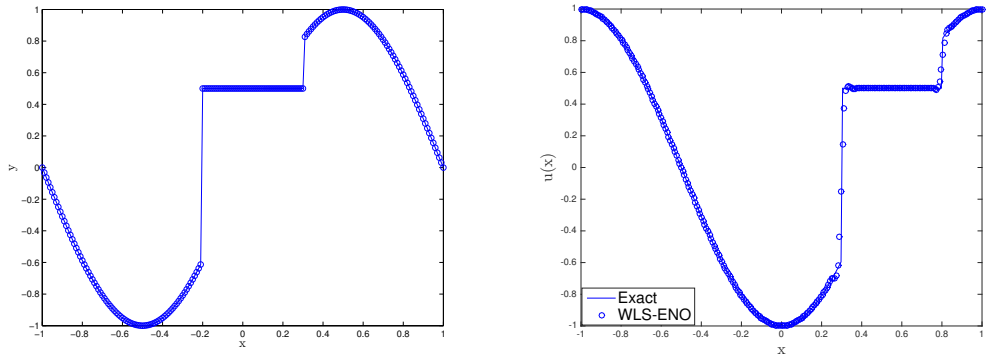
Figure 5.3: Convergence of fifth-order WENO and WLS-ENO for wave equation at $t = 1$ on 1-D grids.

uniform grids, similar to the results of reconstruction, both WLS-ENO and WENO delivered fifth-order convergence under grid refinement, and the solution of WLS-ENO is more accurate. For non-uniform grids, we used the WENO reconstruction in [32], which converged at a slower rates, and was about an order of magnitude less accurate than WLS-ENO on the finest grid.

To demonstrate the accuracy and stability of WLS-ENO for discontinuous solutions, we change the initial condition to be a piecewise smooth function

$$u(x,0) = \begin{cases} \sin(\pi x) & -1 \le x < -0.2 \cup 0.3 < x \le 1 \\ 0.5 & -0.2 \le x \le 0.3 \end{cases}, \tag{5.1.4}$$

as shown in Figure 5.4(a). Figure 5.4(b) shows the solution at $t = 0.5$ using WLS-ENO. The results show that the WLS-ENO scheme well preserved the sharp feature and maintained the non-oscillatory property.




(a) Initial condition. (b) Numerical solution at $t = 0.5$.

Figure 5.4: Discontinuous initial condition (left) and numerical solution with fifth-order WLS-ENO (right) at $t = 0.5$ for the linear wave equation.

### 1-D Burgers' Equation

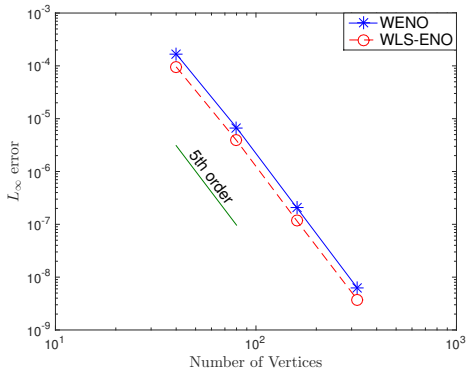
Next, we test WLS-ENO with the 1-D Burgers' equation,

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} = 0,\ 0 \le x \le 2\pi, \tag{5.1.5}$$

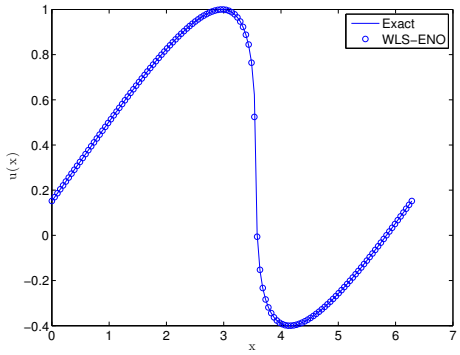
with periodic boundary conditions and the initial condition

$$u(x, 0) = 0.3 + 0.7\sin(x),\ 0 \le x \le 2\pi. \tag{5.1.6}$$

Although the initial condition is smooth, a discontinuity develops at time $t = 1.4$. To assess the order of accuracy, Figure 5.5(a) shows the solutions from the fifth-order WLS-ENO, compared with the fifth-order WENO under grid refinement at time $t = 1$, starting from a grid with 64 grid points. The results show that both WLS-ENO and WENO delivered fifth-order accuracy, while WLS-ENO is more accurate. Figure 5.5(b) shows the numerical solution from WLS-ENO overlaid on top of the exact solution at $t = 1.4$. We can see that the WLS-ENO scheme




(a) Errors on uniform grids at $t = 1$.




(b) Numerical and exact solution at $t = 1.4$.

Figure 5.5: Comparison of WENO and WLS-ENO schemes for 1-D Burgers' equation at $t = 1$ (left) and numerical solution with WLS-ENO (right) at $t = 1.4$.

approximated the solution very well.

### 1-D Euler Equations

The above tests demonstrate the accuracy and stability of WLS-ENO for 1-D benchmark problems. For a more realistic problem, we consider the 1-D Euler equations

$$\begin{pmatrix} \rho \\ \rho v \\ E \end{pmatrix}_t + \begin{pmatrix} \rho v \\ \rho v^2 + p \\ v\,(E+p) \end{pmatrix}_x = \mathbf{0}, \tag{5.1.7}$$

with the equation of state for ideal polytropic gas

$$E = \frac{p}{\gamma - 1} + \frac{1}{2}\rho v^2, \tag{5.1.8}$$

where $\rho$ denotes the gas density, $v$ the velocity, $p$ the pressure, $E$ the energy, and $\gamma = 1.4$ a constant specific to air. We perform characteristic decomposition [21] and solve the conservation law characteristic-wise using the fifth-order WLS-ENO

scheme on an unstructured (i.e., non-uniform) grid. In more detail, if we introduce the speed of sound $c$ by

$$c = \sqrt{\frac{\gamma p}{\rho}}, \tag{5.1.9}$$

and enthalpy $H$ by

$$H = \frac{E + p}{\rho}, \tag{5.1.10}$$

we have the eigenvalue decomposition for the Jacobian as

$$R^{-1}(u) f'(u) R(u) = \Lambda(u), \tag{5.1.11}$$

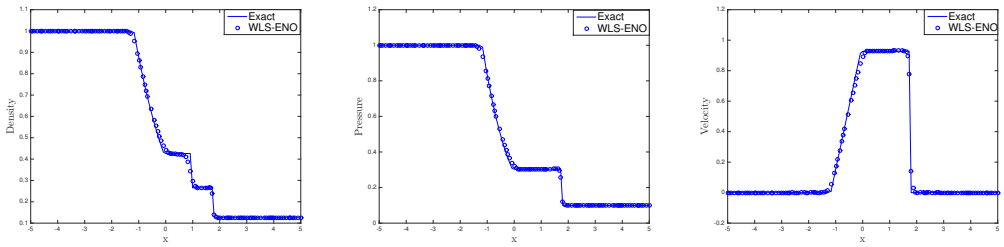
where

$$f'(u) = \begin{pmatrix} 0 & 1 & 0 \\ \left(\frac{\gamma-3}{2}\right) v^2 & (3-\gamma)v & \gamma - 1 \\ \left(\frac{\gamma-1}{2}\right) v^3 - vH & H - (\gamma-1)v^2 & \gamma v \end{pmatrix}, \tag{5.1.12}$$

$$\Lambda(u) = \begin{pmatrix} v - c & & \\ & v & \\ & & v + c \end{pmatrix}, \tag{5.1.13}$$

and

$$R(u) = \begin{pmatrix} 1 & 1 & 1 \\ v - c & v & v + c \\ H - cv & \frac{1}{2}v^2 & H + cv \end{pmatrix}. \tag{5.1.14}$$




(a) Density. (b) Pressure. (c) Velocity.

Figure 5.6: Solutions of 1-D Euler equations at $t = 1$ with fifth-order WLS-ENO on non-uniform grid.

**Sod's Problem** We first compute the density, velocity and pressure with the initial condition given by Sod's problem [32]

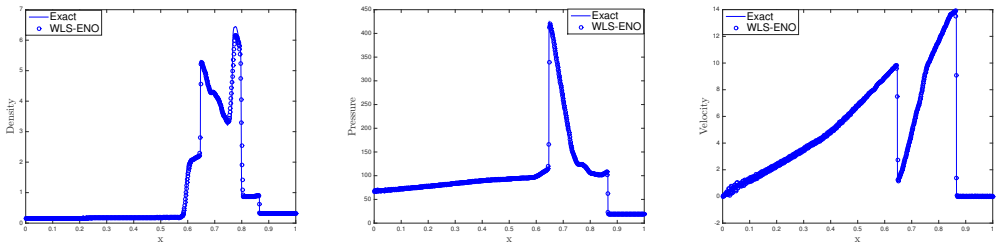
$$(\rho_L, v_L, p_L) = (1, 0, 1)\,, (\rho_R, v_R, p_R) = (0.125, 0, 0.1)\,. \tag{5.1.15}$$

Figure 5.6 shows the numerical solutions at $t = 1$ compared against the exact solution. The results matched the exact solution very well and were non-oscillatory near discontinuities.

**Interacting Blast Waves** Next, we consider the 1-D blast wave problem [42], which has the initial condition

$$(\rho, u, P) = \begin{cases} (1, 0, 1000) & 0 \le x < 0.1 \\ (1, 0, 0.01) & 0.1 \le x < 0.9\,, \\ (1, 0, 100) & 0.9 \le x < 1 \end{cases} \tag{5.1.16}$$

and reflective boundary conditions at both sides. For this test, sharp resolution of discontinuities is critical for the accuracy of the overall flow solution. Figure 5.7




(a) Density. (b) Pressure. (c) Velocity.

Figure 5.7: Solutions of 1-D interacting blast waves at $t = 0.038$ with fifth-order WLS-ENO.

shows the result using fifth-order WLS-ENO scheme at $t = 0.038$, which agreed with the exact solution very well.

### 5.1.2 2-D Results

We now present results of WLS-ENO with 2-D unstructured meshes for problems with smooth or piecewise smooth solutions, including the wave equation, Burgers' equation, and the Euler equations with two different initial conditions.

#### 2-D Wave Equation

As in 1-D, we first consider the wave equation,

$$u_t + u_x + u_y = 0, \tag{5.1.17}$$

with periodic boundary conditions and the initial condition

$$u_0(x, y) = \sin\left(\frac{\pi}{2}(x + y)\right), \qquad -2 \le x \le 2,\ -2 \le y \le 2. \tag{5.1.18}$$

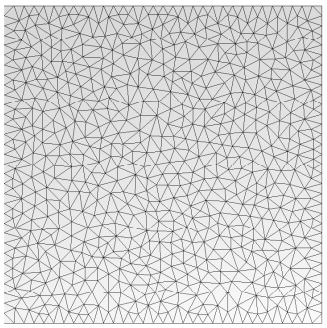


Figure 5.8: Sample unstructured triangular mesh for solving 2-D wave equation.

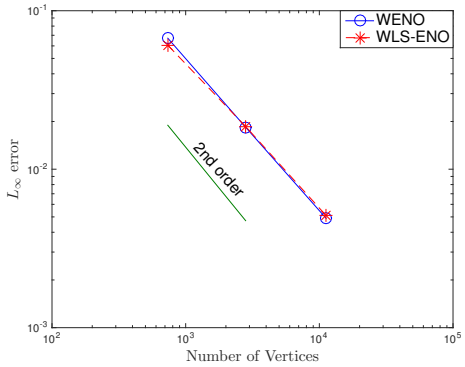



Figure 5.9: Errors in numerical solutions of 2-D wave equation with WENO and WLS-ENO at $t = 1$.

For this problem, the solution remains smooth over time. We solve the problem using third-order WLS-ENO scheme and third-order WENO scheme. Figure 5.9 shows the errors at $t = 1$, and it can be seen that both methods achieved only second order convergence. This convergence rate is expected, because the derivatives can only be approximated to second-order accurate by polynomial approximations over nonuniform unstructured meshes without symmetry. When applying WLS-ENO on a uniform mesh, such as that shown in Figure 5.10, it would deliver the convergence rate one order higher due to error cancellation, similar to WENO and other finite difference methods, as illustrated with the fourth-order WLS-ENO scheme and fourth-order WENO scheme in Figure 5.11. Note that on uniform meshes, WLS-ENO may be slightly less accurate than WENO because it uses a larger stencil.

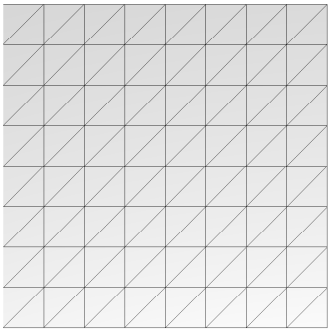


Figure 5.10: Sample uniform triangular mesh for solving 2-D wave equation.

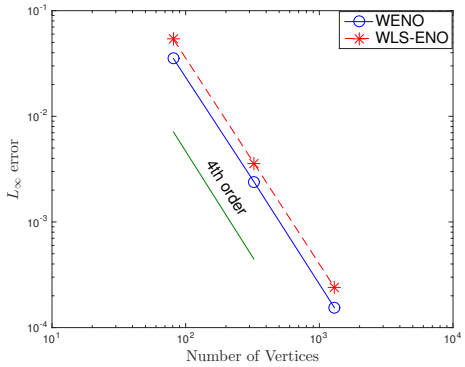



Figure 5.11: Errors in numerical solutions of 2-D wave equation with WENO and WLS-ENO on uniform meshes.

### 2-D Burgers' Equation

For piecewise smooth solutions, we solve the 2-D Burgers' equation

$$u_t + \left(\frac{u^2}{2}\right)_x + \left(\frac{u^2}{2}\right)_y = 0 \tag{5.1.19}$$

over $[-2, 2]^2$, with periodic boundary conditions and the initial condition

$$u_0(x, y) = 0.3 + 0.7 \sin\left(\frac{\pi}{2}(x + y)\right). \tag{5.1.20}$$

Although the initial condition is smooth, discontinuities develop over time. Figure 5.12(left) shows the exact solution at $t = 0.5$, when the solution becomes discontinuous, and Figure 5.12(right) shows the result of fourth-order WLS-ENO scheme under non-uniform grid refinement. It can be seen that the overall solution remained accurate as discontinuities developed.

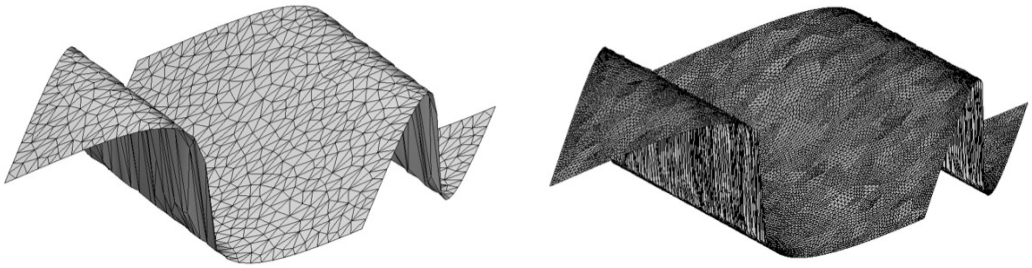


Figure 5.12: Exact (left) and numerical solutions (right) with fourth-order WLS-ENO for 2-D Burgers' equation at $t = 0.5$.

**2-D Euler Equations**

The 2-D Euler equations have the following form

$$\begin{pmatrix} \rho \\ \rho u \\ \rho v \\ E \end{pmatrix}_t + \begin{pmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ u\left(E+p\right) \end{pmatrix}_x + \begin{pmatrix} \rho v \\ \rho uv \\ \rho v^2 + p \\ v\left(E+p\right) \end{pmatrix}_y = \mathbf{0}, \tag{5.1.21}$$

where

$$E = \frac{p}{\gamma - 1} + \frac{1}{2}\rho\left(u^2 + v^2\right). \tag{5.1.22}$$

In our tests, we use $\gamma = 1.4$ as in 1-D, and use characteristic decomposition to split variables as described in [15]. WLS-ENO is then applied to each of the characteristic field.

**Vortex Evolution Problem** This is one of the few problems that has exact solutions for the compressible Euler equations. The test case involves the convection of an isentropic vortex in inviscid flow, and it tests the ability of numerical schemes to capture vortical flows. We consider an idealized setting over $[0, 10]^2$ with periodic

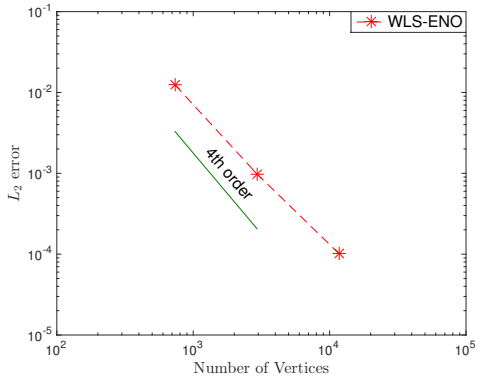



Figure 5.13: Convergence result of $\rho$ for 2-D vortex evolution problem with fifth order WLS-ENO on triangular meshes.

boundary conditions. The mean flow is $\rho_\infty = 1$, $p_\infty = 1$, and $(u_\infty, v_\infty) = (1, 1)$. For the initial condition, we place an isentropic vortex to the mean flow field. The perturbation values are given by

$$(\delta u, \delta v) = \frac{\beta}{2\pi} e^{\frac{1-r^2}{2}} (-\overline{y}, \overline{x}), \tag{5.1.23}$$

$$\delta T = -\frac{(\gamma - 1)\beta^2}{8\gamma\pi^2} e^{1-r^2} \tag{5.1.24}$$

where $(\overline{x}, \overline{y}) = (x - 5, y - 5)$, $r^2 = \overline{x}^2 + \overline{y}^2$, and the vortex strength $\beta = 5$. The exact solution of this problem convecting the vortex along the diagonal direction. We used fifth-order WLS-ENO scheme and computed the results up to $t = 1$. Figure 5.13 shows the result, which achieved the expected convergence rate.

**Explosion Test Problem** In this test, we solve the 2-D explosion test [10, 20], which solves the Euler equations over a unit disk centered at the origin. The initial

condition is given by

$$(\rho, u, v, p)^T = \begin{cases} (1, 0, 0, 1)^T & \sqrt{x^2 + y^2} \leq 0.2 \\ (0.125, 0, 0.1)^T & \sqrt{x^2 + y^2} > 0.2 \end{cases}. \tag{5.1.25}$$

We triangulated the unit disk with meshes similar to that in Figure 5.8, and ran the test up to $t = 0.1$ to ensure that the explosion waves do not reach the boundary. To obtain a reference solution, note that this problem is mathematically equivalent to the axisymmetric Euler equations [36]

$$\frac{\partial}{\partial t} \begin{pmatrix} \rho \\ \rho u \\ E \end{pmatrix} + \frac{\partial}{\partial r} \begin{pmatrix} \rho u \\ \rho u^2 + p \\ u\,(E + p) \end{pmatrix} = -\frac{d-1}{r} \begin{pmatrix} \rho u \\ \rho u^2 \\ u\,(E + p) \end{pmatrix}, \tag{5.1.26}$$

where $r$ is the radial coordinate. We solved this 1-D problem on a very fine mesh composed of 4,000 grid points and use its solution as the reference. Figures 5.14 and 5.15 show the numerical solution for the density at $t = 0.1$ with the third-order WLS-ENO scheme. The results agreed very well with the 1-D solutions.

### 5.1.3 3-D Results

One advantage of WLS-ENO is that it generalizes to 3-D in a straightforward fashion. We present some numerical results over unstructured meshes in 3-D, including the wave equation, Burgers' equation, and the Euler equations.

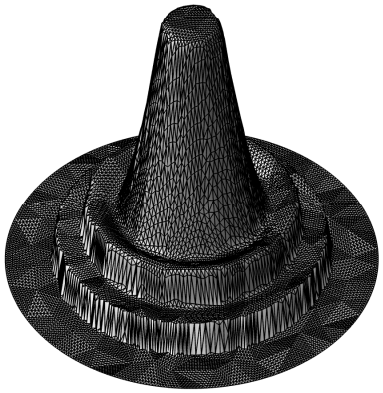


Figure 5.14: Numerical solution of $\rho$ for 2-D explosion test with third-order WLS-ENO at $t = 0.1$.

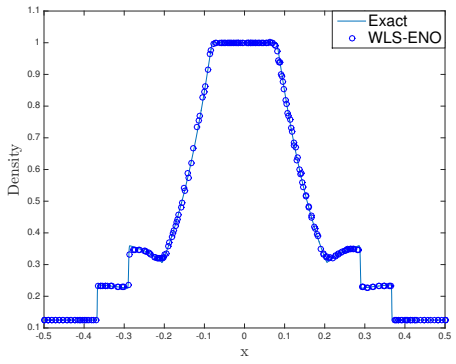



Figure 5.15: Numerical solution of $\rho$ along $x$ axis of 2-D explosion test with third-order WLS-ENO vs. 1-D solution at $t = 0.1$.

### 3-D Wave Equation

We first solve the 3-D linear wave equation

$$u_t + u_x + u_y + u_z = 0 \tag{5.1.27}$$

over $[-2, 2]^3$, with periodic boundary conditions and the initial condition

$$u(x, y, z, 0) = \sin\left(\frac{\pi}{2}(x + y + z)\right). \tag{5.1.28}$$

We solve the problem using WLS-ENO over a series of unstructured meshes, where the coarsest mesh is depicted in Figure 5.16. Figure 5.17 shows the errors with third-order and fourth-order WLS-ENO schemes under mesh refinement. It is clear that both schemes achieved the convergence rate close to three, and the error of fourth-order WLS-ENO scheme was about half of that of the third-order scheme.

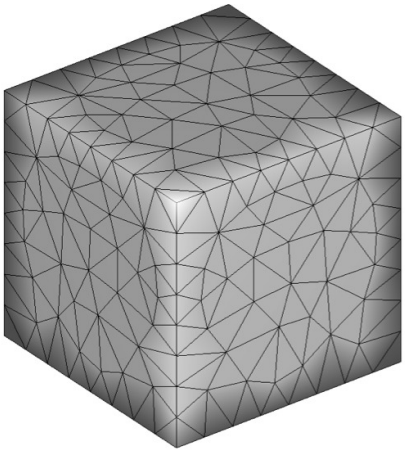


Figure 5.16: Sample unstructured mesh for solving 3-D wave equations.

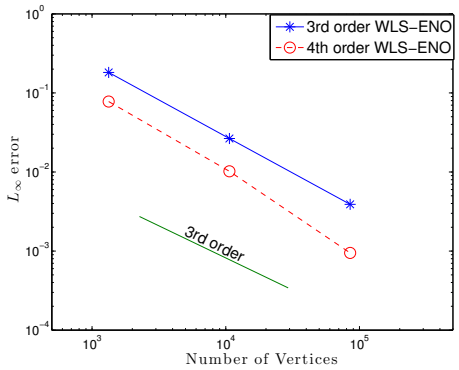



Figure 5.17: Convergence of third- and fourth-order WLS-ENO schemes for wave equation on tetrahedral meshes.

### 3-D Burgers' Equation

In this test, we solve the 3-D nonlinear Burgers' equation

$$u_t + \left(\frac{u^2}{2}\right)_x + \left(\frac{u^2}{2}\right)_y + \left(\frac{u^2}{2}\right)_z = 0 \tag{5.1.29}$$

over $[-2, 2]^3$, also with periodic boundary conditions and the initial condition

$$u(x, y, z, 0) = 0.3 + 0.7 \sin\left(\frac{\pi}{2}(x + y + z)\right). \tag{5.1.30}$$

Similar to the 2-D case, discontinuities develop at $t = 0.5$. Figure 5.18 shows a 1-D cross section of the numerical solutions along $x = y$ and $z = 0$ with third-order and fourth-order WLS-ENO schemes at $t = 0.5$, overlaid with the exact solution. Both solutions are non-oscillatory. In contrast, the third-order WENO scheme in

[47] was unstable over a non-uniform unstructured mesh. On a tetrahedral mesh obtained by decomposing a structured mesh, WLS-ENO and WLS-ENO achieved comparable accuracy for the problem.

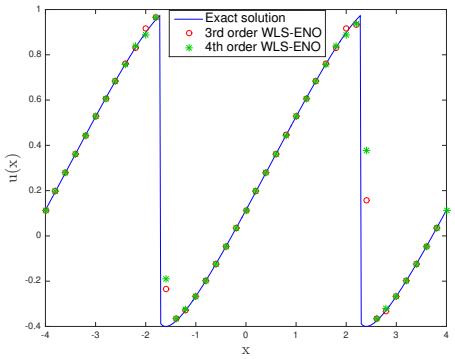



Figure 5.18: 1-D cross-sections of 3-D Burgers' equation using third- and fourth-order WLS-ENO schemes at $t = 0.5$.

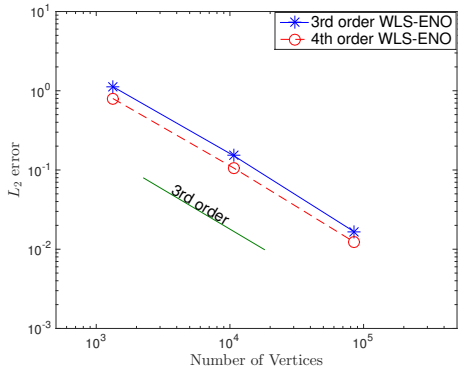



Figure 5.19: Convergence of third- and fourth-order WLS-ENO for Burgers' equation at $t = 0.5$ away from singularities.

### 3-D Euler Equations

As our final test, we solve the 3-D version of the explosion test [10, 20]. The 3-D Euler equations have the form

$$U_t + \frac{\partial F}{\partial x} + \frac{\partial G}{\partial y} + \frac{\partial H}{\partial z} = 0, \tag{5.1.31}$$

where

$$U = \begin{pmatrix} \rho \\ \rho u \\ \rho v \\ \rho w \\ E \end{pmatrix}, \quad F = \begin{pmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ \rho uw \\ u\left(E + p\right) \end{pmatrix}, \quad G = \begin{pmatrix} \rho v \\ \rho uv \\ \rho v^2 + p \\ \rho vw \\ v\left(E + p\right) \end{pmatrix}, \quad H = \begin{pmatrix} \rho w \\ \rho uw \\ \rho vw \\ \rho w^2 + p \\ w\left(E + p\right) \end{pmatrix}, \tag{5.1.32}$$

and

$$E = \frac{p}{\gamma - 1} + \frac{1}{2}\rho\left(u^2 + v^2 + w^2\right). \tag{5.1.33}$$

As in 1-D and 2-D, we chose $\gamma = 1.4$. The computational domain is a unit ball centered at the origin, which we tessellate with a tetrahedral mesh. The initial condition of this problem is given by

$$\left(\rho, u, v, w, p\right)^T = \begin{cases} \left(1, 0, 0, 0, 1\right)^T & \sqrt{x^2 + y^2 + z^2} \leq 0.2 \\ \left(0.125, 0, 0, 0, 0.1\right)^T & \sqrt{x^2 + y^2 + z^2} > 0.2 \end{cases}. \tag{5.1.34}$$

We solved the problem in a component-by-component fashion up to $t = 0.1$, and obtained a reference solution by solving the 1-D problem (5.1.26). Figures 5.20 and 5.21 show the numerical solutions of the density at $t = 0.1$, which agreed with the 1-D solutions very well.

Figure 5.20: Cross section of numerical solution of $\rho$ in $xy$ plane of 3-D explosion test with third-order WLS-ENO at $t = 0.1$.

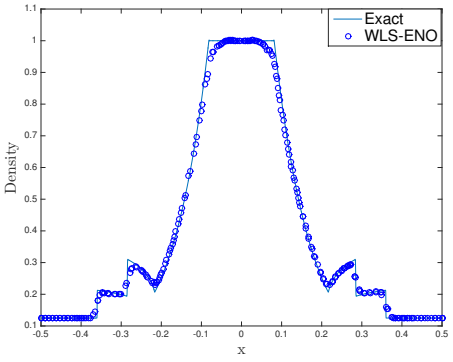



Figure 5.21: Numerical solution of $\rho$ along $x$ axis of 3-D explosion test with third-order WLS-ENO vs. 1-D solution at $t = 0.1$.

## 5.2 Numerical Experiments for WLS-ENO Finite Difference Schemes

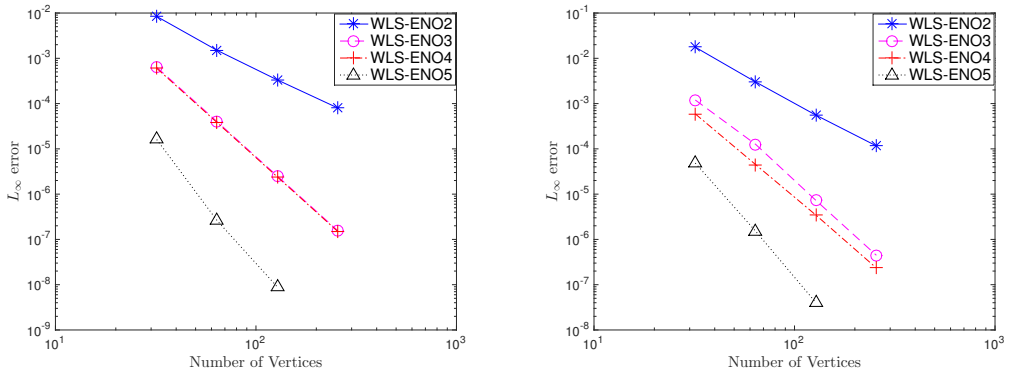
### 5.2.1 1-D Results

We first solve hyperbolic problems in 1-D for both structured and unstructured mesh, including a linear wave equation, Burgers' equation, and Euler equations.

#### 1-D Wave Equation

The first problem we solve is a simple hyperbolic linear wave equation

$$u_t + u_x = 0, \qquad -1 \leq x \leq 1, \tag{5.2.1}$$




(a) Errors on uniform grids at $t = 1$.

(b) Errors on unstructured grids at $t = 1$.

Figure 5.22: WLS-ENO schemes for 1-D wave equation at $t = 1$ on structured grid (left) and unstructured grid (right).

with periodic boundary conditions. We use WLS-ENO schemes from degree 2 to degree 5 on both structured and unstructured mesh. To obtain the accuracy for smooth solutions, we consider the following smooth initial condition

$$u(x, 0) = \sin(\pi x), \tag{5.2.2}$$

We assess the order of accuracy of the solutions at $t = 1$ under grid refinement. Figure 5.22 shows the results for uniform and non-uniform grids. For even degree schemes, they achieved the accuracy with the same order as their degrees. However, for odd degree schemes, they achieved one order higher than the order of degree they use. This is due to error cancellation.

We further compared the efficiency of this scheme with the well-known WENO schemes for finite difference and WLS-ENO finite volume schemes [24]. From Figure 5.23 we can see that WLS-ENO3 and WLS-ENO5 for GFD are as efficient as 3rd order and 5th orderWENO schemes for finite difference respectively and all the finite difference schemes are more efficient than WLS-ENO scheme for finite

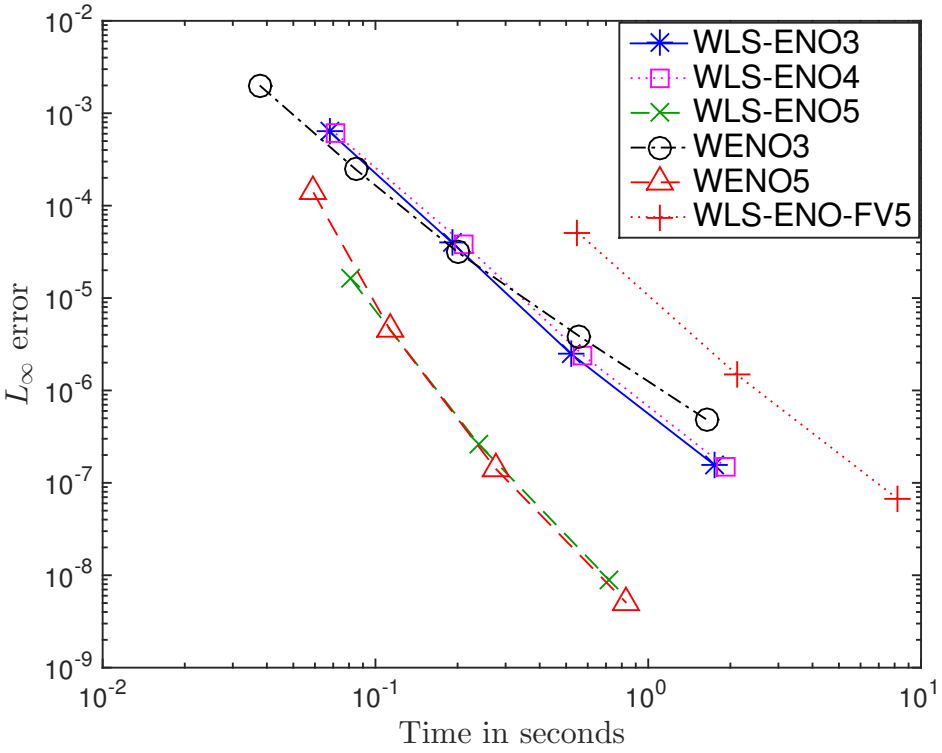


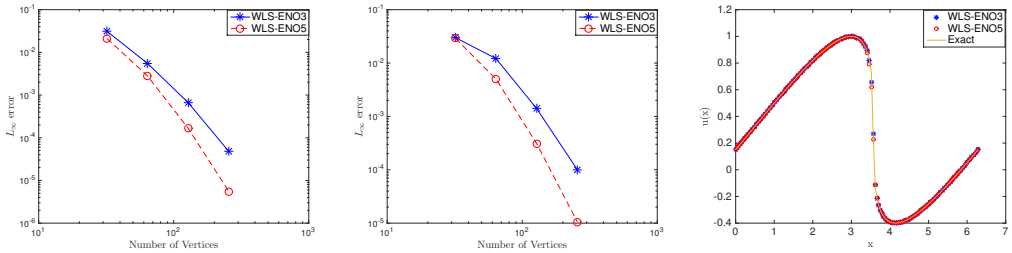

Figure 5.23: Timing comparison in 1D.

volume. However, WLS-ENO enjoys the benefit of being applicable to unstructured meshes. Another thing to notice is that errors for high order schemes tend to decrease faster than low order schemes.

### 1-D Burgers' Equation

Next, we use WLS-ENO schemes to solve 1-D Burgers' equation,

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} = 0,\ 0 \leq x \leq 2\pi, \tag{5.2.3}$$




(a) Errors on uniform grids at $t = 1$.

(b) Errors on unstructured mesh at $t = 1$.

(c) WLS-ENO schemes for Burger's equation at $t = 1.4$.

Figure 5.24: Numerical solutions of WLS-ENO schemes for 1D Burger's equation.

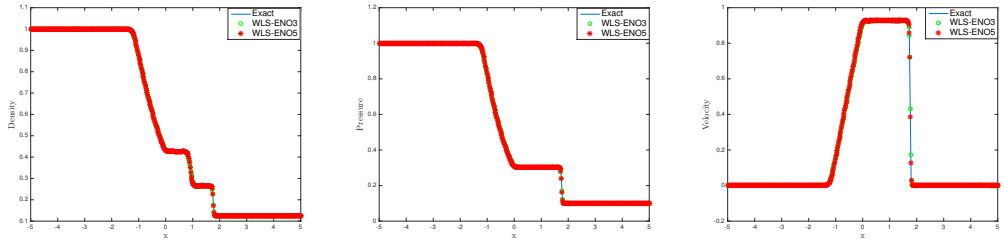
with periodic boundary conditions and the following initial condition

$$u(x, 0) = 0.3 + 0.7 \sin(x),\ 0 \leq x \leq 2\pi. \tag{5.2.4}$$

The solution can develop discontinuities at time $t = 1.4$. To assess the order of accuracy, Figure 5.24(a) and Figure 5.24(b) shows the degree 3 and 5 WLS-ENO schemes for both structured and unstructured mesh at $t = 1$ when the function is still smooth. We can see the convergence is similar to 1-D wave equation. Figure 5.24(c) shows the numerical solutions from WLS-ENO with the exact solution at $t = 1.4$. We can see that WLS-ENO schemes maintained non-oscillatory property and approximated the solution very well.

### 1-D Euler Equations: Sod's Problem

We have shown the accuracy and stability of WLS-ENO schemes for 1-D scalar problems. To assess the performance for 1-D system, we consider the 1-D Euler




(a) Density. (b) Pressure. (c) Velocity.

Figure 5.25: Solutions of 1-D Euler equations at $t = 1$ by WLS-ENO schemes on non-uniform grid.

equation

$$\begin{pmatrix} \rho \\ \rho v \\ E \end{pmatrix}_t + \begin{pmatrix} \rho v \\ \rho v^2 + p \\ v\left(E + p\right) \end{pmatrix}_x = \mathbf{0}, \tag{5.2.5}$$

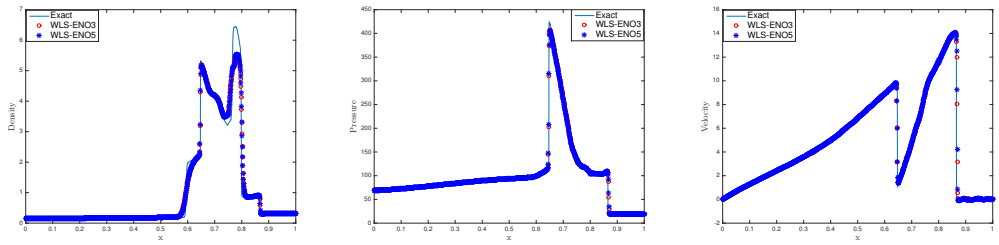
with the equation of state for ideal polytropic gas

$$E = \frac{p}{\gamma - 1} + \frac{1}{2}\rho v^2, \tag{5.2.6}$$

where $\rho$ denotes the gas density, $v$ the velocity, $p$ the pressure, $E$ the energy, and $\gamma = 1.4$ a constant specific to air. We solve this problem using the degree 3 and 5 WLS-ENO schemes on an unstructured (i.e., non-uniform) grid and compare the results with exact solution. The initial condition of Sod's problem [32] is given by

$$\left(\rho_L, v_L, p_L\right) = \left(1, 0, 1\right), \left(\rho_R, v_R, p_R\right) = \left(0.125, 0, 0.1\right) \tag{5.2.7}$$

Figure 5.25 shows the numerical solutions against the exact solution at $t = 1$. It demonstrates the good approximation of numerical solutions to the exact solution and non-oscillation near discontinuities.




(a) Density. (b) Pressure. (c) Velocity.

Figure 5.26: Solutions of 1-D interacting blast waves at $t = 0.038$ by fifth order WLS-ENO.

**Interacting Blast Waves**

This is a test problem specially designed by Woodward and Collela [42] to illustrate the strong relationship between the accuracy of the overall flow solution and the sharp resolution of discontinuities. It has the following the initial condition:

$$(\rho, u, P) = \begin{cases} (1, 0, 1000) & 0 \leq x < 0.1 \\ (1, 0, 0.01) & 0.1 \leq x < 0.9. \\ (1, 0, 100) & 0.9 \leq x < 1 \end{cases} \tag{5.2.8}$$

Reflective boundary conditions are applied at both sides. For this example, we use degree 3 and 5 WLS-ENO schemes and plot the results at time $t = 0.038$. Figure 5.26 shows excellent agreement of the numerical solution with the exact solution.

### 5.2.2 2-D Results

For 2-D cases, we use degree 2 and 3 WLS-ENO schemes with unstructured meshes for problems with smooth or piecewise smooth solutions, including the wave equa-

tion, Burgers' equation, and the Euler equations with two different initial conditions.

### 2-D Wave Equation

As in 1-D, we first consider the wave equation,

$$u_t + u_x + u_y = 0, \tag{5.2.9}$$

with periodic boundary conditions and the initial condition

$$u_0(x, y) = \sin\left(\frac{\pi}{2}(x + y)\right), \qquad -2 \le x \le 2,\ -2 \le y \le 2. \tag{5.2.10}$$

This is a test for accuracy purpose. We solve the problem using degree 2 and 3 WLS-ENO schemes as mentioned in Section 3.2.2. Figure 5.28 shows the errors at $t = 1$. We can see that the schemes achieved 2nd and 3rd order accuracy respectively, as expected.

We also did the timing comparison against 3rd order WENO scheme with degree 2 and 3 WLS-ENO schemes. From Figure 5.29 we can see that the slopes for 3rd order WENO scheme and degree 2 WLS-ENO scheme are basically the same, except that WLS-ENO for generalized finite difference is faster. Also, WLS-ENO for finite volume is the slowest. This is because the linear system is bigger than WENO schemes and it needs to solve the system every time for each stencil. Another thing to notice is that higher order scheme is more efficient than lower order scheme. This is important if we want to obtain solutions within certain accuracy.

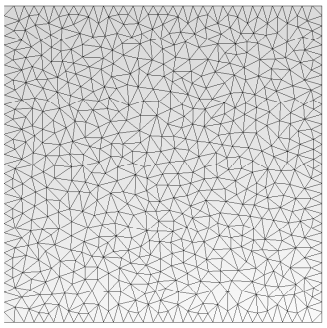


Figure 5.27: A sample unstructured triangular mesh for solving 2-D wave equation.

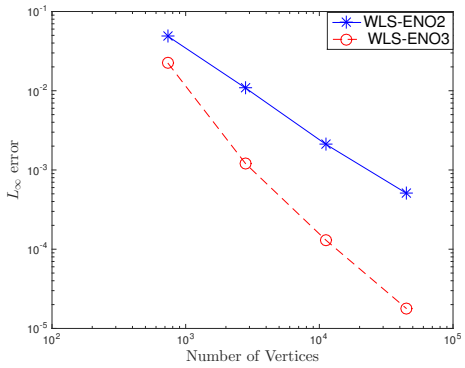



Figure 5.28: Errors of solutions of 2-D wave equation with WLS-ENO at $t = 1$.

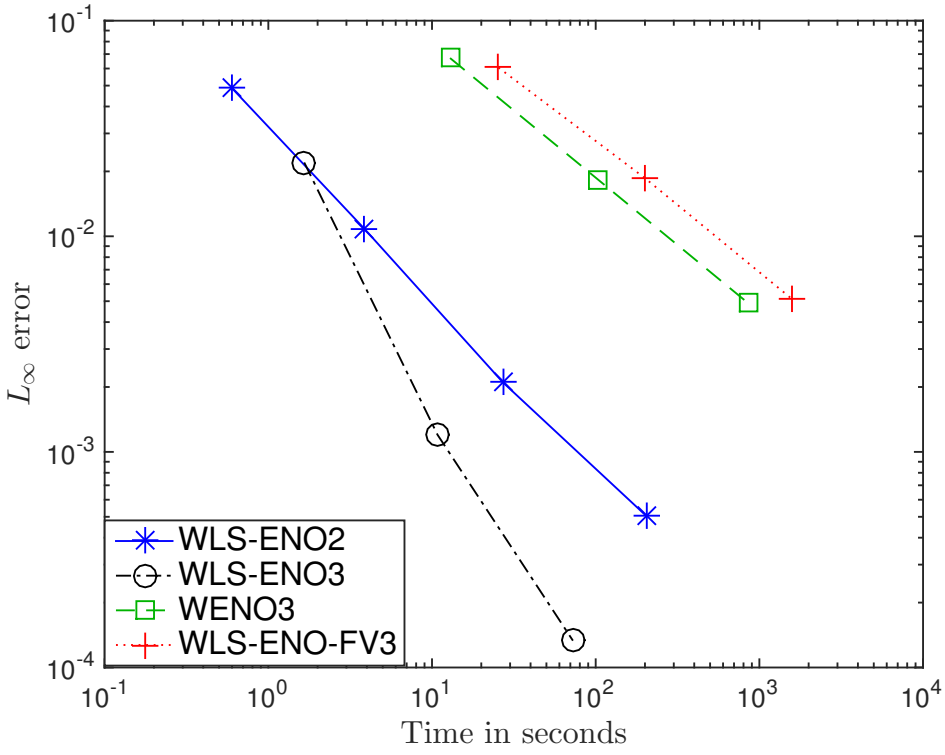



Figure 5.29: Timing comparison in 2D.

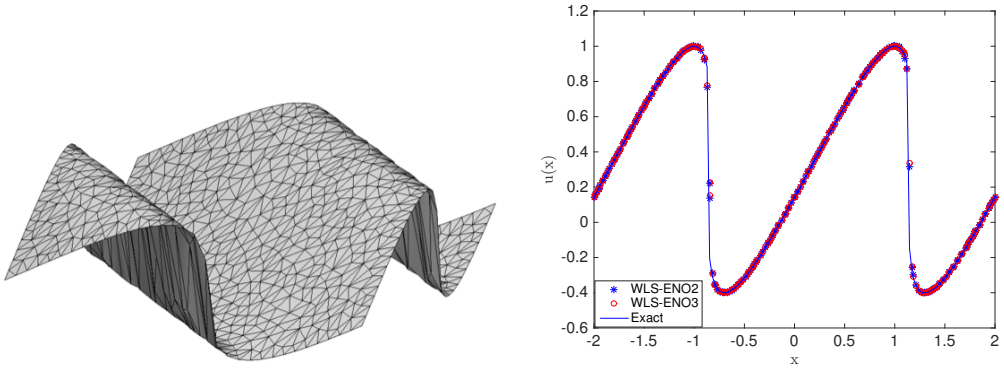



Figure 5.30: Exact solution (left) and numerical solution (right) with degree 2 and 3 WLS-ENO schemes for the 2-D Burgers' equation at $t = 0.5$.

**2-D Burgers' Equation**

For piecewise smooth solutions, we solve the 2-D Burgers' equation

$$u_t + \left(\frac{u^2}{2}\right)_x + \left(\frac{u^2}{2}\right)_y = 0 \tag{5.2.11}$$

over $[-2, 2]^2$, with periodic boundary conditions and the initial condition

$$u_0(x, y) = 0.3 + 0.7 \sin\left(\frac{\pi}{2}(x + y)\right). \tag{5.2.12}$$

Although the initial condition is smooth, discontinuities develop over time. Figure 5.30(left) shows the exact solution at $t = 0.5$, when the solution becomes discontinuous, and Figure 5.30(right) shows results of the 1-D cross-sections along $x = y$. We can see that the schemes approximate the exact solution and maintain non-oscillatory property very well.

**2-D Vortex Evolution Problem**

This is a test to demonstrate the accuracy of WLS-ENO schemes for Euler systems, which have the following form

$$\begin{pmatrix} \rho \\ \rho u \\ \rho v \\ E \end{pmatrix}_t + \begin{pmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ u\left(E+p\right) \end{pmatrix}_x + \begin{pmatrix} \rho v \\ \rho uv \\ \rho v^2 + p \\ v\left(E+p\right) \end{pmatrix}_y = \mathbf{0}, \tag{5.2.13}$$

where

$$E = \frac{p}{\gamma - 1} + \frac{1}{2}\rho\left(u^2 + v^2\right). \tag{5.2.14}$$

The test case involves the convection of an isentropic vortex in inviscid flow. We consider an idealized problem: The mean flow is $\rho_\infty = 1$, $p_\infty = 1$, $(u_\infty, v_\infty) = (1, 1)$. As an initial condition, an isentropic vortex with no perturbation in entropy ($\delta S = 0$) is added to the mean flow field. The perturbation values are given by

$$(\delta u, \delta v) = \frac{\beta}{2\pi} e^{\frac{1-r^2}{2}} \left(-\overline{y}, \overline{x}\right), \tag{5.2.15}$$

$$\delta T = -\frac{\left(\gamma - 1\right)\beta^2}{8\gamma\pi^2} e^{1-r^2} \tag{5.2.16}$$

where $(\overline{x}, \overline{y}) = (x - 5, y - 5)$, $r^2 = \overline{x}^2 + \overline{y}^2$, and the vortex strength $\beta = 5$. The computational domain is taken as $[0, 10] \times [0, 10]$, extended periodically in both directions. The exact solution of this problem is simply a vortex convecting with the speed $(1, 1)$ in the diagonal direction. In this example, we use degree 2 and 3 WLS-ENO schemes and compute the results up to $t = 1$. Figure 5.31 shows that

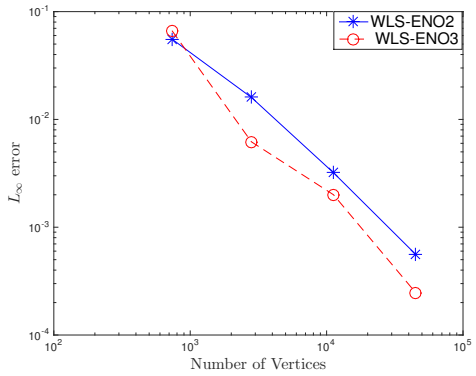



Figure 5.31: Density errors in 2-D Vortex Evolution problem by WLS-ENO schemes on triangular meshes.

the results achieved the expected convergence rate.

### 2-D Explosion Test Problem

In this test, we solve the Euler equation with the initial condition

$$(\rho, u, v, p)^T = \begin{cases} (1, 0, 0, 1)^T & \sqrt{x^2 + y^2} \leq 0.2 \\ (0.125, 0, 0, 0.1)^T & \sqrt{x^2 + y^2} > 0.2 \end{cases}. \tag{5.2.17}$$

We solve the problem on a domain of a unit disk centered at the origin, triangulated with meshes similar to that in Figure 5.27. We run the test up to $t = 0.1$ to ensure that the explosion wave do not reach the boundary.

Because of symmetry, this example is equivalent to the spherical one dimensional Euler equation with source terms [36],

$$\frac{\partial}{\partial t}\begin{pmatrix} \rho \\ \rho u \\ E \end{pmatrix} + \frac{\partial}{\partial r}\begin{pmatrix} \rho u \\ \rho u^2 + p \\ u\left(E + p\right) \end{pmatrix} = -\frac{d-1}{r}\begin{pmatrix} \rho u \\ \rho u^2 \\ u\left(E + p\right) \end{pmatrix}, \tag{5.2.18}$$

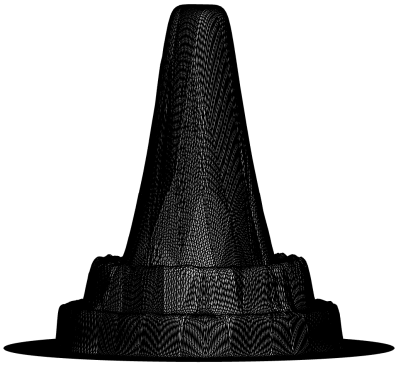


Figure 5.32: Numerical solution of 2-D explosion test by WLS-ENO scheme at $t = 0.1$.

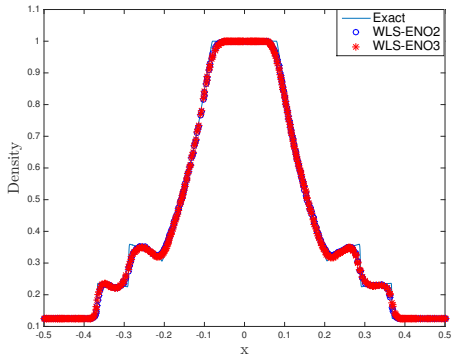



Figure 5.33: Numerical solution along $x = y$ by WLS-ENO schemes versus analytic solution at $t = 0.1$.

where $r$ is the radial coordinate. Thus, as a reference solution, we solve this 1-D problem on a very fine mesh composed of 4000 grid points and compare it with our numerical solution. Figures 5.32 and 5.33 show that the numerical solutions agree with the exact solutions very well.

# Chapter 6

# Conclusion and Further Work

In this paper, we introduced a new family of essentially non-oscillatory schemes, called *WLS-ENO*, in the context of both finite volume and finite difference methods for solving hyperbolic conservation laws. These new schemes can solve both the conservation form and the differential form of conservation laws and can be applied to both structured and unstructured meshes, which enables a wider range of applications than WENO schemes. In contrast, WENO for finite difference schemes can only be applied to structured meshes. The basic framework of WLS-ENO schemes includes Taylor series expansions and a weighted least squares formulation. We showed that over structured meshes, WLS-ENO delivers similar and even better accuracy compared to WENO, while enabling a larger stability region. For unstructured meshes, we showed that WLS-ENO enables accurate and stable solutions. Its accuracy and stability are rooted in the facts that the convexity requirement is satisfied automatically in WLS-ENO, and the stencil can be adapted more easily to ensure the stability of the approximations. We presented detailed analysis of WLS-ENO in terms of accuracy in 1-D and 2-D, and its stability for hyperbolic conser-

vation laws in 1-D. We also assessed the WLS-ENO with a large collection of test problems in 1-D, 2-D, and 3-D, including wave equations, Burgers' equation, and the Euler equations with fairly complicated initial conditions. Our numerical results demonstrated that WLS-ENO is accurate and stable over unstructured meshes for very complex problems. We also did timing comparison with WENO to test the efficiency. The comparison demonstrated that the WLS-ENO schemes for generalized finite difference is as fast as WENO schemes in 1-D and even faster than WENO in 2-D.

As we see from above, WLS-ENO provides a more general tool for dealing with piecewise smooth functions over unstructured meshes for engineering applications involving complex geometries. For future works, we intend to improve the efficiency of WLS-ENO finite volume schemes by developing a hybrid method that utilizes the traditional WENO on structured meshes in the interior and utilizes WLS-ENO over unstructured meshes near complex boundaries. For WLS-ENO finite difference schemes, we plan to extend WLS-ENO schemes to 3D and solve more complicated problems like shallow water problems [43] and Hamilton Jacobi problems [2] to test the performance. Also, the schemes use local variables only, which makes it ideal for parallelization. We will explore it in the future.

# Bibliography


[1] R. Borges, B. Costa, and W. S. Don. An improved weighted essentially non-oscillatory scheme for hyperbolic conservation laws. *J. Comput. Phys.*, 227:3191–3211, 2008.

[2] R. F. Carlini, Elisabetta and G. Russo. A weighted essentially nonoscillatory, large time-step scheme for Hamilton–Jacobi equations. *SIAM J. Sci. Comput.*, 27.3:1071–1091, 2005.

[3] M. Castro, B. Costa, and W. S. Don. High order weighted essentially non-oscillatory WENO-Z schemes for hyperbolic conservation laws. *J. Comput. Phys.*, 230:1766–1792, 2011.

[4] M. R. Charest, T. R. Canfield, N. R. Morgan, J. Waltz, and J. G. Wohlbier. A high-order vertex-based central ENO finite-volume scheme for three-dimensional compressible flows. *Comput. Fluids*, 114:172–192, 2015.

[5] M. R. Charest, C. P. Groth, and P. Q. Gauthier. High-order CENO finite-volume scheme for low-speed viscous flows on three-dimensional unstructured mesh. *Convergence*, 59:60, 2012.

[6] T. S. Cheng and K. S. Lee. Numerical simulations of underexpanded supersonic jet and free shear layer using WENO schemes. *Int. J. heat and fluid flow*, 26.5:755–770, 2005.

[7] A. J. Christlieb, Y. Liu, Q. Tang, and Z. Xu. High order parametrized maximum-principle-preserving and positivity-preserving WENO schemes on unstructured meshes. *J. Comput. Phys.*, 281:334–351, 2015.

[8] B. Cockburn and C.-W. Shu. The Runge-Kutta discontinuous Galerkin method for conservation laws V: multidimensional systems. *J. Comput. Phys.*, 141.2:199–224, 1998.

[9] R. Conley, T. J. Delaney, and X. Jiao. Overcoming element quality dependence of finite elements with adaptive extended stencil FEM (AES-FEM). *Int. J. Numer. Meth. Engrg.*, 2016. To appear. Preprint available at http://arxiv.org/abs/1508.01435.

[10] M. Dumbser, O. Zanotti, A. Hidalgo, and D. S. Balsara. ADER-WENO finite volume schemes with space–time adaptive mesh refinement. *J. Comput. Phys.*, 248:257–286, 2013.

[11] V. Dyedov, N. Ray, D. Einstein, X. Jiao, and T. J. Tautges. AHF: Array-based half-facet data structure for mixed-dimensional and non-manifold meshes. In J. Sarrate and M. Staten, editors, *Proceedings of the 22nd International Meshing Roundtable*, pages 445–464. Springer International Publishing, 2014.

[12] S. Gottlieb and C.-W. Shu. Total variation diminishing Runge-Kutta schemes. *Math. Comput.*, 67(221):73–85, 1998.

[13] F. Grasso and S. Pirozzoli. Shock-wave vortex interactions: Shock and vortex deformations, and sound production. *Theoret. Comput. Fluid Dynamics*, 13:421–456, 2000.

[14] A. K. Henrick, T. D. Aslam, and J. M. Powers. Mapped weighted essentially non-oscillatory schemes: Achieving optimal order near critical points. *J. Comput. Phys.*, 207:542–567, 2005.

[15] C. Hu and C.-W. Shu. Weighted essentially non-oscillatory schemes on triangular meshes. *J. Comput. Phys.*, 150:97–127, 1999.

[16] G. Jiang and C.-W. Shu. Efficient implementation of weighted ENO schemes. *J. Comput. Phys.*, 126:202–228, 1996.

[17] G.-S. Jiang and D. Peng. Weighted ENO schemes for Hamilton-Jacobi equations. *SIAM J. Sci. Comput.*, 21(6):2126–2143, 2000.

[18] X. Jiao and D. Wang. Reconstructing High-Order Surfaces for Meshing. In S. Shontz, editor, *Proceedings of the 19th International Meshing Roundtable*, pages 143–160. Springer Berlin Heidelberg, 2010.

[19] X. Jiao and H. Zha. Consistent computation of first- and second-order differential quantities for surface meshes. In *Proceedings of the 2008 ACM symposium on Solid and physical modeling*, pages 159–170. ACM, 2008.

[20] M. Lahooti and A. Pishevar. A new fourth order central WENO method for 3D hyperbolic conservation laws. *Appl. Math. Comput.*, 218(20):10258–10270, 2012.

[21] R. J. LeVeque. Nonlinear conservation laws and finite volume methods. In O. Steiner and A. Gautschy, editors, *Computational Methods for Astrophysical Fluid Flow*, volume 27 of *Saas-Fee Advanced Courses*, pages 1–159. Springer, 1998.

[22] D. Levy, G. Puppo, and G. Russo. A fourth-order central WENO scheme for multidimensional hyperbolic systems of conservation laws. *SIAM J. Sci. Comput.*, 24(2):480–506, 2002.

[23] R. Liska and B. Wendroff. Two dimensional shallow water equations by composite schemes. *Int. J. Numer. Meth. Fluids*, 30:461–479, 1999.

[24] H. Liu and X. Jiao. WLS-ENO: Weighted-least-squares based essentially non-oscillatory schemes for finite volume methods on unstructured meshes. *J. Comput. Phys.*, 314:749–773, 2016.

[25] M. V. Liu, Yen and Z. J. Wang. Spectral difference method for unstructured grids I: basic formulation. *J. Comput. Phys.*, 216.2:780–801, 2006.

[26] Y. Liu and Y.-T. Zhang. A robust reconstruction for unstructured WENO schemes. *J. Sci. Comput.*, 54(2,3):603–621, 2013.

[27] J. S. Park, S.-H. Yoon, and C. Kim. Multi-dimensional limiting process for hyperbolic conservation laws on unstructured grids. *J. Comput. Phys.*, 229:788–812, 2010.

[28] K. Sebastian and C.-W. Shu. Multidomain WENO finite difference method with interpolation at subdomain interfaces. *J. Sci. Comput.*, 19.1-3:405–438, 2003.

[29] Y. Shen and G. Zha. Improvement of the WENO scheme smoothness estimator. *Int. J. Numer. Meth. Fluid*, 64(6):653–675, 2010.

[30] Y. Shen and G. Zha. Improvement of the WENO scheme smoothness estimator. *Int. J. Numer. Meth. Fluid*, 64(6):653–675, 2010.

[31] J. Shi, C. Hu, and C.-W. Shu. A technique of treating negative weights in WENO schemes. *J. Comput. Phys.*, 175:108–127, 2002.

[32] C.-W. Shu. Essentially non-oscillatory and weighted essentially non-oscillatory schemes for hyperbolic conservation laws. In E. A. Quarteroni, editor, *Advanced Numerical Approximation of Nonlinear Hyperbolic Equations*, volume 1697 of *Lecture Notes in Mathematics*, pages 325–432. Springer, 1998.

[33] C.-W. Shu. Essentially non-oscillatory and weighted essentially non-oscillatory schemes for hyperbolic conservation laws. In A. Quarteroni, editor, *Advanced Numerical Approximation of Nonlinear Hyperbolic Equations*, volume 1697 of *Lecture Notes in Mathematics*, pages 325–432. Springer, 1998.

[34] C.-W. Shu. High order weighted essentially nonoscillatory schemes for convection dominated problems. *SIAM Rev.*, 51(1):82–126, 2009.

[35] V. Titarev and E. Toro. Finite-volume WENO schemes for three-dimensional conservation laws. *J. Comput. Phys.*, 201(1):238–260, 2004.

[36] E. F. Toro. *Riemann Solvers and Numerical Methods for Fluid Dynamics: A Practical Introduction*. Springer, 2009.

[37] P. Tsoutsanis, V. A. Titarev, and D. Drikakis. WENO schemes on arbitrary mixed-element unstructured meshes in three space dimensions. *J. Comput. Phys.*, 230(4):1585–1601, 2011.

[38] Y. L. G. M. Wang, Z. J. and A. Jameson. Spectral difference method for unstructured grids II: extension to the euler equations. *J. Sci. Comput.*, 32.1:45–71, 2007.

[39] Z. J. Wang and Y. Liu. Spectral (finite) volume method for conservation laws on unstructured grids: II. Extension to two-dimensional scalar equation. *J. Comput. Phys.*, 179.2:665–697, 2002.

[40] Z. J. Wang and Y. Liu. Spectral (finite) volume method for conservation laws on unstructured grids: III: One dimensional systems and partition optimization. *J. Comput. Phys.*, 20.1:137–157, 2004.

[41] W. Wolf and J. Azevedo. High-order ENO and WENO schemes for unstructured grids. *Int. J. Numer. Meth. Fluid*, 55(10):917–943, 2007.

[42] P. Woodward and P. Colella. Numerical simulation of two-dimensional fluid flows with strong shocks. *J. Comput. Phys.*, 54:115, 1984.

[43] Y. Xing and C.-W. Shu. High order finite difference weno schemes with the exact conservation property for the shallow water equations. *J. Comput. Phys.*, 208.1:206–227, 2005.

[44] J.-Y. Yang, S.-C. Yang, Y.-N. Chen, and C.-A. Hsu. Implicit weighted ENO schemes for the three-dimensional incompressible Navier–Stokes equations. *J. Comput. Phys.*, 146(1):464–487, 1998.

[45] S. Zhang and C.-W. Shu. A new smoothness indicator for the WENO schemes and its effect on the convergence to steady state solutions. *J. Comput. Phys.*, 31(1-2):273–305, 2007.

[46] S. Zhang and C.-W. Shu. A new smoothness indicator for the WENO schemes and its effect on the convergence to steady state solutions. *SIAM J. Sci. Comput.*, 31(1-2):273–305, 2007.

[47] Y.-T. Zhang and C.-W. Shu. Third order WENO scheme on three dimensional tetrahedral meshes. *Commun. Comput. Phys.*, 5(2-4):836–848, 2009.

[48] J. Zhu and J. Qiu. Finite volume Hermite WENO schemes for solving the Hamilton-Jacobi equations II: Unstructured meshes. *Comput. Math. Appl.*, 68.10:1137–1150, 2014.