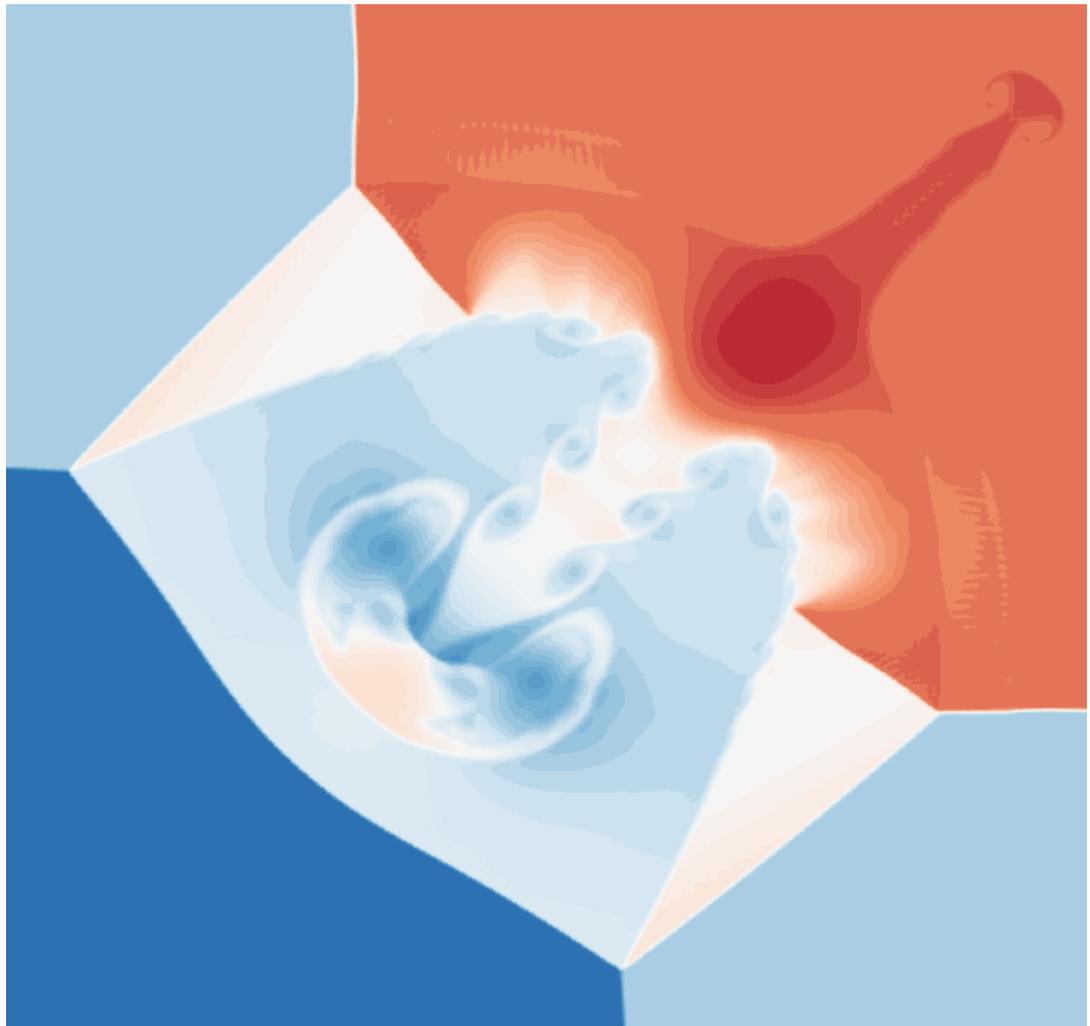


High-Order Numerical Schemes for Compressible Flows

V. Satheesh Kumar Nair

Delft University of Technology



High-Order Numerical schemes for Compressible Flows

By

Vishnu Satheesh Kumar Nair

in partial fulfilment of the requirements for the degree of

Master of Science
in Aerospace Engineering

at the Delft University of Technology,
to be defended publicly on Friday July 29, 2016 at 9:15 AM.

Supervisor:	Dr. R.P. Dwight,	TU Delft
Thesis committee:	Dr. ir. M.I. Gerritsma,	TU Delft
	Dr.-ing. R. Schmehl,	TU Delft

This thesis is confidential and cannot be made public until August 1, 2016.

An electronic version of this thesis is available at <http://repository.tudelft.nl/>.



Preface

The last ten months have been the hardest I have ever worked in my life, and I hope this report reflects the effort I have put in. However this would not have been possible without the support of a few very important people.

I couldn't have asked for a better thesis supervisor than Richard. I will always be thankful for all the help you have given me, starting from the very first meeting in which you explained the very basics of CFD and numerical methods, to asking me to be your TA (which I really enjoyed), all the recommendations, and to just being a really cool supervisor. You have always given me the feeling that we were working together rather than you being a supervisor in a literal sense. I remember you mentioning that you saw me as a canary in a coal mine as far as the solver was concerned, and there would be more thesis opportunities for future students if I was successful. Well I guess the canary is still alive and kicking (I hope).

This thesis and diploma is dedicated to my parents, my brother and my really awesome sister-in-law without whose support I wouldn't even be in Delft. Thank you for always believing in me and for never saying no. A special mention to my girlfriend Reshma for standing by my side through everything during the last thirteen years. I can't wait to return to India and get married to you.

Thank you Maria for keeping me sane through the last six months. Meeting you has been the best thing to happen to me in Delft. Spending time with you and watching you scream at innocent ducklings in the city centre was a welcome distraction from the thesis!

A big thank you to my friend Minaxi. I have known you since we were kids and you know how much you mean to me. Thank you for being the best human being on the planet.

I end with a big shoutout to my team-mates at Ariston 80 Football Club. I will always cherish the memories of playing in the league during the last two years with you guys.

Vishnu Nair
Delft, July 2016

Abstract

High-order numerical methods for Computational Fluid Dynamics have undergone significant fundamental developments over the last two decades owing to combined efforts from the applied mathematics and engineering communities. Even though low-order numerical methods are still the standard in industry, the increased requirements of engineering applications have led to significant scientific interest in developing efficient and robust numerical methods. Applications that would benefit from high-order numerical methods include Direct Numerical Simulations (DNS), Large Eddy simulations (LES), Computational Aero-Acoustics (CAA) and vortex dominated flows.

The objective of this thesis is to successfully implement and validate a fifth order traditional WENO scheme in a finite volume framework, for a solver currently being developed in the Aerodynamics group of TU Delft. A detailed literature study of classical numerical schemes has been performed along with a study of the traditional WENO schemes. The quality of results using the fifth order scheme is studied for a variety of test cases to study the shock capturing ability of the scheme.

Implementing the finite volume WENO schemes includes the calculation of numerical flux at cell faces using Gaussian quadrature formulas. The effect of varying the number of Gaussian quadrature points while calculating the numerical flux is investigated. Also, the effect of the approximate Riemann solvers on the quality of results is studied by implementing four different Riemann solvers and studying the results for different test cases using these Riemann solvers.

The test cases are governed by the inviscid Euler equations and deals with flow in the compressible regime. They involve shocks, other discontinuities and often also complicated structures in the smooth part of the solution which tests the design of the schemes to be non-oscillatory at the discontinuities and still gives a high order of accuracy in the smooth parts of the flow. Convergence tests of the error for test cases using the linear advection equation is used to study the order of accuracy of the scheme using different number of Gaussian quadrature points. The tests clearly show that the order of accuracy remains the same irrespective of the number of quadrature points used. This result is important as it allows simulation run with just one quadrature point which is less expensive, and saves memory. This result is highly relevant while running test cases for LES where very fine grids have to be used.

WENO schemes have been considered to be too dissipative for LES in their traditional form. This is indeed true as seen by Kelvin-Helmholtz type small scale vortices (which are characteristic of high Reynolds number flows), even in the test cases using the inviscid Euler equations, due to the inherent dissipation in the schemes. However, this could be seen as motivation for using the WENO schemes for Implicit LES where no explicit sub-grid scale models are used to represent the unresolved scales. The different Riemann solvers exhibit different levels of dissipation and recommendations are made for the choice of Riemann solvers according to the application.

Contents

Abstract	6
1. Introduction	12
1.1 Background	12
1.2 Thesis goal and Research question	14
1.3 Thesis outline	15
2. Classical numerical schemes	16
2.1 Introduction	16
2.2 Theory of Conservation laws.....	17
2.3 Selection of discretization method	18
2.4 Godunov type schemes.....	20
2.4.1 Upwind schemes.....	22
2.4.2 Optimized WENO schemes.....	26
3. Numerical scheme implementation	29
3.1. Finite Volume Framework in three dimensions	29
3.2 WENO reconstruction	32
3.2.1 General reconstruction from cell averages in 1d	32
3.2.2 WENO approximation in 1d.....	33
3.2.3 Fifth order WENO reconstruction:.....	36
3.2.4 Reconstruction on quadrature points	39
3.2.5 Reconstruction for system of equations	40
3.3 Approximate Riemann solvers	41
3.3.1 Harten-Lax-Van Leer schemes	42
3.3.2 HLLC schemes	43
3.3.3 Roe solver (with entropy fix)	45

3.3.4 Lax-Friedrichs flux:.....	46
3.4 Time discretization.....	47
3.5 Boundary conditions.....	49
3.5.1 Periodic Boundary conditions.....	49
3.5.2 Fixed Boundary condition (Dirichlet).....	49
3.5.3 Solid Boundaries.....	50
3.5.4 Far-Field Boundaries.....	50
3.5.5 Supersonic Inflow/Outflow Boundary Condition:.....	51
4. Numerical study of WENO scheme.....	52
4.1 Linear Advection.....	52
4.2 The Doswell Frontogenesis problem.....	55
4.3 Sod's shock tube problem.....	61
4.4 Spherical Explosion Problem.....	67
4.5. Two dimensional Riemann problems:.....	70
4.5.1 Configuration 3.....	72
4.5.2 Configuration 17.....	75
4.6. Wind tunnel with Forward Facing Step:.....	78
5. Conclusion and Recommendations.....	82
5.1 Conclusions.....	82
5.2 Recommendations and Future work:.....	84
Bibliography.....	85
Appendix.....	88
A. Basic Numerical Methods for the Euler Equations.....	88
A.1 Flux Approach.....	88
A.1.1 Lax- Friedrichs Method.....	88
A.1.2 Lax Wendroff Methods:.....	89

A.2.1 Wave approach: Flux Vector Splitting	90
A.2.1.1 Van-Leers Flux split first order upwind method	90
A.2.2 Wave approach: Reconstruction-evolution	92
A.2.2.1 Kurganov-Tadmor scheme with MUSCL reconstruction	92

1

Introduction

1.1 Background

Research in Computational Fluid Dynamics is almost always devoted to flows in which turbulence plays a major role. The physical nature of turbulence, aptly termed ‘deterministic chaos’, is still not fully understood but research has reached a level where the simulation and modeling of turbulent flows can be achieved with satisfying degrees of accuracy. Turbulence is always three dimensional and unsteady and is characterized by a wide range of scales in space and time. For computation of turbulent flows, the accuracy of the results depend on the range of scales which are resolved. In Direct Numerical Simulations (DNS), all scales are resolved by solving the Navier-Stokes equations on a very fine mesh to adequately capture and resolve all scales down to the Kolmogorov scale in a given flow. This results in highly accurate results but comes at the cost of very high computational expense. Hence DNS of compressible turbulent flows are limited to low Reynolds number flows and is often used to give a database for comparing results. At the other end of the spectrum lies the RANS approach where the instantaneous fluid flow is not simulated; instead, the time averaged or ensemble averaged Navier Stokes equations are solved, which introduces a new apparent stress called Reynolds stress. A turbulence model is then introduced to provide closure to the turbulence problem that arises due to the additional unknowns generated by the Reynolds stress term. RANS approach is computationally much cheaper, as a result of which it is preferred in industry and has become the backbone in CFD engineering applications over the last few decades.

An alternate approach introduced by Smagorinsky in 1963 is the Large Eddy Simulation (LES). In this approach only the large scale motions or the large eddies are resolved and the effect of the smaller scales (sub-grid scales) are modeled. Hence in terms of computational cost, this approach falls between the DNS and RANS approach. The LES approach is significantly more accurate than the RANS approach since the large eddies contain the majority of the turbulent energy and are responsible for most of the momentum transfer and turbulent mixing. LES captures these large eddies in full detail while the effect of the large eddies are modeled in the RANS approach. Hence currently, LES can be considered to be the most computationally viable and numerically accurate tool for simulating complex turbulent flows.

Most numerical methods employed by the RANS approach and where earlier research was focused on, provide mostly first or second order accuracy in the discretization of the governing equations. However, the use of such low order numerical methods can result in the

discretization error of the numerical scheme masking the effect of the sub-grid scale model in LES. This necessitates the use of high order numerical methods for successful implementation of LES compared to the widespread current practice of implementing second order central finite difference numerical schemes. At this point, it is important to mention that in this work, a numerical scheme qualifies as a high-order scheme if it has an order of three or above.

It is important to realize that for most practical engineering applications, for example, smooth piecewise linear problems, or with isolated discontinuities, first and second order schemes are good choices especially when taking into account a combination of the factors of time, speed and importantly the ease of coding and implementation. However for more complicated applications, with both discontinuities and complex flow structures, such as compressible turbulence, high-order schemes are highly recommended. (Cockburn, Johnson, Shu, & Tadmor, 1997). A second order finite volume solver is the current industrial standard, and it is safe to say that for a wide range of applications, such solvers are sufficient and provides satisfactory performance. Hence the introduction of solvers based on high-order schemes is challenging, especially when the application for which the solver is being used is well served by lower order solvers. This is one of the primary reasons why presently the application of such schemes are restricted to an academic setting, and mostly for research applications. In addition to LES problems, there are other research fields for which high order schemes are not just preferable, but necessary. These include Computational Aero-Acoustics (CAA) problems which require schemes with excellent dissipation and dispersion properties since the CAA waves must be tracked over longer distances; vortex dominated flows which must be convected over a longer period of time and for which it makes more economic sense to use high order schemes. This is because high-order schemes resolve smaller scales to a much higher accuracy on a given grid compared to a lower order scheme. Another property of higher order schemes which makes them very attractive is the inherently lesser dissipation which leads to lesser interference with the development of the turbulent energy cascade. Hence while considering the same grid, a high-order scheme can achieve a higher order of accuracy. Also, the amount of computational effort to achieve a smaller error tolerance is much lesser compared to a lower order scheme. (Lopez-Morales, et al., 2014)

LES as a turbulence modelling approach is rapidly gaining approval from the computational fluid dynamics group. This approach has still not reached a stage where researchers inexperienced in LES can operate an LES solver similar to a commercial RANS solver. Most LES techniques are problem specific and this requires a good in-depth knowledge towards how the approach and how the numerical scheme employed in LES works. Several universities and research institutes have developed LES codes and solvers which implement a wide variety of high order numerical methods: HiFiLES from the Department of Aeronautics at Stanford University, PyFR from the Department of Aeronautics at Imperial College London, SPARK-LES from CIRA Italy, are a few of the solvers. The Aerodynamics group at TU Delft is developing an LES code "*Vayu*" for the simulation of turbulent compressible flows. *Vayu* is a multi-block Python-C library on a finite-volume/difference framework.

The most important step in the development of the solver for the simulation of compressible turbulent flows is the identification and implementation of a high order, high fidelity numerical scheme with robust shock capturing ability. This thesis deals with such high order

numerical schemes and its implementation on the TU Delft solver. In particular, focus is on the shock-capturing ability of high order schemes. Even though numerical simulation of compressible flows has a long history, no optimal computational scheme has been unanimously identified in the field of computational gas dynamics. An ideal numerical scheme for the simulation of such flows would be free of numerical dissipation in the smooth part of the flow, and must capture shock waves without any significant Gibbs oscillations. Shock capturing relies on the addition of explicit artificial dissipation as a substitute for physical viscosity to obtain non-oscillatory simulations (Pirozzoli, 2011).

1.2 Thesis goal and Research question

The main goal of the thesis is the implementation of a high-order numerical scheme in the TU Delft solver *Vayu* which is capable of simulating a wide range of compressible flows. One of the major criterion on which the performance of the scheme is measured, is in the simulation of turbulent compressible flows. An in-depth survey of existing numerical schemes and evaluation based on their suitability for the simulation of compressible turbulence, has identified the Weighted Essentially Non-oscillatory (WENO) schemes as the best candidate. The WENO schemes of Jiang and Shu (1996) are a class of high-order schemes which gives high spatial accuracy over the smooth parts of the flow problem and also provide robust shock capturing ability in the presence of shocks. Over the last decade, several variants of the WENO schemes have been developed with varying degrees of success, to improve and adapt the traditional scheme to various applications. Given the complexity in coding a fifth order WENO scheme, validation of the schemes with various test cases is highly critical to ensure correct implementation. This led to the main research question,

“What is the quality of results with respect to shock capturing, using the traditional WENO schemes for various test cases using the TU Delft solver Vayu?”

The ideal method to validate the scheme and ensure the correct implementation on the solver is to study the results for different simple and challenging test cases, and this has motivated the framing of this research question. The literature survey also identifies different test cases which are best suited for validation. Two sub questions were identified after the literature study to help answer the main research question. They are,

“What is the effect of the approximate Riemann solver on the quality of results for the different test cases?”

“What is the effect of using a one-point and two-points Gaussian quadrature formula for calculating the numerical fluxes at each cell interface in the finite volume scheme employed in the solver?”

The final goal of this thesis can hence be summarized very briefly as:

“Successful implementation of a traditional fifth-order shock capturing Weighted Essentially Non-Oscillatory scheme on the TU Delft solver Vayu, and successful validation for test cases with various degrees of difficulty”

Certain milestones are defined on the path to reaching this final goal. They are:

1. Implementation of the finite volume framework in the solver.
2. Implementation of the one dimensional finite-volume WENO reconstruction scheme in the framework, for the linear advection equation and validation using test cases.
3. Implementation of the WENO reconstruction for the Euler equations and validation using test cases in one, two and three dimensions.

Even though the selection of the numerical schemes was based on its ability to simulate turbulent compressible flows, the first step towards using the solver for such flows begins with the implementation and validation of the scheme for inviscid Euler equations. Test cases dealing with compressible flows and governed by the Euler equations will be considered. The WENO schemes are not commonly used for the discretization of viscous terms and hence the decision was taken to study the performance of the scheme for the inviscid Euler equations. However since the ultimate purpose of the solver is for LES of practical flows in the turbulent compressible regime, the implementation and development will always be done keeping this as the future goal.

1.3 Thesis outline

The layout of the thesis is described in this section. Chapter 2 explains some theoretical concepts which are central to this work. Also, a detailed literature study and evaluation of classical numerical schemes is presented. Finite volume Godunov type schemes are explained in detail since most advanced high order numerical schemes have their foundations in such schemes. A brief theoretical study of different variants of the WENO schemes is presented. Chapter 3 explains the traditional WENO schemes on a finite volume framework and the corresponding mathematical formulations involved. The steps involved in a fifth order WENO reconstruction in three dimensions on a finite volume framework, using a multi-point Gaussian quadrature formula is explained in detail. The stencils involved in the scheme, the different coefficients and important equations are illustrated in this chapter. A brief explanation of the different Riemann solvers used in the solver follows. The fourth order Runge-Kutta scheme which is used to forward the solution in time is explained along with a simple result for validating the time discretization. The different kinds of boundary conditions used in the different test cases are also mentioned briefly. Chapter 4 deals with the results obtained by using the fifth order WENO scheme on a variety of test cases governed by the advection equation and the Euler equations. Various contour plots and order of accuracy convergence test results are illustrated. The report ends with Chapter 5 which provides a conclusion of the results and the work in general, and further recommendations based on the conclusions. Very briefly, future avenues which can be explored and ways to extend the solver is presented in this chapter.

2

Classical numerical schemes

2.1 Introduction

A simple start towards explaining the backdrop of the thesis would be with the three steps in the computational modelling of any physical process:

1. Problem definition: Defining the problem under study in terms of measurable quantities such that a well-posed problem is obtained, ie, a problem with a unique solution for a given set of conditions.
2. Mathematical model: The problem has to be represented by a mathematical model described by partial differential equations that serve as the governing equations; for example the Navier Stokes equation for simulation of fluid motion, equations of elasticity for structural mechanics etc.
3. Computer simulation: Solution of the mathematical model using a particular numerical scheme under the given initial and boundary conditions.

Stringent requirements are posed on the mathematical model, and the numerical scheme when solutions containing discontinuities such as shock waves are computed. The formulation of the governing equations can be either in the differential or the integral form. Also a choice has to be made from the different types of variables that are available, for example, the conserved variables, primitive variables, characteristic variables etc. One obvious choice for the variables is the conserved variables. Schemes based on non-conserved variables fail at shock waves since they give the wrong jump conditions, which leads to wrong shock strength, speed and ultimately wrong position (Toro, 1999).

The classical result established by Lax and Wendroff (Lax & Wendroff, 1960) has very big implications on how the numerical schemes are designed because of the benefits of using discrete conservation form. The theorem states,

“If the solution of a conservative numerical scheme for a hyperbolic system of conservation laws converges, it will always converge towards a weak solution of the conservation law.”

For the simulation of compressible flows, we require shock capturing numerical schemes. A scheme is said to *capture* shocks when the shocks or discontinuities appear as regions of large gradients in the solution without having to give them any special treatment. According to the Lax-Wendroff theorem, if we use conservative schemes, if convergence occurs, it is

guaranteed that it will be to a weak solution. And the weak solution satisfy the jump conditions and hence give the correct shock speed.

The terms *conservative numerical schemes* and *conservation law* appear in the theorem and the concept is central to this work. Hence it is important to highlight the principle behind the conservation laws and this is done in the next section.

2.2 Theory of Conservation laws

The governing equations of continuum mechanics are commonly referred to as conservation laws. They represent the conservation of mass, momentum and energy. The main principle behind a conservation law is (Lax P. D., 1972),

The rate of change of a conserved flow quantity within a given volume is due to the combined effect of an internal source and the net flow of the quantity, or the flux of the quantity across the boundary surface

ie, the sum of the rate of change of a conserved variable $\mathbf{U}(x, t)$ within a volume \mathbf{V} and the flux \mathbf{f} of \mathbf{U} through the boundary \mathbf{A} is equal to the rate of production $\mathbf{S}(U, x, t)$.

Hence the integral form of the conservation law can be written as,

$$\frac{d}{dt} \int_{\mathbf{V}} U(x, t) dV = - \oint_{\mathbf{A}} \mathbf{f}(U) \cdot \mathbf{n} dA + \int_{\mathbf{V}} S(U, x, t) dV = 0 \quad (1)$$

Where \mathbf{n} is the outward unit normal vector at the surface \mathbf{A} .

Assuming a fixed control volume \mathbf{V} , the Gauss divergence theorem $\oint_{\mathbf{S}} \nabla \cdot \mathbf{f}(U) dV = \oint_{\mathbf{S}} \mathbf{f} \cdot \mathbf{n} dA$ can be applied,

$$\int_{\mathbf{V}} \left(\frac{\partial U}{\partial t} + \nabla \cdot \mathbf{f}(U) - S \right) dV = 0$$

Dividing by the volume and shrinking \mathbf{V} to a point where all the partial derivatives of \mathbf{U} and \mathbf{f} are continuous, strong or differential conservation law can be obtained,

$$\frac{\partial U}{\partial t} + \nabla \cdot \mathbf{f}(U) - S = 0 \quad (2)$$

The conservation law in a discrete form ensures that the amount of conserved quantity leaving one cell exactly enters the next cell. Hence all the flux on the interior of the domain telescopically cancels leaving only the flux across the boundaries, and hence provides a direct approximation to the weak form of the governing PDE. In this work, we shall assume that the walls of the computational domain are impermeable and there is no creation or destruction of the conserved quantity inside the domain. Hence the effect of the source term shall be neglected.

2.3 Selection of discretization method

The three classical approaches to solving the partial differential equations in the mathematical model are finite difference methods, finite volume methods and the finite element methods. In finite volume methods and finite element methods, the integral form of the conservative equations is used whereas finite difference methods use the strong form of the governing equations. The finite volume discretization method was selected for this work.

Finite volume schemes have proved to be successful in approximating solutions for a wide range of hyperbolic conservation systems. They are extensively used in fluid mechanics, meteorology, electromagnetics, semi-conductor device simulation, models of biological processes and many other engineering areas governed by conservative systems that can be written in integral control volume form (Barth & Ohlberger, 2004). There are certain advantages to the finite volume methods which justified their selection as the discretization method for the solver:

- Numerically robust: by satisfying discrete maximum (minimum) principles.
- Resulting schemes possess intrinsic local conservation properties: Finite Volume methods are based on cell averaged values as a result of which the discretization form is a set of finite volumes, and hence a conservative discretization is automatically obtained.
- High applicability on very general unstructured meshes. Hence better suited to handle complex geometries as they do not rely on any special mesh structure.
- Regardless of the size of the mesh, finite volume methods ensures the conservation of mass, momentum and energy, ie, satisfies the property of discrete conservation. It is beneficial to preserve the conservation form in the discretization for nonlinear problems in which the solution may develop shock waves. Assuring this will ensure that the discrete solution satisfies the correct shock jump conditions (according to the Lax-Wendroff theorem), provided that it converges as the mesh spacing is reduced to zero. Hence the finite volume framework enhances both the stability and accuracy of the numerical scheme.
- The use of integral formulations of the conservation laws provides a more natural treatment of Neumann boundary conditions as well as that of discontinuous source terms due to their reduced requirements on the regularity or smoothness of the solution.

Before we go into detail about the mathematical formulation of different classical numerical schemes, a few notions which play an important role and an understanding of which is necessary to grasp the theory of non-linear conservation laws are explained in the next subsection.

2.3.1 Monotonicity Preservation:

Given a certain initial condition $U(x,0)$ which is monotone increasing or decreasing, the solution of a scalar conservation law on an infinite spatial domain can be considered to be monotonicity preserving if the solution $U(x,t)$ is monotone increasing or decreasing respectively for all time.

The importance of the concept of monotonicity preservation for a numerical scheme is that, for monotone initial conditions, such schemes do not allow spurious oscillations and hence can be seen as an indication of non-linear stability of the scheme.

Incidentally, the concept of monotonicity preservation is applicable also in linear schemes. In fact, Godunov formulated his famous order-barrier theorem which links the concept of monotonicity preservation and linear schemes. The theorem states,

“A linear monotonicity-preserving scheme is first order accurate at best”

First order accurate numerical schemes are generally considered inefficient and inaccurate for most applications because of the need for a very fine mesh for acceptable results. Second-order schemes are usually the minimum requirement. Hence the Godunov order barrier theorem can be used to justify inherently non-linear methods even when the governing equations are linear (Laney, 1998).

Even though monotonicity preservation is an integral part of non-linear stability theory, it has its own drawbacks, mainly the fact that it fails to acknowledge non-monotone solutions. Hence we look beyond the concept of monotonicity preservation to the Total Variation Diminishing condition.

2.3.2 Total Variation Diminishing (TVD):

The concept of Total Variation Diminishing was first introduced by Harten in 1983. Originally introduced as Total Variation Non Increasing (TVNI) by Harten, it has since been mostly referred to as TVD in literature. Total Variation is a fundamental concept in analysis defined as,

$$TV(U(\cdot, t)) = \max_{\text{all possible } x_i} \sum_{i=-\infty}^{i=+\infty} |U(x_{j+1}, t) - U(x_j, t)|$$

Where the max notation indicates that the maximum is taken over the entire set of samples in the infinite domain. This definition ensures that all kind of functions are covered, however arbitrary. The meaning of total variation can be illustrated best by the following result from (Laney, 1998):

“The total variation of a function on an infinite domain is a sum of extrema, maxima counted positively and minima counted negatively. The two infinite boundaries are always extrema and they both count once; every other extrema counts twice.”

This result indirectly defines Total Variation as the measure of total amount of oscillation in a function. From the definition it is also clear that the total variation increases only if an existing maxima increases or a minima decreases. Some of the ways in which the TV can change are illustrated in the figures below

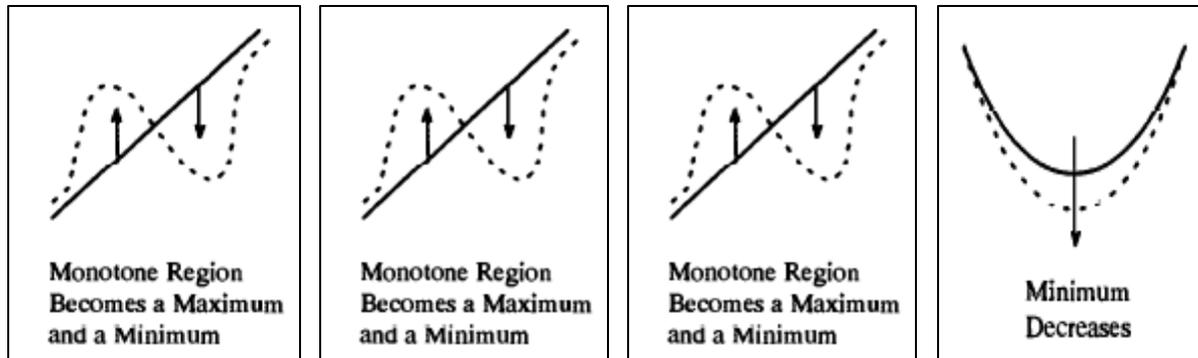


Figure 1: Change in Total Variation (Laney, 1998)

Exact solutions to a scalar conservation law have the property that, no new maxima or minima are created and existing maxima does not increase and minima do not decrease. Hence the solutions of scalar conservation laws can be considered to be total variation diminishing, ie,

$$TV(U^{n+1}) \leq TV(U^n), \quad \forall n > 0$$

Where

$$TV(U^n) = \sum_{i=-\infty}^{i=+\infty} |U_{i+1}^n - U_i^n|$$

since the equation for TV is applicable to discrete functions also.

The main advantage of imposing TVD condition for numerical schemes is that oscillations cannot grow indefinitely without violating the TVD condition since oscillations always add to the total variation. Assuming that the total variation of the initial condition is finite, the TVD condition prevents the unbounded growth of oscillations. Since TVD conditions also implies monotonicity preservation, it has become a very important concept in non-stability theory.

2.4 Godunov type schemes

The Godunov scheme introduced in 1959 (Godunov, 1959), had a profound impact on the field of Computational Fluid Dynamics. Many of the modern high-resolution schemes are based on Godunov type upwind differencing. Even though we have come a long way from the original first order Godunov scheme, it is necessary to explain the structure of Godunov type schemes as the understanding of the scheme is essential before moving on to high order schemes.

The Godunov schemes can be explained on the basis of a system of hyperbolic PDEs. The differential form of the hyperbolic system of conservation laws in one dimension is,

$$\frac{\partial U}{\partial t} + \frac{\partial f(U)}{\partial x} = 0 \quad (3)$$

The spatial domain can be discretized into cells or finite volumes,

$$I_j = \left[x_{j-\frac{1}{2}}, x_{j+\frac{1}{2}} \right], \text{ where } \Delta x = x_{j+\frac{1}{2}} - x_{j-\frac{1}{2}} \text{ is the grid spacing with } j = 1, \dots, N.$$

The sliding average of the weak solution $U(x, t)$ is introduced,

$$\bar{U}_j(\cdot, t) = \frac{1}{\Delta x} \int_{I_j} U(x, t) dx \quad (4)$$

Integrating equation 4 over each cell, the sliding averages give,

$$\frac{\partial}{\partial t} \bar{U}_i(\cdot, t) = -\frac{1}{\Delta x} [f(U(x_{j+1/2}, t)) - f(U(x_{j-1/2}, t))] \quad (5)$$

To forward the scheme in time, a small time step Δt is introduced, and integration is done over $t \leq \tau \leq t + \Delta t$, to obtain the discrete form,

$$\bar{U}_i(x, t + \Delta t) = \bar{U}_i(x, t) - \frac{1}{\Delta x} \left[\int_{\tau=t}^{t+\Delta t} f\left(U\left(x_{j+\frac{1}{2}}, \tau\right)\right) d\tau - \int_{\tau=t}^{t+\Delta t} f\left(U\left(x_{j-\frac{1}{2}}, \tau\right)\right) d\tau \right] \quad (6)$$

This equivalent form of the conservation law is the starting point of construction for Godunov-type schemes. It gives the relation between the sliding averages $\bar{U}(\cdot, t)$ and the underlying point values $U(\cdot, t)$.

An approximation solution denoted by $w(\cdot, t^n)$, at discrete time levels $t^n = n\Delta t$ is constructed. $w(x, t^n)$ is a piecewise polynomial of the form,

$$w(x, t^n) = \sum p_j \chi_j(x), \quad \chi_j(x) := 1_{I_j} \quad (7)$$

Where $p_j(x)$ are algebraic polynomials supported at the discrete cells, I_j .

An exact evolution of $w(\cdot, t^n)$ based on equation 6 gives,

$$\bar{w}(x, t^{n+1}) = \bar{w}(x, t^n) - \frac{1}{\Delta x} \left[\int_{t^n}^{t^{n+1}} f\left(w\left(x_{j+\frac{1}{2}}, \tau\right)\right) d\tau - \int_{t^n}^{t^{n+1}} f\left(w\left(x_{j-\frac{1}{2}}, \tau\right)\right) d\tau \right]$$

(8)

To construct Godunov type schemes we realize above equation at discrete grid points (Cockburn, Johnson, Shu, & Tadmor, 1997). According to the way of sampling equation 8, we can distinguish between the numerical schemes as upwind schemes and central schemes.

2.4.1 Upwind schemes

We begin by defining the cell-average as

$$\bar{w}_j^n = \frac{1}{\Delta x} \int_{I_j} w(x, t) dx$$

(9)

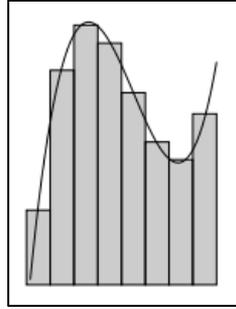


Figure 2 : Cell averaging of analytic data (Barth & Ohlberger, 2004)

Equation 8 is sampled at the mid cells $x = x_j$ and an evolution scheme is obtained for the cell averages,

$$\bar{w}_j^{n+1} = \bar{w}_j^n - \frac{1}{\Delta x} \left[\int_{\tau=t^n}^{t^{n+1}} f\left(w\left(x_{j+\frac{1}{2}}, \tau\right)\right) d\tau - \int_{\tau=t^n}^{t^{n+1}} f\left(w\left(x_{j-\frac{1}{2}}, \tau\right)\right) d\tau \right]$$

(10)

The next step is to express the point values $\{w(x_{j+1/2}, \tau)\}_j$, $t^n \leq \tau \leq t^{n+1}$ in terms of the known cell average $\{\bar{w}_j^n\}$. This is done by a step often referred to as reconstruction-evolution.

- Reconstruction: The point-wise values of $w(\cdot, \tau)$ at $\tau = t^n$ is recovered by a piecewise polynomial approximation,

$$w(x, t^n) = \sum_j p_j(x) \chi_j(x)$$

(11)

With the additional condition $\bar{p}_j(x_j) = \bar{w}_j^n$

- Evolution: $w(x_{j+1/2}, \tau \geq t^n)$ is determined as the solution of the generalized Riemann problem at each interface. For example, at the interface $x_{j+1/2}$,

$$\begin{aligned} \partial_\tau w_{j+1/2}(\xi, \tau) + \partial_\xi f\left(w_{j+1/2}(\xi, \tau)\right) &= 0, \quad t \geq t^n; \\ w_{j+1/2}(\xi, 0) &= \begin{cases} p_j(x) & \text{if } \xi < 0 \\ p_{j+1}(x) & \text{if } \xi > 0 \end{cases} \end{aligned}$$

(12)

A second important aspect of Godunov type schemes is the idea of replacing the true flux at the interfaces f by a *numerical flux function* $g(u_L, u_R): \mathbb{R} \times \mathbb{R} \mapsto \mathbb{R}$, which is a Lipschitz continuous function of the two interface states (u_L, u_R) . The two interface states constitute a Riemann problem, the solution of which consists of a family of left and right going non-linear waves as shown in figure 3. An exact or approximate Riemann solver distributes these non-linear waves between the two neighbouring cells I_j and I_{j+1} , and this distribution is responsible for the upwinding.

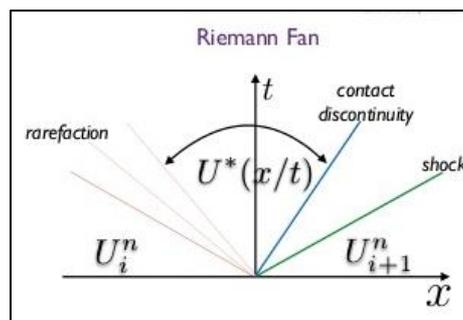


Figure 3 : Solution of Riemann Problem

In his original first-order scheme, Godunov proposed the evolution of the solution from one time state to another by assuming a piece-wise constant reconstruction of the solution over the spatial grid, ie, $w(x, t^n) = \sum \bar{w}_j^n \chi_j$. This results in a left and right solution at each interface and hence a jump, thus generating a sequence of Riemann problems as shown in figure 4. The flux at each interface is then obtained by solving the self-similar solution of the Riemann problem by using an exact Riemann solver (Cockburn, Johnson, Shu, & Tadmor, 1997).

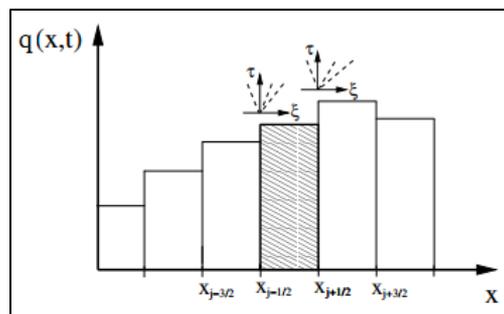


Figure 4: Godunov's interface Riemann problems from piecewise constant interface states (Barth & Ohlberger, 2004)

First order schemes are considered to be generally inaccurate for most practical calculations unless the mesh is made too fine which further drives up the computational cost. This prompted the research into high order schemes which would provide greater accuracy while also being more efficient. However as already mentioned, Godunov's order barrier theorem clearly states that a *linear* monotonicity preserving scheme can be first order accurate at best. Therefore, to construct a higher order scheme, researchers such as Van Leer chose to circumvent this theorem by focusing on non-linear schemes.

Van Leer in 1979, through a series of papers, generalized Godunov's method by using piece-wise linear polynomial reconstructions as shown in figure 5. He considered his schemes to be a second order sequel to Godunov's method. In addition to linear reconstructions, several monotonicity algorithms were applied to ensure monotonicity and non-linear stability (Van Leer, 1979). The information contained in the slopes of the linear reconstructions ensures that second order reconstruction is attained.

The piece-wise linear reconstruction can be represented as,

$$w(x, t^n) = \sum p_j(x) \chi_j(x) \tag{13}$$

With

$$p_j(x) = \bar{w}_j^n + w_j' \left(\frac{x - x_j}{\Delta x} \right) \tag{14}$$

so that, $\bar{p}_j(x_j) = \bar{w}_j^n$

Here w_j' is a slope limiter which is also reconstructed from the known cell averages. There are numerous choices for the slope limiter available in literature. By altering the slope of the linear reconstructions, non-oscillatory resolution of discontinuities can be obtained. According to Van Leer's initial scheme, in a monotonic sequence of cell averages, the onset of numerical oscillations are prevented by imposing a cap or limit on the slope of the linear distribution inside each cell. The limit should be strong enough to ensure that the value of the linear distribution does not exceed the value of the neighboring cell averages.

Even though obtaining the exact solution of the scalar nonlinear conservation law with linear initial data is a difficult task, the solution obtained at each cell interface for small time can be considered as the same as that obtained from the Riemann problem. Hence the numerical flux function can be used for the schemes of van Leer also (Barth & Ohlberger, 2004). Van Leer comments that the most pleasant property of his scheme is the clear physical picture it offers, as the discretization of the initial values gives rise to "*real gas slabs with fully specified internal distributions of state quantities*". Both conservation and nonlinear wave propagation are accounted for and additionally implementation of boundary conditions and adding extra physics such as radiation or multi-fluids are quite straightforward for the schemes.

In his paper, Van Leer mentions that an all-purpose code for compressible gas dynamics was being written along the above principles by P.R Woodward at the Leiden Observatory under the name of MUSCL (Monotone Upwind-Centered Scheme for Conservation Laws). Along the years, it has become common to refer to different variants of the schemes as MUSCL schemes.

The NASA Navier Stokes Solver OVERFLOW 2 which has both RANS and hybrid RANS-LES capabilities originally employed such MUSCL schemes (Nichols, Tramel, & Buning, 2008).

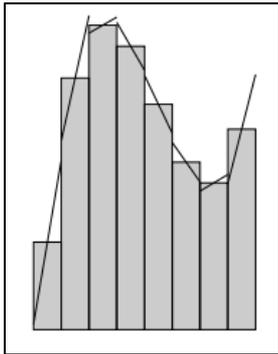


Figure 5: Linear reconstruction

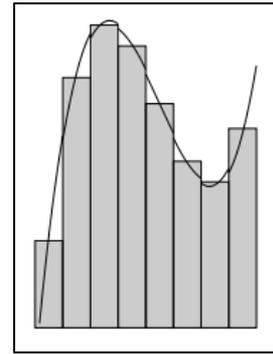


Figure 6: Quadratic reconstruction

The Piecewise Parabola Method (PPM) of Woodward and Colella and the Essentially Non-Oscillatory (ENO) schemes offer third and higher order Godunov-type upwind schemes respectively. PPM uses a quadratic approximation in each cell as shown in figure 6. The successful family of Essentially Non-Oscillatory schemes (Harten, Engquist, Osher, & Chakravarthy, 1987) is based on the reconstruction of several candidate stencils to approximate the flux at each cell interface, and selecting the stencil with the maximum smoothness. The stencil interpolated across the discontinuity is never used and hence the non-oscillatory property is achieved. The Weighted Essentially Non-Oscillatory schemes developed by Liu, Osher and Chan (Liu, Osher, & Chan, 1994) used a convex combination of all the candidate stencils instead of selecting just one stencil. This was achieved by assigning weights to each stencil corresponding to the contribution of each stencil to the numerical flux approximation. Any stencil which crossed a discontinuity was identified using a smoothness indicator and assigned a zero weight, hence achieving the non-oscillatory property. Such WENO schemes were found to be $(k + 1)^{th}$ order accurate in smooth monotone regions, where k is the order of the base ENO scheme.

Jiang and Shu in 1996, improved the WENO schemes by proposing a new way of measuring the smoothness of the numerical solution by minimizing the total variation for the approximation of the numerical flux in the semi discrete form of the conservation laws. This new smoothness indicator when replaced in the traditional equation for the calculation of the weight of the stencil resulted in a scheme that can be made $(2k - 1)^{th}$ order accurate in space, in smooth regions of the flux function. However at discontinuities, all WENO schemes are just k^{th} order accurate. The paper also suggests the use of pressure and entropy for the calculation of weights of the stencil for Euler systems, instead of the projected values of conserved variables in the characteristic field. However, results from other literature

referenced later on in this review indicates that the selection of primitive variables or projected characteristic variables give better and faster results (Jiang & Shu, 1996).

Of all the higher order schemes considered, the Essentially Non-Oscillatory (ENO) schemes and the Weighted Essentially Non-Oscillatory (WENO) schemes have displayed potential to replace the widely used MUSCL approach in codes for more accurate inviscid fluxes. Currently the WENO algorithm is widely used in high fidelity simulations, especially in the simulation of shock-turbulence interactions. This has resulted in the WENO schemes becoming the main candidate for the numerical method for the TU Delft solver.

It is also important to point out that the WENO schemes have come a long way from their traditional form introduced by Jiang and Shu. Several variants of the WENO schemes have been introduced over the years, a review of which is done in the next section. However the reader is suggested to read the explanation of the WENO schemes in chapter 3 before starting the review of the optimized schemes.

2.4.2 Optimized WENO schemes

An ideal numerical method would be non-dissipative and non-dispersive in the smooth parts of the flow while having efficient shock capturing capability, hence preventing the formation of oscillations (Gibbs phenomenon) which leads to nonlinear instability. However these requirements contradict themselves as shock capturing requires the addition of artificial dissipation as a surrogate for physical viscosity to obtain non-oscillatory transitions. Another requirement is to have adequate spatial resolution (Johnsen, et al., 2010). Many researchers have tried to improve the performance of the WENO schemes especially with regards to their performance in the smooth regions of flow.

Fourier analysis is a good method to conduct an investigation into the scale resolving capability of a numerical scheme. The phase and amplitude error can be determined by such an analysis. Wavenumber κ is defined for the scheme in Fourier space. A modified wave number κ' , which is a function of the scaled wavenumber, and characterizes the numerical scheme through its dependence on the stencil coefficients, is calculated. The real part of this modified wavenumber is an indication of the phase error of the scheme and the imaginary part indicates the amplitude error, which is in turn an indication of the dispersion and dissipation of the scheme. A scheme with no error would satisfy $\kappa'(\kappa) = \kappa$, for all $0 \leq \kappa \leq \pi$. The phase error can usually be denoted simply by $(\kappa' - \kappa) / \kappa$. Lele (Lele, 1992) defines the resolving efficiency as the fraction of the range of κ such that the error is below some tolerance ϵ .

Weirs and Candler assessed WENO schemes for DNS of compressible turbulence (Weirs & Candler, 1997). The focus in this paper is on the range of scales that is accurately resolved by the scheme. By conducting a Fourier analysis, they have concluded from calculations that when the optimal stencil is chosen, a fifth order traditional WENO scheme requires at least 6.3 Points Per Wavelength ($PPW = 2\pi / \kappa$, is the minimum number of grid points required to resolve one wavelength). Also a resolving efficiency of 0.35 is found for the same scheme. Two sources of numerical dissipation were identified, the first being the optimal stencil which is specified for maximum order of accuracy and the second being the smoothness

measurement, which can allow the stencil to adapt at times when the solution is smooth. Bandwidth optimizing techniques are proposed to minimize the approximation error at high frequencies which correspond to the small scales, and to develop symmetric optimal stencils with reduced dissipation and greater resolving efficiency according to the method employed by Tam and Webb (1993). Optimization of the optimal stencils is done similar to the method employed by Tam and Webb in the development of Dispersion-relation-preserving schemes for computational aeroacoustics, where the focus is on optimizing the phase error on symmetric stencils. Weirs and Candler achieved the optimization by minimizing an integrated error function which is a linear combination of the phase error and the amplitude error, and hence calculating new values for the optimal linear weights. The dissipation error is reduced by identifying the source of dissipation in the order optimized stencil, which is the upwinding, and then converting the stencil to symmetric by adding a candidate stencil. Numerical experiments using such bandwidth optimized schemes have shown marginal improvements, as the adaptation mechanism with the smoothness measurement still introduces nonlinear errors which limits such schemes.

Wang and Chen (Wang & Chen, 2001) improved the optimization process in two levels. First, optimized scheme coefficients for all candidate stencils are developed to minimize the solution error in wavenumber space. Then these optimized schemes are convexly combined with weights optimized to achieve both high order of accuracy and better resolution for short waves. Also new smoothness indicators were presented as they were identified to be a source of numerical dissipation for traditional WENO schemes. This scheme was shown to give very good results for a variety of test cases. However when applied to Euler equations in the conservation form, it was found that for the scheme to adequately capture the shocks, the linearized Euler equation should be used.

Ponziani, Pirozzoli and Grasso (Ponziani, Pirozzoli, & Grasso, 2003) combined the above two approaches, and optimized the interpolation coefficients for each candidate stencil and the linear weights. Also, to minimize the dissipation error and to prevent the adaption mechanism from selecting stencils with discontinuities, an additional candidate stencil is chosen thus giving symmetric stencil support to the scheme. Results from numerical experiments with test cases with a range of scales indicate that this scheme is indeed capable of extending the range of well resolved wave numbers. But the optimization becomes less effective when the grid is refined due to the effect of the drop in the order of accuracy the scheme had to sustain in the optimization process. However, the scheme was able to show considerable improvements when tested for nonlinear problems for a DNS test case.

Martin et al. (Martin, EM, Wu, & Weirs, 2006) carried out a similar optimization of WENO for application to DNS of compressible turbulence. He introduced an additional candidate stencil to render the scheme stencil symmetric and also conducted bandwidth optimization for the linear optimal weights. Applications on test cases with smooth linear solutions indeed revealed that these schemes have reduced numerical dissipation compared to the traditional WENO schemes. However for nonlinear solutions, the presence of the shock makes the scheme drop the order of accuracy much lower, possibly to first order, leading to much slower grid convergence. He suggests the optimization of the nonlinear part of the scheme in order to realize the full effects of bandwidth optimization of the WENO scheme for LES.

Taylor et al., (Taylor, Wu, & Martin, 2001) in his study on optimization of nonlinear errors for WENO in DNS, identifies the smoothness measurement and the coefficients of the candidate stencils when the adaptation kicks in, as the sources of nonlinear errors. He suggests that “because the optimal stencil of a WENO scheme provides optimal performance in smooth regions, any technique that discourages unnecessary adaptation can improve WENO dissipation characteristics in compressible turbulence. This approach requires modification of the smoothness measurement but does not affect the candidate stencil coefficients or the optimal stencil weights.” A modified limiting procedure is introduced for the smoothness indicators, which ensures that the indicators always take on effectual values. This is important because the previous definition of the indicator triggers the adaptation mechanism very easily which leads to degradation of the scheme performance. Results for the scheme for DNS of compressible turbulence, with both linear and nonlinear errors optimized, show increased order of accuracy.

3

Numerical scheme implementation

This chapter explains the skeleton of the solver and explains the steps involved in the implementation of a numerical scheme in the solver. Section 3.1 details the finite volume framework of the solver. Section 3.2 explains the mathematical foundation on which the WENO schemes are built on and the different steps involved in the WENO reconstruction.

3.1. Finite Volume Framework in three dimensions

For a three dimensional Cartesian grid, the control volume can be represented by,

$$I_{ijk} = \left[x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}} \right] \times \left[y_{i-\frac{1}{2}}, y_{i+\frac{1}{2}} \right] \times \left[z_{i-\frac{1}{2}}, z_{i+\frac{1}{2}} \right],$$

And the dimensions by,

$$\Delta x = \left[x_{i+\frac{1}{2}} - x_{i-\frac{1}{2}} \right], \quad \Delta y = \left[y_{i+\frac{1}{2}} - y_{i-\frac{1}{2}} \right], \quad \Delta z = \left[z_{i+\frac{1}{2}} - z_{i-\frac{1}{2}} \right]$$

The three dimensional hyperbolic system in conservative form can be written as,

$$\frac{\partial q}{\partial t} + \frac{\partial}{\partial x} F(U) + \frac{\partial}{\partial y} G(U) + \frac{\partial}{\partial z} H(U) = 0,$$

(15)

Where $U(x, y, z, t)$ is the vector of conservative variables and $F(U), G(U), H(U)$ are the vectors for the physical flux of the conservative variables in the three co-ordinate directions x, y, z respectively. Similar to the one dimensional case, the semi-discrete finite volume relation in a Cartesian grid can be expressed as

$$\frac{d\bar{U}_{ijk}(t)}{dt} = - \left[\left(F_{i+\frac{1}{2},j,k} - F_{i-\frac{1}{2},j,k} \right) + \left(G_{i,j+\frac{1}{2},k} - G_{i,j-\frac{1}{2},k} \right) + \left(H_{i,j,k+\frac{1}{2}} - H_{i,j,k-\frac{1}{2}} \right) \right]$$

(16)

Where $\bar{U}_{ijk}(t)$ is the cell volume average of the solution in cell I_{ijk} at time t :

$$\bar{U}_{ijk}(t) = \frac{1}{\Delta x \Delta y \Delta z} \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \int_{y_{i-\frac{1}{2}}}^{y_{i+\frac{1}{2}}} \int_{z_{i-\frac{1}{2}}}^{z_{i+\frac{1}{2}}} U(x, y, z, t) dz dy dx$$

(17)

And $F_{i+\frac{1}{2},j,k}$, $G_{i,j+\frac{1}{2},k}$, and $H_{i,j+\frac{1}{2},k}$, are the averages of the physical fluxes over the cell faces at time t:

$$F_{i+\frac{1}{2},j,k} = \frac{1}{\Delta x \Delta y \Delta z} \int_{y_{i-\frac{1}{2}}}^{y_{i+\frac{1}{2}}} \int_{z_{i-\frac{1}{2}}}^{z_{i+\frac{1}{2}}} F(U(x + 1/2, y, z, t)) \cdot \hat{n}_1 \cdot dz dy$$

$$G_{i,j+\frac{1}{2},k} = \frac{1}{\Delta x \Delta y \Delta z} \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \int_{z_{i-\frac{1}{2}}}^{z_{i+\frac{1}{2}}} G(U(x, y + 1/2, z, t)) \cdot \hat{n}_2 \cdot dz dx$$

$$H_{i,j,k+\frac{1}{2}} = \frac{1}{\Delta x \Delta y \Delta z} \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \int_{y_{i-\frac{1}{2}}}^{y_{i+\frac{1}{2}}} H(U(x, y, z + 1/2, t)) \cdot \hat{n}_3 \cdot dy dx$$

(18)

By creating higher order numerical approximations to the exact flux values F, G and H, high order accurate semi-discrete schemes can be constructed. These approximations of the physical fluxes, ie, the numerical fluxes will henceforth be represented as $\hat{F}_{i+\frac{1}{2},j,k}$, $\hat{G}_{i,j+\frac{1}{2},k}$, and $\hat{H}_{i,j+\frac{1}{2},k}$.

Evaluating the numerical approximation of the physical flux at the cell faces starts with the discretization of the integrals over the faces using a Gaussian numerical quadrature. An expression for the numerical flux in the x-direction at the cell face is given below:

$$F_{i+\frac{1}{2},j,k} = \frac{W_\alpha W_\beta}{\Delta x \Delta y \Delta z} \sum_{\alpha=1}^N \sum_{\beta=1}^N F(U(x + 1/2, y_\alpha, z_\beta))$$

(19)

Where α and β are the different Gaussian quadrature points and W_α, W_β are the corresponding weights for these points. Similar expressions can be written for the cell face flux in the y and z coordinate directions.

The next step is the calculation of the point value U at the cell face, by reconstruction from the cell averaged values at the cell centres and obtaining high order approximations at the quadrature points. This reconstruction using the WENO method is detailed in section 3.2. The

WENO reconstruction yields two sets of values, corresponding to the left and right reconstructed values at $x_{i+1/2}$, ie, corresponding to $x_{i+1/2} - 0$ and $x_{i+1/2} + 0$.

$$U_{i+\frac{1}{2},\alpha,\beta}^L = U\left(x_{i+\frac{1}{2}} - 0, y_\alpha, z_\beta\right),$$

$$U_{i+\frac{1}{2},\alpha,\beta}^R = U\left(x_{i+\frac{1}{2}} + 0, y_\alpha, z_\beta\right)$$

In the final step, at each quadrature point, a Riemann problem is solved using an approximate Riemann solver (detailed in section 3.3). The left and right reconstructed values at each quadrature point from the previous step acts as the two states in the Riemann problem. The numerical flux at the face is then calculated using the left and right extrapolated values,

$$F_{i+\frac{1}{2},j,k} = \frac{W_\alpha W_\beta}{\Delta x \Delta y \Delta z} \sum_{\alpha=1}^N \sum_{\beta=1}^N \hat{F}(U_{i+1/2,\alpha,\beta}^L, U_{i+1/2,\alpha,\beta}^R)$$

(20)

Similar expressions can be written for the numerical flux in y and z coordinate directions. The steps described in this section is summarized below:

Step 1: Approximations of the point value U at the cell face centres $U^L(x_{i+\frac{1}{2}}, y, z)$ and $U^R(x_{i+\frac{1}{2}}, y, z)$ is done by reconstructing the cell-averaged value using WENO reconstruction on the stencils S_i and S_{i+1} respectively.

Step 2: High order approximations are then obtained at the quadrature points using these reconstructed cell face centred values $U_{i+\frac{1}{2},\alpha,\beta}^L$ and $U_{i+\frac{1}{2},\alpha,\beta}^R$.

Step 3: At each quadrature point, a Riemann problem is solved using the approximate Riemann solver and the numerical flux is calculated by assigning weights to the flux at the different quadrature points according to equation 20.

Step 4: The solution is advanced in time using a sufficiently high order scheme as described in section 3.4

3.2 WENO reconstruction

3.2.1 General reconstruction from cell averages in 1d

Given the location I_i of a cell, and the order of accuracy k , a stencil is chosen including I_i , r cells to the left, and s cells to the right, with,

$$r + s + 1 = k$$

Therefore, a typical stencil can be expressed as,

$$S(i) \equiv \{I_{i-r}, \dots, I_{i+s}\}$$

(21)

The k^{th} order approximation polynomial which is to be calculated is represented by $p_i(x)$. This polynomial is calculated by considering a unique polynomial of degree at the most $k - 1 = r + s$, whose cell average in each of the cells in the stencil $S(i)$ agrees with that of $U(x)$, ie,

$$\frac{1}{\Delta x_j} \int_{x_{j-\frac{1}{2}}}^{x_{j+\frac{1}{2}}} p_i(\xi) d\xi = \bar{U}_j, \quad j = i - r, \dots, i + s$$

(The subscript i for the polynomial p is dropped in equation to avoid confusion)

Approximations to the values of $U(x)$ at the cell faces are required to calculate the numerical flux at the face. There exists constants c_{rj} and \bar{c}_{rj} which depend on the number of points to the left r in equation 21, the order of accuracy k and on the cell size Δx_j in the stencil S_i , ie,

$$U_{i+\frac{1}{2}}^- = \sum_{j=0}^{k-1} c_{rj} \bar{U}_{i-r+j}$$

$$U_{i-\frac{1}{2}}^+ = \sum_{j=0}^{k-1} \bar{c}_{rj} \bar{U}_{i-r+j}$$

(22)

At the same location $x_{i+\frac{1}{2}}$, there will be two different values corresponding to the superscripts \pm because of the possibility of different stencils for cells I_i and I_{i+1} . One way of removing these superscripts is by using the stencil S_i to approximate $x_{i+\frac{1}{2}}$. Also it should be noted that $\bar{c}_{rj} = c_{r-1,j}$ which allows \bar{c}_{rj} also to be dropped, giving the relation,

$$U_{i+\frac{1}{2}} = \sum_{j=0}^{k-1} c_{rj} \bar{U}_{i-r+j}$$

(23)

which is the k -th order accurate approximation of U at $x_{i+\frac{1}{2}}$. The constants c_{rj} are independent of the value of the function U . They can be computed using the following equation,

$$c_{rj} = \left[\sum_{m=j+1}^k \frac{\sum_{l=0}^k \prod_{\substack{q=0 \\ q \neq m, l}}^k \left(x_{i+\frac{1}{2}} - x_{i-r+q-\frac{1}{2}} \right)}{\prod_{\substack{l=0 \\ l \neq m}}^k \left(x_{i-r+m-\frac{1}{2}} - x_{i-r+l-\frac{1}{2}} \right)} \right] \Delta x_{i-r+j} \quad (24)$$

If the grid is uniform, $\Delta x_i = \Delta x$, then the constants are not dependent on i or Δx_i giving,

$$c_{rj} = \sum_{m=j+1}^k \frac{\sum_{l=0}^k \sum_{\substack{q=0 \\ q \neq m, l}}^k (r - q + 1)}{\prod_{\substack{l=0 \\ l \neq m}}^k (m - l)} \quad (25)$$

3.2.2 WENO approximation in 1d

Piecewise smooth functions were always deemed interesting for the solution of hyperbolic conservation laws. A function can be termed as piecewise smooth if it is able to satisfy the existence of derivatives as required by the numerical scheme, except for at finite isolated points. At these isolated points, the function is still assumed to have finite left and right limits. But at these discontinuities a fixed stencil approximation as described in Section 3.2.1 will not be successful (Cockburn, Johnson, Shu, & Tadmor, 1997).

The problem of involving the cell with the discontinuity in the stencil were overcome by the ENO approximations which are based on the idea of *adaptive stencils*. As the name suggests, the stencil adapts itself in the area of a discontinuity and make sure the “bad” cell is not included. The idea of adaptive stencils requires a method to identify bad cells or in other words, to determine the smoothness of a stencil. For the ENO schemes, the Newton divided differences act as the smoothness measurement of the function inside a stencil. The WENO schemes were introduced as an improvement on the ENO schemes. WENO uses a convex combination of all candidate stencils, compared to the ENO practise of using just one of the candidate stencils. This means that if k candidate stencils are considered,

$$S_r(i) = \{x_{i-r}, \dots, x_{i-r+k-1}\}, \quad r = 0, \dots, k-1$$

Then k different reconstructions for the value at the face $x_{i+\frac{1}{2}}$ will be produced, ie,

$$U_{i+\frac{1}{2}}^r = \sum_{j=0}^{k-1} c_{rj} \bar{U}_{i-r+j}, \quad r = 0, \dots, k-1 \quad (26)$$

The WENO schemes would then take a convex combination of all $U_{i+\frac{1}{2}}^r$ defined in above equation and form a new approximation to the cell boundary value $U(x_{i+\frac{1}{2}})$ as:

$$U_{i+\frac{1}{2}} = \sum_{r=0}^{k-1} w_r U_{i+\frac{1}{2}}^r$$

(27)

To satisfy the requirements of consistency and stability, the weights are chosen such that it satisfies,

$$w_r \geq 0, \quad \sum_{r=0}^{k-1} w_r = 1$$

In the case of a smooth function in all the candidate stencils, constants d_r exist such that

$$U_{i+\frac{1}{2}} = \sum_{r=0}^{k-1} d_r U_{i+\frac{1}{2}}^r = U(x_{i+\frac{1}{2}}) + O(\Delta x^{2k-1})$$

The values of these constant d_r is always positive and due to consistency,

$$\sum_{r=0}^{k-1} d_r = 1$$

Therefore, in such cases the preferred selection of weights should be such that,

$$w_r = d_r + O(\Delta x^{k-1}), \quad r = 0, \dots, k-1$$

Hence from equations 1 and 2,

$$\begin{aligned} \sum_{r=0}^{k-1} w_r U_{i+\frac{1}{2}}^r - \sum_{r=0}^{k-1} d_r U_{i+\frac{1}{2}}^r &= \sum_{r=0}^{k-1} (w_r - d_r) \left(U_{i+\frac{1}{2}}^r - U(x_{i+\frac{1}{2}}) \right) \\ &= \sum_{r=0}^{k-1} O(\Delta x^{k-1}) O(\Delta x^k) = O(\Delta x^{2k-1}) \end{aligned}$$

This implies an order of accuracy of $2k - 1$ and hence,

$$U_{i+\frac{1}{2}} = \sum_{r=0}^{k-1} w_r U_{i+\frac{1}{2}}^r = U(x_{i+\frac{1}{2}}) + O(\Delta x^{2k-1})$$

In the event of the function being discontinuous in one or more stencils, the weights corresponding to these stencils need to be zero. The weights selected for this work are of the form,

$$w_r = \frac{\alpha_r}{\sum_{s=0}^{k-1} \alpha_s}, \quad r = 0, \dots, k-1$$

Where

$$\alpha_r = \frac{d_r}{(\epsilon + \beta_r)^2}$$

(28)

Where β_r is the smoothness indicator and $\epsilon = 10^{-6}$ is introduced to prevent the denominator from becoming zero. There are different choices for the smoothness indicator. However for any smoothness indicator, it is clear that equation 1 satisfies equation 2. The smoothness indicator selected for this work is the one created by Jiang and Shu, based on minimizing the total variation of the reconstruction polynomial inside a cell. This is an effective method because the total variation is a good measurement of smoothness; and on each stencil $S_r(i)$, a $(k-1)$ -th degree reconstruction polynomial which approximates $q(x + \frac{1}{2})$ when evaluated at $x = x_{i+\frac{1}{2}}$ can be constructed. Minimizing the total variation of this polynomial inside each cell is desirable and hence this can be used as a measure for the smoothness of the stencil.

Let the reconstruction polynomial on stencil $S_r(i)$ be denoted by $p_r(x)$, then the smoothness indicator β_r is defined as

$$\beta_r = \sum_{l=1}^{k-1} \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \Delta x^{2l-1} \left(\frac{\partial^l p_r(x)}{\partial^l x} \right)^2 dx$$

(29)

For a given cell I_i , the WENO reconstruction procedure can be summarized in the following steps:

Step 1: The stencils S_r , $r = 0, \dots, k-1$, are identified such that I_i is part of each stencil. The larger stencil which contains all the cells from the smaller k stencils is denoted by T .

Step 2: For each of the stencils S_r , a lower order approximation polynomial $p_r(x)$ is constructed. A higher order polynomial $P_i(x)$ for the larger stencil T is also constructed.

Step 3: The coefficients to combine the lower order polynomials $p_r(x)$, ie, the linear weights d_j , are found out such that $P_i(x) = \sum d_r p_r(x)$

Step 4: The smoothness indicator β_r is computed for each stencil S_r according to equation 29, which gives the measure of smoothness of the polynomial p_r in the cell I_i .

Step 5: The non-linear weight w_r is calculated based on the smoothness indicators according to equation 28.

Step 6: The final WENO reconstruction is obtained by a convex combination of all the polynomials using the weights according to equation 27.

3.2.3 Fifth order WENO reconstruction:

In this work, the fifth order WENO reconstruction is used. The steps described in the previous section is illustrated with $k = 3$ to obtain two values at a cell face corresponding to \pm or L and R which form the two states for the Riemann problem. A uniform mesh is assumed.

Consider a cell $I_i = \left(x_{i-\frac{1}{2}} - x_{i+\frac{1}{2}}\right)$ with cell averages of a function U represented by \bar{U}_i . We proceed step-by-step to obtain a fifth order WENO reconstruction to the point value $U\left(x_{i+\frac{1}{2}}\right)$ represented by $U_{i+\frac{1}{2}}^L$.

Step 1: Identification of stencils: The stencil is based on five cells $\{I_{i-2}, I_{i-1}, I_i, I_{i+1}, I_{i+2}\}$. The three stencils will be

$$S_0 = \{I_{i-2}, I_{i-1}, I_i\}$$

$$S_1 = \{I_{i-1}, I_i, I_{i+1}\}$$

$$S_2 = \{I_i, I_{i+1}, I_{i+2}\}$$

as shown in figure 7. Hence the larger stencil \mathcal{T} will be $\mathcal{T} = \{I_{i-2}, I_{i-1}, I_i, I_{i+1}, I_{i+2}\}$

Step 2: Constructing polynomials: Three polynomials p_0, p_1, p_2 represent the lower order approximations. They are of degree at the most two with their cell averages equal to those of the function U in the five cells in the stencil and each of these polynomials are third order approximations to $U\left(x_{i+\frac{1}{2}}\right)$. They are,

$$p_0\left(x_{i+\frac{1}{2}}\right) = \frac{1}{3}\bar{U}_{i-2} - \frac{7}{6}\bar{U}_{i-1} + \frac{11}{6}\bar{U}_i$$

$$p_1\left(x_{i+\frac{1}{2}}\right) = \frac{1}{6}\bar{U}_{i-1} - \frac{5}{6}\bar{U}_i + \frac{1}{3}\bar{U}_{i+1}$$

$$p_2\left(x_{i+\frac{1}{2}}\right) = \frac{1}{3}\bar{U}_i - \frac{5}{6}\bar{U}_{i+1} + \frac{1}{6}\bar{U}_{i+2}$$

(30)

The high order polynomial $Q\left(x_{i+\frac{1}{2}}\right)$ of order at the most four and which is a fifth order approximation to $U\left(x_{i+\frac{1}{2}}\right)$ is,

$$Q\left(x_{i+\frac{1}{2}}\right) = \frac{1}{30}\bar{U}_{i-2} - \frac{13}{60}\bar{U}_{i-1} + \frac{47}{60}\bar{U}_i + \frac{9}{20}\bar{U}_{i+1} - \frac{1}{20}\bar{U}_{i+2}$$

(31)

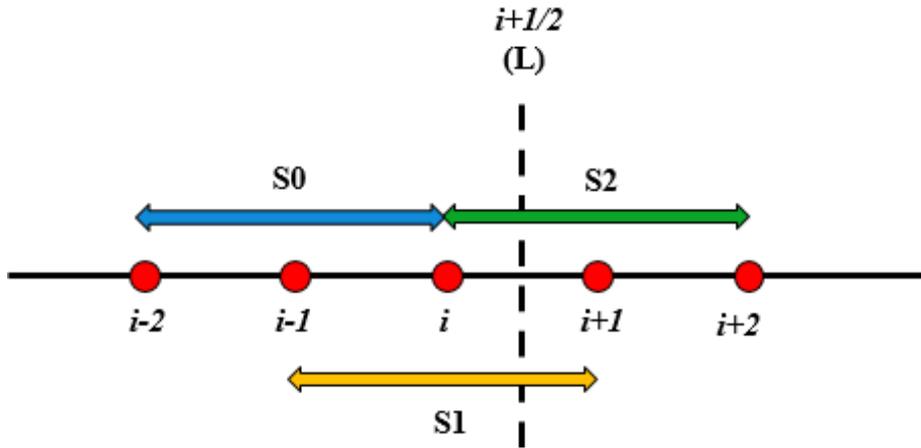


Figure 7: Stencils for WENO reconstruction at cell face (L) with $k = 3$

A simple algorithm is written to calculate the constants $\bar{c}_{r,j}$ from equation 25. For $k=3$ they are calculated and tabulated in Table 1.

K	r	j = 0	j = 1	j = 2	j = 3	j = 4
3	-1	11/6	-7/6	1/3		
	0	1/3	5/6	-1/6		
	1	-1/6	5/6	1/3		
	2	1/3	-7/6	11/6		
5	-1	137/60	-163/60	137/60	-21/20	1/5
	0	1/5	77/60	-43/60	17/60	-1/20
	1	-1/20	9/20	47/60	-13/60	1/30
	2	1/30	-13/60	47/60	9/20	-1/20
	3	-1/20	17/60	-43/60	77/60	1/5
	4	1/5	-21/20	137/60	-163/60	137/60

Table 1: Coefficients for lower order polynomials in WENO reconstruction

Step 3: Calculation of linear weights: The linear weights are calculated to satisfy,

$$Q\left(x_{i+\frac{1}{2}}\right) = d_0 p_0\left(x_{i+\frac{1}{2}}\right) + d_1 p_1\left(x_{i+\frac{1}{2}}\right) + d_2 p_2\left(x_{i+\frac{1}{2}}\right) \quad (32)$$

Again, the algorithm is used to calculate the weights and the resulting values are,

$$d_0 = \frac{1}{10} \quad d_1 = \frac{3}{5} \quad d_2 = \frac{3}{10} \quad (33)$$

Step 4: Smoothness indicators: Equation 29 is solved for $k = 3$ to obtain the relations for smoothness indicators for each stencil,

$$\beta_0 = \frac{13}{12}(U_{i-2} - 2U_{i-1} + U_i)^2 + \frac{1}{4}(U_{i-2} - 4U_{i-1} + 3U_i)^2$$

$$\beta_1 = \frac{13}{12}(U_{i-1} - 2U_i + U_{i+1})^2 + \frac{1}{4}(U_{i-1} - U_{i+1})^2,$$

$$\beta_1 = \frac{13}{12}(U_i - 2U_{i+1} + U_{i+2})^2 + \frac{1}{4}(3U_i - 4U_{i+1} + U_{i+2})^2$$

(34)

Step 5 and **Step 6** are then straightforward calculations using the relations from Steps 1-4.

The calculations to obtain the Right fifth order WENO reconstruction to the point value $U(x_{i+\frac{1}{2}})$, ie, $U_{i+\frac{1}{2}}^R$ is exactly symmetric to its Left counterpart. This is achieved by replacing i with $i+1$, ie, repeating the above procedures for cell I_{i+1} . The stencils are based on the cells $\{I_{i-1}, I_i, I_{i+1}, I_{i+2}, I_{i+3}\}$ as shown in figure 8. The rest of the steps are intuitive and hence not explained to avoid repetition.

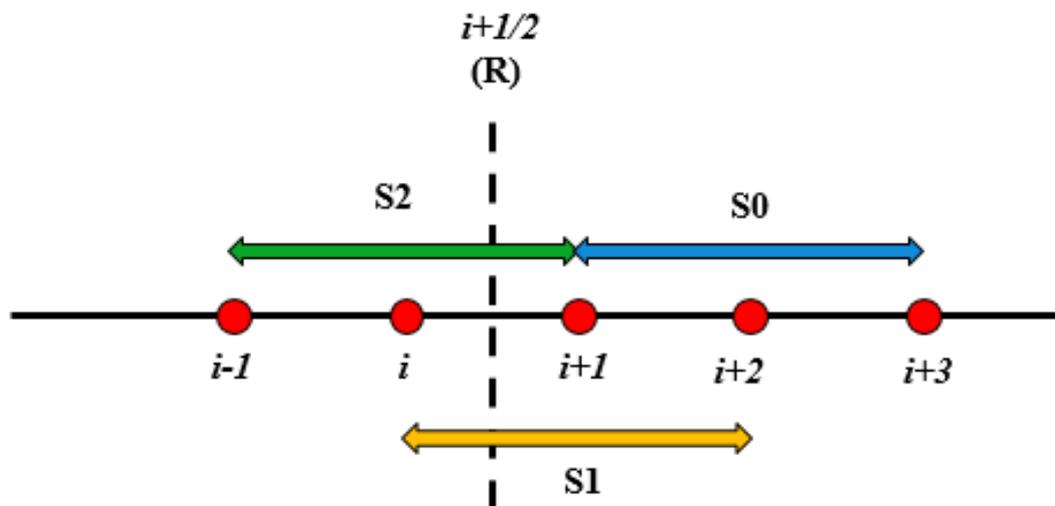


Figure 8 : Stencils for WENO reconstruction at cell face (R^+) with $k = 3$

3.2.4 Reconstruction on quadrature points

According to equation 20, the flux has to be calculated at each quadrature point which means that the function has to be reconstructed at each quadrature point. The steps considered above is valid for the WENO reconstruction in one dimension at the point $\bar{x}_{i+\frac{1}{2}}$. For multi-dimensions, when Gaussian quadrature points are involved, reconstruction is done to compute point-wise value of U at the Gaussian quadrature points $x_{i+\frac{1}{2}}, y_\alpha, z_\beta$ where α and β are the location of the quadrature points. In this work, for the fifth order WENO reconstruction, the two point Gaussian quadrature is used. This means that the reconstructions have to be made at $-\frac{1}{\sqrt{3}}$ and $\frac{1}{\sqrt{3}}$.

$$\int_{-1}^1 \phi(\xi) d\xi = \phi\left(-\frac{1}{\sqrt{3}}\right) + \phi\left(+\frac{1}{\sqrt{3}}\right)$$

(35)

The reconstruction is done dimension by dimension as stated in (Titarev & Toro, 2004). Considering a cell denoted by $I_{i_x i_y i_z}$, in the first step, for all indexes i_y, i_z from the stencil, one dimensional WENO reconstruction is done in the x direction, ie, normal to the cell face. This results in two-dimensional averages with respect to y-z coordinate,

$$V_{i_y i_z}^L = \frac{1}{\Delta y \Delta z} \int_{y_{i_y-1/2}}^{y_{i_y+1/2}} \int_{z_{i_z-1/2}}^{z_{i_z+1/2}} q\left(x_{i+\frac{1}{2}} - 0, y, z\right) dz dy$$

(36)

In the second step, one-dimensional reconstruction in y direction for all values of i_z and obtain one-dimensional averages of the solution with respect to z coordinate direction in lines which corresponds to the Gaussian quadrature points on the y axis, ie along $x = x_{i+\frac{1}{2}}, y = y_\alpha$

$$W_{i_y i_z}^L = \frac{1}{\Delta z} \int_{z_{i_z-1/2}}^{z_{i_z+1/2}} q\left(x_{i+\frac{1}{2}} - 0, y_\alpha, z\right) dz$$

(37)

In the final step, for each line, reconstructed point-wise values $q^L\left(x_{i+\frac{1}{2}}, y_\alpha, z_\beta\right)$ by the one dimensional reconstruction of $W_{i_y i_z}^L$ in the z direction.

The WENO reconstruction in the x coordinate direction, ie, normal to the cell face corresponds to the left and right reconstructed values at $x_{i+\frac{1}{2}}$. But the y and z sweeps, ie, the second and final step, requires reconstruction at the quadrature points for which the coefficients and the linear weights in the reconstruction formulas should be recalculated. This is done by replacing $x_{i+\frac{1}{2}}$ in equation 24 to the quadrature points, ie, $x_{i-\frac{1}{\sqrt{3}}}$ and $x_{i+\frac{1}{\sqrt{3}}}$. All the other steps are same as before. The corresponding coefficients and linear weights are given in Table 2 and equation 38.

Quadrature location	R	j = 0	j = 1	j = 2
$i - \frac{1}{\sqrt{3}}$	0	$-\sqrt{3}/12$	$4\sqrt{3}/12$	$12-3\sqrt{3}/12$
	1	$\sqrt{3}/12$	1	$-\sqrt{3}/12$
	2	$12+3\sqrt{3}/12$	$-4\sqrt{3}/12$	$\sqrt{3}/12$
$i + \frac{1}{\sqrt{3}}$	0	$\sqrt{3}/12$	$-4\sqrt{3}/12$	$12+3\sqrt{3}/12$
	1	$-\sqrt{3}/12$	1	$\sqrt{3}/12$
	2	$12-3\sqrt{3}/12$	$4\sqrt{3}/12$	$-\sqrt{3}/12$

Table 2: Coefficients for lower order polynomials for the two point Gaussian quadrature

$$d_0 = 210 + \frac{\sqrt{3}}{1080}; d_1 = \frac{11}{18}; d_2 = 210 - \frac{\sqrt{3}}{1080}$$

(38)

3.2.5 Reconstruction for system of equations

The WENO reconstruction process for systems is carried out using characteristic variables. This is done by transforming the conserved variables to characteristic space and then performing the steps for WENO reconstruction. Then these reconstructed characteristic variables are transformed back to the conserved variables. This decomposition is expensive, but it is necessary to prevent the occurrence of spurious oscillations in some cases.

In this work a hyperbolic 5 x 5 system, ie, the Euler equations are considered. The Jacobian of the Euler equations, $A(U)$, has 5 real Eigen values,

$$\lambda_1(U) \leq \dots \leq \lambda_5(u),$$

(39)

a set of independent right Eigen vectors,

$$r_1(U) \leq \dots \leq r_5(U)$$

(40)

The matrix whose columns are the Eigen vectors is given by,

$$R(U) = (r_1(U), \dots, r_5(U))$$

(41)

This leads to the relation,

$$R^{-1}(U)A(U)R(U) = \Lambda(U)$$

Where $\Lambda(U)$ is the diagonal matrix with $\lambda_1(U), \dots, \lambda_4(U)$ as the diagonal elements, and $R^{-1}(U)$ is the matrix whose rows are represented by the left Eigen vectors of the Jacobian, $l_1(U), \dots, l_4(U)$.

The steps involved in the characteristic decomposition are now listed. At each cell interface $x_{i+1/2}$,

Step 1: An approximation to the flux Jacobian at the cell interface is required. For this, an average state of U at the cell face is calculated using the Roe average (density weighted average).

Step 2: The right eigenvectors, left eigenvectors and the Eigen values of the Jacobian are computed and denoted by,

$$R = R \left(\bar{U}_{i+\frac{1}{2}} \right) \quad L = R^{-1} \left(\bar{U}_{i+\frac{1}{2}} \right) \quad \Lambda = \Lambda \left(\bar{U}_{i+\frac{1}{2}} \right)$$

Step 3: The state variables in the conservative form which are in the potential stencil of the WENO reconstruction is transformed to the characteristic space by,

$$\bar{V}_j = R^{-1} U_j$$

where j is in a neighbourhood of i

Step 4: The WENO reconstruction is performed according to Sections 3.2.3 and 3.2.4 for each characteristic variable \bar{V} to obtain $V_{i+\frac{1}{2}}^\pm$

Step 5: Each of the reconstructed variable $\left(V_{i+\frac{1}{2}} \right)^\pm$ in the characteristic space is transformed back into conserved variables by

$$\left(U_{i+\frac{1}{2}} \right)^\pm = R \left(V_{i+\frac{1}{2}} \right)^\pm$$

Step 6: An approximate Riemann solver is used to compute the flux at the cell face for all i

3.3 Approximate Riemann solvers

Most numerical schemes solve the hyperbolic conservation laws by solving a series of Riemann problems. Riemann solvers, either exact or approximate provide an element of physical realism, in that they correctly identify and discriminate between information which should propagate with different speeds, with the most basic of the differentiation being between which information should travel to the left and which to the right (Roe, 1981).

As stated by Roe, the very high cost of using an exact Riemann solver is justifiable only if we are using the abundance of information made available to some sophisticated use. Hence the way in which we intend to use the solution can determine the level of accuracy with which the Riemann problem need to be solved. Less accurate but faster and hence less costly solutions can be used in numerical schemes. This prompted the use of approximate Riemann

solvers. In this section various approximate Riemann solvers for non-linear Riemann problems are discussed.

3.3.1 Harten-Lax-Van Leer schemes

The exact solution of the non-linear Riemann problem, for example, the Euler equations is shown in figure 7. There are five family of waves associated with each of the Eigen values $u - a$, u and $u + a$. The eigenvalue u has a multiplicity of three and hence represents three wave families.

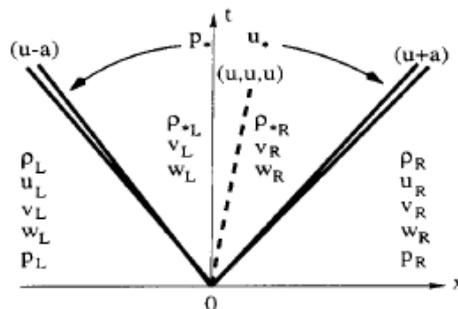


Figure 9: Exact solution of a Riemann problem (Toro, 1999)

The Harten-Lax-Van Leer (HLL) schemes use the upper and lower bounds of the characteristic speeds for the solution of the Riemann problem. They were first proposed by Harten, Lax and Van Leer (Harten, Lax, & Van Leer, 1983).

The structure of the HLL approximate solution to the Riemann problem is shown in figure 9.

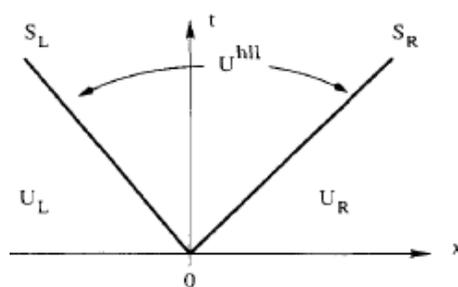


Figure 10: HLL approximate solution to the Riemann problem (Toro, 1999)

It is clear that only three constant states exist which are separated by two waves. The single constant state U^{hll} represents the star region. All the intermediate states in the star region (ie, the region inside the Riemann fan as shown in figure 10), are bundled into this constant state.

The following approximate solution was proposed,

$$U(x,t) = \begin{cases} U_L & \frac{x}{t} \leq S_L \\ U^{hll} & S_L \leq \frac{x}{t} \leq S_R \\ U_R & \frac{x}{t} \geq S_R \end{cases}$$

Where U^{hll} is the constant state vector given by,

$$U^{hll} = \frac{S_R U_R - S_L U_L + F_L - F_R}{S_R - S_L}$$

And S_L and S_R are the fastest wave speeds perturbing the initial data states U_L and U_R .

The corresponding flux F^{hll} is obtained from the following relation,

$$F^{hll} = \frac{S_R F_L - S_L F_R + S_L S_R (U_R - U_L)}{S_R - S_L}$$

Thus, the flux at each cell interface is given by,

$$F_{i+\frac{1}{2}}^{hll} = \begin{cases} F_L & 0 \leq S_L \\ F^{hll} & S_L \leq 0 \leq S_R \\ F_R & 0 \geq S_R \end{cases}$$

The presence of contact discontinuities, shear waves and material interfaces in a flow under study brings out the shortcomings of the HLL scheme since these waves are associated with the multiples Eigenvalue u . HLL schemes consider only the average across the wave structure as the constant values and therefore the spatial variation of the problem across the star region is not captured. The missing waves are restored by the HLLC scheme which is explained next.

3.3.2 HLLC schemes

This is a modification of the HLL solver where the missing contact and shear waves are restored. As is evident from figure 11, a middle wave of speed S^* is introduced in addition to the slowest and fastest wave speeds S_L and S_R . This middle wave corresponds to the Eigen value u . This middle wave separates two averaged states between the two acoustic waves S_L and S_R .

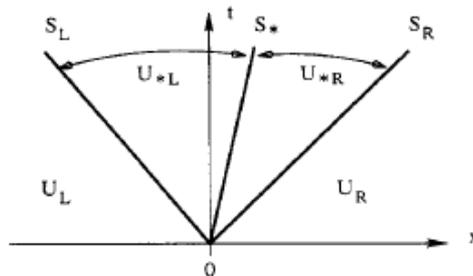


Figure 11: HLLC approximate solution to the Riemann solver (Toro, 1999)

The solution of the HLLC approximate solver can be written as

$$U(x, t) = \begin{cases} U_L & \frac{x}{t} \leq S_L, \\ U_L^* & S_L \leq \frac{x}{t} \leq S_*, \\ U_R^* & S_* \leq \frac{x}{t} \leq S_R, \\ U_R & \frac{x}{t} \geq S_R, \end{cases}$$

Where the solution vectors

$$U_K^* = \rho_K \begin{pmatrix} \frac{S_K - u_K}{S_K - S_*} \\ 1 \\ S_* \\ v_K \\ w_K \\ \frac{E_K}{\rho_K} + (S_* - u_K) \left[S_* + \frac{p_K}{\rho_K(S_K - u_K)} \right] \end{pmatrix}$$

For $K = L$ and $K = R$. The following assumptions are made to arrive at the above solution vector relation,

$$\left. \begin{aligned} u_L^* &= u_R^* = u_*, \\ p_L^* &= p_R^* = p_*, \\ v_L^* &= v_L, \quad v_R^* = v_R, \\ w_L^* &= w_L, \quad w_R^* = w_R, \end{aligned} \right\}$$

And $S^* = u_*$. The exact solution of the Riemann problems justifies the assumptions that are made here. The contact velocity can then be computed by,

$$S^* = \frac{\rho_R u_R (S_R - u_R) - \rho_L u_L (S_L - u_L) + p_L - p_R}{\rho_R (S_R - u_R) - \rho_L (S_L - u_L)}$$

The HLLC flux can then be computed from the following relations,

$$F_{i+\frac{1}{2}}^{hll} = \begin{cases} F_L & 0 < S_L \\ F_L^* = F_L + S_L (U_L^* - U_L) & S_L \leq 0 \leq S_* \\ F_R^* = F_R + S_R (U_R^* - U_R) & S_* \leq 0 \leq S_R \\ F_R & 0 > S_R \end{cases}$$

For both the HLL and HLLC schemes the wave speeds S_L and S_R are calculated from the below formula which preserves the scheme positivity,

$$S_L = \min[u_L - c_L, \tilde{u} - \tilde{c}]$$

$$S_R = \max[u_R + c_R, \tilde{u} + \tilde{c}]$$

3.3.3 Roe solver (with entropy fix)

According to Godunov, for a hyperbolic system of equations, if the Jacobian of the flux vector is constant, the exact value of the flux at any face can be computed by,

$$F_{i+\frac{1}{2}} = \frac{1}{2} (F_L + F_R) - \frac{1}{2} R|\Lambda|L \left(q_{i+\frac{1}{2},R} - q_{i+\frac{1}{2},L} \right)$$

Where L and R are the left and right Eigen vectors and $|\Lambda|$ is the diagonal matrix with the absolute values of the eigenvalues of the Jacobian of the flux vector at the face. The above equation is valid if the Jacobian is constant. However, for the Euler equations, the Jacobian of the flux vector is not a constant, but a function of the conserved variables, ie, $A = A(q)$.

The flux at the face computed using the Roe solver can be represented in a similar form,

$$F_{i+\frac{1}{2}} = \frac{1}{2} (F_L + F_R) - \frac{1}{2} \tilde{R}|\tilde{\Lambda}|\tilde{L} \left(q_{i+\frac{1}{2},R} - q_{i+\frac{1}{2},L} \right)$$

where $\tilde{R}, |\tilde{\Lambda}|, \tilde{L}$ represent the Eigen matrices for the Roe average between the left and right states at the face.

The Roe averaged variables can be expressed as

$$\begin{aligned} \tilde{\rho} &= \frac{\rho_R \sqrt{\rho_R} + \rho_L \sqrt{\rho_L}}{\sqrt{\rho_R} + \sqrt{\rho_L}} \\ \tilde{u} &= \frac{u_R \sqrt{\rho_R} + u_L \sqrt{\rho_L}}{\sqrt{\rho_R} + \sqrt{\rho_L}}, \quad \tilde{v} = \frac{v_R \sqrt{\rho_R} + v_L \sqrt{\rho_L}}{\sqrt{\rho_R} + \sqrt{\rho_L}}, \quad \tilde{w} = \frac{w_R \sqrt{\rho_R} + w_L \sqrt{\rho_L}}{\sqrt{\rho_R} + \sqrt{\rho_L}} \\ \tilde{h} &= \frac{h_R \sqrt{\rho_R} + h_L \sqrt{\rho_L}}{\sqrt{\rho_R} + \sqrt{\rho_L}} \end{aligned}$$

The Roe averaged speed of sound is given by,

$$\tilde{c} = \left\{ (\gamma - 1) \left[\tilde{h} - \frac{1}{2} (\tilde{u}^2 + \tilde{v}^2 + \tilde{w}^2) \right] \right\}^{\frac{1}{2}}$$

The left and right values at each face are the WENO reconstructed point values. The Jacobian for the Flux $F(U)$ for the three dimensional Euler equations can be expressed as (the tilde notation is removed for Roe averaged variables for ease of representation) ,

$$[A] = \frac{\partial F_i}{\partial q_j}$$

$$= \begin{bmatrix} 0 & n_x & n_y & n_z & 0 \\ (\gamma - 1)e_k n_x - uu_n & (\gamma - 1)e_k n_x - uu_n & un_y - (\gamma - 1)vn_x & un_z - (\gamma - 1)wn_x & (\gamma - 1)n_x \\ (\gamma - 1)e_k n_y - vu_n & vn_x - (\gamma - 1)un_y & u_n - (\gamma - 2)vn_y & vn_z - (\gamma - 1)wn_y & (\gamma - 1)n_y \\ (\gamma - 1)e_k n_z - wu_n & wn_x - (\gamma - 1)un_z & wn_y - (\gamma - 1)vn_z & u_n - (\gamma - 2)wn_z & (\gamma - 1)n_z \\ [(\gamma - 1)e_k - h_0]u_n & h_0 n_x - (\gamma - 1)uu_n & h_0 n_y - (\gamma - 1)vu_n & h_0 n_z - (\gamma - 1)wu_n & \gamma u_n \end{bmatrix}$$

Where $u_n = \vec{v} \cdot \hat{n} = un_x + vn_y + wn_z$

$$n_x^2 + n_y^2 + n_z^2 = 1$$

The stagnation energy and enthalpy per unit mass are,

$$e_0 = e + e_k, \quad h_0 = h + e_k,$$

$$e_k = \frac{1}{2}(u^2 + v^2 + w^2), \text{ the kinetic energy per unit mass}$$

One of the limitations of the Roe flux is that the stationary expansion shocks are not dissipated properly by this scheme. Hence a method to fix the entropy in the expansion shocks is to be added to the Roe solver. This is done by replacing the Roe averaged eigenvalues with,

$$|\tilde{\Lambda}_i| = \begin{cases} |\tilde{\lambda}_i|, & \text{if } |\tilde{\lambda}_i| \geq 2\varepsilon\tilde{c} \\ \tilde{\lambda}_i^2/4\varepsilon\tilde{c} + \varepsilon\tilde{c}, & \text{if } |\tilde{\lambda}_i| < 2\varepsilon\tilde{c} \end{cases}$$

Where $\varepsilon = 0.1$ and λ represents Eigen values of the Roe averaged matrix.

3.3.4 Lax-Friedrichs flux:

For a given Riemann problem, a positive speed S^+ can be identified. For the HLL solver, if we choose $S_L = -S^+$ and $S_R = +S^+$, we obtain the formulation for the Rusanov solver.

$$F_{i+\frac{1}{2}} = \frac{1}{2}(F_L + F_R) - \frac{1}{2}S^+(U_R - U_L)$$

There are different choices to the speed S^+ . For the Rusanov flux, the speed is given by

$$S^+ = \max\{|u_L - c_L|, |u_R - c_R|, |u_L + c_L|, |u_R + c_R|\}$$

The form of Lax-Friedrichs numerical flux used in this work is a special form of Rusanov flux in which the choice of speed S^+ is,

$$S^+ = \max_U \{F'(U)\}$$

For the Euler equations, this is the maximum of the absolute value of the Eigen values of the Roe matrix at the face, ie, $|\tilde{u} + \tilde{c}|$. For the advection equation S^+ takes the value of the advection speed in the corresponding coordinate direction.

3.4 Time discretization

The high order of the spatial discretization scheme dictates the need for a high order of accuracy in temporal discretization also. A fourth order Runge-Kutta time discretization, which ensures uniformly high order of time accuracy, is used to forward the solution in time. It is implemented in four stages as shown below,

$$U^{(1)} = U^{(n)} + \frac{\Delta t}{2} R(U^{(n)})$$

$$U^{(2)} = U^{(n)} + \frac{\Delta t}{2} R(U^{(1)})$$

$$U^{(3)} = U^{(n)} + \Delta t R(U^{(2)})$$

$$U^{(n+1)} = U^{(n)} + \frac{\Delta t}{6} R(U^{(n)}) + \frac{1}{3} \Delta t R(U^{(1)}) + \frac{1}{3} \Delta t R(U^{(2)}) + \frac{1}{6} \Delta t R(U^{(3)})$$

Where R is the spatial operator which represents the RHS of equation at a particular time level.

The use of such an explicit scheme necessitates the calculation of a time step Δt which ensures the stability of the scheme. In this work, Δt is chosen as,

$$\Delta t = \text{CFL} \cdot \min \left(\frac{\Delta x}{|S^{n,x}|}, \frac{\Delta y}{|S^{n,y}|}, \frac{\Delta z}{|S^{n,z}|} \right)$$

Where $S^{t,m}$ is the speed of the fastest wave at a particular time level t travelling along a particular coordinate direction $m = x, y, z$. The fastest wave is taken as the greatest wave speed found in the Riemann solutions. CFL is the constant which satisfies the Courant-Friedrichs-Lewy condition for the linear stability of the scheme.

A convergence study of the error in time discretization is conducted for the propagation of a smooth wave over one time period by solving a linear advection equation. A grid of 80 points is used. A reference solution is obtained by simulating the solution over one time period using a fifth order WENO scheme and a second order MUSCL scheme for a very small time step. The results are tabulated in Tables 3 and 4. Expected order of accuracy of four was obtained.

Time step (dt)	Error (L₁)	Order (L₁)
3.90×10^{-4}	4.88×10^{-7}	
1.95×10^{-4}	3.05×10^{-8}	3.99
9.77×10^{-5}	1.91×10^{-9}	3.99
4.88×10^{-5}	1.19×10^{-10}	4.00
2.44×10^{-5}	7.42×10^{-12}	4.00

Table 3: Convergence study for RK-4 time discretization (MUSCL scheme)

Time step (dt)	Error (L₁)	Order (L₁)
3.90×10^{-4}	3.11×10^{-5}	
1.95×10^{-4}	1.69×10^{-6}	4.19
9.77×10^{-5}	1.01×10^{-7}	4.07
4.88×10^{-5}	6.23×10^{-9}	4.02
2.44×10^{-5}	3.88×10^{-10}	4.00

Table 4: Convergence study for RK-4 time discretization (WENO scheme)

3.5 Boundary conditions

In this work, the implementation of boundary conditions is done through the concept of ghost cells which are an extra layer of cells surrounding the boundary. The cell-centred ghosts are assigned values such that the desired behaviour is obtained at the boundaries. For the fifth order WENO schemes, three sets of ghost layers are necessary to obtain fifth order accurate reconstructed values at the boundaries.

3.5.1 Periodic Boundary conditions

Periodic Boundary Conditions can be considered to fall under the category of free boundaries. In effect, the periodic boundary conditions eliminates surfaces and is the most popular choice of boundary conditions. They are mostly used when the flow solution and the physical geometry over which the flow is simulated both have a periodically repeating nature.

At a periodic boundary the flow is treated as if the opposing periodic boundary is a direct neighbour to the cells adjacent to the first periodic boundary. Thus when the flow variables at a particular cell adjacent to a periodic boundary are being calculated, values of the flow variables from the opposite periodic boundary are used. In the cell-centred finite volume framework used in this work, this is implemented in the following way,

$$U(\text{ng}) = U(N)$$

$$U(\text{ng} - 1) = U(N - 1)$$

$$U(\text{ng} - 2) = U(N - 2)$$

$$U(N + 1) = U(\text{ng})$$

$$U(N + 2) = U(\text{ng} + 1)$$

$$U(N + \text{ng}) = U(\text{ng} + 2)$$

Where U is the conserved variable, N is the number of grid points along a coordinate direction and $\text{ng} = 3$ is the number of ghost layers in the same coordinate direction. The equation $U(\text{ng}) = U(N)$, implies that the ng -th term of the conserved variable array represented by U is assigned the N -th value of the same array.

3.5.2 Fixed Boundary condition (Dirichlet)

This is the simplest boundary condition available in the solver. The ghost cells are simply set to a constant value that the solution needs to take at the boundary of the computational domain.

$$U(0:\text{ng}) = c$$

Where c is the desired constant value of the solution at the boundary. The equation implies that the conserved variable U is assigned the value c at all locations from 0 to ng , including 0, the first array location, and excluding the ng -th location.

3.5.3 Solid Boundaries

Solid boundaries for inviscid flow are characterized by the no-penetration condition, or the commonly called reflecting or slip wall boundary condition. For a stationary wall or any solid boundary to be reflecting,

$$\vec{U} \cdot n_1 = 0$$

Where U is the velocity vector and n_0, n_1, n_2 are the three components of the outward unit normal at the wall or boundary. In the solver, the reflecting boundary condition is implemented by reversing the direction of the normal component of the velocity at the wall interior cells and assigning these values to the velocity variables in the ghost cells. The density variable is the same as the interior values. The energy variable is calculated using the new ghost values of velocity, pressure and density.

3.5.4 Far-Field Boundaries

Open, artificial, absorbing or remote boundaries are a few of the boundaries that can be included under the category of far-field boundaries. They are mostly inflow or outflow boundaries which absorb existing waves and emit new ones. If there are any waves present that enter the far-field boundaries, the true domain of dependence is either partially or fully outside the computational domain. This leads to a violation of the CFL condition as the numerical domain of dependence should always be contained in the computational domain for stability. Hence the numerical scheme should be designed to identify the entry of waves through the far-field boundaries. Hence for simulations with inflow and outflow boundaries, a specific set of boundary conditions are used which should specify incoming waves and prevent reflection of outgoing waves.

The design of far-field boundary conditions is based on characteristic theory. The characteristic nature of the flow at the boundaries provides the critical information required for specifying the boundary conditions. The Eigen values of the flow equations at the boundaries indicate the speed of the acoustic waves and specify whether they are entering or leaving the flow domain at the boundary. This information is used to determine how much information is to be taken from inside the domain, ie, the numerical domain and how much from outside the domain or the physical domain.

The five Eigen values for the Euler equations are given by

$$\lambda_1 = u - a, \lambda_2 = u, \lambda_3 = u, \lambda_4 = u, \lambda_5 = u + a$$

3.5.5 Supersonic Inflow/Outflow Boundary Condition:

For a supersonic inflow or outflow, all Eigen values are positive. Hence all the flow information should originate outside or inside the domain respectively. The implementation is thus simple.

For a supersonic outflow, the conserved variables at the external ghost cells are assigned using the values from the internal cells adjacent to the boundary, ie,

$$\rho(N:N + ng) = \rho(N - 1)$$

$$u(N:N + ng) = u(N - 1)$$

$$v(N:N + ng) = v(N - 1)$$

$$w(N:N + ng) = w(N - 1)$$

$$P(N:N + ng) = P(N - 1)$$

For a supersonic inflow,

$$\rho(0:ng) = \rho(ng)$$

$$u(0:ng) = u(ng)$$

$$v(0:ng) = v(ng)$$

$$w(0:ng) = w(ng)$$

$$P(0:ng) = P(ng)$$

Where ρ is the density, P is the pressure, $u, v, and w$ represent the flow velocity in the three co-ordinate directions x, y, z

4

Numerical study of WENO scheme

4.1 Linear Advection

In this section, we solve the linear advection equation

$$\frac{\partial q}{\partial t} + a \frac{\partial q}{\partial x} + b \frac{\partial q}{\partial y} + c \frac{\partial q}{\partial z} = 0$$

(42)

Where a , b and c represent the constant advection speeds along the x , y and z directions respectively and q is the advected quantity.

The linear advection equation can be treated as the prototype for the development of numerical methods for systems of hyperbolic equations. The linear advection equation is studied here as the first and simple step in validating the implementation of the WENO schemes in the solver. A smooth solution governed by a linear advection equation is a very good test case to check the order of accuracy of the numerical scheme by running a convergence test of the errors of the solution in comparison to an exact solution. A simple test case, ie, the propagation of a two-dimensional sine wave along both the x and y coordinate directions are studied. The simple sine wave is chosen since the solution is infinitely smooth at all points. The least expensive and least complex Lax-Friedrichs Riemann solver was used. Other Riemann solvers considered in this work are not suitable for scalar advection. The scheme is run with both a two-point and single point Gaussian quadrature formula.

The order of accuracy is checked by studying the error of the solution against an exact solution after the wave has propagated over one time period. Figures 12 to 17 show the sine wave over one time period with an advection velocity $a = 1.0$ along the x -coordinate. Figures 18 to 23 simulates the propagation of the sine wave with advection velocities $a = 1.0, b = 1.0$ along both the x and y coordinates. Again the figures show that the scheme simulates the propagation of the wave across one time period with very good resolution.

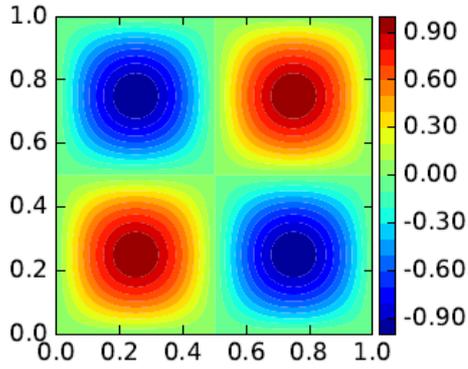


Figure 12: Time $t = 0.0s$

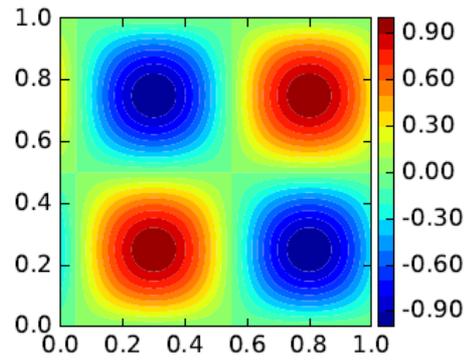


Figure 13: Time $t = 0.05s$

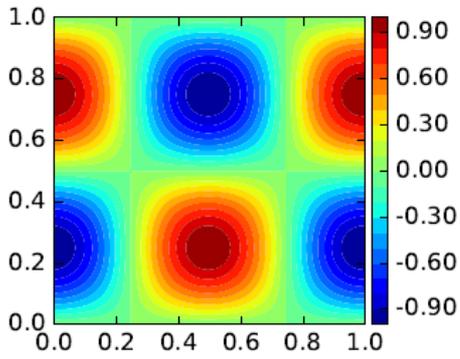


Figure 14: Time $t = 0.25s$

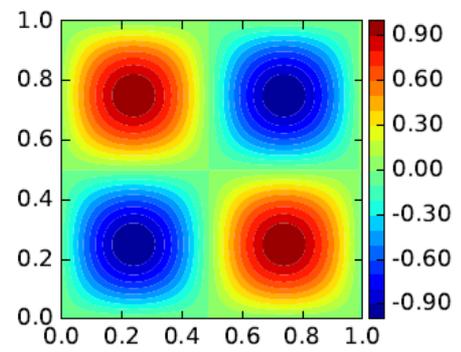


Figure 15: Time $t = 0.5s$

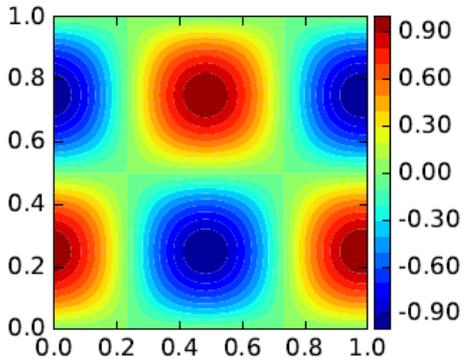


Figure 16: Time $t = 0.75s$

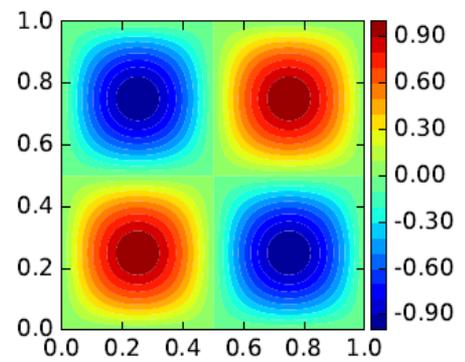


Figure 17: Time $t = 1s$

N	Error (L₁)	Order (L₁)	Error (L₂)	Order (L₂)
10	7.70×10^{-2}		8.78×10^{-2}	
20	5.15×10^{-3}	3.90	5.75×10^{-3}	3.93
25	2.18×10^{-3}	3.85	2.42×10^{-3}	3.87
30	1.09×10^{-3}	3.73	1.22×10^{-3}	3.73
40	3.88×10^{-4}	3.60	4.30×10^{-4}	3.62

Table 5: Convergence study for Case 2 with advection along x axis (with two quadrature points)

N	Error (L ₁)	Order (L ₁)	Error (L ₂)	Order (L ₂)
10	3.87×10^{-2}		4.39×10^{-2}	
20	2.55×10^{-3}	3.92	2.85×10^{-3}	3.94
25	1.05×10^{-3}	3.98	1.17×10^{-3}	3.99
30	5.09×10^{-4}	3.98	5.67×10^{-4}	3.96
40	1.62×10^{-4}	3.99	1.80×10^{-4}	3.99

Table 6: Convergence study for Case 2 with advection along x axis (with one quadrature point)

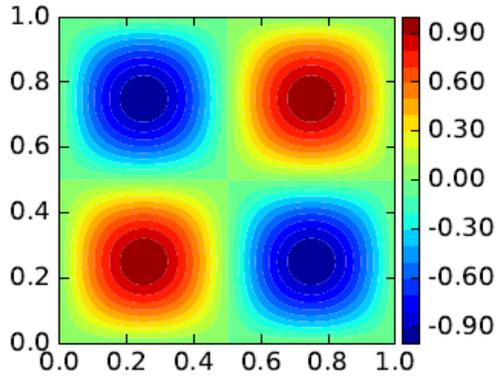


Figure 18: Time $t = 0s$

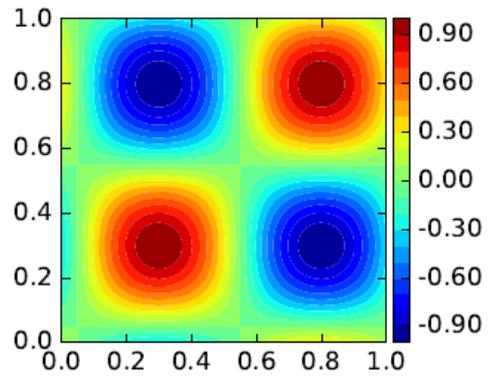


Figure 19: Time $t = 0.05s$

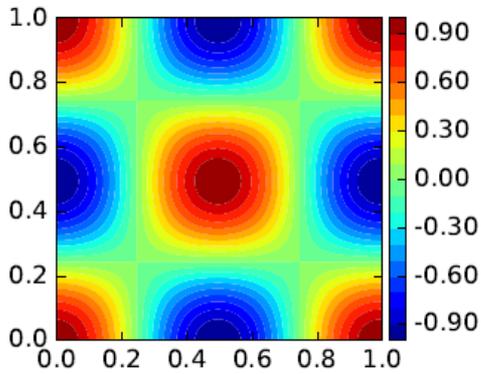


Figure 20: Time $t = 0.25s$

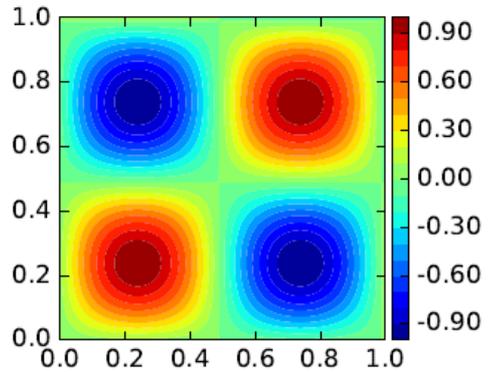


Figure 21: Time $t = 0.5s$

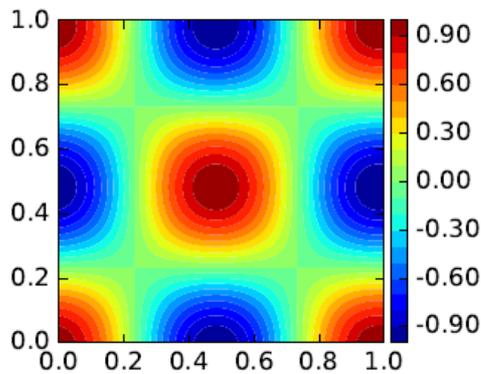


Figure 22: Time $t = 0.75s$

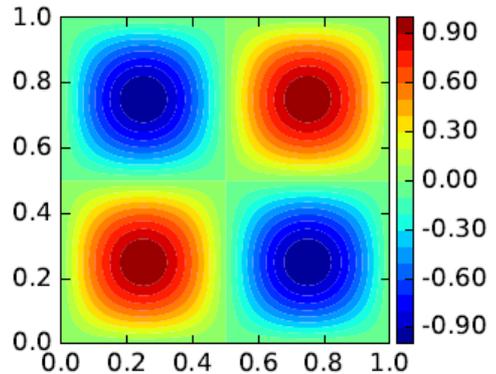


Figure 23: Time $t = 1s$

N	Error (L₁)	Order (L₁)	Error (L₂)	Order (L₂)
10	7.70 x 10 ⁻²		8.78 x 10 ⁻²	
20	5.14 x 10 ⁻³	3.91	5.75 x 10 ⁻³	3.93
25	2.17 x 10 ⁻³	3.86	2.42 x 10 ⁻³	3.87
30	1.09 x 10 ⁻⁴	3.76	1.22 x 10 ⁻³	3.76
40	3.88 x 10 ⁻⁴	3.60	4.30 x 10 ⁻⁴	3.61

Table 7: Convergence study for Case 2 with advection along both x and y coordinate axes (two quadrature point)

N	Error (L₁)	Order (L₁)	Error (L₂)	Order (L₂)
10	3.87 x 10 ⁻²		4.39 x 10 ⁻²	
20	2.55 x 10 ⁻³	3.92	2.85 x 10 ⁻³	3.94
25	1.05 x 10 ⁻³	3.98	1.17 x 10 ⁻³	3.99
30	5.09 x 10 ⁻⁴	3.98	5.67 x 10 ⁻⁴	3.96
40	1.62 x 10 ⁻⁴	3.99	1.80 x 10 ⁻⁴	3.99

Table 8: Convergence study for Case 2 with advection along both x and y coordinate axes (one quadrature point)

The most striking feature of the convergence studies illustrated in Tables 7-10 is that the error of the solution is almost similar while using one and two quadrature points for calculating the numerical flux. This results in the overall accuracy of the scheme being independent of the number of quadrature points used for numerical flux approximation. The expected order of accuracy for a two point Gaussian quadrature formula is four (Titarev & Toro, 2004). The overall order of accuracy is close to four for all cases.

The purpose of this test case was to study the convergence results and order of accuracy and hence demonstrate the effect of the Gaussian quadrature points on the scheme. Since the errors are very similar, both the one-point and two-point Gaussian quadrature formula gives similar results for the calculation of numerical flux.

4.2 The Doswell Frontogenesis problem

The Doswell Frontogenesis test problem is a standard test in atmospheric modelling. From a numerical point of view, this problem allows the assessment of the performance of the scheme in the treatment of sharp fronts and moving discontinuities in two space dimensions. A number of numerical schemes, especially using dimensional splitting have failed for this test problem (Titarev & Toro, 2004). The flow is governed by the two dimensional advection equation. A velocity field represented by a and b (Figures 24 and 25) is imposed on the solution field and solution is advected to a time of 4 seconds.

The domain is set at $[-5, 5]$. The flow is represented by,

$$\begin{aligned}
 a &= -y\omega(r), & b &= x\omega(r), \\
 \omega(r) &= \frac{1}{r}v(r), & v(r) &= \bar{v} \operatorname{sech}^2(r) \tanh(r), \\
 r &= \sqrt{x^2 + y^2}, & \bar{v} &= 2.59807,
 \end{aligned}$$

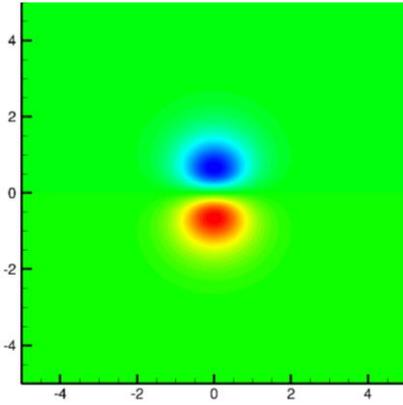


Figure 24: Advection velocity a

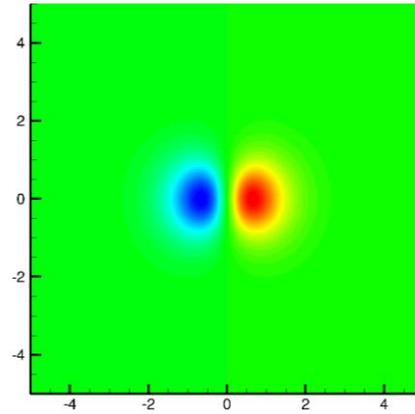


Figure 25: Advection velocity b

The initial conditions are defined by,

$$Q(x, y, 0) = \tanh\left(\frac{y}{\delta}\right),$$

The value of δ defines the thickness of the front zone. A value of $\delta = 1$ generates a smooth solution and a value of $\delta = 10^{-6}$ gives sharp fronts. The initial conditions on a grid 201×201 is shown in figures 26 and 27 below which clearly illustrates the sharp front and the smooth structure for two different values of δ .

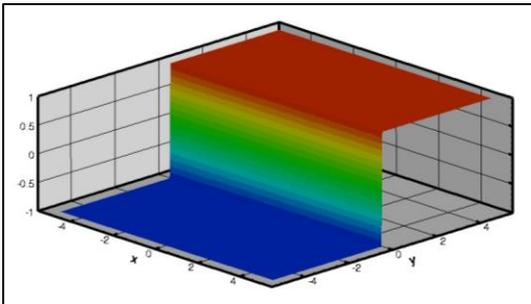


Figure 26: Initial conditions for $\delta = 10^{-6}$

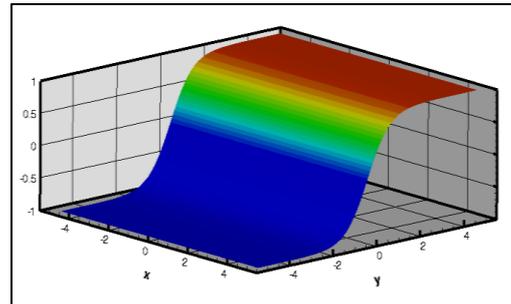


Figure 27: Initial conditions for $\delta = 1$

The numerical solution at time $t = 4$ using a mesh of 401×401 is studied for both smooth and sharp solutions. The fourth order Runge-Kutta method is used for time integration. A CFL number of 0.45 is used. Again the Lax-Friedrichs Riemann solver was used. The 3-d solution for the smooth front is shown in figure 28. Figure 29 shows the solution along a plane perpendicular to the Z-axis. The figures show a smooth and well resolved solution with the velocity field applied along the center. Again, convergence studies of the errors for this case is done and tabulated in Tables 11 and 12.

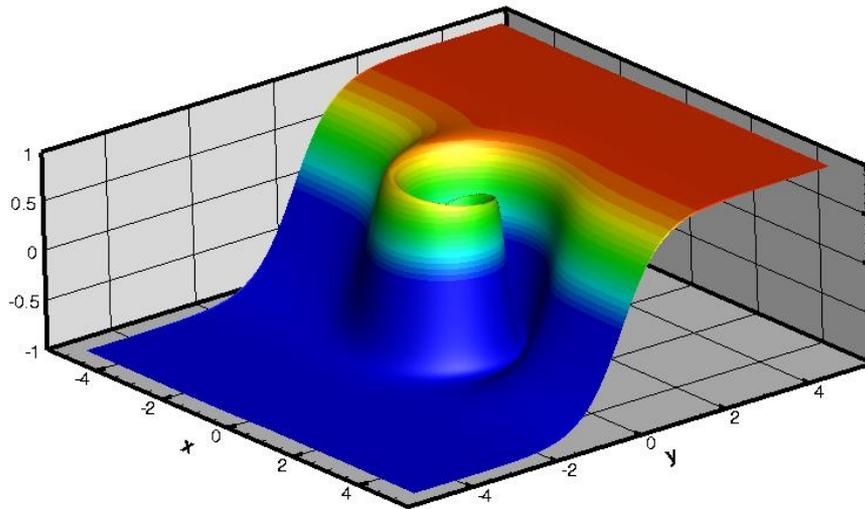


Figure 28: Solution at time $t = 4$ on a grid 400×400 and $\delta = 1$

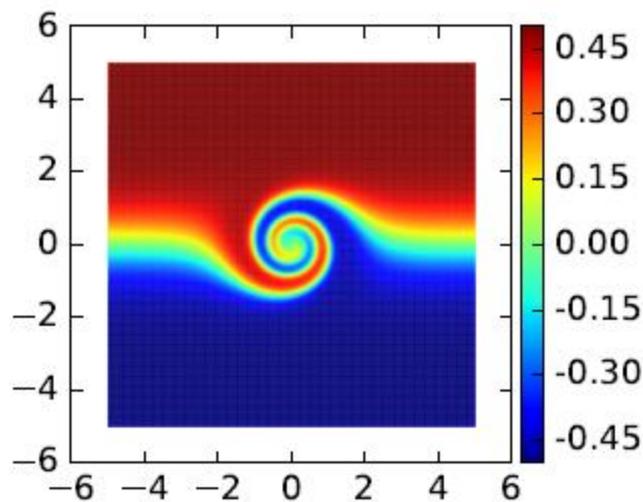


Figure 29: One dimensional cut along plane normal to Z -axis for $\delta = 1$

The convergence study again reveals that there is very small difference in the errors while using one and two point Gaussian quadrature formulas. The two point Gaussian quadrature formula is expected to give fourth order of accuracy. The order of accuracy of the scheme as shown in Table 12 for two quadrature points matches this. And also, there is no drop in order of accuracy when a single quadrature point is used.

N	Error (L₁)	Order (L₁)	Error (L₂)	Order (L₂)
30	1.66×10^{-1}		2.10×10^{-1}	
40	6.11×10^{-2}	3.58	1.01×10^{-1}	2.62
50	2.44×10^{-2}	4.21	5.20×10^{-2}	3.04
60	1.11×10^{-2}	4.40	2.87×10^{-2}	3.31
70	5.39×10^{-3}	4.75	1.58×10^{-2}	3.92
80	2.96×10^{-3}	4.55	1.01×10^{-2}	3.41
90	1.76×10^{-3}	4.48	6.75×10^{-3}	3.46

Table 9: Convergence study for smooth solution (one quadrature point)

N	Error (L₁)	Order (L₁)	Error (L₂)	Order (L₂)
30	1.66×10^{-1}		2.10×10^{-1}	
40	6.11×10^{-2}	3.58	1.01×10^{-1}	2.62
50	2.45×10^{-2}	4.19	5.24×10^{-2}	3.02
60	1.12×10^{-2}	4.39	2.91×10^{-2}	3.28
70	5.48×10^{-3}	4.69	1.62×10^{-2}	3.87
80	3.02×10^{-3}	4.52	1.04×10^{-2}	3.36
90	1.80×10^{-3}	4.44	7.00×10^{-3}	3.39

Table 10: Convergence study for smooth solutions (two quadrature points)

Figures 30 and 31 illustrate the solutions for $\delta = 10^{-6}$ which creates a sharp front. High quality results are obtained with the sharp front captured and the velocity field imposed at the center. No spurious oscillations are visible in the figures even though the solution front is considerably sharp.

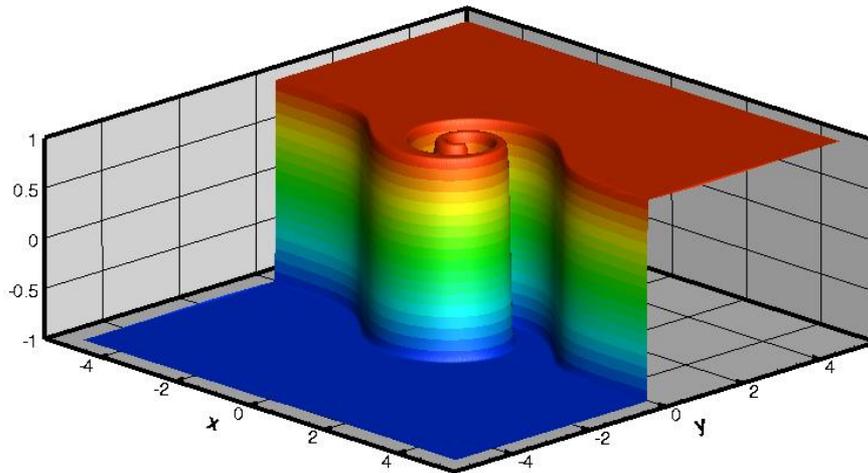


Figure 30: Solution at time $t = 4$ on a grid 400×400 and $\delta = 10^{-6}$

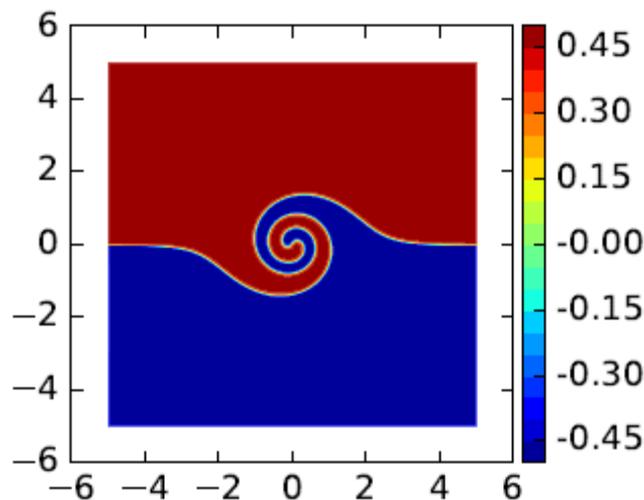


Figure 31: One dimensional cut along plane normal to Z-axis for $\delta = 10^{-6}$

Comparison of the results for the sharp front is done to see how well the front or discontinuity has been captured by the scheme. This is done by taking a one dimensional section along the y axis. The exact solution at a time t is given by,

$$Q(x, y, t) = \tanh\left(\frac{y\cos(vt) - x\sin(vt)}{\delta}\right)$$

The exact solution represents a rotation of the initial discontinuous distribution around the origin with variable angular velocity $\omega(r)$. It should be mentioned that as time evolves, scales will eventually be developed in the solution which is beyond the resolution of the mesh being considered (Titarev & Toro, 2004).

Figure 32 shows the comparison with the exact solution. Quite sharp resolution of all the discontinuities is seen. The solution is also monotone.

The results and the convergence study can be considered to be very satisfactory and validates the implementation of the WENO schemes for the advection equations, with the shock capturing performing exceptionally well.

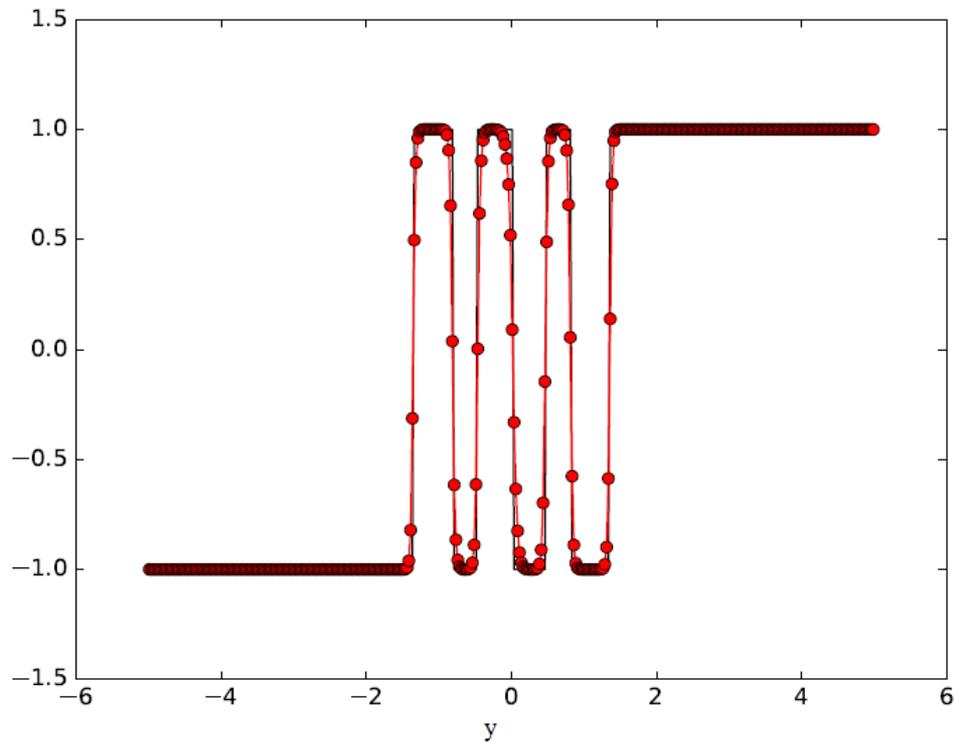


Figure 32: One dimensional cut along y axis at output time $t = 4$. Solid line corresponds to exact solution.

4.3 Sod's shock tube problem

The Euler equations considers only the effects of macroscopic convection and the effects of collision of molecules through pressure forces. These equations have several important mathematical properties. With regards to the development of numerical schemes, the most important property is that the system of equations can be written in the characteristic form, which allows the projection of equations in any spatial direction giving rise to a system of coupled wave like equations (Pirozzoli, 2011).

The Euler equations for compressible flow in three dimensions are

$$\begin{aligned} \frac{\partial \rho}{\partial t} + \frac{\partial(\rho u)}{\partial x} + \frac{\partial(\rho v)}{\partial y} + \frac{\partial(\rho w)}{\partial z} &= 0 \\ \frac{\partial \rho u}{\partial t} + \frac{\partial(\rho u \cdot u + p)}{\partial x} + \frac{\partial(\rho uv)}{\partial y} + \frac{\partial(\rho uw)}{\partial z} &= 0 \\ \frac{\partial \rho v}{\partial t} + \frac{\partial(\rho v \cdot u)}{\partial x} + \frac{\partial(\rho v \cdot v + P)}{\partial y} + \frac{\partial(\rho vw)}{\partial z} &= 0 \\ \frac{\partial \rho w}{\partial t} + \frac{\partial(\rho w \cdot u)}{\partial x} + \frac{\partial(\rho wv)}{\partial y} + \frac{\partial(\rho w \cdot w + P)}{\partial z} &= 0 \\ \frac{\partial E}{\partial t} + \frac{\partial(u(E + P))}{\partial x} + \frac{\partial(v(E + P))}{\partial y} + \frac{\partial(w(E + P))}{\partial z} &= 0 \end{aligned}$$

The equation of state is used to close the system,

$$P = (\gamma - 1) \left(E - \rho \frac{u^2 + v^2 + w^2}{2} \right) \quad (43)$$

Where ρ is the density, P is the pressure, $u, v, and w$ represent the flow velocity in the three co-ordinate directions x, y, z , E is the total Energy per unit volume and $\gamma = 1.4$ is the ratio of specific heat for an ideal gas.

The Riemann problems are one of the few test cases with an analytical solution for the Euler equations. One such Riemann problem is the Sod's shock tube problem. G.A. Sod introduced a simple one dimensional model of a gas in 1978 which allowed a study of the ability of numerical schemes to simulate flow in the presence of shocks. The availability of an exact solution makes this an attractive test case as comparisons can be made to study the effect of different Riemann solvers on the solution. Some results of the Sod's shock tube problem using first and second order numerical schemes have been given in the Appendix.

Sod considered a one dimensional tube of unit length and the following initial conditions at time $t = 0$.

$$\phi(x, 0) = \begin{cases} \phi_L & x < 0 \\ \phi_R & x \geq 0 \end{cases}$$

where

$$\phi_L = \begin{bmatrix} \rho_L \\ u_L \\ p_L \end{bmatrix} = \begin{bmatrix} 1 \text{ kg/m}^3 \\ 0 \text{ m/s} \\ 100,000 \text{ N/m}^2 \end{bmatrix}, \quad \phi_R = \begin{bmatrix} \rho_R \\ u_R \\ p_R \end{bmatrix} = \begin{bmatrix} 0.125 \text{ kg/m}^3 \\ 0 \text{ m/s} \\ 10,000 \text{ N/m}^2 \end{bmatrix}.$$

This initial condition can be achieved physically by having a diaphragm in the middle of the tube. The gas to the left and right sides of the diaphragm is initially at rest. The bursting of the diaphragm at time $t = 0$ causes a one-dimensional flow with two types of singularities, namely one contact discontinuity and a steadily propagating shock wave, through the gas in the tube. The solution can be considered to be self-similar.

The solution is made challenging by the fact that the shock and contact are close together. Even at the final time, when their separation is greatest, the shock and the contact are separated only by 2.5 m, which corresponds to only about five grid cells. It is almost impossible for most numerical methods to capture both a shock and a contact in five grid cells or less, unless they incorporate special features like sub-cell resolution or artificial compression (Laney, 1998). These factors make this test case a very attractive and interesting test case for the WENO schemes.

A computational domain [-10m, 10m] divided into 101 control volumes is used. The Initial CFL number is 0.5. The solution is plotted at a time of $t = 0.01s$. The results are studied for four different Riemann solvers, Lax-Friedrichs scheme, HLL scheme, HLLC scheme and the Roe scheme by comparison with the exact solution obtained by solving the initial conditions using gas dynamics. The density, velocity and pressure is plotted for all four Riemann solvers in figures 33-44. The differences are not obvious for the first three schemes, but the Roe flux offers a much better result especially with respect to shock capturing. The dissipation is much lower for the Roe flux as a result of which the shocks are smeared less. The Lax-Friedrichs solver smears the solution most, followed by the HLL solver. The HLLC solver offers appreciably better results, even though the quality falls well short of that offered by the Roe flux. However, it is important to realize that the solver provides high quality results on the basis of comparison with the exact solution for all four Riemann solvers which is good validation of the scheme for the one dimensional Euler equations.

This is the first test case in which we test the part of the solver which performs transformation from conserved to characteristic space, consequent WENO reconstruction, and the transformation back to conserved space. Even though this is complicated and slightly expensive, the good agreement of the results with respect to the exact solution has vindicated this procedure. Furthermore, this test problem is a tough test case for non-characteristic-based schemes of order at least three and oscillations can easily appear for such schemes (Jiang & Shu, 1996).

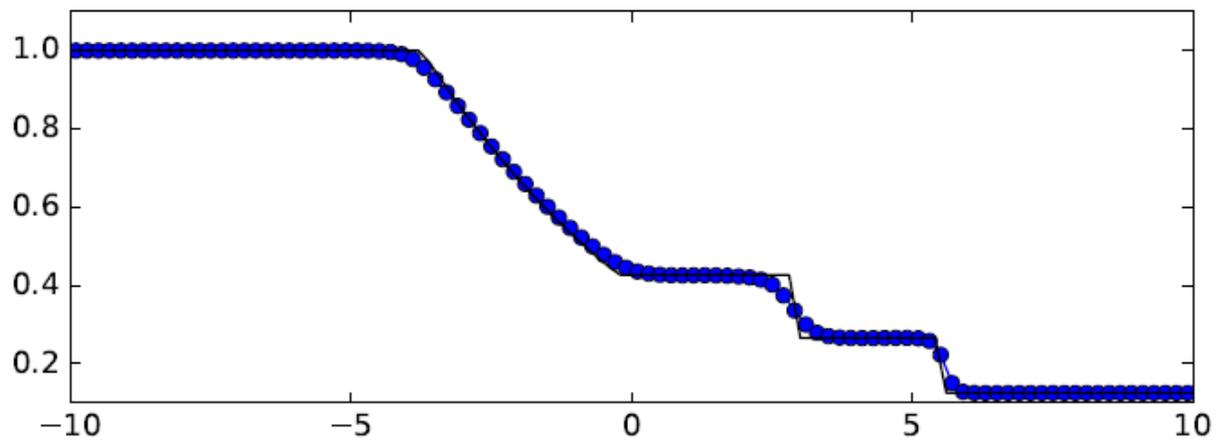


Figure 33 Density distribution at $t=0.01$ (Lax-Friedrichs flux)

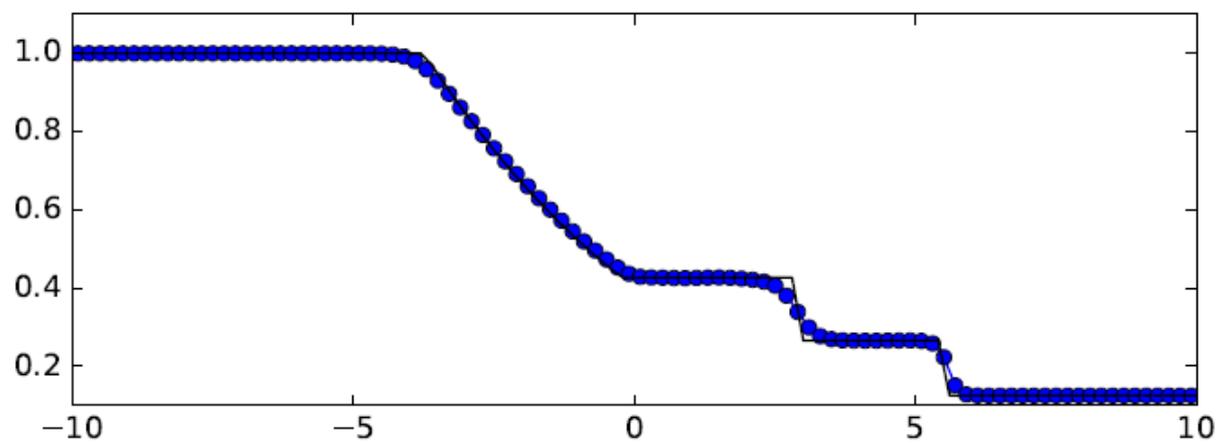


Figure 34: Density distribution (HLL flux)

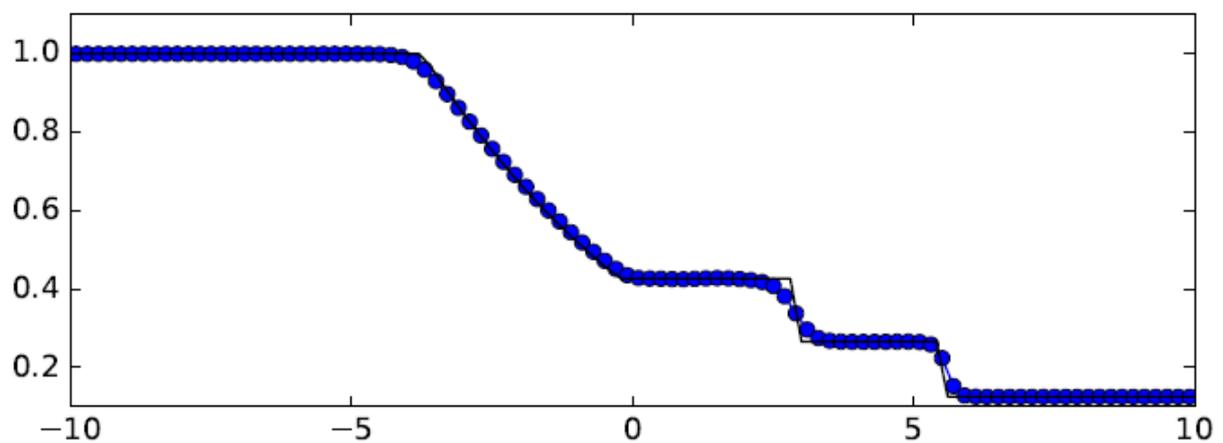


Figure 35: Density distribution (HLLC flux)

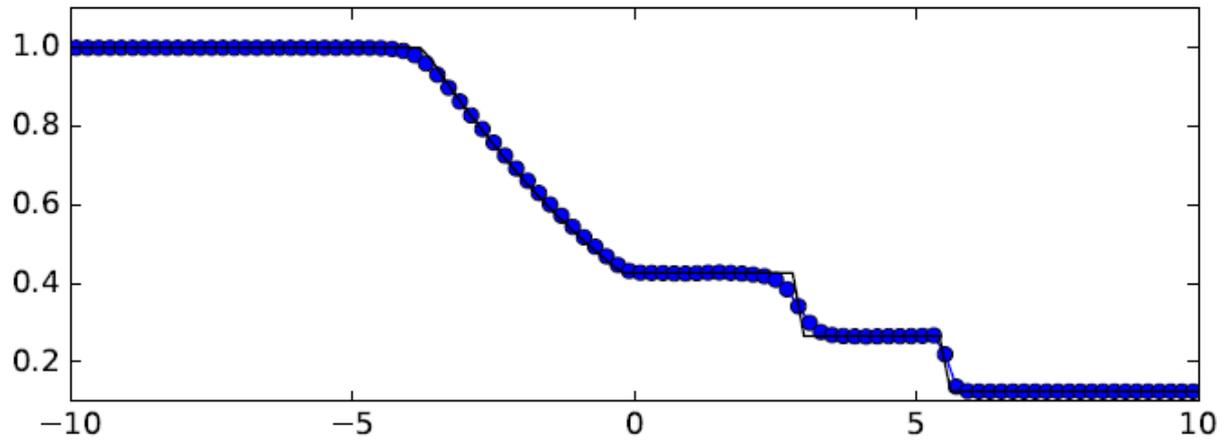


Figure 36: Density distribution (Roe flux with entropy fix)

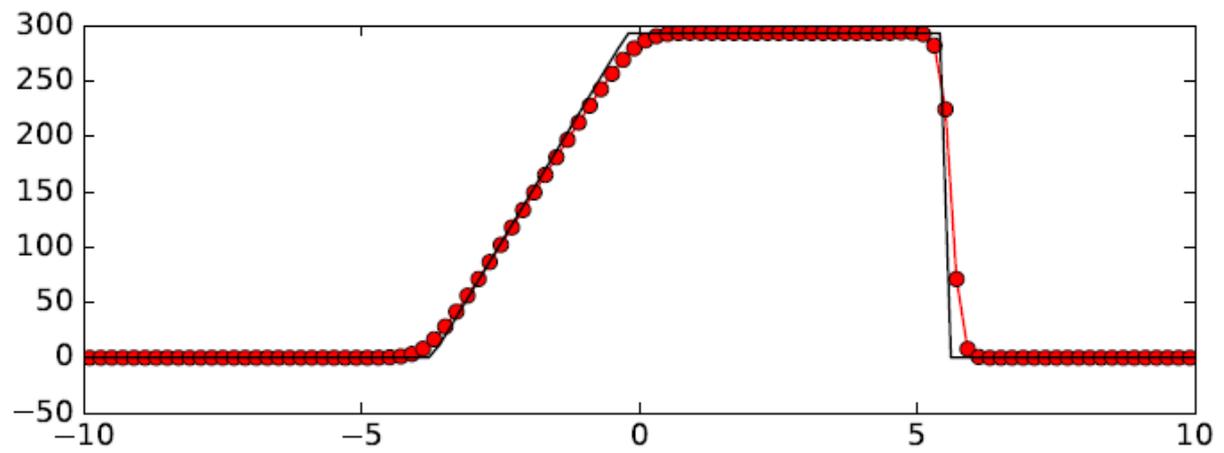


Figure 37: Velocity distribution (Lax-Friedrichs Flux)

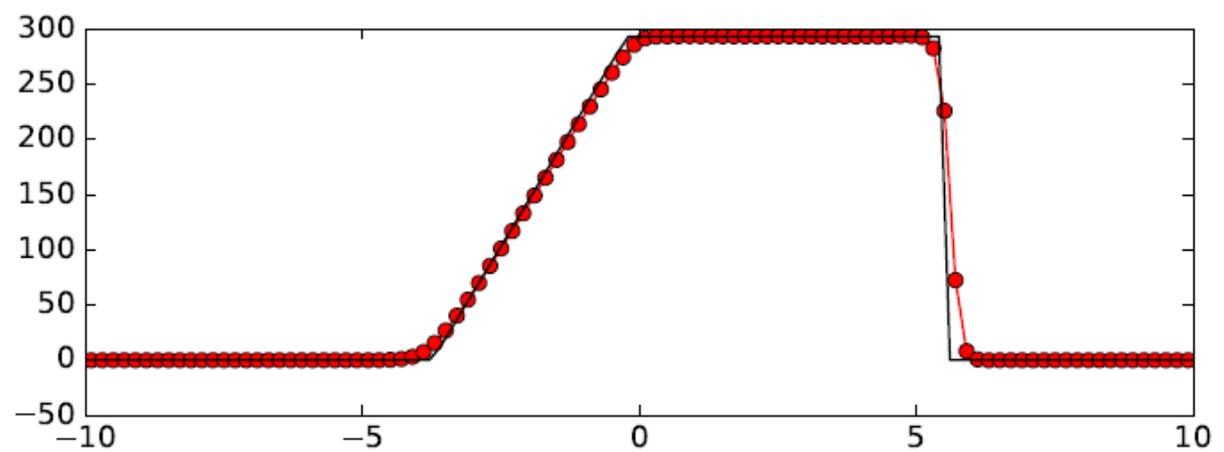


Figure 38: Velocity distribution (HLL Flux)

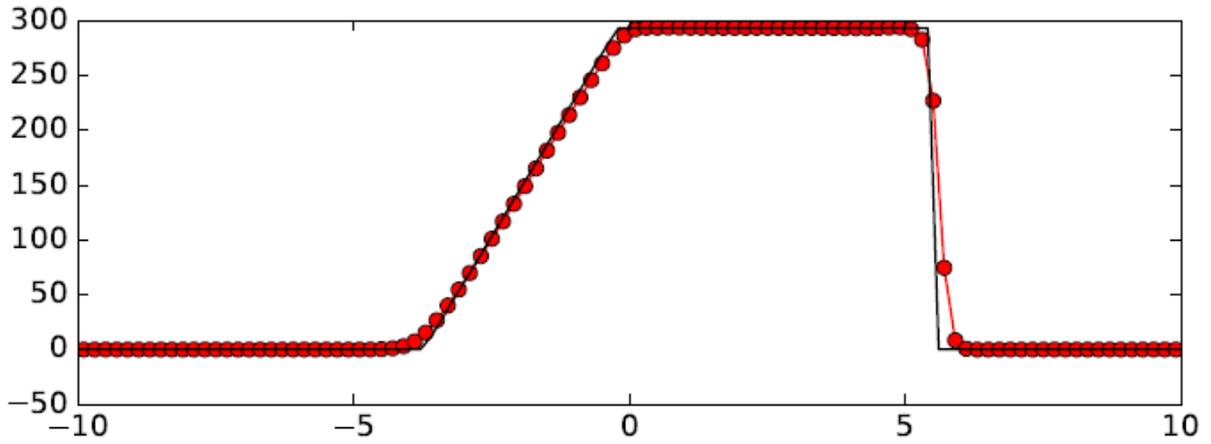


Figure 39: Velocity distribution (HLLC Flux)

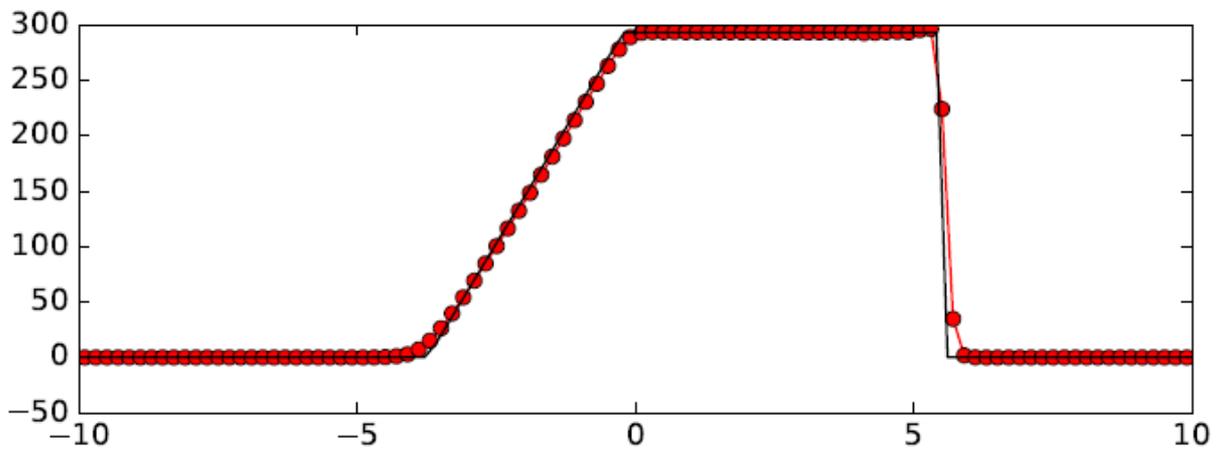


Figure 40: Velocity distribution at $t = 0.01s$ (Roe Flux)

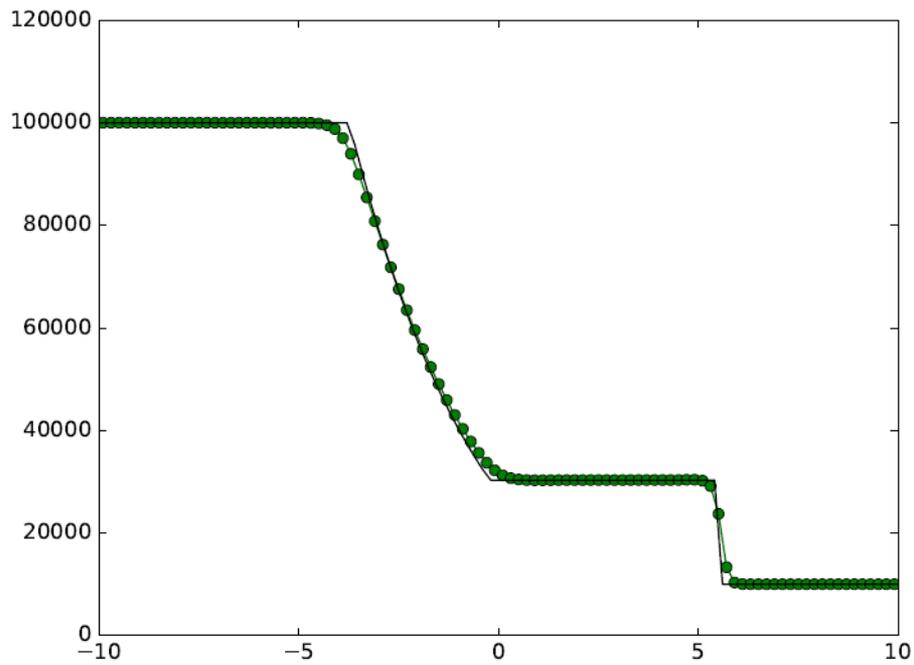


Figure 41: Pressure distribution (Lax-Friedrichs flux)

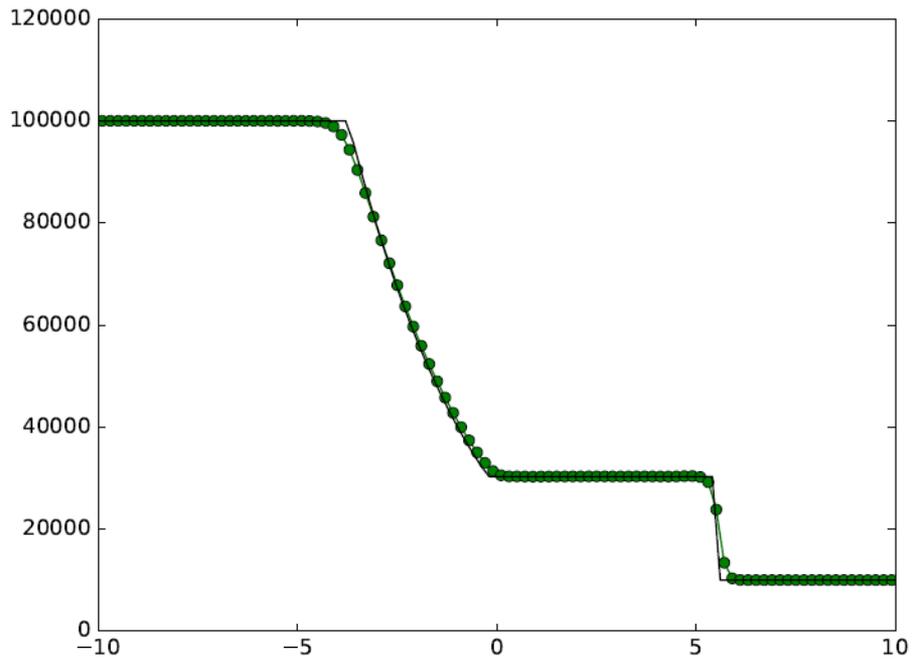


Figure 42: Pressure distribution HLL flux

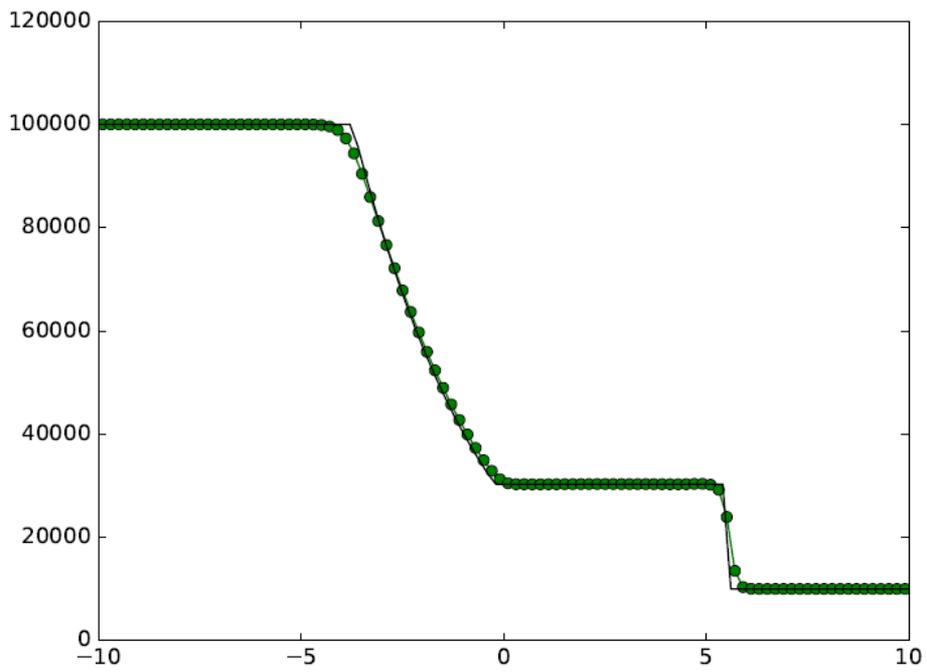


Figure 43: Pressure distribution (HLLC Flux)

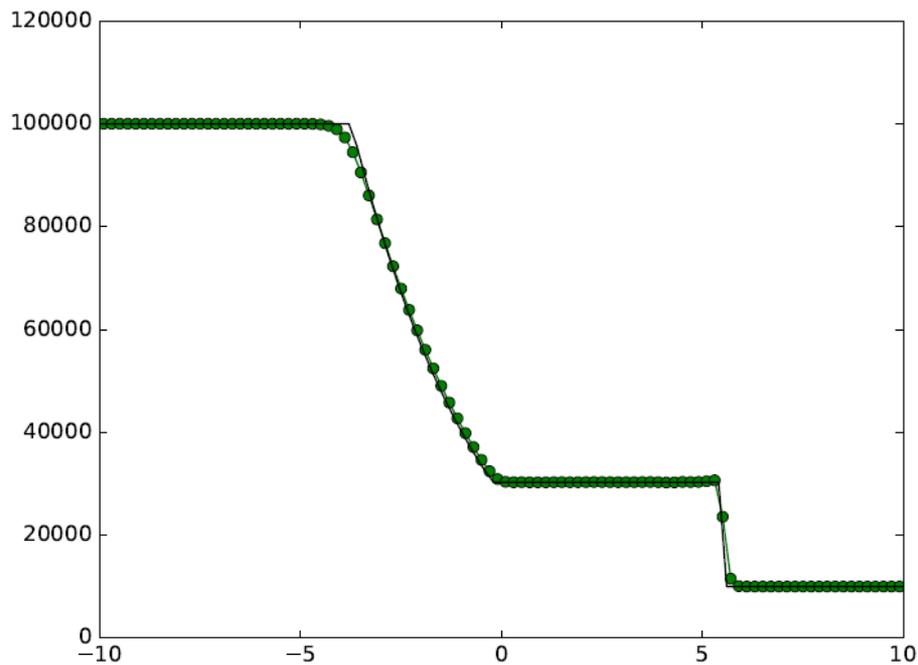


Figure 44: Pressure distribution (Roe Flux)

4.4 Spherical Explosion Problem

The spherical explosion test case is a test problem for three dimensional Euler equations. This is quite a challenging test case as it involves WENO reconstruction and flux in all three dimensions, and capturing a spherical shock, discontinuity and expansion waves. This test case demand a very strong implementation because of this reason.

The geometry and the initial conditions are such that spherical symmetry is imposed. A spherical high pressure core region is surrounded by a constant pressure environment as shown in figure 45. The initial conditions are defined in two regions, one being the region inside a sphere of radius 0.4 and origin at (0, 0, 0), and the second region being the section outside the sphere. Separate constant values are given in each section and the two sections can be considered to be joined by a spherical discontinuity at initial time $t = 0$. This test case can be hence considered to be a three dimensional extension of the shock tube problem. And as in the shock tube problem the initial velocities both inside and outside the sphere are set to zero. Here, the three dimensional Euler equations are solved in a cubic domain $[-1 : 1] \times [-1 : 1] \times [-1 : 1]$. The test case is simulated until a time of $t = 0.25$. At this time the waves have still not reached the boundaries and hence no additional boundary conditions are necessary. The initial data is:

$$(p, \rho) = \begin{cases} (1.0, 1.0), & r \leq 0.4 \\ (0.125, 0.1), & r > 0.4 \end{cases}$$

$$u = v = w = 0$$

$$r^2 = x^2 + y^2 + z^2$$

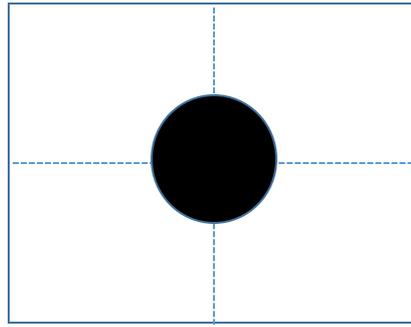


Figure 45 ; Spherical explosion test problem set up

The density distribution at time $t = 0.25$ is shown in figure 46. The solution for the density distribution is qualitatively similar to the shock tube problem with a spherical shock wave that is propagating outwards from the center, followed by a spherical contact discontinuity which is travelling in the same direction but with a smaller velocity; and a spherical rarefaction wave travelling towards the origin. However the strength of the shock becomes weaker as it travels outwards and the velocity profile between the shock and the discontinuity stops being constant, which is contrary to the results for the shock tube case. Figure 47 shows the pressure distribution. The pressure is continuous along the contact. Figure 48 shows the internal energy distribution. The results are well resolved with no spurious oscillations present.

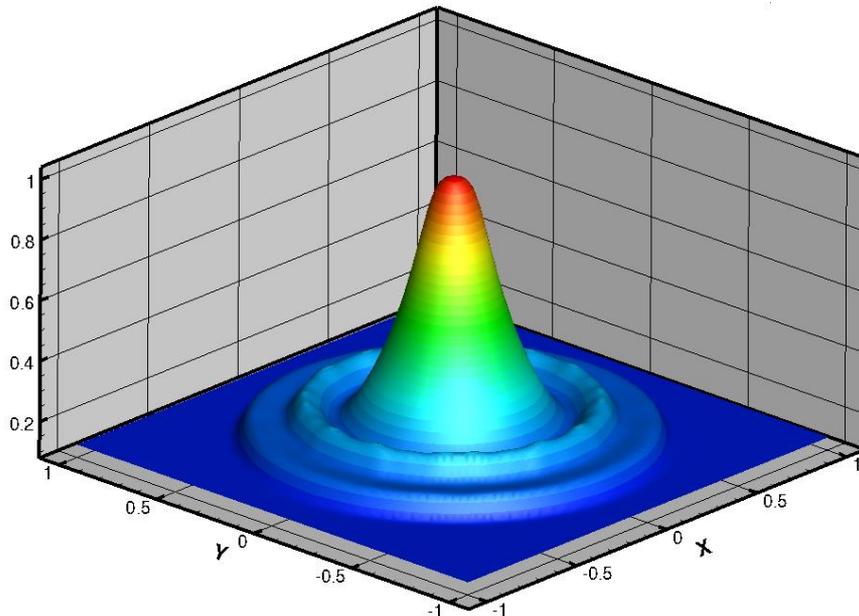


Figure 46: Density distribution at time $t = 0.25$ on plane $Z = 0$

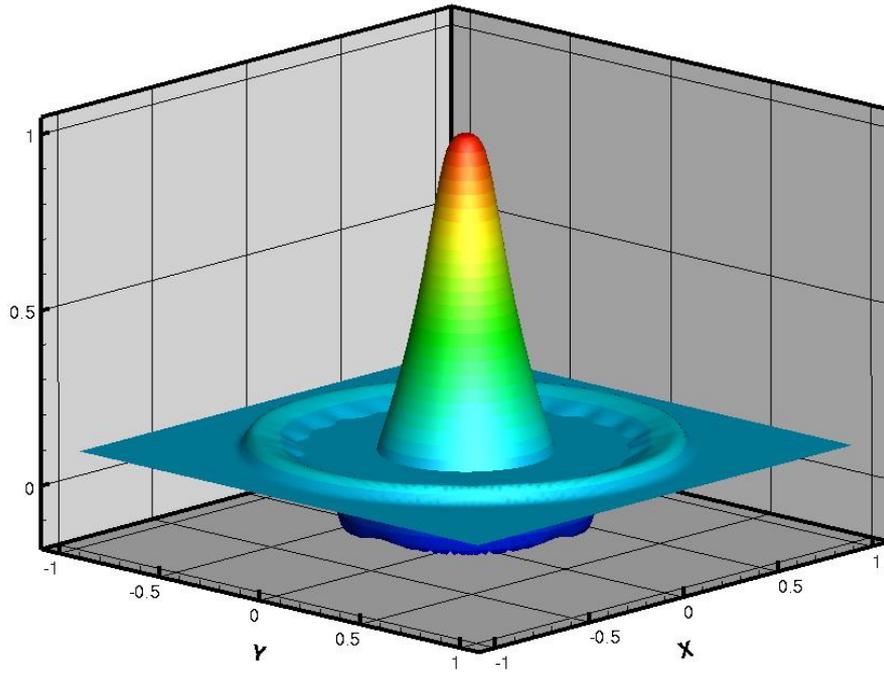


Figure 47: Pressure distribution at time $t = 0.25$ on plane $Z = 0$

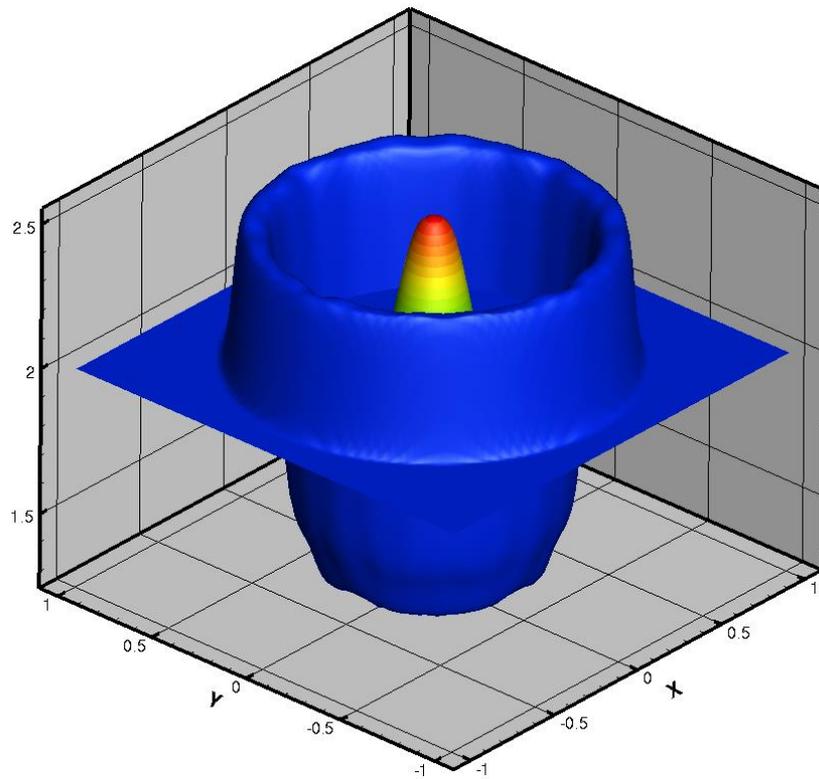


Figure 48: Internal Energy distribution at time $t = 0.25$ on plane $z = 0$

4.5. Two dimensional Riemann problems:

The Riemann problems for two dimensional gas dynamics provides a good set of scenarios to gauge the performance of a numerical scheme. In a two dimensional plane, four different sets of initially independent and constant states in four quadrants are allowed to interact, much like the one dimensional Sod shock tube problem. This results in the creation of very complex patterns as an aftermath of the interaction between shock wave, rarefaction wave and contact discontinuities. This provides the ideal platform to study the effect of the different Riemann solvers. Lax and Liu tested their positive numerical scheme for different such configurations (Lax & Liu, 1998), hence providing a library of contour plots which shall serve as the standard of comparison for this work. Another good reference set is provided by (San & Kara, 2015) and (San & Kara, 2014).

A two dimensional Riemann problem is the initial value problem for the Euler equation (43) with initial data

$$(P, \rho, u, v)(x, y, 0) = (P_i, \rho_i, u_i, v_i), \quad i = 1, 2, 3, 4$$

The Riemann problems studied by Lax and Liu are ones in which the initial condition is restricted so that each planar wave at an interface between two quadrants contains a single elementary wave which could be a single one-dimensional rarefaction wave, a one-dimensional shock, or a two-dimensional slip line (contact discontinuity).

Configurations 3, 5, 6, 11, 13, 14 and 17 described by Lax and Liu are recreated. The results of the WENO schemes (figures 49-54) are very similar to those in the literature. A 400 x 400 grid is used to study the results. A one-point Gaussian quadrature formula is used. Time integration is carried out by the fourth order Runge Kutta scheme. The results also demonstrate that two-dimensional gas flow with Riemann initial data is very complicated because of wave patterns including Mach reflection, rolling up of slip-lines, possible instability of slip lines etc. (Lax & Liu, 1998).

Configuration 3		Configuration 5		Configuration 11	
$\rho_2 = 0.5323$	$\rho_1 = 1.5$	$\rho_2 = 2.0$	$\rho_1 = 1.0$	$\rho_2 = 0.5313$	$\rho_1 = 1.0$
$u_2 = 1.206$	$u_1 = 0.0$	$u_2 = -0.75$	$u_1 = -0.75$	$u_2 = 0.8276$	$u_1 = 0.1$
$v_2 = 0.0$	$v_1 = 0.0$	$v_2 = 0.5$	$v_1 = -0.5$	$v_2 = 0.0$	$v_1 = 0.0$
$P_2 = 0.3$	$P_1 = 1.5$	$P_2 = 1.0$	$P_1 = 1.0$	$P_2 = 0.4$	$P_1 = 1.0$
$\rho_3 = 0.138$	$\rho_4 = 0.5323$	$\rho_3 = 1.0$	$\rho_4 = 3.0$	$\rho_3 = 0.8$	$\rho_4 = 0.5313$
$u_3 = 1.206$	$u_4 = 0.0$	$u_3 = 0.75$	$u_4 = 0.75$	$u_3 = 0.1$	$u_4 = 0.1$
$v_3 = 1.206$	$v_4 = 1.206$	$v_3 = 0.5$	$v_4 = -0.5$	$v_3 = 0.0$	$v_4 = 0.7276$
$P_3 = 0.029$	$P_4 = 0.3$	$P_3 = 1.0$	$P_4 = 1.0$	$P_3 = 0.4$	$P_4 = 0.4$

Configuration 13

$\rho_2 = 2.0$ $\rho_1 = 1.0$
 $u_2 = 0.0$ $u_1 = 0.0$
 $v_2 = 0.3$ $v_1 = -0.3$
 $P_2 = 1.0$ $P_1 = 1.0$

$\rho_3 = 1.0625$ $\rho_4 = 0.5313$
 $u_3 = 0.0$ $u_4 = 0.0$
 $v_3 = 0.8145$ $v_4 = 0.4276$
 $P_3 = 0.4$ $P_4 = 0.4$

Configuration 14

$\rho_2 = 1.0$ $\rho_1 = 2.0$
 $u_2 = 0.0$ $u_1 = 0.0$
 $v_2 = -1.2172$ $v_1 = -0.5606$
 $P_2 = 8$ $P_1 = 8.0$

$\rho_3 = 0.4736$ $\rho_4 = 0.9474$
 $u_3 = 0.0$ $u_4 = 0.0$
 $v_3 = 1.2172$ $v_4 = 1.1606$
 $P_3 = 2.6667$ $P_4 = 2.6667$

Configuration 17

$\rho_2 = 2.0$ $\rho_1 = 0.5313$
 $u_2 = 0.0$ $u_1 = 0.1$
 $v_2 = -0.3$ $v_1 = 0.1$
 $P_2 = 1$ $P_1 = 0.4$

$\rho_3 = 0.8$ $\rho_4 = 1.0$
 $u_3 = 0.1$ $u_4 = 0.1$
 $v_3 = 0.1$ $v_4 = 0.8276$
 $P_3 = 1.0$ $P_4 = 1.0$

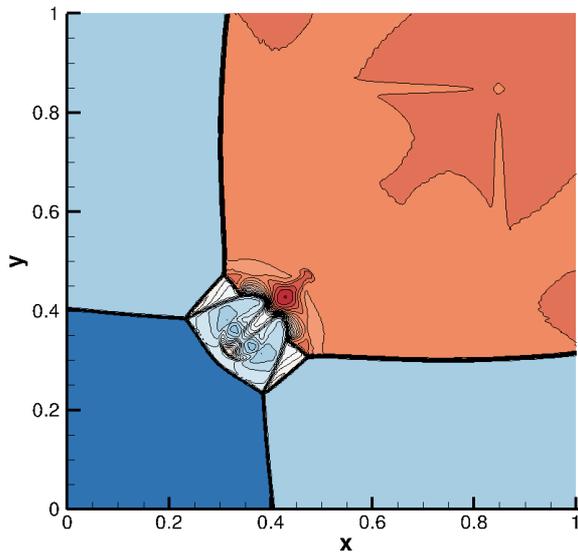


Figure 49: Configuration 3 ($t = 0.3$)

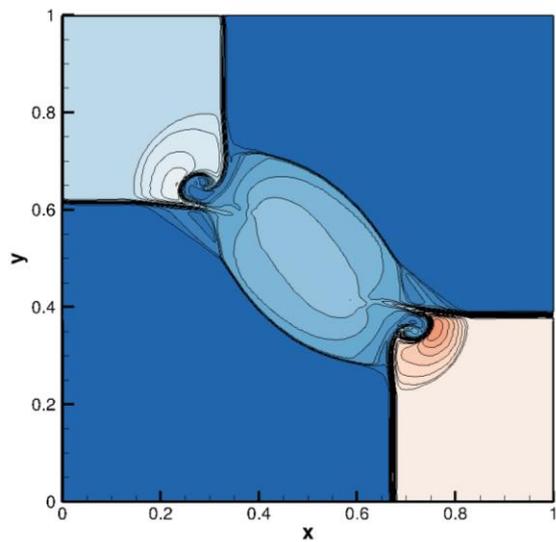


Figure 50: Configuration 5 ($t=0.23$)

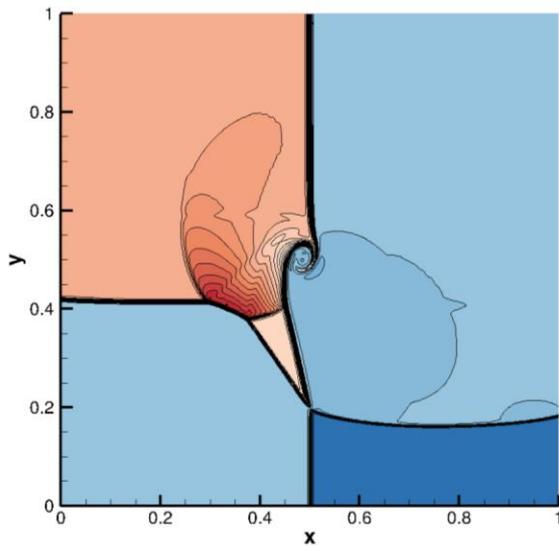


Figure 51: Configuration 13 ($t = 0.3$)

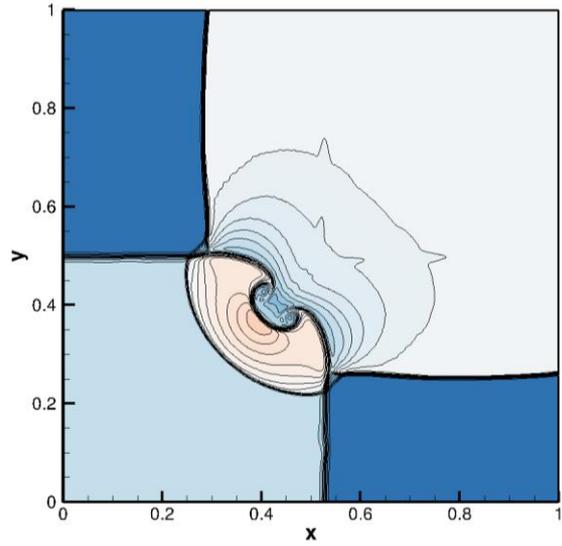


Figure 52: Configuration 11 ($t = 0.3$)

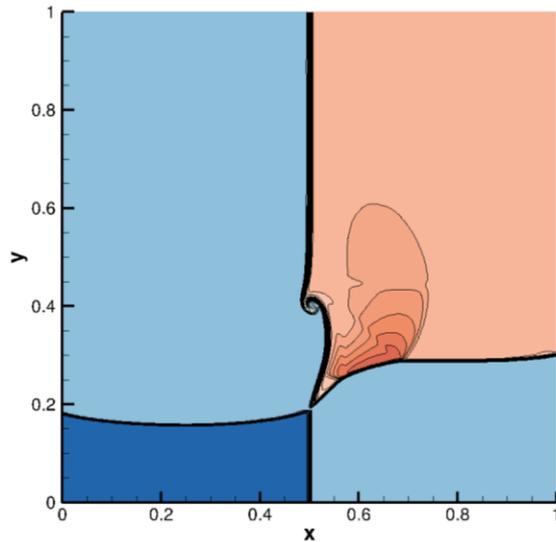


Figure 53 : Configuration 14 (t = 0.2)

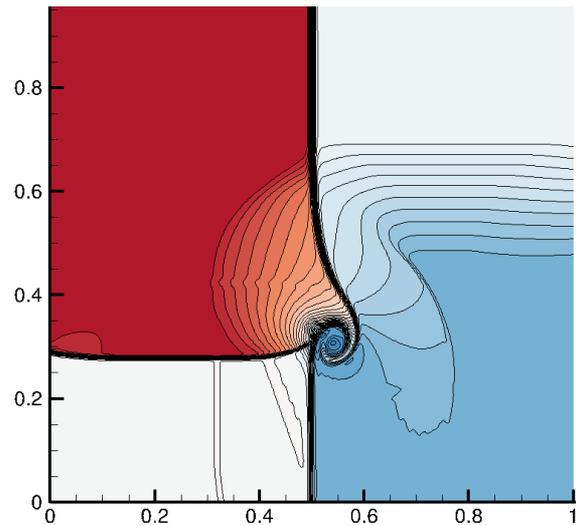


Figure 54: Configuration 17 (t = 0.3)

The performance of different Riemann solvers (approximate) are investigated using Configuration 3 and Configuration 17. Two different grids 400 x 400 and 800 x 800 are used to evaluate the results.

4.5.1 Configuration 3

Configuration 3 consists of four shocks travelling in the same direction as shown below.

$$\begin{array}{ccc}
 & \overline{s}_{21} & \\
 \overline{s}_{32} & & \overline{s}_{41} \\
 & \overline{s}_{34} &
 \end{array}$$

The four shock structure produces an intersection region which becomes clearer as grid resolution is improved. 38 density contours between 0.13 and 1.7 is shown in all figures.

Since no exact solution is available for these test cases, a reference solution is obtained by running the same initial conditions on a very fine grid of 1600 x 1600 with a Roe solver. The simulation results in a solution which captures a lot of features including a Kelvin-Helmholtz type vortices at the intersection of the four shocks. Most of the interesting features are present at the intersection and hence we focus on the solution at this section to study the performance of the Riemann solvers. This zoomed section is shown in figure 55. In a study of different 2d Riemann problems conducted by San and Kara (San & Kara, 2014), the same configuration was studied using second order MUSCL schemes, third-order WENO schemes using both the flux-splitting and reconstruction method, and a fifth order WENO scheme using the flux splitting approach. The small-scale vortical flow structures which appear in the solution are characteristic of viscous flows with high Reynold's number. These vortices were not captured in either of the second order MUSCL schemes or the third order WENO schemes

(both flux splitting and reconstruction approach) even at a high resolution of 3200 x 3200. San and Kara were able to capture these vortices using the flux splitting approach, specifically the Van Leer's flux splitting scheme. The results from this work, mainly figure 55, shows that similar vortices can also be captured by the WENO reconstruction approach using approximate Riemann solvers.

Eventhough the Euler equations represent inviscid flow, numerical dissipation is inherent in shock capturing schemes (as a result of upwinding) which acts as artificial viscosity. This dissipation reduces as the order of the scheme is increased. Also, increasing the resolution of the grid reduces the dissipation range. Hence a simultaneous increase of the order of accuracy of the scheme and the resolution of the scheme results in an increase in the artificial Reynold's number in each cell. The solution convects the vorticity and generates vortice like small scale flow structures which are common to high Reynolds number turbulent flows (San & Kara, 2014). This explains the appearance of the vortice type structures in a fine grid using the fifth order WENO schemes and the absence in corresponding third order WENO schemes.

Figures 56-63 shows the performance of the different approximate Riemann solvers on different grids. The differences are not clear on a grid of 400 x 400. However on a grid of 800 x 800, the approximate Riemann solvers distinguish themselves with the HLLC flux standing out. Even on a grid of 800 x 800, certain wiggles are seen in the intersection region where the vortices show up in the exact solution. This is a very good result since in the flux splitting approach using Van Leer's flux splitting on a similar grid of 800 x 800 in (San & Kara, 2014), these features do not appear. The results for both the Lax-Friedrichs and the HLL flux are similar because the dissipation is too high in both the schemes to clearly resolve the small scale features on the given grid.

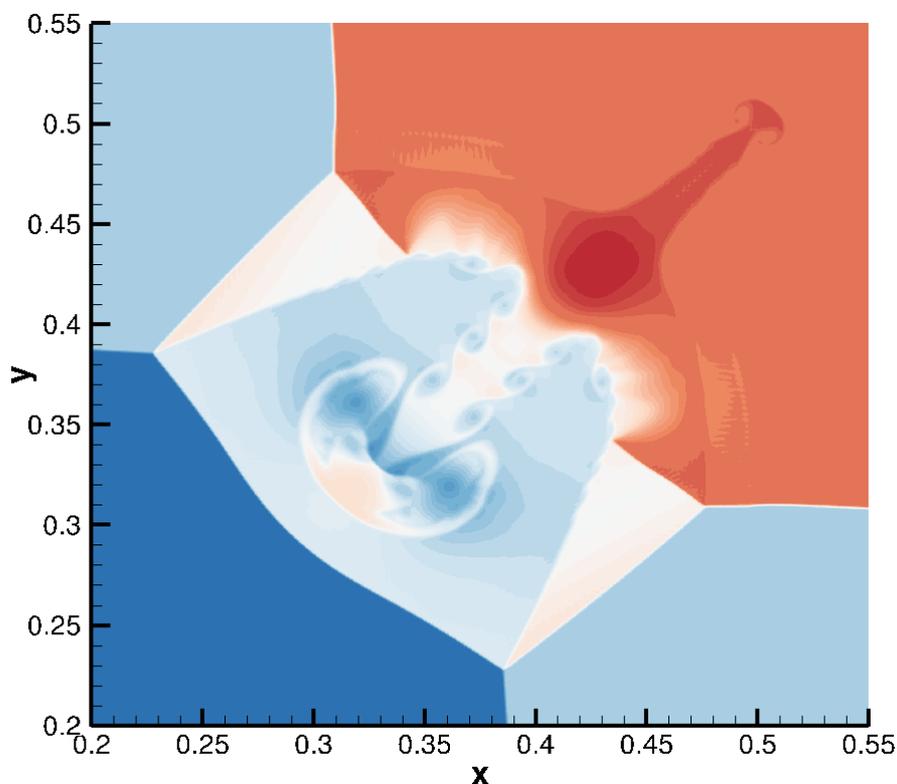


Figure 55: Exact solution on a grid 1600 x 1600 using Roe flux with Entropy fix (38 contours between 0.13 and 1.7)

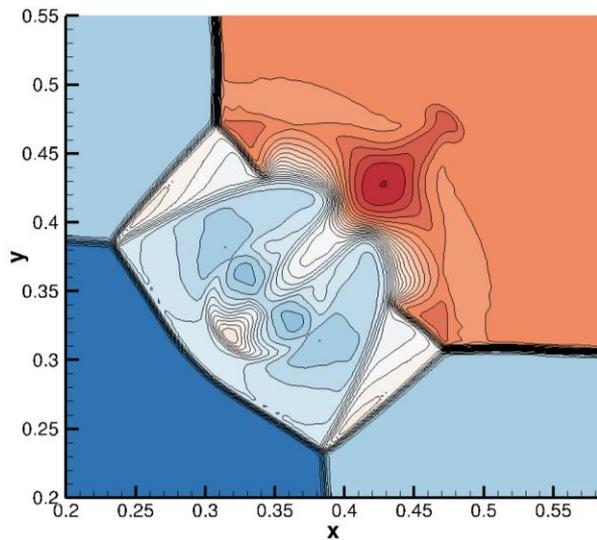


Figure 56: Lax- Friedrichs (400 x 400)

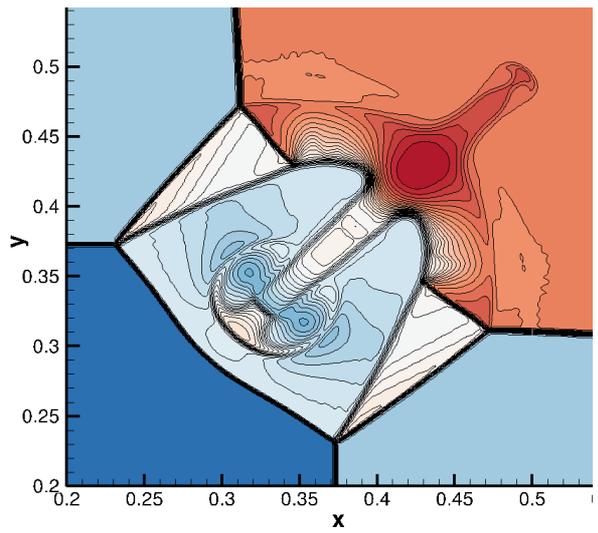


Figure 57: Lax-Friedrichs (800 x 800)

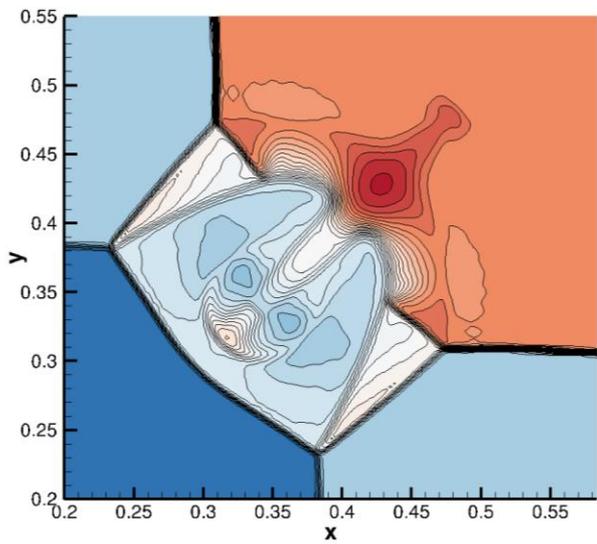


Figure 58: HLL (400 x 400)

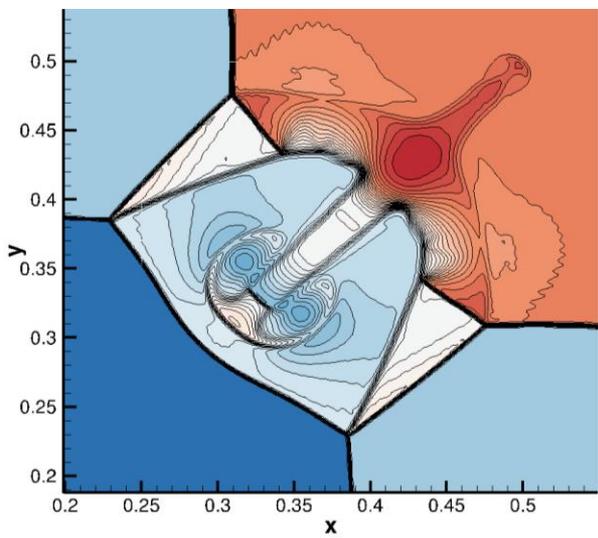


Figure 59: HLL (800 x 800)

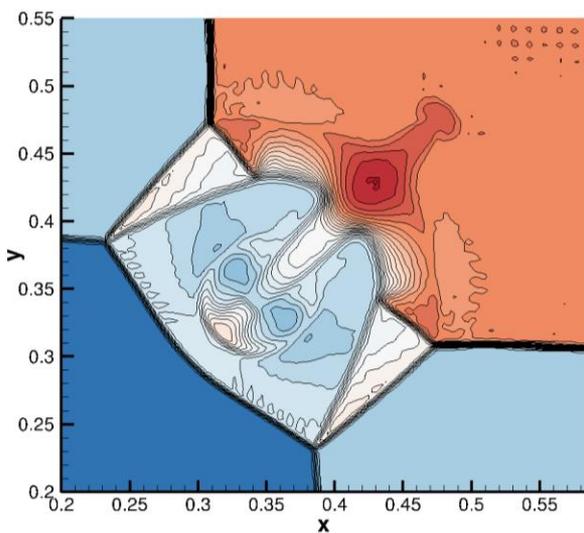


Figure 60: Roe Flux with Entropy Fix (400 x 400)

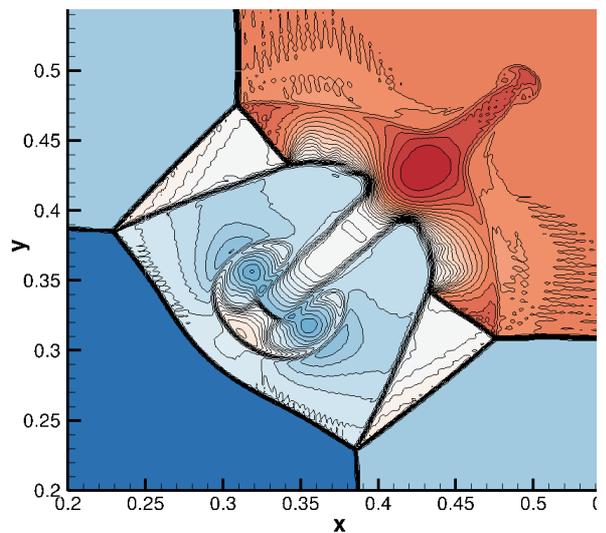


Figure 61: Roe Flux with Entropy Fix (800 x 800)

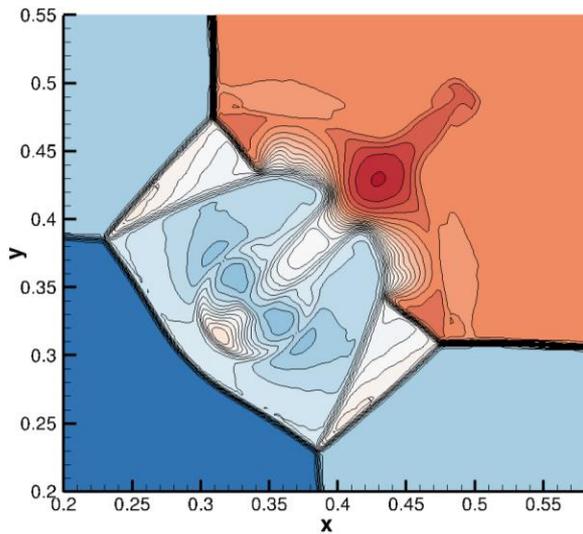


Figure 62: HLLC Flux (400 x 400)

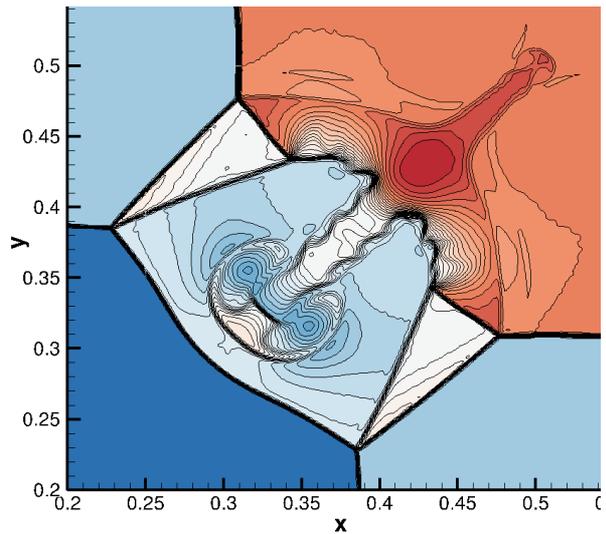


Figure 63: HLLC Flux (800 x 800)

4.5.2 Configuration 17

Configuration 17 has a shock, a rarefaction wave and two contact discontinuities as shown below.

$$\begin{array}{ccc}
 & J_{21}^- & \\
 \overleftarrow{S}_{32} & & \overrightarrow{R}_{41} \\
 & J_{34}^- &
 \end{array}$$

A similar analysis as done for Configuration 3 is conducted for Configuration 17 as well. Again, the reference solution is obtained using a fine grid of 1600 x 1600 as shown in 64.

Figures 65 – 72 show the results obtained using the different Riemann solvers on grids of 400 x 400 and 800 x 800. The exact solution shows similar vortices like flow structures. The vortices in this case are possibly formed due to the discontinuities in the flow. These are captured by the Roe flux in a grid of 800 x 800. All the other schemes prove to be too dissipative to capture these fine features on the given grids. However the shock is captured satisfactorily by all the schemes. The same applies for the rarefaction wave.

Both the Roe flux and the HLLC flux have shown to be very good approximate Riemann solvers. The Roe solver is slightly more expensive than the HLLC scheme. Even though the Lax-Friedrichs flux and the HLL flux are computationally cheap and less complex, they are too dissipative to be considered for practical flow simulation.

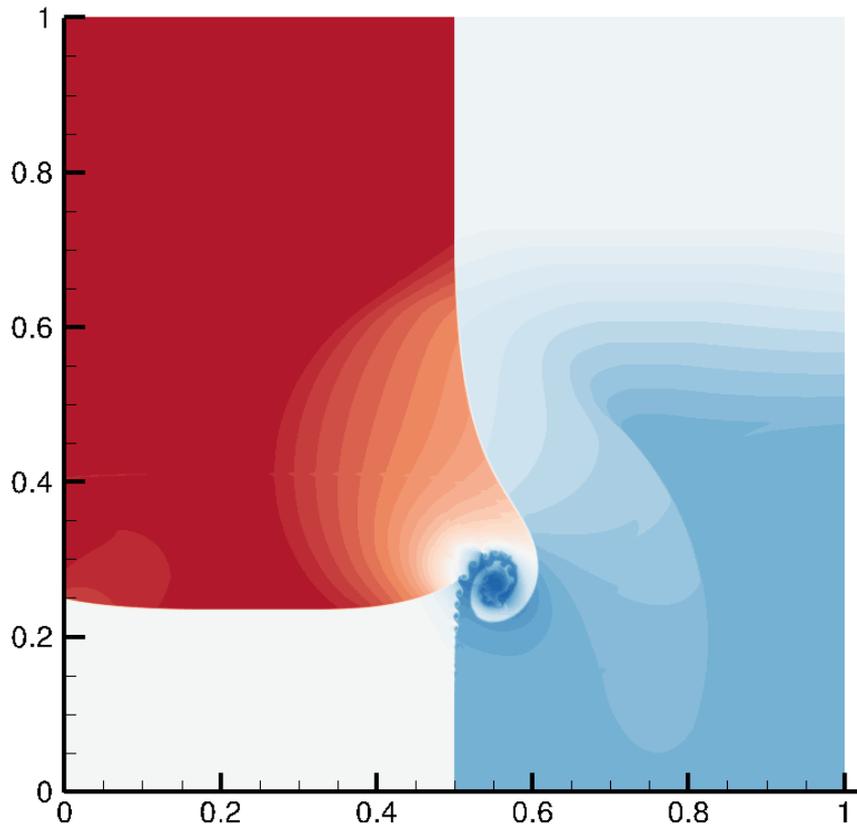


Figure 64: Exact solution on a grid 1600 x 1600 using HLLC flux (37 contours between 0.2 and 1.95)

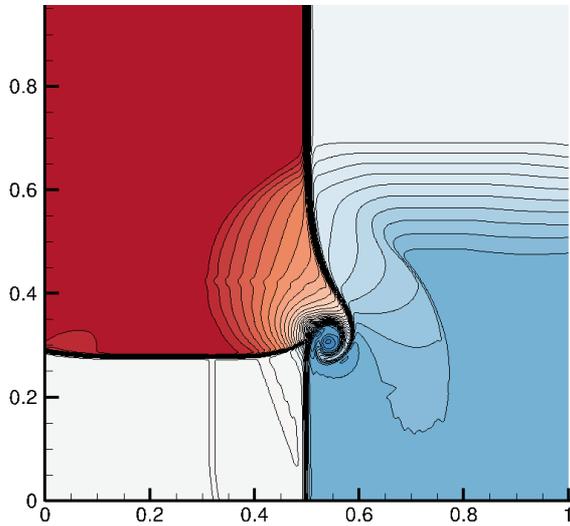


Figure 65: Lax-Friedrichs flux (400 x 400)

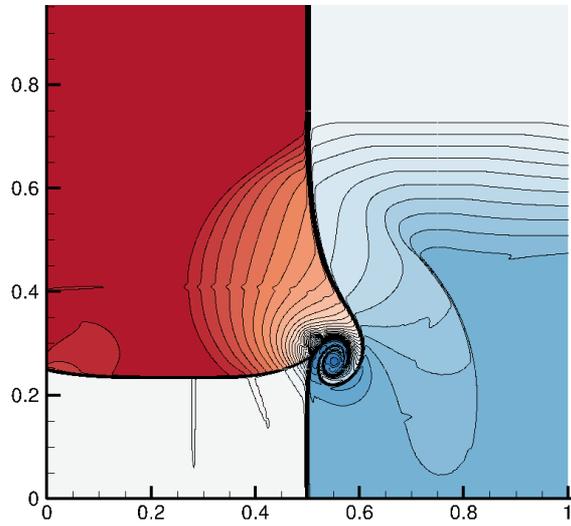


Figure 66: Lax-Friedrichs flux (800 x 800)

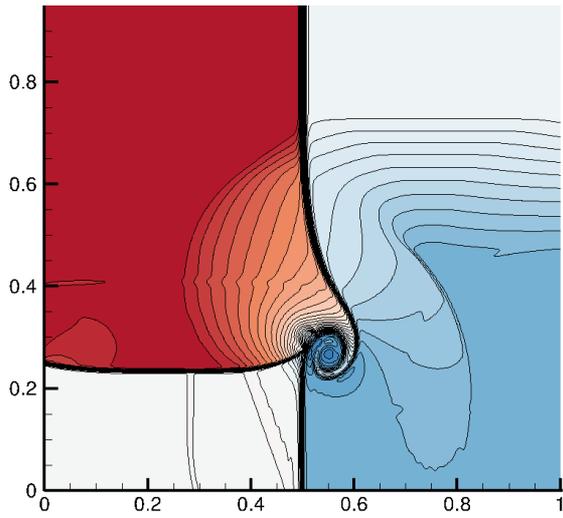


Figure 67: HLL flux (400 x 400)

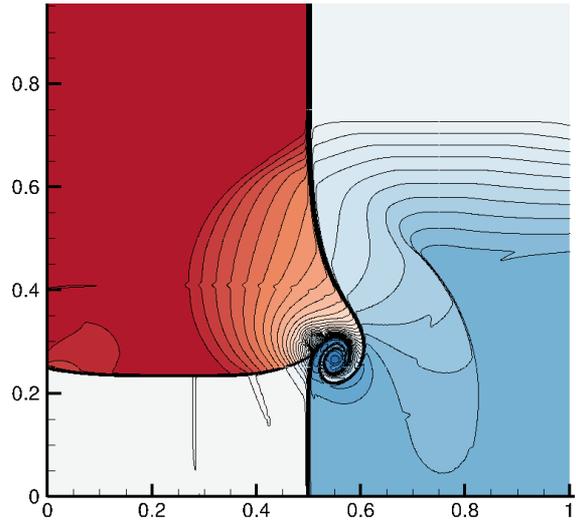


Figure 68: HLL flux (800 x 800)

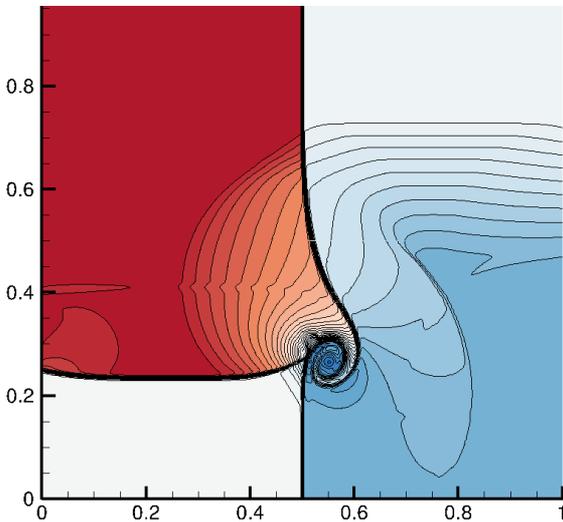


Figure 69: HLLC flux (400 x 400)

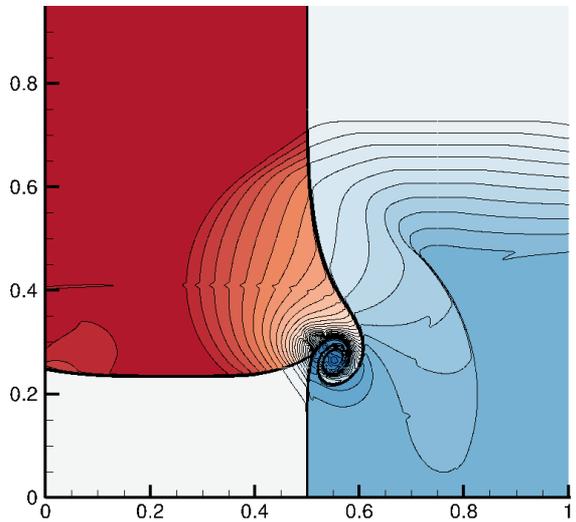


Figure 70: HLLC flux (800 x 800)

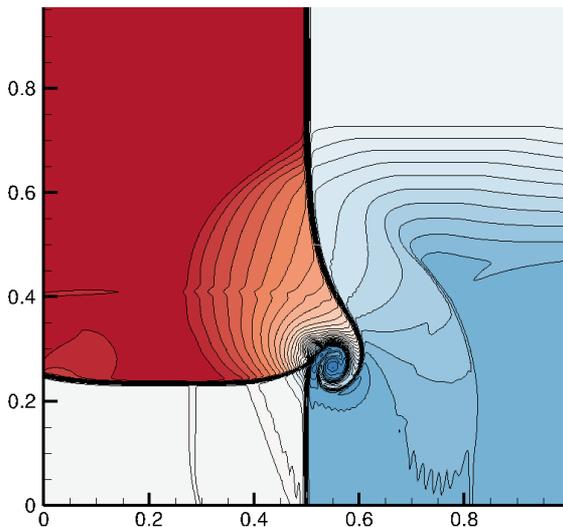


Figure 71: Roe flux with entropy fix (400 x 400)

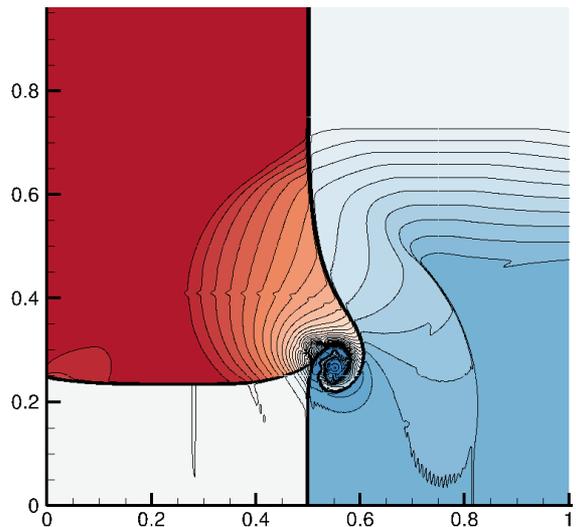


Figure 72: Roe flux with entropy fix (800 x 800)

4.6. Wind tunnel with Forward Facing Step:

This two dimensional problem, though introduced almost fifty years ago, is a very strong test case as it involves the simulation of fluid flow with strong shocks in two dimensions. For such flows, a careful treatment of flow discontinuities is required to obtain accurate numerical results. Multiple shocks are present that reflect off the walls of the tunnel and also from the top of the step in the tunnel. This makes it a very complicated test case, although it can be considered to be the ultimate test to evaluate the shock capturing ability of the scheme.

A uniform Mach 3 flow is introduced into a wind tunnel with a step. The dimensions of the wind tunnel is 3 length units x 1 length unit. It is assumed to have an infinite length in the span wise direction. The step is located at 0.6 length units to the right of the inlet of the wind tunnel, and is 0.2 length units high. At the inlet, a supersonic inflow boundary condition is imposed, and a supersonic outflow boundary condition is imposed at the outlet as the exit velocity is always supersonic. At the start of the simulation, the wind tunnel is assumed to be filled with a gamma-law gas with the following initial values for the state variables:

$$\begin{aligned}P &= 1.0 \\ \rho &= 1.4 \\ u &= 3.0\end{aligned}$$

Gas with these properties is continuously fed in from the inlet boundary. A reflective boundary condition is implemented along the walls of the wind tunnel.

For this fluid flow simulation, steady flow develops at a time of 12 seconds. However since the flow at this time has very little structure, the flow is simulated up to a time of 4 seconds and the density distribution is studied. The results for a uniform grid of 1440 x 60 is shown. A single quadrature point is used and the Lax Friedrichs Riemann solver is employed. Again, the fourth order Runge Kutta time integration scheme marches the solution forward in time.

As noted by Woodward and Colella, the corner of the step acts as a singularity in the flow as it is the center of a rarefaction fan (Woodward & Colella, 1984). Most papers perform special treatments at this corner to remove this singularity and obtain better visualization. However for the current work, no special treatment is done at this point as focus is more on analyzing the performance of the code with regards to shock capturing and the correct implementation of the reflecting boundary conditions. Therefore, certain numerical errors are generated and introduced into the flow in the vicinity of this point. The errors cause the formation of an artificial boundary layer on the step of the wind tunnel. A weak oblique shock is also formed at the corner of the step. Shocks interact with the boundary layer resulting in the formation of a recirculation zone at the time of 4 seconds. However these errors are greatly reduced for the highly refined grid of 1440 x 600, for which the results are shown for a series of time steps (Figures 73-79).

Immediately after the start of the simulation, a shock is formed directly in front of the step. It slowly moves forward while curving around the corner of the step, extending further downstream while growing in size and finally hitting the top wall of the wind tunnel.

As the shock reaches the top wall, it is reflected and as it again moves left, the angle between the incident shock and the wall increases. After some time, it exceeds the maximum angle for regular reflection and begins to form a Mach stem. Meanwhile further reflection takes place from the top of the step, and the point of intersection moves left. The second reflection moves back toward the top of the wind tunnel, and forms a third reflection. Due to the presence of the numerical boundary layer, now we see a second Mach stem and a recirculation zone forming at the shock reflection from the surface of the step as the shock interacts with the numerical boundary layer. However the length of this Mach stem reduces as the grid is refined.

Behind the first Mach stem, an interesting numerical effect occurs. Kelvin-Helmholtz amplification of numerical errors generated at the shock intersection start becoming visible. The waves thus generated propagate downstream and are refracted by the second and third reflected shocks. The effect does not disappear with increasing resolution, for two reasons. First, the instability amplifies numerical errors generated at the shock intersection, no matter how small. Second, the effect of numerical viscosity, which can diffuse away instabilities on coarse grids, is greatly reduced at high resolution.

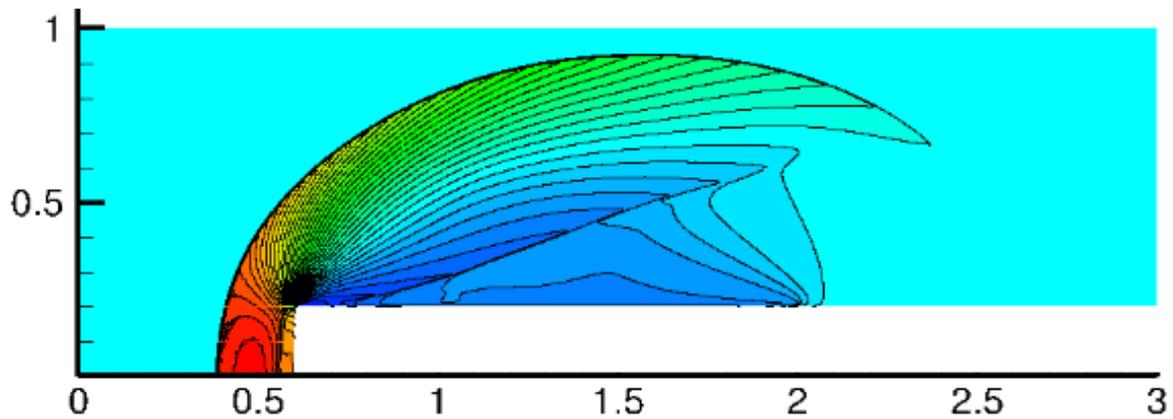


Figure 73: Density distribution at $t = 0.5s$

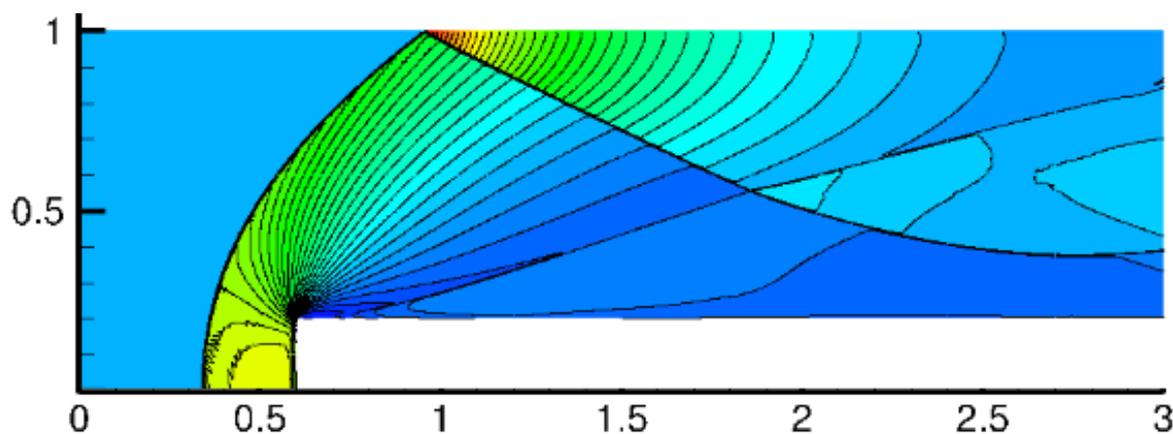


Figure 74: Density distribution at $t = 1.0s$

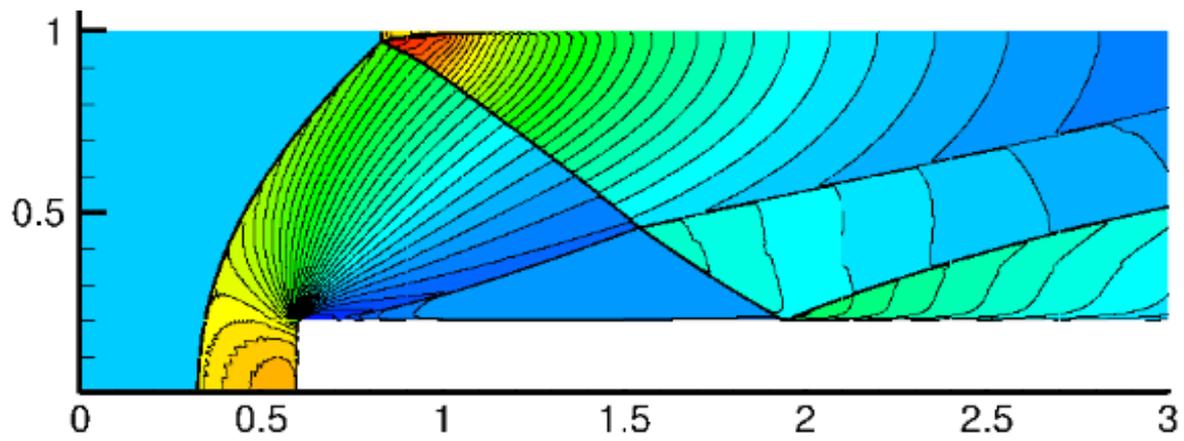


Figure 75: Density distribution at $t = 1.5$ s

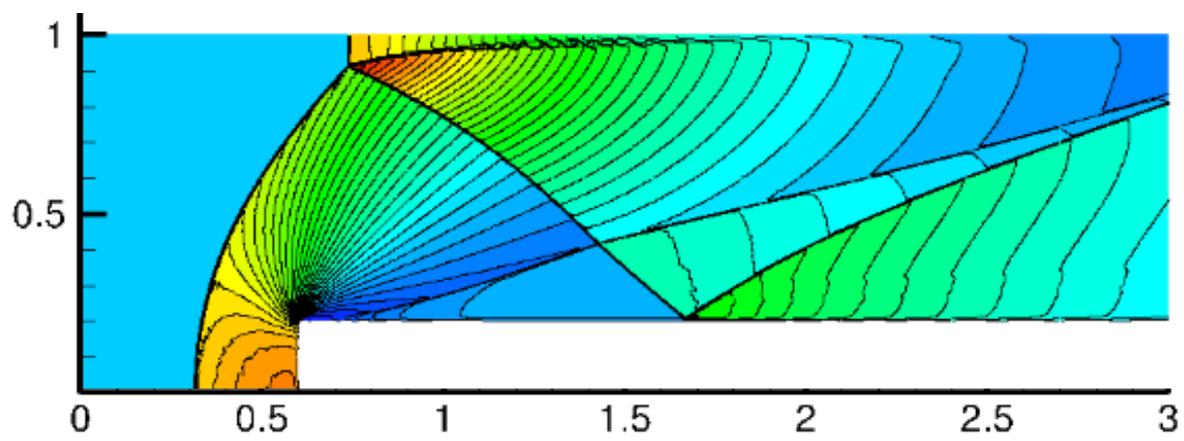


Figure 76: Density distribution at $t = 2.0$ s

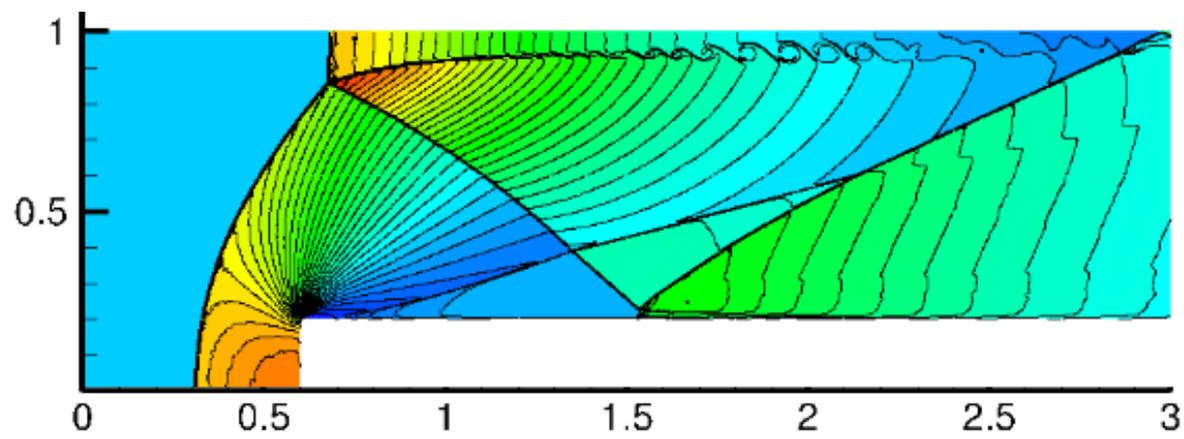


Figure 77: Density distribution at $t = 2.5$ s

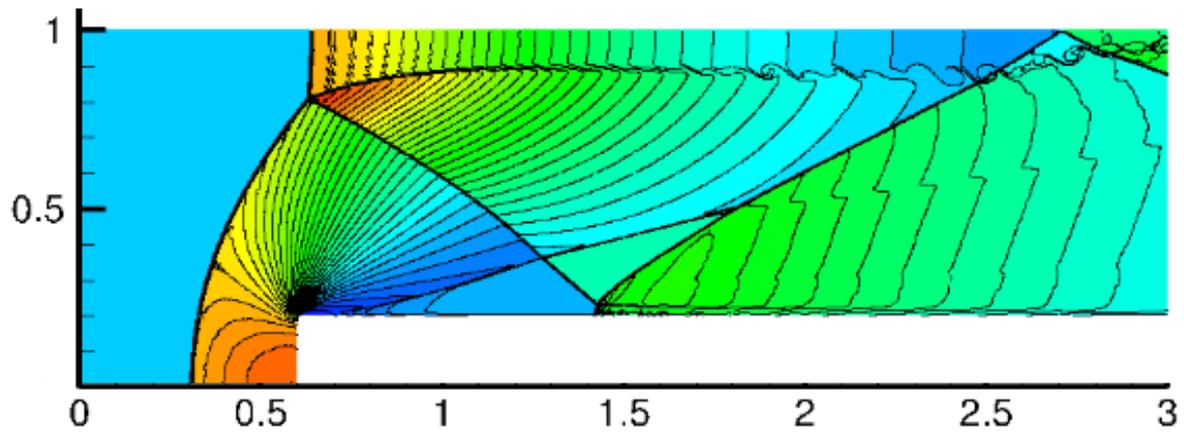


Figure 78: Density distribution at $t = 3.0s$

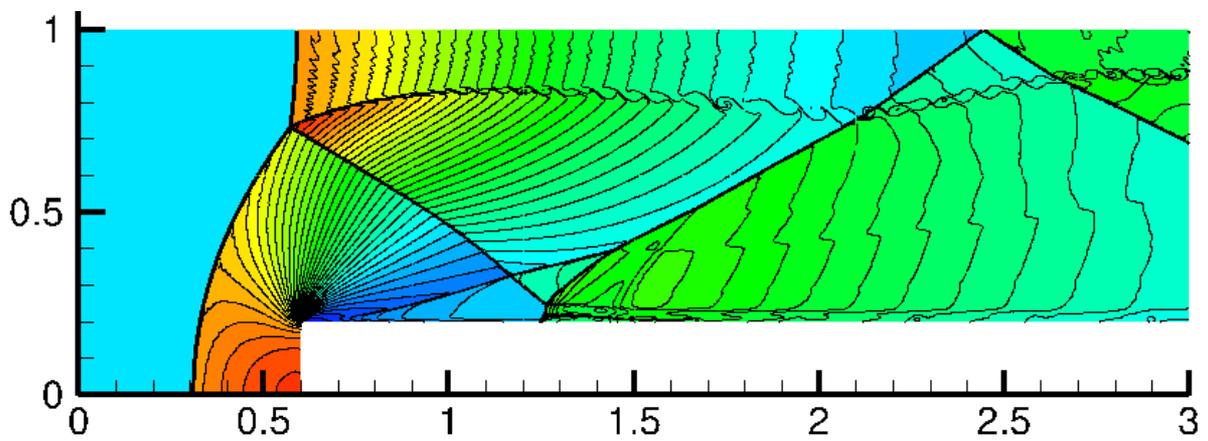


Figure 79 Density distribution at $t = 4s$

5

Conclusion and Recommendations

5.1 Conclusions

The main goal of the thesis is to successfully implement a fifth order Weighted Essentially Non-Oscillatory scheme in the TU Delft Solver *Vayu* and validate the scheme using different test cases. The governing equations for the test cases involved both the advection equations and the Euler equations.

A detailed explanation of the steps involved in the implementation process, for both the finite volume framework and the WENO reconstruction is given in Chapter 3. A variety of test cases, both simple and difficult, were selected for validation. Simple test cases such as the propagation of a two dimensional sine wave in 1d and 2d are studied to analyse the order of convergence and the effect of the Gaussian quadrature formula in flux approximation. No variation was observed in the results and the order of convergence for simulations with one and two point Gaussian quadrature formulas. The approximation error when using only one quadrature point was too small to make a significant difference, and hence same order of convergence was observed. This answers the sub question regarding the influence of quadrature points on the flux approximation at the cell-faces.

The Doswell Frontogenesis test case is a challenging test case using the advection equations, and the solver successfully captured the front. Comparison with the exact solution gave very impressive plots. Also an order of convergence test with a smooth front gave very good and expected high order of accuracy of four.

The Spherical Explosion test case was probably the most computationally expensive simulation with 1.03 million points; and personally one of the most difficult test cases to implement. However the excellent results which are very identical to those seen in literature justified the time spent on this test case. This test case was the only problem which involved WENO reconstruction in all three dimensions and used practically all the modules of the solver.

The second research sub-question deals with the influence of the approximate Riemann solvers on the quality of the results. The Sod shock tube problem was the first test case in which all the effect of all the Riemann solvers were studied. The solver performed very well in capturing the shock and resolving the contact discontinuity, and the comparison with the exact solution for density, velocity and pressure showed excellent results. The effect of

different approximate Riemann solvers was also studied. However only the Roe solver showed any significant visible difference.

Different configurations of the two dimensional Riemann problems were simulated using different Riemann solvers and the results were studied. First, certain selected configurations were simulated to see how well the WENO schemes captured the variety of shocks, expansion waves and/or contact discontinuities. Since no exact solution is available for these problems, comparisons were made with results obtained in previous literature. The results were more than satisfactory. Next, two configurations were selected and were simulated for four different Riemann solvers on grids with two different levels of refinement. The HLLC and Roe solvers were the least dissipative and capable of capturing most of the features present in the solution. Results were compared with a solution obtained on a very fine grid which was run in parallel on the cluster on 16 processors after dividing the grid into 16 blocks. The exact solution revealed the presence of small-scale vortices in the flow which are usually common in high Reynolds number flows. Even though the governing equations were the inviscid Euler equations, the presence of these Kelvin-Helmholtz like flow structures are understandable due to the numerical dissipation inherent in shock capturing schemes.

The study of the above two test cases led to the somewhat known conclusion that the Roe solver is the least dissipative. However it is also the most expensive, at least in the form it has been implemented currently in the solver. The HLLC solver is the second best candidate and is considerably less expensive even though it captures most of the information in the solution of the Riemann problem. The HLL and the Lax Friedrichs flux were similar in terms of the quality of the results since they are both highly dissipative. Hence ideally, the choice of the Riemann solver should be made based on the application.

The final test case was the flow in a wind tunnel with a forward facing step. This test case is challenging as it deals with multiple shocks reflecting off the walls of the tunnel and the surface of the step. Personally, the most interesting test case, the solver showed excellent shock capturing ability and perfect reflection at the walls and the step. On a refined grid, Kelvin-Helmholtz like vortices were visible at one of the Mach stems formed at the walls. These vortices were caused due to the interaction of the shocks with the numerical errors at the Mach stem. The simulation was run on the cluster on 32 processors after dividing the computational grid into 32 blocks. This was one of the first test cases to be run in parallel on the cluster and the solver exceeded expectations by providing excellent results without a single glitch.

The main research question dealt with the quality of results provided by the fifth order WENO schemes on the TU Delft solver *Vayu* with respect to shock capturing ability. All the test cases dealt with compressible flows and provided high quality results. There is no questioning the robust shock capturing ability of the WENO schemes. Even in the smooth portions of the flow where it is important that the scheme does not overly dissipate the flow features, the WENO schemes have provided satisfactory results as seen in the 2d Riemann problems especially. The high quality results are testament to the robust implementation of the fifth order schemes on the TU Delft solver *Vayu*. The fifth order WENO schemes can be confidently considered to be successfully implemented and validated on the solver.

5.2 Recommendations and Future work:

An important result of the work was that the calculation of the numerical flux at each cell interface in the computational grid of a high order finite volume scheme can be done with just the one point Gaussian quadrature rule without any significant loss in accuracy. This realization allows us to make the code less expensive by switching off the section of the code which performs the WENO reconstruction at the different quadrature points. It would also significantly reduce the complexity level of the code, and allows us to take away a significant number of fields which would bring down the memory requirements considerably. Hence it is recommended to use only one quadrature point for memory intensive simulations, especially for LES of practical flows using really fine grids.

The effect of the Riemann solvers have been studied. Even though these results have not been quantified numerically, it is quite clear visually from the obtained results that there is a significant influence of the Riemann solvers. The HLLC solvers would be the ideal choice for practical applications, especially LES. The HLL and Lax Friedrichs fluxes are too dissipative and the smaller flow features could be overwhelmed. The Roe flux with an entropy fix could be too expensive for a practical application. However, it should also be mentioned that the implementation of the HLLC flux in the solver can be made much more efficient. During development, coding was done to study the effects at each step and hence there is significant room for optimization before running a simulation for actual applications. This will further increase the appeal of using the HLLC flux.

The appearance of vortices like small-scale flow features in the Riemann problems on a fine grid, and the flow in the wind tunnel with a forward step, even while solving the inviscid Euler equations could be taken as motivation for using the WENO schemes in Implicit LES (ILES) codes. In ILES, no explicit sub-grid scale model is used to model the unresolved part of the flow. The numerical scheme is assumed to introduce enough dissipation to model the subgrid terms.

A module has been implemented in the solver to introduce the effect of the viscous terms and hence allow the simulation of flows governed by the Navier-Stokes equation. A no-slip boundary condition has also been implemented. A test case of sub-sonic laminar flow over a flat plate has also been set up. However validation is still pending. Simultaneously, an ILES test case of flow in a channel constricted by periodic hills has also been setup. This test case has been classified under the difficult section in the first three International Workshop on High-Order CFD methods. A very fine grid is required to view the smaller turbulence scales which is computationally very expensive for the fifth order WENO scheme. Due to restrictions in time, work on both these test cases had to be put on hold. However preliminary results on a coarse grid has given enough confidence to suggest that correct and high quality results are around the corner. Future work along the lines of ILES and also traditional LES can be done to develop the solver further.

Bibliography

- Barth, T., & Ohlberger, M. (2004). Finite volume methods: foundation and analysis. *Encyclopedia of Computational Mechanics Vol 1: Fundamentals, 2004*.
- Cockburn, B., Johnson, C., Shu, C.-W., & Tadmor, E. (1997). *Advanced Numerical Approximation of Nonlinear Hyperbolic Equations*. Cetraro, Italy: Springer.
- Godunov, S. (1959). A Difference Scheme for Numerical Solution of Discontinuous Solution of Hydrodynamic Equations. *Math. Sbornik, 47, 271-306*.
- Harten, A., Engquist, B., Osher, S., & Chakravarthy, S. (1987). Uniformly high-order accurate essentially non-oscillatory schemes III. *Journal of Computational Physics 71, 231-303*.
- Harten, A., Lax, P., & Van Leer, B. (1983). On upstream differencing and Godunov-type schemes for hyperbolic conservation laws. *Siam Review, 25(1): 35-61*.
- Jiang, G.-S., & Shu, C.-W. (1996). Efficient Implementation of Weighted ENO schemes. *Journal of Computational Physics 126, 202-228*.
- Johnsen, E., Larsson, J., Bhagatwala, A., Cabot, W., Moin, P., Olson, J., . . . Lele, S. (2010). Assessment of high-resolution methods for numerical simulations of compressible turbulence with shock-waves. *Journal of Computational Physics 229, 1213-1237*.
- Kurganov, A., & Tadmor, E. (2000). New high-resolution central schemes for nonlinear conservation laws and convection-diffusion equations. *Journal of Computational Physics 160, 241-282*.
- Laney, C. B. (1998). *Computational Gas Dynamics*. Cambridge University Press.
- Lax, P. D. (1972). Hyperbolic systems of conservation laws and the Mathematical Theory of Shock Waves. *SIAM*.
- Lax, P. D., & Liu, X.-D. (1998). Solution of Two Dimensional Riemann Problems of Gas Dynamics by Positive schemes. *SIAM Journal on Scientific Computing 19 Vol 2, 319-340*.
- Lax, P., & Wendroff, B. (1960). Systems of Conservation Laws. *Communications on Pure and Applied Mathematics, Vol XIII, 217-237*.
- Lele, S. (1992). Compact Finite Difference Schemes with Spectral-Like Resolution. *Journal of Computational Physics 103, 16-42*.
- Liu, X.-D., Osher, S., & Chan, T. (1994). Weighted Essentially non-oscillatory schemes. *Journal of Computational Physics 71, 200-212*.

- Lopez-Morales, M. R., Bull, J., Crabill, J., Economon, T. D., Manosalvas, D., Romero, J., . . . Jameson, A. (2014). Verification and Validation of HiFiLES: a High-Order LES unstructured solver on multi-GPU platforms. *32nd AIAA Applied Aerodynamics Conference*, (pp. AIAA-2014-3168).
- Martin, M., EM, T., Wu, M., & Weirs, V. (2006). A bandwidth optimized WENO scheme for the effective direct numerical simulation of compressible turbulence. *Journal of Computational Physics*, 220:270-289.
- Nichols, R., Tramel, R., & Buning, P. (2008). Evaluation of Two High Order Weighted Essentially Non-Oscillatory schemes. *AIAA Journal Vol 46*.
- Pirozzoli, S. (2011). Numerical Methods for High-Speed Flows. *The Annual Review of Fluid Mechanics* , 43:163-94.
- Ponziani, D., Pirozzoli, S., & Grasso, F. (2003). Development of optimized weighted-ENO schemes for multiscale compressible flows. *International Journal for Numerical methods in Fluids*, 42: 953-977.
- Roe, P. (1981). Approximate Riemann solvers, Parameter Vectors, and Difference Schemes. *Journal of Computational Physics* 43, 357-372.
- San, O., & Kara, K. (2014). Numerical assessments of high-order accurate shock capturing schemes: Kelvin-Helmholtz type vortical structures in high-resolutions. *Computers and Fluids* 89, 254-276.
- San, O., & Kara, K. (2015). Evaluation of Riemann flux solvers for WENO reconstruction schemes: Kelvin-Helmholtz instability. *Computers and Fluids* 117, 24-41.
- Taylor, E., Wu, M., & Martin, P. (2001). Optimization of nonlinear error for weighted essentially non-oscillatory methods in direct simulations of compressible turbulence. *Journal of Computational Physics*, 174:381-404.
- Titarev, V., & Toro, E. (2004). Finite Volume WENO schemes for three dimensional conservation laws. *Journal of Computational Physics Volume 201* , 238-260.
- Toro, E. (1999). *Riemann Solvers and Numerical Methods for Fluid Dynamics*. Springer.
- Van Leer, B. (1979). Towards the Ultimate Conservative Difference Scheme. V. A Second Order Sequel to Godunov's Method. *Journal of Computational Physics* 32, 101-136.
- Wang, Z., & Chen, R. (2001). Optimized Weighted Essentially Non-Oscillatory schemes for computational aeroacoustics. *Journal of Computational Physics* 174, 381-404.

Weirs, V., & Candler, G. (1997). Optimization of weighted ENO schemes for DNS of compressible turbulence. *AIAA Paper no 97-1940*.

Woodward, P., & Colella, P. (1984). The Numerical Simulation of Two Dimensional Fluid Flow with Strong Shocks. *Journal of Computational Physics*, 115-173.

Xu-Dong, L., Osher, S., & Chan, T. (1994). Weighted Essentially Non-oscillatory Schemes. *Journal of Computational Physics* 115, 200-212.

Appendix

A. Basic Numerical Methods for the Euler Equations

There are two different approaches for solving Euler equations - flux approaches and wave approaches. Wave approaches can be further subdivided into two categories – flux splitting approaches and reconstruction-evolution approaches. Whereas flux approaches consider only fluxes, wave approaches model the interaction between various families of waves, using either flux splitting or Riemann solvers, which makes them more physical and accurate but also more expensive and complicated (Laney, 1998).

A.1 Flux Approach

A.1.1 Lax- Friedrichs Method

The Lax-Friedrichs method is a first order scheme considered as an alternative to the first Godunov’s method. It is a forward in time, centered in space numerical scheme which employs the addition of artificial viscosity for shock capture. This replaces the need to solve a Riemann problem at each cell face. The general expression to forward the solution one step in time using a first order Forward Euler time marching scheme is given below:

$$U_i^{n+1} = \frac{1}{2} (U_{i+1}^n + U_{i-1}^n) - \frac{\lambda}{2} (f(U_{i+1}^n) - f(U_{i-1}^n))$$

Results are shown for the solution of the Sod’s shock tube case at time $t = 0.01s$. A detailed explanation of the test case is given in Section 4.3. A total of 50 grid points is used. CFL number is taken as 0.55. The domain is similar to that given in Section 4.3. Figures 80-83 show the result for the Lax-Friedrichs scheme. The method exhibits considerable smearing and dissipation, as well as a number of odd-even plateaus.

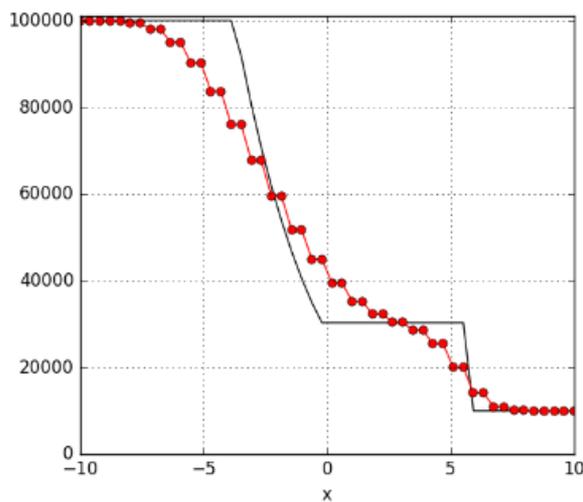


Figure 80: Pressure distribution

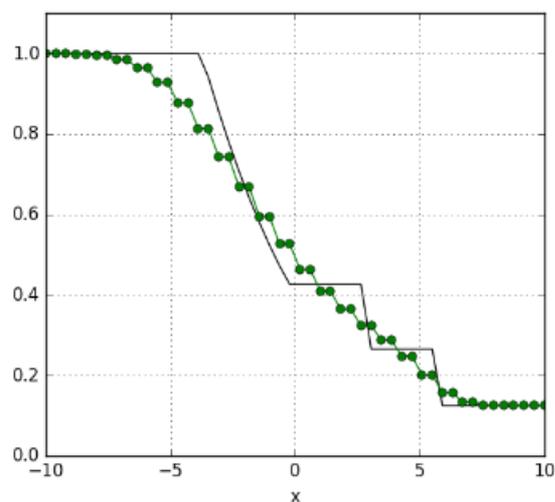


Figure 81: Density distribution

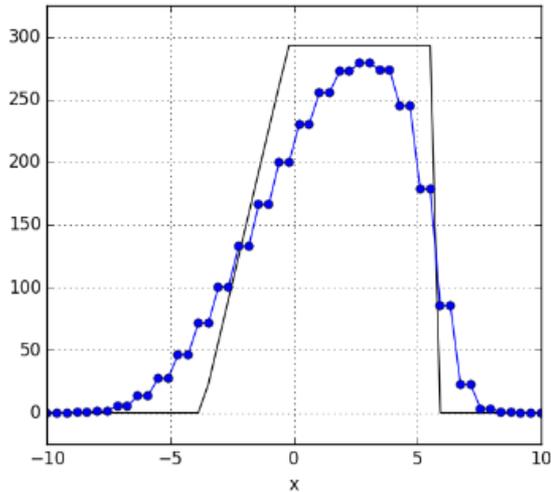


Figure 82: Velocity distribution

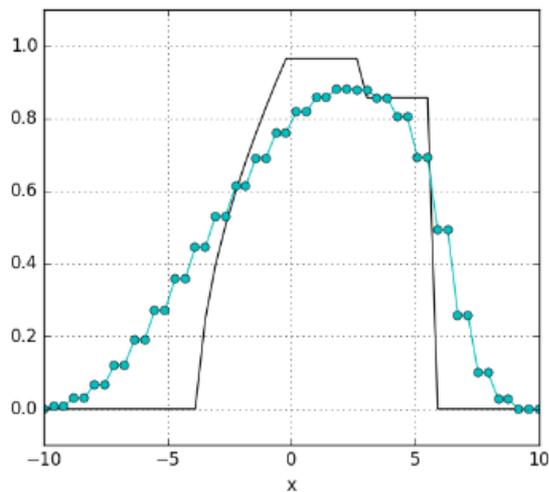


Figure 83: Mach number

A.1.2 Lax Wendroff Methods:

The Lax-Wendroff method for the Euler equations can be represented by,

$$U_i^{n+1} = U_i^n - \frac{\lambda}{2} (f(U_{i+1}^n) - f(U_{i-1}^n)) + \frac{\lambda^2}{2} \left[A_{i+\frac{1}{2}}^n (f(U_{i+1}^n) - f(U_i^n)) - A_{i-\frac{1}{2}}^n (f(U_i^n) - f(U_{i-1}^n)) \right]$$

There are many possible definitions for $A_{i+\frac{1}{2}}^n$. For example,

$$A_{i+\frac{1}{2}}^n = A \left(\frac{U_{i+1}^n + U_i^n}{2} \right)$$

Another possibility is the Roe-average Jacobian matrix. Thus the vector version of Lax-Wendroff method is actually an entire class of methods that differ only in their choice of $A_{i+\frac{1}{2}}^n$.

However, forming the average Jacobian matrices and the vector-matrix multiples are major expenses, to the extent that the Lax-Wendroff method may become uncompetitive with other methods that do not require such computations. A cheaper version of the Lax-Wendroff methods is a two-step variant that completely avoids Jacobian matrices called the Richtmeyer method. It is defined as follows:

$$U_{i+1/2}^{n+1/2} = \frac{1}{2} (U_{i+1}^n + U_i^n) - \frac{\lambda}{2} (f(U_{i+1}^n) - f(U_i^n))$$

$$U_i^{n+1} = U_i^n - \frac{\lambda}{2} (f(U_{i+1/2}^{n+1/2}) - f(U_{i-1/2}^{n+1/2}))$$

The first step is called the predictor. In this case, the predictor is the Lax-Friedrichs method. The second step, is the corrector. The Richtmeyer method uses two different grids- the standard grid x_i and the staggered grid $x_{i+1/2} = (x_{i+1} + x_i)/2$. The predictor maps the standard

grid to the staggered grid, and the corrector maps that staggered grid back to the standard grid.

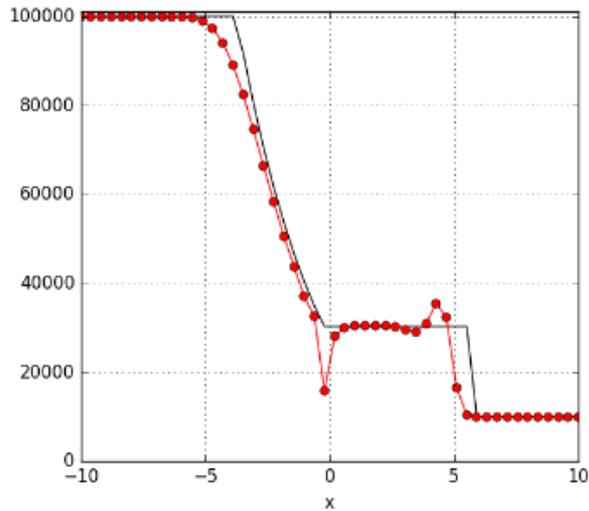


Figure 84: Pressure distribution

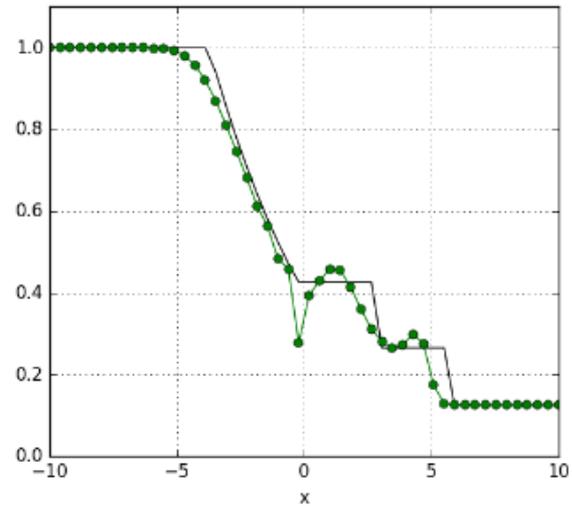


Figure 85: Density distribution

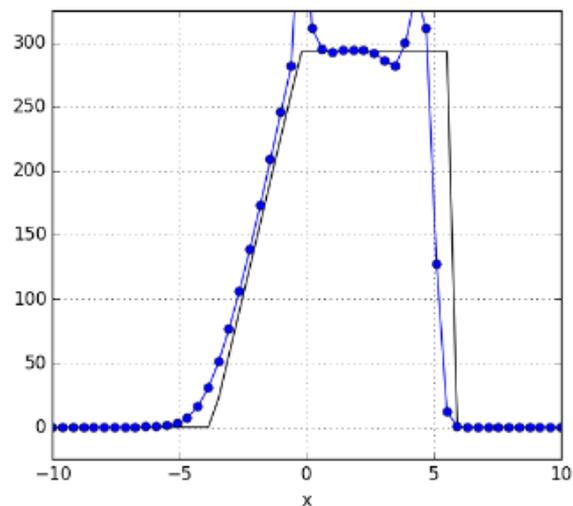


Figure 86: Velocity distribution

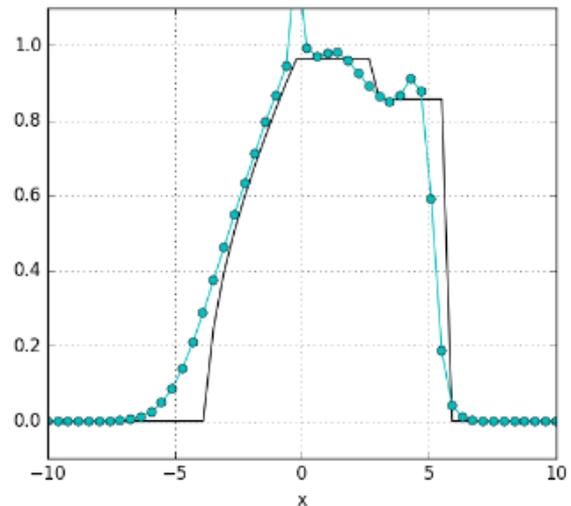


Figure 87: Mach number distribution

The Richtmeyer method has difficulties with both the shocks and the discontinuity as is evident from the oscillations. The Lax-Wendroff type methods were not designed for test cases for Euler equations with strong shocks or discontinuities. Hence unless the code includes an additional viscosity, the presence of oscillations is characteristic of such schemes.

A.2.1 Wave approach: Flux Vector Splitting

A.2.1.1 Van-Leers Flux split first order upwind method

Van Leer bases his flux vector splitting on Mach number splitting to help address sonic points. Sonic points are natural flux splitting points; at the very least sonic points require special consideration to avoid numerical problems. The flux vector for Euler equations can be written in terms of the Mach number as follows:

$$f = \begin{bmatrix} f_1 \\ f_2 \\ f_3 \end{bmatrix} = \begin{bmatrix} \rho a M \\ \frac{\rho a^2}{\gamma} (\gamma M^2 + 1) \\ \rho a^3 M \left(\frac{1}{2} M^2 + \frac{1}{\gamma - 1} \right) \end{bmatrix}$$

The flux splitting is given by

$$f^\pm = \pm \frac{\rho a}{4} (M \pm 1)^2 \begin{bmatrix} \frac{1}{(\gamma - 1)u \pm 2a} \\ \gamma \\ \frac{((\gamma - 1)u \pm 2a)^2}{2(\gamma + 1)(\gamma - 1)} \end{bmatrix}$$

for $|M| < 1$. Otherwise $f^\pm = f$ and $f^\mp = 0$ for $|M| \geq 1$, and $f^- = f$ and $f^+ = 0$ for $|M| \leq 1$. Van Leer's flux vector splitting correctly attributes all of the flux to right-running waves for right-running supersonic flow and all of the flux to left-running waves for left-running supersonic flow. Furthermore, in the original paper, Van Leer proves that his splitting satisfies $\frac{df^+}{du} \geq 0$ and $\frac{df^-}{du} \leq 0$, as required.

A first order upwind method using flux splitting can be expressed as:

$$U_i^{n+1} = U_i^n - \lambda (f^+(U_i^n) - f^+(U_{i-1}^n) + f^-(U_{i+1}^n) - f^-(U_i^n))$$

The results are given in Figures 88-91. The absence of spurious oscillations are a striking feature of the results. Even though the shocks are smeared, the results are still of good quality.

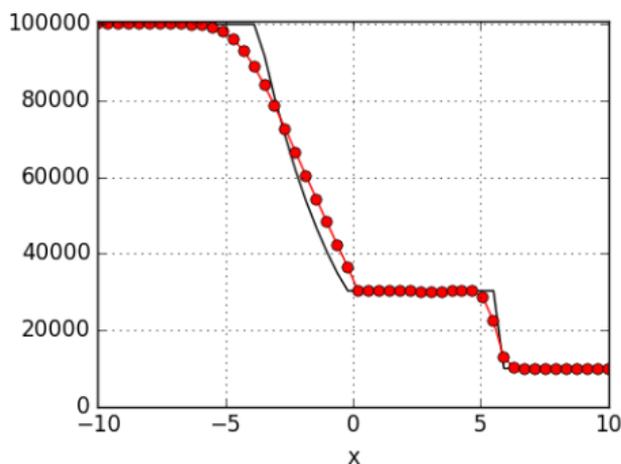


Figure 88: Pressure distribution

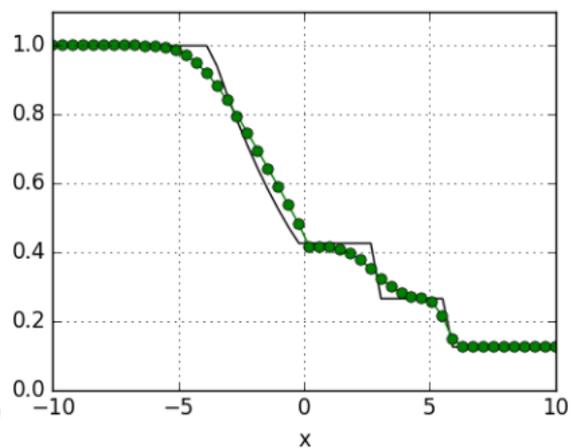


Figure 89: Density distribution

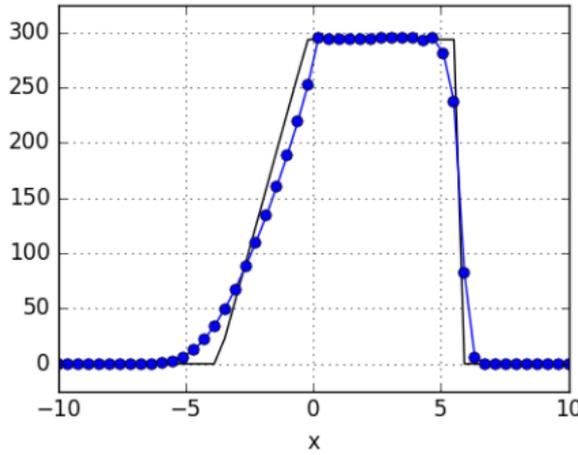


Figure 90: Velocity distribution

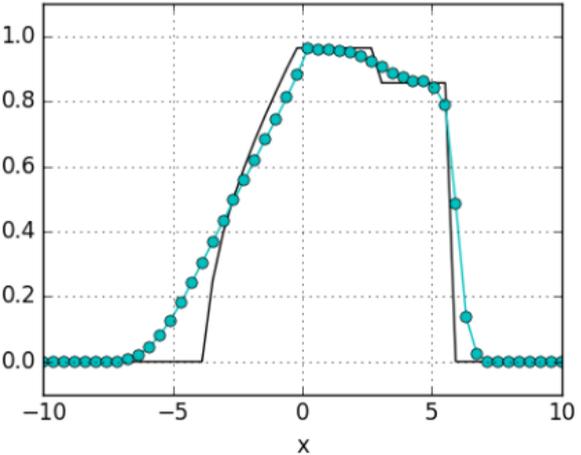


Figure 91: Mach number distribution

A.2.2 Wave approach: Reconstruction-evolution

A.2.2.1 Kurganov-Tadmor scheme with MUSCL reconstruction

The Kurganov-Tadmor scheme is a second-order high-resolution central scheme that uses MUSCL reconstruction (Kurganov & Tadmor, 2000). Since it is a central scheme, it is Riemann solver free. One of the main advantages of the scheme is that it is a fully discrete scheme which is a second order modification to the Lax-Friedrichs scheme. The second order scheme can be written in the conservative form (semi-discrete),

$$\frac{dU_i(t)}{dx} = - \frac{F_{i+\frac{1}{2}}(t) - F_{i-\frac{1}{2}}(t)}{\Delta x}$$

And the numerical flux H can be described as below:

$$F_{i+1/2} = \frac{1}{2} \left\{ \left[f \left(U_{i+\frac{1}{2}}^R \right) + f \left(U_{i+\frac{1}{2}}^L \right) \right] - a_{i+\frac{1}{2}} \left[U_{i+\frac{1}{2}}^R - U_{i+\frac{1}{2}}^L \right] \right\}$$

Where the local propagation speed $a_{i+\frac{1}{2}}$ is the maximum absolute value of the eigenvalue of the Jacobian of f over the cells i and $i \pm 1$, ie,

$$a_{i+\frac{1}{2}}(t) = \max \left[\rho \left(\frac{\partial f \left(U_{i+\frac{1}{2}}^L(t) \right)}{\partial U} \right), \rho \left(\frac{\partial f \left(U_{i+\frac{1}{2}}^R(t) \right)}{\partial U} \right) \right]$$

Where ρ is the spectral radius of $\frac{\partial f(U(t))}{\partial U}$. The intermediate values are given by,

$$U_{i+\frac{1}{2}}^L = U_i + \frac{1}{2} \Phi(r_i)(U_i - U_{i-1})$$

$$U_{i+\frac{1}{2}}^R = U_{i+1} - \frac{1}{2} \Phi(r_{i+1})(U_{i+1} - U_i)$$

Where $\Phi(r_i)$ is a slope limiter that limits the slope of the linear reconstruction to make sure that the solution is TVD. The slope limiter used is the van Albada limiter given by,

$$\Phi(r_i) = \frac{2r}{r^2 + 1}$$

$$r_i = \frac{U_{i+1} - U_i}{U_i - U_{i-1}}$$

The simulation is carried out on a grid of 50 points to ensure the comparison with the previous first order schemes is consistent. The CFL number is kept low at 0.1. A fourth order Runge-Kutta scheme is used for time integration to forward the solution in time. The results are shown in figures 92-94. The results show how effective the scheme is in solving the Euler equations without introducing any spurious oscillations.

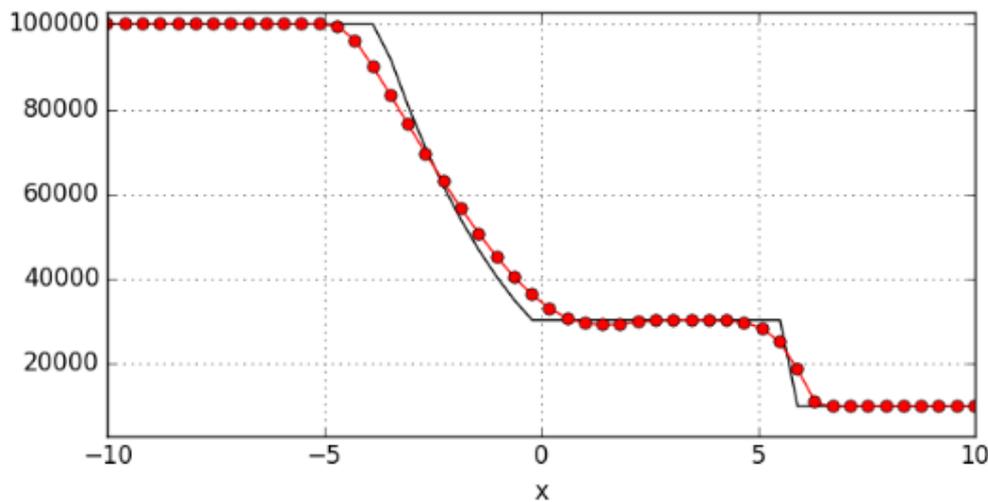


Figure 92: Pressure distribution

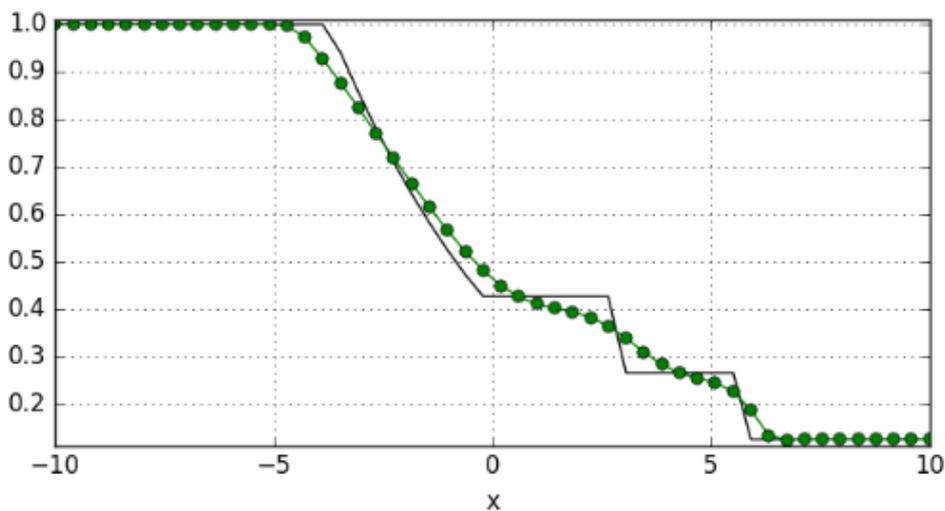


Figure 93: Density distribution

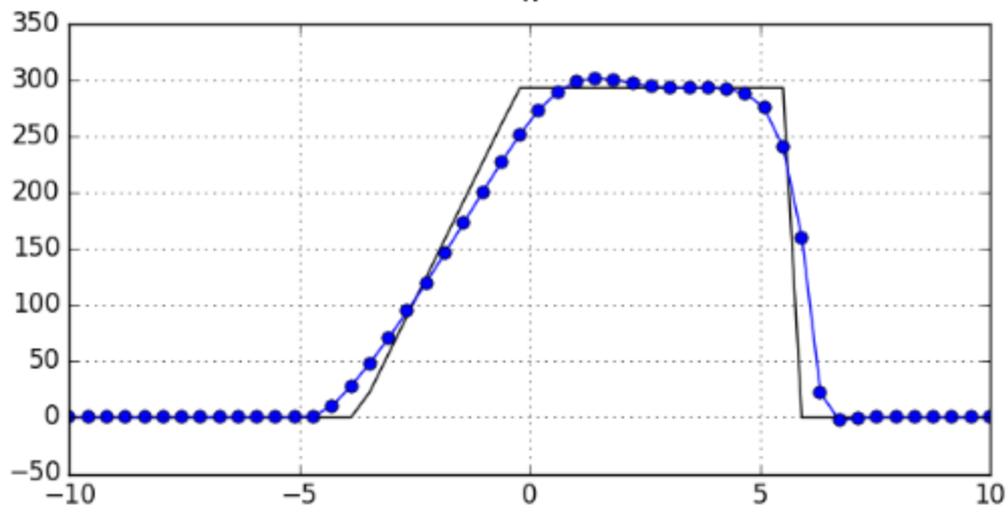


Figure 94: Velocity distribution