HIGH-ORDER SCHEMES

FOR COMPRESSIBLE VISCOUS FLOWS

SHYAM SUNDAR DHANABALAN

(M.Eng., NUS)

A THESIS SUBMITTED

FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

DEPARTMENT OF MECHANICAL ENGINEERING

NATIONAL UNIVERSITY OF SINGAPORE

2011

Acknowledgements

I wish to express my deepest gratitude to my supervisor, A/Professor Khoon Seng Yeo, for his invaluable guidance, patience, supervision and support throughout the study. His guidance helped me all through my research and writing of this thesis.

I would also like to express my appreciation to the National University of Singapore (NUS) for providing tuition fee waiver during the course of my study. In addition, I would like to thank the super computing facilities at NUS, which I had used extensively for carrying out my research works.

My sincere thanks also goes to Dr. Neelakantam Venkatarayalu for his valuable insights during the various discussions we had.

I would like to thank all of my friends and family members, especially my mom Sundrambal Baby, my beloved fiancee Vasuki Ranjani, my sister Karthigha and her husband Senthil for being alongside me at all times.

Contents

Summary

This thesis presents development of a set of high order, high resolution numerical methods for efficient solution of inviscid and viscous compressible flows. Numerical methods are developed by considering finite information propagation within the elements over a give time integration step. Spatial discretization schemes of up to 5^{th} order accuracy have been developed and successfully tested for unstructured grids. The concept is also extended for computing the solution gradients at the element interface, thereby providing the framework to perform viscous computations. Numerical experiments demonstrate that the proposed schemes can reproduce higher order characteristics in all the cases. Based on the experience in development of these spatial discretization schemes, a higher order adaptive time stepping algorithm is formulated. The proposed algorithm is simple, efficient to implement and has a significant reduction in computational cost.

A preliminary investigation was conducted for influence of solution at element boundaries in the presence of shocks. The analysis highlighted that the onset of spurious oscillations indeed occur at the element boundary while the internal solutions still remain smooth. This behavior of numerical schemes was exploited to formulate a high resolution WENO shock capturing scheme. Adaptive methods are formulated to selectively apply the costly WENO procedure only for those elements with high solution oscillations. The high resolution property of the new WENO scheme is demonstrated with various examples involving inviscid shock interactions.

The entire set of numerical methods developed in this work are tested on viscous compressible flow problems involving aero-acoustic sound generation. In all cases, the results were comparable to the existing experimental results, thus demonstrating the applicability of the proposed scheme in simulating complex non-linear flow problems involving shock interactions, aerodynamic sound generation, etc..

List of Tables

List of Figures

Nomenclature

(ξ, η)	Local coordinates of triangle
$[\cdot]_\infty$	Free stream quantity
χ	Solution oscillation detector
Δt	Time step size
δ	Kronecker delta function
ℓ_n	Size of neighboring elements in WENO stencil
ϵ	Corrective term in solution evolution
γ	Specific heat ratio
$\hat{F}$	Riemann flux at interface
λ_w	Bias used for central stencil in WENO reconstruction
$\mathbb{C}$	Courant-Friendrichs-Lewy (CFL) number
I	Pressure
$\mathcal{C}$	Desired CFL number for ExRi formulation
$\mathcal{F}$	Riemann flux correction
$\mathcal{M}_R$	Coefficient matrix in solution evolution
$\mathcal{N}$	Ratio of time step size between mesh blocks
$\mathcal{O}(Q)$	Oscillation of solution Q

$\mathcal{R}$	Relative solution oscillation in element
$\mathcal{S}$	Number of time steps to defer synchronization of ghost cells
$\mathcal{U}_S$	Reconstructed stencil solution
Ω	Element under consideration
$\partial\Omega$	Boundary of element Ω
ψ	Arbitrary trial function used in Galerkin formulation
ρ	Density
σ	Absorption coefficient used at absorbing boundaries
$\tilde{F}$	Approximate continuous flux
$\vec{n}$	Normal vector on surface
ζ	Local coordinate at element boundary
a	Characteristic wave speeds
B	Mesh block
B_{cf}	Fine layer of block B_c
B_{fc}	Coarse layer of block B_f
c	Speed of sound
$C_{m,k}$	k^{th} coeffient of m-stage Runge Kutta scheme
e	Internal energy of fluid
$F(Q)$	Flux corresponding to solution Q
$f^w(Q)$	Function used in detection of solution oscillation
F^x, F^y	Flux along x and y directions
G	Amplification matrix (for matrix stability analysis)

h Element size

$H(Q_L, Q_R)$ Riemann solver at interface

N_b Number of basis functions

n_K Number of stencils used in WENO reconstruction

n_w Exponent used in determining WENO weights

p Degree of approximation of solution

Q Solution variable

S Element interface (belonging to element boundary)

s Wave speed estimates used in Riemann flux formulation

t Time variable

u_i Velocity along direction x_i

V Volume of element

w_K Stencil weight associated with stencil K

Chapter 1

Introduction

Fluid dynamics is governed by a set of non linear equations that admit a wide range of physical phenomena involving multiple scales. The early work of Claude-Louis Navier and George Gabriel Stokes on the inclusion of viscous terms led to currently widely known equation set representing the fluid dynamics. Based on their contributions, these equations were named as the Navier-Stokes (NS) equation. The inherent complexities of these equations makes it extremely difficult for obtaining analytical solutions for even a slightly complicated configuration. Before the invention of computers, numerical solution of NS flow problem was virtually impossible. With the invention of computers, this enormous task of numerical computations was allocated to computers, resulting in some of the earliest known Computational Fluid Dynamics (CFD) simulations.

The daunting task of CFD has always been to make a compromise between the approximations on representation of fluid flow and the available computational power. The most accurate method to solve a fluid flow problem would be Direct Numerical Simulation (DNS), in which, the NS equations are solved in its exact form without any approximations. A DNS computation is accurate only if it can capture and represent all possible scales in the fluid flow being simulated. As the dynamics of flow intensifies, the length scales become much smaller, resulting in higher grid resolution requirements for DNS. This makes the DNS computations impractical for simulating real life flows. The limited computational power results

in under-resolution of the smaller length scales. These unresolved length scales are represented using appropriate fluid flow models. Some of them in the order of increasing computational cost are: Reynold's Averaged NS (RANS) equation models, Detached Eddy Simulation (DES) models and Large Eddy Simulation (LES) models. In spite of RANS models having the lowest computational cost requirement, the first 3D RANS based computation of a complete aircraft was computed only in the year of 2007[4].

With the steady increase in computational power, the current trend is shifting towards the application of LES and DES methods for practical flow problems that are predominantly unsteady in nature. These schemes rely mainly on the resolution capabilities of the numerical scheme rather than the approximation of underlying equations as in the case of RANS simulations. Recent years has witnessed an increasing trend in application of various numerical schemes to LES methods [5, 6, 7, 8, 9, 10, 11, 12] with applications on multi-scale problems involving turbulence, shocks and acoustics etc. Due to the increased application of methods such as LES, DES and DNS, the demand for numerical accuracy also increases. A quick analysis of the trend of numerical schemes shows a great interest in the popularity of higher order numerical schemes in recent years. However, the higher order schemes still face the common hurdles of a CFD simulation namely stricter stability condition, time step restriction and the handling of solution discontinuities.

There is still a huge demand for unified, efficient numerical schemes to compute complex multi-scale solutions with acceptable accuracy. The current work is hugely motivated by this requirement of high resolution schemes for the solution of NS equations.

1.1 Background

1.1.1 Historical developments in CFD

Complex flows consisting of vortex interactions, propagation of vortex structures, acoustic sound generation etc. involve convection and interaction of the flow structures in large domains over prolonged periods of time. The numerical schemes approximating these flows need to represent the evolution of the flow structures with adequate accuracy. This has been the unending task of researchers in the field of CFD. The major portion of any CFD tool comprises of techniques for spatial and temporal approximations.

The earliest known CFD computation was accomplished using finite difference (FD) method in early 1940s. Since then, various techniques were developed that changed the course of applied CFD. One of the significant milestones was achieved by Mac Cormack (1969) [13] with his development of the predictor-corrector scheme along with the artificial viscosity term to handle flows with strong gradients. The simplicity and effectiveness of the scheme led to wide spread application of the scheme to CFD and has been taught to students even now. However, one of the important problem in CFD is the handling of transonic flows. Based on the domain of dependence of the solution at a given point, the governing equations of fluid dynamics can exhibit elliptic, parabolic and hyperbolic characteristics depending on whether the flow is subsonic, transonic or supersonic. Murman and Cole (1971) [14] handled this problem by using a mixed elliptic-hyperbolic equation to describe the fluid flow. The hyperbolic and elliptic regions are solved using different finite difference techniques based on windward and central differencing. The significance of the method is that it is the first method to use the physical wave propagation characteristics to accurately model the underlying flow behaviour.

The shock capturing methods were further enhanced with the well known contribution to the shock capturing methods by pioneer Jameson with the development of Jameson Schmidt Turkel (JST) [15] scheme for the Euler equations. The JST

scheme used Runge Kutta time stepping method and a mix of second and fourth order differences to control oscillations and to provide dissipation. Jameson showed that the scheme can consistently converge to a steady state solution for a wide range of problems.

Application of early CFD tools to practical problems were limited by the available computational power in terms of speed and memory. In the early 1970s, the first three dimensional computation of a hypersonic shock-boundary-layer interaction was carried out using the Mac Cormack scheme. The domain was adapted to capture only the relevant physics and resulted in a pyramidal mesh system of $(8 \times 32 \times 36)$ bounded by a flat-plate and a wedge. Though the total number of points was a mere 9216, it consumed nearly all of the available computational memory at that time.

As the computational power advanced to new levels, the field of CFD witnessed significant improvement in computational techniques and application to large scale problems. The quest for highly efficient numerical schemes was actively undertaken by the highly creative CFD community. Various solution acceleration techniques were developed on various fronts to reduce the computational cost. The Alternating Direction Implicit (ADI) schemes [16] were applied to Navier Stokes equation in 1971 by Briley et. al. [17] for solution of incompressible Navier Stokes equation. Jameson implemented the popular "Dual time stepping" method [18] in which the intermediate equation arising from the implicit formulation is solved by introducing a pseudo-time marching scheme using multi-grid acceleration.

During 1970s, almost all of the CFD solvers were written for structured grids. The use of structured grids results in simplified solvers and easier optimization for computational speed. Geometric flexibility is achieved by using multi-block grids in which the domain is divided into simple structured blocks. Though the schemes are computationally efficient, topological restrictions on the grids makes it increasingly difficult to define complex geometries and often results in unnecessary points due to clustering of grid points at a particular region. As a result, an undesirably

large amount of time is devoted to grid generation and problem setup compared to the actual solution time. With CFD being increasingly applied to very large scale problems involving complex geometries, the topological restrictions of structured grids became a huge bottle neck in arriving at solutions. As an example, the first F-16A aircraft simulation took 11 months to generate the three-dimensional grids while it took just 3 months to run the actual simulation. The Finite Volume (FV) schemes are a class of numerical methods which are constructed by applying the integral form of the NS equations on the control volume. With the application of Gauss divergence theorem, the volume integral of the spatial terms are conveniently represented in terms of the integrals of surface fluxes. Due to the piecewise nature of the FV formulation, the solution states and hence their corresponding fluxes could be discontinuous at the element boundary. With the availability of solution states at either side of the boundary, an exact or approximate Riemann solver [19, 20, 21, 22] is used to obtain unique solution states and fluxes at the boundary. The resulting unique fluxes at the element boundaries are used to evaluate the surface integrals and thus provide the closure for formulation of FV schemes. The second order FV schemes traditionally enjoyed great popularity because of the involvement of only immediate neighboring elements for flux reconstruction. With its compact stencil and the conservative integral formulation, the second order FV schemes were readily extended to unstructured meshes [23, 24, 25, 26, 27, 28] in the early 1980s.

1.1.2 Recent developments in spatial schemes

The unstructured second order FV schemes dominated the field of CFD for over two decades. Their small memory footprint and relative simplicity enabled various solvers to adopt the scheme readily. The schemes are also stable and can handle shocks with the help of suitable limiters. In spite of these advantages, they suffer from an important property, characteristic of lower order schemes. These schemes often require very large number of grid points to achieve grid-independent solutions.

In other words, the convergence of the numerical solution to the actual solution is very low for lower order schemes. This is specifically the case for Direct Numerical Simulation (DNS), where the accuracy of the schemes play an important role in the solution outcome.

Higher order schemes, have higher rate of convergence of the numerical solution to the actual solution as the grid is progressively refined. This property enables the higher order schemes to achieve an accurate solution in a relatively coarse grid. Since the higher order schemes involve additional terms in solution/function approximation, the computational cost can be significantly larger than the lower order counterparts for the same number of volumes or elements. However, this difference in the computational cost is compensated by the grid-convergence property of the higher order scheme. If we compare the accuracy of the solution with respect to the net computational time taken, it will be obvious that the higher order schemes indeed has significant advantages over lower order schemes.

The higher order schemes, with their obvious advantage of having lower dispersive and diffusive errors and a higher rate of grid convergence, have been extensively developed and applied to various flow problems in the last decade. The initial development of higher order schemes were focused on structured grids [29, 30, 31, 32, 33]. The difficulty in structured grid generation is circumvented to some extent by overset schemes [8, 34, 35] which allow non-conforming grids to overlap with each other. Due to the interpolation involved between the overlapping grids, these methods have difficulties in extension to high orders, especially in presence of shocks and discontinuities in the solution.

The Finite Element (FE) community for long has adopted the unstructured meshes. FE schemes were primarily developed to solve linear problems involved in Structural analysis, Computational ElectroMagnetics (CEM) etc.. The fundamental equations for CFD, on other hand, are non-linear in nature. The appearance of discontinuities in flow solutions further complicate the FE spatial discretizations, which were initially formulated for smooth solutions. The Spectral Element (SE)

[36, 37] method, which is similar to FE method, were applied for DNS and LES problems due to their very high resolution property. But, similar to FE methods, the SE methods also require the solution to be smooth. A few attempts [38, 39] have been made to use SE method for computing solutions with discontinuities with limited success.

The FV methods can inherently support the discontinuous solutions across the element. In spite of their advantage over the FE methods in treating the discontinuous solutions, the FV schemes suffer from their difficulty when extending to higher order schemes. Since a typical FV scheme stores only the average cell values, it becomes increasingly complicated to interpolate the flux and solution values to high order of accuracy. These constraints often limit the application of unstructured FV schemes to second order. It is due to this complexity that the unstructured FV schemes formulated using higher order flux reconstructions (more than 3^{rd} order) are less common in the literature[40, 41, 42, 43].

Over the last few years, the CFD community has witnessed the development of various methods that can be easily extended to very high orders and at the same time have flexibility in representation of complex geometries. Schemes that have a compact stencil (depend only on immediate neighbors), are increasingly favored due to the ease of implementation and parallelization. The Discontinuous Galerkin (DG) method [44, 45] is one such method, which has combined the inherent advantages of both the FV and the FE methods. Similar to the FV method, DG schemes can support discontinuous solutions at the element interface and is conservative in nature. At the same time, akin to FE methods, the DG schemes can be easily extended to higher orders by changing their basis function set to represent the solutions. Schemes with similar properties such as Spectral Difference (SD) method [46], Spectral Volume (SV) Method [47] etc. have also been developed for CFD. The Spectral Difference method uses a set of solution and flux points to interpolate the fluxes and solution inside an element. The Spectral Volume method on the other hand, divides the element space into sub-cells. These sub-cells are

used to interpolate the fluxes required for solution evolution. In both the SD and SV methods, the fluxes at element boundaries are formulated with the appropriate Riemann fluxes.

With their unstructured nature, schemes like DG, SD and SV automatically support the h-adaptation, which can be achieved in a variety of ways. Also, since these schemes depend only on the Riemann solutions at the boundaries, they can support different orders of accuracy between the neighboring elements. This provides a mean to refine the local order of accuracy with respect to the variation of solution, which is commonly denoted as p-refinement. Various applications of the localized h and p adaptation methods can be found in the literature [48, 49, 50]. Due to their geometric flexibility, hp adaptivity, ease of mesh generation and simple parallelization properties, it is expected that the higher order unstructured schemes such as DG, SD and SV methods will eventually replace the traditional lower order methods and schemes based on structured grids [4].

1.1.3 Efficient time stepping schemes

Even when the spatial resolution is improved, the numerical schemes for CFD face further hurdles in terms of temporal resolution. In a spatio-temporal simulation, the errors generated in spatial and/or temporal discretization can increase exponentially and corrupt the solution. Thus, a scheme with very high spatial accuracy can produce accurate results only when it has a temporal scheme of adequate accuracy. The most common method of temporal discretization is the explicit time discretization in which the temporal accuracy is obtained by evaluating a series of backward Euler steps. Among many such schemes, the Runge Kutta schemes enjoy high popularity due to their good stability conditions and ease of implementation. The conventional Runge Kutta time stepping schemes are restricted by the stability condition given by the Courant Friedrichs Lewy (CFL) condition. The CFL condition relates the maximum time step size for a numerical scheme to the element size. The maximum allowable time step size is directly proportional to the

ratio of local element size and maximum wave speed in that region.

For complex configurations, optimizing available computational resources often involves the use of finer meshes at regions of interests while retaining a coarse mesh in the rest of the computational field. In the case of fluid dynamics, the resolution of flow field is of higher importance in areas such as shocks, contact discontinuities, boundary layers, flow separation etc.. Such requirement of mesh resolution are also present in other applications such as electromagnetics, heat conduction etc.. The presence of elements of smaller size results in increased computational cost. This is denoted as "grid induced stiffness". Various schemes have been developed in the last decade to avoid the problem of grid induced stiffness. These include the Implicit-Explicit [51, 52, 53, 54] schemes and the multi-rate schemes. The Implicit-Explicit schemes increase the global time step size by using implicit schemes for the computationally expensive stiff regions and use explicit schemes for other regions. The entire solution is marched with a constant time step size. However, for a multi-scale simulation, certain regions such as turbulent boundary layers, require adequate temporal resolution along with the spatial resolution. For such regions, a choice of large time step size would result in inaccurate representation of underlying physics. The primary advantage of the implicit schemes is hence not realized for such cases. The multi-rate schemes use adaptive time stepping methods that can support different time steps for different elements. Since they respect the local time step size restriction of each element, they result in an accurate representation of the time scale at stiff regions. One of the significant advantage of the multi-rate schemes is the ease of adaptation and parallelization due to their explicit nature. The multi-rate schemes can be categorized into local-time stepping schemes [55, 56, 57] and multi-time stepping schemes. The local-time stepping schemes are devised such that each element can have its own time step, independent of its neighbors. The Multi-Time Stepping schemes (MTS), on the other hand, divide the computational domain into blocks based on the element sizes. Each domain is then evolved in time with its local time step limit. Special treatment is provided

at the block-block interfaces in order to allow for different time steps across the interface. These schemes are particularly attractive as they require minimal change for implementation in the existing solvers. However, as pointed out in [58], the multi-time stepping schemes that exist in the literature are limited to 2^{nd} order schemes.

The MTS schemes can further be classified into two categories based on the treatment of solution evolution at block-block interfaces. In order to support different time steps across the block boundaries, the schemes can either use a specialized time stepping scheme at the block boundary [59, 60, 61] or use a layer of supporting elements at the block boundaries [62, 63]. The latter is commonly termed as domain splitting or domain decomposition methods. Lohner et. al. [62] initially proposed one of the earliest multi-rate scheme using a domain splitting method for solving hyperbolic equations. In his work, he used adaptive time steps for two different domains while using an interpolation region to perform solution updates between each block. One of the key point mentioned in [62] is the maximum speed of information (or error) propagation in a hyperbolic system. The overlapping region of the domains are chosen such that the errors at overlapping regions do not propagate into the domain interiors. Later, van der Ven et. al. [63] implemented a similar scheme for curvilinear grids. He named the scheme as multi-time stepping scheme. Though these schemes have high potential in terms of simplicity and ease of implementation, they were demonstrated only on second order finite difference schemes. Chauviere et. al. [64] proposed an MTS scheme for higher orders where the entire mesh is marched with the desired time step and the unstable (fine mesh) regions are later corrected using the available solution in the coarse mesh. The drawback of this method is the requirement to compute the entire solution more than once for a single time step. In all the domain splitting schemes, the influence of the boundary conditions at the overlapping regions and their effect on stability and accuracy were not studied in detail. The applicability of the multi-rate schemes for higher order schemes (such as DG) is an interesting

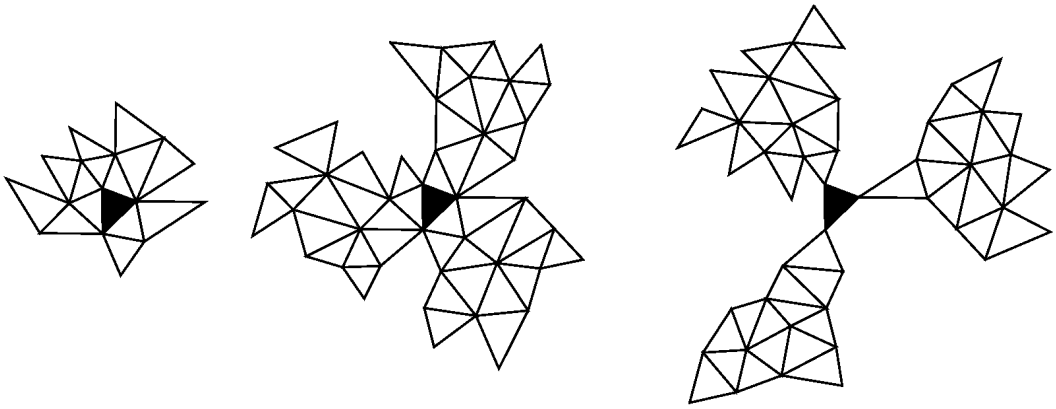


Figure 1.1: Reconstruction stencils for FV WENO schemes on unstructured triangular meshes.

topic to be explored. The primary interest arise due to its various advantages such as ease of implementation and independence of the adaptive time stepping scheme on the nature of spatial and temporal discretizations.

1.1.4 Accurate shock capturing schemes

A major step in formulating any numerical scheme for CFD is the handling of discontinuities arising in the flow solution. The lower order schemes, due to their diffusive nature, can better handle the solution discontinuities compared to the higher order schemes. The discontinuities themselves are first order in nature as the solution gradients across the discontinuities are undefined. However, the discontinuities often affect the accuracy of the solution in their vicinity rather than just at the discontinuities themselves. The higher order schemes can exhibit unphysical oscillations similar to that of the Gibb's phenomenon [65, 66] in the vicinity of these discontinuities. These oscillations can be further aggravated at element boundaries. In many cases, these oscillations can cause the solution to become unstable and hence eventually diverge.

Various schemes have been developed to handle these discontinuities in the flow solution. These schemes have two common tasks: (i) identification of discontinuities (ii) processing the flow solution to handle oscillations. The earliest shock capturing schemes were proposed by Von Neumann and Richtmyer in 1950 [67] in which an

artificial viscous term was added to the original set of equations. In this method, the introduction of artificial viscosity eliminates the spurious oscillations generated at regions of high gradients. Another common technique is to use limiters at the shock region, thereby reducing the order of accuracy at the region near discontinuities[68]. These schemes are often restricted to second order accuracy and in some cases may reach third order accuracy. Note that the order of accuracy in this context pertains to the accuracy near the shock rather than accuracy of the shock resolution themselves.

In the late 1990s, a new class of shock capturing scheme started to emerge. These schemes reconstruct a smooth solution of given order of accuracy even in the vicinity of discontinuities. This is achieved by sampling various realizations of the solution and selecting the smoothest solutions. The Essentially Non Oscillatory (ENO) and Weighted ENO (WENO) [69, 70] schemes are belong to this category. While the ENO schemes choose the smoothest solution with a set of available stencils, the WENO schemes reconstruct the solution as a linear combination of all the stencil solutions weighted by a normalized smoothness factor. The overall order of accuracy of the solution is hence preserved in these shock capturing schemes. Due to the low numerical diffusion involved in these schemes, they are suitable for multi-scale problems such as aero-acoustics. The WENO schemes have been developed and tested for structured/unstructured Finite Volume (FV) [70, 41] and DG [71] schemes.

The FV WENO formulation [71, 41] derives the smooth solution by reconstructing a smooth solution polynomial from the integral solution values of the neighboring cells. Fig. 1.1 shows some reconstruction stencils used in a FV WENO scheme. The stencil sizes increase with increase in order of the scheme. Unlike the traditional FV method which uses the average value of the solutions within the cell, the high resolution schemes such as DG, SV, SD etc. employ higher order functional representation for the solution within the cells. For such schemes, the FV WENO formulation would result in loss of resolution as the higher order infor-

mation within the cell may be lost in the process of reconstruction. This renders the reconstruction to be more diffusive since the finer structures within the cells can not be resolved. Rather than using only the cell average values, it is possible to utilize the higher order solution available within the cells to reconstruct a smooth solution. Such reconstruction stencils can be formed with much fewer elements compared to the FV-WENO stencils. The Hermite WENO reconstruction [72] proposed by Luo et. al. and the hierarchical reconstruction presented by Shu. et. al. [1] focus on utilizing the higher order solution terms within the element. The stencils in these cases are formed with only the immediate neighbors of the element, thus utilizing a maximum of 3+1=4 elements for higher order reconstruction over the 2D triangle elements. Since the size of the stencils does not increase with the order of the elements, the higher order sub-cell resolution of the flow structures are preserved. The success of these compact WENO schemes depend on their ability to address the solution oscillation within the elements. For schemes of very high orders ($\geq 5^{th}$ order), these sub-cell oscillations become more significant and influence the stability of the scheme to a greater extent.

1.2 Motivation

Any numerical scheme used to solve a hyperbolic equation system, is concerned with propagation of waves. A close examination of the numerical schemes can be conducted by examining the propagation of information within the discrete elements. This could then provide us with valuable information for further improvement on the numerical schemes in the context of efficiency, accuracy and stability. One of the major motivations for this work is to explore the possibility of developing a new numerical method by analyzing their physical wave propagation characteristics. Some of the notable areas of interest are listed below:

1.2.1 Influence of discontinuous solutions at element interface

The behaviour of numerical schemes are best understood by analyzing their solution evolution at each explicit time step. At each time step, the numerical schemes evolve their internal solutions with the information within the element and the information obtained from the neighboring elements. While the internal solution is readily available, the neighboring solutions can be discontinuous at the element interfaces. A Riemann solver is then used at these interfaces to obtain the solution. This forms the basis of temporal evolution in schemes such as DG, SD, SV etc. The Riemann solution at the interface need not match the internal solution of the element. In such cases, this difference can be regarded as an additional information that will be propagating from the element interface into the element interior. Due to the finite time step size and the wave propagation speed, this information can influence only a finite region within the element interior. This influence has not been taken into account in the existing numerical schemes. It is possible to obtain approximations to the influence of the Riemann solutions over the element internal solution. It will be interesting to study the characteristics of the numerical schemes that consider the influence of Riemann solution on the internal solution.

1.2.2 High resolution shock capturing schemes

Given a smooth initial solution, our numerical experiments suggest that the onset of sub-cell oscillations initially occur at the element boundaries. These oscillations are typically localized to the faces of elements, while the rest of the internal elements solution remain relatively smooth. This phenomena is also numerically confirmed in this work. This provides us with an alternative approach to formulate a WENO scheme. A reconstruction technique can be formulated to suppresses the oscillatory solutions arising at element boundary at each time step before it has the chance to propagate into the element interior. With this reconstruction technique, the majority of the element solution can be re-used during the reconstruction process,

resulting in very high resolution of shocks and associated flow structures.

1.2.3 Grid induced stiffness

Depending on problem requirements, the process of discretization can result in a mesh of varying density. For a normal Euler backward time stepping scheme, the maximum time step size is dependent on the smallest element in the mesh. In the context of implicit time stepping schemes, a smaller element size would result in a denser, stiffer matrix to invert. This increase in the stiffness of the problem due to the geometry and its corresponding discretization is called "grid induced stiffness".

The multi-time stepping schemes as described before, allow the time step size to vary across elements. Of the various methods developed before, one of the interesting method is the one proposed y Lohner et. al. [62]. In this method, he divided the computational domain into blocks and used a "buffer" layer for interpolating the values from one block to another. This buffer layer is updated at a specific time interval such that the error from it's boundary do not reach/contaminate the interior solution. This method, however, was restricted to second order schemes. By extending the same physical argument that the error in the element boundary can propagate only a finite distance within a time step, it could be possible to formulate a similar multi-time stepping method for higher order DG like schemes.

1.3 General plan of Research

This work is aimed at developing robust higher order numerical methods for accurate simulation of Navier Stokes equations. The different objectives of this work are listed as follows:

- Formulate a numerical scheme with element flux computation method accounting for the influence of the Riemann solution at the boundary on the element interior solution

- Extend the new flux formulation to represent the viscous terms in NS equations
- Develop a high order adaptive multi-time stepping scheme based on wave propagation characteristics of hyperbolic equations
- Formulate a compact WENO scheme for reconstruction of solutions in oscillatory regions by studying the onset of solution oscillations in a higher order scheme
- Demonstrate the advantages of higher order schemes with two applications in the direct computation of acoustic noise

1.4 Outline of Thesis

The thesis commences in Chapter 2 detailing the basic theory relevant to the numerical approximation of the Euler equations based on Discontinuous Galerkin (DG) spatial approximation and Runge Kutta (RK) temporal approximation. Here, the higher order convergence properties of the RKDG method is demonstrated for an inviscid compressible flow problem. Chapter 3 explores a new numerical solution evolution method by considering the influence of the Riemann solution and fluxes on the internal solution of the element. The wave propagation characteristics of the numerical scheme is studied in detail for the 1D schemes. The method is further extended to unstructured grids in two dimensions. Based on the proposed scheme, a one step method is formulated in Chapter 4 for computing the gradients required in computing viscous flows. All the proposed schemes are tested for order of accuracy. Chapter 5 details the development of higher order adaptive time stepping schemes based on the wave propagation characteristics of hyperbolic equations. The stability of the proposed scheme and the validity of its application is demonstrated with application to isentropic vortex evolution and cavity tone problems. Chapter 6 deals with one of the challenging problems of higher order methods: capturing

discontinuities arising in the flow field. A new adaptive WENO formulation is proposed for resolving a shock using a compact stencil and at the same time maintain the order of the scheme in the vicinity of the discontinuities. The scheme is also extended to three dimensional flows to demonstrate its applicability to real world problems. Chapter 7 demonstrates the application of the proposed higher order schemes in computing the acoustic tones of Cavity Tones and Reed problems. The effectiveness of the schemes are tested in terms of their ability to reproduce the acoustic tones arising from the flow. The thesis concludes with a summary and a list of possible future work in Chapter 8.

Chapter 2

Theoretical Background

In this chapter, we detail the numerical discretization process involved with computing solutions for scalar hyperbolic equations. We begin the discussion with the formulation of the widely used Runge Kutta Discontinuous Galerkin (RK-DG) schemes [44, 73, 74] with application to inviscid Euler equations. Since a major portion of the thesis involves development of new numerical methods for computation of inviscid and viscous flows, this chapter serves as an introduction to these numerical methods.

The chapter begins with the description of the hyperbolic equations and the process of discretization in space and time. Later, the spatial discretization technique using DG method is presented. A m-stage Runge Kutta time integration technique is then applied on the DG scheme to march the solution in time. The RK-DG scheme formulation is completed by defining the Riemann fluxes at the cell boundaries. The property of the RK-DG method is demonstrated on standard isentropic vortex evolution test problem and its grid convergence properties are examined.

2.1 A general Hyperbolic equation system

A hyperbolic equation governs the propagation of information in space and time. A general hyperbolic system is represented by

$$\frac{\partial Q}{\partial t} + \nabla \cdot \vec{F} = 0 \tag{2.1}$$

where Q is the solution and $\vec{F}$ is the corresponding flux vector. This equation system represents temporal evolution of solution Q with respect to the flux quantities $\vec{F}$. The hyperbolic system of equations in Eqn. (2.1) can also be written as

$$\frac{\partial Q}{\partial t} + \vec{a} \cdot \nabla Q = 0 \tag{2.2}$$

where $\vec{a}$ is the wave speed vector given by

$$\vec{a} = \frac{\partial \vec{F}}{\partial Q} \tag{2.3}$$

The hyperbolic equation given in Eqn. (2.2) provides various interesting insights:

1. The information travels at a finite wave speed.

2. The wave speeds can be directional. Or, in other words, the wave speeds can have different values along different spatial coordinates.

3. For a given space, a maximum wave speed can be estimated and hence the maximum distance a wave propagates within that space can be deduced.

The last information mentioned above is often used to fix the time step size of the numerical schemes. The maximum distance travelled by the wave over a given period is given by

$$d = a\Delta t \tag{2.4}$$

where d is the distance of propagation and a is the magnitude of the wave speed $\vec{a}$.

While solving the hyperbolic equation set, the given space is discretized into smaller elements. Within this space, the maximum distance of wave propagation can be estimated from Eqn. (2.4). For any numerical scheme to be physically accurate, the information cannot bypass an element. For example, consider a one

dimensional case with three adjacent elements of indices $i-1$, i and $i+1$ respectively. The information from element $i-1$ cannot travel to element $i+1$ without passing through element i. If the size of the element is given by h, this gives us the relation

$$h \geq a_{max}\Delta t \tag{2.5}$$

where a_{max} corresponds to the maximum wave speed within the given space. Eqn. (2.5) is required for ensuring continuous flow of information from one element to another. The above relation can also be written as

$$\mathbb{C}h = a_{max}\Delta t \tag{2.6}$$

where $\mathbb{C}$ is the CFL condition.

2.2 Hyperbolic equations for inviscid flow

For the Euler equations, the conserved variables and the corresponding fluxes are given by

$$Q = \begin{pmatrix} \rho \\ \rho u_i \\ \rho e \end{pmatrix} \qquad F_j = \begin{pmatrix} \rho u_j \\ \rho u_i u_j + \boldsymbol{p}\delta_{ij} \\ u_j\left(\rho e + \boldsymbol{p}\right) \end{pmatrix} \tag{2.7}$$

where ρ, u_i, $\mathbf{p}$ and e are the density, velocity, pressure and internal energy of the fluid respectively, and δ_{ij} is the Kronecker delta function. The Euler equation is closed with the equation of state given by

$$\mathbf{p} = (\gamma - 1)\,\rho\left(e - \frac{1}{2}\left|u\right|^2\right) \tag{2.8}$$

where γ is the specific heat ratio (1.4 for air) and $|u|^2$ is the velocity magnitude. The wave speeds corresponding to the Euler equations are given by the corresponding eigenvalues of the system. The eigenvalues pertaining to the Euler equations are

$\{u, u+c, u-c\}$ where c is the speed of sound given by

$$c = \sqrt{\gamma \frac{p}{\rho}} \tag{2.9}$$

2.3 Discontinuous Galerkin Method

2.3.1 Formulation of DG method

The finite difference schemes [21, 30, 31] solve Eqn. (2.1) in the difference form. For the Finite Volume schemes [23, 15], Eqn. (2.13) is integrated over the element Ω resulting in the following formulation:

$$\int_{\Omega} \frac{\partial Q}{\partial t} dV + \int_{\Omega} \nabla \cdot \vec{F} dV = 0 \tag{2.10}$$

which can be rewritten using the Gauss divergence theorem as

$$\int_{\Omega} \frac{\partial Q}{\partial t} dV + \int_{\partial\Omega} \vec{F} \cdot \vec{n} dS = 0 \tag{2.11}$$

where $\partial\Omega$ is the boundary of the cell Ω and $\vec{n}$ is the unit normal vector on the surface respectively. The fluxes at the boundary need to be replaced with the Riemann fluxes in order to allow distinction of solutions across the interface. This results in the formulation as

$$\int_{\Omega} \frac{\partial Q}{\partial t} dV + \int_{\partial\Omega} \vec{H} \cdot \vec{n} dS = 0 \tag{2.12}$$

where $\vec{H}$ is the Riemann flux at the surface. This formulation is conservative in nature and also can support discontinuous solutions across interface.

The DG formulation is obtained in a similar manner as that of the FV method. An additional step is performed on the Eqn. (2.1) by multiplying the equation with an arbitrary trial function ψ. The resulting equation set is given by

$$\psi \frac{\partial Q}{\partial t} + \psi \nabla \cdot \vec{F} = 0 \tag{2.13}$$

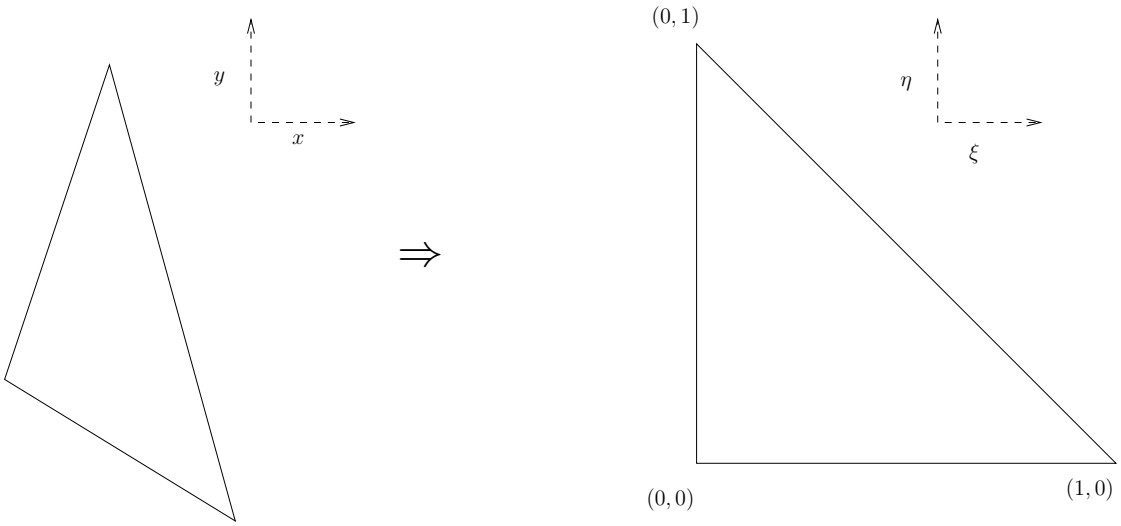


Figure 2.1: Transformation of a triangle element from physical (x, y) plane to reference (ξ, η) coordinates.

Integrating over the element Ω and applying Gauss divergence theorem, we get

$$\int_{\Omega} \psi \frac{\partial Q}{\partial t} d\Omega + \int_{\partial\Omega} \psi \vec{F} \cdot \vec{n} d\Gamma - \int_{\Omega} (\nabla \psi) \cdot \vec{F} d\Omega = 0 \tag{2.14}$$

Similar to the FV formulation, the flux $\vec{F}$ at the surface $\partial\Omega$ are replaced with the Riemann flux $\vec{H}$ resulting in the DG formulation:

$$\int_{\Omega} \psi \frac{\partial Q}{\partial t} d\Omega + \int_{\partial\Omega} \psi \vec{H} \cdot \vec{n} d\Gamma - \int_{\Omega} (\nabla \psi) \cdot \vec{F} d\Omega = 0 \tag{2.15}$$

2.3.2 Transformation from physical to reference element

Though the DG scheme in Eqn. (2.15) can be applied to elements of arbitrary shape, in this work, we shall restrict ourselves to linear triangular type elements. The DG formulation (Eqn. 2.15) is applied on the reference triangle defined in terms of a reference coordinate system (ξ, η). The relation between the reference coordinates (ξ, η) and the physical coordinates (x, y) is given by

$$\begin{aligned} x &= x_1 + (x_2 - x_1)\xi + (x_3 - x_1)\eta \\ y &= y_1 + (y_2 - y_1)\xi + (y_3 - y_1)\eta \end{aligned} \tag{2.16}$$

where (x_i, y_i) represent the physical coordinate of i^{th} vertex of the triangle. Fig. 2.1 illustrates the transformation of a triangle from a physical (x, y) plane to a reference (ξ, η) plane. The solution polynomial Q and flux polynomial $\vec{F}$ are defined in the reference coordinate system. Appropriate transformations are applied to transform Eqn. (2.15) from physical to reference plane. The solution Q may be represented in polynomial form as

$$Q = \sum_{i=1}^{N_b} \alpha_i \phi_i = [\alpha][\phi] \tag{2.17}$$

where α_i is the coefficient corresponding to the polynomial spatial basis function $\phi_i(\xi, \eta)$ and N_b is the number of basis functions used in representing the solution Q. For convenience, we represent Eqn. (2.17) as

$$Q = [\alpha][\phi] \tag{2.18}$$

where $[\alpha] = [\alpha_1, \alpha_2, \alpha_3, ...]$ and $[\phi] = [\phi_1, \phi_2, \phi_3, ...]$. It should be noted that the basis functions ϕ_i are functions of spatial coordinates while their coefficients α_i are functions of time t.

The minimum number of basis functions required for a p^{th} degree polynomial in two dimensions is given by

$$N_b = \frac{(p+1)(p+2)}{2} \tag{2.19}$$

Note that p is the degree of solution polynomial Q while $\mathbf{p}$ denotes the pressure in Eqn. (2.7).

The DG formulation is obtained by using the N_b basis functions ϕ_i as the trial functions (ψ) in Eqn. (2.15). For simplicity, the resulting equation is expressed using tensor notation, implying the summation of index appearing twice. On replacing the trial functions with the basis functions, the Eqn. (2.15) becomes

$$\int_\Omega \phi_j \phi_i d\Omega \cdot \frac{d\alpha_i}{dt} + \int_\Gamma \phi_j \vec{H} \cdot \vec{n} d\Gamma - \int_\Omega (\nabla \phi_j) \cdot \vec{F} d\Omega = 0 \tag{2.20}$$

The solution states required in evaluation of Riemann fluxes are recovered from Eqn. 2.18.

2.4 Temporal discretization

The order of temporal discretization should match the order of spatial discretization for a scheme to be formally higher order accurate. Since the DG scheme can have an arbitrary order of accuracy in space, here we use a temporal discretization scheme that can be formulated to any higher order accuracy if required. For this work, a low storage m-stage m^{th} order Strong Stability Preserving Runge Kutta (SSP-RK) [75, 76] scheme is used to march the Eqn. (2.20) in time. Consider a function Ψ for which the first order time derivative is known. The m-stage SSP-RK time integration scheme is given by

$$\begin{aligned} \Psi^{(0)} &= \Psi^n \\ \Psi^{(i)} &= \Psi^{(i-1)} + \Delta t \tfrac{d}{dt}\Psi^{(i-1)} \\ \Psi^{n+1} &= \textstyle\sum_{k=0}^{m} C_{m,k}\Psi^{(k)} \end{aligned} \tag{2.21}$$

where $\Psi^{(k)}$ is the intermediate values of Ψ at k^{th} RK stage and the coefficients $C_{m,k}$ of the m stage RK scheme are obtained as [75]

$$\begin{aligned} C_{1,0} &= 1 \\ C_{m,k} &= \tfrac{1}{k} C_{m-1,k-1}, \quad k = 1, \ldots, m-2 \\ C_{m,m} &= \tfrac{1}{m!} \\ C_{m,m-1} &= 0 \\ C_{m,0} &= 1 - \textstyle\sum_{k-1}^{m-1} C_{m,k} \end{aligned} \tag{2.22}$$

As previously mentioned, in order to maintain the overall order of accuracy, the order of time integrator is matched with the spatial order of representation of the solution Q in Eqn. (2.17).

2.5 Interface Fluxes

The RK-DG scheme formulated above requires an exact or approximate Riemann solver for computing the interface fluxes H. It is known that the Riemann solvers have a significant influence in the resolution of the discontinuities [77]. The sensitivity of the schemes to the interface fluxes become more significant for higher orders ($p > 2$). Here, we consider two formulations for the Riemann fluxes: LLF (Local Lax Friedrich) or Rusanov fluxes and the HLLC (Harten, Lax and van Leer with Contact wave resolution) flux. These fluxes are chosen for their different properties in the vicinity of strong shocks.

The formulation of HLLC flux [78] is given by

$$H_{HLLC} = \begin{cases} F_L & 0 \leq s_L \\ F_L + s_L\left(Q_{*L} - Q_L\right) & s_L \leq 0 \leq s_* \\ F_R + s_R\left(Q_{*R} - Q_R\right) & s_* \leq 0 \leq s_R \\ F_R & 0 \geq s_R \end{cases} \tag{2.23}$$

where s is the wave speed estimate and the subscripts L, $*$ and R represent left, intermediate and right states respectively. It is observed that for HLLC fluxes, the choice of wave speed estimates s_L, s_R and s_* plays a significant role in achieving numerical stability and oscillation free solution. We use the wave speed estimates as given by E.F. Toro [78]:

$$s_L = Min(u_L - c_L,\, \bar{u} - \bar{c}), \qquad s_* = u_* \qquad s_R = Max(u_R + c_R,\, \bar{u} + \bar{c}) \tag{2.24}$$

where, c is the speed of sound. The intermediate velocity u_* is given by

$$u_* = \bar{u} - \frac{1}{2}\frac{\mathbf{p}_R - \mathbf{p}_L}{\bar{\rho}\bar{c}} \tag{2.25}$$

where $\bar{\rho}$, $\bar{u}$ and $\bar{c}$ are the averaged density, velocity and sound speed across the interface $\left(\bar{[]} = \frac{1}{2}([]_L + []_R)\right)$. For the computation of intermediate variables Q_*, the

reader is referred to chapter 10 in [78]. The more diffusive LLF fluxes are computed as

$$F_{LLF} = \frac{1}{2}\left(F_L + F_R\right) + \frac{s^+}{2}\left(Q_L - Q_R\right) \tag{2.26}$$

where s^+ is the maximum wave speed given by

$$s^+ = Max(|u_L| + c_L, |u_R| + c_R) \tag{2.27}$$

Compared to LLF flux, the HLLC flux can resolve the contact discontinuities more sharply, thus resulting in a more accurate solution. Here, unless mentioned, the HLLC flux is used for the formulation of interface flux $\vec{H}$.

2.6 Order of error convergence

The order of convergence represents the rate of convergence of errors in the numerical solution with respect to the resolution of the domain. Higher order of (error) convergence is obtained by using higher order approximations for the solutions and their fluxes along with higher order temporal discretization. The rate of error reduction with respect to the element size is called the "grid convergence". This property can be explained by representing a numerical approximation in the form of an one dimensional Taylor Series expansion. For a given function $f(x)$, the numerical approximation can be written as

$$f(x)_h = f(x)_{exact} + g_1 h + g_2 h^2 + g_3 h^3 + O(h^4) \tag{2.28}$$

where $f(x)_{exact}$ is the exact value of the function, $f(x)_h$ is the approximated function on a grid with average spacing of h, g_i is the i^{th} error coefficient and $O(h^k)$ is the error of order k. A scheme of order $(p+1)$ can represent the solution up to a degree of p. Hence, the error coefficients $g_1, g_2, g_3, ...g_p$ becomes zero in Eqn. (2.28). Neglecting the errors of order higher than $(p+1)$, Eqn. (2.28) can be generalized

as

$$f(x)_{h,p} = f(x)_{exact} + g_{p+1}h^{p+1} \tag{2.29}$$

where $f(x)_{h,p}$ is the function $f(x)_h$ approximated to a degree of p. The error in numerical approximation $\epsilon_{h,p}$ is defined as the deviation of approximated function $f(x)_{h,p}$ from the exact function $f(x)_{exact}$. This error is related to the grid spacing and the degree of approximation as

$$\epsilon_{h,p} = \|f(x)_{h,p} - f(x)_{exact}\|$$

or,

$$\epsilon_{h,p} \propto h^{p+1} \tag{2.30}$$

The error term decreases at the rate of $(p+1)$ with respect to decrease in h. This rate of convergence of error term is taken to be the order of the underlying numerical scheme. As it can be seen in Eqn. (2.30), the rate of error convergence increases with increase in order of scheme. This property of higher order schemes enables it to obtain a highly accurate solution on a coarse grid compared to the second order schemes. While the error decreases at the rate of $(p+1)$, the computational cost of the numerical scheme increases in a relatively slower rate.

The order of convergence for a numerical scheme can be estimated numerically using the formula given in Eqn. (2.30). The actual order of accuracy of the numerical schemes can be obtained by rewriting Eqn. (2.30) as

$$O_p = \frac{log\left(\frac{\epsilon_{h_1,p}}{\epsilon_{h_2,p}}\right)}{log\left(\frac{h_1}{h_2}\right)} \tag{2.31}$$

or,

$$O_p = \frac{log\left(\epsilon_{h_1,p}\right) - log\left(\epsilon_{h_2,p}\right)}{log\left(h_1\right) - log(h_2)} \tag{2.32}$$

where O_p is the numerical order of scheme which approximates the solution to p^{th} degree. Thus the order of (grid) convergence of the numerical scheme can be

extracted by performing computations on a mesh with different element sizes (h) and plotting the results in a log-log scale. This technique will often be used in this work to test the convergence characteristics of the numerical schemes.

The actual order of convergence of the numerical scheme can be estimated by performing two or more computations with different values of h. The numerical order of accuracy of the function approximation is given by

$$(p+1)_{app} = \frac{ln\left(\frac{\epsilon_{h_1,p}}{\epsilon_{h_2,p}}\right)}{ln\left(\frac{h_1}{h_2}\right)} \tag{2.33}$$

where $(p+1)_{app}$ is the approximate order of the numerical scheme and $\epsilon_{h,p}$ is the error corresponding to a solution obtained using a $(p+1)^{th}$ order scheme on a grid with element size h. The numerical error norm can be maximum absolute error in the domain (L_∞ error) or the average of absolute value (L_1 error) or the root mean square value (L_2 error).

Though the numerical scheme is designed to be formally $(p+1)^{th}$ order accurate, it is often difficult to numerically reproduce the designed order. The non-linearity in the equation set and the limitation of the numerical accuracy of the computers can affect the numerical order of accuracy of the underlying discretization schemes. This makes it imperative to estimate the actual order of the scheme compared to the designed order of scheme. For this purpose, we use a time evolving solution to obtain the numerical order of accuracy.

2.7 Numerical validation for RK-DG scheme

An isentropic vortex evolution is commonly used for Euler equations since the exact analytical solution is known at all time. An isentropic vortex at a point (0,0) is initialized by the following equations

$$\rho = \left(1 - \frac{(\gamma-1)\,\beta^2 exp\,(1-r^2)}{8\gamma\pi^2}\right)^{\frac{1}{\gamma-1}} \tag{2.34}$$

Order	h	L_1 err.	L_∞ err.	$O(L_1)$	$O(L_\infty)$
2	1	3.67e-3	1.42e-3		
	$\frac{1}{2}$	9.27e-4	2.93e-4	1.98	2.28
	$\frac{1}{4}$	1.88e-4	6.87e-5	2.3	2.09
3	1	8.24e-4	4.06e-4		
	$\frac{1}{2}$	7.86e-5	2.08e-5	3.39	4.29
	$\frac{1}{4}$	6.85e-6	3.75e-6	3.52	2.47
4	1	1.51e-4	3.75e-5		
	$\frac{1}{2}$	8.79e-6	4.75e-6	4.1	2.98
	$\frac{1}{4}$	4.75e-7	3.15e-7	4.21	3.91
5	1	3.15e-5	1.09e-5		
	$\frac{1}{2}$	1.18e-6	4.18e-7	4.74	4.7

Table 2.1: Solution errors at $t = 10$ for convection of isentropic vortex

$$u = 1 - \frac{\beta y}{2\pi} exp\left(\frac{1 - r^2}{2}\right) \tag{2.35}$$

$$v = \frac{\beta x}{2\pi} exp\left(\frac{1 - r^2}{2}\right) \tag{2.36}$$

$$\mathbf{p} = \rho^\gamma \tag{2.37}$$

where β is the strength of the vortex and $r = \sqrt{x^2 + y^2}$ is the distance from the center of the vortex. A $\beta = 3$ is used in the present simulation. The computation is performed in a periodic domain of size 10x10. The unstructured mesh is generated using the open source mesh generator GMSH [79] with frontal algorithm. The refined grids are obtained by splitting each triangle into four smaller triangles, thereby reducing the grid size by 50%.

The numerical solutions obtained at time $t = 10$ are compared against the analytical solutions. The resulting solution errors obtained using the RK-DG method are listed in Table 2.1. As described in Section 2.6, the rate of error reduction with grid refinement increases with increase in order of the scheme. Thus, the higher order schemes are able to attain very high accuracy in a relatively coarse grid compared to the lower order schemes. As expected, the numerical order of convergence differs slightly from the expected order of accuracy. However, the deviation from

the designed order of the scheme decreases with mesh refinement. The numerical schemes, however, are also constrained by the machine accuracy, thereby restricting the reduction in solution error beyond a certain level of tolerance.

2.8 Summary

In this chapter the basic foundation for numerical methods is presented. The spatial and temporal discretization methods are detailed along with the definition of interface fluxes. The Discontinuous Galerkin method is chosen for spatial discretization while Runge Kutta method is chosen for temporal discretization. Theory of error reduction in numerical scheme is also presented along with numerical examples illustrating the convergence characteristics of RKDG schemes on an isentropic vortex evolution problem. Grid convergence tests show that the scheme indeed can achieve the designed order of accuracy in all cases. The various components of the numerical method described in this chapter will provide the building blocks of the numerical schemes developed later in this work.

Chapter 3

Riemann solvers on Extended Domains

It is known that the physical phenomenon represented by hyperbolic equations are governed by propagation of waves within the medium. Very little consideration has been given for analyzing the wave propagation characteristics within the element. Most of the numerical methods intrinsically assume that the boundary Riemann flux has an immediate influence on the internal solution. But, even within the elements, for an explicit scheme, the information propagation (and hence the influence of the boundary) is restricted by the limited size of time step. In such cases, only a part of the element in the vicinity of its boundary would be affected by the Riemann solution, while the interior region away from the element boundary remains unaffected.

In this chapter, we propose a new generalized method of flux approximation for higher-order solution of hyperbolic conservation laws in which the Riemann solution at the element boundary is represented as a correction to the internal solution evolution process. The influence of this correction term on the internal solution is approximated based on the wave propagation characteristics of the scheme. The numerical schemes are constructed by computing the fluxes based on this approximated solution. The class of schemes formulated using the modified

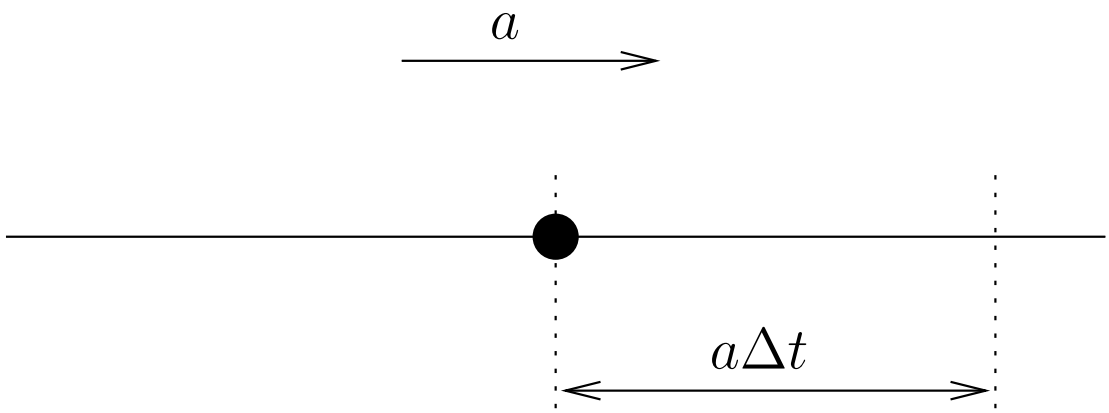


Figure 3.1: Region influenced by a perturbation traveling at a wave speed of a over a time period of Δt.

flux approximation is named as the Extended Riemann (ExRi) schemes. Optimally stable schemes of up to 7^{th} order accuracy are formulated in 1D and up to 5^{th} order accuracy are formulated for 2D triangular elements. The influence of Riemann corrections on the wave propagation characteristics of the schemes are analyzed by studying their dispersive and diffusive characteristics. Numerical tests show that all the 1D and 2D schemes proposed here achieve the desired convergence characteristics.

3.1 Theoretical Formulation

3.1.1 Wave Propagation Characteristics

A general hyperbolic equation is represented in conservative form as in Eqn. (2.1). Here, we shall first consider the equation in one dimension to explain the proposed concept. The hyperbolic equation in one dimension is given by

$$\frac{\partial Q}{\partial t} + \frac{\partial F(Q)}{\partial x} = 0 \tag{3.1}$$

where Q is the solution variable (in conserved form) and $F(Q)$ is the corresponding flux. Here, the source term is neglected for simplicity. The equation can also be

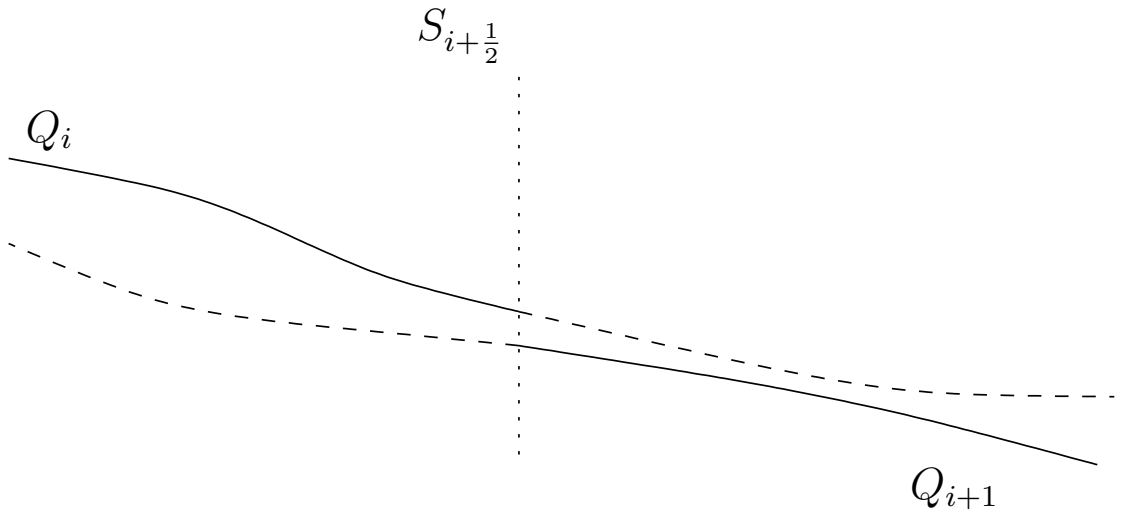


Figure 3.2: Discontinuous solution at an interface.

expressed as a wave equation as

$$\frac{\partial Q}{\partial t} + a\frac{\partial Q}{\partial x} = 0 \tag{3.2}$$

where the speed of wave propagation a is given by the Flux Jacobian

$$a = \frac{\partial F(Q)}{\partial Q} \tag{3.3}$$

In equations Eqn. (3.2) and Eqn. (3.3), the characteristics of the hyperbolic equation Eqn. (3.1) is governed by the wave speed a, as this represents the propagation of information from one point to another. For a one dimensional system with a constant wave speed a, the region influenced by an arbitrary point over a time period of Δt is the region within a distance of $a\Delta t$ along the direction of wave propagation. Fig. 3.1 illustrates the domain of influence for a linear hyperbolic equation for a perturbation traveling from left to right over a time period Δt. For non-linear equations, an approximate estimate of the wave speed can be obtained for a very small time step ($\Delta t \to 0$) by linearizing the equation at that time instance.

3.1.2 Influence of Riemann Solution in a discrete element

Consider a discrete element over which the solution Q is approximated by Q_i where the subscript i represents the element index. The flux F_i with respect to element i can be calculated from the solution Q_i. At any given time instance, the solution at an element interface is influenced by the internal solution of the element and the corresponding neighbor sharing that interface. Hence, the flux F_i computed only from the internal solution Q_i is not sufficient to march the solution Q_i as it lacks the additional information from the neighbors. Due to the piecewise representation of Q_i, the neighboring solution information can be discontinuous at interfaces. The discontinuous solutions represented by Q_i and Q_{i+1} at the interface $S_{i+\frac{1}{2}}$ is illustrated in Figure 3.2. A Riemann solver is then used to obtain a solution across such discontinuities. Assuming that we know an exact or approximate Riemann solver for the given hyperbolic equation, the Riemann flux $\hat{F}^S$ at the interface S is expressed as

$$\hat{F}^S = H\left(Q^L, Q^R\right) \tag{3.4}$$

where $H(Q^L, Q^R)$ is the Riemann solver for the left and right states Q^L and Q^R.

To march the solution in time, an approximate flux $\tilde{F}_i$ should be calculated from the internal flux F_i and the Riemann fluxes $\hat{F}_i^S$. Note that the Riemann fluxes are defined only at the element boundaries. The characteristics of the approximate flux $\tilde{F}_i$ can be determined by analyzing the effect of Riemann solution on the element interior. Let Δt be the time period through which we need to evolve the solution Q_i. For an element with wave speed a, the region influenced by the Riemann solution over a time period of Δt is the region at a distance of $a\Delta t$ from the interface. This width of this region can be represented in terms of a non-dimensional number $\mathcal{C}$ and the element width h as

$$\mathcal{C} = \frac{a}{h}\Delta t \tag{3.5}$$

The formulation is synonymous with the definition of CFL number. However, here, we allow the value of $\mathcal{C}$ to differ from the actual CFL number as will be

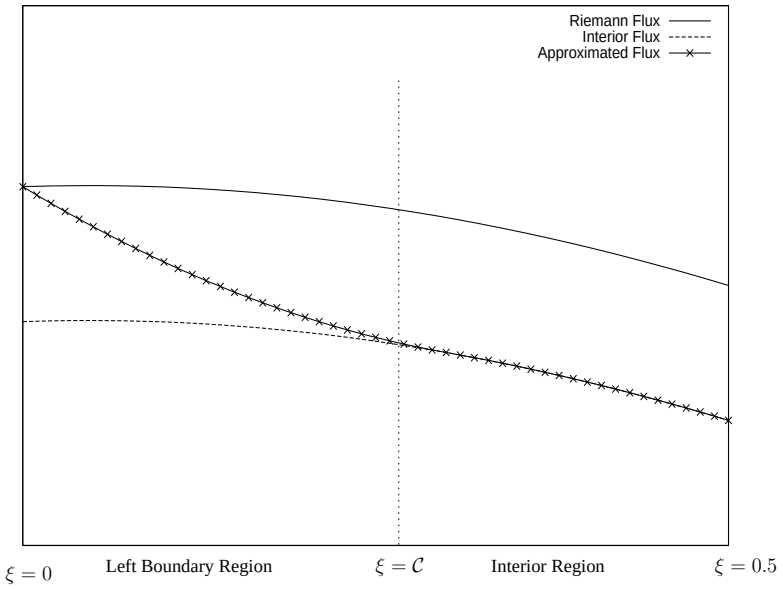


Figure 3.3: Illustration of modified flux for a 1D element. (the Riemann flux is illustrated as a higher order flux)

explained later in the following sections. The width of the region influenced by the Riemann solution at time instance $t + \Delta t$ can now be defined as $\mathcal{C}h$. Hence, at time $t + \Delta t$, the Riemann interface flux $\hat{F}^S$ does not have any influence on the interior solution Q_i beyond a distance of $\mathcal{C}h$ from the interface. This forms the basis of the current formulation. The proposed formulation is firstly detailed for one dimensional hyperbolic systems, and will be extended to multi-dimensional systems further on.

We consider an element Ω bounded by its boundary given by $\partial\Omega$. This boundary consists of various interfaces denoted by S. Each interface is shared either with the neighboring element, or, the computational boundary. Figure 3.3 illustrates the availability of information across the left interface. For clarity, the local variable ξ of the element is used and only the left half of the element is shown in the figure. As illustrated in the figure, the approximate flux $\tilde{F}_i$ should respect the Riemann solution $\hat{F}_i$ at the corresponding interface. As one moves towards the element interior, the approximate flux $\tilde{F}_i$ should gradually blend into the internal flux F_i. The region where the influence of the Riemann solution at the interface S is felt, is denoted as the boundary region Ω^S and the region not influenced by

the Riemann solution is denoted as the internal region Ω^{in}. The nature of the blending of approximate flux $\tilde{F}_i$ in the region Ω^S cannot be defined exactly due to the discontinuity at the interface. However, an approximate form of $\tilde{F}_i$ can be derived based on the boundary conditions at the interface and the element interior.

To define the properties of the approximate flux $\tilde{F}_i$, we expand the approximate flux $\tilde{F}_i$ in terms of the internal flux F_i and a correction term $\mathcal{F}_i$ arising due to Riemann fluxes at the element boundary. The approximate flux for an element is expressed as

$$\tilde{F}_i(\xi) = F_i(\xi) + \sum_{j=1}^{N_S} \mathcal{F}_i^j(\xi) \tag{3.6}$$

where ξ is the local coordinate of the element, N_S is the number of interfaces ($N_S = 2$ for 1D element) and $\mathcal{F}_i^S$ is the Riemann correction corresponding to interface S of i^{th} element. By nature of definition, the Riemann flux correction $\mathcal{F}_i^S$ takes the form:

$$\mathcal{F}_i^S(\xi) = \omega(\zeta_S(\xi)) \left(\hat{F}_S(\xi_S) - F_i(\xi_S) \right) \tag{3.7}$$

where ω is an arbitrary blending function, $\zeta_S(\xi)$ is the coordinate transformation of ξ in the region Ω^S corresponding to the interface S. The transformed coordinate ζ_S takes the value of 1 at the interface and decreases to zero towards the interior. As an example, for a one dimensional element, the ζ_S coordinates can be represented by the following formulations:

$$\begin{aligned} \zeta_L(\xi) &= \left(1 - \frac{\xi}{\mathcal{C}}\right) \\ \zeta_R(\xi) &= \left(1 - \frac{1-\xi}{\mathcal{C}}\right) \end{aligned} \tag{3.8}$$

where ζ_L and ζ_R are the transformed coordinates corresponding the left and right interfaces.

The flux correction term presented in this work [80] (Eqn. 3.7) is similar to the generic "correction flux polynomial" detailed in the lifting collocation penalty formulation by Wang. et. al. [81]. In the lifting collocation formulation a correction field is computed using polynomial interpolation techniques such as lagrangian

interpolation. However, the current work, we emphasise on utilization of physical wave propagation characteristics to model the flux correction and limiting the influence of flux correction to a localized region inside the element space. The resulting scheme would be based on an analytical formulation that is computationally efficient as compared to the lifting collocation method [81].

3.1.3 Properties of the blending function ω

To derive the properties of the blending function ω, we represent the solution evolution in terms of the contributions from within the element and its neighbors. For an arbitrary element, the solution evolution can be represented as

$$\frac{\partial Q_i(\xi)}{\partial t} = \frac{\partial Q_i^*(\xi)}{\partial t} + \sum_{j=1}^{N_s} \frac{\partial \varepsilon_i^j(\xi)}{\partial t} \tag{3.9}$$

where $\partial Q_i/\partial t$ and $\partial Q_i^*/\partial t$ are the solution evolution with respect to approximate flux $\tilde{F}_i$ and internal flux F_i respectively using Eqn. (3.6). Note that the internal flux F_i is computed using the original element solution Q_i. The correction term $\partial\varepsilon/\partial t$ is represented in terms of the Riemann flux correction $\mathcal{F}$ as

$$\frac{\partial \varepsilon_i^j}{\partial t} + \frac{\partial \mathcal{F}_i^j}{\partial x} = 0 \tag{3.10}$$

It is important to note that while evolving the internal solution Q_i, the corrected time derivative $\partial Q_i/\partial t$ (Eqn. 3.9) is used instead of the internal solution evolution term given by $\partial Q_i^*/\partial t$. It is possible to represent any higher order scheme in the form of Eqn. (3.9). In the conventional methods, the correction term ε is not explicitly decoupled as in Eqn. (3.9). In this work, we aim at modeling the correction term in an explicit manner so that a uniform flux function is used for solution evolution.

Formulation of the Riemann correction term $\mathcal{F}$ requires appropriate boundary condition. Following the wave propagation characteristics within a given element, it can be deduced that the Riemann flux correction $\mathcal{F}$ and its corresponding con-

tribution to solution evolution ε should vanish as one moves a distance of $\mathcal{C}h$ away from the boundary. This gives us the resultant form of $\mathcal{F}$ as

$$\frac{\partial^n \mathcal{F}}{\partial \xi^n} = 0 \qquad \xi \in \Omega^{in}, n = 0 \ldots . (p+1) \tag{3.11}$$

where p is the degree of solution representation and Ω^{in} is the internal region not influenced by the Riemann flux correction. At the interface itself, the Riemann correction is given by

$$\mathcal{F}_i^j(\xi) = \hat{F}_S(\xi) - F_i(\xi) \text{ when } \zeta(\xi) = 1 \tag{3.12}$$

Based on the equation Eqn. (3.7) and constraints given by Eqns. (3.11-3.12), the conditions for the blending function ω are

$$\omega(\zeta = 1) \quad = \quad 1 \tag{3.13}$$

$$\frac{\partial^n \omega}{\partial \zeta^n} = 0 \qquad n = 0 \ldots (p+1), \zeta \leqslant 0 \tag{3.14}$$

The influence of the Riemann solution always increases as one moves towards the interface ($\zeta \rightarrow 1$). Hence, an additional constraint can be posed as

$$\frac{\partial \omega}{\partial \zeta} \geqslant 0 \qquad 0 \leqslant \zeta \leqslant 1 \tag{3.15}$$

The conditions given by Eqns. (3.13)-(3.15) can be satisfied by any suitable choice of blending function. While a variety of choices are possible, in this work we restrict to simple polynomial based blending functions. The formulations for 1D and 2D schemes are detailed later in Sec. 3.2.2 and Sec. 3.3.1. With the definition of ω, the Riemann correction $\mathcal{F}$ is now valid over a region Ω^S rather than only at the interface S. Equation 3.7 can be viewed as an approximate Riemann solver for the boundary region Ω^S.

3.2 Implementation in 1D Schemes

In this section, the proposed method is formulated for a 1D element. Two different solution evolution schemes based on Spectral-Difference like co-location scheme and Discontinuous Galerkin schemes are tested. The current formulation is dependent on the blending function ω and the resulting Riemann flux correction $\mathcal{F}$. The wave propagation characteristics and stability of the schemes are studied in detail using dispersion-relation analysis [82, 83]. The formulated schemes are then tested for accuracy and robustness on different problems.

3.2.1 Numerical approximation using Basis functions

For a higher order element, the internal solution Q_i within the element can be expanded with the help of basis function ϕ as

$$Q_i \quad = \quad = \sum_{j=1}^{p+1} \phi_j \alpha_j^i = [\phi]^T \left[\alpha^i\right] \tag{3.16}$$

where α_j^i is the coefficient of the basis function ϕ_j for element of index i. Note that for the basis coefficient α^i, the element index is in super-script while for the solution Q_i the element index is denoted in the subscript. The vectors are of length $(p+1)$ where p is the degree of approximated solution. The approximate numerical flux $\tilde{F}_i$ for an element is given by the Eqn. (3.6). The fluxes can be treated explicitly as two individual terms (F_i and $\mathcal{F}_i$) or they can be treated as a single polynomial. If the flux is approximated as a single polynomial, the Eqn. (3.1) becomes

$$\frac{\partial Q_i}{\partial t} + \frac{\partial \tilde{F}_i}{\partial x} = 0 \tag{3.17}$$

where $\partial Q_i/\partial t$ is the modified time evolution with respect to the modified flux $\tilde{F}_i$. This will be used to advance the solution of element i at a given time step. If the modified flux is treated explicitly in terms of Riemann correction $\mathcal{F}$, the equation

can be expanded as

$$\frac{\partial Q_i}{\partial t} + \frac{\partial F_i}{\partial x} + \sum_{j=1}^{N_S} \frac{\partial}{\partial x}\left(\mathcal{F}_i^j\left(\xi\right)\right) = 0 \tag{3.18}$$

A scheme can be optimized to deal with the third term (in Eqn. 3.18), efficiently while the second term can be handled by an exact or approximate method. Numerical experiments suggest that the stability of the schemes are relatively insensitive to the choice of approximation of the internal flux F_i and its corresponding solution evolution term $\partial Q_i^*/\partial t$.

3.2.2 Formulation of blending functions in 1D

The Riemann correction $\mathcal{F}$ can be defined by a suitable choice of blending function. The blending function should satisfy the conditions given by Eqns. (3.13-3.15). Many different choices of the blending function formulation can be established. For example, trigonometric functions can be used to represent the smooth variation of the function from 1 to 0. In this work, we adopt a polynomial function given by

$$\omega\left(\zeta\right) = \begin{cases} \zeta^{p_\omega} & 1 \geqslant \zeta \geqslant 0 \\ 0 & \zeta \leqslant 0 \end{cases} \tag{3.19}$$

where p_ω is the degree of the blending function. The degree of the blending functions are taken to be equal to that of the flux reconstruction. The form of the blending function ω results in simpler formulation of the derivative terms and satisfies the constraints posed for formulating the blending function. An example of the proposed blending function ω corresponding to the left and right interfaces in a 1D element is illustrated in Figure 3.4.

With the formulation of the blending function, the modified/approximated flux term $\tilde{F}$ can be formulated as a continuous function. Once the flux $\tilde{F}$ is available, for any point within the element, we can write the equation Eqn. (3.17) as

$$[\phi]^T \left[\frac{d\alpha^i}{dt}\right] + \frac{\partial \tilde{F}_i}{\partial x} = 0 \tag{3.20}$$

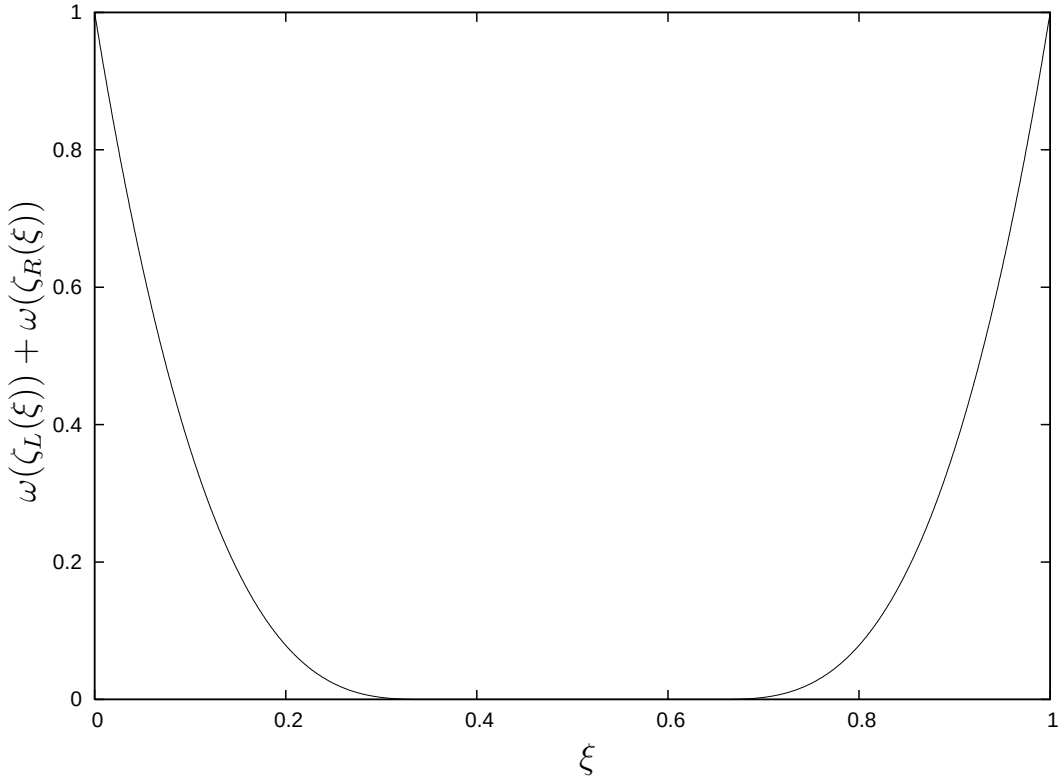


Figure 3.4: Illustrative plot of blending function ω for $\mathcal{C} = 0.35$ and $p = 4$.

Riemann fluxes are used for the solution of the discontinuous states at the boundary. For a 1D scalar wave equation with unit wave speed $\left(\frac{\partial Q}{\partial t} + \frac{\partial Q}{\partial x} = 0\right)$, the Riemann fluxes for the left ($\hat{F}^L$) and right ($\hat{F}^R$) interfaces are given by

$$\begin{aligned} \hat{F}_i^L &= F_{i-1}(1) \\ \hat{F}_i^R &= F_i(1) \end{aligned} \tag{3.21}$$

Substituting 3.21 in 3.6, we get the full form of the flux as

$$\tilde{F}_i(\xi) = \mathcal{F}_i^L(\xi) + F_i(\xi) + \mathcal{F}_i^R(\xi) \tag{3.22}$$

where

$$\mathcal{F}_i^L(\xi) = \omega\left(\zeta_L(\xi)\right)\left(\hat{F}_i^L - F_i(0)\right) \tag{3.23}$$

and

$$\mathcal{F}_i^R(\xi) = \omega\left(\zeta_R(\xi)\right)\left(\hat{F}_i^R - F_i(1)\right) \tag{3.24}$$

Note that the flux corrections depends only on the correction term at the boundary and decay towards the interior.

In the case of a scalar wave equation with unit wave speed, the internal flux is given by

$$F_i = Q_i = [\phi]^T[\alpha^i] \tag{3.25}$$

3.2.3 Implementation in Numerical Schemes

There are different methods of spatial discretization for obtaining higher order solution of hyperbolic conservation laws. Here we choose co-location and Galerkin methods for demonstrating the proposed ExRi formulation in one dimension. The standard blending function formulation given in Eqn. (3.19) is used throughout.

3.2.3.1 Method of Co-location

In the method of co-location, the quantities are sampled and evaluated at specific co-location points. The solution is still represented in the basis expansion form. However the polynomial approximation of a given function is achieved by minimizing the errors at the co-location points. An exact matrix inversion is used in cases where the number of points are same as that of the number of coefficients in the polynomial approximation. The resulting scheme is simular to using a lagrangian interpolant. In situations of over-constrained system as in the current work (detailed at the later part of this section), a least squares approach is used to minimize the errors at co-location points.

In order to maintain the order of accuracy over all the terms in the equation, the order of approximation of first order spatial derivative of flux should match the order of approximation of solution state. To ensure this condition, the flux functions should be reconstructed to at least one order higher than the order of spatial approximation of solutions (and hence it's time derivatives). In this work, unless mentioned, we use $N_Q = p + 1$ co-location points for the state variable Q_i and $N_F = p + 2$ co-location points for reconstructing the flux $\tilde{F}_i$. The co-location

points are derived from a modified version of Chebyshev nodes given by

$$x_j = \cos\left(b_j \lambda^{1-|b_j|}\right) \tag{3.26}$$

where λ is the scaling factor for the points and

$$b_j = \left(\frac{2j}{N-1}\right) - 1 \tag{3.27}$$

with N as the number of points (N_Q for solutions and N_F for fluxes). The value of λ affects the distribution of co-location points. The points are pushed towards the extreme (0, 1) as λ increases. The original Chebyshev [7] nodes can be recovered by setting $\lambda = 1$. This modification of the point distribution may be required to make the scheme stable.

The fluxes can be expanded in terms of the flux basis function ϕ^F as

$$\tilde{F}_i \;=\; \left[\phi^F\right]^T \left[\beta^i\right] \tag{3.28}$$

where β^i is the basis coefficients corresponding to the flux basis function ϕ^F for i^{th} element. Note that the degree of reconstruction for the flux is $(p+2)$. The process of flux reconstruction includes the information from the neighbors through the Riemann solution at the interfaces. For simple cases, the fluxes can also be treated in an alternative form given by Eqn. (3.18). Here, we represent the flux as in Eqn. (3.28). The flux coefficients β can be obtained by evaluating the flux at the co-location points.

Once the reconstructed flux $\tilde{F}_i$ is available, the time derivatives can be computed by solving the linear system of equations (Eqn. 3.20) at the co-location points. The resulting equation will be

$$[\phi]^T \left[\frac{\partial \alpha^i}{\partial t}\right] + \left[\frac{\partial \phi^F}{\partial x}\right]^T \left[\beta^i\right] = 0 \tag{3.29}$$

The conservation is enforced by applying the Gauss divergence theorem given by

$$\int_{\Omega} \frac{\partial Q_i}{\partial t} dV = -\int_{\partial\Omega} \hat{F}_i \cdot \vec{n} dS \tag{3.30}$$

where Ω is the element and S is the boundary with normal $\vec{n}$. Constrained least squares can be used to enforce the above condition. But, the computation can be simplified by choosing the appropriate basis functions with the properties below

$$\int_{\Omega} \phi_j dV = 0 \qquad j > 1 \tag{3.31}$$

and

$$\phi_1 = 1 \tag{3.32}$$

The properties of the basis functions remain unaffected as Eqn. (3.31) results in addition of a constant value to the functions. In this work, we adopt a simple polynomial basis function given by

$$[\phi] = \left(1, \;\; \xi - \tfrac{1}{2}, \;\; \xi^2 - \tfrac{1}{3}, \;\; \xi^3 - \tfrac{1}{4}, \;\; \xi^5 - \tfrac{1}{5}, \;\; \ldots \right) \tag{3.33}$$

which satisfies the conditions given in Eqn (3.31) and (3.32). With this formulation of basis functions, the conservation law in Eqn. (3.30) is given by

$$\frac{\partial \alpha_1^i}{\partial t} = -\frac{1}{V} \int_S \hat{F}_i \cdot \vec{n} dS \tag{3.34}$$

The coefficients of the rest of basis functions are obtained as

$$\sum_{r=2}^{p+1} \frac{\partial \alpha_r^i}{\partial t} \phi_r = -\frac{\partial \tilde{F}_i}{\partial x} - \frac{\partial \alpha_1^i}{\partial t} \tag{3.35}$$

where V is the volume of element Ω. Equation 3.34 can be evaluated as a conventional finite volume method. Equation 3.35 can be evaluated at N_Q co-location points and the resulting linear system of equations can be solved to estimate the

evolution of the state coefficients. However, in this case, a least square approximation is needed as the the system of Eqns. (3.35) would result in a rectangular coefficient matrix of size $(p+1) \times p$.

3.2.3.2 Obtaining matrix form of numerical scheme for scalar hyperbolic equation

The matrix form of the equations are required to perform the stability analysis of the numerical schemes. In order to form the matrix equivalent of the original hyperbolic equation, the fluxes and the solutions are expanded in terms of basis functions. For clarity, we present the original hyperbolic equation as

$$\frac{\partial Q}{\partial t} + \frac{\partial F}{\partial x} = 0 \tag{3.36}$$

Upon discretization, the solution Q and the flux F are replaced by the discretized solution Q_i and the approximate flux F_i respectively as in Eqn (3.17). Here, we consider a scalar hyperbolic equation with unit wave speed and assume that the elements are of uniform size. For a scalar hyperbolic equation with unit wave speed, the internal flux F_i is same as the solution Q_i. Hence, the approximate flux can be conveniently expressed as a linear combination of solutions Q_{i-1}, Q_i and Q_{i+1}. Considering wave propagation from left to right, the approximate flux (Eqn. 3.22-3.24) for the current case can be written as

$$\tilde{F}_i(\xi) = \omega(\zeta_L(\xi))(Q_{i-1}(1) - Q_i(0)) + Q_i(\xi) + \omega(\zeta_R(\xi))(Q_i(1) - Q_i(1)) \tag{3.37}$$

or

$$\tilde{F}_i(\xi) = \omega(\zeta_L(\xi))(Q_{i-1}(1) - Q_i(0)) + Q_i(\xi) \tag{3.38}$$

The flux coefficients β^i corresponding to the approximate flux $\hat{F}_i$ is calculated by evaluating the value of approximate flux $\tilde{F}_i$ (Eqn. 3.38) at each flux co-location

point. The resulting matrix form is given by

$$\left[\beta^i\right] = \left[\mathcal{N}_F^{-1}\right]\left[\alpha^{i-1}\right] + \left[\mathcal{N}_F^{0}\right]\left[\alpha^{i}\right] + \left[\mathcal{N}_F^{+1}\right]\left[\alpha^{i+1}\right] \tag{3.39}$$

where $\mathcal{N}_F^{-1}$, $\mathcal{N}_F^{0}$ and $\mathcal{N}_F^{+1}$ are the coefficient matrices corresponding to the $(i-1)^{th}$, i^{th} and $(i+1)^{th}$ elements. As the elements are of equal size, these coefficients do not vary between elements.

With the solution and flux coefficients available, we can rewrite Eqn. (3.36) in matrix form as

$$[\phi]^T \left[\frac{\partial \alpha^i}{\partial t}\right] + \left[\frac{\partial \phi^F}{\partial x}\right]\beta^i = 0 \tag{3.40}$$

Evaluating the above equation at each solution point results in a matrix equation:

$$[\mathcal{N}_Q]\left[\frac{\partial \alpha^i}{\partial t}\right] + [\mathcal{N}_{\partial F}]\left[\beta^i\right] = 0 \tag{3.41}$$

where $\mathcal{N}_Q$ and $\mathcal{N}_{\partial F}$ are matrices created by evaluating the basis functions ϕ and $\left(\partial \phi^F/\partial x\right)$ respectively at the co-location points. These can be written as

$$\begin{aligned} \left(\mathcal{N}_Q\right)_{j,k} &= \phi_k\left(\xi_j\right) \quad k = 1\ldots(p+1) \\ \left(\mathcal{N}_{\partial F}\right)_{j,k} &= \left.\frac{\partial \phi_k^F}{\partial x}\right|_{\xi_j} \quad k = 1\ldots(p+2) \end{aligned} \tag{3.42}$$

where the index $j = 1...N_Q$ represents the j^{th} solution point and k is the index of the basis function. The coefficient matrix $\mathcal{N}_Q$ is of dimensions $(N_Q \times (p+1))$. The coefficient matrix $\mathcal{N}_{\partial F}$ is of dimensions $(N_Q \times (p+2))$.

The complete matrix form is obtained by substituting 3.39 in equation Eqn. (3.41) resulting in a matrix system given by

$$[\mathcal{N}_Q]\left[\frac{\partial \alpha^i}{\partial t}\right] + [\mathcal{N}_{\partial F}]\left(\left[\mathcal{N}_F^{-1}\right]\left[\alpha^{i-1}\right] + \left[\mathcal{N}_F^{0}\right]\left[\alpha^{i}\right] + \left[\mathcal{N}_F^{+1}\right]\left[\alpha^{i+1}\right]\right) = 0 \tag{3.43}$$

Depending on the number of solution points considered, the inversion of the matrix $\mathcal{N}_Q$ might require a least squares inversion process. A least squares inversion of

N_Q would result in:

$$\left[\frac{\partial \alpha^i}{\partial t}\right] + \left[\mathcal{N}_Q^T \mathcal{N}_Q\right]^{-1} \left[\mathcal{N}_Q\right]^T \left[\mathcal{N}_{\partial F}\right] \left(\left[\mathcal{N}_F^{-1}\right] \left[\alpha^{i-1}\right] + \left[\mathcal{N}_F^0\right] \left[\alpha^i\right] + \left[\mathcal{N}_F^{+1}\right] \left[\alpha^{i+1}\right]\right) = 0 \tag{3.44}$$

This can be rearranged in terms of the solution coefficients α as

$$\left[\frac{\partial \alpha^i}{\partial t}\right] + \left[\mathcal{N}_Q^T \mathcal{N}_Q\right]^{-1} \left[\mathcal{N}_Q\right]^T \left[\mathcal{N}_{\partial F}\right] \begin{bmatrix} \left[\mathcal{N}_F^{-1}\right] \\ \left[\mathcal{N}_F^0\right] \\ \left[\mathcal{N}_F^{+1}\right] \end{bmatrix}^T \begin{bmatrix} \left[\alpha^{i-1}\right] \\ \left[\alpha^i\right] \\ \left[\alpha^{i+1}\right] \end{bmatrix} = 0 \tag{3.45}$$

or,

$$\left[\frac{\partial \alpha^i}{\partial t}\right] + \begin{bmatrix} \left[\mathcal{M}_R^{-1}\right] \\ \left[\mathcal{M}_R^0\right] \\ \left[\mathcal{M}_R^{+1}\right] \end{bmatrix}^T \begin{bmatrix} \left[\alpha^{i-1}\right] \\ \left[\alpha^i\right] \\ \left[\alpha^{i+1}\right] \end{bmatrix} = 0 \tag{3.46}$$

In the case of uniform element size, the coefficient matrices $\mathcal{M}_R$ is constant for all the elements. This coefficient matrix will be used later for performing Fourier stability analysis of the numerical schemes.

3.2.3.3 Galerkin Method

Here, we detail the implementation of Extended Riemann flux approximation in the DG formulation. The Galerkin method is obtained by multiplying the governing equations by an arbitrary trial function ψ and integrating over the element. The conservation equation Eqn. (3.18) becomes

$$\int_\Omega \psi \frac{\partial Q_i}{\partial t} dV + \int_\Omega \psi \frac{\partial \tilde{F}_i}{\partial x} dV = 0 \tag{3.47}$$

By expanding $\tilde{Q}_i$ in terms of basis functions ϕ and applying Gauss divergence rule, we get the classical Discontinuous Galerkin method as

$$\int_\Omega \left[\phi\phi^T\right] dV \left[\frac{\partial \alpha^i}{\partial t}\right] + \int_S [\phi]\, \tilde{F}_i dV - \int_\Omega \left[\frac{\partial \phi}{\partial x}\right] \tilde{F}_i dV = 0 \tag{3.48}$$

The flux $\tilde{F}_i$ is expanded in terms of the internal flux $\tilde{F}_i$ and the Riemann Correction $\mathcal{F}$. This results in

$$\int_\Omega \left[\phi\phi^T\right] dV \left[\frac{\partial \alpha^i}{\partial t}\right] + \int_S [\phi]\, \hat{F}_i dV - \int_\Omega \left[\frac{\partial \phi}{\partial x}\right] (F_i + \mathcal{F}_i)\, dV = 0 \tag{3.49}$$

The final equation in matrix form is obtained as

$$\int_\Omega \left[\phi\phi^T\right] dV \left[\frac{\partial \alpha^i}{\partial t}\right] + \int_S [\phi]\, \hat{F}_i dV - \int_\Omega \left[\frac{\partial \phi}{\partial x}\right] F_i dV - \int_\Omega \left[\frac{\partial \phi}{\partial x}\right] \mathcal{F}_i dV = 0 \tag{3.50}$$

Note that if the correction term (last LHS term) is removed, Eqn. (3.50) is reduced to the classical DG formulation. The matrix form can further simplified as in Eqn. (3.41) and eventually derived to a more generalized form as given in Eqn. (3.46).

3.2.4 Time Integration

Equation 3.46 provides us with the first order time derivative of appropriate accuracy of the scheme. To maintain the accuracy of the scheme, the order of accuracy of the temporal scheme should match that of the spatial scheme. Higher order time integrators should hence be used to yield a stable and accurate result. From Eqn. (3.46), the global operator for the scheme can be expressed in a generalized from as

$$\frac{d\,[\psi]}{dt} = -\,[R]\,[\psi] \tag{3.51}$$

where $[R]$ is the global coefficient matrix obtained by assembling the coefficient matrix $[\mathcal{M}_R]$ and ψ is any arbitrary set of coefficients. Equation 3.51 is marched using a low storage m-stage m^{th} order Strong Stability Preserving Runge Kutta (SSP-RK) [75, 76] scheme as detailed in Chapter 2. For a Linear hyperbolic equa-

tion, since R is a constant, the time marching schemes represented by equation 2.21 can be combined as

$$[\psi]^{n+1} = [G]\,[\psi]^n \tag{3.52}$$

where $[G]$ is called the amplification matrix. The Von Neumann condition states that for a stable scheme the eigenvalues of G should lie within a circle of unit radius ($|G| < 1$). However, this is only a necessary condition. For the scheme to be completely stable, the the eigenvalues of R should not lie in the positive real axis.

3.2.5 Numerical Dispersion Relation

3.2.5.1 Dispersion relation of a numerical scheme

The wave propagation characteristics of the scheme is analyzed by establishing the dispersion relation of the underlying numerical formulation. Here, we follow the methodology adopted by Abeele et al. [82] and Hu et al [83]. The linear advection of a wave is governed by the 1D scalar hyperbolic equation given by Eqn. (3.2) with a constant wave speed a. The element solution in the proposed numerical scheme depends only on its immediate neighbors. Now, consider a numerical scheme in which the solution is expanded in terms of the basis functions and its corresponding coefficients as

$$Q_i = [\phi]^T \left[\alpha^i\right] \tag{3.53}$$

The corresponding spatial derivative is approximated as

$$\frac{\partial Q_i}{\partial x} = \sum_{r=-1}^{1} [D_r] \left[\alpha^{i+r}\right] \tag{3.54}$$

where D_{-1}, D_{+1} and D_0 are the spatial operators acting on the left and right neighbors and the current element respectively. The form of D_r is considered to be arbitrary. Substituting Eqn. (3.53) and Eqn.(3.54) in the basic hyperbolic equation

given in Eqn. (3.2) with unit wave speed, we get

$$[\phi]^T \left[\frac{\partial \alpha^i}{\partial t}\right] + \sum_{r=-1}^{1} [D_r] \left[\alpha^{i+r}\right] = 0 \tag{3.55}$$

Based on the numerical scheme employed, Eqn. (3.55) can be rewritten as

$$\frac{d\alpha^i}{dt} + \sum_{r=-1}^{1} M_r \alpha^{i+r} = 0 \tag{3.56}$$

Here, the brackets are omitted for clarity. The solution coefficients are assumed to be harmonic of the form:

$$\alpha_l^i = q_l \mathrm{e}^{j(kx_i - \omega t)} \tag{3.57}$$

Substituting this harmonic solution in Eqn. (3.56), we get

$$\sum_{l=1}^{p+1} \left(-j\omega + \sum_{r=-1}^{1} M_r \mathrm{e}^{j(krh)}\right) \phi_l = 0 \tag{3.58}$$

or,

$$[\phi_l]^T \left[-j\omega + \sum_{r=-1}^{1} M_r \mathrm{e}^{j(krh)}\right] = 0 \tag{3.59}$$

The solution of 3.59 is non trivial only when the determinant of the matrix is zero. With this condition, the dispersion relation can be written as an eigenvalue problem as

$$det\left(-j\omega + \sum_{r=-1}^{1} M_r \mathrm{e}^{j(krh)}\right) = 0 \tag{3.60}$$

The formulation of dispersion relation does not assume any solution approximation and is valid for any numerical scheme. To apply this formulation for a given numerical scheme, one needs to provide the matrix coefficient set M_r in Eqn. (3.56). For an exact solution, the dimensions of M_r is infinite as the solution can represent all wave numbers. However, in the numerical scheme, we have a finite set of basis functions and corresponding coefficients required for approximating the solution in Eqn. (3.53). This results in a finite form of coefficient matrix M_r. On comparison with Eqn. (3.46), we can see that this coefficient matrix M_r is synonymous with

the coefficient matrix $\mathcal{M}_R^r$.

The dispersion relation of the numerical scheme is obtained by comparing the numerical wave number (Eqn. 3.60) with the analytical wave number (ω/a). For a stable scheme, the real part of the Fourier foot print $-j\omega$ should be negative for all real values of k. The solution of 3.59 would have $(p+1)$ values of $j\omega$, where p is the degree of solution representation. These solutions are then mapped [84] to the numerical wave number as

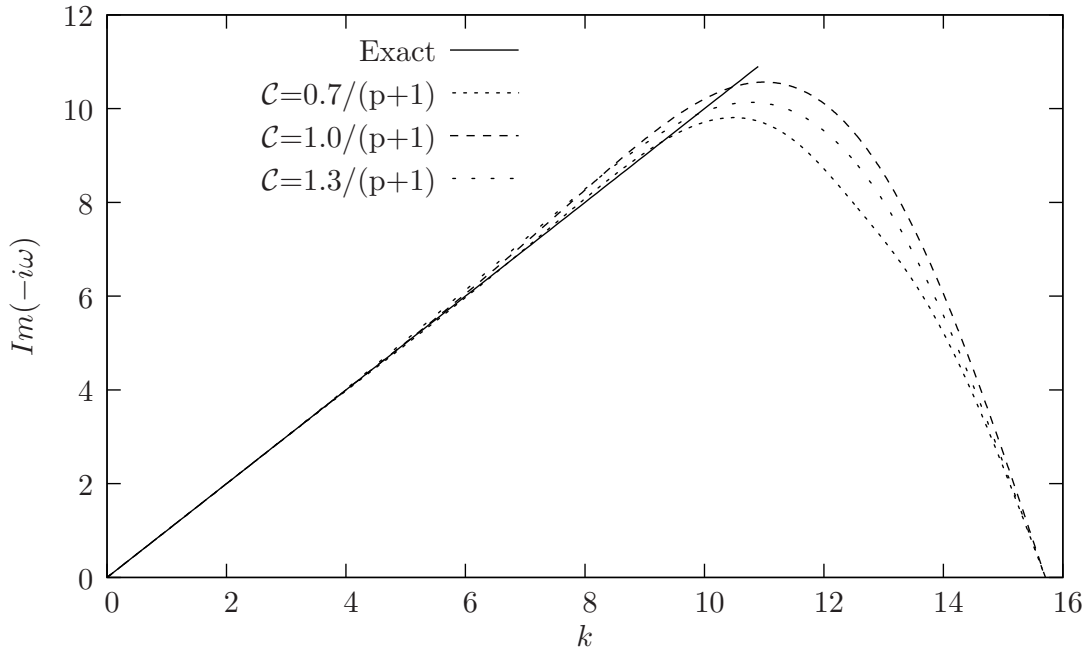
$$k = 2\pi i + k_i i = 0, 1, \ldots, p$$

where k_i is the i^{th} solution of 3.59. The real part of k represents the dispersion and the imaginary part represents the diffusion. For a scheme to be stable, the imaginary part should not be greater than zero. For numerical formulation, we take the number of solution and flux reconstruction points as $N_Q = p + 1$ and $N_F = p + 2$ respectively.

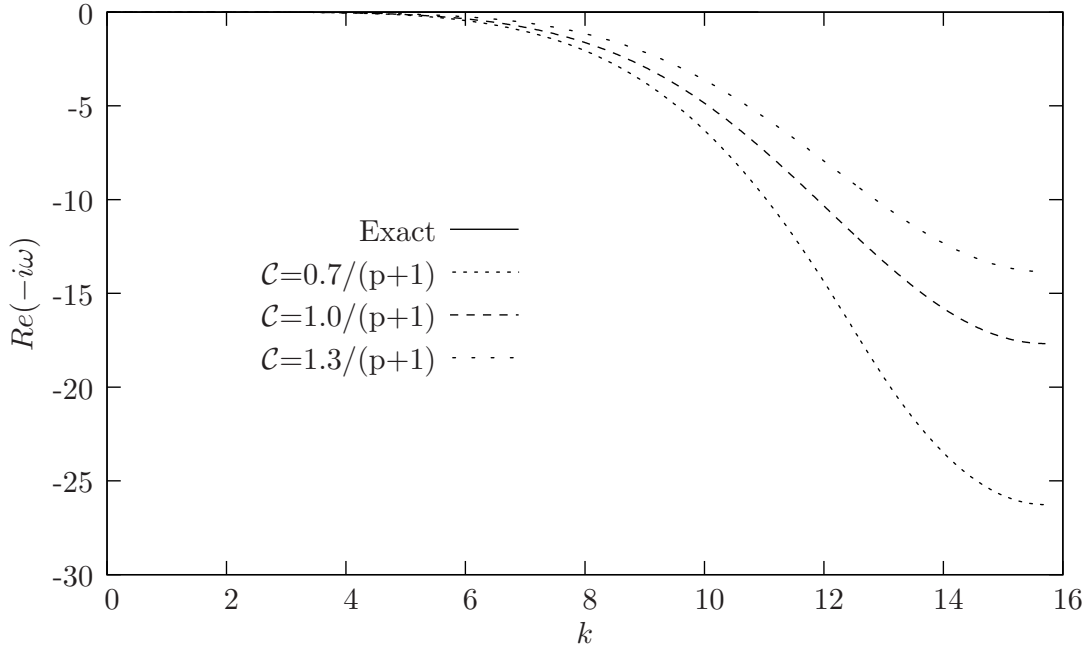
3.2.5.2 Wave propagation characteristics of 1D ExRi schemes

The influence of the maximum extent of wave propagation $\mathcal{C}$ on the wave propagation characteristics of a 5^{th} order ($p = 4$) scheme is shown in Figure 3.5. The scheme has similar dispersion characteristics for different values of $\mathcal{C}$. However, a significant change is observed in the diffusion characteristics. With increase in $\mathcal{C}$, the scheme becomes less diffusive and the Fourier footprint ($-j\omega$) becomes smaller, resulting in an increase in CFL condition. A plot of $\mathcal{C}$ and the corresponding CFL limit for different orders of the scheme is shown in Figure 3.6. The CFL limit has almost a linear relationship with $\mathcal{C}$ for different orders of the scheme. Various numerical experiments suggest that the numerical schemes have a optimal accuracy and stability for a value of $\mathcal{C}$ given by

$$\mathcal{C} = \frac{1}{p+1} \tag{3.61}$$



(a) Dispersion Characteristics



(b) Diffusion Characteristics

Figure 3.5: Influence of $\mathcal{C}$ on wave propagation characteristics of a 5^{th} order ERC scheme ($\lambda = 1$).

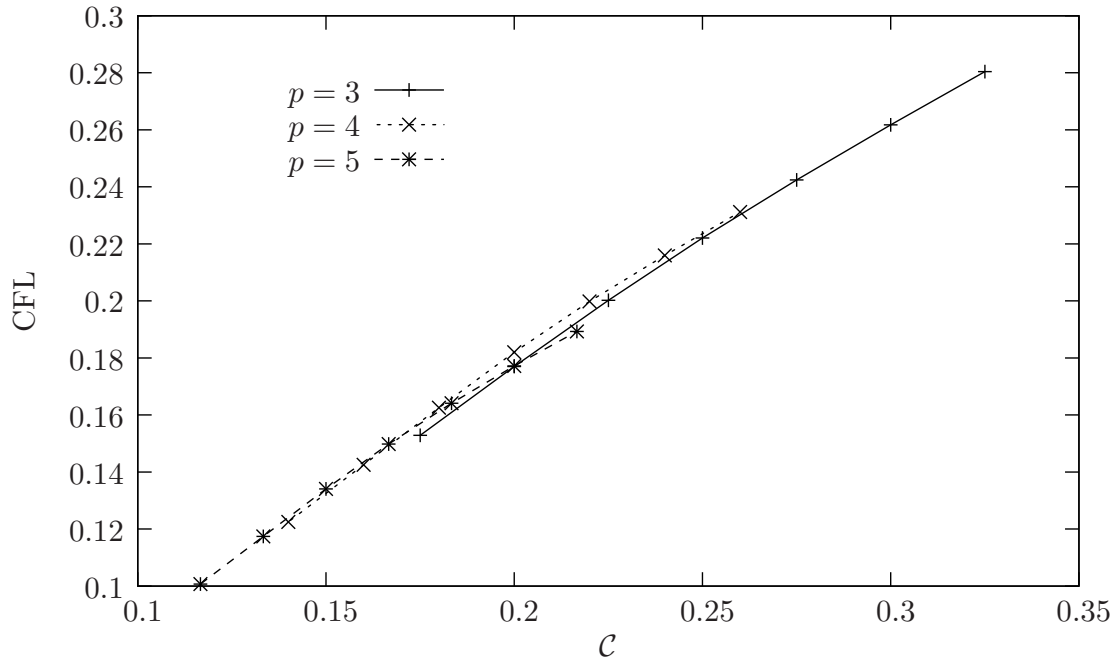


Figure 3.6: Relation of $\mathcal{C}$ and CFL for ERC schemes.

p	1	2	3	4	5	6	7	8
λ	1.00	1.00	1.00	1.00	1.00	1.2683	1.2514	1.1013

Table 3.1: Values of λ for stable 1D ERC schemes

This value corresponds to the maximum CFL limit of a numerical scheme with $(p+1)$ degrees of freedom and backward Euler time integration.

The stability of the ERC scheme is dictated by the parameter λ introduced in Eqn. (3.26). A minimum value of the λ is estimated for which the scheme is stable. Figure 3.7 shows the effect of the point distribution on the stability of the ERG scheme. The values of λ obtained for different orders are listed in Table 3.1. The unmodified Chebyshev points ($\lambda = 1$) results in a stable scheme for orders up to 6 ($p = 5$). For higher orders, the points are clustered towards the element boundary (by varying λ) to make the scheme stable.

The dispersion relation of a fourth order scheme ($p = 3$) is shown in Figure 3.8. Due to symmetry, only half of the actual representable wave space is presented. The ERC scheme seems to have a lower dispersion error than the ERG and DG schemes. The ERG and DG schemes tend to be more diffusive for larger wave numbers. It can be inferred that the noise due to the unresolved (larger) wave

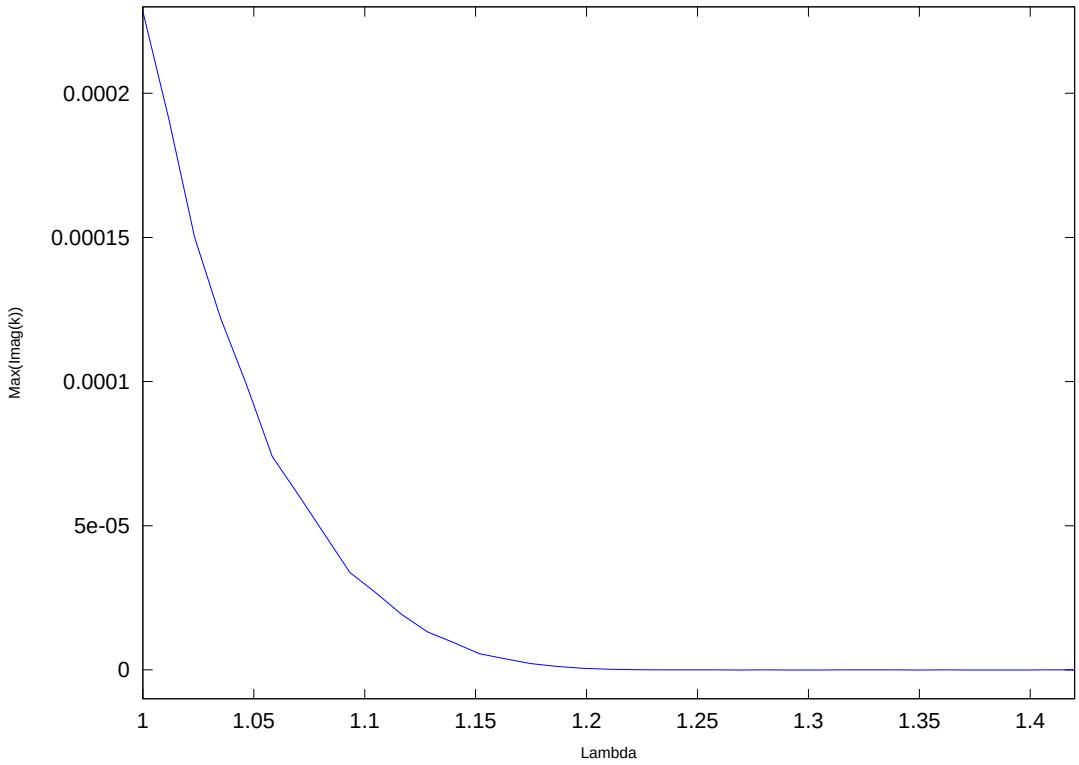


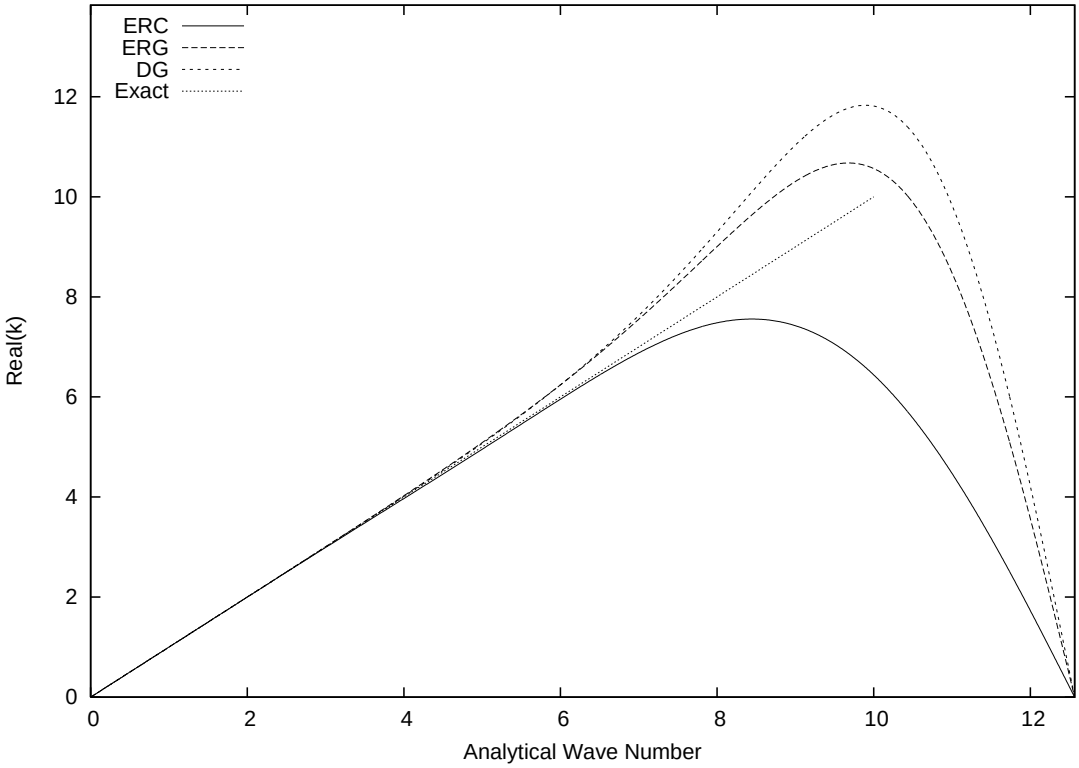
Figure 3.7: Effect of λ on stability of a seventh order scheme.

numbers are filtered out more efficiently in the Galerkin schemes. However, it is also observed that these schemes have a lower CFL limit compared to the ERC schemes.

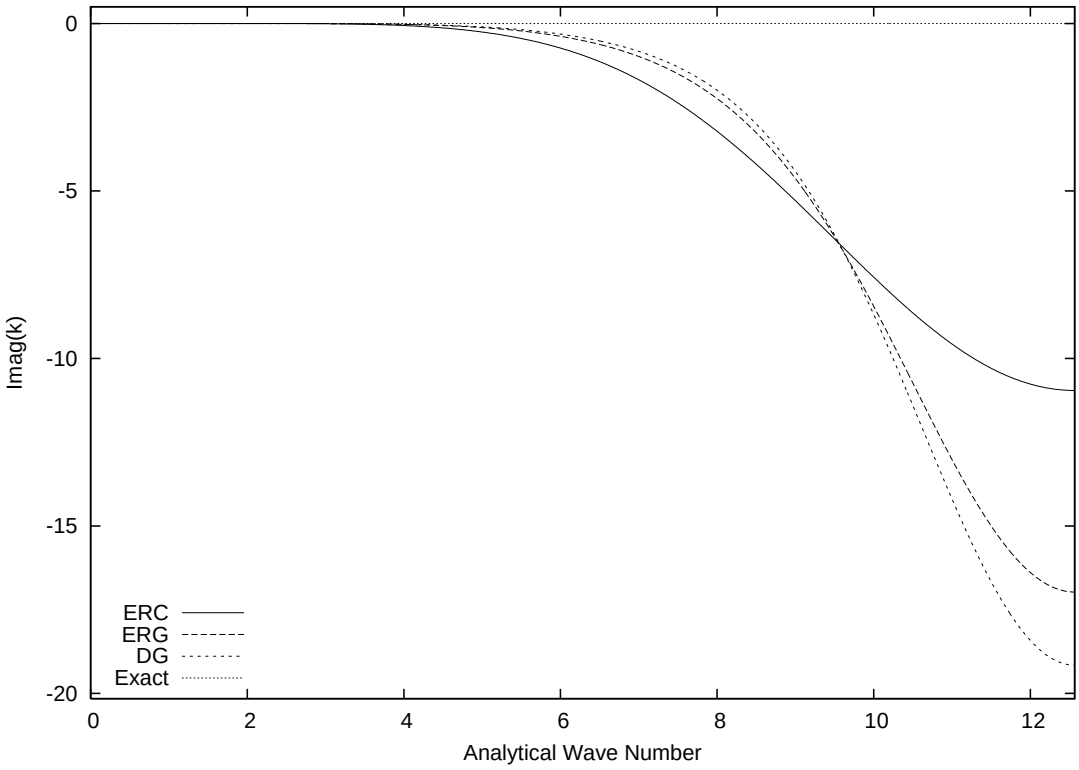
p	ERC	ERG	DG
1	0.500	0.378	0.333
2	0.347	0.231	0.210
3	0.234	0.164	0.145
4	0.197	0.133	0.118
5	0.169	0.106	0.094

Table 3.2: CFL condition for higher order schemes

The dispersion and diffusion characteristics of ERC schemes of different orders are plotted in Figure 3.9. The increase in resolution of the higher order scheme in terms of wave numbers is obvious in Figure 3.9. To estimate the corresponding CFL condition, the properties of the amplification matrix G in Eqn. (3.52) is analyzed. For a stable spatio-temporal scheme, the eigenvalues of G should lie within a unit circle ($|G| \leqslant 1$). A plot of G in eigen space is shown in Figure 3.10. We also compare the matrix $CFL\,|R|$ with the exact roots of the time integrator. The

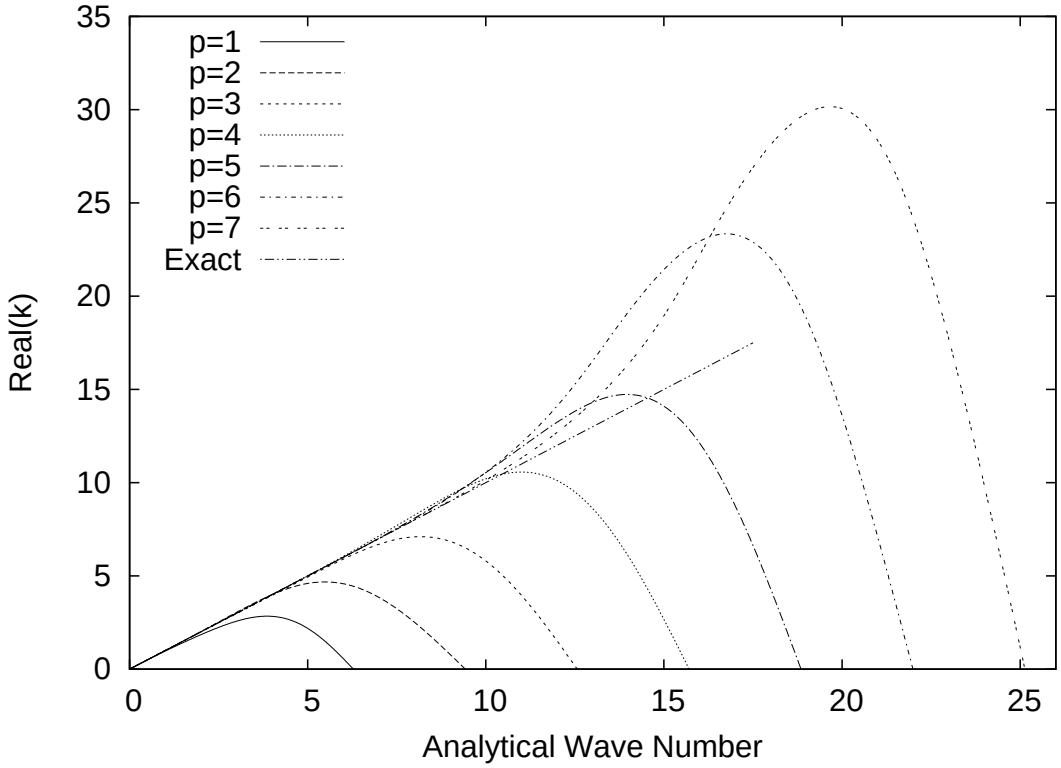


(a) Dispersion Characteristics

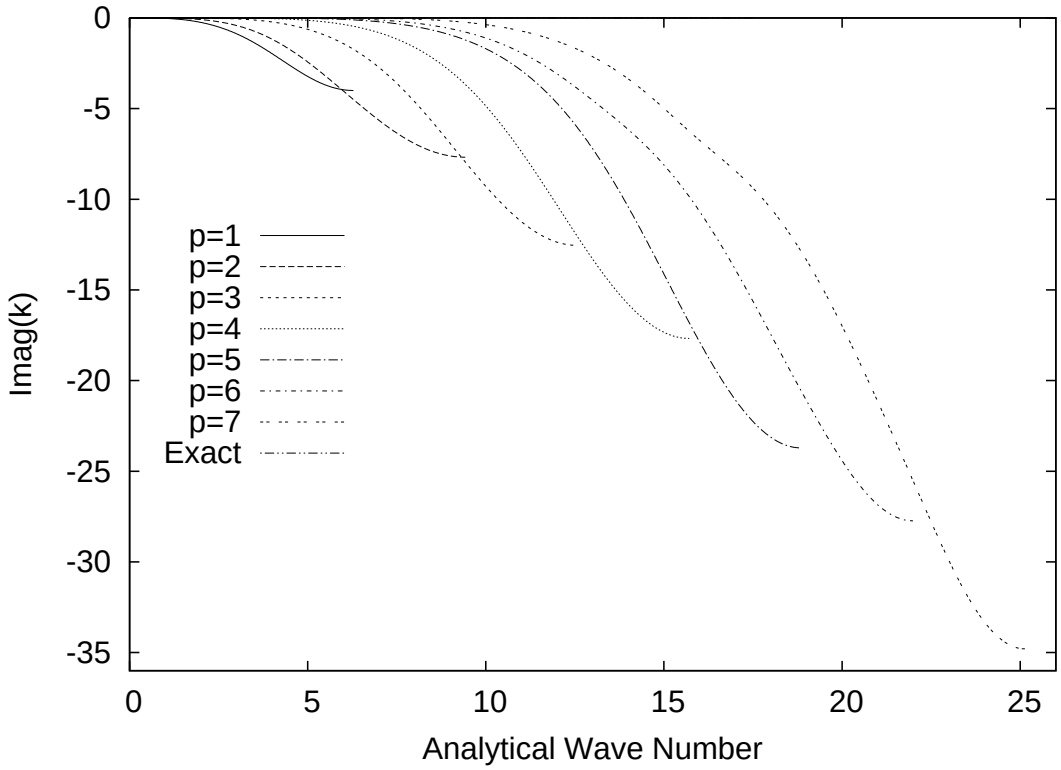


(b) Diffusion Characteristics

Figure 3.8: Comparison of dispersive and diffusive properties for fourth order schemes.



(a) Dispersion Characteristics



(b) Diffusion Characteristics

Figure 3.9: Variation of dispersive and diffusive properties of ERC schemes for different orders.

eigenvalues of the ERC scheme lies within that of the time integrator itself. The maximum CFL limit is chosen in obtaining the plot in Figure 3.10. The optimal CFL conditions obtained for different schemes are tabulated in Table 3.2.

It is known that the DG schemes have a stricter stability criteria of $CFL = 1/(2p+1)$ [85] for schemes of order $(p+1)$. The CFL limit of ERG schemes reported in Table 3.2 is around 10% higher than the DG schemes. The ERC schemes seem to have a more relaxed stability limit close to $CFL = 1/(p+1)$.

3.2.6 Numerical Tests

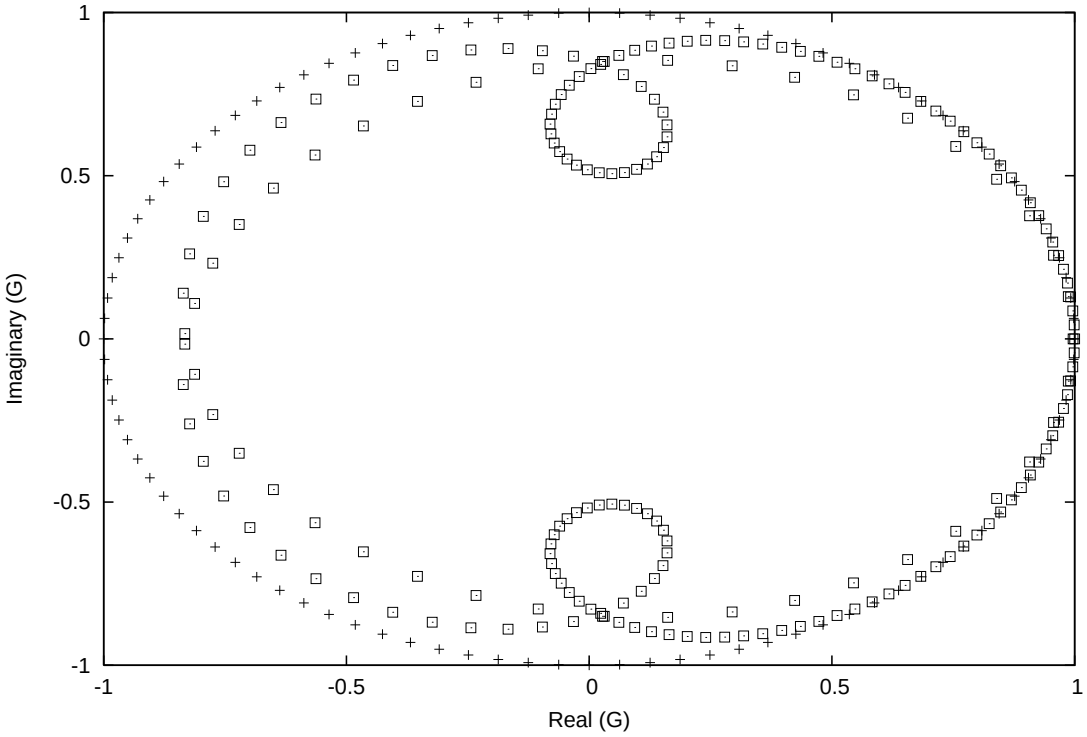
Numerical tests are conducted for two classical problems of linear advection and non-linear advection. The order of accuracy (grid convergence) is numerically tested for both the test cases. To reduce the error due to temporal discretization, we choose a very small CFL number of $0.1/(2p+1)$. Since ERC scheme has better stability characteristics than the ERG scheme, we study only the ERC scheme in detail.

For the linear advection equation we take a unit wave speed ($a = 1$). The solution is initialized with a Gaussian pulse of half width 0.1. The errors and orders of convergence for the linear advection problem at time $t = 1$ are given in Table 3.3. The results confirm that the scheme exhibits the designed order of accuracy of $(p+1)$ and converge to machine accuracy. The decrease in order of accuracy in case of 6^{th} and 7^{th} order schemes is due to the limit of machine accuracy.

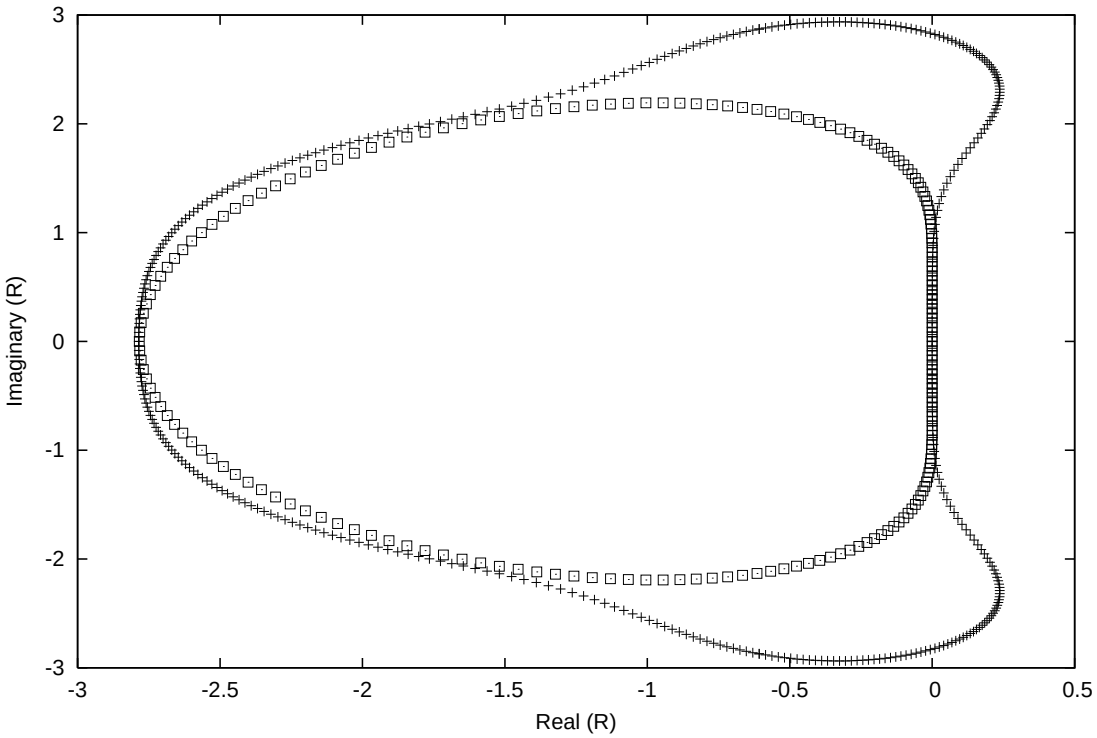
The non-linear Burger's equation can be obtained by setting the wave speed to $a = Q$ and hence, the flux becomes $F = Q^2/2$. The solution is initialized with

$$Q(0,x) = \frac{1}{4} + \frac{1}{2}\sin(2\pi x) \tag{3.62}$$

The equation is solved in a periodic domain of $0 \leqslant x \leqslant 1$. Due to the different wave speeds, the solution eventually develops a shock at $t = 1/\pi$. We compare the solutions with the analytical solution at time $t = 0.15$ when there is no occurrence



(a) Plot of $|G|$



(b) Plot of $|CFL \cdot R|$

Figure 3.10: Plot of $|G|$ in complex plane for 4^{th} order ERC schemes.

p	NCells	L_1 error	$L_1 order$	L_∞ error	$L_\infty order$
1	10	1.06e-01		2.96e-01	
	20	7.15e-02	0.56	3.03e-01	-0.03
	40	3.63e-02	0.98	2.50e-01	0.27
	80	1.12e-02	1.70	1.06e-01	1.25
	160	2.05e-03	2.45	2.37e-02	2.15
	320	3.03e-04	2.76	3.81e-03	2.64
2	10	8.93e-02		2.70e-01	
	20	4.73e-02	0.92	1.95e-01	0.47
	40	1.52e-02	1.64	1.13e-01	0.78
	80	2.53e-03	2.58	2.73e-02	2.05
	160	3.21e-04	2.98	3.83e-03	2.84
	320	3.97e-05	3.02	4.78e-04	3.00
3	10	5.38e-02		1.92e-01	
	20	1.34e-02	2.01	6.02e-02	1.67
	40	8.69e-04	3.95	7.38e-03	3.03
	80	3.04e-05	4.84	3.32e-04	4.47
	160	1.46e-06	4.38	1.68e-05	4.31
	320	8.35e-08	4.13	9.58e-07	4.13
4	10	6.07e-02		1.87e-01	
	20	1.28e-02	2.24	4.73e-02	1.98
	40	2.43e-04	5.72	1.81e-03	4.71
	80	4.11e-06	5.89	4.41e-05	5.36
	160	1.61e-07	4.68	1.92e-06	4.52
	320	5.29e-09	4.93	6.42e-08	4.90
5	10	2.42e-02		6.50e-02	
	20	8.96e-04	4.76	2.94e-03	4.47
	40	1.46e-05	5.94	1.17e-04	4.65
	80	1.80e-07	6.34	2.02e-06	5.86
	160	2.38e-09	6.24	2.63e-08	6.26
	320	3.57e-11	6.06	4.18e-10	5.97
6	10	1.58e-02		5.04e-02	
	20	5.77e-04	4.78	3.29e-03	3.94
	40	6.34e-06	6.51	5.06e-05	6.02
	80	3.82e-08	7.37	4.39e-07	6.85
	160	2.26e-10	7.40	2.62e-09	7.39
	320	2.65e-11	3.09	2.47e-10	3.41

Table 3.3: Linear advection of Gaussian pulse

p	NCells	L_1 error	$L_1 order$	L_∞ error	$L_\infty order$
1	10	6.92e-03		1.44e-02	
	20	1.86e-03	1.89	4.07e-03	1.82
	40	4.54e-04	2.04	1.10e-03	1.89
	80	1.11e-04	2.04	2.88e-04	1.93
	160	2.71e-05	2.03	7.33e-05	1.97
	320	6.72e-06	2.01	1.84e-05	1.99
2	10	1.90e-03		8.15e-03	
	20	2.20e-04	3.11	1.19e-03	2.77
	40	2.48e-05	3.15	1.60e-04	2.90
	80	2.85e-06	3.12	1.92e-05	3.05
	160	3.36e-07	3.08	2.38e-06	3.01
	320	4.08e-08	3.04	2.94e-07	3.02
3	10	2.38e-04		9.97e-04	
	20	2.10e-05	3.50	1.40e-04	2.84
	40	1.33e-06	3.98	9.97e-06	3.81
	80	8.24e-08	4.01	6.87e-07	3.86
	160	4.98e-09	4.05	4.40e-08	3.96
	320	3.04e-10	4.03	2.71e-09	4.02
4	10	2.39e-04		1.27e-03	
	20	9.13e-06	4.71	7.76e-05	4.04
	40	2.85e-07	5.00	3.37e-06	4.52
	80	8.50e-09	5.07	1.14e-07	4.89
	160	2.43e-10	5.13	3.28e-09	5.12
	320	7.05e-12	5.11	9.42e-11	5.12
5	10	2.76e-05		1.62e-04	
	20	6.72e-07	5.36	5.53e-06	4.87
	40	1.30e-08	5.70	1.72e-07	5.01
	80	2.34e-10	5.79	3.01e-09	5.83
	160	3.72e-12	5.98	4.86e-11	5.95
	320	3.45e-13	3.43	9.65e-13	5.65
6	10	1.69e-05		1.23e-04	
	20	2.74e-07	5.95	2.13e-06	5.85
	40	2.63e-09	6.70	3.83e-08	5.80
	80	3.83e-11	6.10	3.69e-10	6.70
	160	9.19e-12	2.06	2.00e-11	4.21
	320	5.63e-12	0.71	1.06e-11	0.92

Table 3.4: Burger's Equation at $t = 0.15$

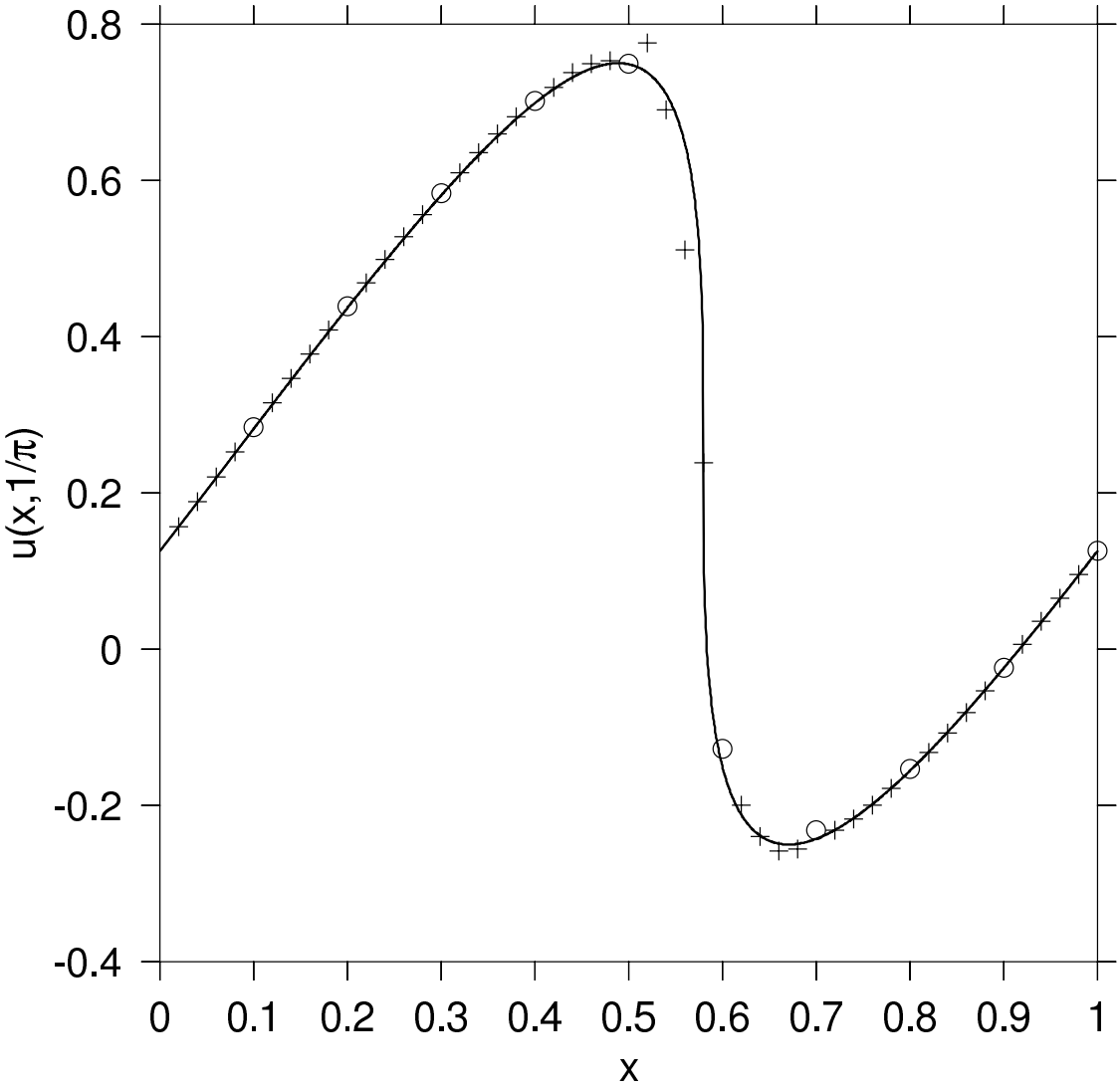


Figure 3.11: Numerical solution of Burger's equation at $t = 1/\pi$ for a 4^{th} order scheme with 10 cells. ($+$: Internal points, $\circ$: Boundary points, $-$: Exact Solution).

of shock in the entire region. The errors and order of convergence are given in Table 3.4. It can be seen that the scheme performs well and the designed order of accuracy is achieved.

For the given initial condition in Eqn. (3.62), the Burger's equation develop a shock at time $t = 1/\pi$. We compare the numerical solution with the analytical solution at $t = 1/\pi$. The simulation is performed for a 4^{th} order scheme with 10 cells and the results are plotted in Figure 3.11. In smooth regions, the numerical solution match the exact solution very well. The solution near the shock exhibits slight oscillations as the entire shock is resolved by just two cells. No shock capturing

schemes are used for current computations.

3.3 Implementation in a generic 2D triangle element

The proposed Extended Riemann solver is applied on 2D triangular elements. We adopt the same notation as in the one dimensional formulation. The multi-dimensional conservation law can be written as

$$\frac{\partial Q}{\partial t} + \nabla \cdot \vec{F} = 0 \tag{3.63}$$

where Q is the solution and $\vec{F} = (F^x(Q), F^y(Q))$ is the flux vector. Eqn. (3.63) can be expanded in an explicit form as

$$\frac{\partial Q}{\partial t} + a_x \frac{\partial Q}{\partial x} + a_y \frac{\partial Q}{\partial y} = 0 \tag{3.64}$$

where a_x and a_y are given by

$$\begin{aligned} a_x &= \frac{\partial F^x}{\partial Q} \\ a_y &= \frac{\partial F^y}{\partial Q} \end{aligned} \tag{3.65}$$

The quantities a_x and a_y provide the direction and magnitude of wave propagation at a given time instance.

For an element i, the solution is represented in an discretized form using the polynomial expansion Q_i given by

$$Q_i = \sum_j \alpha_j \phi_j = [\phi]^T [\alpha] \tag{3.66}$$

where α_j is the coefficient of the basis function $\phi_j(\xi, \eta)$. The conservation property is enforced similar to the 1D formulation by applying the constraints Eqn. (3.31-

3.32). Here, we choose a simple set of basis functions given by

$$[\phi] = \left[\; 1, \;\; \xi - \tfrac{1}{6}, \;\; \eta - \tfrac{1}{6}, \;\; \xi^2 - \tfrac{1}{12}, \;\; \xi\eta - \tfrac{1}{24}, \;\; \eta^2 - \tfrac{1}{24}, \;\; \xi^3 - \tfrac{1}{20}, \;\; \ldots \; \right] \tag{3.67}$$

The evolution of solution Q_i of element i is influenced by the internal flux and the flux contributions from the boundaries. The temporal evolution of the solution Q_i can be expressed as

$$\frac{\partial Q_i(\xi,\eta)}{\partial t} + \nabla \cdot \tilde{F}_i(\xi,\eta) = 0 \tag{3.68}$$

where $\partial Q_i/\partial t$ is the solution evolution of element i, and $\tilde{F}_i$ is the approximate flux given by

$$\tilde{F}_i\left(\xi,\eta\right) = \vec{F}_i\left(\xi,\eta\right) + \sum_{j=1}^{3} \vec{\mathcal{F}}_i^j\left(\xi,\eta\right) \tag{3.69}$$

where $\vec{F}_i$ is the internal flux and $\vec{\mathcal{F}}_i^S$ is the boundary contributions corresponding to boundary S of the triangle. Eqn. (3.68) and Eqn. (3.69) can be expressed explicitly in terms of internal and boundary contributions as

$$\frac{\partial Q_i\left(\xi,\eta\right)}{\partial t} = \frac{\partial Q_i^*\left(\xi,\eta\right)}{\partial t} + \sum_{j=1}^{3} \frac{\partial \varepsilon_i^j\left(\xi,\eta\right)}{\partial t} \tag{3.70}$$

where

$$\frac{\partial Q_i^*\left(\xi,\eta\right)}{\partial t} + \nabla \cdot \vec{F}_i(\xi,\eta) = 0 \tag{3.71}$$

$$\frac{\partial \epsilon_i^j\left(\xi,\eta\right)}{\partial t} + \nabla \cdot \vec{\mathcal{F}}_i^j(\xi,\eta) = 0 \tag{3.72}$$

Since the internal flux information $\vec{F}_i$ is readily available, Eqn. (3.71) can be solved by a standard Galerkin method which is exact if Eqn. (3.63) is linear. For the boundary contributions given by Eqn. (3.72), the approximate flux $\vec{\mathcal{F}}_i$ is required. The formulation of $\vec{\mathcal{F}}_i$ is given in following section.

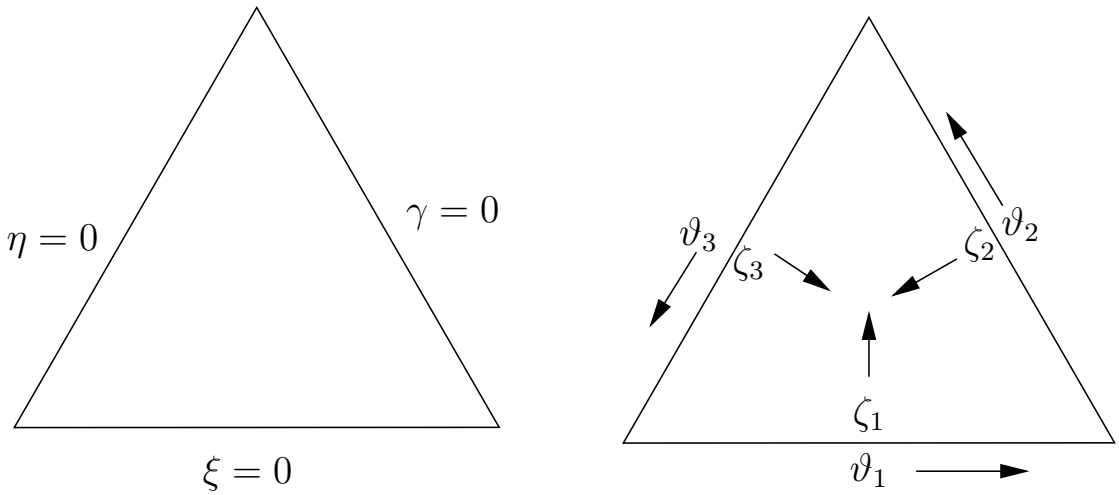


Figure 3.12: Reference Coordinates of Triangle element.

3.3.1 Representation of Boundary flux contributions

Similar to the 1D formulation, the distance of propagation of an information from the interface S into the element interior over a time period of Δt is given by $\mathcal{C}h$ where h is the size of the element. In the case of triangles, the elements are mapped to a reference element ($h = 1$) as shown in Figure 3.12. The interior of the reference element can now be divided into two regions: Boundary region Ω^S influenced by the boundary interfaces S and the interior regions Ω^{in} not influenced by the interfaces. The boundary flux corrections $\vec{\mathcal{F}}$ should vanish for the interior regions. For simplicity, we neglect the vector notation $\vec{[]}$ for the correction term $\mathcal{F}$ and it is considered to be a vector implicitly. The properties of the boundary correction is now given by

$$\frac{\partial^n \mathcal{F}}{\partial \xi^{n-i} \partial \eta^i} = 0 \qquad (\xi, \eta) \in \Omega^{in},\ i = 0...n,\ n = 0...(p+1) \tag{3.73}$$

$$\mathcal{F} = \hat{F}_S \qquad (\xi, \eta) \in S,\ i = 0...n,\ n = 0...(p+1) \tag{3.74}$$

where $\hat{F}$ is the Riemann flux at the interface. For an exact representation of $\mathcal{F}$ in Eqn. (3.74), multi-dimensional Riemann solutions need to be evaluated at boundary integration/interpolation points. In the current work, we consider only the

corrections normal to the face, thereby avoiding additional computational cost of 2D Riemann solvers. A simplified formulation similar to one dimensional formulation is used for the flux correction $\mathcal{F}$. Here, we introduce a coordinate $\vartheta_S(\xi,\eta)$ as the local coordinate along the interface S. The representation of $\vartheta_S(\xi,\eta)$ is shown in Figure 3.12. The Riemann fluxes at an interface S can now be represented as $\hat{F}_S(\vartheta_S(\xi,\eta))$. The flux correction $\mathcal{F}^S$ due to the Riemann problem at the interface S is now given by

$$\mathcal{F}_i^S(\xi,\eta) = \omega(\zeta_S)\left(\hat{F}_S(\vartheta_S) - F_i(\xi_S,\eta_S)\right) \tag{3.75}$$

where ω is a function for satisfying the condition 3.73 and (ξ_S,η_S) are the coordinates corresponding to ϑ_S on the interface S. The coordinates ϑ and (ξ_S,η_S) are used to restrict the flux computations to the element interface S. This is required as the number of interior points required to solve Eqn. (3.72) can be large for higher order schemes. Solving a Riemann problem at each point will result in increase in computational cost. The blending function ω is formulated such that at a distance of $\mathcal{C}$ from the boundary S, the corresponding flux quantity $\mathcal{F}^S$ vanishes. Though various choices exist for formulation of ω, here we use a simple polynomial form similar to that of a 1D formulation. The blending function is given by

$$\omega(\zeta) = \begin{cases} \zeta^{p_\omega} & 1 \geqslant \zeta \geqslant 0 \\ 0 & \zeta \leqslant 0 \end{cases} \tag{3.76}$$

where the local coordinates of the interface are given by

$$
\begin{aligned}
\zeta_1 &= 1 - \frac{\eta}{\mathcal{C}} \\
\zeta_2 &= 1 - \frac{\gamma}{\mathcal{C}} \\
\zeta_3 &= 1 - \frac{\xi}{\mathcal{C}} \\
\vartheta_1 &= \frac{\xi}{1-\eta} \\
\vartheta_2 &= \frac{\eta}{1-\gamma} \\
\vartheta_3 &= \frac{\eta}{1-\xi}
\end{aligned}
\tag{3.77}
$$

Though the definition of the coordinates ζ and ϑ in Eqn. (3.77) do not align with the corresponding face normals in physical space, these approximations result in desired stability and grid convergence characteristics. Also, since we use only the non-dimensionalized quantity $\mathcal{C}$, there is no requirement of transformation of wave speeds from physical to reference plane.

In Eqn. (3.75), approximation of $\mathcal{F}_i^S$ requires evaluation of the Riemann correction for each internal point. To avoid the costly computation of Riemann fluxes, this quantity is approximated with an interpolating polynomial f_S representing the Riemann corrections $\mathcal{F}_i^S$ on the interface S. The polynomial is constructed using the available boundary integration/interpolation points. The rest of the values are interpolated using f_S. With this formulation, no additional computation is required in solution evolution process as it requires flux evaluations only at the integration/interpolation points. To maintain the order of accuracy of the scheme, the degree of polynomial f_S should be consistent with the degree of approximation of flux quantities. Since the current formulation is based on the reference element, the maximum distance of propagation can still be represented as $CFLh$ where h is the element size.

3.3.2 Evolution of boundary contributions

The internal flux contributions in Eqn. (3.71) are solved using a standard Galerkin method. For a linear hyperbolic equation, this result in an exact solution of Eqn. (3.71). But, in the case of boundary flux contributions in Eqn. (3.72), approximations are required to represent the correction flux $\mathcal{F}$. The characteristics of the proposed 2D scheme depends on this approximation. Different methods of approximations can be adopted. Here, we choose the sub-domain method (ExRi-SD) as it results in a more stable scheme.

The sub-domain method is similar to co-location and least-squares method where the equations are evaluated at co-location points, except that here, the equations are evaluated at a finite non-overlapping region. This method is chosen due its improved stability characteristics and ease of implementation. The sub-domain method is given by

$$\frac{1}{\mathcal{V}_k}\int_{\mathcal{V}_k}\frac{\partial \epsilon_i^S}{\partial t} + \frac{1}{\mathcal{V}_k}\int_{\mathcal{V}_k}\nabla\cdot\vec{\mathcal{F}}_i^S = 0 \tag{3.78}$$

where $\mathcal{V}_k$ is the k^{th} sub-domain of the element V. For a triangle, these domains are defined by sub-dividing the element into three quadrilaterals, and further sub-dividing them according to the boundary points specifications as shown in Figure 3.13. To ensure the correct order of convergence, the number of sub-regions at the boundary should be at least same as the order of the scheme. For odd orders, we divide the region similar to the next order. If the total number of sub-domains is larger than the number of coefficients used to represent the solution Q_i, the equation set is solved by least squares method.

For the third order scheme, the element is divided in the same way as that of fourth order scheme. With the representation of the correction flux $\vec{\mathcal{F}}_i^j$ with the interpolation polynomial f_S and the blending function given by Eqn. (3.76), the correction quantity ϵ_i^S will depend only on the boundary flux evaluations on interface S. After simplifications, only the coefficients of f_S is required. While solving the equation, only the coefficients of polynomial f_S need to be computed. With the current formulation, no additional computational cost is incurred with

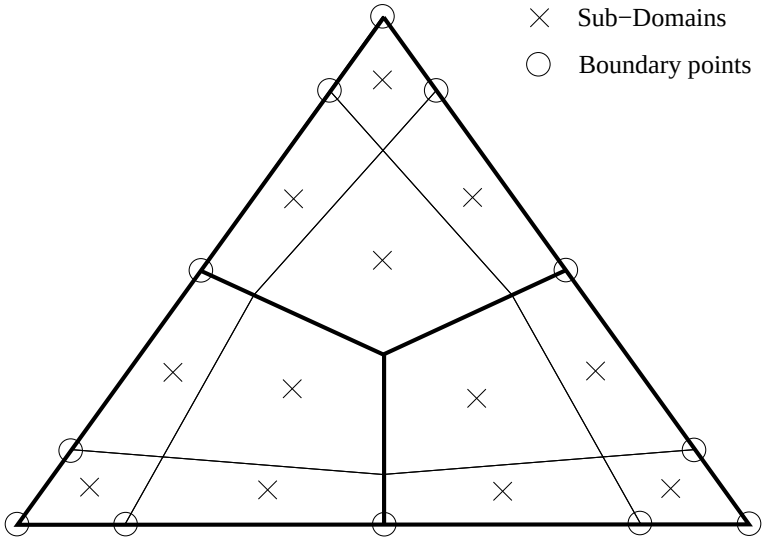
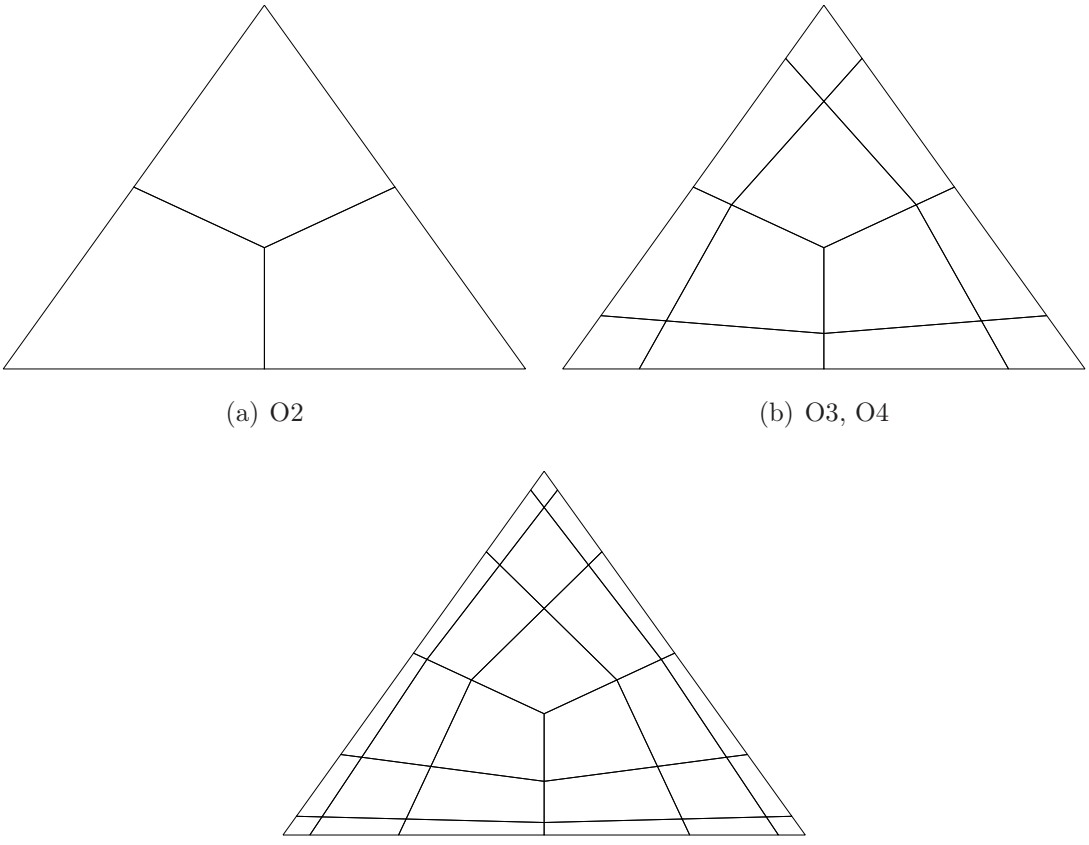


Figure 3.13: Dividing triangles into Sub-Domains with respect to boundary point distribution.



(a) O2

(b) O3, O4

(c) O5

Figure 3.14: Sub domains of a triangle for different orders.

p	2	3	4
λ	1.19	1.3	1.3

Table 3.5: Values of λ for a stable scheme based on Sub-Domain method

the increase in number of sub domains.

As mentioned before, the stability of the scheme depends on the distribution of the sub-domains within the element. The element subdivision algorithm adopted in Figure 3.14 is dependent on the boundary point distribution defined by a modified version of Chebyshev nodes given by

$$x_j = \cos\left(b_j \lambda^{1-|b_j|}\right) \tag{3.79}$$

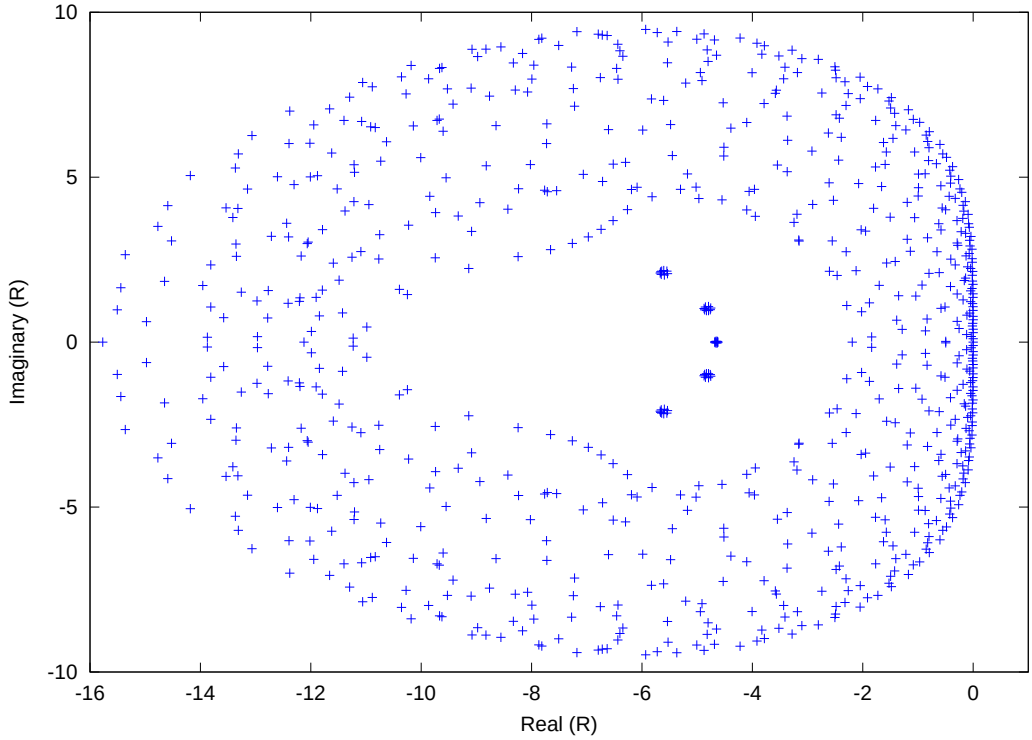
where λ is the scaling factor for the points similar to the 1D formulation and

$$b_j = \left(\frac{2j}{N-1}\right) - 1 \tag{3.80}$$

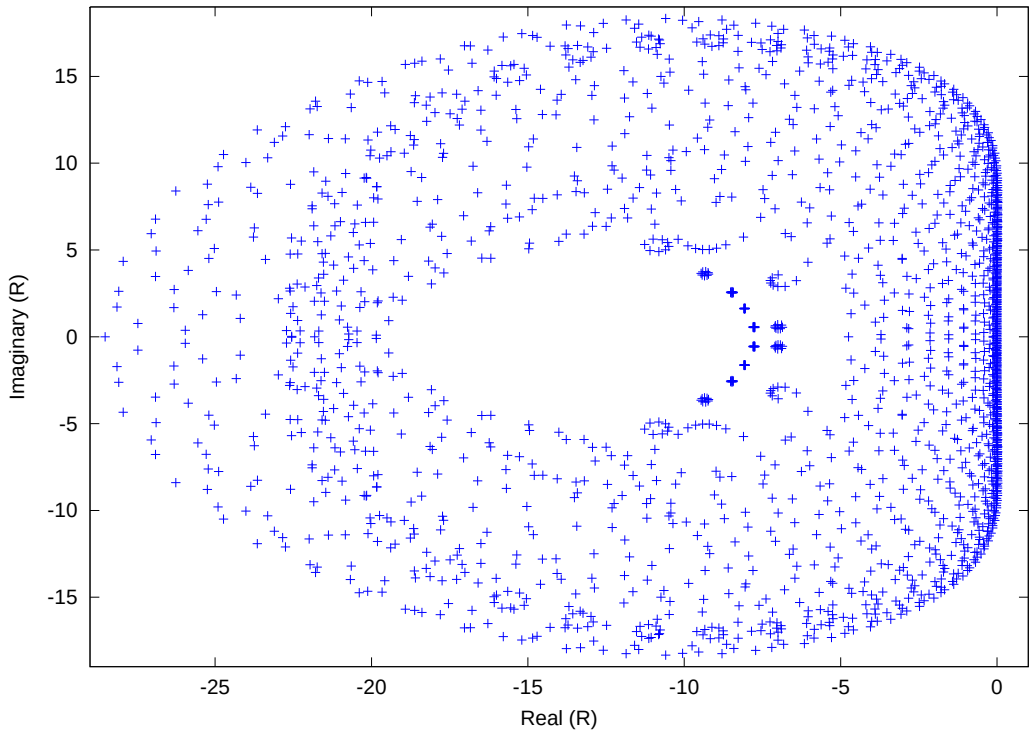
where N is the total number of points on the boundary. As the value of λ increases, the boundary points are clustered towards the corners. This modification is found to make the scheme more diffusive. The optimal value of the parameter λ is obtained from the matrix stability analysis detailed in Section 3.3.3.

3.3.3 Matrix Stability Analysis

A matrix stability analysis is performed by formulating the numerical scheme for a linear advection problem. The final equation set is represented in a matrix form as given in Eqn. (3.52). An unstructured grid of $8 \times 8 \times 2$ triangular elements is used for this analysis. The procedure is similar to that of one-dimensional case and is not detailed here for brevity. The stability of the schemes are analyzed from the eigenvalues of the spatial operator R in Eqn. (3.51) and amplification matrix G in Eqn. (3.52). Unlike one dimensional analysis, the stability of the two dimensional numerical schemes depend on the direction of wave propagation as well. If θ is the



(a) 3^{rd} order



(b) 5^{th} order

Figure 3.15: Representation of spatial operator R in eigen space for ExRi-SD schemes ($\theta = \pi/6$).

p	ExRi-SD	DG	% improvement
2	0.192	0.151	27.2
3	0.138	0.095	45.3
4	0.122	0.084	45.2
5	0.098	0.063	55.5

Table 3.6: Maximum CFL condition for 2D schemes

direction of wave propagation, the quantities a_x and a_y in Eqn. (3.64) are given as

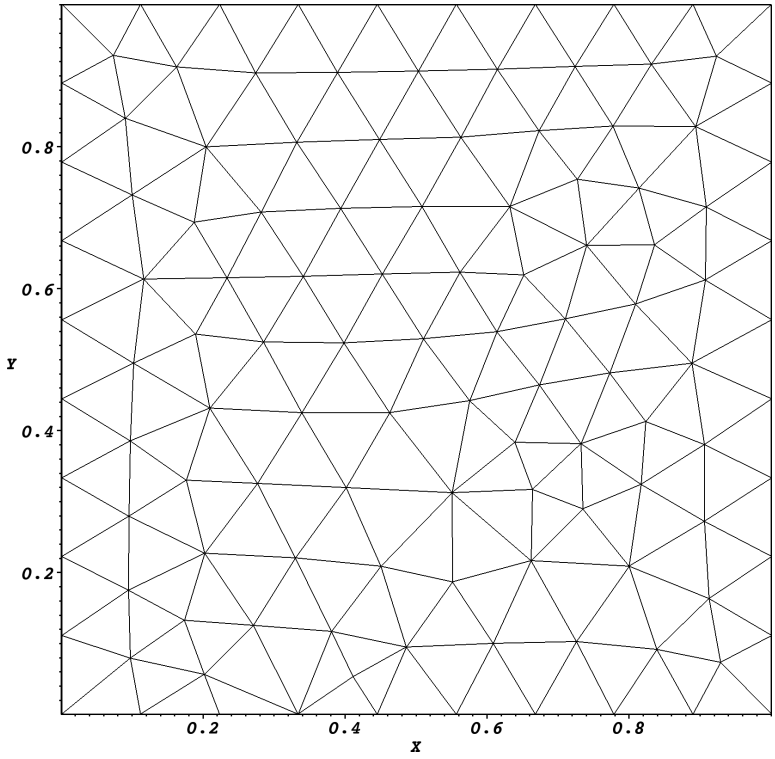
$$(a_x, a_y) = (cos(\theta),\ sin(\theta)) \tag{3.81}$$

For a scheme to be stable, the stability condition ($Real(R) = 0$) should be satisfied for all values of θ. the choice of parameter λ dictates the stability of the proposed Sub-Domain method. The optimal values of λ for different orders are listed in Table 3.5. The plot of spatial operator R in eigen space for 3^{rd} and 5^{th} order schemes are shown in Figure 3.15. The 4^{th} and 5^{th} order ExRi-SD schemes are weakly unstable with some of the eigenvalues of R lying on the positive real axis. However, both the schemes produce the desired order of convergence.

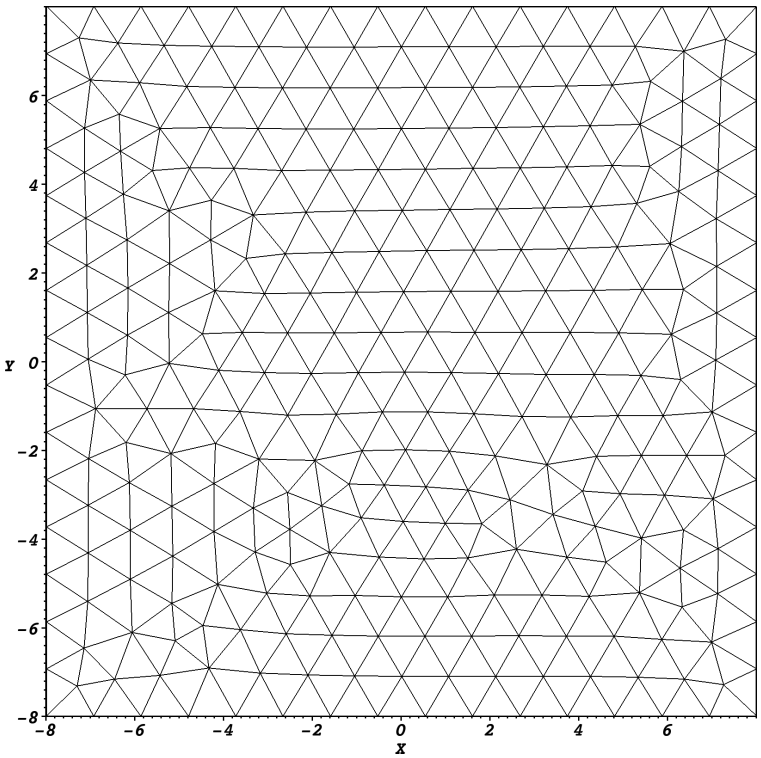
The maximum CFL limit of the schemes after application of SSP-RK time integrator is listed in Table 3.6. The reported CFL conditions are obtained by taking into consideration all values of θ. It is observed that the ExRi-SD method has an improved stability limit of greater than 40% for $p = 3, 4, 5$ compared to the standard Discontinuous Galerkin method of respective orders.

3.3.4 Numerical Validation

The proposed scheme is validated using the linear advection problem and 2D non linear Euler equations. A pure unstructured grid is used for all the cases. The mesh is generated by the open source grid generator GMSH using the "frontal" mesh generation algorithm. The refined grids are obtained by successively dividing each triangle into four elements.



(a) Linear Advection



(b) Isentropic vortex evolution

Figure 3.16: Unstructured meshes used for numerical validation.

p	NCells	L_1 error	L_∞ error	L_1 order	L_∞ order
2	200	2.13e-3	3.62e-2		
	800	2.46e-4	8.25e-3	3.11	2.13
	3200	2.80e-5	1.10e-3	3.13	2.91
3	200	2.77e-4	5.08e-3		
	800	1.93e-5	5.44e-4	3.84	3.22
	3200	1.45e-6	4.79e-5	3.74	3.5
4	200	1.24e-4	3.04e-3		
	800	4.18e-6	1.44e-4	4.9	4.4
	3200	1.30e-7	5.78e-6	5	4.64

Table 3.7: Solution errors in computation of advection of Gaussian pulse over a time period $t = 1$

3.3.4.1 2D Linear Advection

The linear advection of a wave is simulated in a periodic domain of size 1x1. The equation is given by

$$\frac{\partial Q}{\partial t} + \frac{\partial Q}{\partial x} + \frac{\partial Q}{\partial y} = 0 \tag{3.82}$$

The solution is initialized with a Gaussian pulse given by

$$Q(x, y, 0) = \exp\left(-\frac{(x-0.5)^2 + (y-0.5)^2}{g^2}\right) \tag{3.83}$$

where g is the half width of the curve. The exact solution at a given time t can be expressed as

$$Q(x, y, t) = \exp\left(-\frac{(x-0.5-t)^2 + (y-0.5-t)^2}{g^2}\right) \tag{3.84}$$

A pure unstructured mesh of 200 elements (Figure 3.16(a)) is used for coarsest mesh. The computed solution errors and the corresponding order of convergence for advection of a Gaussian pulse are plotted in Table 3.7. It is observed that the scheme produces the expected order of convergence for all the orders.

3.3.4.2 Inviscid Euler Equations

The non-linear Euler equations can be expressed as a 2D hyperbolic equation Eqn. (3.63) as detailed in Sec. 2.2. HLLC[86] Riemann Flux is used as the approxi-

p	NCells	L_1 error	L_∞ error	L_1 order	L_∞ order
2	550	1.90e-1	7.17e-2		
	2200	2.75e-2	7.50e-3	2.79	3.26
	8800	3.27e-3	1.15e-3	3.07	2.7
3	550	5.11e-2	1.27e-2		
	2200	4.08e-3	5.42e-4	3.65	4.55
	8800	2.27e-4	3.69e-5	4.17	3.88
4	550	1.47e-2	2.96e-3		
	2200	5.37e-4	1.16e-4	4.78	4.67
	8800	3.05e-5	5.68e-6	4.14	4.36

Table 3.8: Solution (density) errors for evolution of isentropic vortex over a time period $t = 1$

mate Riemann solver at element interfaces. For the inviscid flow computation, the problem is initialized with an isentropic vortex. The initial solution is defined by Eqns. (2.34-2.37). For a system of equations, the wave speed "a" is taken to be the maximum eigen value of the system. In this case, it would be ($|u| + c$) where c is the speed of sound propagation and $|u|$ is the velocity magnitude.

The computational domain is taken large enough such that the influence of vortex becomes negligible at the domain boundaries. In this aspect, a square domain extending from (-8,-8) to (8,8) is considered for simulation. Figure 3.16(b) shows the coarsest mesh used for the computation. The simulation is run for a non dimensional time period of $t = 16$ at which the analytical solution exactly corresponds to that of the initial solution. The errors in density and the corresponding order of convergence for 3^{rd}, 4^{th} and 5^{th} order ExRi-SD schemes are listed in Table 3.8. Expected order of convergence is obtained in all the cases.

3.4 Summary

A new way of flux approximation using Extended Riemann solvers is proposed for higher order schemes. The flux is now explicitly modeled as a continuous flux from the element boundary towards the interior. To illustrate the method, a co-location (ERC) and Galerkin (ERG) formulation are implemented in the case of one dimensional elements and a Sub-Domain method (ExRi-SD) is implemented

for 2D triangular elements.

For one dimensional schemes, the Fourier analysis of the schemes show that the ERC scheme has a better wave propagation characteristics and improved stability compared to the ERG scheme. The proposed ERC method enjoys a more relaxed CFL stability limit compared to the standard DG schemes, thus allowing a larger time step and a more efficient scheme. A similar trend is observed in two dimensional formulations as well. For both the 1D and 2D formulations, grid convergence tests are carried out for linear and non-linear hyperbolic equations. In all cases considered, the schemes exhibit the desired convergence characteristics.

Chapter 4

Extension of ExRi method to viscous flows

The ExRi method developed in Chapter 3 provides a way to construct smooth fluxes that vary from Riemann fluxes at the boundary to the internal fluxes at the element interior. In this chapter, the ExRi concept is applied to construct an approximate smooth solution at the element interface. With this, a new one-step method of gradient approximation for the viscous fluxes is proposed for higher order schemes. An approximate analytical solution is constructed across the Riemann interface and the resulting gradient information is used to formulate the corresponding diffusion terms in computing the viscous fluxes. The proposed formulation for viscous fluxes does not require any auxiliary variables and has no additional computational cost in approximating the solution gradients at element interface. Detailed numerical tests show that the scheme can reproduce expected order of convergence and exhibits higher order characteristics in both steady and transient viscous flow simulations.

4.1 Background

The higher order methods such as DG, SV, SD and the previously developed ExRi methods can inherently support discontinuous solutions at the element boundary. In the case of convection equations such as inviscid Euler equations, an exact or

approximate Riemann solver is used to obtain the fluxes at the interface. However, for the diffusion equations, the approximation of diffusion fluxes at the interface require the solution gradient information across the interface. Various schemes were proposed to model the diffusion fluxes for second order diffusion equations. A detailed comparison of different schemes was performed by Arnold et. al. [87]. Two of the most commonly used methods to address diffusion terms are the Bassi-Rebay scheme [88] and the Local Discontinuous Galerkin (LDG) [89] scheme. Compared to the LDG schemes, the Bassi-Rebay schemes require only individual gradient computations for each element interface. This results in a more localized scheme compared to LDG scheme. The Bassi-Rebay scheme is also computationally more efficient than the LDG schemes as they do not require storage of additional auxiliary variables to compute the gradient information.

In this work, we propose a new direct method of approximating the solution gradients at the interface for solution of Navier Stokes equation by applying the ExRi method to compute the solution gradient information. The solution gradients are estimated by analytically constructing a local approximate solution satisfying the Riemann state at the interface. The resulting formulation involves no additional computational cost (matrix operations) and can be implemented into the existing DG like schemes with ease. Various numerical tests are conducted to establish the order of convergence and to test the influence of different parameters.

4.2 Navier-Stokes Equations

The Navier Stokes equation can be expressed as a general hyperbolic system given by

$$\frac{\partial Q}{\partial t} + \nabla \cdot (\vec{F}^C + \vec{F}^V) = 0 \tag{4.1}$$

The solution Q , convective flux F^C and viscous flux F^V are expanded as

$$Q = \begin{pmatrix} \rho \\ \rho u_i \\ \rho e \end{pmatrix} \qquad F_j^C = \begin{pmatrix} \rho u_j \\ \rho u_i u_j + \boldsymbol{p}\delta_{ij} \\ u_j \left(\rho e + \boldsymbol{p}\right) \end{pmatrix} \quad F_j^V = \begin{pmatrix} 0 \\ -\tau_{ji} \\ -u_i\tau_{ji} - k\frac{\partial T}{\partial x_j} \end{pmatrix} \tag{4.2}$$

where ρ, u_i $\boldsymbol{p}$ and e are the density, velocity, pressure and internal energy of the fluid respectively, δ_{ij} is the Kronecker delta function, k is the thermal conductivity and τ_{ij} is the viscous stress tensor given by

$$\tau_{ii} = -\frac{2}{3}\mu(\nabla \cdot U) + 2\mu\frac{\partial u_i}{\partial x_i} \tag{4.3}$$

$$\tau_{ij} = 2\mu\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right) \quad i \neq j \tag{4.4}$$

where μ is the molecular viscosity coefficient. The compressible Navier Stokes equation Eqn. (4.2) is closed with the equation of state given by

$$\boldsymbol{p} = (\gamma - 1)\,\rho\left(e - \frac{1}{2}U^2\right) \tag{4.5}$$

where γ is the specific heat ratio (1.4 for air) and U is the velocity magnitude. To solve the governing equation set given in Eqn. (4.1), the standard RK-DG method described in Chapter 2 or the proposed ExRi method formulated in 3 can be employed. Here we use the previously developed ExRi method for discretizing Eqn. (4.1).

4.3 Gradient approximation at interface

4.3.1 Gradient corrections from Riemann solutions

Due to it's piecewise continuous nature, the solution can be discontinuous at the element-element interface. The solution of these discontinuous states can be obtained by application of an exact or approximate Riemann solver. Here, we use the

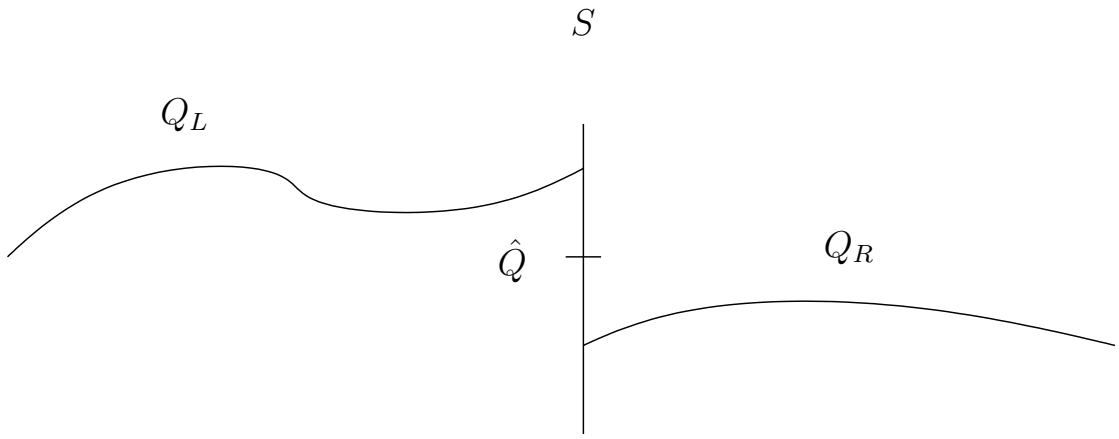


Figure 4.1: Schematic representation of solution Q across element interface S.

HLL Riemann state as the approximate Riemann state, given by

$$\hat{Q}_{HLL} = \begin{cases} Q_L & if \quad s_L > 0 \\ Q^*_{HLL} & if \quad s_L \leq 0 \leq s_R \\ Q_R & if \quad s_R < 0 \end{cases} \tag{4.6}$$

where Q_L and Q_R are the solutions corresponding to left and right states, s_L and s_R are the left and right wave speeds and the intermediate state Q^*_{HLL} is given by

$$Q^*_{HLL} = \frac{s_R Q_R - s_L Q_L - (F_R - F_L)}{s_R - s_L} \tag{4.7}$$

where F_R and F_L are the fluxes corresponding to the states Q_R and Q_L respectively.

The discontinuous solution across the interface is illustrated in Figure 4.1. The Riemann solutions are only first order accurate and can not be used to compute the gradient information as such. If an approximate solution $\tilde{Q}_S$ is constructed such that it satisfies the Riemann solution at the interface S and the internal solution at the element interior, the corrected gradient information (at the interface) can be obtained from this solution approximation. A weighted residual based reconstruction like least squares or Galerkin methods would consume huge computational resources in terms of memory and computational cost. Here, we propose a method of analytical formulation of the approximate solution $\tilde{Q}_S$ such that it satisfies the Riemann solution at the interface S and retains the internal solution away from

the given interface. The resulting approximation is given by

$$\tilde{Q}_S = Q_i + \omega_S(\hat{Q}_S - Q_i|_S) \tag{4.8}$$

where $Q_i|_S$ is the internal solution at the interface S and ω_S is a blending function that enables a smooth variation of the solution from Q_i in the interior of the element to the Riemann solution $\hat{Q}_S$ at the element interface S.

4.3.2 Construction of blending function ω

Similar to the ExRi formulation described in Chapter 3, the construction of the blending function ω is one of the key factors in the formulation given in Eqn. (4.8). From the distance of propagation of information is given by $\mathcal{C}h$. If the length of the element is taken to be 1 unit, the non-dimensional distance of propagation of information is given by the desired CFL number $\mathcal{C}$. Thus, for a given element, the region is divided into two parts: (i) region Ω^S influenced by the element boundary S (ii) internal region Ω^{in} not influenced by element boundary. Outside the boundary region, we have the condition

$$\frac{\partial \tilde{Q}_S^m(\xi,\eta)}{\partial \xi^{m-k}\partial \eta^k} = \frac{\partial Q_S^m(\xi,\eta)}{\partial \xi^{m-k}\partial \eta^k}, \;\; (\xi,\eta) \notin \Omega^S \tag{4.9}$$

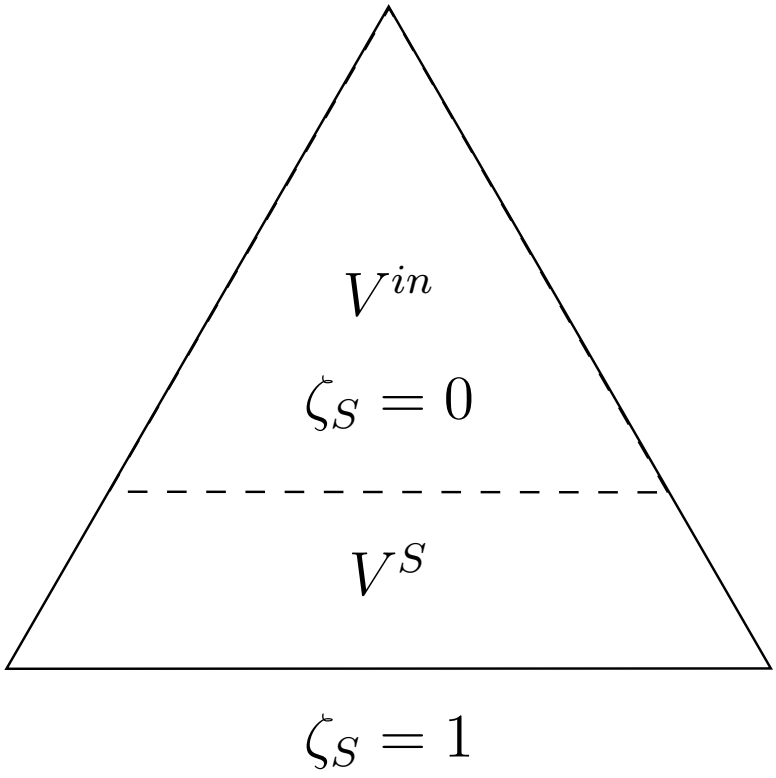
At the boundary, the solution satisfies the Riemann solution, resulting in the condition

$$\tilde{Q}_S(\xi,\eta) = \hat{Q}(\xi,\eta), \;\; (\xi,\eta) \in S \tag{4.10}$$

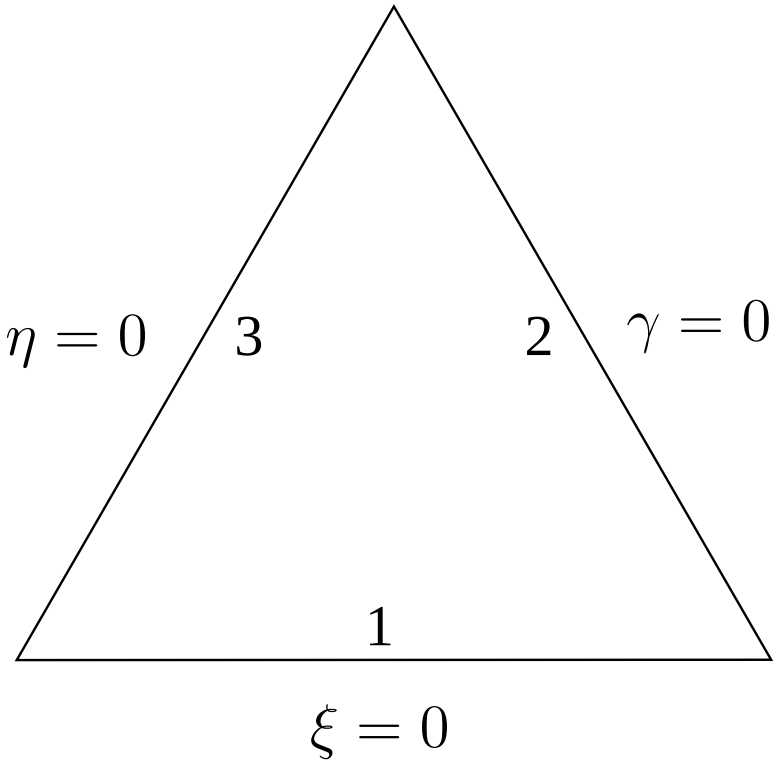
To satisfy the above conditions, the general shape of the blending function is taken as

$$\omega_S = \begin{cases} \zeta_S^p & 1 > \zeta_S > 0 \\ 0 & \zeta_S \leq 0 \end{cases} \tag{4.11}$$

where ζ_S is the local coordinate of the interface S having a value of 1 at the boundary S and decreases to zero towards the element interior and p is the degree



(a) Regions of approximation



(b) Reference coordinates and face numbering

Figure 4.2: Schematic representation of reference triangle and regions of approximation.

of solution approximation. The local coordinate has a positive value only in the region Ω^S illustrated in Figure 4.3.2. Many choices exists for the formulation of ζ. Here we define this coordinate in a reference coordinate system (ξ, η), similar to the definitions used in ExRi-SD method (Eqn. 3.77). The equations are repeated here for clarity. The formulation for ζ is given by

$$\zeta_1 = 1 - \frac{\eta}{\mathcal{C}} \tag{4.12}$$

$$\zeta_2 = 1 - \frac{\gamma}{\mathcal{C}} \tag{4.13}$$

$$\zeta_3 = 1 - \frac{\xi}{\mathcal{C}} \tag{4.14}$$

The corresponding gradient terms at the faces can now be written from Eqn. (4.8) as

$$\frac{\partial \tilde{Q}_S}{\partial \xi} = \frac{\partial Q_i}{\partial \xi} + \frac{\partial \omega_S}{\partial \xi}(\hat{Q} - Q|_S) \tag{4.15}$$

$$\frac{\partial \tilde{Q}_S}{\partial \eta} = \frac{\partial Q_i}{\partial \eta} + \frac{\partial \omega_S}{\partial \eta}(\hat{Q} - Q|_S) \tag{4.16}$$

These corrected gradients are used to approximate the viscous fluxes at the interface.

4.3.3 Corrected gradients for viscous flux discretization

The formulation of solution gradients in Eqn. (4.15) still results in two different solution gradients. Here we denote the corrected gradients as $\nabla\tilde{Q}_S^L$ and $\nabla\tilde{Q}_S^R$ corresponding to the left and right elements sharing the interface S. If an averaged state is used in place of the Riemann state $\hat{Q}$, the second term in RHS of Eqns. (4.15-4.16) resembles the jump operator used in the Bassi-Rebay [88] scheme. The viscous flux at the interface is then approximated with the average of viscous fluxes corresponding to the neighboring elements. The corresponding viscous interface

flux $\vec{H}_S^V$ corresponding to interface S is now given by

$$\vec{H}_S^V = \frac{1}{2}\left(\vec{F}^V(Q^L, \nabla\tilde{Q}_S^L) + \vec{F}^V(Q^R, \nabla\tilde{Q}_S^R)\right) \tag{4.17}$$

where Q^L and Q^R are the left and right states across the interface S. Note that the viscous flux $\vec{H}_S^V$ is now unique for a given interface S. The proposed formulation is applied only on the gradient values and does not change approximation of solution states. Due to this decoupling of gradient approximation, the proposed scheme can be applied on top of any higher order scheme such as ExRi, Discontinuous Galerkin, Spectral Volume, Spectral Difference methods etc.

4.4 Absorbing Boundary Regions

Any practical simulation of a flow would require application of artificial boundary conditions. A computational boundary too far away from the region of interest results in increase in computational cost while a computational boundary that is very near to the region of interest would have an influence on the solution. It is necessary to choose a boundary condition such that it can be applicable in the near field region with a very minimum influence on the computed solution. Absorbing boundary conditions (ABCs) are frequently employed in the wave propagation problems such as acoustics, electromagnetics etc. Recently, the absorbing boundary conditions based on perfectly matched layers were developed for non-linear equations [90]. Here, we implement a simplified form of the set of equations for the absorbing region based on the formulations given in [90]. Assuming a steady uniform mean flow, the simplified equations are given by

$$\frac{\partial Q}{\partial t} + \nabla \cdot \vec{F} + \vec{\sigma} \cdot \vec{q} = 0 \tag{4.18}$$

where σ is the absorption coefficient and the vector $\vec{q} = (q_x, q_y)$ is given by

$$\frac{\partial q_x}{\partial t} + \sigma_x q_x + \frac{\partial F^x}{\partial x} = 0 \tag{4.19}$$

$$\frac{\partial q_y}{\partial t} + \sigma_y q_y + \frac{\partial F^y}{\partial y} = 0 \tag{4.20}$$

No additional modifications are introduced for the viscous terms. The absorption coefficient is taken as

$$\sigma_x = \sigma_{max} \left| \frac{D_{AB}}{W_{AB}} \right|^{\alpha} \tag{4.21}$$

where W_{AB} is the width of the absorption region, D_{AB} is the shortest distance to the boundary of interior domain (which is same as the starting of absorbing boundary). The constants σ_{max} and α are taken to be 5 and 4 respectively. For higher order methods, it is observed that the variables q_x and q_y can be computed with only the internal fluxes without compromising the performance of the absorbing region. Thus, the scheme becomes more compact and the ABC can be easily implemented in the existing codes. Numerical experiments suggest that higher order implementations can absorb the fluctuations with a few number of elements, while the lower order implementations require more elements for effective absorption of the fluctuations.

4.5 Numerical Analysis

4.5.1 Order of Convergence

The proposed scheme is tested for numerical convergence using the vortex diffusion case. The dynamic viscosity is taken as 0.01. The solution is initialized with an isentropic vortex defined by Eqns. (2.34-2.37). The magnitude of vortex strength β is taken as 5 units. A periodic domain of width 14x14, centered at (0,0) is considered for simulation. The coarsest mesh corresponding to the simulation is shown in Figure 4.3. To test the accuracy of the schemes in a viscous flows of various regimes, two different sets of equations are simulated. For the first case, the full Navier Stokes equation is solved for a 16 time periods. In the second case,

	Full Navier Stokes				Diffusion Equation			
p	L_1 error	L_∞ error	L_1 order	L_∞ order	L_1 error	L_∞ error	L_1 order	L_∞ order
1	9.61e-3	7.24e-1			2.81e-3	3.37e-1		
	2.97e-3	2.79e-1	1.69	1.37	7.49e-4	8.20e-2	1.91	2.04
	5.05e-4	5.81e-2	2.56	2.26	1.95e-4	1.18e-2	1.94	2.8
2	2.84e-3	1.73e-1			6.76e-4	8.44e-2		
	2.84e-4	1.95e-2	3.32	3.15	1.11e-4	1.41e-2	2.61	2.58
	2.48e-5	3.12e-3	3.52	2.65	1.63e-5	1.44e-3	2.77	3.29
3	9.89e-4	5.97e-2			1.96e-4	2.53e-2		
	3.33e-5	6.31e-3	4.89	3.24	1.51e-5	1.37e-3	3.7	4.2
	1.31e-6	2.38e-4	4.67	4.73	9.76e-7	1.03e-4	3.95	3.74
4	1.95e-4	2.49e-2			4.03e-5	3.83e-3		
	4.24e-6	4.93e-4	5.52	5.66	1.76e-6	2.21e-4	4.51	4.11
	7.85e-8	1.32e-5	5.76	5.23	6.70e-8	9.27e-6	4.72	4.58

Table 4.1: Solution (internal energy) errors and experimental order of convergence

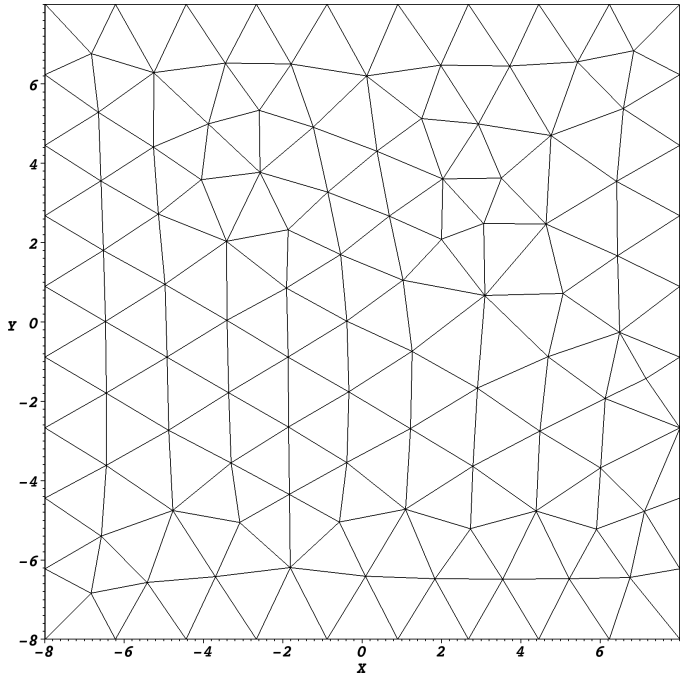


Figure 4.3: Coarsest mesh used for vortex diffusion test case.

the convective fluxes are set to zero, resulting in pure diffusion equation. In this case, the solution is computed for 5 time periods. In both the cases, we use the fine grid solution as the reference solution. The solution errors and the numerical order of convergence for schemes of up to 5^{th} order are listed in Table 4.1. The scheme achieves expected order of convergence in both the cases. For the Navier Stokes equation, the experimental order of convergence is in general higher than the designed order of accuracy. This could be due to the non-linearity involved in the NS equations.

4.5.2 Laminar boundary layer over a flat plate

The flat plate boundary layer is a common test problem to test the capability of the scheme to effectively capture the boundary layer profile. The accuracy of the wall boundary condition in implementation of no-slip condition can also be evaluated with this method. Here we compute the laminar boundary layer profile over a flat plate of width 1 unit length. The parameters are set such that the Reynolds number based on the plate width is 10^6. A low mach number of 0.1 is chosen for the free stream flow. The solution is converged to machine accuracy using the ILU

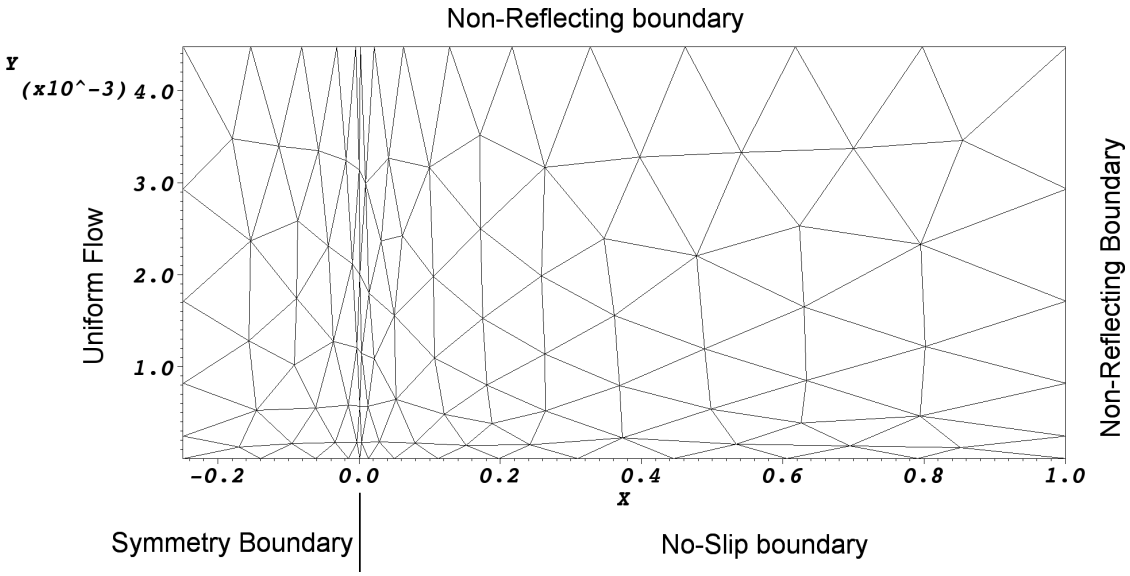
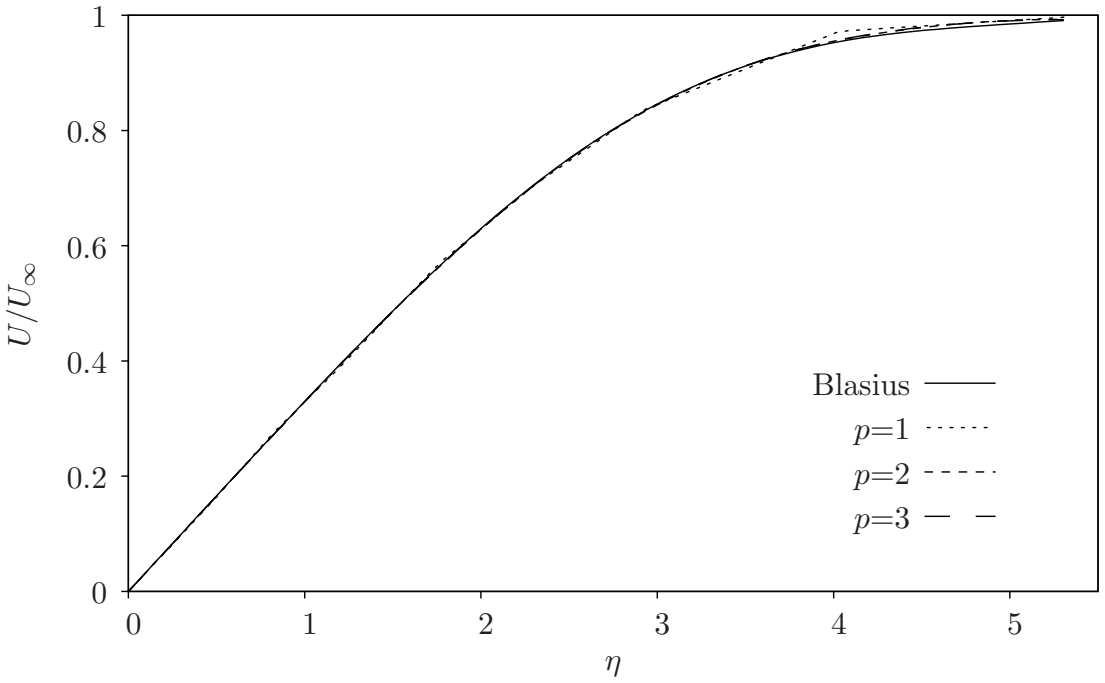


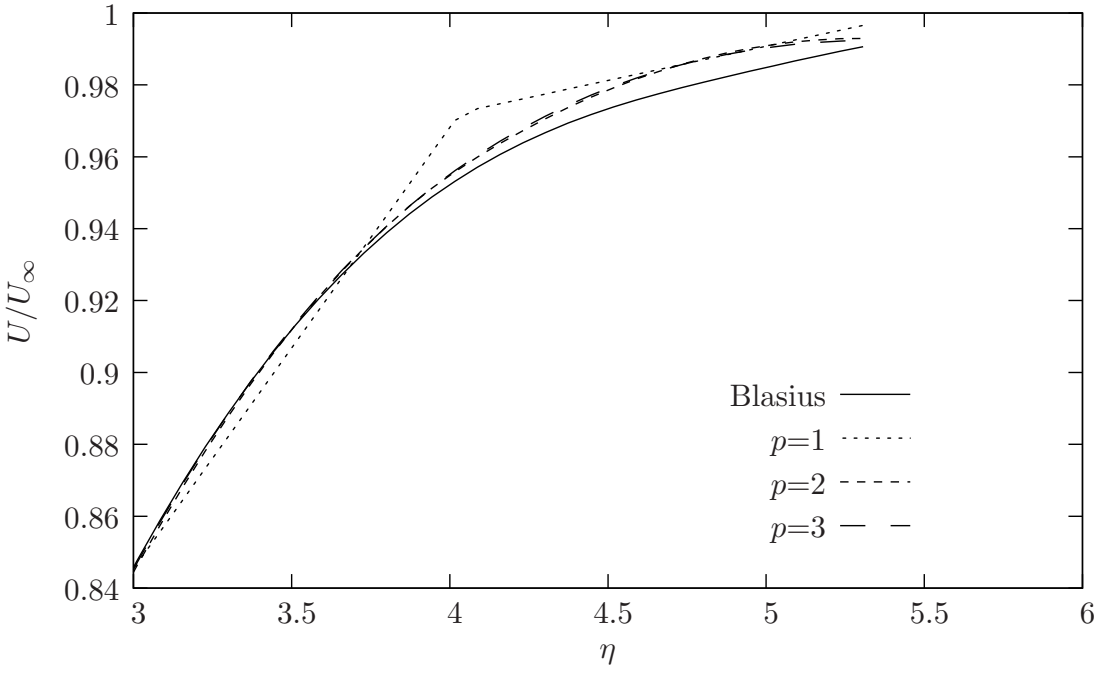
Figure 4.4: Unstructured mesh used for simulation of laminar boundary layer flow over a flat plate.

preconditioned GMRES implicit method. To illustrate the advantages of the higher order discretization, a very coarse mesh is employed for all the computations. A pure unstructured mesh of 179 elements is constructed for a domain extending from (-0.5,0) to (1, 0.0045). Appropriate clustering is applied at the leading edge (0,0) and at wall region. The constructed mesh is shown in Figure 4.4. Symmetry boundary condition is applied for the computational boundary upstream of the plate. No absorbing boundaries are used for the computations.

All the results are obtained by direct interpolation of the solution data from the mesh elements. The results are compared with the numerical solution [91] of the Blasius's problem. Figure 4.5 shows the velocity profile at $x = 0.7$. The entire velocity profile is resolved with only 4 elements. The velocity profiles corresponding to the 3^{rd} and 4^{th} order schemes closely follow the Blasius's solution. The 2^{nd} order solution is approximated by a piecewise linear solution as compared to quadratic and cubic solution approximation of 3^{rd} and 4^{th} order schemes. This is evident from the expanded view of the velocity profile as shown in Figure 4.5(b). The advantages of the higher order schemes are further illustrated in the plot of skin friction coefficient in Figure 4.6. The results suggest that the solution resolution is greatly enhanced with increase in the order of the schemes. The skin friction

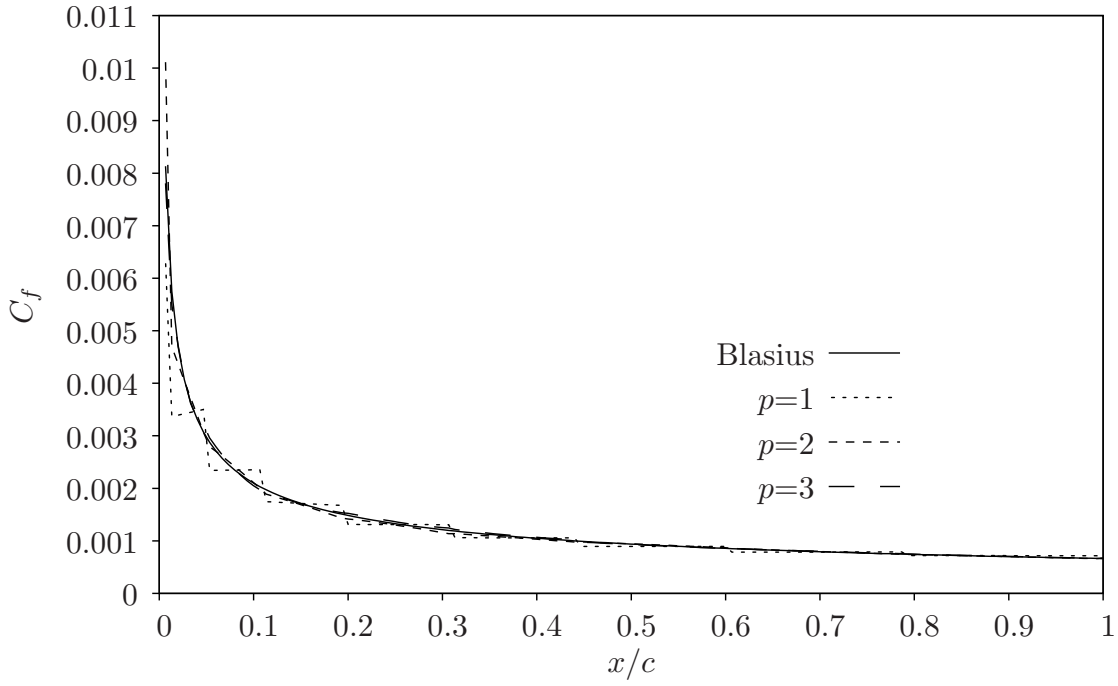


(a) Velocity Profile

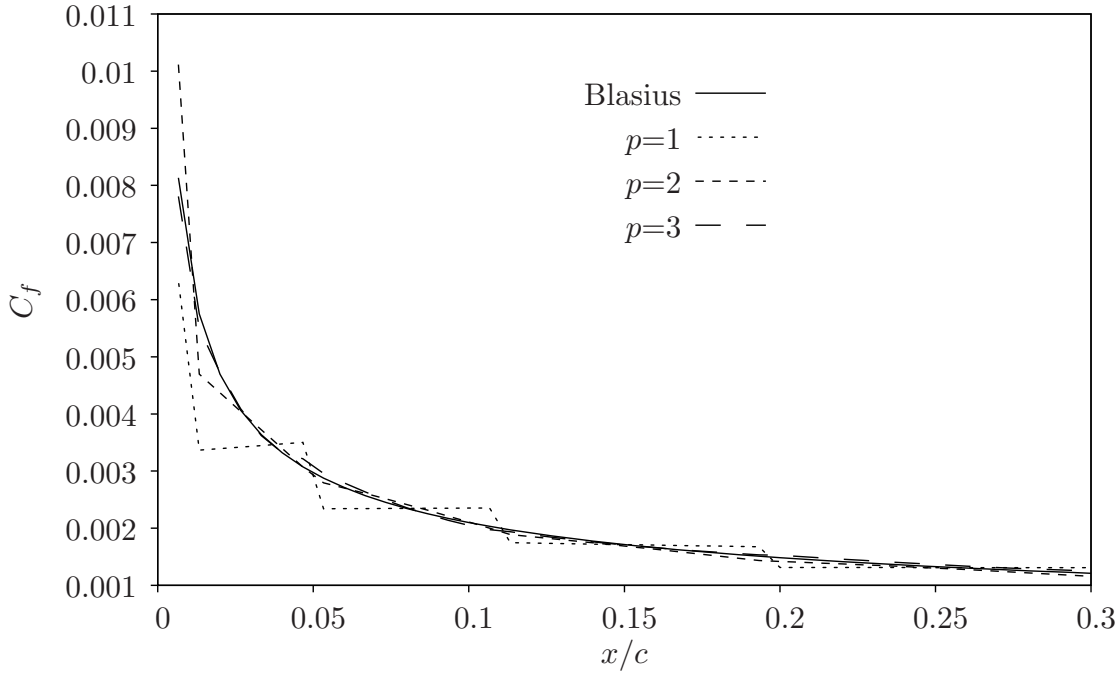


(b) Zoomed

Figure 4.5: Laminar flat plate boundary layer velocity profile at $x/c = 0.7$ (Dimensionless coordinate η is given by $\eta = y\sqrt{\frac{\rho_\infty U_\infty}{\mu x}}$).



(a) Velocity Profile



(b) Zoomed

Figure 4.6: Laminar flat plate boundary layer skin friction.

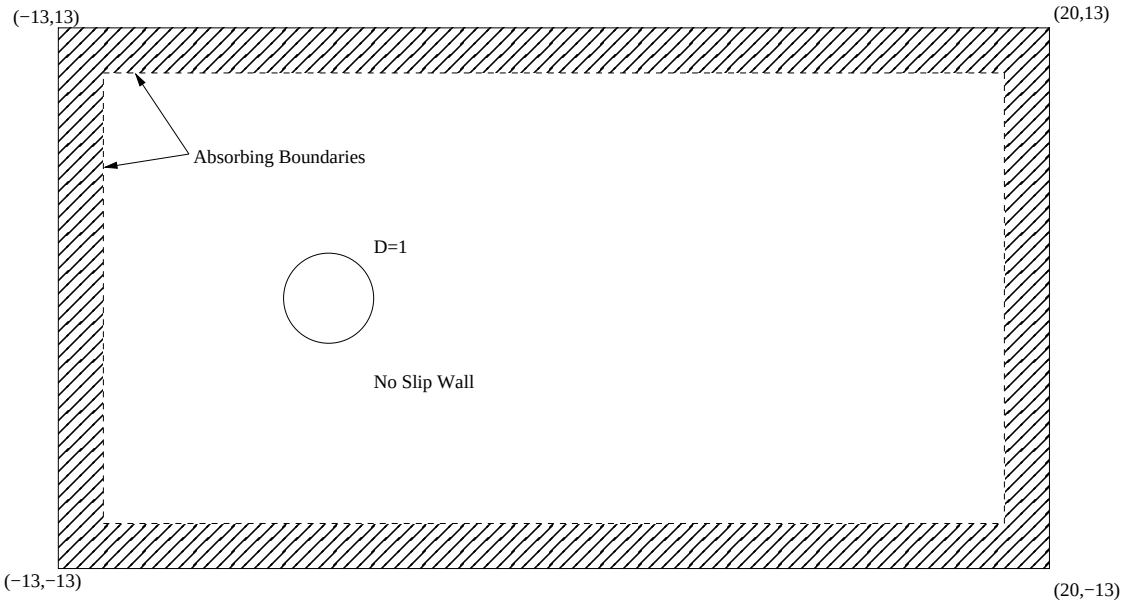


Figure 4.7: Computational domain used for simulating flow past a circular cylinder.

coefficient computed with the 4^{th} order scheme is almost identical to the Blasius solution even near the leading edge of the plate where the solution has a singularity. The results also show that the proposed viscous flux discretization method works relatively well on severely skewed meshes as well.

4.5.3 Vortex shedding in flow over a circular cylinder

A low Reynolds number flow over a circular cylinder is a standard test case to estimate the accuracy of the viscous schemes in a transient flow condition. A flow of Mach number 0.2 over a cylinder of unit diameter is considered for simulation. The Reynolds number corresponding to the flow is taken to be 100 at which the flow develops instability and an alternating vortex shedding is observed in the wake of the cylinder. Burbeau et. al. [92] conducted a detailed numerical study on the convergence properties of higher order DG schemes in capturing the transient vortex shedding phenomenon.

A schematic representation of computational domain is given in Figure 4.7. The meshes are generated using GMSH [79] using the "frontal" mesh generation algorithm. Four different meshes are chosen to illustrate the grid independence characteristics of the higher order schemes. The size of the mesh and the total

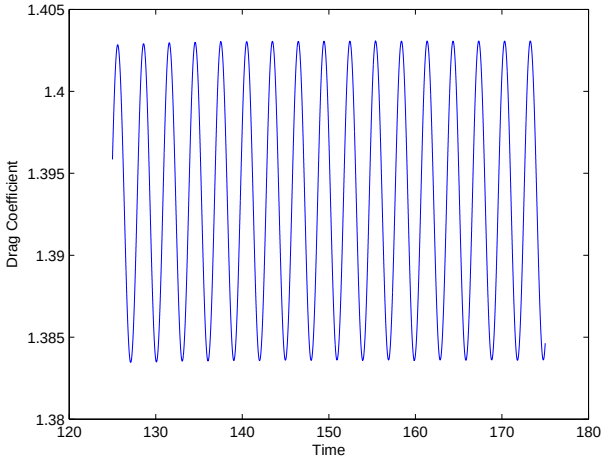
Mesh	$\Delta x_{cylinder}$	$\Delta x_{farfield}$	Wall Elements	Total Elements
1	0.12	3	24	1166
2	0.08	2	36	2517
3	0.04	1.5	74	5406
4	0.03	1.3	98	7583

Table 4.2: Details of mesh used in simulation of flow past a circular cylinder.

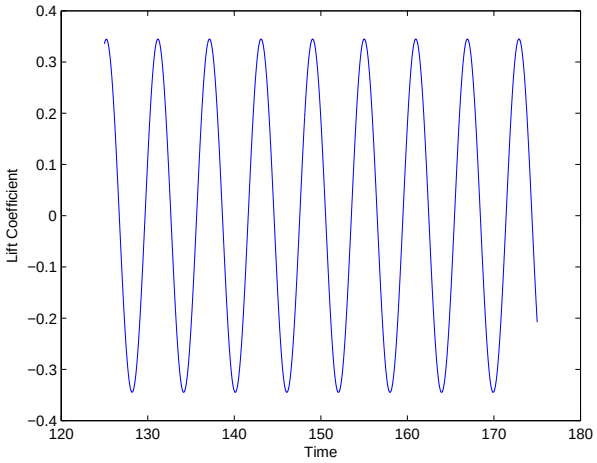
number of nodes are given in Table 4.2. Absorbing boundaries of width 3 length units are employed on all the external computational boundaries. The simulation is started with a mean flow of Mach number 0.2. For ease of representation, we denote a solution as $M_{p,l}$ where p is the degree of solution approximation, l is the level of refinement of mesh. For example, a 2^{nd} order solution on the finest level mesh (4^{th} refinement) denoted as $M_{1,4}$.

The time history of the drag and lift values and are plotted in Figure 4.8. A Fast Fourier Transform (FFT) of the lift coefficient shows a distinct peak in the frequency spectrum. The corresponding Strouhal number of the flow is estimated as $St = fD/U$ where D is the diameter of the cylinder and U is the free stream velocity. The values of Strouhal number, average drag coefficient and lift coefficient for different orders of scheme are listed in Table 4.3. The results show that the third and fourth order schemes achieve grid independence much faster than the second order scheme. The 4^{th} order schemes produce acceptable results in the coarsest grids, especially when comparing the average C_D values. The coarsest 4^{th} order solution ($M_{3,1}$) has better characteristics than the finest 2^{nd} order solution ($M_{1,4}$), clearly illustrating the effectiveness of higher order schemes in capturing complex flow phenomena in much coarser grids.

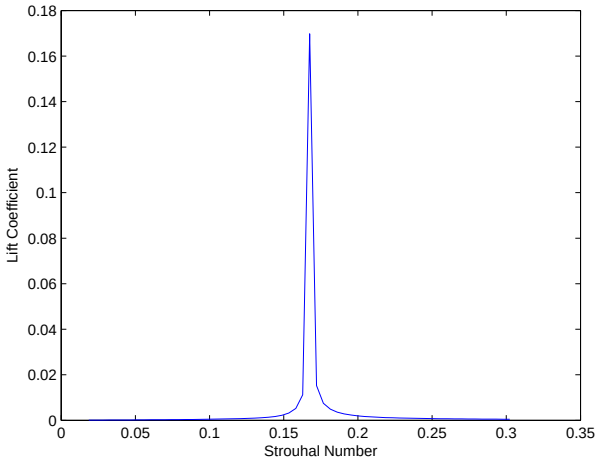
The vorticity contour of a single time period of vortex shedding for an $M_{3,2}$ solution is plotted in Figure 4.9. The fluctuating vorticity field may be seen to be dissipated by the (downstream) absorbing boundary region. The vortices decay completely before reaching the computational boundary. In generating the contours in Figure 4.9, high resolution contours are computed by subdividing the triangle element into smaller elements depending on the order of the scheme.



(a) Drag Coefficient

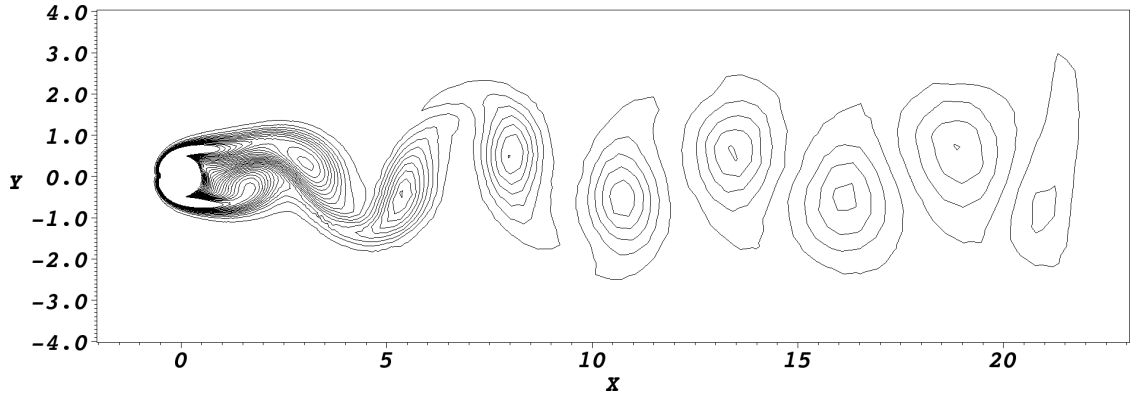


(b) Lift Coefficient

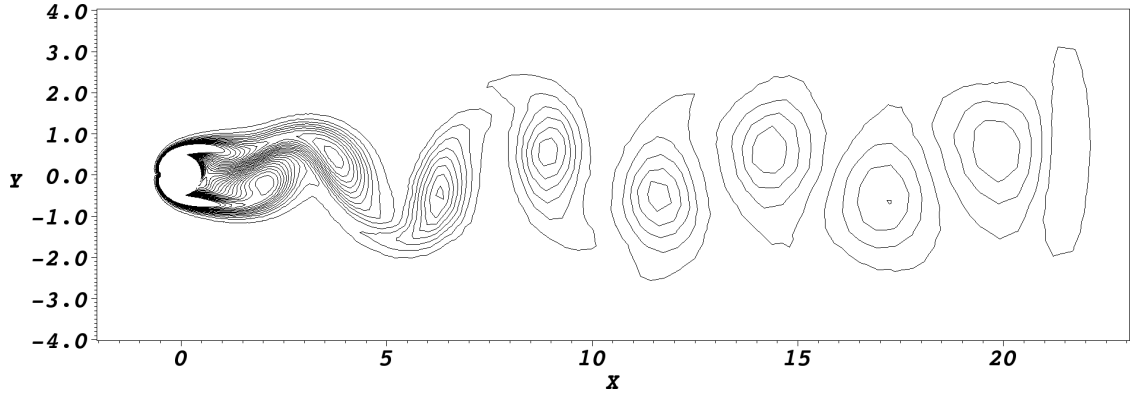


(c) FFT of C_L

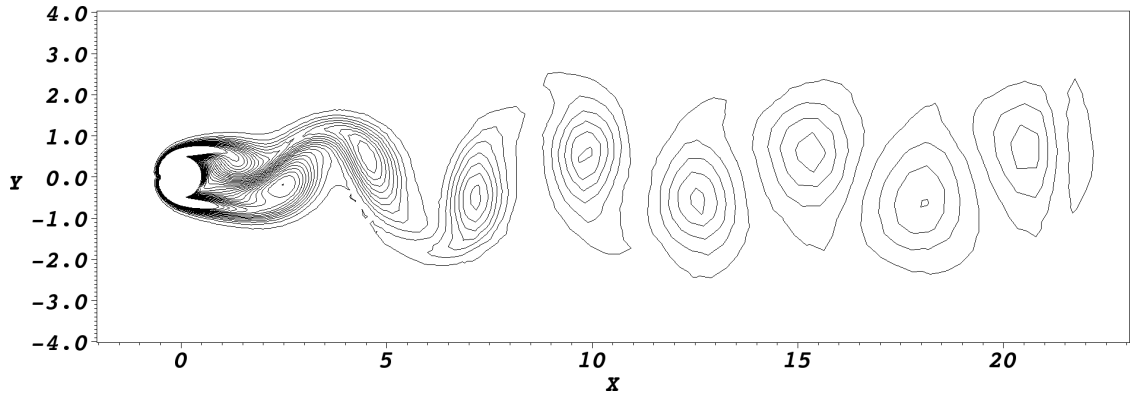
Figure 4.8: Time history of drag and lift values for a 4^{th} order simulation of flow over circular cylinder (Mesh = 2).



(a) $t = 302$

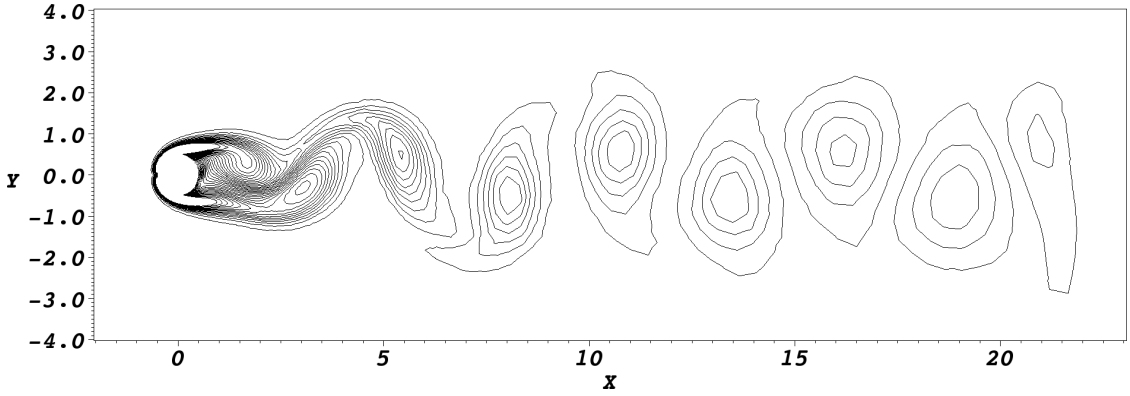


(b) $t = 303$

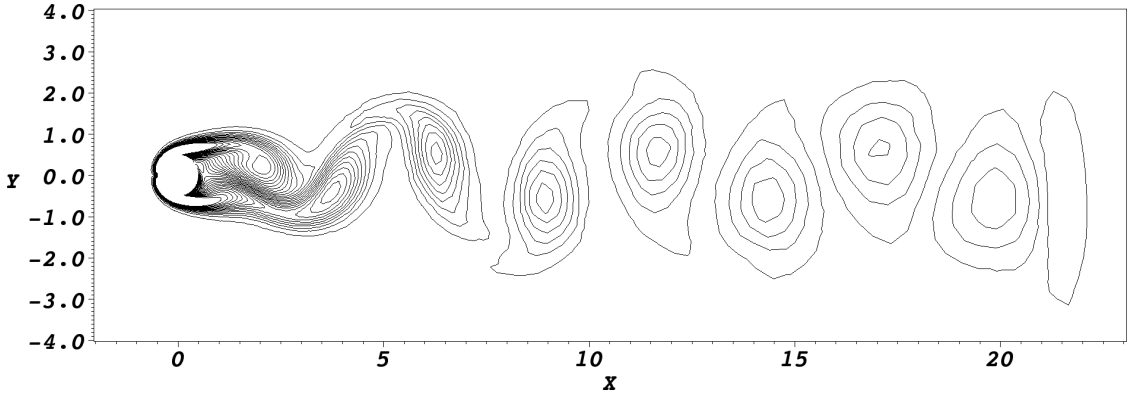


(c) $t = 304$

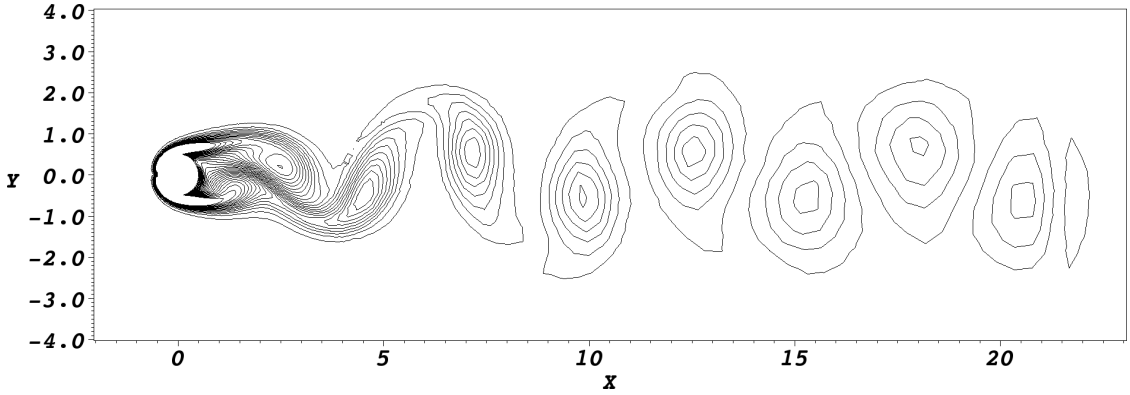
Figure 4.9: Vorticity contours over a single time period of vortex shedding.



(d) $t = 305$



(e) $t = 306$



(f) $t = 307$

Figure 4.9: Vorticity contours over a single time period of vortex shedding (contd.).

$M_{p,Mesh}$	Average C_D	Peak-to-peak C_L	Time Period	Strouhal Number
$M_{1,1}$	1.299	0.604	6.383	0.157
$M_{1,2}$	1.328	0.646	6.125	0.163
$M_{1,3}$	1.343	0.671	5.986	0.167
$M_{1,4}$	1.352	0.675	5.973	0.167
$M_{2,1}$	1.377	0.688	5.975	0.167
$M_{2,2}$	1.379	0.687	5.963	0.168
$M_{2,3}$	1.391	0.688	5.963	0.168
$M_{3,1}$	1.392	0.691	5.960	0.168
$M_{3,2}$	1.393	0.690	5.959	0.168
$M_{3,3}$	1.394	0.688	5.954	0.168

Table 4.3: Higher order solutions of vortex shedding at wake of cylinder.

4.6 Summary

An efficient method of higher order viscous flux approximation at the element interface is formulated based on wave propagation characteristics of the elements. The proposed method is simple and computationally effective as it does not require any additional computation or auxiliary variables. As mentioned in Sec. 4.3.3, since the proposed method is applied only on the gradient approximation at the boundary while leaving the solution state untouched, the proposed method can be easily extended to other higher order methods such as Discontinuous Galerkin, Spectral Volume, Spectral Difference, etc..

The numerical order of convergence is tested for both Navier Stokes equations and a diffusion equation involving only the viscous fluxes of NS equations. The scheme is found to reproduce expected order of convergence up to 5^{th} order. The scheme is validated for the standard steady state laminar boundary layer over a flat plate and the transient vortex shedding of flow past a circular cylinder. For both the cases, the mesh is refined near the boundary in order to capture the viscous effects accurately. The grid convergence study clearly demonstrate the advantages of higher order schemes in representing complex flow structures in a much coarser grid compared to lower order schemes.

Chapter 5

A generic higher order multi-level time stepping scheme

The ExRi method developed in Chapter 3 demonstrated that the information propagation within the elements can be exploited to construct schemes with improved stability. In this chapter, the wave propagation characteristics within the element is used to formulate a high order adaptive time stepping method in which solutions in different mesh regions can be evolved with different time steps. In this method, the given grid is decomposed into regions according to their time-step restrictions. A recursive time stepping method is employed to march each region with their respective local time-steps. Since the scheme can work with any spatial and temporal schemes, only minimal modifications are required for implementation in existing solvers. Numerical experiments show that the scheme indeed preserves the formal order of accuracy while having a significant reduction in the computational cost. To illustrate the simplicity, the details of implementation of the scheme in an existing RK-DG solver is presented. While the adaptive time stepping algorithm can be applied to any spatial discretizations, in this work, the RK-DG scheme is chosen due to its well established stability characteristics. The computational results using the adaptive time stepping schemes show good agreement with the pure RK-DG computations.

The current work is organized as follows: Sec. 5.1 details the wave propagation characteristics of a scheme and the dependency of the solution on its neighboring solution over a time period. The MTS algorithm is detailed in Sec. 5.2.1 with illustrations in 1D and 2D unstructured elements. In Sec. 5.3, a 1D stability analysis is conducted for the MTS scheme to study the effect of increasing the time step size of the solution exchange between mesh blocks. Finally, in Sec. 5.4, the effectiveness and speedup of MTS scheme is demonstrated on Euler equations for isentropic vortex evolution problem and direct computation of acoustics for flow over an open cavity.

5.1 Propagation of perturbations and CFL condition

For a physical system represented by the hyperbolic equation Eqn. (3.1) with the maximum wave speed a, the corresponding maximum distance any disturbance can travel is given by

$$r = a\Delta t \tag{5.1}$$

where r is the distance of propagation and Δt is a finite time step. From Eqn. (5.1), it is understood that, over a time period of Δt, the solution at a given point can influence the neighborhood within a distance of r length units. Any solution beyond this region is independent of the solution at the given point. This is illustrated in Figure 5.1 for a 2D hyperbolic equation.

In the case of explicit time stepping schemes, for a given time step of size Δt, the equation Eqn. (5.1) can be used to identify the neighboring elements influencing or influenced by the solution of the current element. With the definition of the CFL condition, the distance r can also be represented as

$$r = \mathbb{C}h \tag{5.2}$$

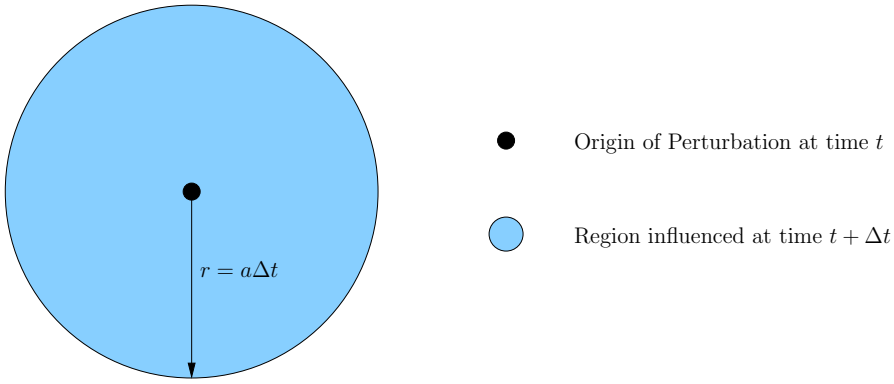


Figure 5.1: Extent of wave propagation at a wave speed a over a time period Δt for a 2D hyperbolic equation.

where $\mathbb{C}$ is the CFL number, and h is the size of the element. A solution evolution of an element over a period of Δt requires the neighboring solution to be defined within a distance of r length units from the element.

5.2 Algorithm formulation

5.2.1 Generic Multi-Time stepping schemes

In a generic multi-time stepping scheme, the mesh is divided into a series of non-overlapping blocks depending on the element sizes. For each of these blocks, the maximum time step size is limited by the local CFL condition of the elements within the block. With suitable representation of solution evolution at the block boundaries, the solution at the block interior can be marched in an adaptive manner with varying time step sizes for each blocks.

The evolution of solution in a typical MTS scheme is illustrated in Fig. 5.2 where each rectangle represents a single explicit time marching step and each column labeled B_f and B_c represent the fine and coarse mesh blocks with local time step limits $\frac{\Delta t}{4}$ and $\frac{\Delta t}{2}$ respectively. For clarity, we use the term "multi-step" to denote a series of $\mathcal{N}$ time integration steps performed in the current block for a single time integration step of the next coarser block. To maintain simplicity and ease of implementation, the value of $\mathcal{N}$ is kept constant for all fine block - coarse block

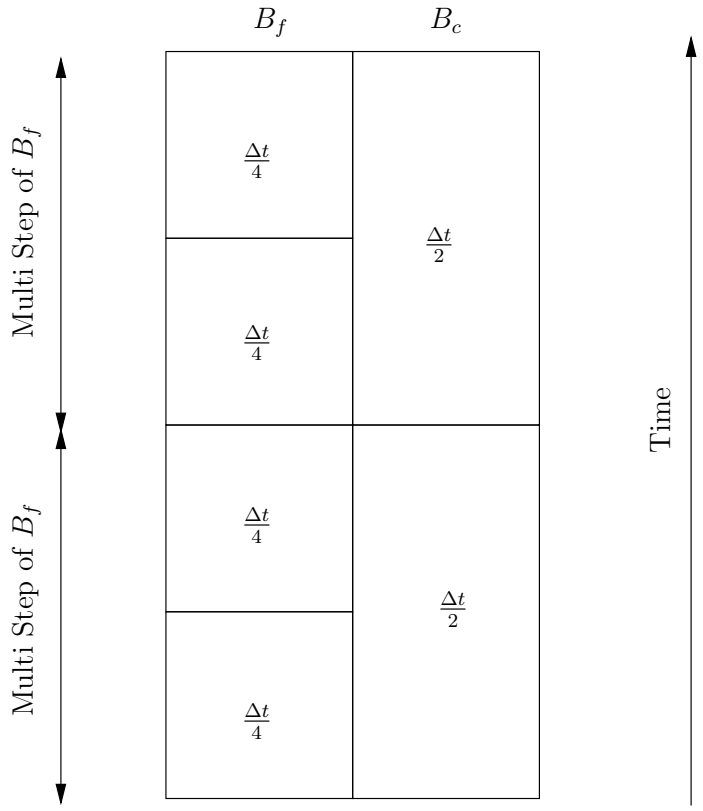


Figure 5.2: Illustration of a 2 level MTS with time step ratio of 2.

combination. For the case illustrated in Fig. 5.2, a single multi-step of block B_f would contain two explicit (RK) time marching steps.

5.2.2 Solution evolution at block interfaces

The main component of a MTS scheme is the treatment of solution evolution at the block-block interfaces such that it can allow different time step sizes across the interfaces. During a multi-step, suitable support need to be provided to evolve the solution at the block boundary. In the current formulation, we use a layer of ghost cells to provide the required data for evolving the solution at the block boundary. This layer is denoted as the "synchronization layer" since it is responsible for the block-block synchronization of the solution data.

To explain the solution evolution process at the block boundaries, we categorize the synchronization layer depending on the neighboring block from which it obtains the fresh solution. The layer of elements that obtains the new solution from the

fine block interior is called as "fine-layer" while the layer of elements that obtains the new solution from the coarse block interior is called as "coarse-layer". A two block mesh is considered for explaining the MTS algorithm. The fine block B_f has an allowable local time step of Δt_f while the coarse block B_c has a maximum allowable time step of Δt_c.

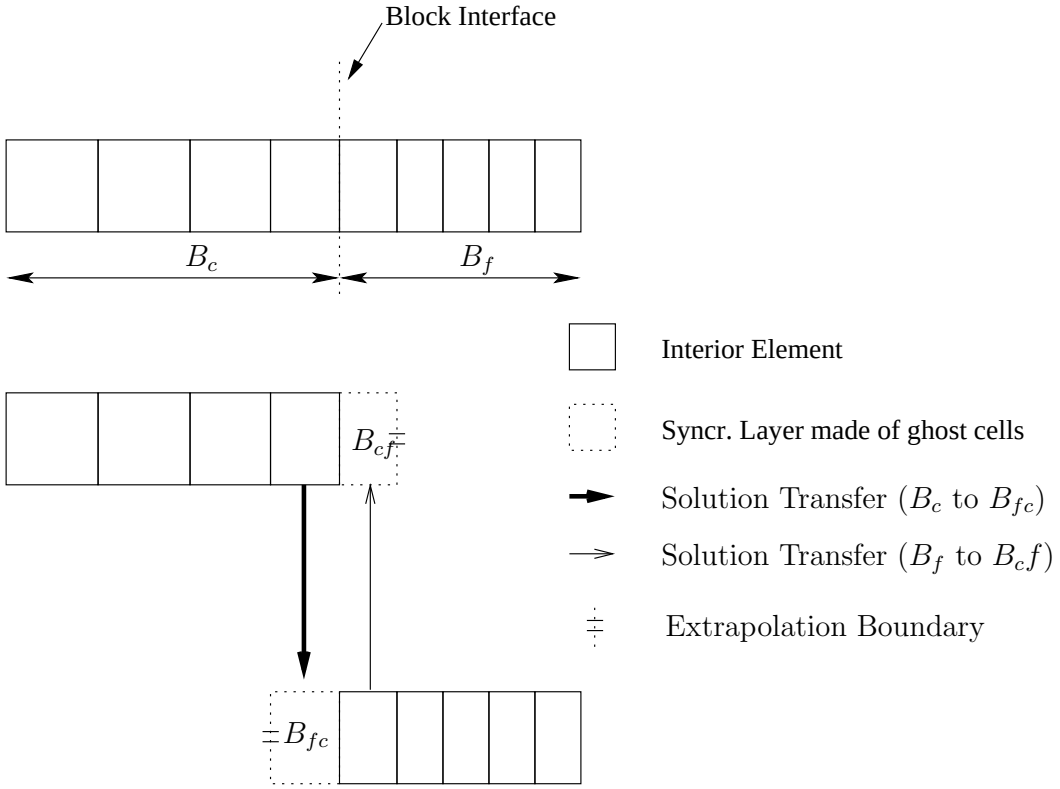
Fig. 5.3 shows an illustration of the different components of the current MTS algorithm. The block interfaces, synchronization layers and the associated solution updates for a 1D grid is shown in Fig. 5.3(a). An example of block division in an unstructured grid is also presented in Fig. 5.3(b). In Fig. 5.3, B_{cf} is the the fine-layer of the block B_c and receives solution updates from B_f . Similarly, B_{fc} is the coarse-layer of block B_f and receives solution updates from B_c. With these definitions, the MTS procedure of a 2-block mesh for a single time step of is detailed below:

Algorithm *MTS2B*

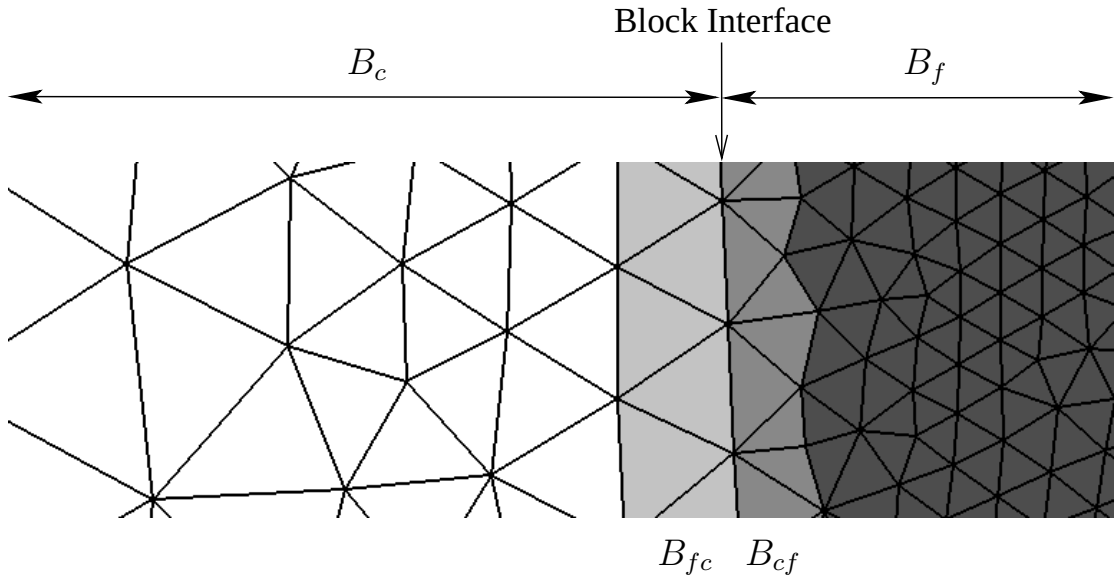
($*$ demonstrates MTS scheme for a 2 block mesh $*$)

1. Update ghost cells in B_{cf} with solutions from B_f at time t
2. Update ghost cells in B_{fc} with solutions from B_c at time t
3. March block B_c and B_{cf} to time $t + \Delta t_c$
4. **for** $i \leftarrow 1$ **to** $\mathcal{N}$
5. **do** March block B_f and B_{fc} to time $t + i\frac{\Delta t_c}{\mathcal{N}}$

The ghost cells in the synchronization layers are updated at the start of the time step. The solutions in these layers are then used to march the entire mesh solution by a time of Δt_c. For each global time step, $\mathcal{N}$ local time steps are performed on the fine block B_f while only one local time step is performed on coarse block B_c. For the case of $\mathcal{N} = 2$ and $\Delta t_c = \Delta t/2$, the MTS scheme would look similar to the illustration in Fig. 5.2. The updation of ghost cells are straight forward as they involve copying the values from the internal element of one mesh to the ghost cells of neighboring mesh.



(a) 1D mesh



(b) Unstructured 2D mesh

Figure 5.3: Illustration of various components of the MTS algorithm (B_f: fine-block, B_c: coarse-block, B_{cf}: fine layer of block B_c, B_{fc}: coarse layer of block B_f).

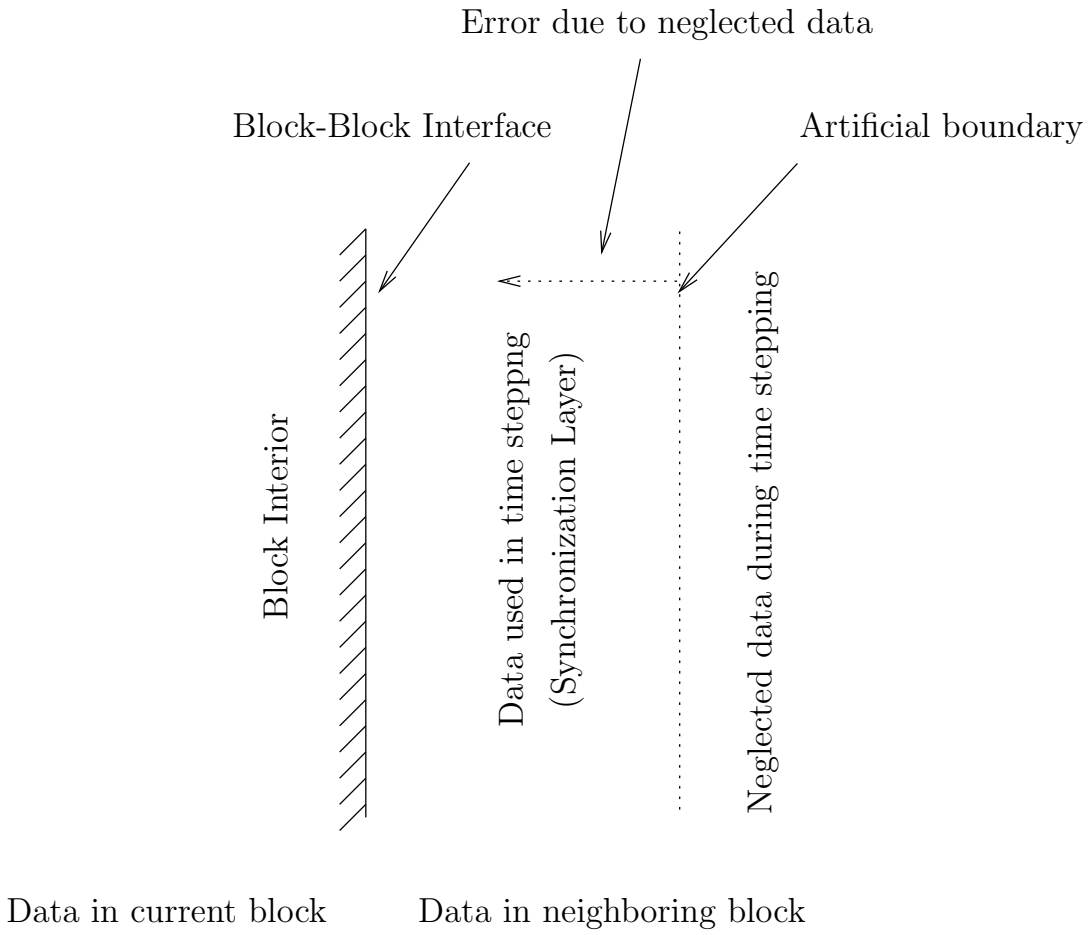


Figure 5.4: Solution evolution of synchronization layers.

The stability of the MTS scheme depends on the local stability limits of the blocks. Note that the local time marching scheme (Steps 3 and 5 of Alg. *MTS2B*) involve solution evolution at both the block interior and its corresponding ghost cells. The maximum local time step size of the block is thus limited by the stability limit of both the interior cells of the block and the ghost cells in its synchronization layers.

The solution evolution process at the synchronization layer is detailed in Fig. 5.4. The truncation of data beyond the synchronization layers results in creation of an artificial boundary. Here, an extrapolation boundary condition is applied for treating this artificial boundary. As the solution evolves in time, the error from these boundaries would propagate towards the block interior as illustrated in Fig. 5.4. The extent of error propagation in the synchronization layer depends on the thickness of the layer and the time interval between the solution updates at the synchronization layers. For a scheme with very low CFL limits, this time interval can be much lower compared to the thickness of the layers. Hence, we perform $\mathcal{S}$ time integration steps at each local time marching step, resulting in the following algorithm:

Algorithm *MTS2BU*

($*$ MTS scheme with deferred updation of a 2 block mesh $*$)

1. Update ghost cells in B_{cf} with solutions from B_f at time t
2. Update ghost cells in B_{fc} with solutions from B_c at time t
3. **for** $i \leftarrow 1$ **to** $\mathcal{S}$
4. **do** March block B_c and B_{cf} to time $t + i\Delta t_c$
5. **for** $i \leftarrow 1$ **to** $\mathcal{SN}$
6. **do** March block B_f and B_{fc} to time of $t + i\frac{\Delta t_c}{\mathcal{N}}$

The MTS algorithm *MTS2BU* now marches the solution by a time step of $\mathcal{S}\Delta t_c$ compared to Δt_c of *MTS2B* algorithm. The value of $\mathcal{S}$ can be viewed as a scaling applied to the CFL number at the synchronization layers. Its value is chosen such that the errors from the artificial boundary do not propagate into the block interior.

A larger value of $\mathcal{S}$ results in more independent time evolution of blocks, which can be exploited to improve the computational performance of the schemes. In this work, the introduction of $\mathcal{S}$ is chiefly used as a way to increase the time step size of multi-step for analyzing the overall stability of the MTS schemes (Section 5.3).

5.2.3 A generic recursive MTS scheme

Practical computations often involve a large number of elements with highly varying element sizes. For such meshes, the MTS scheme can be generalized to handle multiple blocks with the help of a recursive algorithm. The mesh is sub-divided into multiple regions based on the element size. Each region is then associated with a block B_l where the coarseness l increases with increase in element size. The local time step size of the block B_l is taken to be Δt_l. The values of $\mathcal{N}$ and $\mathcal{S}$ are kept constant for all the coarse block - fine block combinations. With these definitions, the recursive MTS algorithm *MTS2BU* is written as:

Algorithm *MTSRecursive(l, t)*

(∗ demonstrates a recursive MTS scheme ∗)

1. **if** $(l+1)$ exists
2. **then**
3. Update ghost cells in coarse layer of B_l with solutions from B_{l+1} at time t
4. **for** $i \leftarrow 0$ **to** $\mathcal{N}-1$
5. **do**
6. **if** $(l+1)$ exists
7. **then**
8. Update ghost cells in fine layer of block B_l with solutions from B_{l-1} at time $t + i\mathcal{S}\Delta t_l$
9. $MTSRecursive(l-1, t+i\mathcal{S}\Delta t_l)$
10. March block B_l and its sync. layers to time $t + (i+1)\mathcal{S}\Delta t_l$
11. **return**

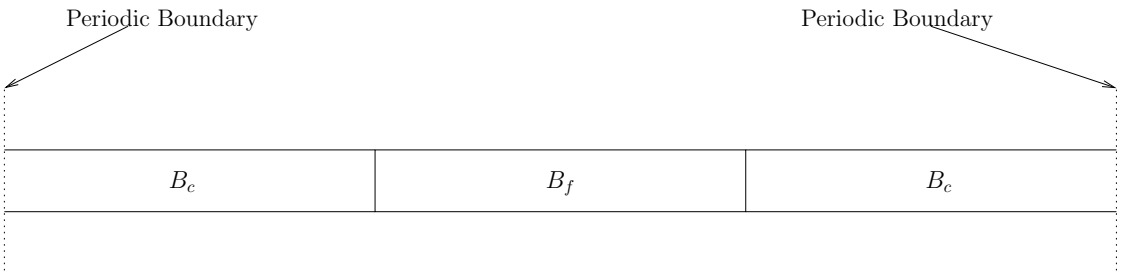


Figure 5.5: Schematic setup for 1D stability analysis of MTS scheme.

The effective time step size of the multi-step algorithm *MTSRecursive(l, t)* of block B_l is $\mathcal{S}\Delta t_l$. During computation, the recursive algorithm is initialized with $MTSRecursive(l_{max}, t^n)$ where l_{max} corresponds to the coarsest block and t^n is the current global time. The actual computation, however, is performed starting from the finest mesh block. At the end of the algorithm, the entire mesh solution is marched to a time of $t^{n+1} = t^n + \mathcal{S}\Delta t_{l_{max}}$. In all the algorithms, the time marching steps (steps starting with "March block") can be performed using any time stepping scheme. This is one of the primary advantage of the proposed MTS scheme, as it inherently supports higher order time integration as well as the hybrid implicit-explicit time stepping schemes [51]. In this work, only the explicit RK-DG schemes are used for all cases.

5.3 1D Stability Analysis

The theoretical limits of the proposed MTS algorithm is examined by analyzing the effective CFL of a single multi-step stage. A periodic domain of unit width is divided in to two blocks B_f and B_c representing the fine and coarse grids. The mesh in this case, however, is constructed using elements of uniform size for realizing the CFL condition across the block interface. A schematic setup of the problem is shown in Fig. 5.5.

In order to perform a matrix stability analysis, the entire process of solution evolution using MTS algorithm *MTS2BU* should be represented in matrix form. The initial step in the MTS algorithm involves refreshing the ghost cell solution

in the synchronization layers with the internal solution of the neighboring blocks. Since the cells are of uniform width, the coarse block - fine block ratio $\mathcal{N}$ is set to a value of 1. The initial solution update process of MTS algorithm *MTS2BU* is written in matrix form as

$$\begin{bmatrix} c^* \\ f^* \end{bmatrix} = [T] \begin{bmatrix} c^n \\ f^n \end{bmatrix} \tag{5.3}$$

where T is the transfer operator used to transfer the solutions between the two blocks and c^*, f^* are the resulting updated values obtained from c^n, f^n. With the fresh solution available at the layers, the blocks B_c and B_f can be marched independently in time. For a linear hyperbolic equation, the time stepping schemes in the individual blocks are simplified as

$$c^{n+1} = G_c c^* \tag{5.4}$$

$$f^{n+1} = G_f f^* \tag{5.5}$$

where c^{n+1} and f^{n+1} are the solutions obtained after one multi-step and G_c, G_f are the amplification matrices obtained as a result of applying $\mathcal{NS}$ time integration steps on the spatial operator. Here the 1D RKDG scheme [44] is used for deriving the amplification matrices G_c and G_f. The derivation of the amplification matrices are not listed here for brevity. Substituting Eqns. (5.4) and (5.5) in Eqn. (5.3), we get

$$\begin{bmatrix} c^{n+1} \\ f^{n+1} \end{bmatrix} = G_{cf} \begin{bmatrix} c^n \\ f^n \end{bmatrix} \tag{5.6}$$

where

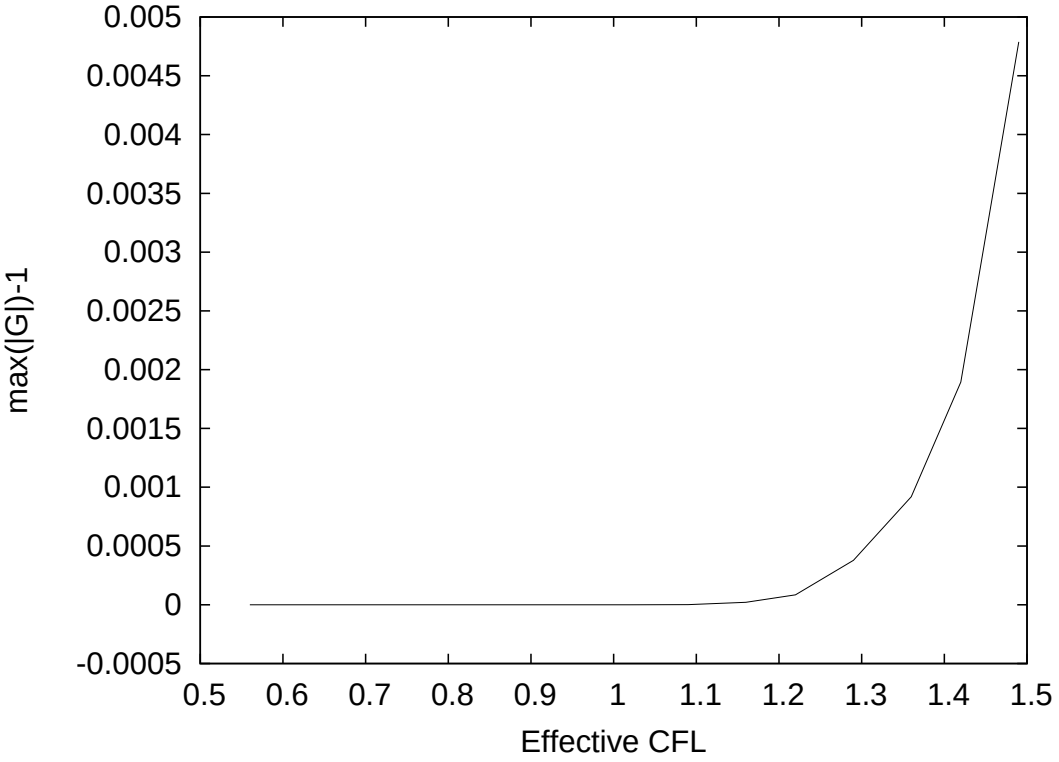
$$G_{cf} = \begin{bmatrix} G_c & 0 \\ 0 & G_f \end{bmatrix} [T] \tag{5.7}$$

The matrix G_{cf} is the so called amplification matrix for the current MTS system $\begin{bmatrix} c \\ f \end{bmatrix}$. The stability of the proposed MTS scheme is determined by analyzing the eigenvalues of the matrix G_{cf}. For the current system, the effective CFL is given by $\mathcal{S}\mathbb{C}$. Since the synchronization layer is of unit cell width, an effective CFL larger than unit value would result in propagation of error into the block boundaries. We perform series of experiments with different $\mathcal{S}$ and a RKDG CFL limit of $\mathbb{C} = 1/(2p+1)$. The influence of effective CFL $\mathcal{S}\mathbb{C}$ on the overall stability of the MTS scheme is analyzed by studying the eigenvalues of the matrix G_{cf} in Eqn. (5.6).

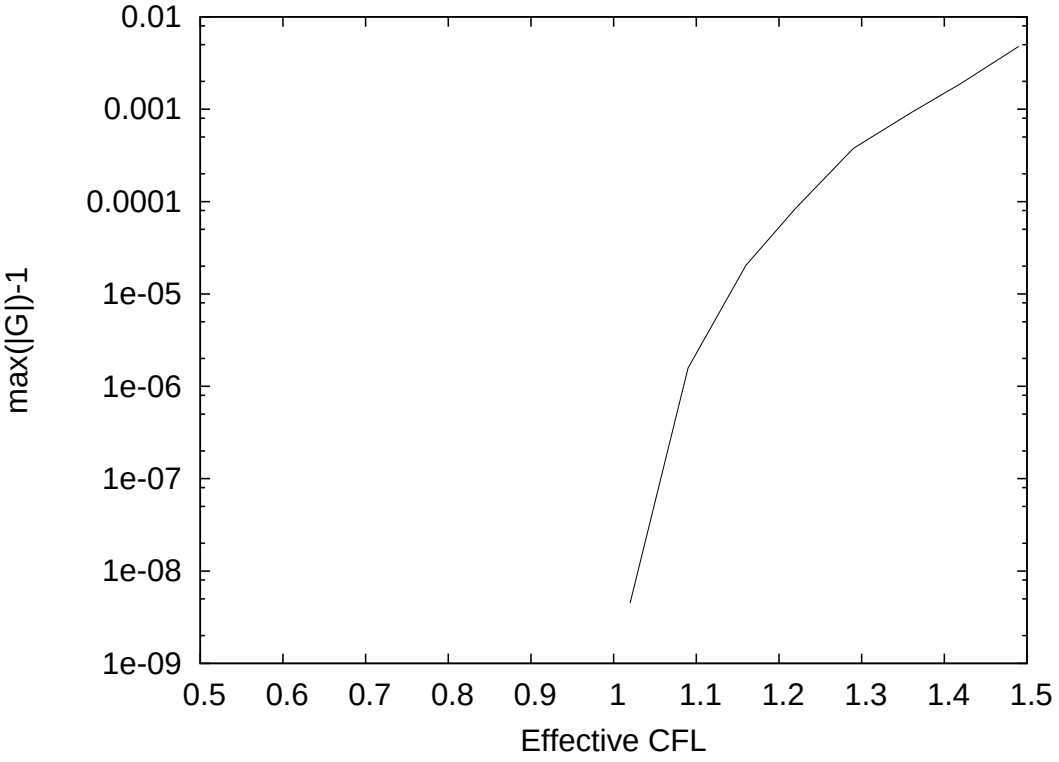
For performing the matrix stability analysis, we use a computational domain of unit length, represented by a grid of 40 elements. The mesh blocks are constructed with 20 elements each. The maximum eigenvalues of the matrix G_{cf} for various values of $\mathcal{S}\mathbb{C}$ is plotted in Fig. 5.6. The scheme becomes unstable when the effective CFL number is greater than unit value. This is expected as the errors from the synchronization layer begin to influence the internal solution of the blocks with increase in $\mathcal{S}\mathbb{C}$. Fig. 5.7 shows the plot of the amplification matrix G_{cf} in eigen space. When $\mathcal{S} = 1$, only a single RK time step is performed between each successive synchronization step. The amplification matrix in this case closely resembles to the pure RK-DG scheme. However, when $\mathcal{S} = 1/\mathbb{C}$, which corresponds to an effective CFL of 1, the eigen modes are pushed towards the maximum limit represented by the circle of unit radius. The eigen modes in all cases ($\mathcal{S}\mathbb{C} \leq 1$) remain within the unit circle.

5.4 Results and Discussion

The proposed MTS algorithm is applied to an existing solver based on the basic unstructured RK-DG solver [44]. The reader is referred to [44] for detailed formulation of the RK-DG scheme. An additional extrapolation boundary condition is required for treatment of the free faces in the synchronization layer. Since each

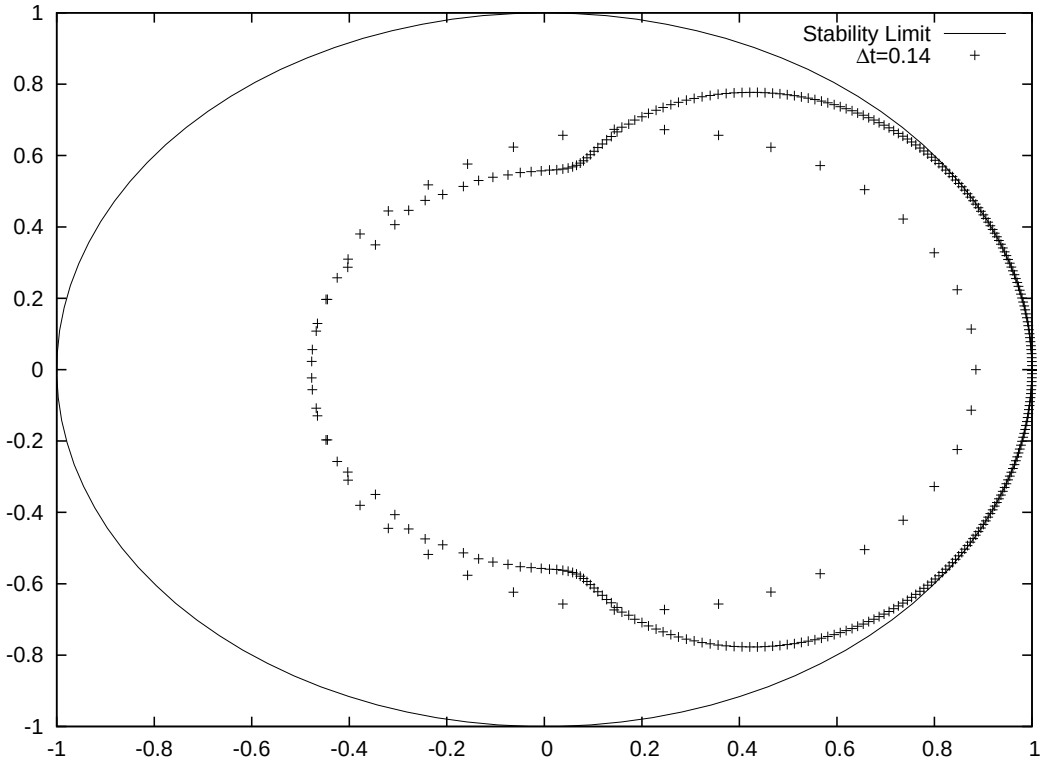


(a) Normal scle

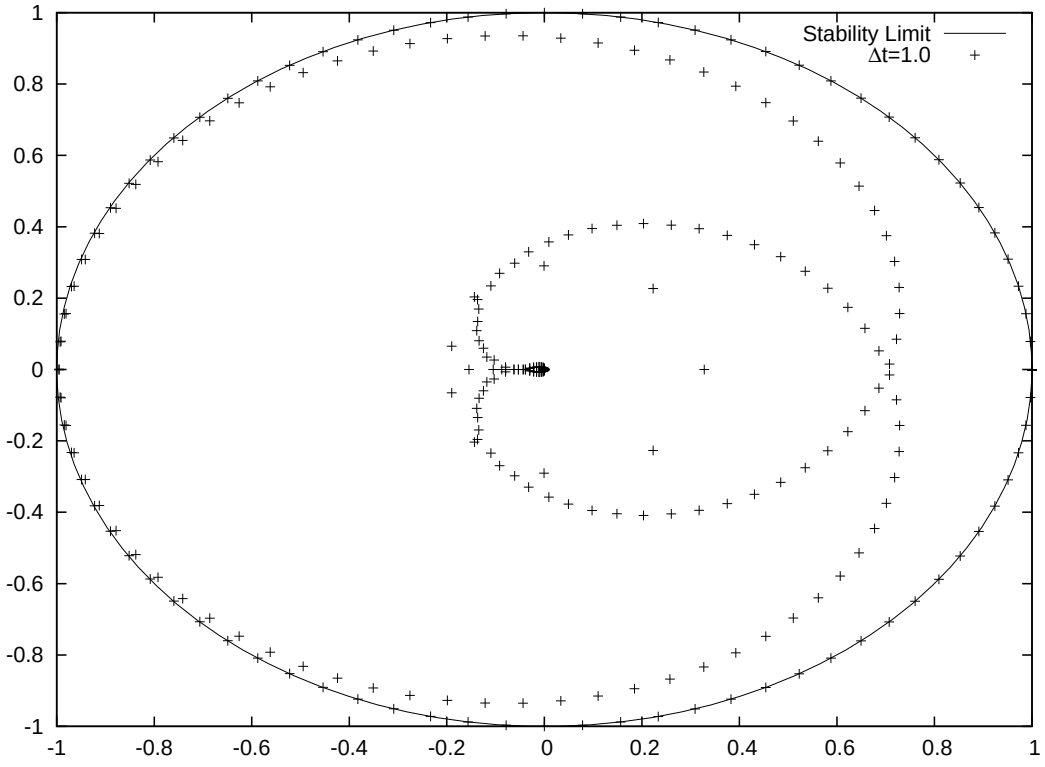


(b) Log scale

Figure 5.6: Stability analysis of the MTS scheme based on a 3^{rd} order RKDG scheme (values ≤ 0 are not included while plotting in log scale).



(a) $\mathcal{SC} = 0.14$



(b) $\mathcal{SC} = 1.0$

Figure 5.7: Plot of amplification matrix G_{cf} corresponding to 4^{th} order RKDG scheme for different values of $\mathcal{N}$ for MTS algorithm (CFL $\mathbb{C} = 1/7$).

block is treated as an individual mesh, all the components of the basic RK-DG solver are retained without any modifications.

The performance of the MTS algorithm is quantified in terms of the computational cost. For a given mesh, the total computational cost per unit time period for a global time stepping scheme (such as Runge Kutta schemes) is given by

$$C_{ERK} = N_{it} N_E \tag{5.8}$$

where N_{it} is the number of iterations per unit time period ($t = 1$), N_E is the number of elements in the mesh. The computational cost is expressed in units of the cost of evaluation of a single element per global time integration step. The number of iterations per unit time period ($t = 1$) is related to the cell size as

$$N_{it} = \frac{1}{\Delta t} = \frac{T_{final}}{\mathbb{C} \min\limits_{\Omega \in \mathcal{M}} \left(\frac{h}{a}\right)} \tag{5.9}$$

where $\mathbb{C}$ is the maximum CFL number of the global time stepping scheme, h is the element size and a is the maximum wave speed inside the element Ω and $\mathcal{M}$ is the global mesh. The computational cost of the explicit RK scheme is now computed as

$$C_{ERK} = \frac{1}{\mathbb{C}} \frac{N_E}{\min\limits_{\Omega \in \mathcal{M}} \left(\frac{h}{a}\right)} \tag{5.10}$$

An ideal explicit scheme utilizes the maximum local time step of all the elements independent of other elements. If the additional costs of local time stepping schemes are neglected, the minimum possible computational cost for a time marching scheme is given by

$$C_{\min} = \sum_{i=1}^{N_E} \frac{1}{\mathbb{C} \frac{h}{a}} \tag{5.11}$$

The MTS schemes, on the other hand, apply the maximum possible time step for the mesh blocks. The mesh is divided into N_B blocks based on the element sizes.

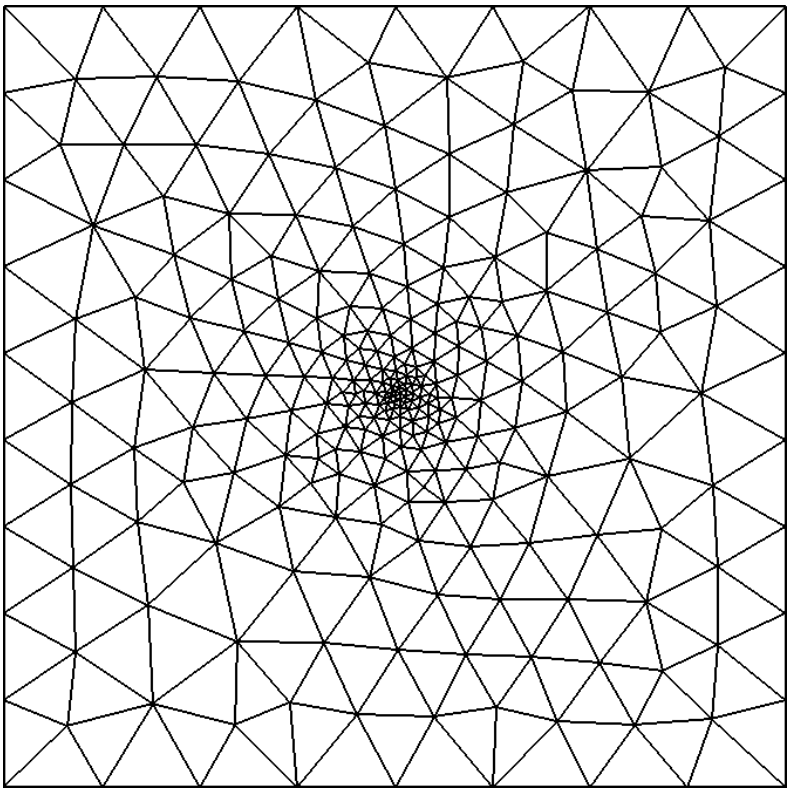


Figure 5.8: Mesh used for isentropic vortex evolution problem.

The associated theoretical computational cost is given by

$$C_{MTS} = \sum_{i=1}^{N_B} \frac{1}{\mathbb{C}} \frac{n_i}{\min\limits_{\Omega \in B_i} \left(\frac{h}{a}\right)} \tag{5.12}$$

where n_i is the number of elements in block B_i and N_B is the number of blocks. The contributions of communication costs and cost of evolving the solution at the synchronization layers is omitted in this approximate analogies of total computational cost.

5.4.1 Isentropic vortex evolution

To test the influence of the MTS scheme on the solution errors, the MTS scheme is applied on a standard case of isentropic vortex evolution. Since the analytical solution is readily available for this case, comparison of the solution errors for different schemes is straight forward. To maintain brevity, the formulation of the isentropic vortex evolution problem is not listed here. The reader is recommended to refer

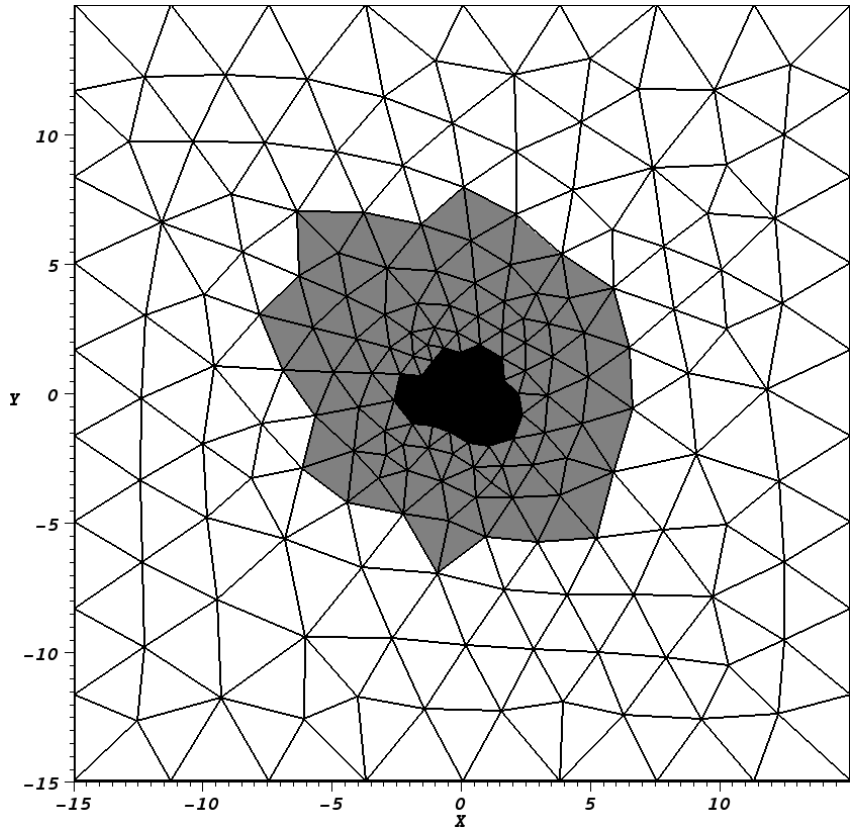


Figure 5.9: Allocation of blocks for the MTS scheme with $N_B = 3$ and $\mathcal{N} = 3$.

		ERK		MTS ($\mathcal{N} = 3$)		
Order	N_E	L_1 error	Cost (sec)	L_1 error	Cost (sec)	Speedup
2	484	4.83E-2	1.77	4.83E-2	0.95	1.86
	1936	1.20E-2	16.01	1.20E-2	6.98	2.29
	7744	2.88E-3	130.83	2.88E-3	52.72	2.48
	30976	7.00E-4	961.37	7.00E-4	407.17	2.36
3	484	8.79E-3	5.59	8.78E-3	3.06	1.83
	1936	1.47E-3	46.89	1.47E-3	21.94	2.14
	7744	2.24E-4	376.25	2.24E-4	164.43	2.29
	30976	3.15E-5	3036.39	3.16E-5	1243.3	2.44
4	484	1.42E-3	20.32	1.42E-3	10.54	1.93
	1936	9.1747e-05	161.62	9.18083e-05	77.94	2.07
	7744	5.62983e-06	1288.87	5.69726e-06	569.09	2.26
	30976	3.72098e-07	10202.81	4.13309e-07	4312.99	2.37

Table 5.1: Solution errors and computational time for explicit RK and MTS schemes ($C_{ERK}/C_{MTS} = 2.42$, N_E : Number of elements).

[93] for the formulation of isentropic vortex evolution test case. For the discontinuous solutions at the faces of the DG elements, Riemann fluxes are required to obtain the solution at the cell interfaces. In this work, the HLLC [86] approximate Riemann fluxes are used as the interface fluxes.

A large periodic domain extending from $(-15, 15)$ to $(15, 15)$ is considered for simulation. The smallest element size is taken to be 0.06 at the center and progressively increasing to a maximum size of 1.5 length units at a distance of 8 length units from the center. The refined meshes are generated by dividing the triangle in to 4 smaller triangles. The coarsest mesh used for current computation is shown in Fig. 5.8.

The formulation of Euler equations and the HLLC fluxes are not detailed here to maintain brevity. The solution is initialized with an isentropic vortex (please see [93] for formulation) of strength 5 and a free-stream velocity of 1 unit along x axis. The computations are performed with the basic explicit Runge Kutta method and the proposed MTS method with $\mathcal{N} = 3$. The solution is evolved over a period of 1 time unit. The errors in the computed solution and the computational cost are compared between the ERK and the MTS scheme.

The solution results are tabulated in Table 5.1. For the mesh employed in obtaining the solution in Table 5.1, the maximum theoretical speed up that can be achieved ($C_{ERK}/C_{\min}$) is around 4.8. For the MTS scheme with three mesh blocks (based on $\mathcal{N} = 3$) as shown in Fig. 5.9, the theoretical speedup (C_{ERK}/C_{MTS}) that can be achieved by the MTS scheme is calculated as 2.42. From Table 5.1, it can be seen that the schemes achieve the required speedup without significant increase in solution errors. It is observed that the speed-up in computation is improved with increase in mesh density. This trend is expected, since the additional cost of synchronization layers relative to the total computational cost will decrease with increase in mesh density.

5.5 Summary

An efficient multi-time stepping scheme is developed for higher order methods on unstructured meshes. Speed up of computational time is achieved by dividing the mesh into blocks and utilizing the local CFL (time step) limit for time marching. The proposed algorithm is tested and validated for higher order Runge Kutta Discontinuous Galerkin method on unstructured meshes in 2D problems. The computed solution is indistinguishable from the original solution and has a speed-up factor of up to 2.5 compared to the original RK-DG scheme. A speed-up factor of 5.2 is achieved in a 6-block (5 recursion levels) configuration with $\mathcal{N} = 2$ in an acoustic wave propagation problem presented in Chapter 7. This shows that the gain in computational cost is dependent on the nature of domain discretization. A more uniform mesh would have lesser gains compared to the mesh with locally refined regions.

The current formulation of the MTS algorithm provides greater solution independence between the blocks of meshes and enables one to use the existing solvers without much change in the source code. Furthermore, the mesh blocks can have independent time stepping schemes. This feature can be exploited by applying Implicit-Explicit schemes as required.

Chapter 6

A new high resolution high order unstructured WENO scheme

The spatial discretization methods such as Discontinuous Galerkin (DG) and Extended Riemann solver (ExRi) are well suited for smooth solutions. In the presence of solution discontinuities, these methods, however tend to produce unphysical oscillations. These oscillations, if not treated, can result in instabilities and corruption of the original solution. Various methods exist to treat these spurious oscillations near the discontinuous solutions. The Weighted Essentially Oscillatory (WENO) schemes are of particular interest since they preserve the formal order of accuracy even near the discontinuous solutions.

An interesting discovery was made in the current work during the analysis of onset of instabilities within an element. When a discontinuity develops in a smooth solution, it was found that the onset of oscillations initially occur only at the element boundaries. These oscillations are typically localized to one face of the element, while the rest of the element solution remains relatively smooth. This phenomena is numerically confirmed and presented in the later part of the chapter (Section 6.3.1). Based on this finding, a reconstruction technique is formulated that suppresses the oscillatory solutions arising at element boundary. Due to the re-use of interior solution, it is possible to restrict the stencil size to just the immediate

neighbors as opposed to the group of neighborhood elements in FV-WENO schemes [70]. These type of schemes with a small stencil size are denoted as compact schemes and hence, here we denote such WENO schemes as compact WENO schemes.

One of the aim of the current work is to extend the compact WENO scheme to very high orders without compromising the overall order of accuracy of the solution and to study the resolution of discontinuities by the higher order schemes. Due to the compact nature of the scheme, higher order reconstructions tend to be highly sensitive to solution oscillations. A constant biasing of the central stencil in the ENO and WENO schemes would therefore result in the corruption of the reconstructed solution by the original oscillatory solution. To prevent this, an adaptive formulation of the WENO stencil weights is proposed here, in which, the central bias is determined based on the relative oscillation of the neighboring solution data. With this formulation, reconstructed solution with accuracy up to 5^{th} order have been achieved, while simultaneously maintaining a high degree of robustness and stability. Efficiency in terms of computational speed and memory is enhanced via the introduction of a oscillation detector and the use of a generalized reconstruction stencil constructed on a reference element. The formulation has also been extended to a three dimensions with suitable applications demonstrating the capability of the scheme to produce oscillation free solution.

6.1 HLLC-LLF Flux formulation

The basic RKDG scheme is same as detailed in Chapter 2 and is not detailed here. The formulation of interface Riemann flux is slightly improved to maintain stability in the vicinity of shocks and to maintain the resolution properties of HLLC scheme. Compared to LLF flux, the HLLC flux can resolve the contact discontinuities thus resulting in a more accurate solution. But, since the HLLC flux is less diffusive, it exhibits a more oscillatory behavior in the vicinity of a strong shock in comparison to the LLF fluxes for the same physical conditions. The oscillations become more severe in the case of higher order schemes ($p > 2$). To make use of the favorable

properties of the HLLC and LLF schemes, a hybrid HLLC-LLF flux is formulated such that the scheme switches to LLF flux in the presence of strong shocks and retains the HLLC flux in other regions. Following the hybrid scheme by Kim et. al. [94], we formulate the HLLC-LLF as

$$F_{HLLC-LLF} = \begin{cases} F_{HLLC} & if \;\; f_F \geq 5 \\ F_{LLF} & otherwise \end{cases} \tag{6.1}$$

where the switch $\mathcal{F}$ is given by

$$f_F = \frac{Min(\boldsymbol{p}_L, \boldsymbol{p}_R)}{|\boldsymbol{p}_L - \boldsymbol{p}_R|} \tag{6.2}$$

As pressure is continuous across contact discontinuities, the scheme naturally uses the HLLC flux. In the presence of shocks with high pressure gradients, the more stable LLF scheme is selected.

6.2 Conventional WENO reconstruction of oscillatory solution

The weak solution computed by the RK-DG schemes generate unphysical oscillations in solution near discontinuities. These oscillations need to be suppressed with the help of artificial diffusion or by suitable solution reconstruction. While the artificial dissipation schemes are formulated as additional terms in the physical equation set, the reconstruction techniques are performed as a post-processing step after each backward Euler step. In the case of RK-DG schemes, this reconstruction process need to be done at each RK stage.

A generic Weighted Essentially Non Oscillatory (WENO) reconstruction procedure can be described as follows:

1. For a given element, construct additional solutions $\mathcal{U}_K$ by including supporting solution data from element neighborhood

2. Estimate the oscillation $\mathcal{O}(\mathcal{U}_K)$ of the solutions $\mathcal{U}_K$

3. Calculate the weights w_K of the solution $\mathcal{U}_K$ such that when oscillation decreases, the corresponding weight increases

4. Provide additional bias to the original solution such that at smooth regions the reconstructed solution reverts back to original solution

5. Compute the WENO solution Q_w as the weighted sum of solutions $\mathcal{U}_\mathcal{K}$

6. Replace the original solution Q with the reconstructed solution Q_w

The solutions $\mathcal{U}_K$ are computed by sampling the solution data over the given stencils K. These stencils include part of the original element solution along with some of the neighborhood solution data. The first stencil ($K = 1$) is taken to be the original solution of the element. The oscillation of the solution Q is calculated [41] as

$$\mathcal{O}(Q) = \frac{1}{V}\int_\Omega \sum_{i=1}^{p}\sum_{j=0}^{i}\left(\frac{1}{i!}\frac{\partial^i f^w(Q)}{\partial\xi^{i-j}\partial\eta^j}\right)^2 d\Omega \tag{6.3}$$

where $f^w(Q)$ is a function used to extract the oscillation information, V is the volume of the element Ω, p is the degree of solution reconstruction and (ξ, η) is the local coordinates of the element. For a component-wise WENO reconstruction, the function $f^w(Q)$ is taken to be the solution Q itself ($f^w(Q) = Q$). For a system of equations, this results in individual estimates of oscillation (and hence WENO weights) for each solution variable. With the available oscillation information of the solutions, the solution weights are computed as

$$w_K = \frac{10^{\lambda_K}\left(\mathcal{O}(\mathcal{U}_K) + \epsilon\right)^{-n_w}}{\sum_{i=1}^{N_K} 10^{\lambda_i}\left(\mathcal{O}(\mathcal{U}_i) + \epsilon\right)^{-n_w}} \tag{6.4}$$

where w_K is the weight of solution $\mathcal{U}_K$, n_w is the exponent to scale the oscillation, ϵ is a small value of 10^{-5} to avoid division by zero and λ_K is the stencil bias with the property ($\lambda_1 \geq 0$ and $\lambda_i = 0, \forall i > 1$). The WENO reconstructed solution Q_w can now be obtained as a linear combination of these solutions as

$$Q_w = \sum_{i=1}^{N_K} w_K \mathcal{U}_K \tag{6.5}$$

where N_K is the number of stencils. For higher order WENO schemes, a larger value of λ_1 is critical for preserving the accuracy and order of convergence of error.

6.3 A new WENO reconstruction scheme for high order schemes

The basic WENO scheme described in Section 6.2 requires

1. Suitable set of stencils K
2. A method to reconstruct the solution $\mathcal{U}_K$ from the stencil data
3. Measure of oscillation of the solution $\mathcal{U}_K$, from which the stencil weights are computed
4. Definition of the bias factor λ_1 for central stencil.
5. A suitable oscillation indicator to determine if WENO reconstruction need to be applied for current element

In the following sections, we detail each of the component of the proposed WENO scheme along with several modifications to make the scheme stable at high orders.

6.3.1 Analysis of onset of oscillation for a pure RK-DG scheme

To reconstruct an oscillation free solution, it is important to analyze the generation of spurious oscillations. For this purpose, a simple Sod shock tube[31] problem is used to test the behavior of the pure RK-DG schemes (without any shock capturing method) in the vicinity of shocks. A Sod shock tube problem is setup with fluids at two different states separated by a thin diaphragm. At the start of the experiment, the diaphragm is removed and the transient flow characteristics are recorded. For

the current experiment, as no shock capturing methods are employed, the higher order RK-DG schemes might become unstable in the first few time steps of solution evolution. To avoid this, the initial solution is smoothed at the diaphragm ($x = 0$) so that the shocks can gradually evolve rather than a sharp initial discontinuity. The initial solution for the diffused Sod shock tube problem is given by

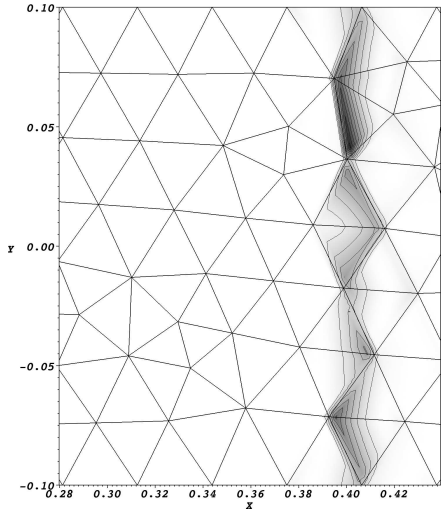
$$(\rho, \mathbf{p}) = (1 - d_f)(\rho_1, \mathbf{p}_1) + d_f(\rho_2, \mathbf{p}_2) \tag{6.6}$$

$$u = 0 \tag{6.7}$$

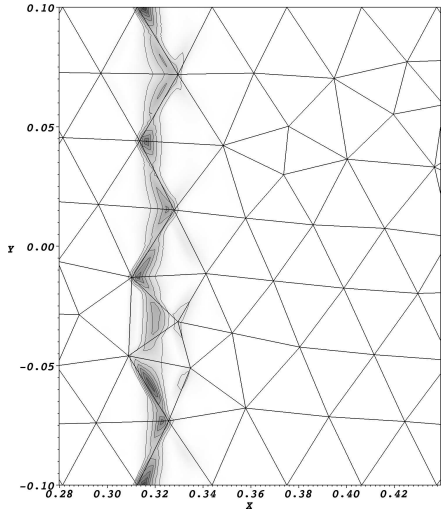
where $(\rho_1, \mathbf{p}_1)$ is the left side density and pressure given by values $(1, 1)$, $(\rho_2, \mathbf{p}_2)$ is the right side density and pressure given by values $(0.125, 0.1)$ and d_f is a smearing function defined as

$$d_f(x) = \begin{cases} 0 & x < -\frac{W_{int}}{2} \\ \frac{(x+1)}{2} & -\frac{W_{int}}{2} \leq x \leq \frac{W_{int}}{2} \\ 1 & x > \frac{W_{int}}{2} \end{cases} \tag{6.8}$$

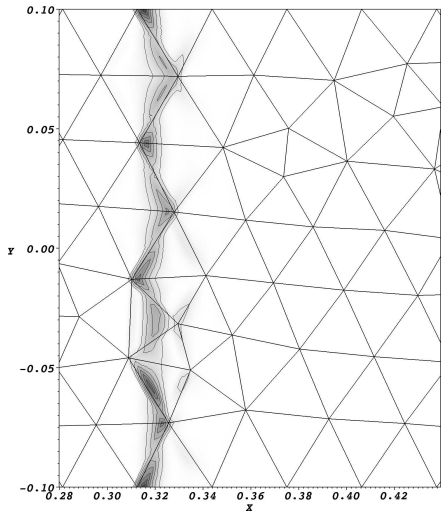
where W_{int} is the width of the interface region describing the change in the states of fluid from left to right. For the current analysis, the width of this region is taken to be $W_{int} = 0.1$. The computational domain is taken to be a rectangular box extending from (-0.5,-0.1) to (0.5,0.1). The solution evolution is carried out till the RK-DG scheme becomes unstable. The plot of density oscillation for 3^{rd}, 4^{th} and 5^{th} order schemes are shown in Fig. 6.1. The solution is extracted at one explicit time step before the onset of instability. The quantity $|\nabla\rho|^2$ is used to visualize the oscillation in the cells. As expected, the higher order schemes fail at an earlier time as they are more sensitive to shocks. A key observation is that, while the solution within the interior of elements is generally smooth, the solution near some nodes and faces of the element exhibit large oscillations. The same phenomena was also observed in other numerical experiments with different initial



(a) p=2, t=0.23



(b) p=3, t=0.176



(c) p=4, t=0.168

Figure 6.1: High resolution plots of oscillation in density ($|\nabla\rho|^2$) for RK-DG schemes at one time step before the solution becomes unstable. Note that the instability occurs at earlier time for higher order schemes (color scales vary with order of scheme).

conditions and meshes. This association suggests that boundary nodal regions may be a source of spurious oscillations; and that to enhance computational stability, a special treatment or consideration may be needed to suppress these spurious oscillations and to prevent them from infecting the solution at the element interior. This is the subject of the new reconstruction scheme described below.

6.3.2 A new reconstruction method for treating oscillatory data

The WENO scheme described in Sec. 6.2 require sufficient number of stencil solutions $\mathcal{U}_K$ to extract the smoothest available data in the vicinity of the element. For a Finite Volume (FV) WENO reconstruction, the solution polynomial is reconstructed by matching the element integral values (or averages) using least squares based technique. Since only the integral values are used, the number of elements in the reconstruction stencil should at least be equal to the number of basis functions. For unstructured grids, in order to avoid singularity problems during coefficient matrix inversion, the number of elements are chosen to be around 50% more than the minimum required [41], resulting in a highly overdetermined system of equations and strong numerical damping, both of which are undesirable.

The DG like schemes can support higher order solutions within the elements. Considering only the integral value of such solution would result in reduction of resolution. It should be noted that, for a reconstruction process, the effective size of the element would be the size of the reconstruction stencil. With the available higher order solution resolution in DG schemes, it is possible to use fewer number of elements for solution reconstruction and yet result in a over determined system of equations. However, reducing the size of the stencil results in a less stable scheme since only solutions near the oscillatory region is considered. Careful treatment of the solution oscillations is therefore vital for computational stability.

In this work, a Weighted Residual (WR) solution reconstruction method is formulated to recover a smooth higher order solution from the available oscillatory

solution data. A simple Galerkin polynomial reconstruction using the element neighbors is given by

$$\sum_{k=1}^{n_K} \int_{\Omega_{j(k)}} \psi \mathcal{U}_K d\Omega = \sum_{k=1}^{n_K} \int_{\Omega_{j(k)}} \psi Q_{j(k)} d\Omega \tag{6.9}$$

where ψ is a trial function, n_K is the number of elements in the stencil K, and $j(k)$ is the mapping of index of the stencil elements to the local index k. The element under consideration corresponds to the first element in the stencil ($k = 1$). In order to ensure the order of accuracy of reconstruction, the set of trial functions (ψ) should be capable of representing all the derivatives of the reconstructed solution $\mathcal{U}$. In this work, we ensure the this condition by choosing the trial function to be same as that of the basis functions used to represent $\mathcal{U}$. With the pre-condition on the trial function (ψ) satisfied, Eqn. 6.9 is now a pure higher order reconstruction of solution states Q. The lack of any stabilization terms in the reconstruction process indicate that no artificial viscosity is introduced in the course of reconstruction.

The preliminary analysis in Sec. 6.3.1 show that the oscillations tend to occur at the element boundaries, while the solution in the element interior is relatively smooth. Thus, when constructing a WENO stencil (for re-sampling the data solution in the element neighborhood), we would like to be able to selectively suppress or remove the oscillations from these regions, so as to produce relatively smoother solution over the whole stencil. To this end, we introduce a filtering operator $\mathbb{F}$ for extracting the oscillation free data from the given element solution. This operator is designed such that the oscillatory solutions near the element boundary are suppressed while the rest of the internal solution is used for reconstruction. In Weighted Residual reconstruction technique based on Galerkin method, the operator $\mathbb{F}$ is introduced via

$$\int_{\Omega_{j(1)}} \psi \mathbb{F} \mathcal{U}_K d\Omega + \sum_{k=2}^{n_K} \int_{\Omega_{j(k)}} \psi \mathcal{U}_K d\Omega = \int_{\Omega_{j(1)}} \psi \mathbb{F} Q_{j(1)} d\Omega + \sum_{k=2}^{n_K} \int_{\Omega_{j(k)}} \psi Q_{j(k)} d\Omega \tag{6.10}$$

where $\Omega_{j(k)}$ represents the k^{th} stencil element and $\Omega_{j(1)}$ is the current element

for which the solution is being reconstructed. In the current reconstruction, the solution oscillations are removed only in the central element for which the reconstruction is performed. Hence, the filter operator $\mathbb{F}$ is not applied on neighboring element solutions. Eqn. (6.10) is used to reconstruct the corresponding stencil solution $\mathcal{U}_K$ from the available neighboring element solution $Q_{j(k)}$. Here, we describe three different types of reconstructions based on different choices of $\mathbb{F}$:

Method of WR by regional weighting (WWR-WENO) In this method, a linear weight function is used for suppressing the oscillatory region near the face or node. The operator for the suppression of Q near the face defined by nodes (i, j) is given by

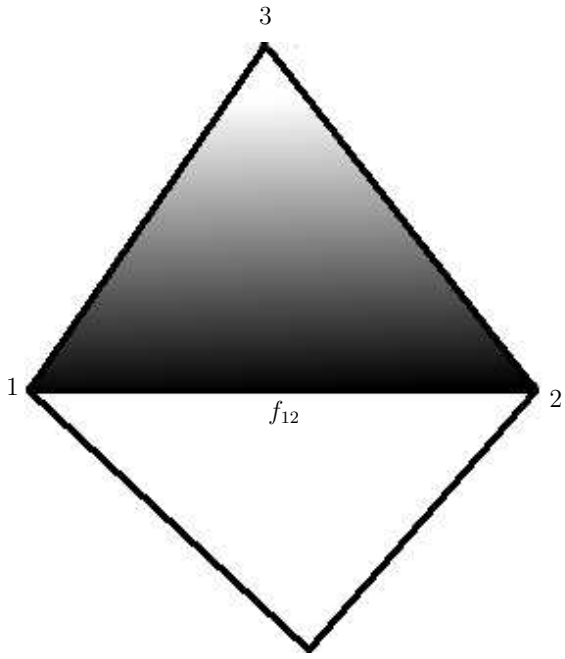
$$\mathbb{F}_{i,j} = \zeta_k \tag{6.11}$$

where ζ_i is the barycentric coordinate having a value of 1 at the i^{th} node and a value of zero at the opposite face defined by nodes (i, j).

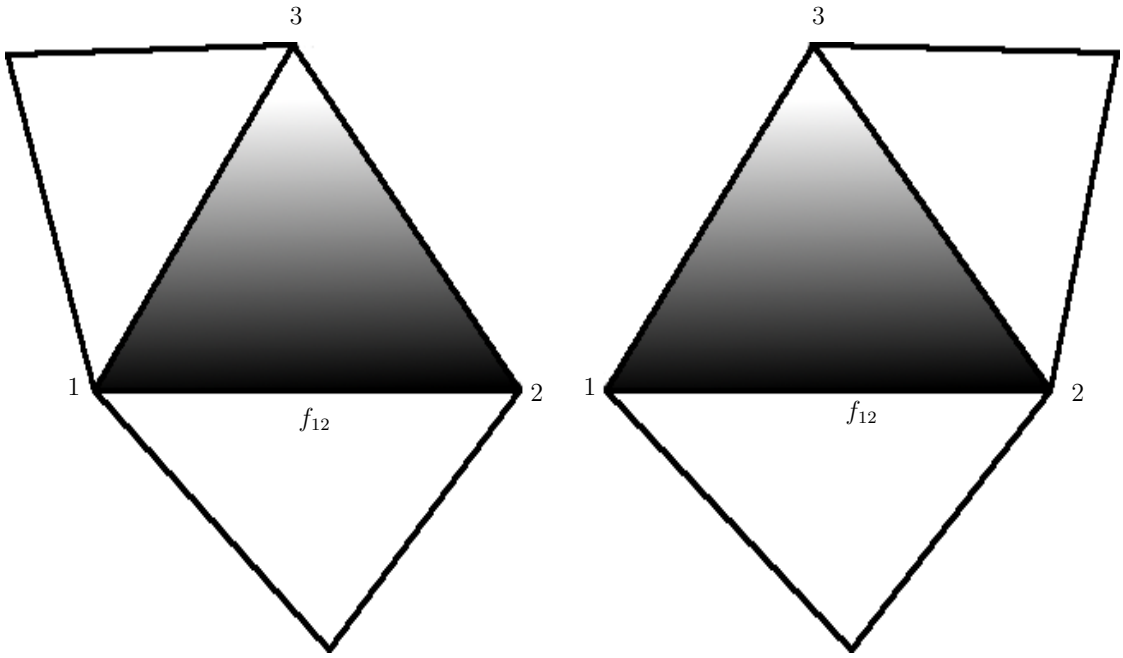
The construction of WWR-WENO stencils based on the operator $\mathbb{F}$ with respect to a single face of the element is illustrated in Fig. 6.2. When the solution near the face is suppressed, both the neighboring elements sharing the face are included in the WENO stencil. This is required to avoid solution overshoots at the face region due to lack of solution data and to enhance the resolution of the discontinuity across the face. Three different solution reconstructions are possible for a given face, thus amounting to a total of 9 stencils. Similar filter operator can be formulated to suppress the regions near the node. Though only the face based operator is used in the current work, we also list the node based operator for completion. The filter operator for the suppression of solution near node i is given by

$$\mathbb{F}_i = (1 - \zeta_i) \tag{6.12}$$

The introduction of the filter operator $\mathbb{F}$ in Eqn. (6.10) could potentially result in loss of accuracy of the solution fit. Nevertheless, it will be later tested with numer-



(a) n_K=2



(b) n_K=3

Figure 6.2: Illustration of operator $\mathbb{F}$ corresponding to face f_{12} for WWR-WENO stencils (Black: $\mathbb{F} = 0$, White: $\mathbb{F} = 1$).

ical experiments to show that the scheme achieves the desired order of accuracy for all orders.

Method of WR by Galerkin method (GWR-WENO) This method can be derived by using a step function for the filter operator. In this filtering scheme, a region near the element boundary is completely neglected ($\mathbb{F} = 0$) while the rest of the region is considered without any modifications ($\mathbb{F} = 1$). This results in reduction of the effective region of the element as shown in Fig. 6.3. The formulation in this case, can be simplified to the standard Galerkin method as

$$\int_{\tilde{\Omega}_{j(1)}} \psi \mathcal{U}_K d\Omega + \sum_{k=2}^{n_K} \int_{\Omega_{j(k)}} \psi \mathcal{U}_K d\Omega = \int_{\tilde{\Omega}_{j(1)}} \psi Q_{j(1)} d\Omega + \sum_{k=2}^{n_K} \int_{\Omega_{j(k)}} \psi Q_{j(k)} d\Omega \quad (6.13)$$

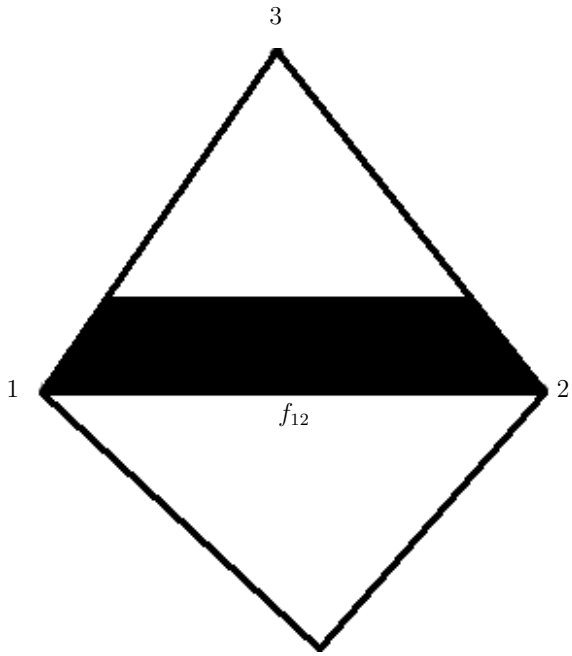
where $\tilde{\Omega}_{j(1)}$ is the shrunk element obtained from $\Omega_{j(1)}$ by scaling the element $\Omega_{j(1)}$ with a scaling factor of l_0 with respect to the node opposite to the oscillatory face. A factor of $l_0 = \frac{1}{3}$ is used in Fig. 6.3. A value of ℓ_0 is set to $\frac{1}{2}$ in the computations reported in this paper

Method of WR by uniform suppression of solution (QWR-WENO) For this method, the entire element solution $Q_{j(1)}$ is suppressed by a constant weighting function. For this configuration, the operator in Eqn. (6.10) is defined as

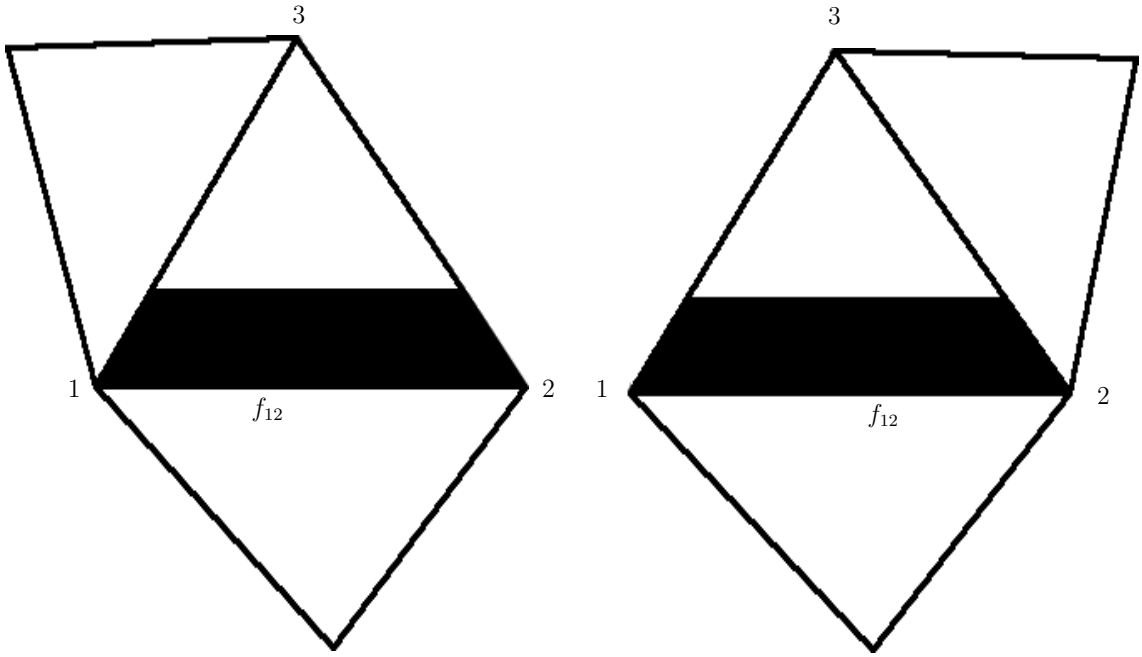
$$\mathbb{F} = \frac{1}{4} \quad (6.14)$$

As illustrated in Fig. 6.4, this configuration does not differentiate the oscillatory regions within the element. Thus only a maximum of 6 stencils (three 2 element stencils and three 3 element stencils) can be formulated for WENO reconstruction.

All the above configurations can be easily extended to higher order schemes without any change in formulation.

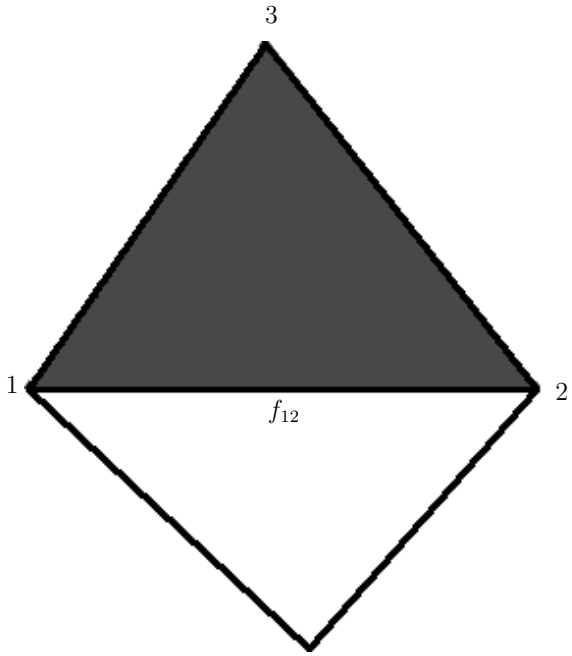


(a) n_K=2

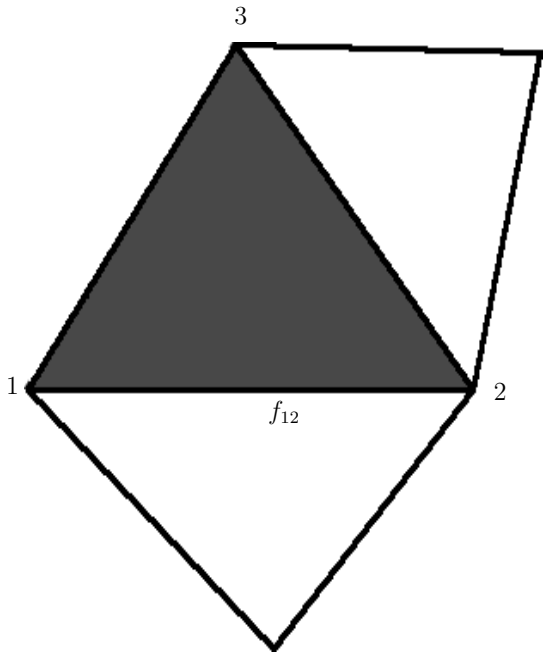


(b) n_K=3

Figure 6.3: Illustration of operator $\mathbb{F}$ corresponding to face f_{12} for GWR-WENO stencils (Black: $\mathbb{F}(1) = 0$, White: $\mathbb{F}(1) = 1$).



(a) n_K=2



(b) n_K=3

Figure 6.4: Illustration of operator $\mathbb{F}$ for QWR-WENO stencils (Black: $\mathbb{F}(1) = 0$, White: $\mathbb{F}(1) = 1$).

6.3.3 Enforcing conservation of variables

The WENO reconstructions formulated above is not conservative unless the conservation condition is imposed while reconstructing the stencil solutions $\mathcal{U}_K$. If exact matrix inversion is used, this conservation property is strictly enforced while computing $\mathcal{U}_K$ as in [72]. In the case of least squares technique, an additional constraint needs to be solved for enforcing the conservation property [41]. Here we enforce the conservation by imposing a simple condition on the basis function, that, for all the basis functions except the unit function ϕ_1, the integrals should vanish. This results in the condition

$$\int_\Omega \phi_i d\Omega = 0 \quad \forall i > 1 \tag{6.15}$$

where ϕ_i is the i^{th} basis function. Since $\phi_1 = 1$, the conservation condition now reduces to

$$\alpha_1 = \frac{1}{V} \int_\Omega Q d\Omega \tag{6.16}$$

where α_i is the coefficient corresponding to ϕ_i. Any set of hierarchical basis functions can be modified to the above form with no additional computational complexity. Since the first basis function α_1 remains unchanged to maintain conservation, it is not included in the reconstruction process.

6.3.4 Generalization of WENO stencils for large scale problems

The WENO reconstructions involve costly matrix inversions (Eqn. 6.10) for sampling the solution polynomial in the stencil. To avoid this inversion at each time step, the coefficient matrices are pre-computed and stored in memory for a given grid. The size of this coefficient matrix increases with the number of basis functions used in expansion of solution polynomial. Also, a larger number of stencils is preferred to perform a smooth WENO reconstruction. Storage of large coefficient

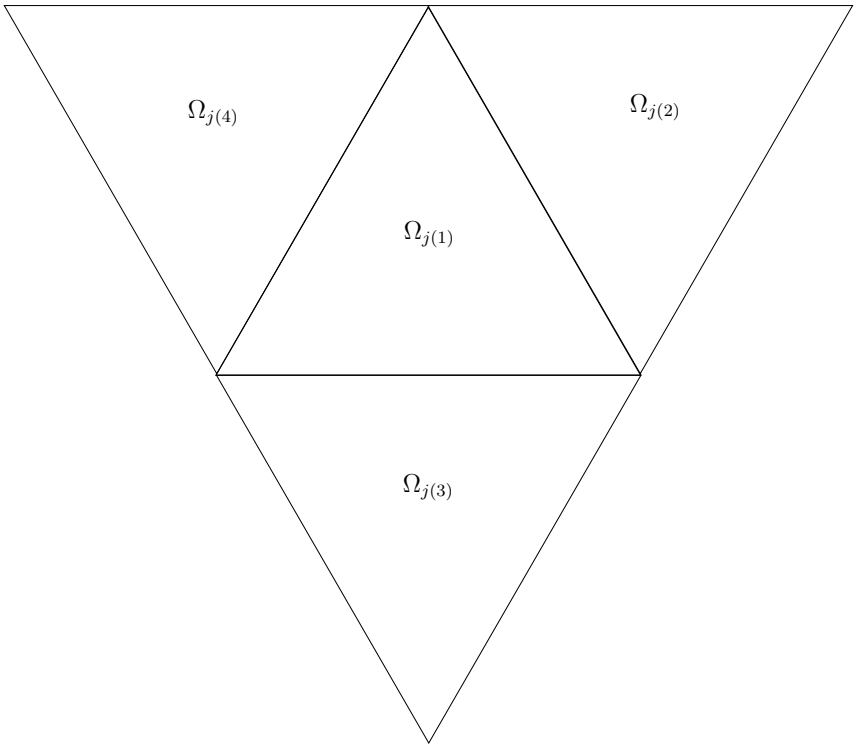
matrices for a large number of stencils for each element will be computationally costly and in some cases might be impractical.

For a mesh of adequate quality, the adjacent element do not differ much in shape and size. In such cases, it is possible to construct the WENO stencils in the reference plane (ξ, η), thereby reducing the memory requirement significantly. Fig. 6.5 shows the schematic setup of the WENO stencils in the reference plane. The original element corresponds to element $\Omega_{j(1)}$ in the figure and the three neighbors are numbered consequently from $j(2)$ to $j(4)$. In an isotropic mesh, the neighboring stencil elements would be of same size as that of $\Omega_{j(1)}$. But, when the mesh is distorted, the nodes of the stencil element could lie outside the actual neighboring element, resulting in extrapolation of the neighboring solution from the actual neighboring element to the stencil element. To avoid this, the stencil elements are scaled down by a factor $\ell_n \leq \frac{1}{2}$ along the normal direction of the face f_i. However, for a highly distorted element, this method would still require extrapolation of neighboring solutions. For such elements, we construct the stencils with the actual neighboring elements rather than the reference element.

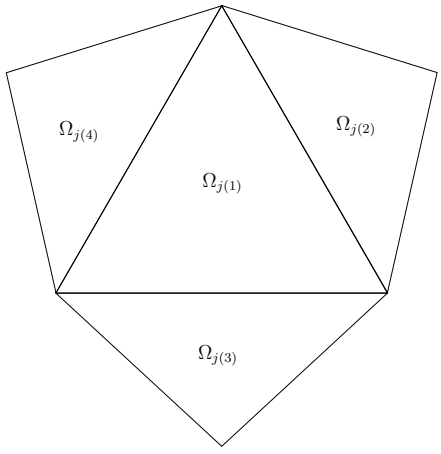
The generalized WENO stencils constructed in reference plane requires evaluation of the solution values at the integration/interpolation points for each stencil element. This additional cost is negligible compared to the total cost of WENO reconstruction. Besides, if we employ the oscillation detectors (Section 6.3.6), the WENO reconstruction is performed only on the troubled cells. A small increase in computational cost of WENO reconstruction is thus justified compared to the savings in the memory requirement. A value of $\ell_n = \frac{1}{2}$ is set as the default value as it results in a stable scheme with reasonable resolution.

6.3.5 A new adaptive WENO weight computation for higher order schemes

Computation of non-linear stencil weights (Eqn. 6.4) is a key component of underlying WENO scheme. A common way to construct a WENO solution is to compute



(a) $\ell_n = 1$



(b) $\ell_n = \frac{1}{2}$

Figure 6.5: WENO stencil in reference plane with different scaling of neighboring elements.

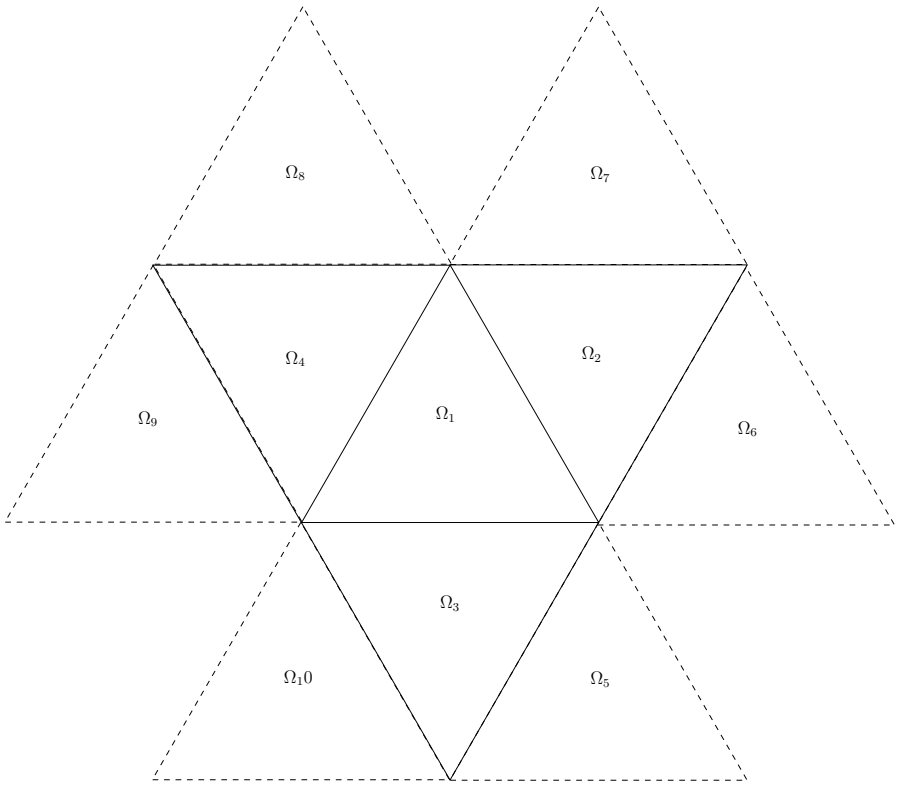


Figure 6.6: Levels of element neighbors used to compute relative oscillation $\mathcal{R}$ (Solid line: first level; Dashed line: second level).

the weights individually for each solution variable ($f^w(Q) = Q$). Rather than using individual weights for each solution variable, here we use a common stencil weighting for all the solution variables. This is equivalent of applying a constant stencil weight for the entire solution state. The choice of this function $f^w(Q)$ is critical, as it needs to address the oscillation of all the solution variables. For Euler equations, the function $K(Q)$ is calculated as

$$f^w(Q) = c = \sqrt{\frac{\gamma \mathbf{p}}{\rho}} \tag{6.17}$$

where c is the speed of sound. It is observed that the use of a common oscillation function results in a more stable reconstruction compared to the component-wise reconstruction, particularly at regions where negative density can occur.

A larger bias towards original solution (central stencil) improves solution accuracy. However, in the case of an oscillatory solution, a larger value of λ_1 will result in corruption of the reconstructed solution by the original oscillatory so-

lution, thereby prevent the WENO reconstruction from effectively removing the oscillations. Numerical experiments suggest that the WENO reconstruction becomes more sensitive to the choice of λ_1 and n_w in Eqn. (6.4) for schemes of order ≥ 4. To avoid this, we choose an adaptive central weight λ_1 and exponent n_w. When the element solution is less oscillatory than the neighboring solutions, the value of $\lambda_1$is increased and n_w is decreased. This increases the bias towards the original solution. If the solution is more oscillatory, this bias is decreased by decreasing λ_1 and increasing n_w. To enable this, we introduce a measure of the relative oscillation $\mathcal{R}$ of the element with respect to its neighbor as

$$\mathcal{R} = \frac{Min\left(\mathcal{O}\left(Q_{j(1)}\right)...\mathcal{O}\left(Q_{j(N_N)}\right)\right) + \epsilon}{\mathcal{O}\left(Q_{j(1)}\right) + \epsilon} \tag{6.18}$$

where $Q_{j(1)}$ is the element solution before reconstruction and N_N is the number of neighbors chosen recursively for computing $\mathcal{R}$. We limit the number of neighbors to two levels of von Neumann neighbors as shown in Fig. 6.6. The value of $\mathcal{R}$ as formulated in Eqn. (6.18) decreases towards zero if the element solution is more oscillatory than the neighboring solutions and increases to 1 if the element solution is smoother compared to the neighboring solution. With the value of $\mathcal{R}$ known, the values of λ_1 and n_w are computed in an adaptive manner as

$$\lambda_1 = \mathcal{R}p \tag{6.19}$$

$$n_w = (1 - \mathcal{R})\, p \tag{6.20}$$

From Eqn. (6.19) and Eqn. (6.20), the bias towards the original solution will be decreased if an oscillatory solution exists in the neighborhood. This enables one to effectively use the WENO reconstruction technique on a very high order scheme without any instability problems. This technique requires no free parameter, and the formulation is fixed for all WENO schemes listed here.

6.3.6 Oscillation detectors

For large scale problems, it is often desirable to perform the expensive WENO reconstructions as and when it is required. The need to carry out a WENO reconstruction for an element is determined by analyzing the variation of solution oscillation $\mathcal{O}(Q)$ in the element neighborhood. The oscillation detector χ used for determining the relative level of oscillation in the element neighborhood is given by

$$\chi = \left(\frac{Min(\mathcal{O}(Q_1)...\mathcal{O}(Q_{N_N})) + \epsilon}{Max(\mathcal{O}(Q_1)...\mathcal{O}(Q_{N_N})) + \epsilon} \right)^p \tag{6.21}$$

The WENO reconstruction is performed only when the following conditions are met:

$$\chi < 0.3 \tag{6.22}$$

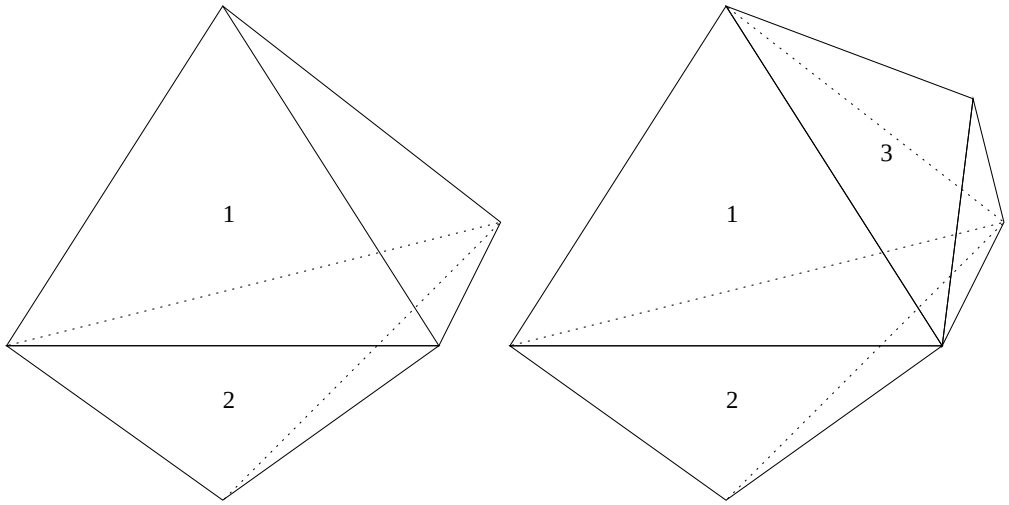
$$\mathcal{O}(Q_1) > Min(\mathcal{O}(Q_1)...\mathcal{O}(Q_{N_N})) \tag{6.23}$$

With these conditions, the reconstruction is now enabled only in the cells that have large oscillations in the element neighborhood. The performance of oscillation detectors is demonstrated on the Double-Mach reflection test case in Sec. 6.4.6.

6.3.7 Extension of WWR-WENO scheme to 3D

The stencil formulations detailed above have been formulated for 2D triangle elements. It is straight forward to extend the same formulation to 3D elements. Much of the formulations of the WENO scheme such as oscillation detection, non-linear weight computation, adaptive biasing etc. are similar to that of the 2D formulations and are not detailed here to avoid repetition.

The WWR-WENO scheme is used for the tetrahedral elements. The solution near the faces are suppressed with the formulation given in Eqn. (6.11). The corresponding solution reconstructions are performed based on Eqn. (6.10). Similar to the 2D reconstruction, while generating the WENO stencils, both the elements sharing the face near the oscillatory region are included. In order to limit the



(a) 2 element stencil (b) 3 element stencil

Figure 6.7: Illustration of stencils for 3D tetrahedral elements for face shared by elements 0,1.

number of stencils, a maximum of two neighboring elements are used to construct the stencils. This results in a set of 4 stencils for each face, thus resulting in a total of 16 stencils for a given element. Fig. 6.7 illustrates the 3D stencils for a tetrahedral element. Only one of the 3-element stencils is shown for illustration.

6.4 Results and Discussion

The proposed WENO reconstruction technique is applied for various benchmark problems. The order of accuracy is established with the isentropic vortex evolution problem. The performance of the schemes in the presence of discontinuities are then analyzed with the standard Sod shock tube problem on a 2D mesh. Numerical solutions are compared with the analytical solutions for both the isentropic vortex and the Sod shock tube problem. A more complex simulation is then performed with the 2D Riemann problem and the shock-bubble interaction problem. The effect of h and p refinement on the shock resolution is studied in detail by simulating a double mach reflection. The capability of method in computing three dimensional flows is demonstrated with the a spherical explosion problem and a 3D shock-bubble interaction problem.

Since the current simulations are conducted with inviscid fluid models, the shock interactions can result in much finer structures. A diffusive scheme like the FV WENO reconstruction scheme would diffuse these small structures even if the scheme has high order of accuracy. Hence, the resolution of these flow structures and discontinuities are used as a qualitative measure of the resolution of the WENO schemes. Both h and p refinement strategies are used to compare the resolution characteristics of the schemes and better understand their applicability for problems involving shocks and contact discontinuities.

The computational domain is discretized with the "frontal" algorithm in the open-source grid generator GMSH [79]. For normal plotting, only the cell average values are used. The high resolution plots are obtained by sub-dividing the elements into smaller elements depending on the order of the scheme. The solution visualization using high resolution (refined elements) plots tend to be more oscillatory as the solutions are discontinuous across the elements. "Visit", a VTK based viewer [95] is used for plotting the solution using cell centered VTK data. The solver is parallelized using Message Passing Interface (MPI) and is run on Rocks cluster with AMD Opteron processors.

6.4.1 Numerical accuracy test using isentropic vortex evolution

The problem consists of an isentropic vortex convecting at a constant speed in the free stream. This is in general used as a test case for estimating the order of accuracy of a scheme since the solution is smooth and analytical results are readily available. An isentropic vortex at a point (0,0) is initialized by the following equations

$$\rho = \left(1 - \frac{(\gamma - 1)\,\beta^2 exp\,(1 - r^2)}{8\gamma\pi^2}\right)^{\frac{1}{\gamma-1}} \tag{6.24}$$

$$u = 1 - \frac{\beta y}{2\pi} exp\left(\frac{1 - r^2}{2}\right) \tag{6.25}$$

		RK-DG				WWR-WENO			
Order	h	L_1 err.	L_∞ err.	$O(L_1)$	$O(L_\infty)$	L_1 err.	L_∞ err.	$O(L_1)$	$O(L_\infty)$
2	1	3.67e-3	1.42e-3			6.58e-3	2.92e-3		
	$\frac{1}{2}$	9.27e-4	2.93e-4	1.98	2.28	2.02e-3	9.27e-4	1.7	1.66
	$\frac{1}{4}$	1.88e-4	6.87e-5	2.3	2.09	4.86e-4	1.68e-4	2.06	2.47
3	1	8.24e-4	4.06e-4			1.50e-3	5.80e-4		
	$\frac{1}{2}$	7.86e-5	2.08e-5	3.39	4.29	1.34e-4	3.68e-5	3.48	3.98
	$\frac{1}{4}$	6.85e-6	3.75e-6	3.52	2.47	9.32e-6	4.17e-6	3.85	3.14
4	1	1.51e-4	3.75e-5			6.63e-4	1.39e-4		
	$\frac{1}{2}$	8.79e-6	4.75e-6	4.1	2.98	1.40e-5	3.67e-6	5.56	5.25
	$\frac{1}{4}$	4.75e-7	3.15e-7	4.21	3.91	6.98e-7	3.64e-7	4.33	3.33
5	1	3.15e-5	1.09e-5			5.94e-5	1.32e-5		
	$\frac{1}{2}$	1.18e-6	4.18e-7	4.74	4.7	1.68e-6	4.21e-7	5.15	4.97
	$\frac{1}{4}$	1.11e-7	2.45e-8	3.41	4.09	1.19e-7	2.43e-8	3.81	4.11
		GWR-WENO				QWR-WENO			
2	1	6.53e-3	2.82e-3			6.35e-3	2.80e-3		
	$\frac{1}{2}$	1.99e-3	8.85e-4	1.71	1.67	1.84e-3	8.46e-4	1.79	1.73
	$\frac{1}{4}$	4.78e-4	1.58e-4	2.06	2.49	4.16e-4	1.43e-4	2.15	2.57
3	1	1.67e-3	5.99e-4			1.24e-3	5.07e-4		
	$\frac{1}{2}$	1.34e-4	3.58e-5	3.64	4.06	1.12e-4	3.09e-5	3.46	4.04
	$\frac{1}{4}$	9.44e-6	4.24e-6	3.83	3.08	8.29e-6	3.85e-6	3.76	3
4	1	5.63e-4	1.39e-4			4.23e-4	1.06e-4		
	$\frac{1}{2}$	1.10e-5	3.35e-6	5.68	5.38	1.08e-5	3.35e-6	5.29	4.99
	$\frac{1}{4}$	5.20e-7	3.58e-7	4.4	3.23	5.23e-7	3.37e-7	4.36	3.31
5	1	5.19e-5	1.33e-5			3.89e-5	1.20e-5		
	$\frac{1}{2}$	1.35e-6	3.93e-7	5.26	5.08	1.24e-6	4.31e-7	4.97	4.8
	$\frac{1}{4}$	1.16e-7	2.44e-8	3.55	4.01	1.16e-7	2.47e-8	3.42	4.13

Table 6.1: Grid convergence analysis of WENO schemes for solution of isentropic vortex evolution at $t = 5$.

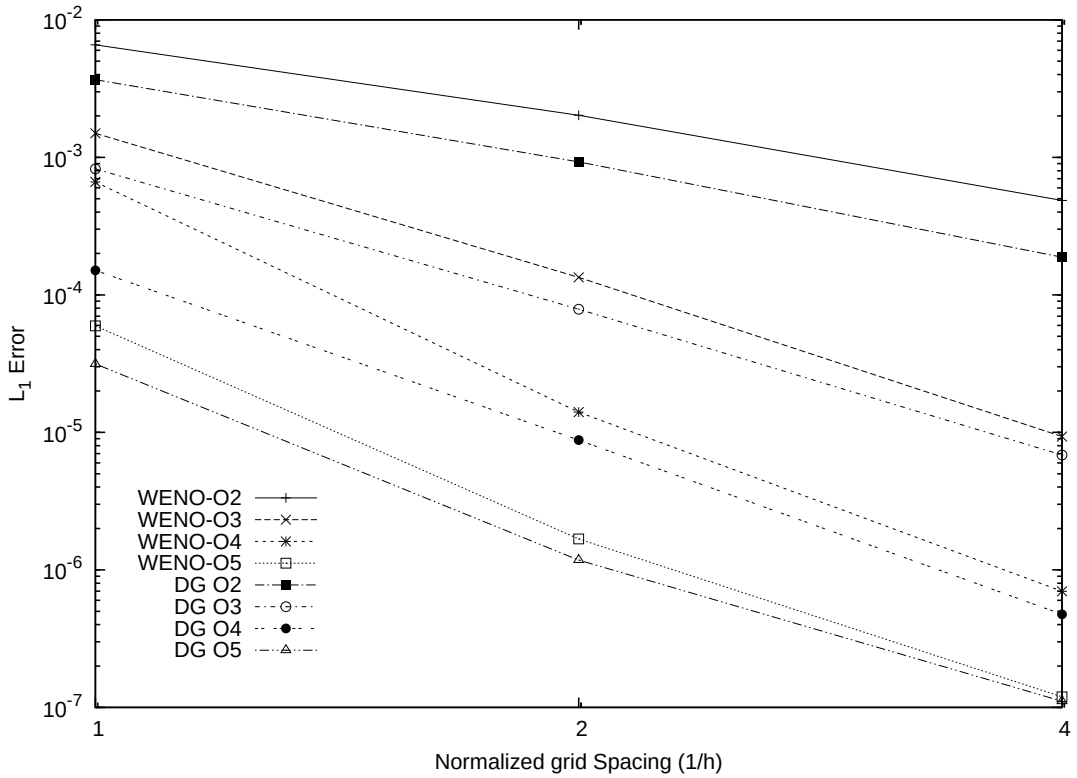


Figure 6.8: Comparison of convergence characteristics of WWR-WENO scheme with respect to pure DG scheme for isentropic vortex evolution problem.

$$v = \frac{\beta x}{2\pi} exp\left(\frac{1-r^2}{2}\right) \tag{6.26}$$

$$\mathbf{p} = \rho^{\gamma} \tag{6.27}$$

where $\beta = 3$ is the strength of the vortex and $r = \sqrt{x^2 + y^2}$ is the distance from the center of the vortex. The computation is performed in a periodic of size 10x10. For numerical analysis, we compare the numerical results at $t = 10$ at which the exact solution coincides with the initial solution. The CFL condition is chosen such that the error due to the time stepping is negligible. Successive refinements are performed by dividing each triangle into four smaller triangles. For the current computation, the WENO scheme is applied on all the cells, irrespective of the relative solution oscillation.

The solution errors and grid convergence characteristics for the WENO schemes are tabulated in Table 6.1. The QWR-WENO scheme has the least solution error. This could be due to the utilization of the original solution without any modifica-

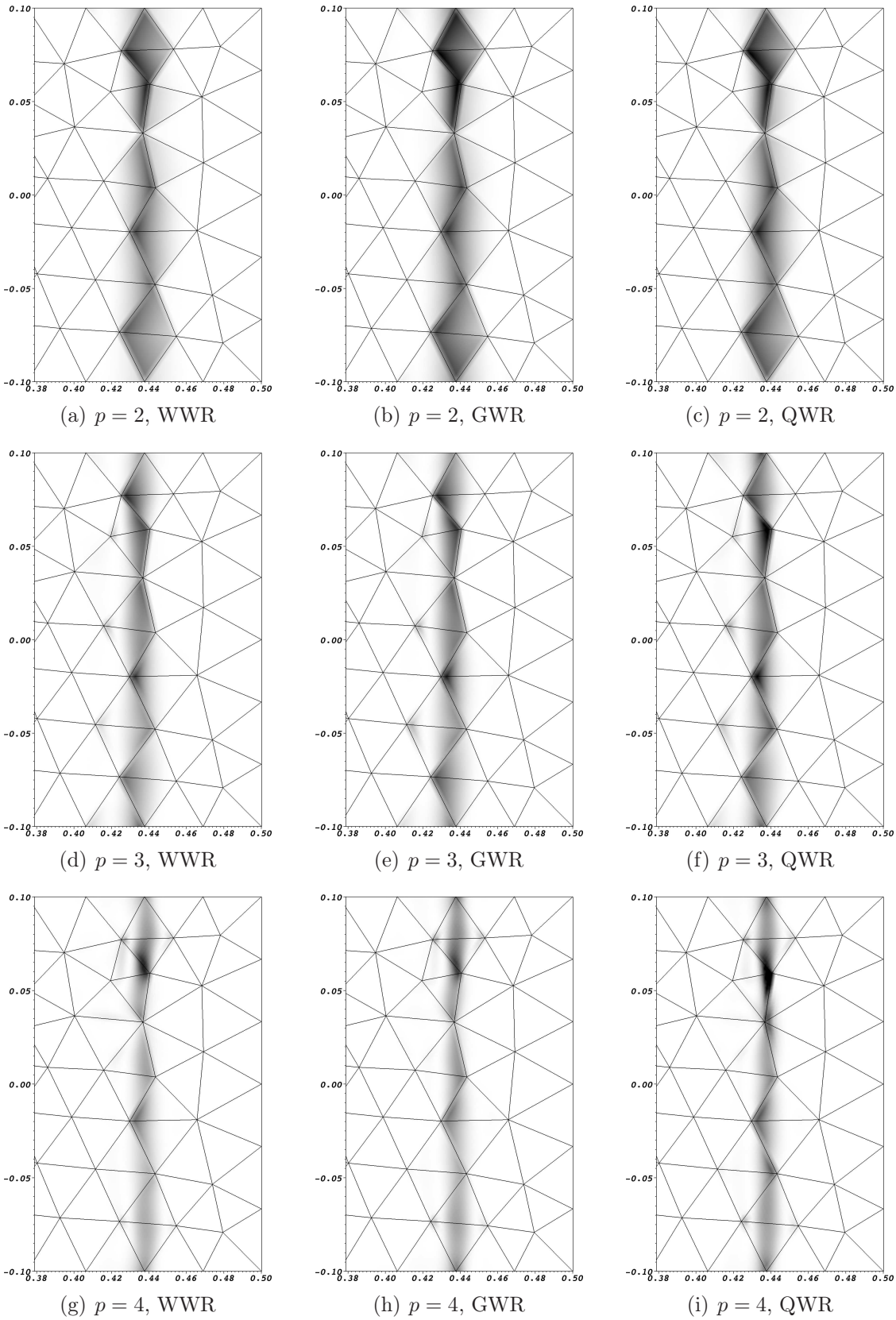
p	Density Error	Velocity Error	Pressure Error
1	0.0101	0.0208	0.0084
2	0.0056	0.0110	0.0046
3	0.0051	0.0090	0.0042
4	0.0048	0.0081	0.0040

Table 6.2: Solution errors (L_1) computed using WWR-WENO schemes for shock tube problem at time $t = 0.25$.

tion. However, it will be shown Sec. 6.4.2 that the scheme tends to be unstable for higher orders. The WWR and GWR scheme have similar performance characteristics. All the schemes exhibit the designed order of accuracy. As grid size reduces, the local solution of an element becomes smoother in comparison to coarse grids. This results in an increase in λ_1 thereby reducing the influence of the WENO reconstruction. As a result, the errors seem to converge faster than the designed order of accuracy. This trend can be observed in the order of convergence listed in in Table 6.1. For all the orders up to 4, the observed order of accuracy in coarser grids is higher than the designed order. In the case of 5^{th} order schemes, the order of convergence tend to decrease with mesh refinement as the error is now limited by the machine accuracy. However, with mesh refinement, the solution errors of the WENO scheme converge to that of unmodified RK-DG scheme, which is the minimum error possible for a smooth solution. This is also confirmed in the plot of solution errors in Fig. 6.8.

6.4.2 1D Riemann problem

The Sod shock tube problem detailed in Section 6.3.1 is used with the width W_{int} set to 0, resulting in exact initial condition at time $t = 0$. All the computations are run till a physical time of $t = 0.25$ on a mesh with average cell size of around 0.03. The density oscillation ($|\nabla\rho|^2$) for various schemes are plotted in the vicinity of the shock in Fig. 6.9. For all the orders of the scheme, the WWR-WENO scheme results in the smoothest solution, while the QWR-WENO scheme results in the most oscillatory solution. The GWR-WENO scheme is more oscillatory relative to

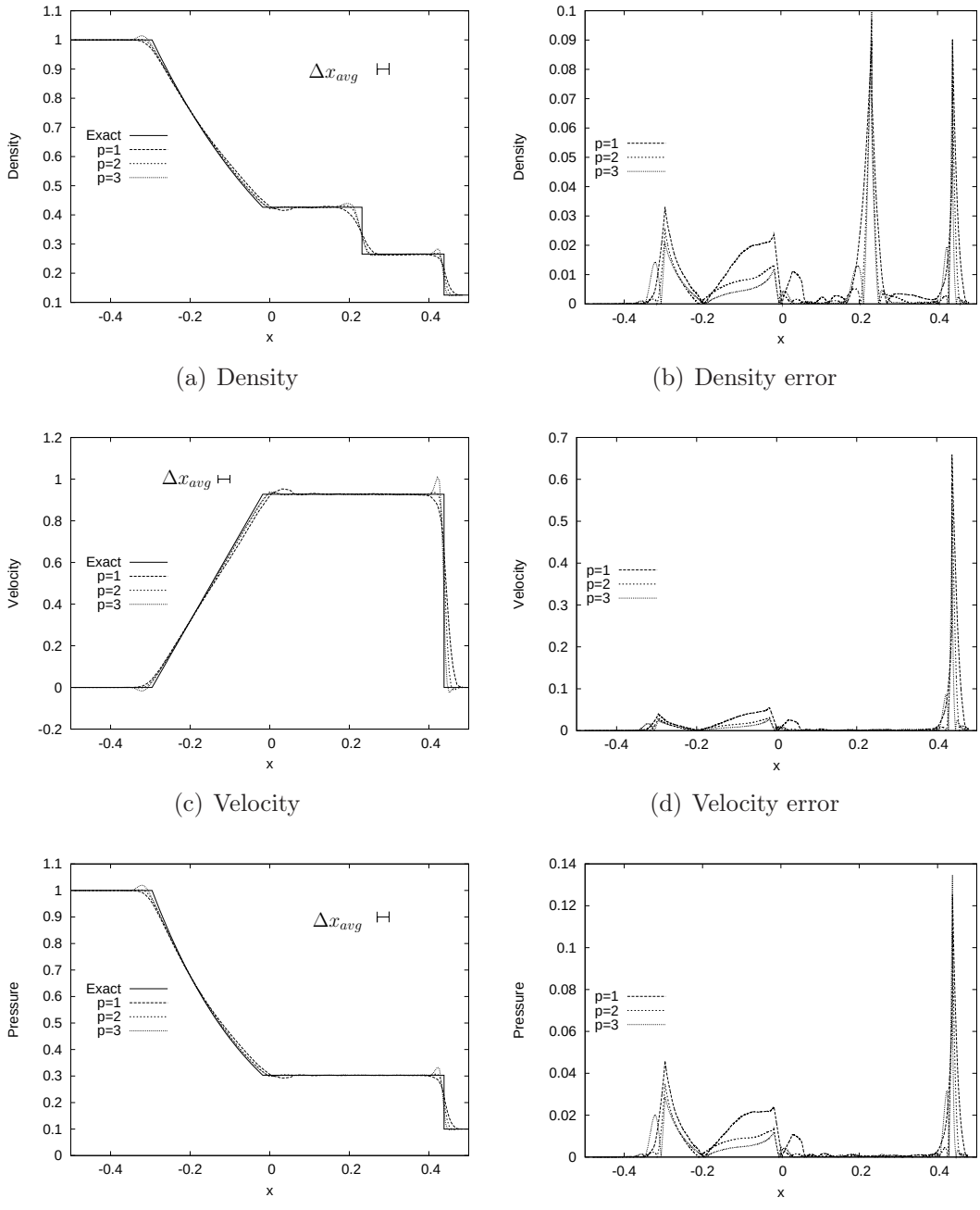


(a) $p = 2$, WWR (b) $p = 2$, GWR (c) $p = 2$, QWR

(d) $p = 3$, WWR (e) $p = 3$, GWR (f) $p = 3$, QWR

(g) $p = 4$, WWR (h) $p = 4$, GWR (i) $p = 4$, QWR

Figure 6.9: Oscillations in density ($|\nabla\rho|^2$) at time $t = 0.25$ for the modified Sod shock tube problem: note that the shocks become narrower and passes cleanly through the elements with increase in p (color scales adapted to order of scheme for visualization purpose, with maximum values of $|\nabla\rho|^2$ taken as 80, 280, 450 for $p = 2, 3, 4$ respectively).



(a) Density (b) Density error

(c) Velocity (d) Velocity error

(e) Pressure (f) Pressure error

Figure 6.10: Comparison of computed solution averaged along y axis at time $t = 0.25$ for shock tube problem. Errors computed as deviation of computed solution from exact solution (L_1 error).

the WWR-WENO scheme. In general, the schemes that suppress the oscillatory regions near the face perform better. Due to the improved reconstruction properties of the WWR-WENO schemes and its relative simplicity in implementation, we adopt the WWR-WENO scheme as the default scheme for the rest of the work.

To demonstrate the improved resolution of the discontinuities, we compare the solution of the shock tube problem with the exact analytical solution at time $t = 0.25$. The 2^{nd}, 3^{rd} and 4^{th} order solutions obtained with the WWR-WENO scheme are compared with the exact solution in Fig. 6.10. The 1D solution is extracted by averaging the 2D data at a given x location. The solution errors with respect to the exact solution are also plotted in Fig. 6.10 to illustrate the behaviour of schemes at various orders.

As expected, the 4^{th} order schemes match the exact solution better than the lower order schemes. Near the discontinuities, the solutions exhibit oscillations similar to the Gibb's phenomena[65, 66]. These oscillations, however are localized to almost a single cell width (see Fig. 6.9). For the 2^{nd} order scheme, however, the solution away from the discontinuity is also considerably affected in-spite of having a smooth solution. The deviation of the solution from the exact solution is clearly visible in the error plots in Fig. 6.10. The area under the curves decrease as the order of the scheme increases. The quantitative L_1 solution errors with respect to the exact solution are listed in Table 6.2. The higher order schemes clearly have an advantage in terms of error magnitude.

6.4.3 Shock Bubble interaction

In this test problem, a shock wave interacts with a cylindrical gas bubble of different density. The interaction produces intricate flow structures evolving with time. The problem is initialized as shown in Fig. 6.11. Reflecting wall boundary condition is specified at the top and bottom surfaces. For the initialization of the gas bubble, a Galerkin fit is applied over the elements bordering the contact discontinuity representing the bubble-fluid interface. The solution at $t = 0.2$ is plotted in Fig.

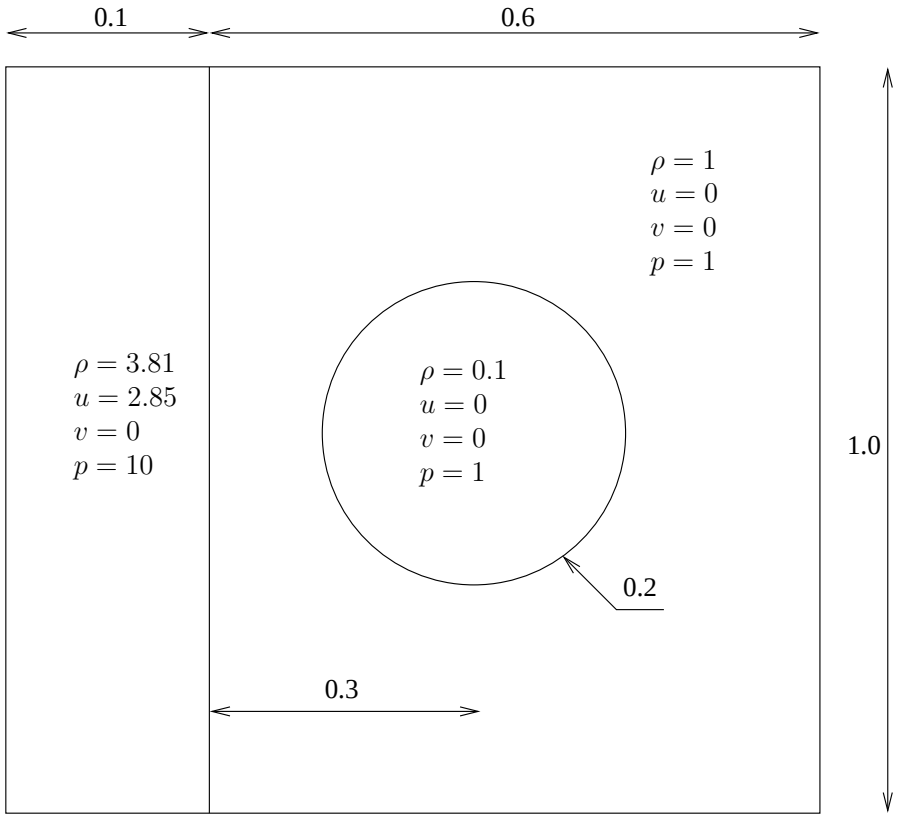
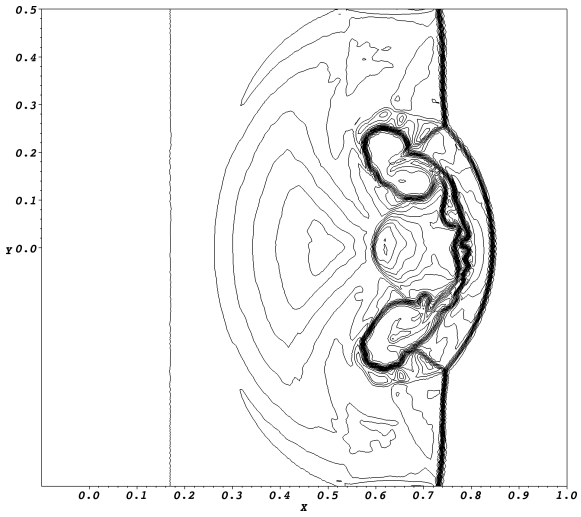


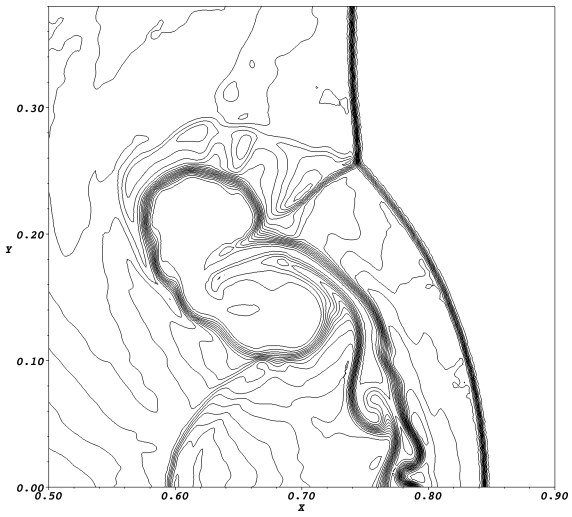
Figure 6.11: Schematic setup of shock-Bubble interaction problem.

6.12. At this time instance, the shock has fully traversed the bubble. Due to the difference in the density, the propagation of shock inside the bubble is faster compared to the outer regions, resulting in complicated flow structures behind the shock wave.

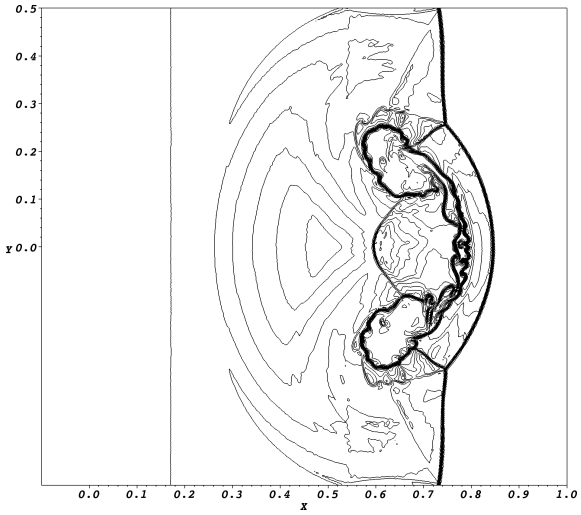
The coarse grid 3^{rd} order solutions are compared with a coarse grid 4^{th} order solution and a fine grid 3^{rd} order solution representing the p and h refinements respectively. For a grid size of $h = 1/160$, the 4^{th} order scheme reproduces finer shock structures and has a better resolution of shocks relative to the 3^{rd} order scheme. Though the 4^{th} order solution has similar shock structure as the fine grid ($h = 1/320$) 3^{rd} order solution, the latter has relatively less oscillation. For the current simulation, the total computational time of the 4^{th} order coarse grid solution is around 75% of the computational time of 3^{rd}order fine grid solution. As with the higher order schemes, the improvement in resolution with grid refinement would be higher for higher orders. Thus, with the increase in resolution of flow



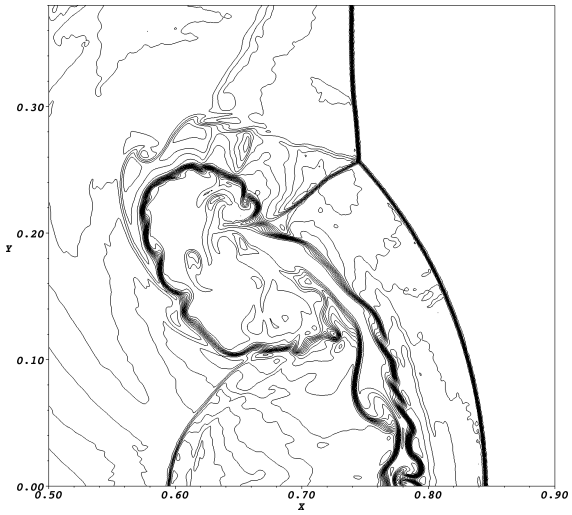
(a) O3, $h = 1/160$



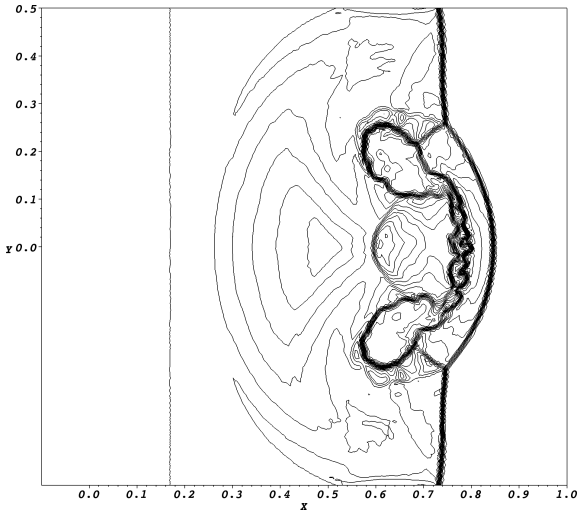
(b) O3, $h = 1/160$, Expanded view



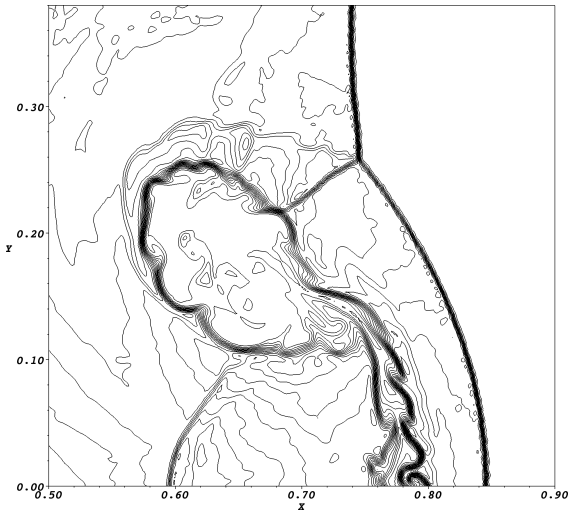
(c) O3, $h = 1/320$



(d) O3, $h = 1/320$, Expanded view

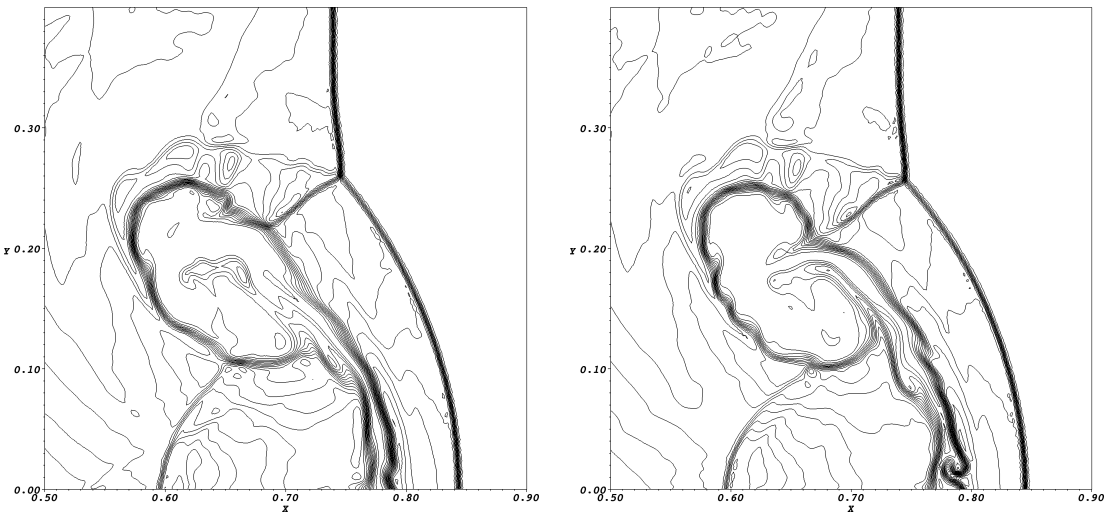


(e) O4, $h = 1/160$



(f) O4, $h = 1/160$, Expanded view

Figure 6.12: Density contours for shock-Bubble interaction problem at $t = 0.2$ computed with WWR-WENO scheme.



(a) GWR-WENO (b) QWR-WENO

Figure 6.13: Comparison of 3^{rd} order GWR and QWR WENO schemes applied to shock-bubble interaction ($t = 0.2$, $h = 1/160$).

structures, the computational cost of 4^{th} order schemes relative to the 3^{rd} order schemes would decrease.

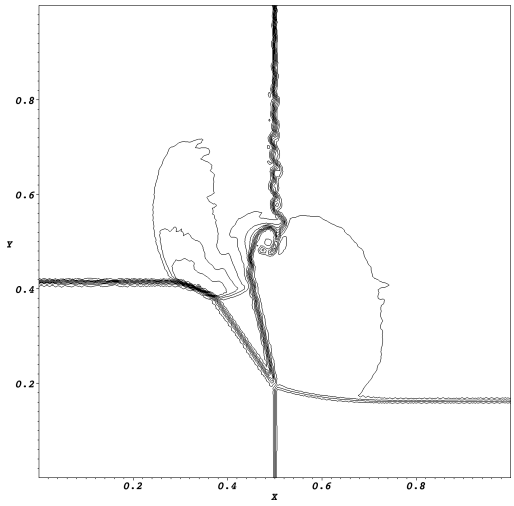
In all the three cases, the overall shock structure remains the same at $t = 0.2$. The resolution of the discontinuities in Fig. 6.12(d) is comparable to the existing solution in the literature [96] obtained on a fine grid ($h = 1/1000$) using a 3^{rd} order limiter function. For comparison purpose, the 3^{rd} order coarse grid solutions obtained using GWR and QWR WENO schemes are plotted in Fig. 6.13. The basic structure of the shock remains the same. The QWR-WENO scheme reproduces more structures compared to the other two schemes. This is expected as the QWR scheme is less diffusive. The solution obtained using GWR is comparable to the solution of WWR-WENO scheme.

6.4.4 2D Riemann Problem

Two dimensional Riemann problems are used to analyze the effectiveness of higher order WENO schemes in resolving the discontinuities and/or shocks and their resulting interactions. While many different configurations of the Riemann problem are possible, here we consider a configuration involving both shocks and contact discontinuities. The computational domain of size 1x1 is divided into four quad-

	Left	Right
Top	$\rho = 2$ $u = 0$ $v = 0.3$ $p = 1$	$\rho = 1$ $u = 0$ $v = -0.3$ $p = 1$
Bottom	$\rho = 1.0625$ $u = 0$ $v = 0.8145$ $p = 0.4$	$\rho = 0.5313$ $u = 0$ $v = 0.4276$ $p = 0.4$

Table 6.3: Initial configuration of quadrants for 2D Riemann Problem.

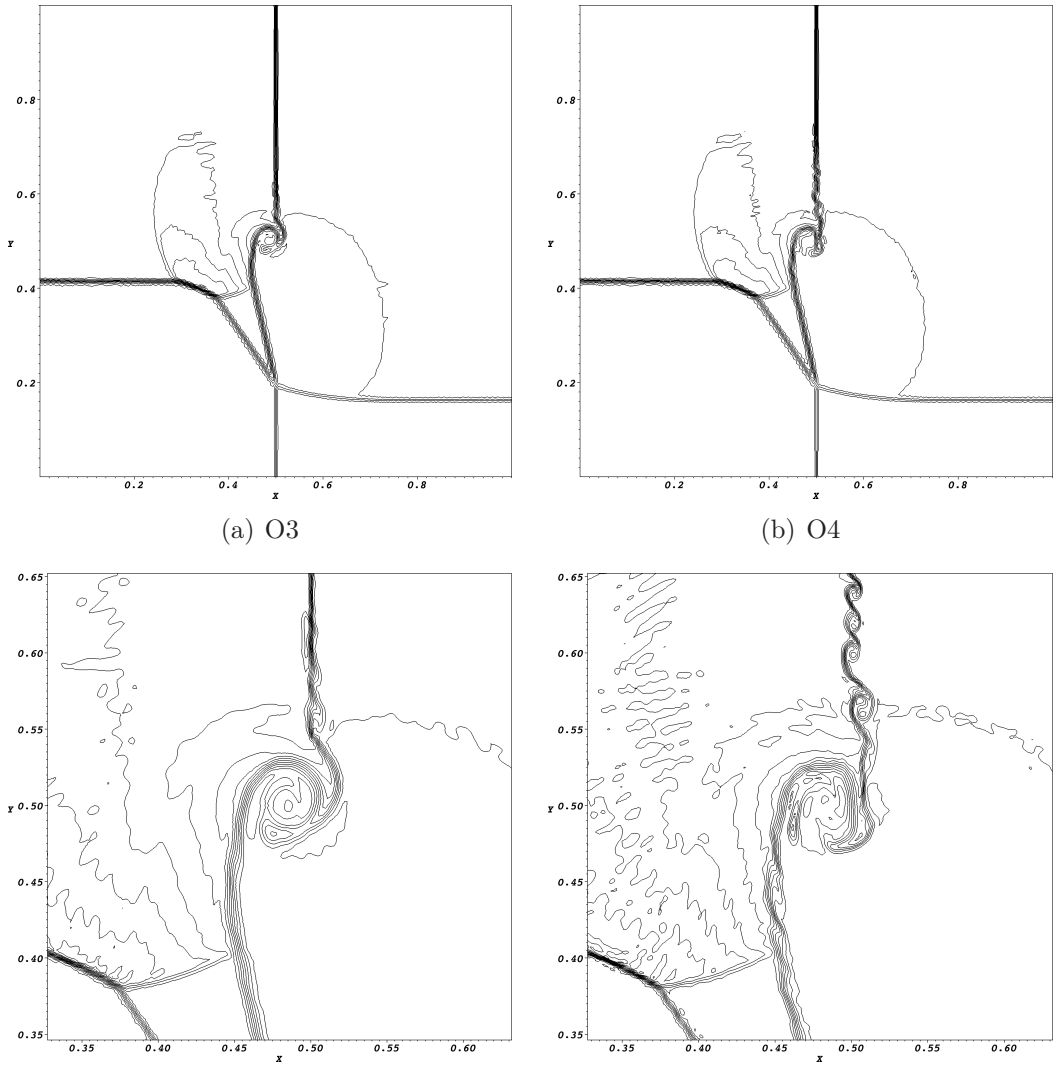


(a) LLF flux

Figure 6.14: 2D Riemann problem computed using LLF Flux (3^{rd} order solution)

rants with discontinuous data across each quadrant at time $t = 0$. The mesh is generated such that in the initial setup, all the discontinuities are confined only to the element faces. The parameters for initializing the data for all four quadrants are given in Table 6.3. The problem is initialized with two contact discontinuities (between top and bottom quadrants) and two shock waves (between left and right quadrants). A slip flow is prescribed at the contact discontinuity between the left and right quadrants. Such difference in velocities across the discontinuity can trigger an inertia-dominated Kelvin-Helmholtz (KH) like flow instabilities. In the absence of viscosity, the KH instability can result in the generation of wave instability along the surface with velocity discontinuity at all length scales.

The solution obtained using the (Local Lax Friedrich) LLF flux is plotted in Fig. 6.14. All the other simulations (Figures 6.15 and 6.16) use HLLC-LLF flux.

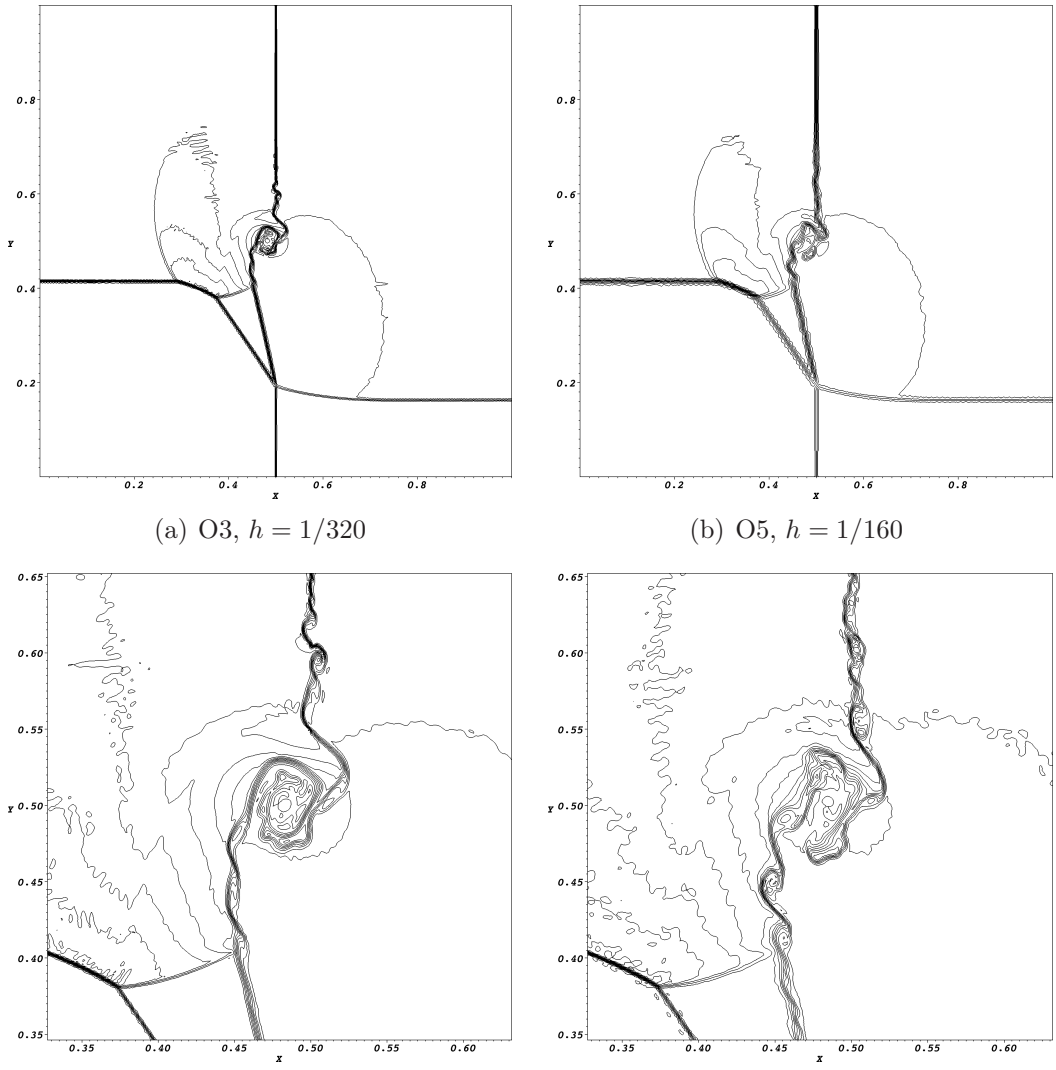


(a) O3 (b) O4

(c) O3, Expanded view (d) O4, Expanded view

Figure 6.15: 2D Riemann Problem, $t = 0.3$, $h = 1/160$ (20 equally spaced contours of density).

The LLF flux can not resolve the contact discontinuities and perturb the interface representing the contact discontinuities. With the HLLC-LLF flux, the scheme switches to HLLC flux across a contact discontinuity, thus enabling the scheme to capture the discontinuity accurately. A comparison of Fig. 6.14 with Fig. 6.15(a) shows that the LLF scheme tends to artificially smear the contact discontinuities as it does not distinguish the contact discontinuity from a shock. A contact discontinuity with a slip-stream condition (varying velocity across discontinuity) is very sensitive to any fluctuations present in the vicinity and often results in Kelvin Helmholtz kind of instability. The increased instability characteristics at the con-



(a) O3, $h = 1/320$

(b) O5, $h = 1/160$

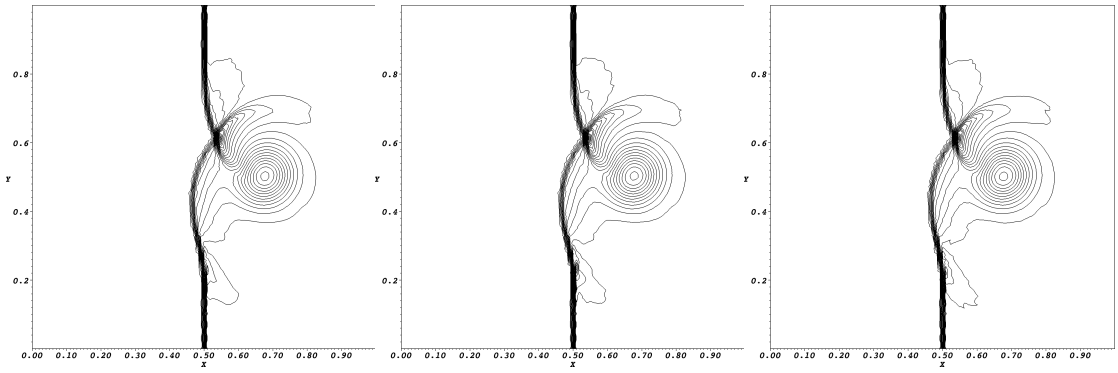
(c) O3, $h = 1/320$, Expanded view

(d) O5, $h = 1/160$, Expanded view

Figure 6.16: 2D Riemann Problem, High Resolution Computations, $t = 0.3$ (20 equally spaced contours of density).

tact discontinuity in Fig. 6.14 could be caused by the inability of the LLF scheme to capture the contact discontinuity.

Detailed computations are performed for 3^{rd}, 4^{th} and 5^{th} order schemes with additional 3^{rd} order computation on a refined grid (h refinement). The results are plotted in Figures 6.15 and 6.16. It is observed that the 3^{rd} order solution reproduces a smooth contour and the overall solution agrees well with the literature. The contact discontinuity is well resolved with the HLLC-LLF fluxes. In the 3^{rd} order solution, the instability at the contact discontinuities are just starting to appear above the vortex structure. The 4^{th} order scheme performs better than the



(a) O3 (b) O4 (c) O5

Figure 6.17: Solution of the shock-vortex interaction problem (30 equally spaced contours of density).

3^{rd} order scheme in terms of resolution of discontinuities and finer shock structures. Due to the increased resolution of the 4^{th} order scheme, the finer structures at the contact discontinuity are clearly visible at the region above the vortex like structure.

Both the coarse ($h = 1/160$, Fig. 6.15(a)) and fine grid ($h = 1/320$, Fig. 6.16(a)) 3^{rd} order solution have a kink in the contour in top left quadrant. The appearance of this kink in the contour could be due the initialization of the shock discontinuity at $t = 0$. However, this feature is less visible in the 4^{th} and 5^{th} order solutions. The fine grid 3^{rd} order solution contains small structures along the contact discontinuities which were not observed in the coarse grid ($h = 1/160$) solution. The flow/vortex structures in a 5^{th} order coarse grid ($h = 1/160$) solution in Fig. 6.16 are comparable to those present in 3^{rd} order fine grid ($h = 1/320$) solution. The 5^{th} order coarse grid solution has a similar shock resolution as that of 3^{rd} order fine grid solution, thus demonstrating the improvement in the sub-cell resolution of the discontinuities with increase in order of scheme.

6.4.5 Shock Vortex interaction

This is a standard test problem to test the interaction of a vortex with a shock wave. The left half of the computational domain ($x < 0.5$) is initialized with the solution $(1, 1.1\sqrt{\gamma}, 0, 1)$. A stationary shock wave of mach number 1.1 is setup at $x = 0.5$. The solution state at the right side of the shock ($x > 0.5$) can be evaluated

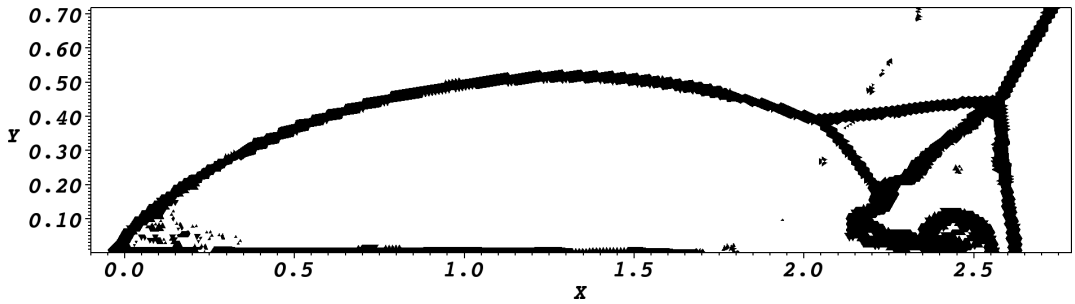


Figure 6.18: Identification of troubled cells using shock detectors ($h = 1/100$, $t = 0.2$, 3^{rd} order).

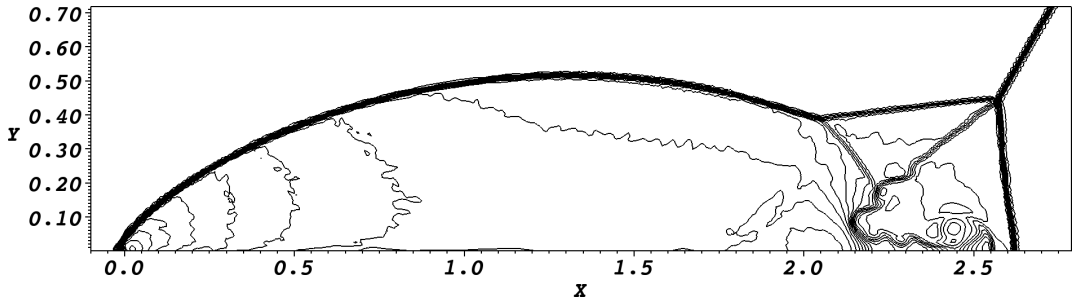
from the solution of the Rankine-Hugoniot jump condition. An isentropic vortex is setup at $(0.25, 0.5)$ as a perturbation over the free stream quantities. These perturbations are given by

$$\begin{aligned} \delta\rho &= \left(1 - \tfrac{\gamma-1}{2}\beta^2 e^{2\theta(1-r_v^2)}\right)^{\frac{1}{\gamma-1}} \\ \delta u &= \tfrac{y}{r}\beta r_v e^{\theta(1-r_v^2)} \\ \delta v &= -\tfrac{x}{r}\beta r_v e^{\theta(1-r_v^2)} \\ \delta \mathbf{p} &= \left(1 - \tfrac{\gamma-1}{2}\beta^2 e^{2\theta(1-r_v^2)}\right)^{\frac{\gamma}{\gamma-1}} \end{aligned} \tag{6.28}$$

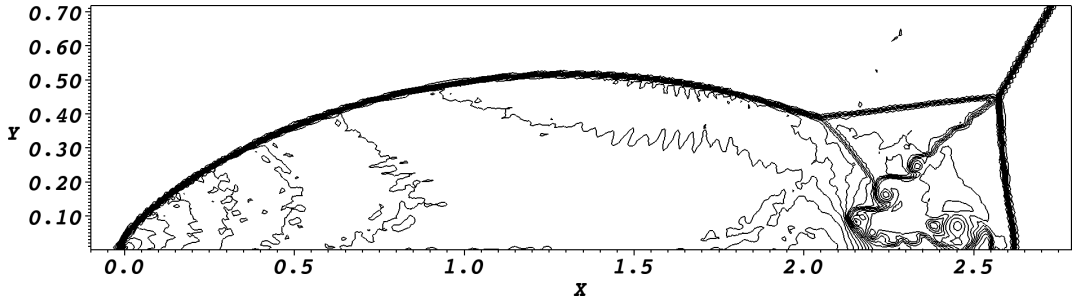
where $r = \sqrt{x^2 + y^2}$ and $r_v = \frac{r}{0.05}$. The constants β and θ are set to 0.3 and 0.204 respectively. The average element size h is set to $1/100$ while generating the mesh. The solution is evolved till a time $t = 0.35$ during which, the vortex has passed through the stationary shock. The solution obtained using 3^{rd}, 4^{th} and 5^{th} order schemes are shown in Fig. 6.17. All the schemes produce non-oscillatory solution and the results are comparable with the existing results in literature [1].

6.4.6 Double Mach Reflection

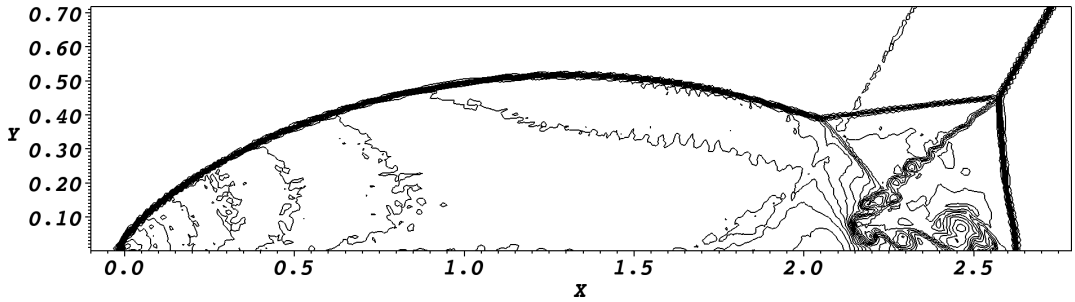
The double Mach reflection of a strong shock wave is a commonly used to evaluate the performance of shock capturing schemes in the presence of strong shocks. This problem is of special interest as it involves both steady and unsteady flow structures. The problem is setup with a Mach 10 shock contacting the wall surface at $x = 0$, making an angle of 60^0 with the wall. The reflecting wall is modeled along the x-axis for $x \geq 0$. For the boundary $x < 0$, exact post-shock condition is applied.



(a) O3

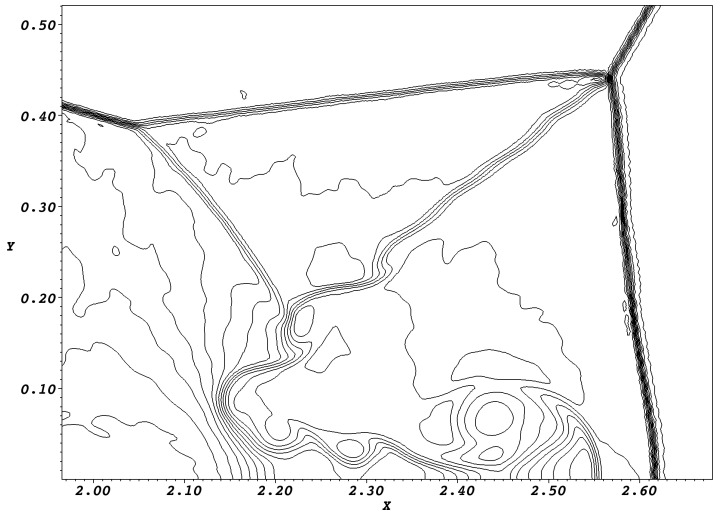


(b) O4

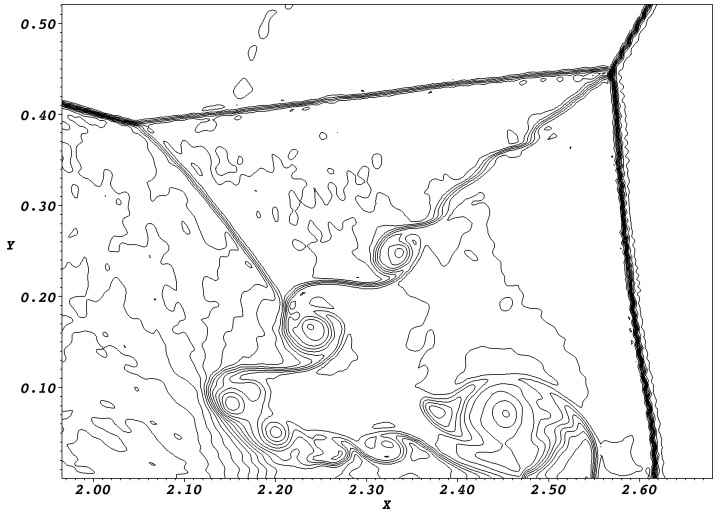


(c) O5

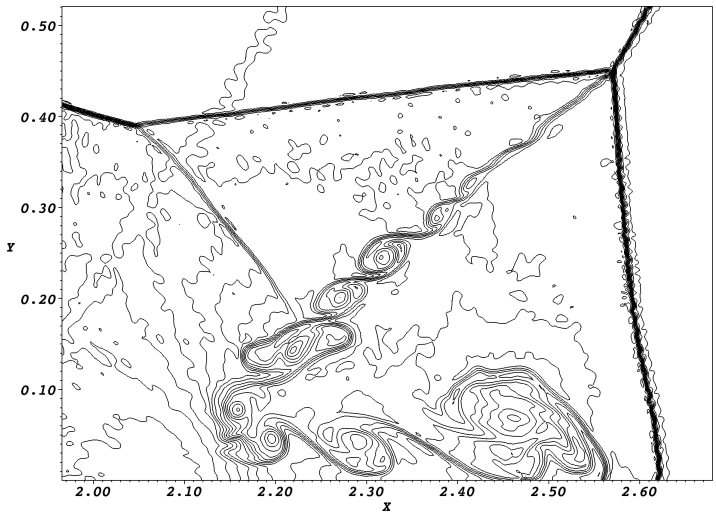
Figure 6.19: Double Mach Reflection of Mach 10 shock: Grid size: $h = 1/100$, $t = 0.2$ (30 equally spaced contours of density).



(a) O3

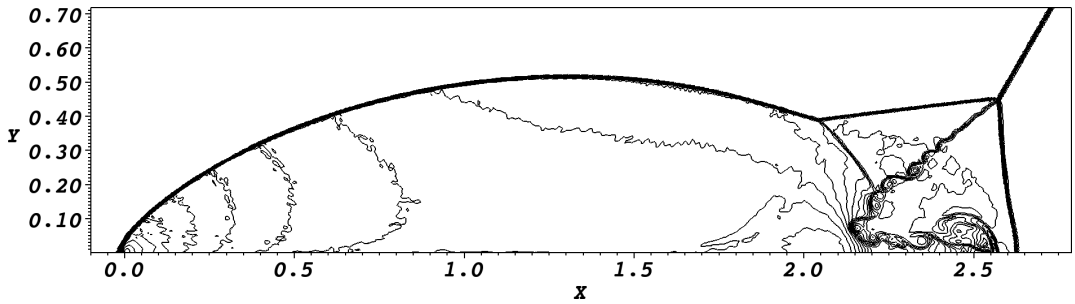


(b) O4

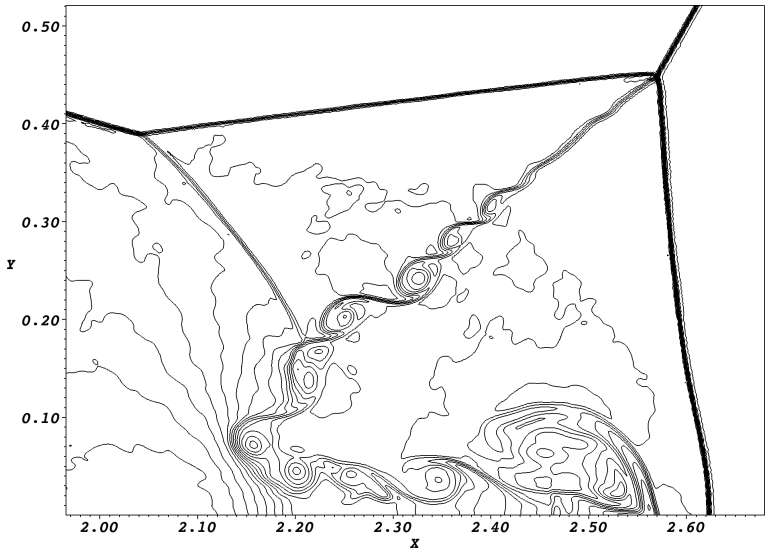


(c) O5

Figure 6.20: Mach stems in Double Mach Reflection of Mach 10 shock: Grid size: $h = 1/100$, $t = 0.2$ (30 equally spaced contours of density).



(a) Normal Contour



(b) Expanded view

Figure 6.21: Double Mach reflection using 3^{rd} order WENO: $h = 1/200$, $t = 0.2$ (30 equally spaced contours of density).

The simulation is carried out till time $t = 0.2$. Computations are performed in grid of size $h = 1/100$ for different orders of scheme. A fine-grid computation with 3^{rd} order scheme is carried out on a grid of size $h = 1/200$ to compare the results of higher order schemes. The regions of oscillatory solutions are effectively identified using oscillation indicators as detailed in Section 6.3.6. The troubled cells that require WENO reconstruction for a 3^{rd} order WENO scheme applied on a mesh of size $h = 1/100$ is shown Fig. 6.18. This case corresponds to WENO reconstruction applied to 10.9% of the total number of elements in the computational domain. The ratio of number of troubled cells to the total number of elements in the mesh reduces with grid refinement. Fig. 6.19 shows the computed solution for the double Mach reflection obtained on a grid size of $h = 1/100$. In all the cases, the overall structure of the flow field agrees well with the literature [97, 1].

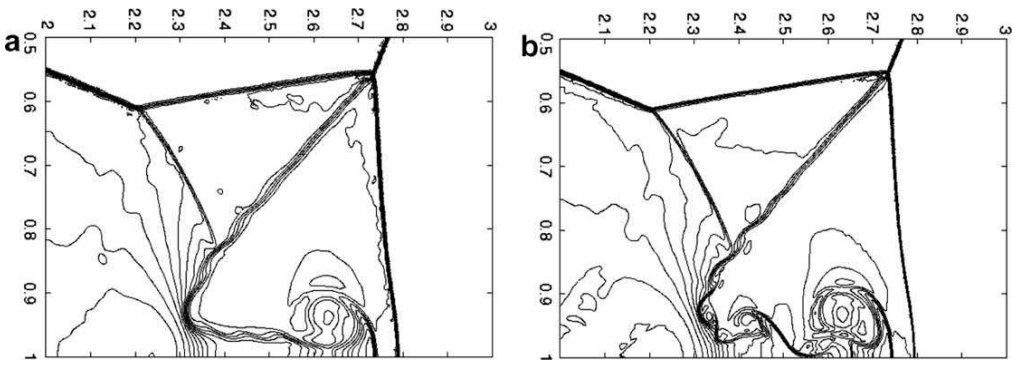


Figure 6.22: Solution of double mach reflection obtained using 3^{rd} order hierarchical reconstruction a:$h = 1/250$ b: $h = 1/500$ (image obtained from Fig. 15 in [1]).

As the order of the scheme increases, the spurious oscillations generated at the shock becomes more visible. However, the finer structures arising due to the shock interactions are captured effectively with the higher order schemes. Similar to the 2D Riemann problem in Sec 6.4.4, a KH like instability is observed across the slip-stream at the triple shock region (see Sec. 7.1.2 in [98] for detailed description of Double Mach reflection problem). A WENO scheme with larger diffusion would diffuse these smaller structures with length scales smaller than the element size. Similarly, the detailed structures of the high density jet at the end of the slip-stream is also well represented by higher order schemes. This is similar to the trend observed in the previous computations where the higher order methods result in more accurate representation of flow field in the vicinity of discontinuities.

The results of fine grid simulation are shown in Fig. 6.21. The flow structures at the contact discontinuity and the high density jet can be seen clearly. The higher order solutions in Fig. 6.19 obtained with grid size 1/100 and the 3^{rd} order solutions obtained with grid size 1/200 are comparable to the solutions obtained using hierarchical reconstruction method presented in [1]. For the purpose of comparison, we have also reproduced the plot of the solutions obtained using hierarchical reconstruction [1] method in Fig. 6.22. The coarse grid 3^{rd} order solution can reproduce the major flow structures such as the shocks, contact discontinuities and the high density jet at the end of contact discontinuity. However, the length scales of the structures at the slip-stream contact discontinuity are larger compared to

the fine grid solution. Similar to previous cases, as the order of scheme increases, the resolution of the shock within the cell increase. This is evident from the high resolution contours at the double shock region plotted in Fig. 6.23. The resolution of the shock and contact discontinuity in a 5^{th} order solution is comparable to that of the 3^{rd} order fine grid solution computed with half the grid size. Comparing the solution oscillations for different grids in Fig. 6.23, it can be derived that the solution oscillations near the shock is of the order of cell size. This is further confirmed with the fine grid solution in Fig. 6.23(d), where the length scale of the solution oscillation is nearly half of the length scale of oscillations in a coarse grid solution. An interesting feature of the current method is its ability to represent a discontinuity across the cell interfaces resulting in a nearly grid-independent resolution of discontinuities. For all the computations, the overall structure of the shock remains unaffected by element size.

Though the default value of $\ell_n = \frac{1}{2}$ produces results with adequate resolution, here we also compute a solution with $\ell_n = \frac{1}{3}$ to illustrate its effect on solution resolution. Depending on the physical problem simulated, this parameter can be modified, if required, to produce a high resolution solution. A 3^{rd} order solution is computed with $\ell_n = \frac{1}{3}$ while retaining all the other parameters same. Due to the reduction of stencil size, the resulting solution (Fig. 6.24) is more oscillatory than the original 3^{rd} order solution. However, the resolution of the flow structures is enhanced compared to the earlier computation (Fig. 6.20(a)). A similar effect was reported by Xu et. al. [1] when partial neighboring cells are considered for WENO reconstruction. A lower value of ℓ_n would however make the schemes unstable, especially for higher orders. Hence, it is advised to use the default value of $\ell_n = \frac{1}{2}$ for the proposed WENO reconstruction.

6.4.7 3D Spherical Explosion

Similar to the shock-tube problem in Sec. 6.4.2 the 3D explosion problem can be used to analyze the resolution of discontinuities by higher order WENO reconstruc-

tion in three dimensions. The problem is initialized with the following solution:

$$(\rho, \mathbf{p}) = \begin{cases} (1,1), & r \leq 0.4 \\ (0.125, 0.1), & r > 0.4 \end{cases}, \; u = v = w = 0 \tag{6.29}$$

where $r = \sqrt{x^2 + y^2 + z^2}$. The computational domain is taken to be a box defined by two corners (0,0,0) and (1,1,1). The corresponding mesh ($h = 1/10$) is plotted in Fig. 6.25. Symmetry boundary condition is used at all the boundaries of the computational domain.

The reference solution is computed by reducing the axi-symmetric problem to a 1D hyperbolic equation given by

$$\frac{\partial Q}{\partial t} + \nabla \cdot \vec{F} = S_{Sph} \tag{6.30}$$

where Q and F are given in Eqn. (2.7) and the source term is given by

$$S_{Sph} = -\frac{2}{r} \begin{pmatrix} \rho u \\ \rho u^2 \\ (\rho e + \mathbf{p}) u \end{pmatrix} \tag{6.31}$$

The reference solution is computed by solving Eqn. (6.30) on a a very fine mesh with the LLF Riemann flux. Three different computations are performed for orders 2 to 4. The solution along the diagonal of the computational domain is used to compare the accuracy of the scheme with respect to the exact solution. The computed solutions of orders 2-4 and the reference 1D solution are plotted in Fig. 6.26. All the schemes can capture the shock and contact discontinuity. The resolution of discontinuities is improved with increase in the order of scheme. The fine grid solution ($h = 1/50$) obtained in [2] using the FV-WENO scheme is reproduced in Fig. 6.27 for reference. For the given discretization, the resolution of the discontinuities using the current RKDG-WENO scheme is better than the existing 3D solutions based on FV WENO schemes[2].

6.4.8 Shock interaction with 3D bubble

This test case is similar to the 2D test case, except that the bubble is now defined as a 3D sphere instead of a 2D cylinder. The computational domain is defined by a box with a width of 1.8 length units, depth of 1 length unit and height of 1 length unit. The solution is setup with initial conditions similar to the 2D shock-bubble problem as defined in Fig. 6.11. The mesh is formed with an approximate element size of $h = 1/50$ and consist of 1.35 million tetrahedral cells. The evolution of the shock bubble interaction computed using a 3^{rd} order WWR-WENO scheme is plotted in Fig. 6.28 in terms of iso-surfaces of the density. The oscillation free nature of the solution is clearly demonstrated in this simulation with the flow features in the post-shock solution (Fig. 6.28(c)-6.28(e)) being captured without any significant oscillation in solution. This demonstrates the capability of the proposed scheme in three dimensions.

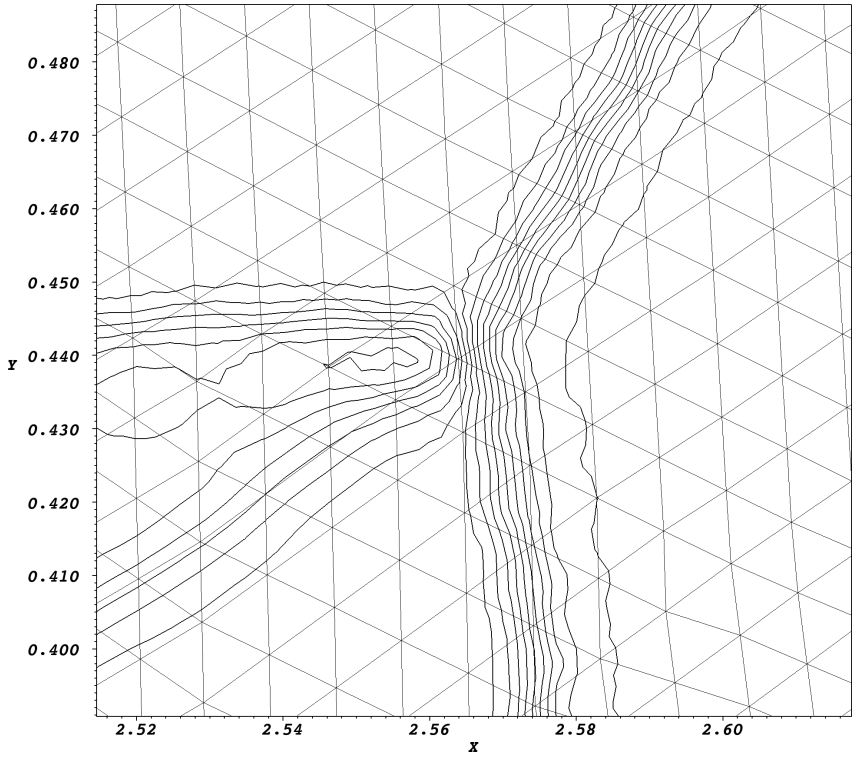
6.5 Summary

A compact, high resolution WENO reconstruction method is proposed for unstructured grids. Various improvements in numerical flux computation, solution reconstruction, oscillation detection and adaptive weights of stencils are proposed to enable the extension of the scheme to very high orders. The improvement in resolution of the higher order schemes are realized with the application of suitable filters to selectively remove the oscillatory solutions from the reconstruction process. The spatial resolution of the reconstruction process is preserved by using a compact stencil comprising of only the immediate element neighbors. With these formulation, for the first time an unstructured, high resolution, compact WENO scheme has been extended to 5^{th} order accuracy and to three dimensions.

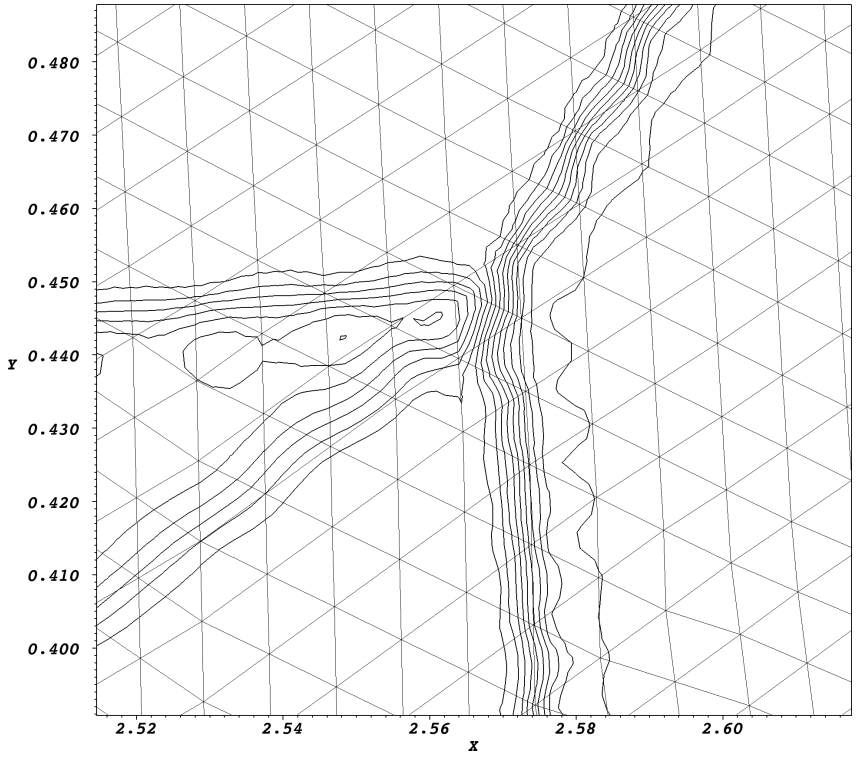
Computational efficiency of the proposed scheme is improved in two ways. Firstly, the stencils are constructed on a reference plane, thus avoiding computation and storage of the huge coefficient matrices for stencil reconstruction. Secondly,

the expensive WENO reconstruction is performed only in the cells identified by the oscillation detector χ described in Section 6.3.6, leading to greater computational efficiency. The difference for solution with those in which reconstruction is performed on all elements is negligible.

The performance of the WENO schemes are tested on the inviscid Euler equations for various benchmark problems. Since no artificial dissipation is used, the 4^{th} and 5^{th} order schemes exhibit slight oscillatory behavior compared to 3^{rd} order schemes. Although the higher order schemes ($> 3^{rd}$) seem to be more oscillatory than the lower order schemes, it is evident from the numerical experiments that they have high sub-cell resolution of discontinuities and their interactions. The h and p refinement studies conducted in this work confirm that the proposed reconstruction method can reproduce a highly resolved smooth solution within the element even in the presence of discontinuities. It has been demonstrated with numerical experiments that the solutions obtained with the current WENO reconstruction on a coarse grid is comparable to the fine grid solutions available in the literature. The proposed WENO reconstruction method has also been extended to three dimensions and tested for problems involving three dimensional shocks and contact discontinuities.

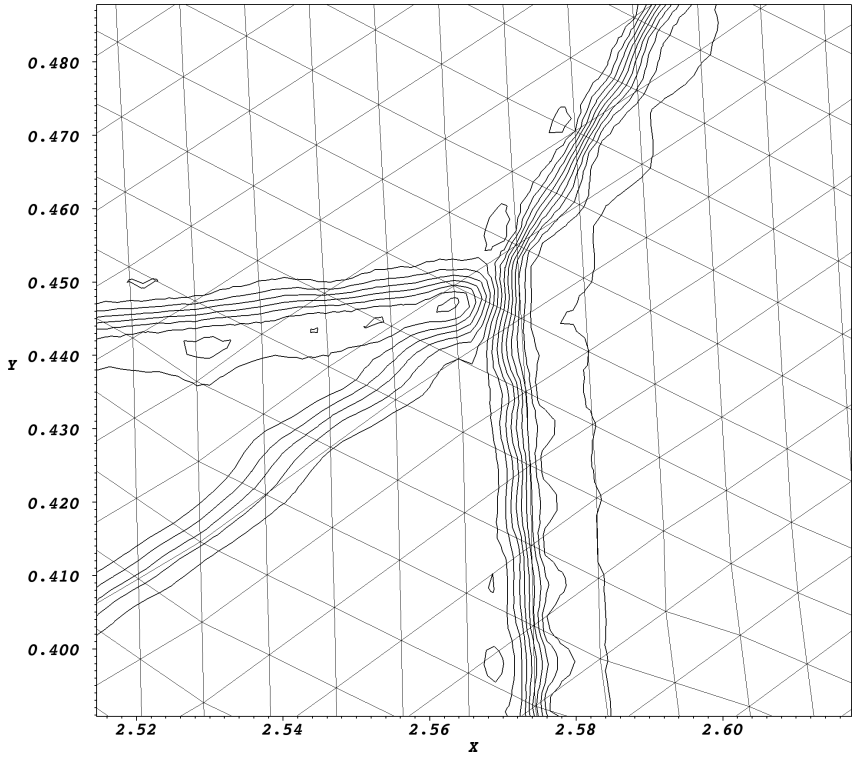


(a) O3, $h = 1/100$

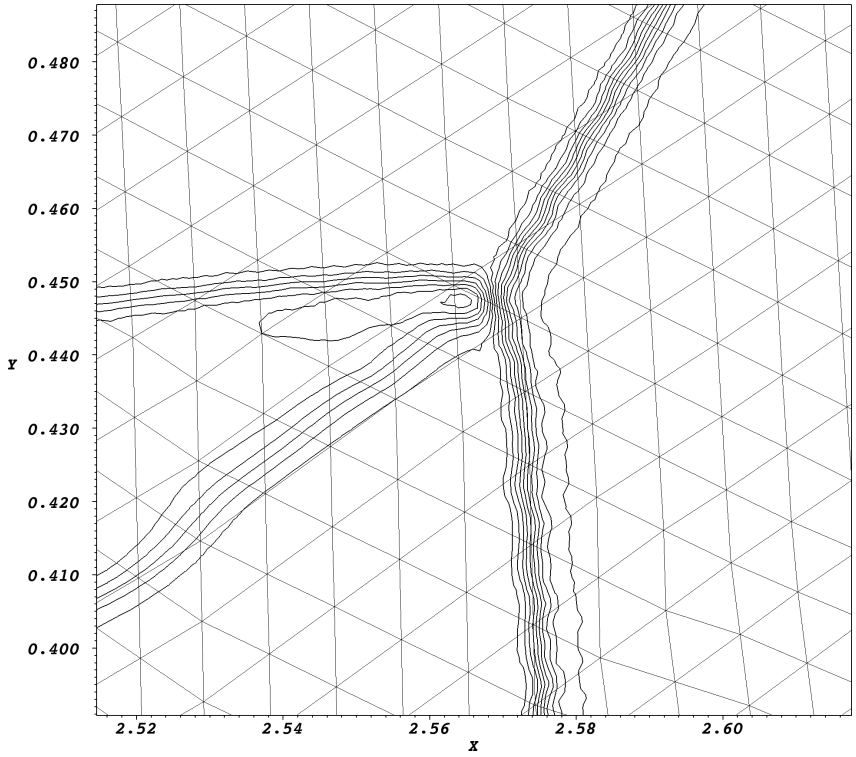


(b) O4, $h = 1/100$

Figure 6.23: Sub-element resolution at double-shock region for Double Mach Reflection (background mesh corresponding to $h = 1/100$).

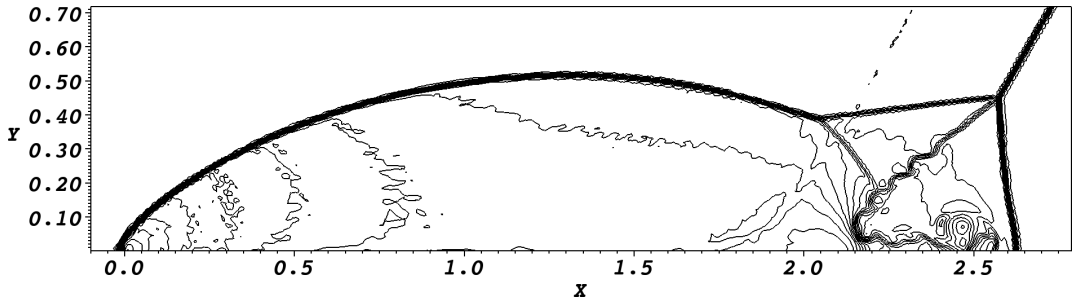


(c) O5, $h = 1/100$

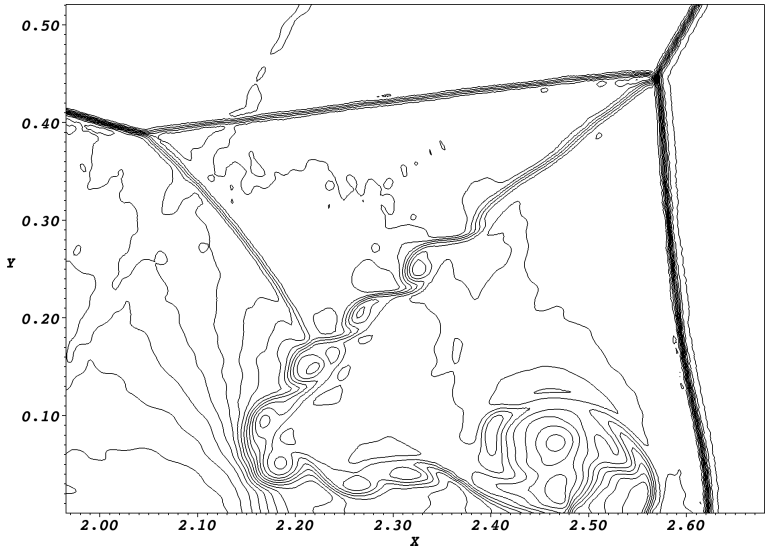


(d) O3, $h = 1/200$

Figure 6.23: Sub-element resolution at double-shock region for Double Mach Reflection (background mesh corresponding to $h = 1/100$) *contd.*



(a) Normal Contour



(b) High resolution contour

Figure 6.24: Double Mach Reflection: 3^{rd} order solution with $h = 1/100$ and $\ell_n = \frac{1}{3}$ (30 equally spaced contours of density).

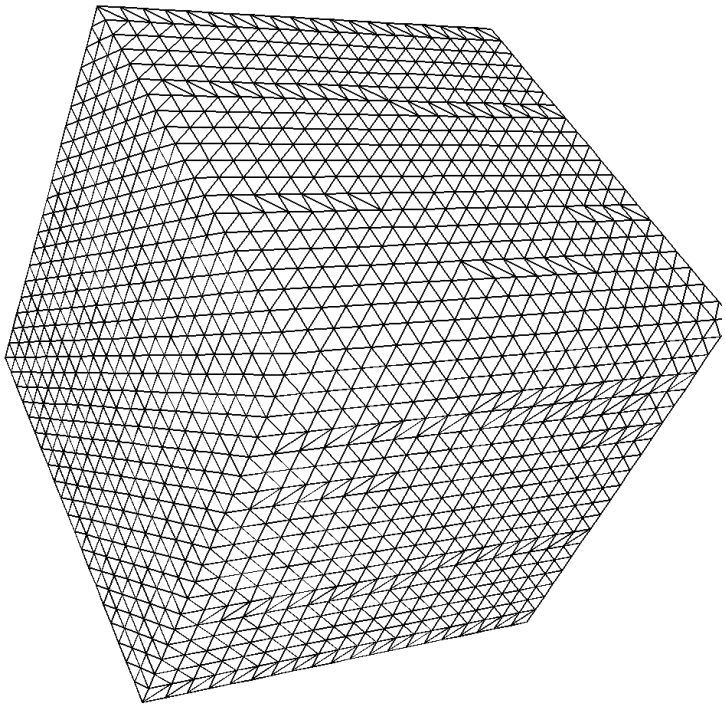


Figure 6.25: Mesh used for 3D explosion problem.

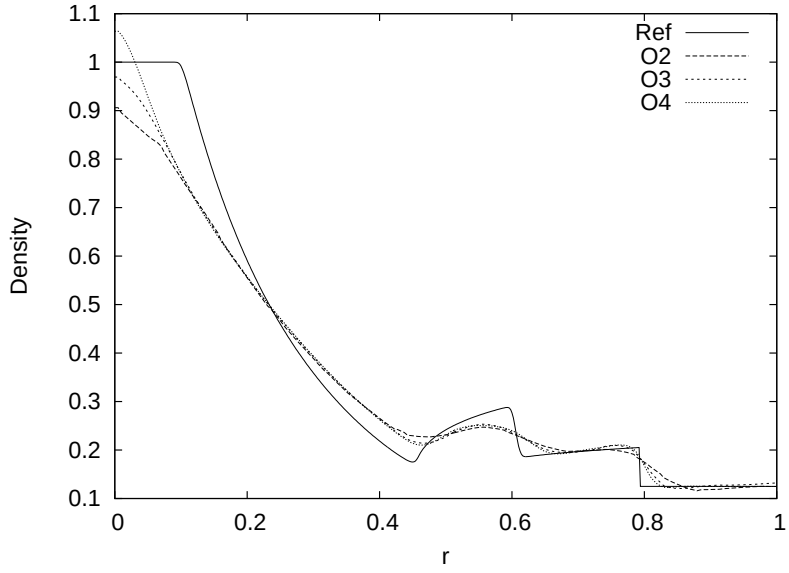


Figure 6.26: Plot of density for a Spherical explosion problem using 2^{nd}, 3^{rd} and 4^{th} order WENO schemes ($h = 1/20$).

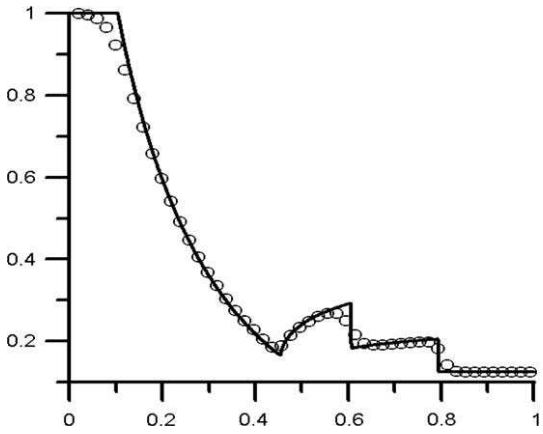
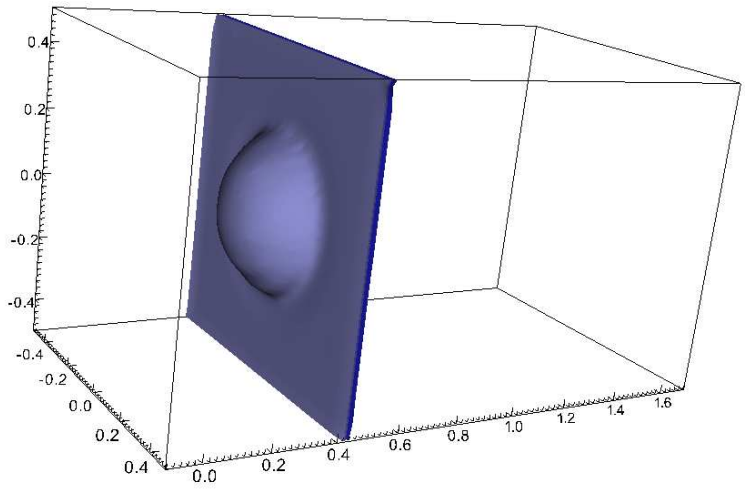
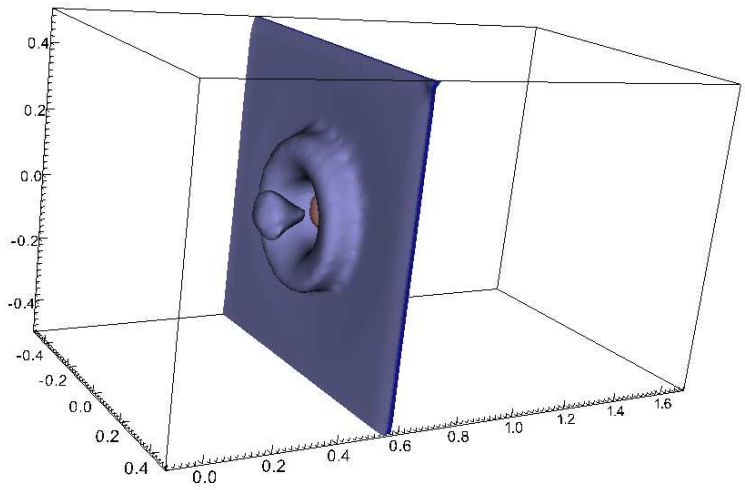


Figure 6.27: Plot of density for a Spherical explosion problem using FV-WENO schemes ($h = 1/50$) (image reproduced from Fig.11 in [2]).

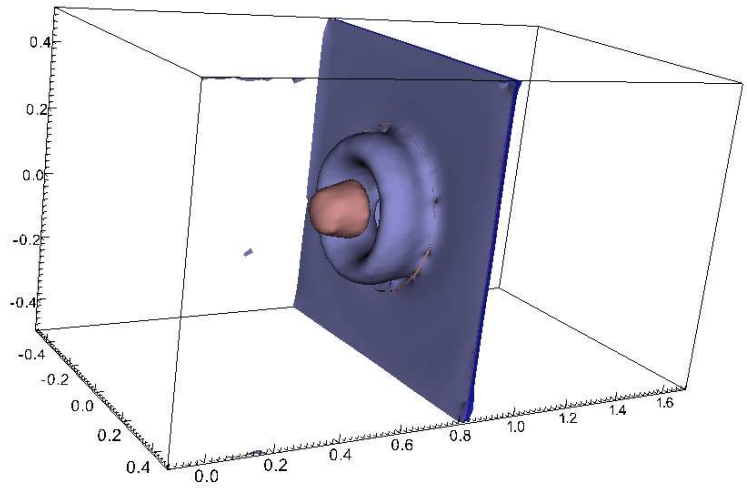


(a) $t = 0.117$

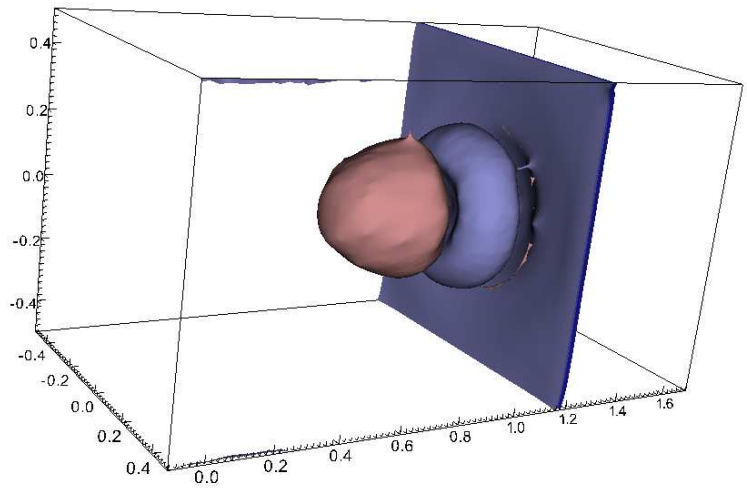


(b) $t = 0.153$

Figure 6.28: Iso-surfaces of density for a 3^{rd} order solution of a 3D shock bubble interaction.

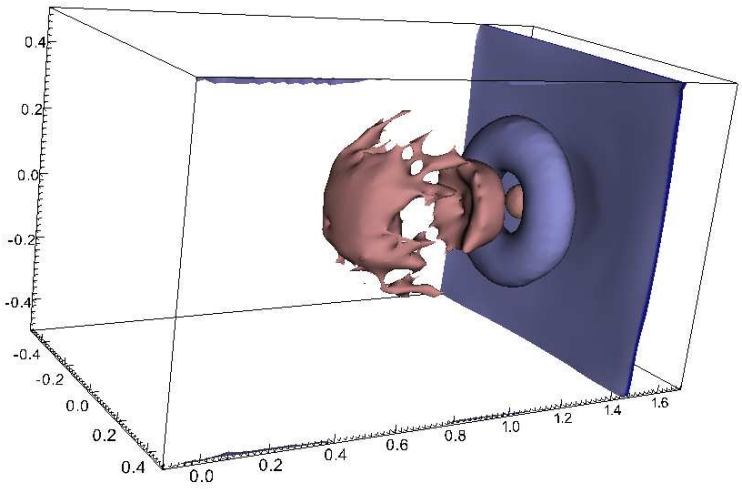


(c) $t = 0.225$



(d) $t = 0.0.320$

Figure 6.28: Iso-surfaces of density for a 3^{rd} order solution of a 3D shock bubble interaction. *(cont.)*



(e) $t = 0.4$

Figure 6.28: Iso-surfaces of density for a 3^{rd} order solution of a 3D shock bubble interaction. *(cont.)*

Chapter 7

Applications to direct computation of sound

The primary aim of this chapter is to apply the previously developed methods to perform coarse DNS simulations of problems involving aero-acoustic sound generation. For the computed solutions, the mechanism of sound generation is analyzed in detail. The aerodynamic sound generated as a result of the flow are recorded at specified locations and a Fast Fourier Transform (FFT) is performed on the acoustic signal to extract the harmonics. Two problems are chosen for the current test: (i) cavity tones generated in a flow over a rectangular cavity (ii) aerodynamic sound generated in a reed-like instrument. In all the cases, absorbing boundary condition (see Sec. 4.4) is used to prevent the spurious reflection of acoustic waves at the computational boundary.

7.1 Cavity tones generated in a open cavity

The cavity tone problems been a topic of active research for the last several decades. The applications of the cavity tone problem are widespread in the aerospace and automobile sectors. The landing gears and the weapons bay of the aircraft, the gaps in doors in the cars, windows of a high rise building etc. are all examples of flow over an open cavity. In all these cases, a periodic pressure fluctuation is

observed in the cavity region. Krishnamurty et. al. [99] did a series of experiments on 2D cavity oscillations. He observed that these oscillations occurred not just in a specific flow regime, but, in a wide range of flows ranging from subsonic to transonic flows. In additional to the pressure fluctuations in the cavity, it was also observed that specific acoustic tones were generated characterized by the geometry of the cavity and the nature of fluid flow. Later, Rossieter et. al. [100] proposed a model to explain the generation of acoustic tones. He proposed that the acoustic tones generated in the cavity propagate upstream to influence the vortex generation in the shear layer. He suggested an empirical formula relating the Strouhal number to the flow property as:

$$St = \frac{m - \vartheta}{M + 1/K_c}, \;\; m = 1, 2, 3, ... \tag{7.1}$$

where m is an integer representing the acoustic modes, M is the free-stream Mach number, ϑ is the empirical phase number between the vortices and pressure waves, K_c is the ratio of convection speed of the vortices to the free-stream velocity and St is the Strouhal number defined as

$$St = \frac{fL}{U_\infty} \tag{7.2}$$

where f is the resonant frequency, L is the length of the cavity and U_∞ is the free stream velocity. The values of ϑ and K_c are taken to be 0.25 and 0.57 for a range of Mach number from 0.4 to 1.2.

In this work, we aim at simulating the flow around the rectangular cavity with length to depth ratio (L/D) of 2. The generated acoustic tones are extracted in the near-field region at a point located at (0, 8D) from the trailing edge of cavity. A Fast Fourier Transform (FFT) is then performed on the pressure signal to determine the corresponding acoustic modes. The problem setup is shown in Fig. 7.1. The far-field boundary is located at a distance of 20 units away from the center of the cavity. Absorbing boundary condition is applied at all the far-field boundaries to prevent

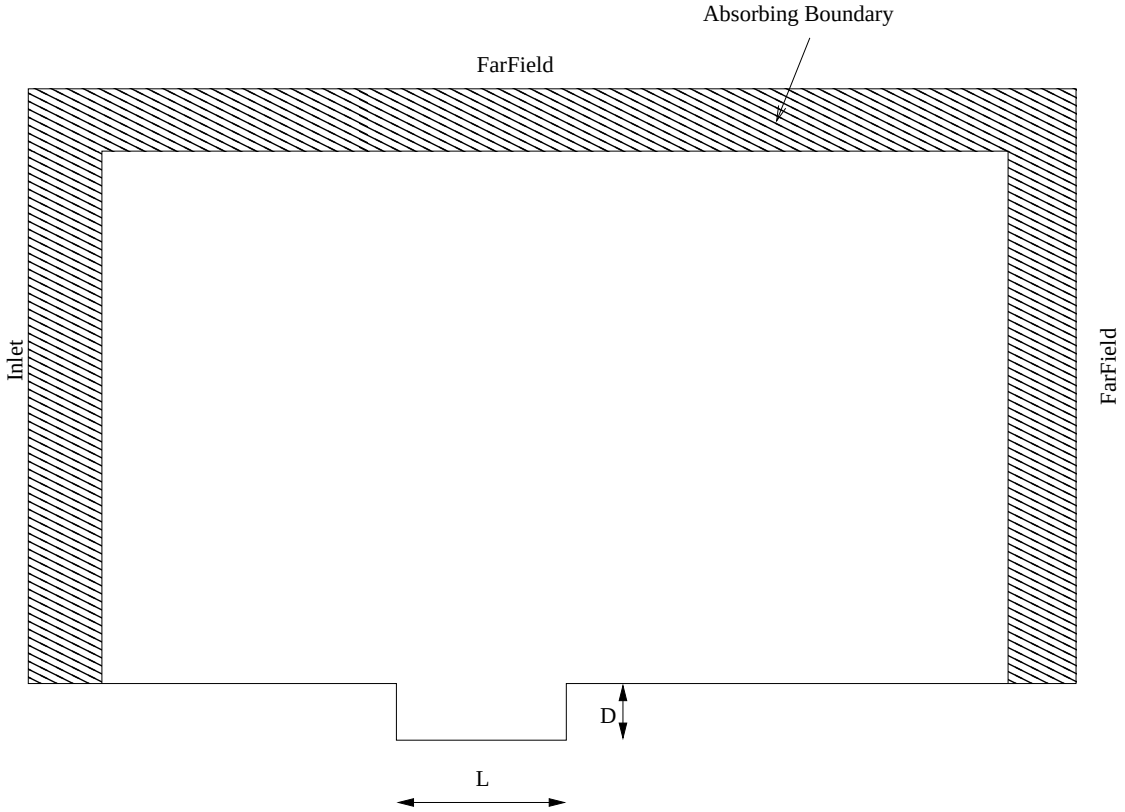


Figure 7.1: Problem setup for flow over open cavity.

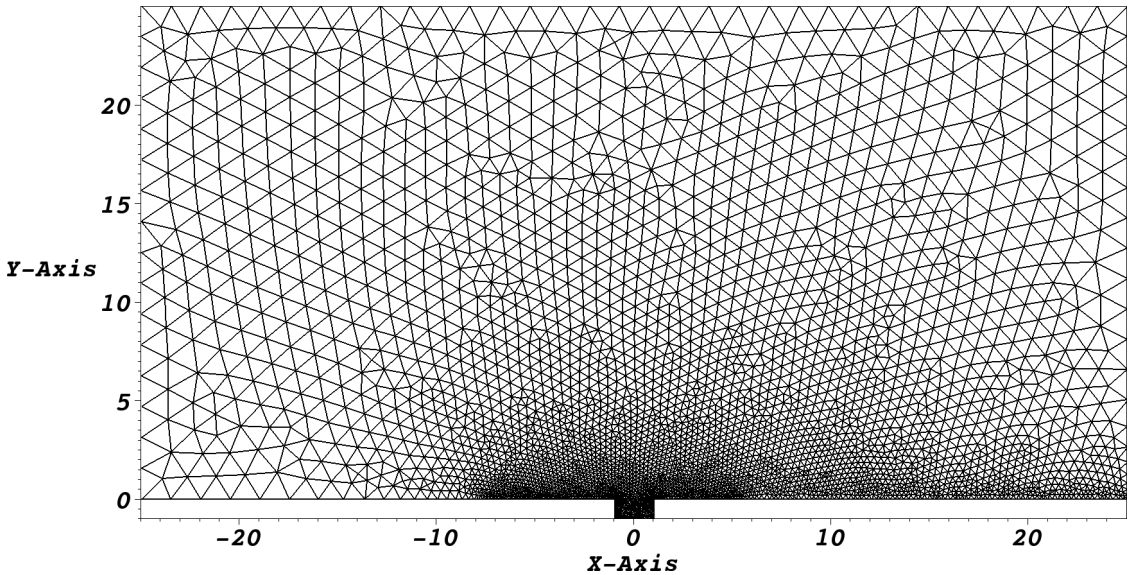


Figure 7.2: Mesh for computation of acoustics for open cavity problem.

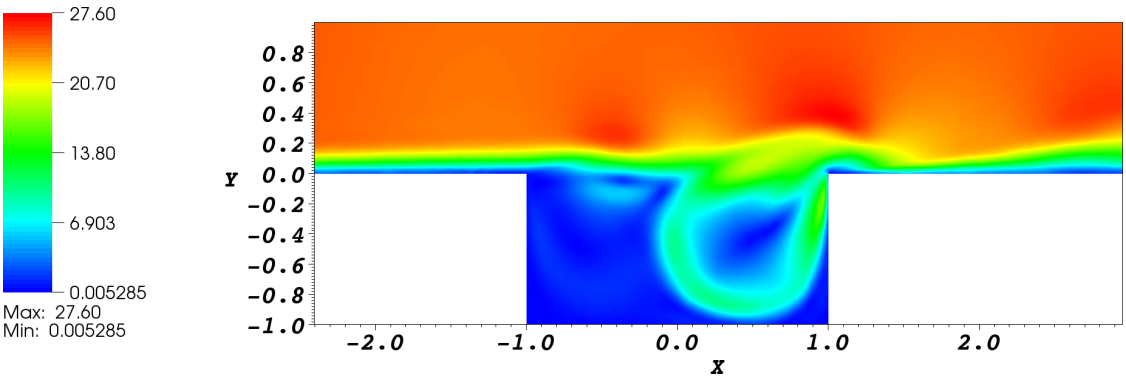


Figure 7.3: Snapshot of contour of velocity magnitude for Re 2500 flow over an open cavity.

spurious reflection of waves. For all the cases, the Reynolds number based on cavity depth is taken to be 2500. A fourth order ExRi-SD method in combination with the proposed WWR-WENO scheme is used to compute the viscous flow over the open cavity. The discretization of the viscous fluxes is performed using the one-step method based on ExRi scheme as proposed in Chapter 4. The computational mesh is generated with the minimum element size taken to be $D/8$ at the edges of the cavity and gradually increasing to $3D/2$ at the far-field. The corresponding mesh used is shown in Fig. 7.2. The free-stream density is set to $1\,\mathrm{Kg/m^3}$ and the coefficient of viscosity is taken to be $\mu = 0.01(\mathrm{Pa\ s})$. The free stream velocity U_∞ is set to 25m/s and the free stream pressure is set according to the free-stream Mach number of the flow by using the formula

$$\mathbf{p}_\infty = \frac{\rho_\infty}{\gamma}\left(\frac{U_\infty}{M_\infty}\right)^2 \tag{7.3}$$

where the subscript $_\infty$ represents free-stream quantities.

7.1.1 Subsonic case with Mach No. 0.5

The velocity contour corresponding to the Mach 0.5 flow over the open cavity is shown in Figure 7.3. The corresponding density contour is shown in Figure 7.4. The generation and propagation of acoustic waves from the shear layer over the cavity is clearly observed from the contour plots. For the case with Mach number

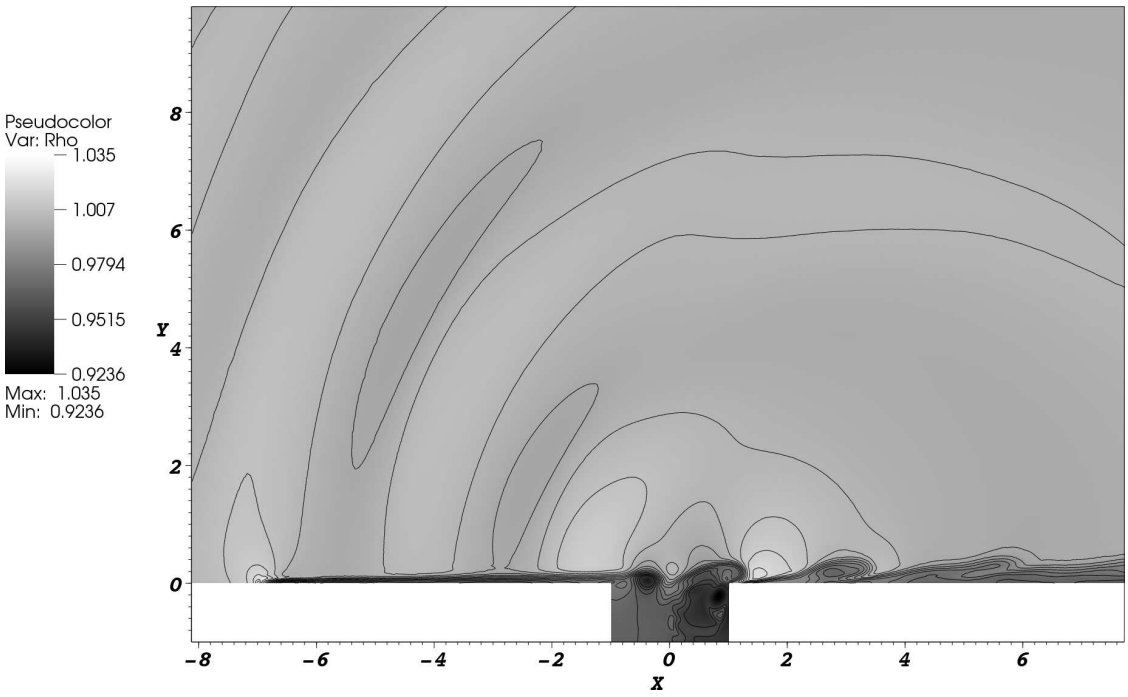


Figure 7.4: Density fluctuations for Re 2500 flow over an open cavity.

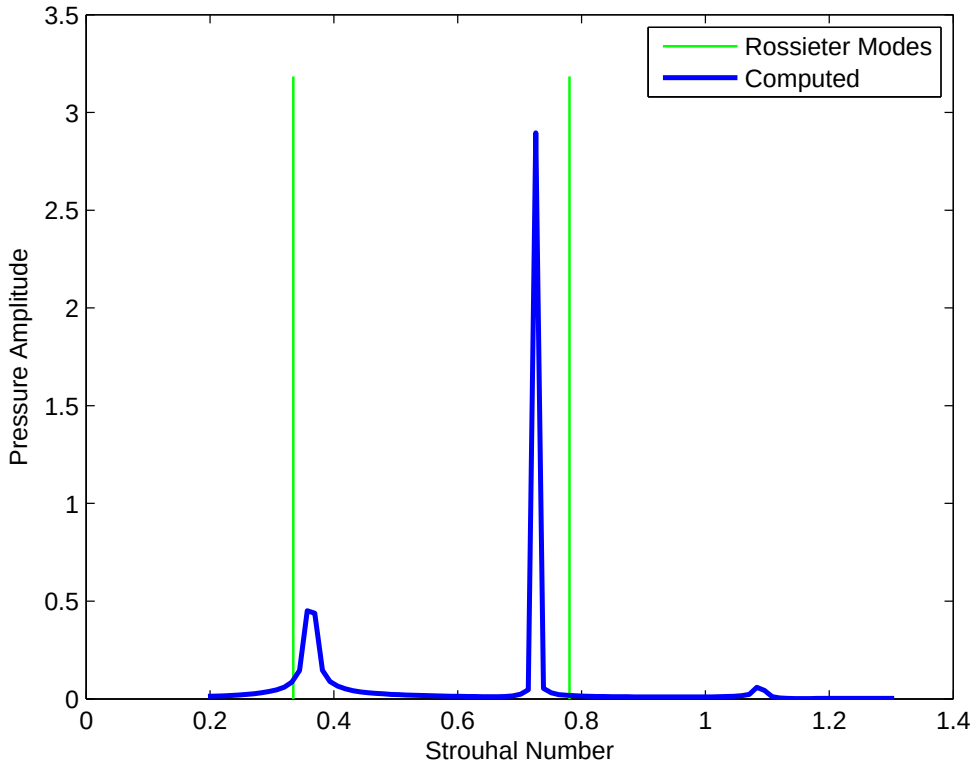


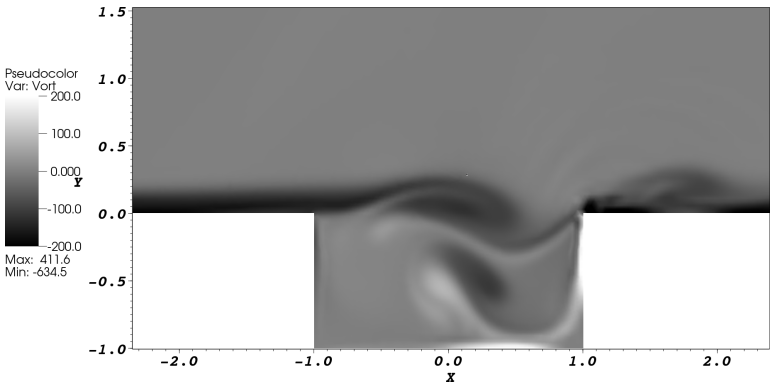
Figure 7.5: Frequency spectrum of the pressure signal for a Re 2500, Mach 0.5 flow over an open cavity with $L/D = 2$. (Rossieter modes correspond to empirically fitted data).

0.5, the flow is fully subsonic with no shocks in the flow field. Figure 7.5 shows the frequency spectrum of the measured acoustic signal for a free-stream flow of Mach number 0.5. Distinct peaks observed correspond to the harmonics of the acoustic tones generated by the cavity. The reference acoustic modes are obtained using the Rossieter's empirical formula [100] corresponding to the flow over open cavity. In Figure 7.5, the Strouhal number for the first two dominant modes match well with values obtained the Rossieter's formula. The acoustic spectrum show that the mode corresponding to $m = 2$ is most dominant. A similar observation was also reported in [101] for free stream Mach number of 0.65.

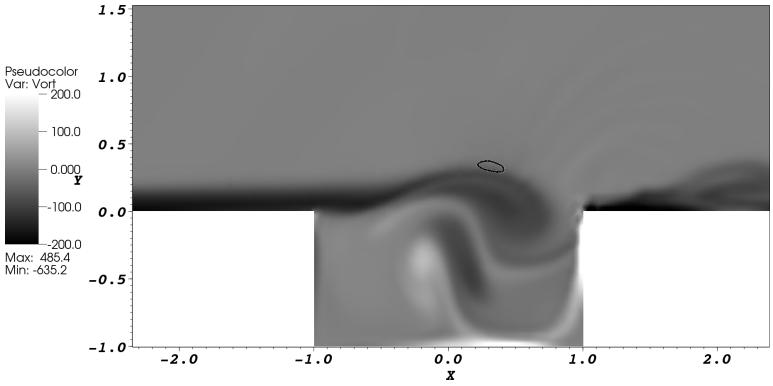
7.1.2 Transonic computations

As the free stream Mach number is increased, weak shock structures (local regions with mach number range $0.95 < M < 1.05$) are observed in the the flow field. These shock structures occur only at specific intervals during the course of vortex roll-up near the downstream edge of the cavity. The evolution of the vortex structure and the occurrence of shocks are highlighted in the contour plots shown in Figure 7.6. The presence of the vortex structure near the downstream edge of the cavity results in a large density and pressure gradient. Since the flow is deviated due to the shear layer instability, there is a relative increase in the velocity magnitude and hence the local Mach number. This results in formation of shock at this region. This process is found to be periodic occurring at each vortex shedding cycle. Fig. 7.7 shows the density, pressure and velocity near the cavity region during the occurrence of shock in the flow field. The frequency spectrum of the acoustic signal for the Mach 0.8 case (Figure 7.8) shows the occurrence of additional weaker modes along with the dominant modes. This could be due to the intermittent occurrence of shocks during the vortex evolution near the leading edge of cavity.

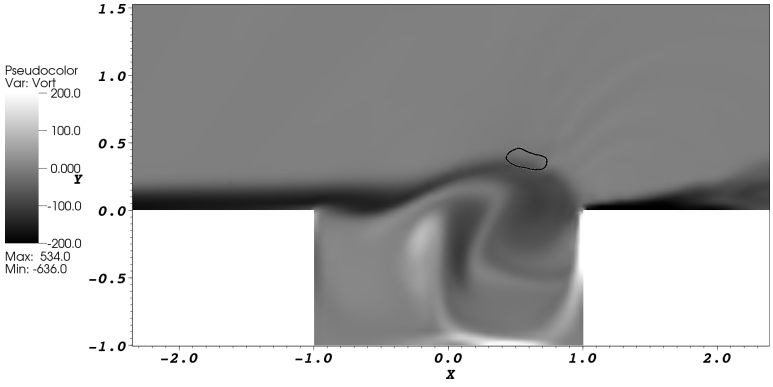
Additional computations were performed for free-stream Mach numbers 0.6 and 0.7. Figure 7.9 shows the plot of first two dominant modes with respect to the Mach number. The computed modes have a similar trend in comparison with



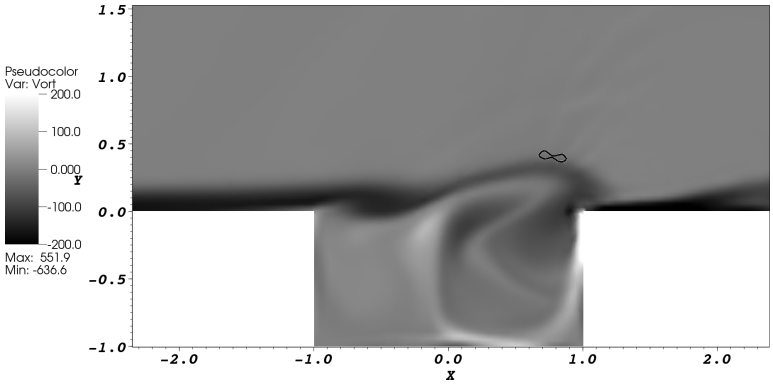
(a) T=0.61



(b) T=0.71

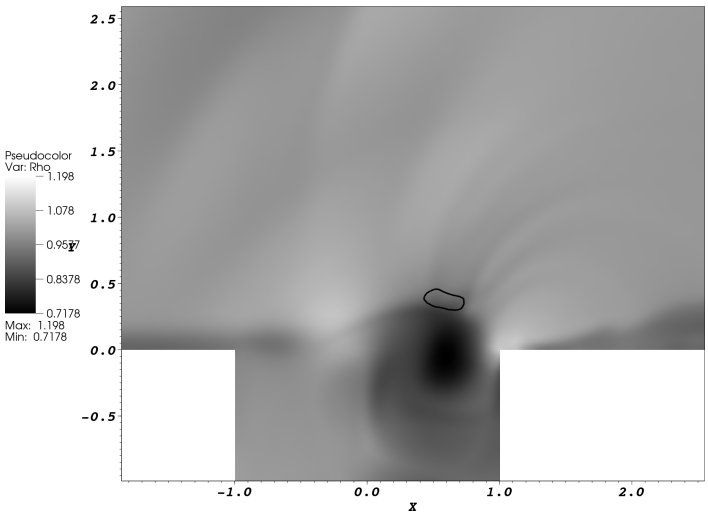


(c) T=0.8

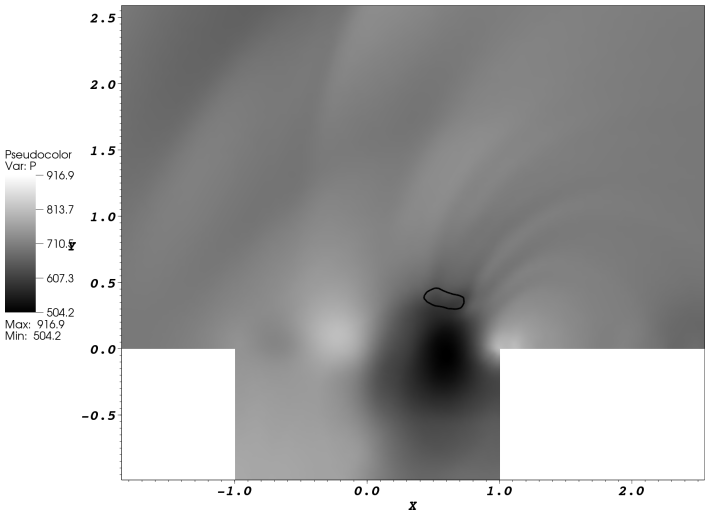


(d) T=0.9

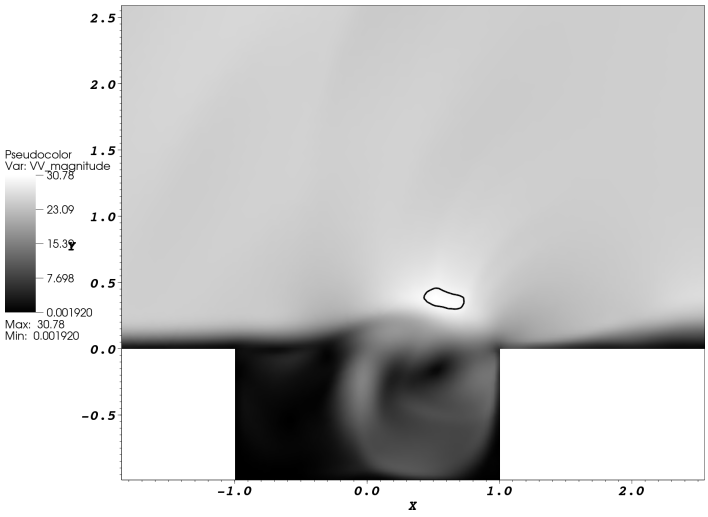
Figure 7.6: Snapshot of vorticity contours for Re 2500, Mach 0.8 flow over an open cavity with $L/D = 2$ (the black marker denotes the occurrence of weak shock in the flow field).



(a) Density



(b) Pressure



(c) Velocity Magnitude

Figure 7.7: Snapshot of flow field during occurrence of intermittent shock in a Mach 0.8 flow over an open cavity.

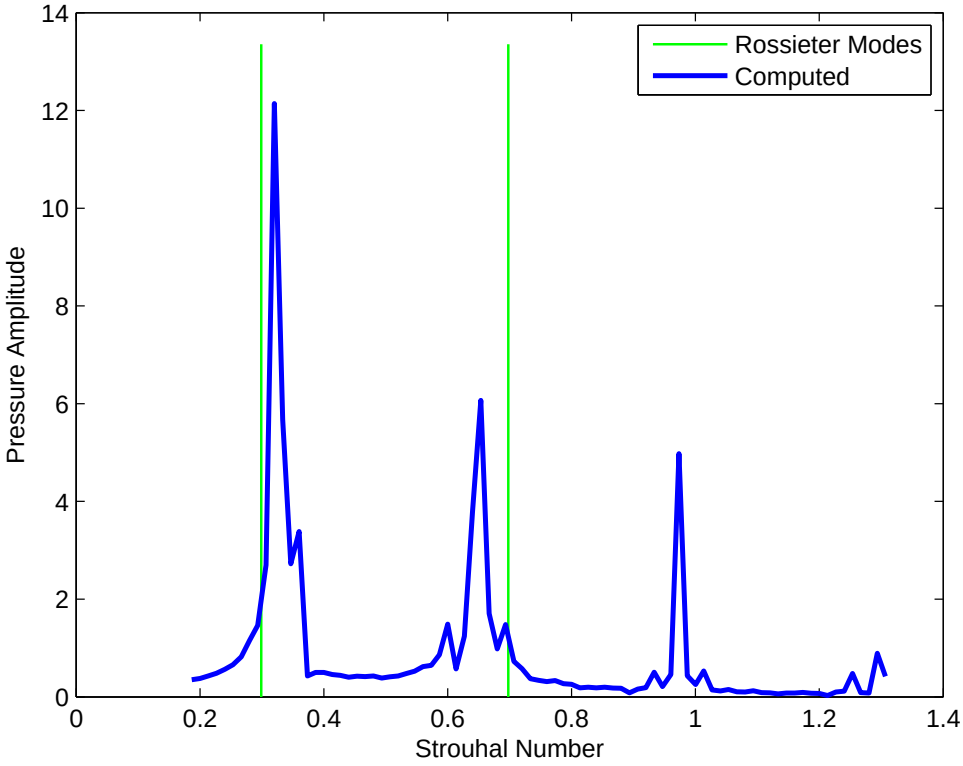


Figure 7.8: Frequency spectrum of the pressure signal for a Re 2500, Mach 0.8 flow over an open cavity with $L/D = 2$ (Rossieter modes are empirically fitted data).

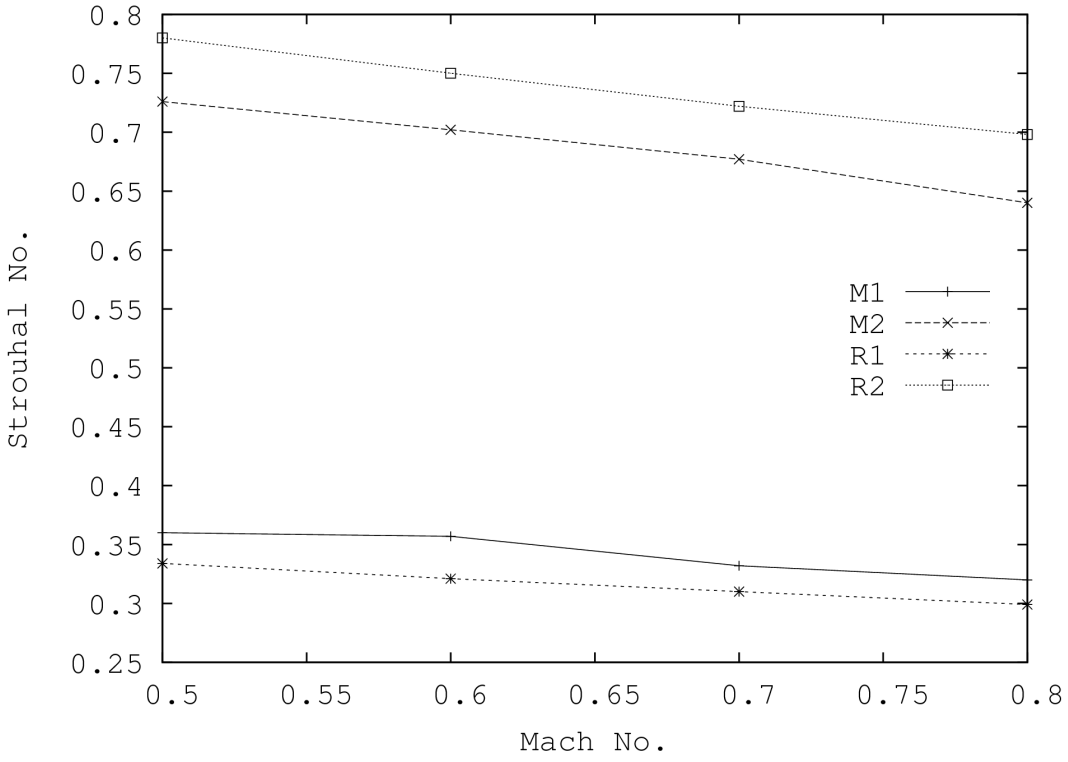


Figure 7.9: Dominant modes of acoustic tones generated by a Re 2500 flow over a open cavity: M1,M2 are the first two modes computed; R1 and R2 are the empirical Rossieter modes).

the Rossieter modes. The computations clearly demonstrate the capability of the proposed schemes to accurately and efficiently simulate the complex non-linear mechanism of acoustic sound generation even in presence of shocks.

7.1.3 Application of adaptive time stepping scheme

The flow around open cavity often requires fine meshes near the leading and trailing edges. This results in anisotropic meshes having smaller elements near the edges and larger elements as one moves towards the far-field, making it a suitable candidate to test the adaptive time stepping algorithm proposed in Chapter 5. To demonstrate the effectiveness of the adaptive time stepping scheme, we choose a more resolved mesh with very fine grids at the edges of cavities. For this case, the minimum element size is taken to be $D/40$ at the edge. The corresponding mesh is shown in Figure 7.2. For the MTS algorithm, the value of $\mathcal{N}$ is set to 2, resulting in 6 block mesh (5 levels of mesh sizes) as shown in Fig. 7.10. The free stream

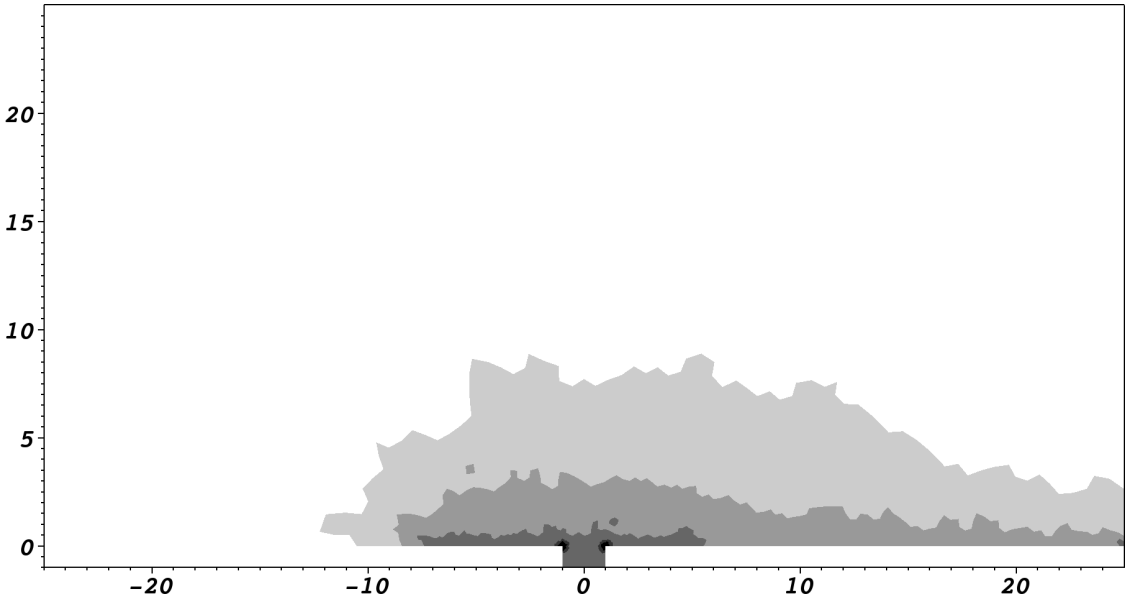


Figure 7.10: Levels of mesh used for computation of acoustics for open cavity problem (6-blocks, 5 recursion levels, $\mathcal{N} = 2$).

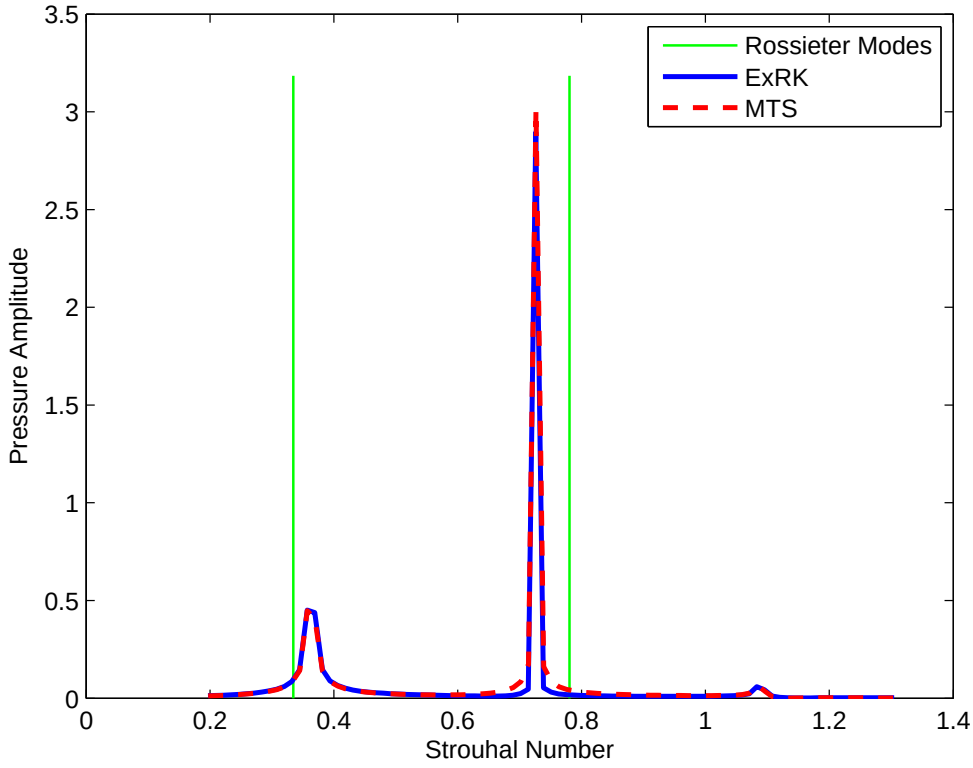


Figure 7.11: Comparison of computed acoustic modes with and without the application of MTS scheme (Rossieter modes are empirically fitted data).

Mach number is chosen to be 0.5 and the Reynolds number based on cavity length is set to 2500. Two computations are performed with and without the MTS algorithm applied on the basic scheme. With the application of the MTS algorithm, a speedup of 5.2 over the basic scheme is achieved. The improved computational efficiency is achieved without sacrificing the solution accuracy. The harmonics of the measured acoustic tones for the two cases are plotted in Figure 7.11. The acoustic spectrum obtained using the MTS scheme is nearly indistinguishable from that obtained using the basic scheme without the MTS scheme.

7.2 Acoustic tones generated in a reed-like instrument

The reed instruments are instruments that generate acoustic tones when the player blows air against a sharp edge or the reed. These instruments generate acoustic tones without use of any mechanical parts. The flow mechanism involved in the reed instruments is equivalent to the classical "edge-tone problem" in which the plane jet oscillates over a wedge shaped object. The oscillations result in generation and shedding of interacting vortices that get convected downstream. The entire phenomenon of acoustic tone generation is highly non-linear and is extremely difficult to capture various modes corresponding to the tones generated by the oscillating flow.

In this work, we consider a particular case of acoustic tone generation by a reed-like instrument. The motivation of this study is more towards analyzing the effectiveness of higher-order schemes in capturing the various harmonics of the generated acoustic tones. Hence, we restrict the current scope of study to a single problem configuration.

The schematic setup of the reed problem is shown in Fig. 7.12. The jet inlet is constructed using a channel of length 10mm. A wedge of angle 20° and thickness of 3mm is placed at a distance of 6mm from the exit of mouthpiece. No-slip

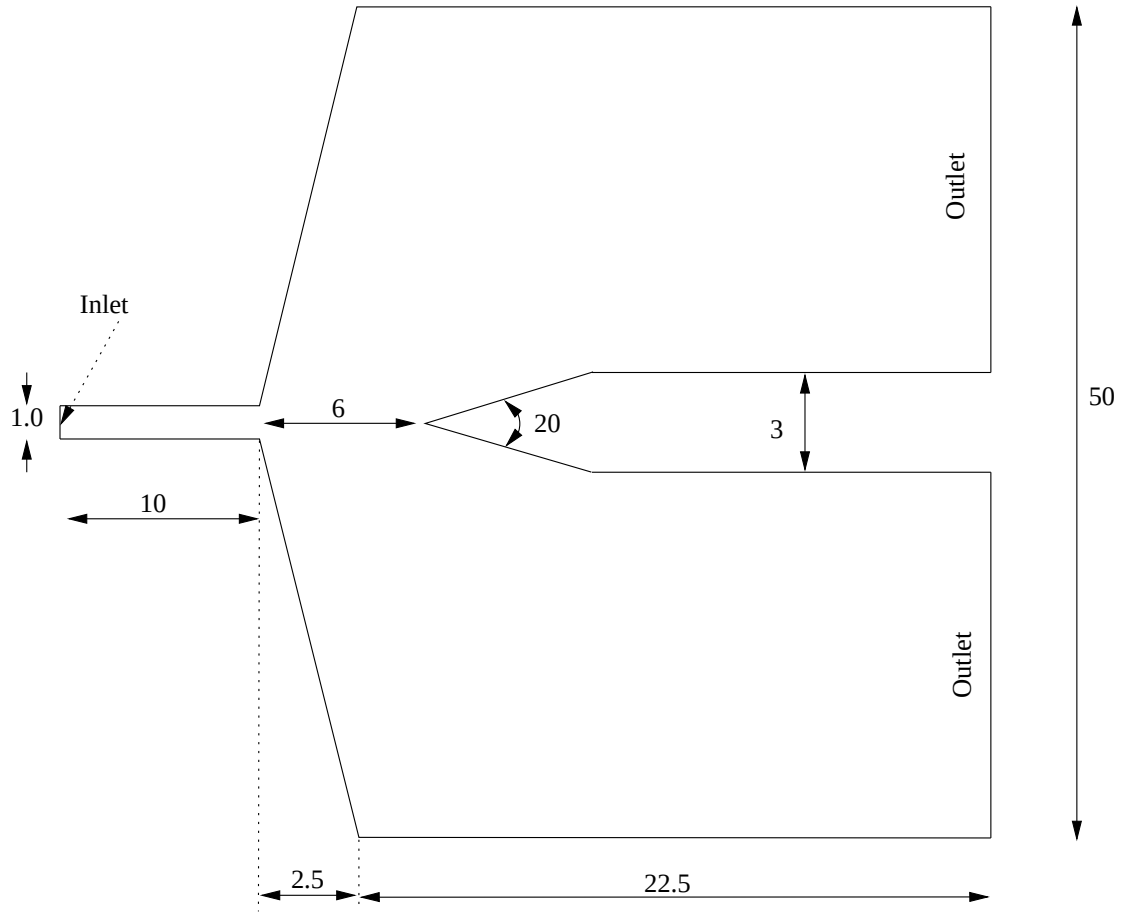


Figure 7.12: Schematic setup for simulating acoustic tones generated in reed-like instrument (units in mm, figure not to scale).

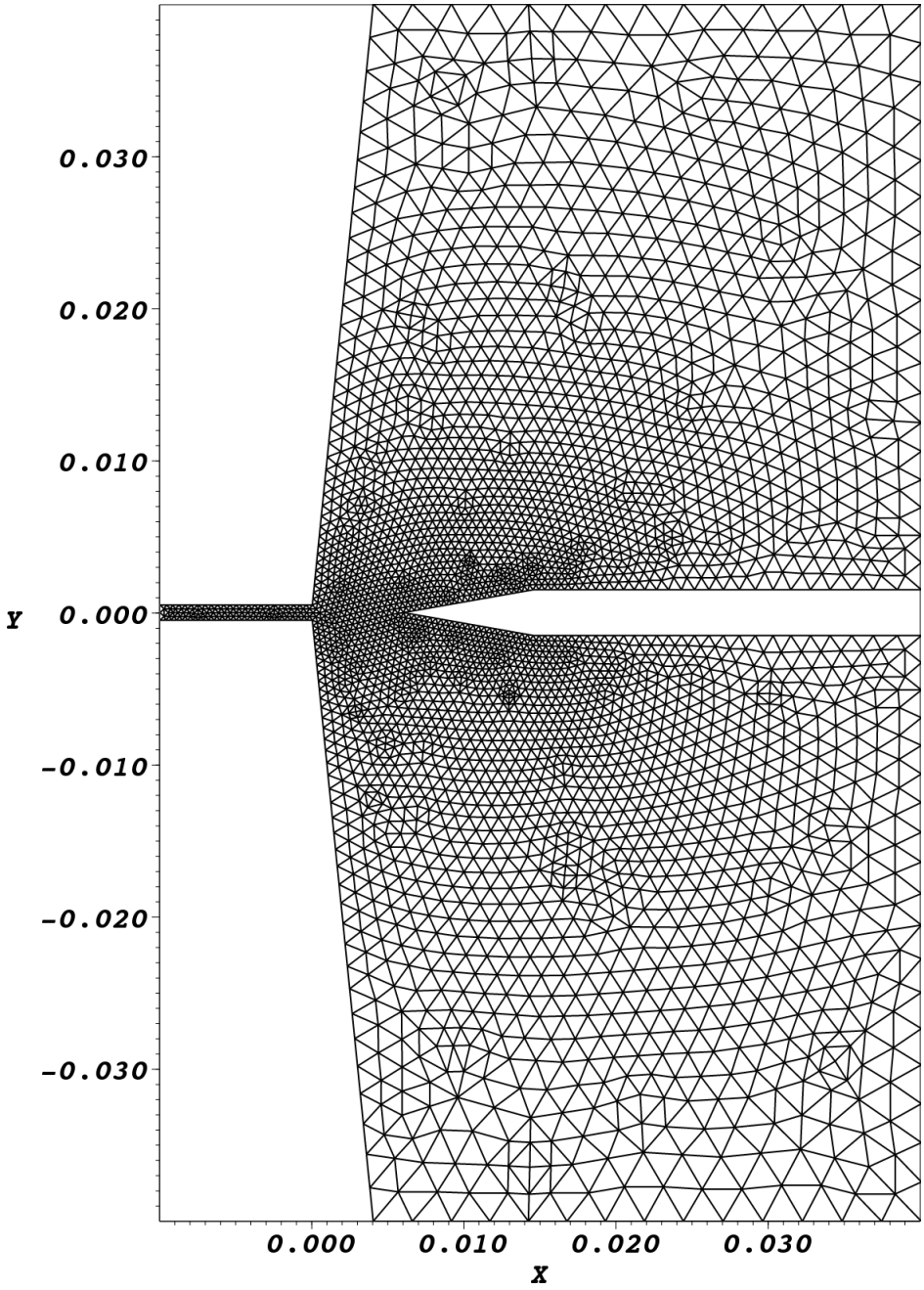
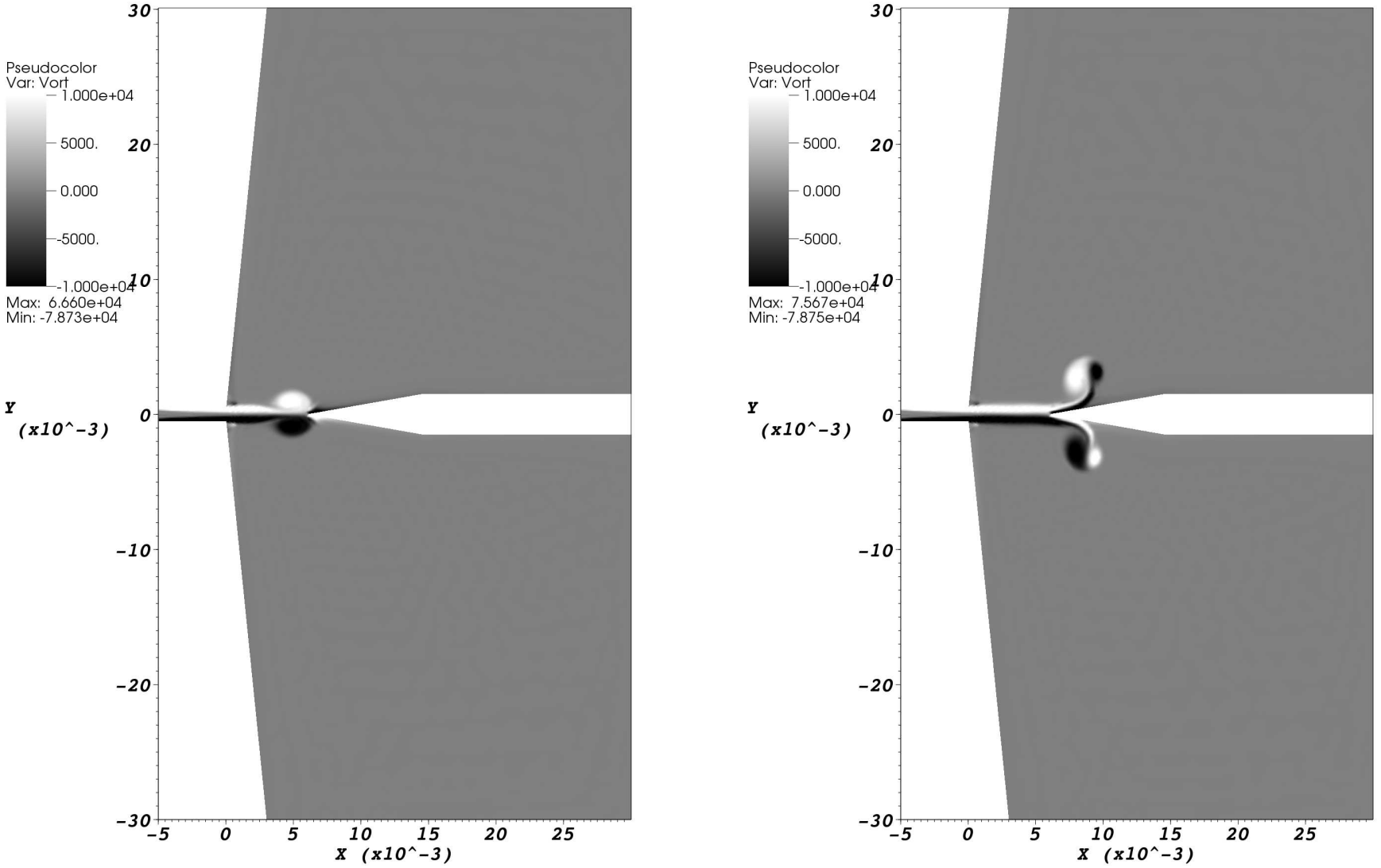


Figure 7.13: Mesh used for simulation of acoustic tones generated in reed-like instruments.

boundary conditions are applied at all the boundaries other than the inlet and outlet. The fluid is taken to be air with properties: density $\rho_0 = 1.184(\text{kg/m}^3)$, pressure $\boldsymbol{p}_0 = 101325(\text{Pa})$, coefficient of viscosity $\mu = 1.71 \times 10^{-5}(\text{Pa s})$, specific heat ratio $\gamma = 1.4$ and gas constant $R = 8.31451$. The inlet jet velocity is set as 5m/s.

The corresponding mesh generated is shown in Figure 7.13. To capture the complex interactions of jet and the vortices, a finer mesh is used for the region between the inlet and the reed resulting in a total of 6852 elements defining the entire computational domain. Absorbing boundaries are used at the outlets and no-slip boundary condition is applied for all the surrounding walls.

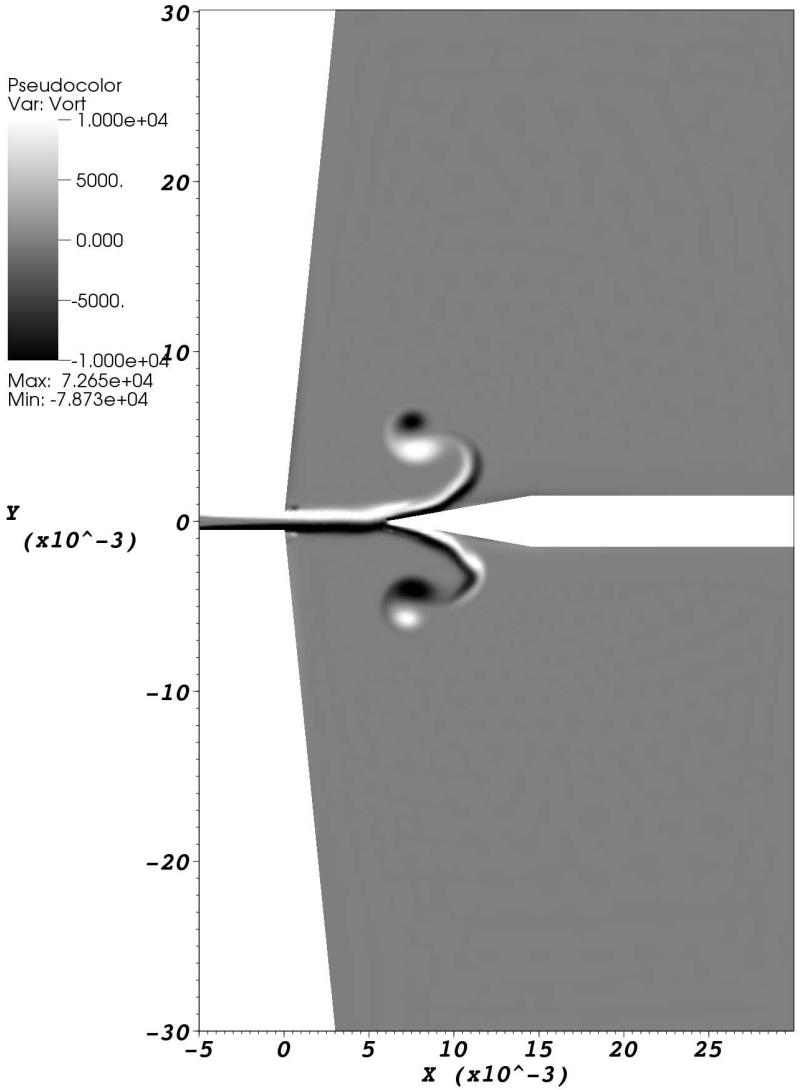
Figure 7.14 shows the initial transient stage when the impinging jet becomes unstable. The smaller vortices generated at either side of the edge roll-up and get pulled in by the primary jet. These vortices later interact with the jet itself, thereby resulting in a more complicated flow pattern. The oscillation of the jet results in periodic shedding of the vortices. Figure 7.15 shows the snapshots at different time instances in the process of vortex shedding. It is inferred from the figure that the vortex shedding takes place approximately at every 3 snap-shot interval. With a time step size of 9.05×10^{-4} seconds between the snapshots, the frequency of vortex shedding is calculated (approximately) as 368 Hz. Note that the vortex shedding occurs on either side of the reed. Figure 7.15 also shows the formation of larger vortical structure arising from the smaller vortices being shed at the reed edge. In order to visualize the larger structures, the contour ranges for vorticity are adjusted to lower levels and the resulting snapshots are presented in Figure 7.16. It is observed that the vortices are reintroduced into the vorticity jet stream near the tip of the reed. This provides a feedback mechanism that is eventually synchronized with the oscillation of the jet. On closer observation of the vortex structures, it was observed that the time taken for a vortex to circulate and re-join the jet stream before the reed is roughly around 0.0281 seconds, which corresponds to a frequency of 35.6 Hz. This suggest that there should be 5 vortices on each



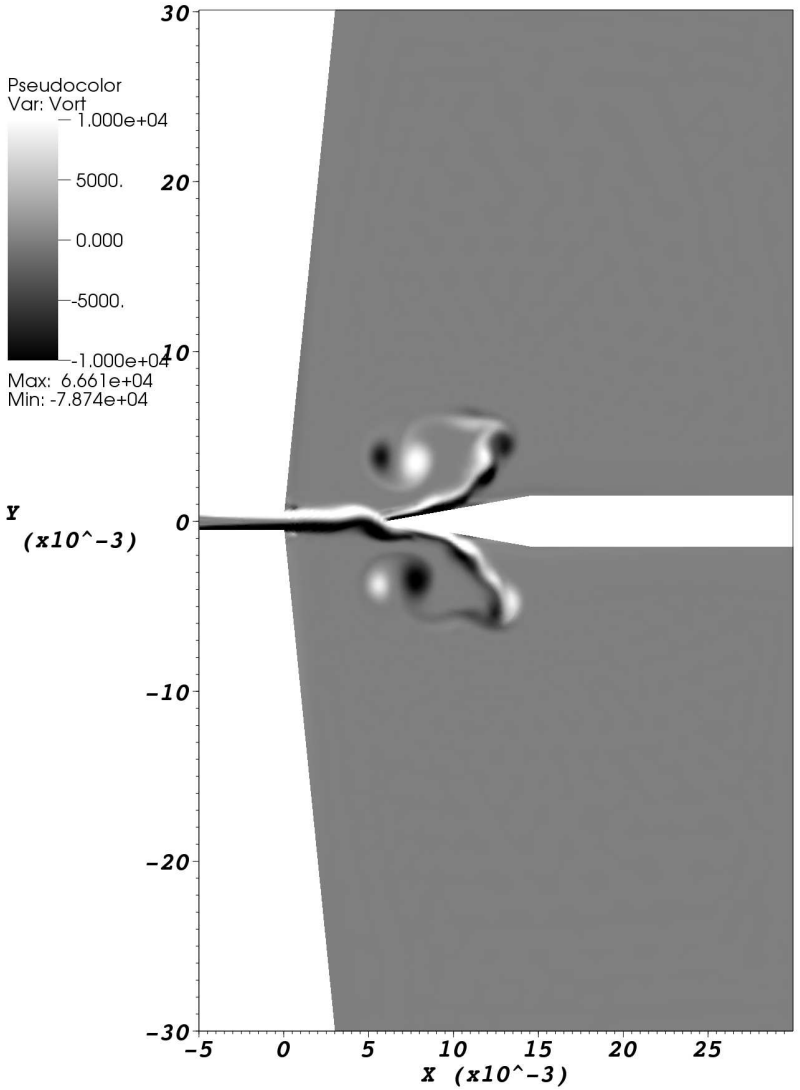
(a) $T = 0.0027$

(b) $T = 0.00464$

Figure 7.14: Jet impinging on a reed: initial onset of instability on jet (Snapshots from $T = 0.0027$ to $T = 0.0124$).

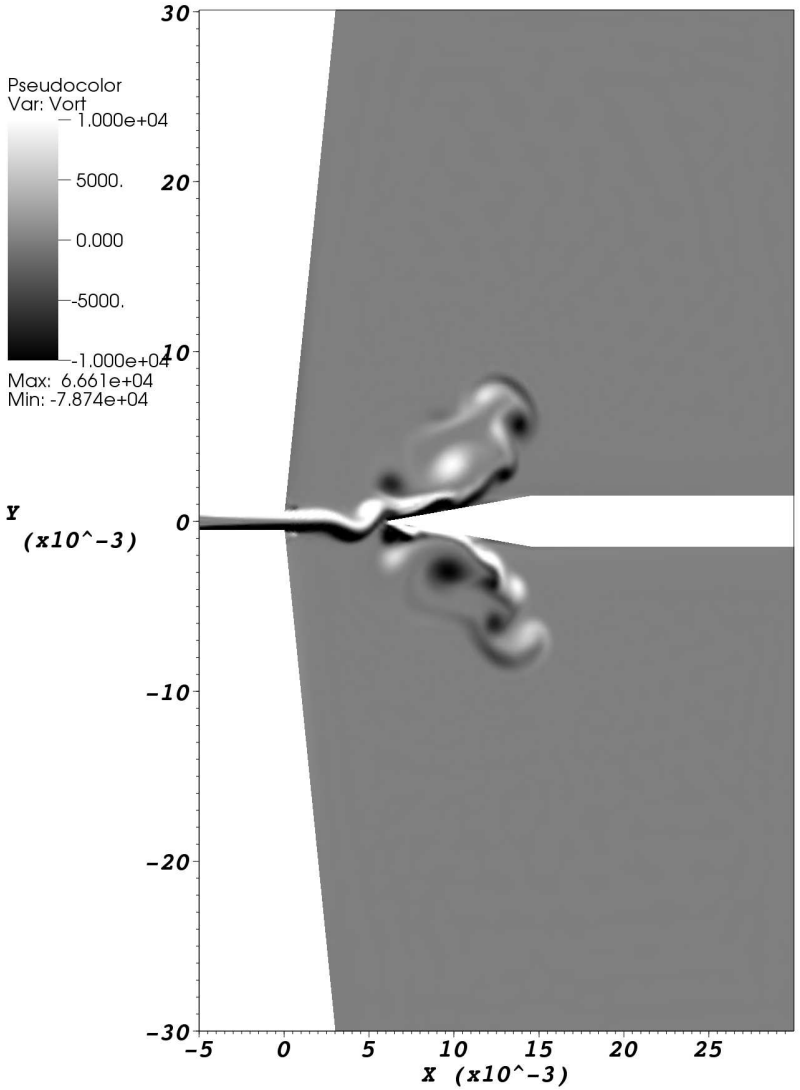


(c) $T = 0.00658$

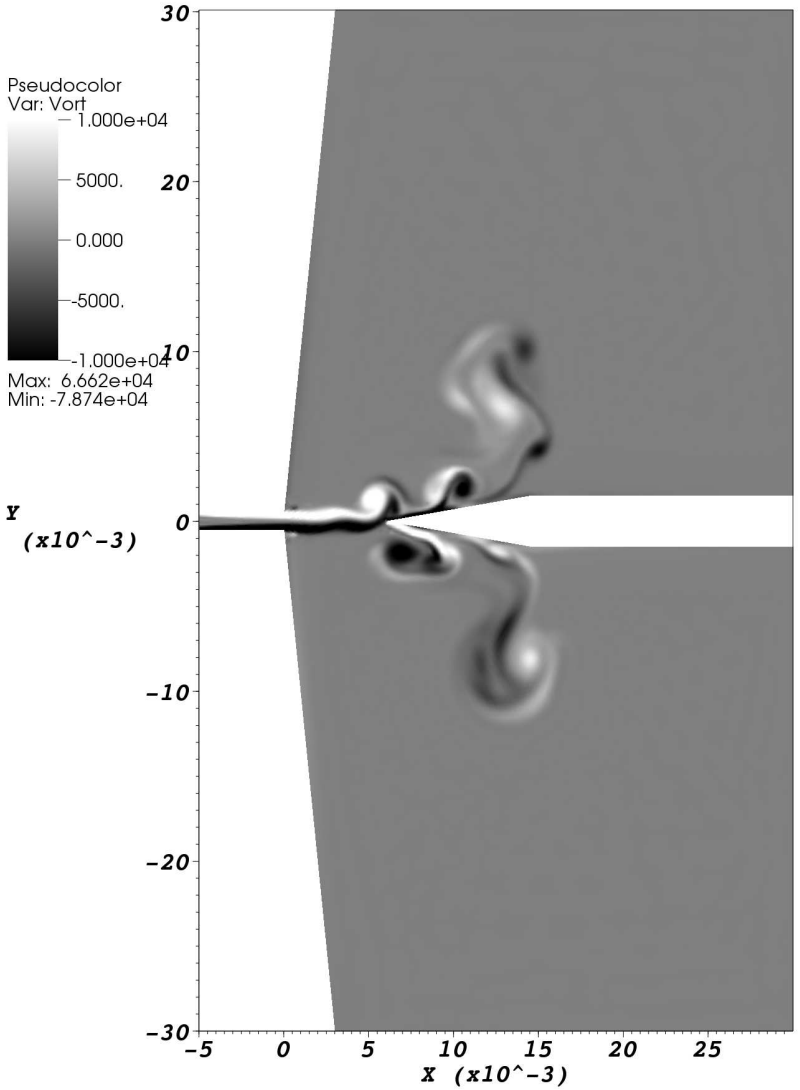


(d) $T = 0.00852$

Figure 7.14: Jet impinging on a reed: onset of instability on jet (Snapshots from $T = 0.0027$ to $T = 0.0124$). *contd.*

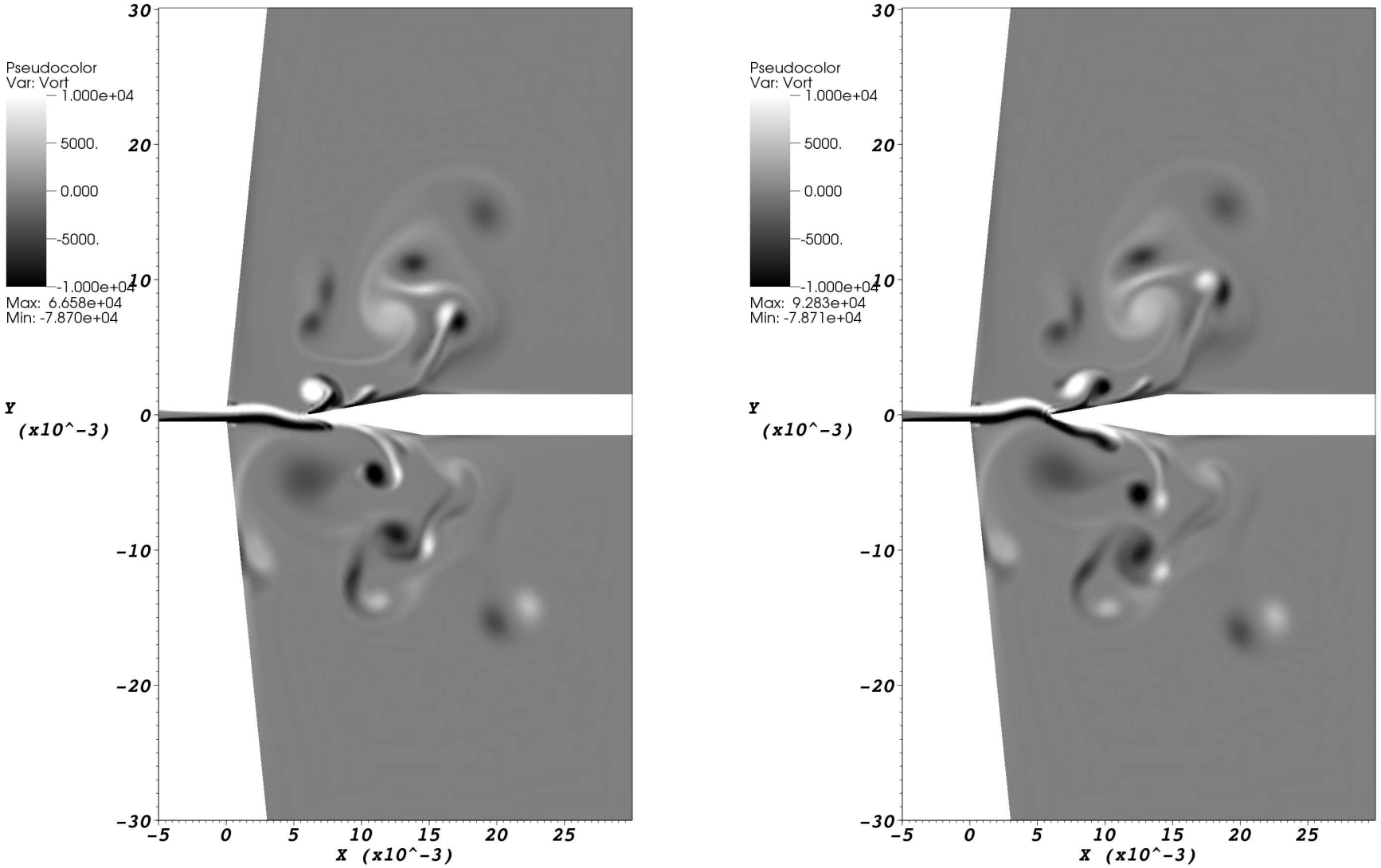


(e) $T = 0.01046$



(f) $T = 0.0124$

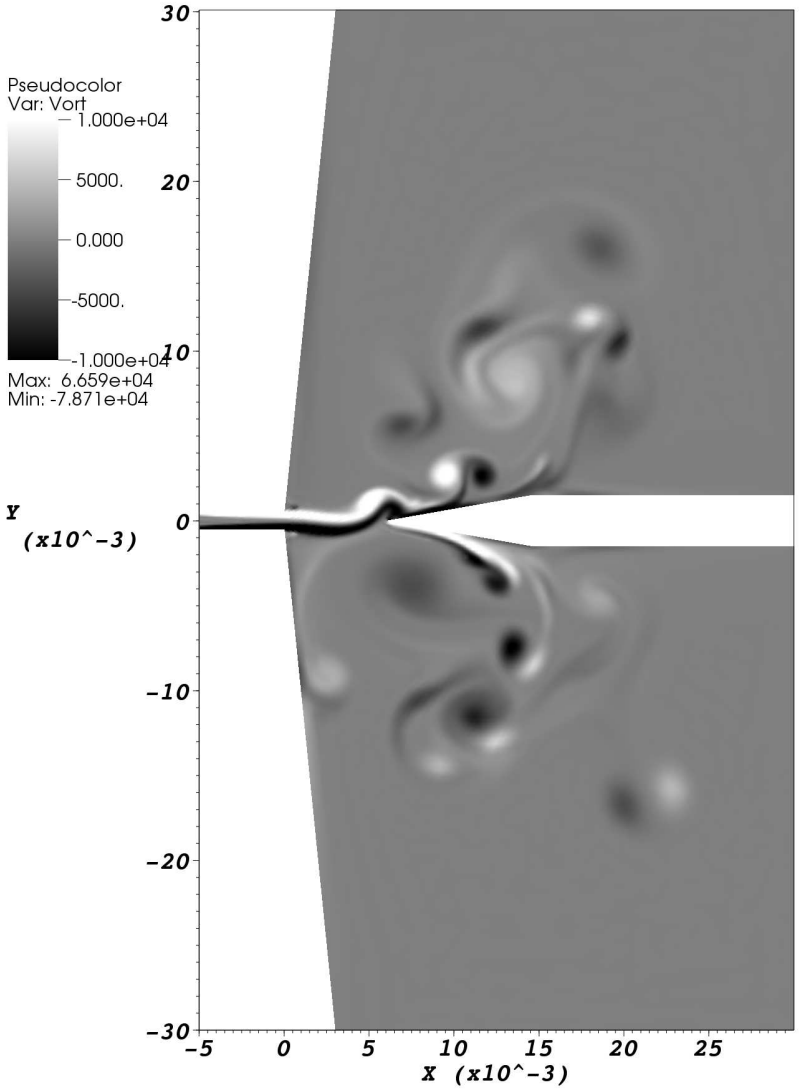
Figure 7.14: Jet impinging on a reed: onset of instability on jet (Snapshots from $T = 0.0027$ to $T = 0.0124$). *contd.*



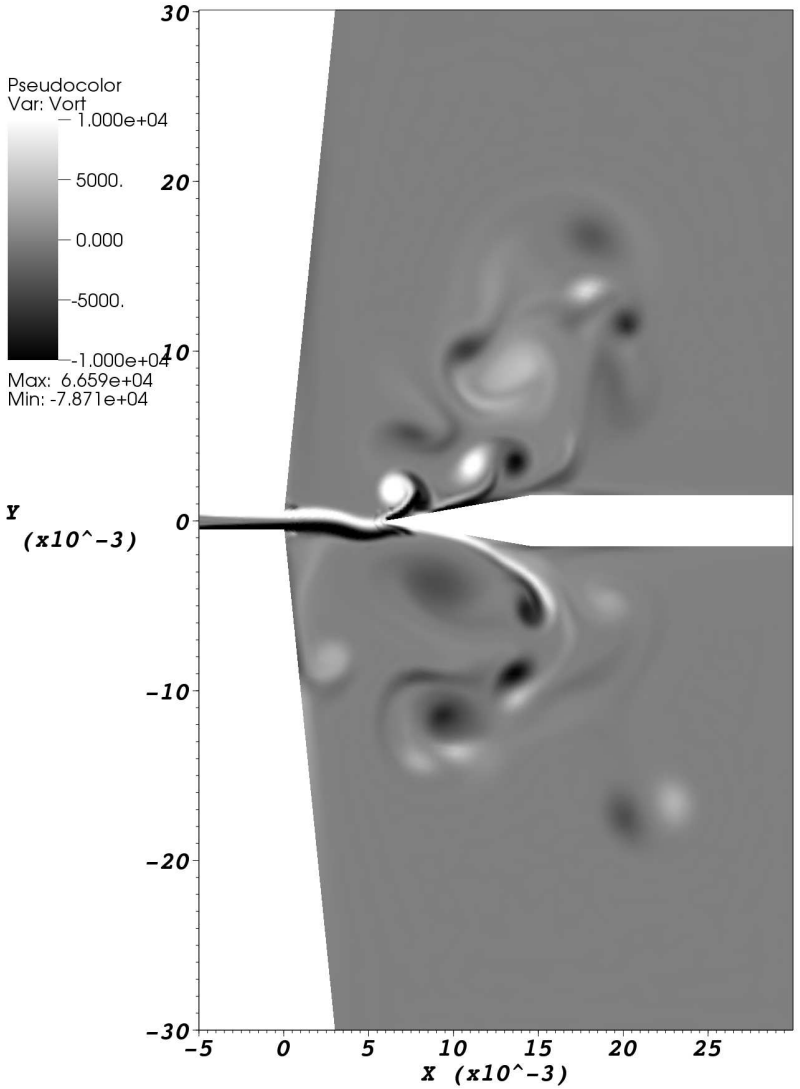
(a) $T = 0.0263$

(b) $T = 0.0.027205$

Figure 7.15: Periodic shedding of vortices near the edge of reed structure (Time step between snapshots=9.05×10^{-4}).



(c) $T = 0.02811$



(d) $T = 0.029015$

Figure 7.15: Periodic shedding of vortices near the edge of reed structure (Time step between snapshots=9.05×10^{-4}). *contd.*

side of the reeds at a given point in time or 10 vortices in total. These vortices are periodically re-ingested into the vorticity jet approaching the reed, giving rise to a frequency of 356 (=10 × 35.6) Hz, which agrees well with the estimated vortex shedding frequency of 368 Hz (mentioned earlier).

During the computation, the absolute pressure is recorded at a point located at (9,15) mm from the reed edge. The acoustic signal and the corresponding Fourier Transform of the signal is given in Figure 7.17. Distinct peaks are observed in the spectral plot, illustrating the harmonics (acoustic modes) corresponding to the signal. As discussed before, the first peak corresponds to a frequency of 33.3 Hz. This coincides with the rough estimate of 35.6 Hz which was obtained by computing the time taken for the vortex to be shed from the jet and to get re-introduced (feedback) in the jet flow. The second peak corresponding to a frequency of 333.3 Hz also coincides with the previous study where the frequency of vortex shedding was roughly estimated as 368 Hz. For the acoustic modes obtained in an edge tone problem, Brown [3] proposed a formulation based on experimental observations. The frequency of the edge tone as per his experiments [3] is given by

$$f = 0.466j(100U_{jet} - 40)\left\{\frac{1}{100l} - 0.07\right\} \quad j = 1.0, 2.3, 3.8, 5.4 \tag{7.4}$$

where f is the frequency of the edge tone in Hertz, U_{jet} (m/s) is the velocity of the jet impinging on the reed edge at a distance of l (m) from the jet nozzle. The harmonics with frequencies 333.3Hz and 800Hz obtained from the pressure signal coincide well with the empirical formula in Eqn. (7.4) which yields to 342.3Hz (j=1) and 787.2Hz (j=2.3).

7.3 Summary

The numerical methods developed in this work are applied to direct computation of aerodynamic sound for two different test cases. For the computation of acoustic tones generated in an open cavity, the measured acoustic spectrum matched

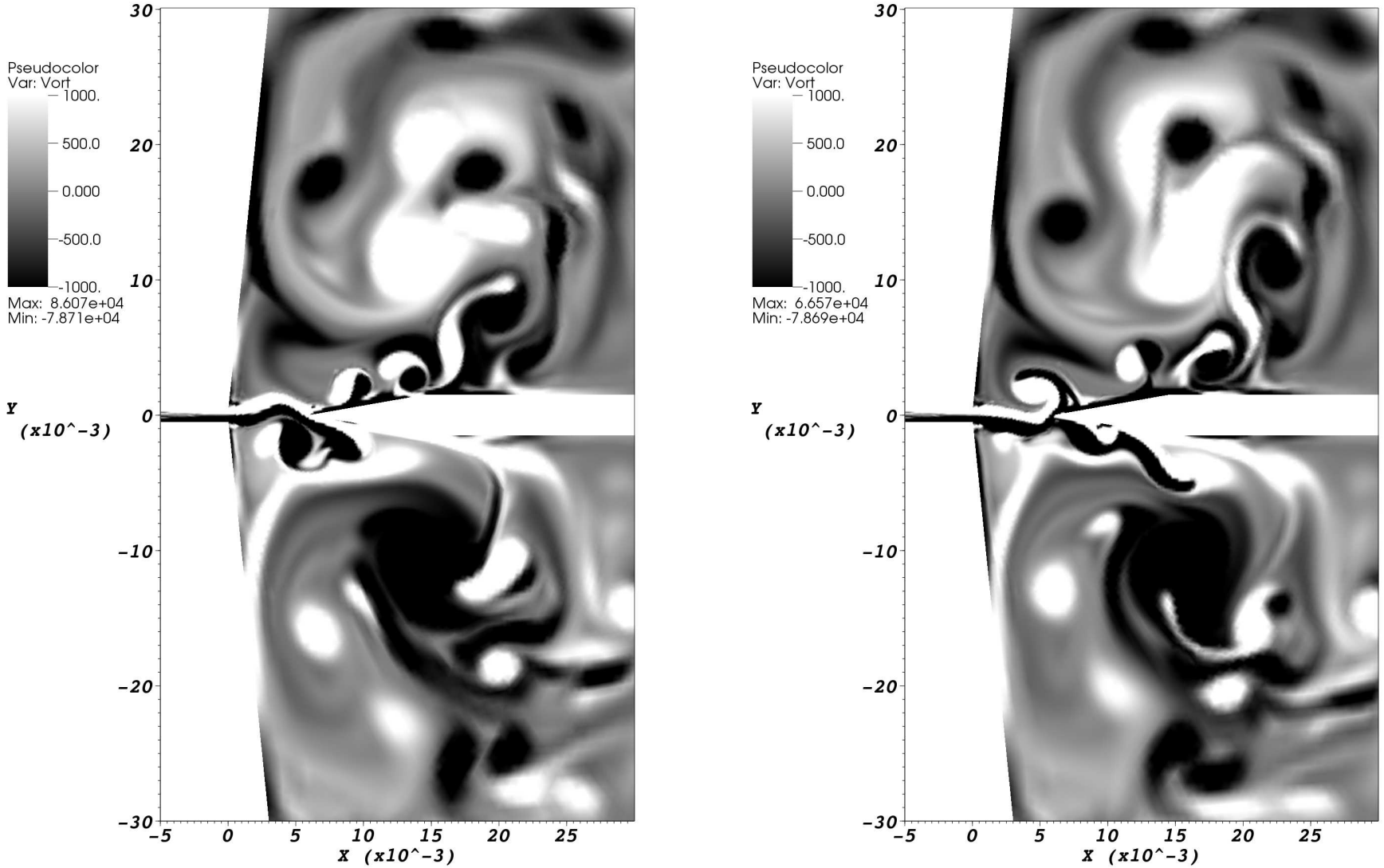


Figure 7.16: Evolution of larger vortical structures and their interaction with the jet (Time step between snapshot: 1.811×10^{-3}).

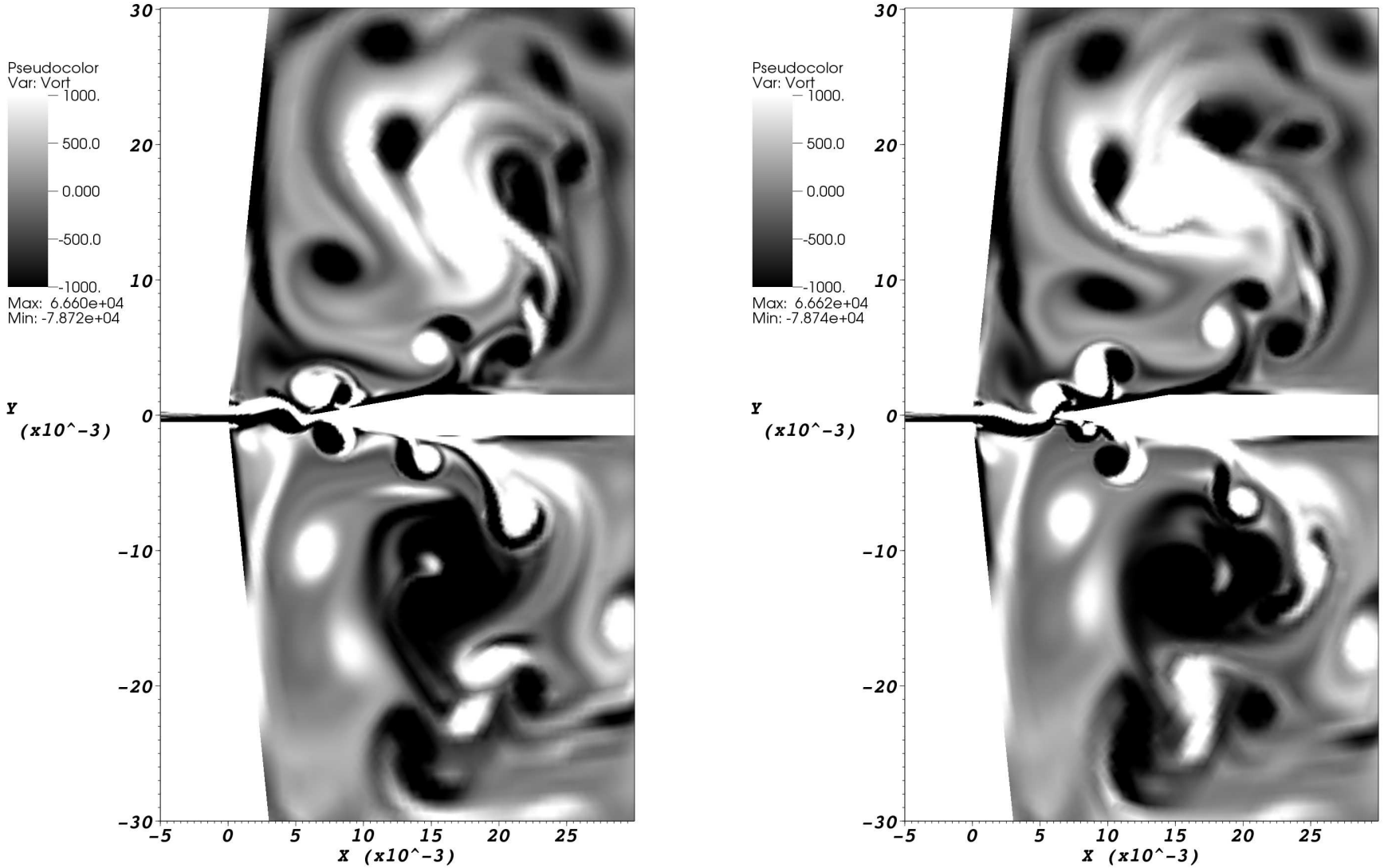


Figure 7.16: Evolution of larger vortical structures and their interaction with the jet (Time step between snapshot: 1.811×10^{-3}). *contd.*

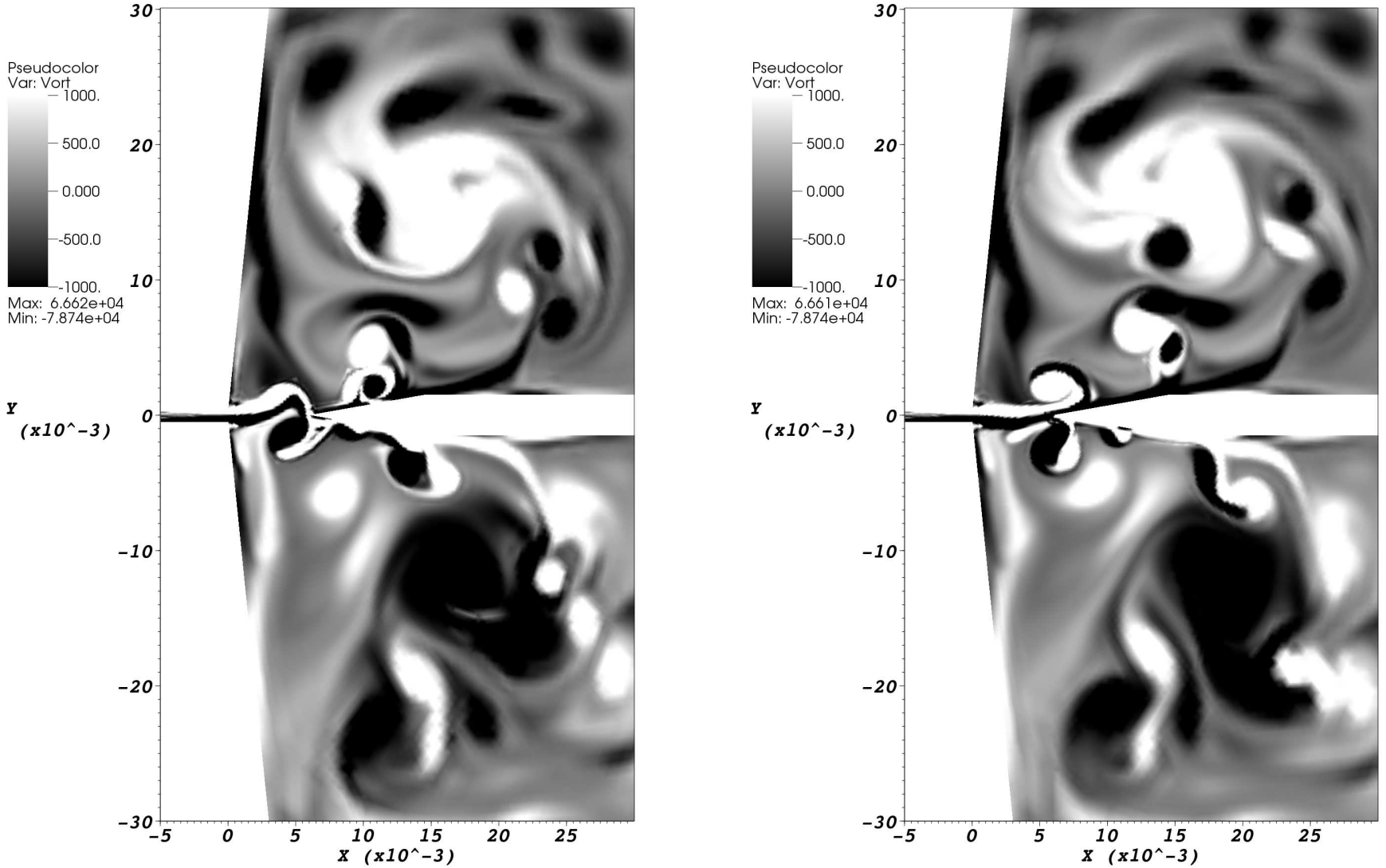
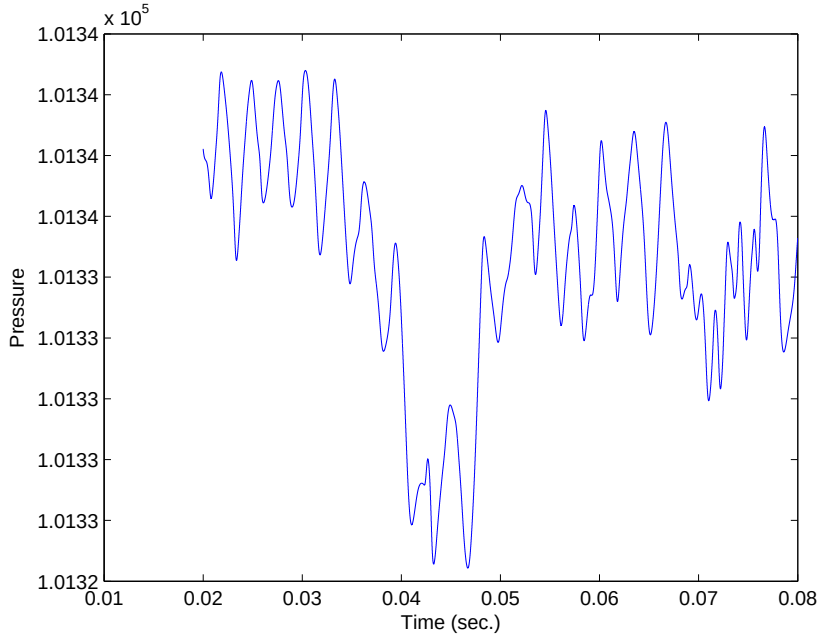
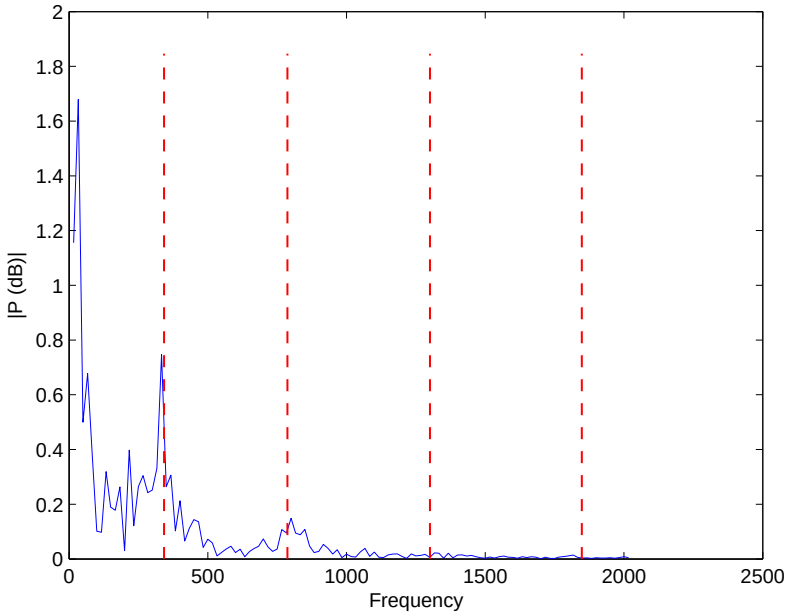


Figure 7.16: Evolution of larger vortical structures and their interaction with the jet (Time step between snapshot: 1.811×10^{-3}). *contd.*



(a) Pressure signal



(b) Acoustic spectrum of pressure signal

Figure 7.17: Acoustic spectrum of the edge tone ($T = 0.02 - 0.08$) generated by a jet impinging on a reed (Dotted lines denote the edge tone frequencies obtained using Brown's [3] formulation).

well with that of the empirical values obtained using Rossieter's formulation. No corruption of acoustic signals were observed in the subsonic computation with free stream Mach number of 0.5. As the flow becomes transonic, the shock structures occur at a specific interval during the roll-up of vortex at the downstream edge of cavity. Shock capturing is facilitated with the high resolution WENO scheme developed in Chapter 6. For the transonic cases, the additional peaks observed in the acoustic spectrum (Mach 0.8) could be due to the intermittent occurrence of shocks in the flow field. The effectiveness of the WENO shock capturing scheme and the multi-time stepping schemes are demonstrated in the open-cavity problem. The second test case corresponding to a jet impinging on a reed has been successfully computed. The mechanisms of sound generation are qualitatively and quantitatively analyzed with appropriate correlations to the observed edge tones. The computed tones match very well with that of the first two harmonics of Brown's empirical equation. The application of the numerical schemes to the aero-acoustic problems clearly demonstrate the capability of the schemes to accurately compute the complex process of aerodynamic sound generation.

Chapter 8

Conclusions and Outlook

8.1 Summary and Conclusion

This thesis details research towards developing efficient, high resolution, high accuracy numerical methods for computing complex non-linear interactions occurring in viscous compressible fluid flows which may contain shocks. The primary contributions in this work are the formulation of a continuous flux function, alternative one-step formulation for obtaining gradients at element interface for viscous flow computation, a simple and efficient multi-time stepping scheme and high resolution compact WENO schemes for shock capturing.

- The Extended Riemann method developed in Chapter 3 is found to have an improved stability and hence support a larger time step compared to the standard DG schemes. The same trend is observed when the scheme was successfully extended to triangle elements. It is observed that the stability of a given numerical scheme depends to a greater extent on the method of blending the Riemann flux corrections with the solution at the interior of the element. Explicit modeling of this influence has direct implications on numerical stability and wave propagation characteristics of the scheme. An interesting observation was made in which, specifying the extent of influence of Riemann corrections through the desired CFL number ($\mathcal{C}$) resulted in direct

influence on the actual/measured CFL number of the numerical scheme. The various theoretical and numerical experiments confirmed that the proposed ExRi schemes can achieve higher-order accuracy while supporting a relatively larger time step.

- The ExRi technique was used to obtain an approximation for the solution inside an element. This approximate solution can now vary smoothly from the element interface to the element interior. This smooth variation of the solution at the interface facilitates the approximation of field gradients at the interface and hence offer a natural extension of the method to viscous compressible flows. Application in viscous compressible flows show that the proposed scheme can achieve higher order accuracy for steady and unsteady flow computations.

- The fundamental understanding of wave propagation within a discrete element is extended to formulate adaptive multi-time stepping schemes. Improved algorithms were formulated for schemes that support higher order solution representation within the element. The developed time stepping algorithm is independent of the nature of spatial and temporal discretization and hence can attain very high orders without any change in the formulation. Numerical tests show that the schemes can significantly reduce the computational cost (up to 80% demonstrated in current work) while maintaining the solution errors in the same level as the original schemes without adaptive time stepping.

- Compressible flows can have solution discontinuities in form of shocks. Following the ExRi concept of the Riemann solution influencing the interior solution evolution, a new method of WENO reconstruction is formulated. It is demonstrated that the onset of spurious oscillations occur predominantly at the element boundaries while the interior element solution remain relatively smooth. This behavior of the numerical schemes is exploited while

reconstructing a smooth solution in the vicinity of shocks. In addition to the reconstruction procedure, improvements on adaptive stencil weight computation are also suggested for the existing WENO schemes. The resulting numerical scheme is found to have high resolution even in the vicinity of shocks. Numerical experiments show that the scheme can resolve a shock within one cell width. The captured shock structure remains smooth and is unaffected by the mesh discretization. The shock resolution in the solution obtained with the proposed WENO scheme is comparable with the existing solution obtained using the state of the art shock capturing schemes on finer grids. A formulation of an oscillation indicator is presented with which we can estimate the extent of solution oscillation in an element and the applicability of WENO numerical scheme to improve the oscillatory solution. The use of this oscillation indicator results in selective application of WENO procedure to less than 10% of the computational domain, thereby achieving significant savings in computational cost.

- The complete set of numerical schemes developed in this research is applied to direct computation of aerodynamic sound. For the flow over open cavity, the computed acoustic tones match closely with the empirical formulation as suggested by Rossieter. Subsonic and transonic computations were performed. It is shown that at high Mach numbers, the flow field has intermittent occurrence of shock structures. The adaptive multi-time stepping algorithm yielded a speed-up of 5.2 when applied to the cavity tones problem, while the results were almost identical to that of the original solution obtained using the plain Runge Kutta time stepping algorithm. The numerical methods were also applied to the acoustic tone generation on a reed instrument. The instability of the jet impinging on a reed edge and the generation of complex vortex interactions are demonstrated. The feed-back mechanism involved in the reed instrument is clearly demonstrated by analyzing the circulation of vortex structures within the reed instrument. The frequencies of the tones

extracted from the computed solution agree very well with the experiments conducted by Brown et. al.[3].

8.2 Future Work

The work represented in the present thesis can be further extended in several directions:

8.2.1 Improvements in ExRi approximation in multi-dimensions

While the 1D ExRi formulation is straight forward and simple, the 2D formulation of the Riemann correction $\mathcal{F}$ involves complications and result in a stringent stability condition. Different methods of approximation of $\mathcal{F}$ can be considered. Further improvements can be made by considering a higher order Riemann solver [102] at the interface and utilizing the resulting solution to construct a more accurate representation of the approximate flux. It will be interesting to study the influence of the ExRi method on such schemes. In addition to the original ExRi scheme, the viscous ExRi formulation can also be applied to a more computation intensive problem such as LES and DES. The savings in terms of computational cost can be significant in such problems.

8.2.2 Application of multi-time stepping algorithm

The MTS algorithm can theoretically support different spatial and temporal schemes between mesh regions. Currently, only the MTS algorithm is applied only on a standard RK-DG method and the ExRi method. The MTS algorithm can be used to construct semi-implicit schemes where each mesh region is marched using an implicit scheme with different time step sizes. The algorithm can also be used to formulate implicit-explicit methods by selectively using implicit methods at stiff regions. In future, the algorithm can also be extended to 3D viscous flow computations involving shock capturing.

8.2.3 Adaptive WENO formulation

The WENO formulation proposed in this work can be used for applications such as detonations where resolution of the shock structure is more critical. Since the scheme can be applied to unstructured grids and does not require any stencil computation, the proposed WENO scheme is a suitable candidate for adaptive refinements and schemes involving moving mesh problems.

Bibliography

[1] Zhiliang Xu, Yingjie Liu, and Chi-Wang Shu. Hierarchical reconstruction for discontinuous Galerkin methods on unstructured grids with a WENO-type linear reconstruction and partial neighboring cells. *Journal of Computational Physics*, 228(6):2194 – 2212, 2009.

[2] V.A. Titarev and E.F. Toro. Ader schemes for three-dimensional non-linear hyperbolic systems. *Journal of Computational Physics*, 204:715–736, April 2005.

[3] G Burniston Brown. The vortex motion causing edge tones. *Proceedings of the Physical Society*, 49(5):493, 1937.

[4] T.-H. LÃª, J.-M. Le Gouez, and E. Garnier. High accuracy flow simulations: Advances and challenges for future needs in aeronautics. *Computers & Fluids*, In Press, Corrected Proof:–, 2010.

[5] M. Meinke, W. Schröder, E. Krause, and Th. Rister. A comparison of second- and sixth-order methods for large-eddy simulations. *Computers & Fluids*, 31(4-7):695 – 718, 2002.

[6] Santhanam Nagarajan, Sanjiva K. Lele, and Joel H. Ferziger. A robust high-order compact method for large eddy simulation. *Journal of Computational Physics*, 191(2):392 – 419, 2003.

[7] S. C. Reddy and J. A. C. Weideman. The accuracy of the chebyshev differ-

encing method for analytic functions. *SIAM Journal on Numerical Analysis*, 42(5):2176–2187, 2005.

[8] S. E. Sherer and M. R. Visbal. Multi-resolution implicit large eddy simulations using a high-order overset-grid approach. *International Journal for Numerical Methods in Fluids*, 55(5):455 – 482, 2007.

[9] Ben Thornber, Andrew Mosedale, and Dimitris Drikakis. On the implicit large eddy simulations of homogeneous decaying turbulence. *Journal of Computational Physics*, 226(2):1902 – 1929, 2007.

[10] Dieter Fauconnier, Chris De Langhe, and Erik Dick. A family of dynamic finite difference schemes for large-eddy simulation. *Journal of Computational Physics*, 228(6):1830 – 1861, 2009.

[11] M. Parsani, G. Ghorbaniasl, C. Lacor, and E. Turkel. An implicit high-order spectral difference approach for large eddy simulation. *Journal of Computational Physics*, 229(14):5373 – 5393, 2010.

[12] Roland Bouffanais. Advances and challenges of applied large-eddy simulation. *Computers & Fluids*, 39(5):735 – 738, 2010.

[13] MacCormack R.W. The effect of viscosity in hypervelocity impact cratering. *AIAA Paper*, pages 69 – 354, 1969.

[14] E.M. Murman and J.D. Cole. Calculation of plane steady transonic flows. *AIAA Journal*, 9:114 – 121, 1971.

[15] Schmidt W. Jameson A. and Turkel. Numerical solutions of the euler equations by finite volume methods with Runge-Kutta time stepping schemes. *American Institute of Aeronautics and Astronautics*, 1981.

[16] Gunn J.E. Douglas Jr., J. A general formulation of alternating direction methods - part i. parabolic and hyperbolic problems. *Numerische Mathematik*, 6(1):428–453, 1964. cited By (since 1996) 129.

[17] W. Roger Briley. A numerical study of laminar separation bubbles using the navier-stokes equations. *Journal of Fluid Mechanics*, 47(04):713–736, 1971.

[18] Antony Jameson. Time Dependent Calculations Using Multigrid, with Applications to Unsteady Flows Past Airfoils and Wings. In *Proc. 10th Comp. Fluid Dyn. Conf., Honolulu, HI, USA, June 24-26, 1991, AIAA-Paper 91-1596*, 1991.

[19] S.K. Godunov. A difference scheme for numerical solution of discontinuous solution of hydrodynamic equations. *Math. Sbornik*, 47:271 – 306, 1959.

[20] Stanley Osher and Sukumar Chakravarthy. Upwind schemes and boundary conditions with applications to euler equations in general geometries. *Journal of Computational Physics*, 50(3):447 – 481, 1983.

[21] P. L. Roe. Approximate riemann solvers, parameter vectors, and difference schemes. *Journal of Computational Physics*, 43(2):357 – 372, 1981.

[22] Stuart E. Rogers and Dochan Kwak. An upwind differencing scheme for the incompressible navier-strokes equations. *Applied Numerical Mathematics*, 8(1):43 – 64, 1991.

[23] Mavriplis D. Jameson, A. Finite volume solution of the two-dimensional euler equations on a regular triangular mesh. *AIAA journal*, 24(4):611–618, 1986. cited By (since 1996) 81.

[24] Peraire J. Peiro J. Hassan O. Morgan, K. The computation of three-dimensional flows using unstructured grids. *Computer Methods in Applied Mechanics and Engineering*, 87(2-3):335–352, 1991. cited By (since 1996) 20.

[25] Peric M. Demirdzic, I. Finite volume method for prediction of fluid flow in arbitrarily shaped domains with moving boundaries. *International Journal for Numerical Methods in Fluids*, 10(7):771–790, 1990. cited By (since 1996) 156.

[26] Roger C. Strawn and Timothy J. Barth. A finite-volume euler solver for computing rotary-wing aerodynamics on unstructured meshes. *Journal of the American Helicopter Society*, 38(2):61–67, 1993.

[27] W.Kyle Anderson. A grid generation and flow solution method for the euler equations on unstructured grids. *Journal of Computational Physics*, 110(1):23 – 38, 1994.

[28] Gaitonde D. Tavares T.S. Aftosmis, M. Behavior of linear reconstruction techniques on unstructured meshes. *AIAA Journal*, 33(11):2038–2049, 1995. cited By (since 1996) 31.

[29] Ami Harten. On a class of high resolution total-variation-stable finite-difference schemes. *SIAM Journal on Numerical Analysis*, 21(1):1–23, 1984.

[30] Sanjiva K. Lele. Compact finite difference schemes with spectral-like resolution. *Journal of Computational Physics*, 103(1):16–42, November 1992.

[31] Gary A. Sod. A survey of several finite difference methods for systems of nonlinear hyperbolic conservation laws. *Journal of Computational Physics*, 27(1):1 – 31, 1978.

[32] Christopher K. W. Tam and Jay C. Webb. Dispersion-relation-preserving finite difference schemes for computational acoustics. *Journal of Computational Physics*, 107(2):262 – 281, 1993.

[33] Olivier Desjardins, Guillaume Blanquart, Guillaume Balarac, and Heinz Pitsch. High order conservative finite difference scheme for variable density low mach number turbulent flows. *Journal of Computational Physics*, 227(15):7125 – 7159, 2008.

[34] Scott E. Sherer and James N. Scott. High-order compact finite-difference methods on general overset grids. *Journal of Computational Physics*, 210(2):459 – 496, 2005.

[35] Wei Liao, Jinsheng Cai, and Her Mann Tsai. A multigrid overset grid flow solver with implicit hole cutting method. *Computer Methods in Applied Mechanics and Engineering*, 196(9-12):1701 – 1715, 2007.

[36] Karol Z. Korczak and Anthony T. Patera. An isoparametric spectral element method for solution of the navier-stokes equations in complex geometry. *Journal of Computational Physics*, 62(2):361 – 382, 1986.

[37] A.T. Patera. A spectral element method for fluid dynamics: Laminar flow in a channel expansion. *Journal of Computational Physics*, 54(3):468–488, 1984. cited By (since 1996) 475.

[38] D. Gottlieb and J. S. Hesthaven. Spectral methods for hyperbolic problems. *Journal of Computational and Applied Mathematics*, 128(1-2):83 – 131, 2001.

[39] David Sidilkover and George Em Karniadakis. Non-oscillatory spectral element chebyshev method for shock wave calculations. *Journal of Computational Physics*, 107(1):10 – 22, 1993.

[40] Carl Ollivier-Gooch and Michael Van Altena. A high-order-accurate unstructured mesh finite-volume scheme for the advection-diffusion equation. *Journal of Computational Physics*, 181(2):729–752, September 2002.

[41] Michael Dumbser and Martin Kaser. Arbitrary higher order non-oscillatory finite volume schemes on unstructured meshes for linear hyperbolic systems. *Journal of Computational Physics*, 221:693–723, 2007.

[42] Luis Cueto-Felgueroso, Ignasi Colominas, XesÃºs Nogueira, FermÃn Navarrina, and Manuel Casteleiro. Finite volume solvers and moving least-squares approximations for the compressible navier-stokes equations on unstructured grids. *Computer Methods in Applied Mechanics and Engineering*, 196(45-48):4712 – 4736, 2007.

[43] X. Nogueira, I. Colominas, L. Cueto-Felgueroso, S. Khelladi, F. Navarrina, and M. Casteleiro. Resolution of computational aeroacoustics problems on

unstructured grids with a higher-order finite volume scheme. *Journal of Computational and Applied Mathematics*, 234(7):2089 – 2097, 2010. Fourth International Conference on Advanced COmputational Methods in ENgineering (ACOMEN 2008).

[44] Bernardo Cockburn and Chi-Wang Shu. The runge-kutta discontinuous galerkin method for conservation laws v: Multidimensional systems. *Journal of Computational Physics*, 141(2):199 – 224, 1998.

[45] Bernardo Cockburn, Fengyan Li, and Chi-Wang Shu. Locally divergence-free discontinuous Galerkin methods for the Maxwell equations. *Journal of Computational Physics*, 194(2):588 – 610, 2004.

[46] Yen Liu, Marcel Vinokur, and Z.J. Wang. Spectral difference method for unstructured grids i: Basic formulation. *Journal of Computational Physics*, 216(2):780–801, 2006.

[47] Z. J. Wang. Spectral (finite) volume method for conservation laws on unstructured grids. basic formulation. *Journal of Computational Physics*, 178(1):210–251, 2002.

[48] Emmanuil H. Georgoulis, Edward Hall, and Paul Houston. Discontinuous galerkin methods on hp-anisotropic meshes ii: a posteriori error analysis and adaptivity. *Applied Numerical Mathematics*, 59(9):2179 – 2194, 2009. Second Chilean Workshop on Numerical Analysis of Partial Differential Equations (WONAPDE 2007).

[49] Ethan J. Kubatko, Shintaro Bunya, Clint Dawson, and Joannes J. Westerink. Dynamic p-adaptive runge-kutta discontinuous galerkin methods for the shallow water equations. *Computer Methods in Applied Mechanics and Engineering*, 198(21-26):1766 – 1774, 2009. Advances in Simulation-Based Engineering Sciences - Honoring J. Tinsley Oden.

[50] Kim S. Bey and J. Tinsley Oden. hp-version discontinuous galerkin methods for hyperbolic conservation laws. *Computer Methods in Applied Mechanics and Engineering*, 133(3-4):259 – 286, 1996.

[51] Alex Kanevsky, Mark H. Carpenter, David Gottlieb, and Jan S. Hesthaven. Application of implicit-explicit high order Runge-Kutta methods to discontinuous-Galerkin schemes. *Journal of Computational Physics*, 225(2):1753 – 1781, 2007.

[52] Ralf Wolke and Oswald Knoth. Implicit-explicit runge-kutta methods applied to atmospheric chemistry-transport modelling. *Environmental Modelling and Software*, 15(6-7):711 – 719, 2000.

[53] R. Pardeshi, V.R. Voller, A.K. Singh, and P. Dutta. An explicit-implicit time stepping scheme for solidification models. *International Journal of Heat and Mass Transfer*, 51(13-14):3399 – 3409, 2008.

[54] Gabor Toth, Darren L. De Zeeuw, Tamas I. Gombosi, and Kenneth G. Powell. A parallel explicit/implicit time stepping scheme on block-adaptive grids. *Journal of Computational Physics*, 217(2):722 – 758, 2006.

[55] Margarete O. Domingues, SÃŽnia M. Gomes, Olivier Roussel, and Kai Schneider. An adaptive multiresolution scheme with local time stepping for evolutionary pdes. *Journal of Computational Physics*, 227(8):3758 – 3780, 2008.

[56] Abdelaaziz Ezziani and Patrick Joly. Local time stepping and discontinuous galerkin methods for symmetric first order hyperbolic systems. *Journal of Computational and Applied Mathematics*, 234(6):1886 – 1895, 2010. Eighth International Conference on Mathematical and Numerical Aspects of Waves (Waves 2007).

[57] Frieder Lörcher, Gregor Gassner, and Claus-Dieter Munz. An explicit discontinuous galerkin scheme with local time-stepping for general unsteady

diffusion equations. *Journal of Computational Physics*, 227(11):5649 – 5670, 2008.

[58] Babak Hejazialhosseini, Diego Rossinelli, Michael Bergdorf, and Petros Koumoutsakos. High order finite volume methods on wavelet-adapted grids with local time-stepping on multicore architectures for the simulation of shock-bubble interactions. *Journal of Computational Physics*, 229(22):8364 – 8383, 2010.

[59] Li Liu, Xiaodong Li, and Fang Q. Hu. Nonuniform time-step runge-kutta discontinuous galerkin method for computational aeroacoustics. *Journal of Computational Physics*, 229(19):6874 – 6897, 2010.

[60] N. M. Maurits, H. van der Ven, and A. E. P. Veldman. Explicit multi-time stepping methods for convection-dominated flow problems. *Computer Methods in Applied Mechanics and Engineering*, 157(1-2):133 – 150, 1998.

[61] Li Liu, Xiaodong Li, and Fang Q. Hu. Nonuniform time-step runge-kutta discontinuous galerkin method for computational aeroacoustics. *Journal of Computational Physics*, 229(19):6874 – 6897, 2010.

[62] R. Löhner, K. Morgan, and O.C. Zienkiewicz. The use of domain splitting with an explicit hyperbolic solver. *Computer Methods in Applied Mechanics and Engineering*, 45(1-3):313 – 329, 1984.

[63] H. van der Ven, B. E. Niemann-Tuitman, and A. E. P. Veldman. An explicit multi-time-stepping algorithm for aerodynamic flows. *Journal of Computational and Applied Mathematics*, 82(1-2):423 – 431, 1997. 7th ICCAM 96 Congress.

[64] C. Chauviere, J.S. Hesthaven, A. Kanevsky, and T. Warburton. High-order localized time integration for grid-induced stiffness. In K.J. Bathe, editor, *Computational Fluid and Solid Mechanics 2003*, pages 1883 – 1886. Elsevier Science Ltd, Oxford, 2003.

[65] Gibbs J.W. Fourier series. *Nature*, 59(200 and 606), 1899.

[66] Bernie D. Shizgal and Jae-Hun Jung. Towards the resolution of the gibbs phenomena. *Journal of Computational and Applied Mathematics*, 161(1):41 – 65, 2003.

[67] J. VonNeumann and R. D. Richtmyer. A method for the numerical calculation of hydrodynamic shocks. *Journal of Applied Physics*, 21:232–238, 1950.

[68] Phillip Colella. A direct eulerian muscl scheme for gas dynamics. *SIAM Journal on Scientific and Statistical Computing*, 6(1):104–117, 1985.

[69] Wang Chi-Shu. Essentially non-oscillatory and weighted essentially non-oscillatory schemes for hyperbolic conservation laws. Technical report, Institute for Computer Applications in Science and Engineering (ICASE), 1997.

[70] Changqing Hu and Wang Chi-Shu. Weighted essentially non-oscillatory schemes on triangular meshes. Technical report, Institute for Computer Applications in Science and Engineering (ICASE), 1998.

[71] Jun Zhu, Jianxian Qiu, Chi-Wang Shu, and Michael Dumbser. Runge-Kutta discontinuous Galerkin method using WENO limiters II: Unstructured meshes. *Journal of Computational Physics*, 227(9):4330 – 4353, 2008.

[72] Hong Luo, Joseph D. Baum, and Rainald Lohner. A hermite WENO-based limiter for discontinuous Galerkin method on unstructured grids. *Journal of Computational Physics*, 225(1):686 – 713, 2007.

[73] Bernardo Cockburn, San-Yih Lin, and Chi-Wang Shu. Tvb runge-kutta local projection discontinuous galerkin finite element method for conservation laws iii: One-dimensional systems. *Journal of Computational Physics*, 84(1):90 – 113, 1989.

[74] Hongwei Liu and Kun Xu. A runge-kutta discontinuous galerkin method

for viscous flow equations. *Journal of Computational Physics*, 224(2):1223 – 1242, 2007.

[75] Sigal Gottlieb, Chi-Wang Shu, and Eitan Tadmor. Strong stability-preserving high-order time discretization methods. *SIAM Review*, 43(1):89–112, 2001.

[76] Ethan J. Kubatko, Joannes J. Westerink, and Clint Dawson. Semi discrete discontinuous Galerkin methods and stage-exceeding-order, strong-stability-preserving Runge-Kutta time discretizations. *Journal of Computational Physics*, 222(2):832 – 848, 2007.

[77] Jianxian Qiu, Boo Cheong Khoo, and Chi-Wang Shu. A numerical study for the performance of the runge-kutta discontinuous galerkin method based on different numerical fluxes. *Journal of Computational Physics*, 212(2):540 – 565, 2006.

[78] E. F. Toro. *Riemann Solvers and Numerical Methods for Fluid Dynamics - A Practical Introduction*. Springer, 1999.

[79] Christophe Geuzaine and Jean-FranÃ§ois Remacle. Gmsh: A 3-d finite element mesh generator with built-in pre- and post-processing facilities. *International Journal for Numerical Methods in Engineering*, 79:1309 – 1331, 2009.

[80] Shyam Sundar .D and Yeo K. S. Riemann solvers on extended domains for higher order schemes. *47th AIAA Aerospace Sciences Meeting including The New Horizons Forum and Aerospace Exposition*, 2009.

[81] H. Gao Z. J. Wang. A unifying lifting collocation penalty formulation including the discontinuous galerkin, spectral volume/difference methods for conservation laws on mixed grids. *Journal of Computational Physics*, 228:8161–8186, 2009.

[82] Kris Van den Abeele, Tim Broeckhoven, and Chris Lacor. Dispersion and

dissipation properties of the 1d spectral volume method and application to a p-multigrid algorithm. *Journal of Computational Physics*, 224:616–636, 2007.

[83] Fang Q. Hu, M. Y. Hussaini, and Patrick Rasetarinera. An analysis of the discontinuous Galerkin method for wave propagation problems. *Journal of Computational Physics*, 151(2):921 – 946, 1999.

[84] B.T. Helenbrook and H.L. Atkins. Application of p-multigrid to discontinuous Galerkin formulations of the Poisson equation. *AIAA J.*, 44(3), 2005.

[85] B. Cockburn, M. Luskin, C.W. Shu, and E. Suli. Post-processing of Galerkin methods for hyperbolic problems. *Discontinuous Galerkin Methods (Cockburn et al. eds)*, pages 291–300, 2000.

[86] P. Batten, N. Clarke, C. Lambert, and D. M. Causon. On the choice of wavespeeds for the HLLC riemann solver. *SIAM Journal on Scientific Computing*, 18(6):1553–1570, 1997.

[87] Douglas N. Arnold, Franco Brezzi, Bernardo Cockburn, and L. Donatella Marini. Unified analysis of discontinuous Galerkin methods for elliptic problems. *SIAM Journal on Numerical Analysis*, 39(5):1749–1779, 2002.

[88] F. Bassi and S. Rebay. A high-order accurate discontinuous finite element method for the numerical solution of the compressible navier-stokes equations. *Journal of Computational Physics*, 131(2):267 – 279, 1997.

[89] Bernardo Cockburn and Chi-Wang Shu. The local discontinuous Galerkin method for time-dependent convection-diffusion systems. *SIAM Journal on Numerical Analysis*, 35(6):2440–2463, 1998.

[90] Fang Q. Hu, X.D. Li, and D.K. Lin. Absorbing boundary conditions for nonlinear euler and navier-stokes equations based on the perfectly matched layer technique. *Journal of Computational Physics*, 227(9):4398 – 4424, 2008.

[91] Faiz Ahmad and Wafaa H. Al-Barakati. An approximate analytic solution of the blasius problem. *Communications in Nonlinear Science and Numerical Simulation*, 14(4):1021 – 1024, 2009.

[92] A. Burbeau and P. Sagaut. Simulation of a viscous compressible flow past a circular cylinder with high-order discontinuous Galerkin methods. *Computers & Fluids*, 31(8):867 – 889, 2002.

[93] H. C. Yee, N. D. Sandham, and M. J. Djomehri. Low-dissipative high-order shock-capturing methods using characteristic-based filters,. *Journal of Computational Physics*, 150(1):199 – 238, 1999.

[94] Sung Don Kim, Bok Jik Lee, Hyoung Jin Lee, and In-Seuck Jeung. Robust hllc riemann solver with weighted average flux scheme for strong shock. *Journal of Computational Physics*, 228(20):7634 – 7642, 2009.

[95] Visit Visualization Tool. https://wci.llnl.gov/codes/visit/home.html.

[96] Miroslav Cada and Manuel Torrilhon. Compact third-order limiter functions for finite volume methods. *Journal of Computational Physics*, 228(11):4118 – 4145, 2009.

[97] Paul Woodward and Phillip Colella. The numerical simulation of two-dimensional fluid flow with strong shocks. *Journal of Computational Physics*, 54(1):115 – 173, 1984.

[98] Irvine I. Glass and J. P. Sislian. *Nonstationary flows and shock waves*. Oxford : Clarendon Press, 1994.

[99] K. Krishnamurty. Acoustic radiation from two-dimensional rectangular cutouts in aerodynamic surfaces. Technical report, 1955.

[100] J. E. Rossiter. Wind-tunnel experiments on the flow over rectangular cavities at subsonic and transonic speeds. *Royal Aircraft Establishcment ARC R&M*, (3438):1394–1414, 1964.

[101] Hiroyuki Hirahara, Masaaki Kawahashi, Maksud Uddin Khan, and Kerry Hourigan. Experimental investigation of fluid dynamic instability in a transonic cavity flow. *Experimental Thermal and Fluid Science*, 31(4):333 – 347, 2007.

[102] V.A. Titarev E.F. Toro. Derivative Riemann solvers for systems of conservation laws and ADER methods. *Journal of Computational Physics*, 212(1):150–165, February 2006.