

Development of a General-Purpose Compressible Flow AnuPravaha Based Solver

Ashwani Assam

A Dissertation Submitted to
Indian Institute of Technology Hyderabad
In Partial Fulfillment of the Requirements for
The Degree of Master of Technology



Department of Mechanical & Aerospace Engineering

July 2014

Declaration

I declare that this written submission represents my ideas in my own words, and where ideas or words of others have been included, I have adequately cited and referenced the original sources. I also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission. I understand that any violation of the above will be a cause for disciplinary action by the Institute and can also evoke penal action from the sources that have thus not been properly cited, or from whom proper permission has not been taken when needed.



(Signature)

(Ashwani Assam)

(ME12M14P000004)

Approval Sheet

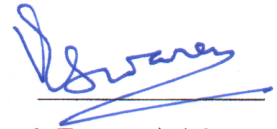
This Thesis entitled Development of a General-Purpose Compressible Flow AnuPravaha Based Solver by Ashwani Assam is approved for the degree of Master of Technology from IIT Hyderabad



(Dr. Vinod Janardhanan) Examiner
Dept. of Chemical Engineering
IITH



(Dr. K. Venkatasubbaiah) Examiner
Dept. of Mechanical & Aerospace Engineering
IITH



(Prof. Vinayak Eswaran) Adviser
Dept. of Mechanical & Aerospace Engineering
IITH

Acknowledgements

I am grateful to my teacher who has taught me how to live a responsible life with his patience and love.

I would like to express my gratitude to my guide Prof. Vinayak Eswaran to allow me the freedom to work and at the same time to keep an eye to see that I do not get drifted from the main objective. I would especially like to thank him for his moral support and calmness, at the time when I have been running on low confidence with the thesis work. His honesty and humility, has always motivated us to work harder and has kept us grounded. And simultaneously, he has been the source of motivation for all our friends. Thus, the work environment, I have got to work has been vibrant and energetic.

I would like to thank Narendra Gajbhiye and Praveen Throvagunta for their valuable suggestion at various point of the thesis. In addition, they have helped us in understanding the *AnuPravaha* solver, which has been the starting point for this thesis. I would also like to thank all my friends who have always maintained an atmosphere of co-operation and friendship. Thus, have made the thesis a joyful experience.

I would like to thank my guide and the entire IIT Hyderabad system for providing us an excellent computational facility to work upon.

My thanks would also go to Dr.Nishanth Dongari, who came at the point in my academic career that has boosted my interest for working in the area of high-speed flow.

At last I would like thank my parents for their support and patience. I know, I have failed to be with them at various moments, they would have liked me to be with them. I would like to thank my younger brother for substituting for my role very nicely as far as possible.

Abstract

With India focussing even more on Aerospace applications , research and development in compressible flow has received a boost in the country. We aim to develop a general-purpose and robust compressible flow solver to help in research in Aerospace problems. The present work aims to make the Euler Solver developed by previous post-graduate students as part of the general purpose AnuPravaha Solver more robust and accurate, and develop it as a stand-alone compressible flow solver. In this work, the solver has been made more general and now incorporates non-reflecting, symmetric and other boundary conditions, taking into the account curvature at the boundary. As a step to move to higher accuracy, the TVD scheme and the upwind-based scheme $AUSM^+ - up$ have been successfully implemented. The latter has been shown to give more accurate, stable and numerical oscillation-free results.

Contents

Declaration	ii
Approval Sheet	iii
Acknowledgements	iv
Abstract	v
1 Introduction	1
1.1 Literature review	2
1.2 Objective of present work	4
2 Mathematical Nature Of Equations And Boundary Conditions	5
2.1 Domain of dependance and zone of influence	6
2.1.1 Scalar Conservation Law	6
2.1.2 System of PDEs	7
2.2 System of partial differential equation in multi-dimensions	9
2.2.1 System of First Order Steady-State PDEs	10
2.2.2 Characteristic and Characteristic Surface in Multi-dimensions	11
2.3 Advantage of Conservation form over the non-conservation form	15
2.4 Boundary Condition Specification in Hyperbolic System	15
2.5 Closure	17
3 Euler Equations, Boundary Conditions and CFL criterion	18
3.1 A short description of Euler Equations	18
3.2 Boundary condition treatment in terms of primitive variables	19
3.2.1 Implementation of Boundary Conditions	20
3.2.2 Inflow BC	21
3.2.3 Outflow BC	22
3.2.4 Wall (or Solid) Boundary	23
3.2.5 Symmetry Boundary Conditions	25
3.3 CFL Condition	25
3.3.1 Physical interpretation	25

3.4	Closure	27
4	Numerical Methodology	28
4.1	Numerical Schemes.	28
4.2	Governing Equations	29
4.3	Discretization of Governing Equation	29
4.4	MacCormack scheme in FVM	33
4.4.1	Artificial Viscosity	34
4.5	TVD Scheme	35
4.5.1	TVD MacCormack Scheme	36
4.6	Advection Upstream Splitting Method (AUSM)	39
4.7	Closure	42
5	Results And Discussion	43
5.1	Shock-tube Problem	43
5.2	Supersonic flow over a wedge	46
5.3	Steady Shock Reflection Problem	50
5.4	External Flow over NACA0012 Airfoil	53
5.4.1	Mach 0.5, Angle of Attack ($\alpha = 0^\circ$)	53
5.4.2	Mach 0.8, Angle of Attack ($\alpha = 1.25^\circ$)	57
5.4.3	Mach 1.2, Angle of Attack ($\alpha = 0^\circ$)	59
5.5	Internal flow in a channel with a circular Bump	62
5.5.1	Subsonic Case	62
5.5.2	Transonic Case	65
5.5.3	3D Case: Flow over re-entry capsule (Mach no. = 5, AOA = 4.66°)	68
6	Conclusions and future work	72
	Appendices	74
	A Artificial Viscosity Formulation	75
	References	77

Chapter 1

Introduction

Compressible flow effects are encountered in numerous engineering applications involving high speed flows, e.g. gas turbines, steam turbines, internal combustion engines, rocket engines, high-speed aerodynamics, high speed propellers, gas pipe flows, etc. Special phenomena such as compression shocks, entropy layers, expansion fans, flow induced noise etc. are of fundamental scientific importance and directly affect the performance and endurance of these engineering applications. Taking this as the motivation, the task of incorporating a density-based solver in the Anupravaha - General purpose CFD solver was undertaken two years ago. This solver, which was initially only for incompressible flows, has been under continuous development since 2004 in the research group of Prof. Vinayak Eswaran. It already has the Pressure-based solver having modules for turbulence, multiphase flows, radiation problems, MHD, Solidifications and Melting, etc. To cater exclusively to aerospace applications and for the ease of management of the solver, we have now tried to separate out the density-based solver in a stand-alone code.

Compressible flows (in contrast to variable density flows) are those where dynamics (i.e pressure) is the dominant factor in density change. Generally, fluid flow is considered to be compressible if the change in density relative to the stagnation density is greater than 5%. Significant compressible effects occur at Mach number of 0.3 and greater.

Compressible flow is divided often into four main flow regimes based on the local Mach number (M) of the fluid flow

- Subsonic flow regime ($M \leq 0.8$)
- Transonic flow regime ($0.8 \leq M \leq 1.2$)
- Supersonic flow regime ($M > 1$)
- Hypersonic flow regime ($M > 5$)

Compressible flow may be treated as either viscous or inviscid. Viscous flows are solved by the Navier-Stokes system of equations and inviscid compressible flows are solved by Euler

equations. The physical behavior of compressible fluid flow is quite different from incompressible fluid flow. The solutions of Euler equation are different (due to their hyperbolic wave-like nature) from the solutions of the elliptic governing equations of incompressible flows. Compressible flow can have discontinuous solutions in certain cases e.g vortex sheets, contact discontinuities or shock waves. So, for compressible flows special attention is required for solution methods which will accurately capture these discontinuities.

In compressible fluid flows, properties are not only transported by the flow, but also by the propagation of waves. This requires the construction of flux interpolations, that take into account that transports can occur in any direction.

Thus, a major difference between solution methods for compressible flow and incompressible flow lies in the boundary conditions implementation. In compressible flows, boundary conditions are imposed based on the characteristic waves coming into the domain boundary, which is very different from the elliptic-type boundary conditions used for incompressible flows.

1.1 Literature review

Hirsch, [1] has discussed the general methodology to analyze the nature of systems of partial differential equations. This systematic procedure to determine the nature of equations and the propagation of their solution is key to the understanding the implementation of boundary conditions. Euler equations are solved in conservative form but require imposition of boundary condition in primitive form; in Chapter 19 [2] discusses the implementation of boundary conditions (both physical and numerical) from characteristic extrapolation for conservative and primitive variables, along with different extrapolation methods. The outflow boundary condition has been scrutinized in the present work in detail after the previous work done by Nikhil, [3] reported unwanted reflections at outflow. This study is more popularly known as the far field boundary condition (FFBC) model for compressible flow. It also permits a substantial reduction of the computational domain, leading to a considerable improvement in the computational efficiency [4]. A theoretical basis of non-reflecting boundary conditions for hyperbolic equations was first established by Kreiss, [5] for initial boundary value problems of multiple-dimension hyperbolic systems. Kreiss examined the well-posedness of the hyperbolic system and emphasized that truncating an infinite domain into a finite domain must be done such that the flow solution is well-posed. Hedstrom, [6] showed that hyperbolic characteristic theory is able to correctly impose non-reflecting boundary conditions for one-dimensional unsteady problems using the eigenvector method. The same year, Engquist et al. [7] introduced a list of non-reflecting boundary conditions of increasing complexity for multi-dimensional flows, the first of which was the one-dimensional condition. Gustafsson, [8], used these boundary conditions to solve the unsteady Euler equations. Non-reflecting boundary conditions are also used to

accelerate convergence of the solution, but can create inaccurate results if the method is not correctly implemented. This is especially evident for Hedstrom's boundary conditions where the converged steady-state is often quite sensitive to initial conditions. A makeshift solution to this problem was proposed in [9], replacing Hedstrom's boundary condition for a subsonic outlet by the constant parameter to enforce convergence to a desired free-stream pressure. We use this method for the outflow condition in the solver. All the boundary conditions have been implemented to incorporate curved surface, based on the book of Blazek, [10] and the work done in [11].

The second focus of this thesis has been the incorporation of High Resolution schemes into the solver. The second volume of [2] discusses almost all basic numerical schemes pertaining to Euler and Navier Stokes equations.

First order schemes avoid numerical oscillations but are diffusive in nature, which causes loss of information in the vicinity of shock and contact discontinuities. Higher order methods create spurious oscillations in the numerical solution in the presence of shocks or contact discontinuities. One set of techniques to avoid the latter phenomenon follow the method suggested by [12] and are popularly known as ENO ("Essentially non-oscillatory") schemes. Such methods use special methods to calculate flux across cell-faces to prevent numerical oscillations. In his text, [13] discusses these methods from first principles after discussing the basic techniques - including the Lax-Friedrichs method, the Lax-Wendroff method, MacCormack's method and Godunov's method.

Nowadays, to solve the Euler equations it is usually preferred to use special higher order methods, that avoid the spurious oscillations usually found in the the classical central methods, to get oscillation-free solutions that accurately represent shock and contact discontinuities. A systematic analysis of conditions required by these schemes was developed by Godunov [2], who first introduced the concept of monotonicity. For the solution of non-linear equations, the more general concept of bounded total variation of solutions was introduced by Harten [12]. This led to criteria, popularly known as the TVD ("Total Variation Diminishing") condition, to ensure unwanted oscillations are not generated by the numerical scheme.

Van Leer in his works [14] [15] [16] presented a high resolution scheme satisfying the Entropy Condition, Monotonicity and TVD, and a general frame-work for the development of such schemes.

There are a large number of schemes presently known for the solution of compressible flow. The overall picture and classification of these schemes can be found in [17]. The compressible flow codes have seen development of two classes of schemes central and up-wind based. The central scheme which is more suited for turbo-machinery applications has been incorporated in various solvers such as *rhoCentralFoam* a solver in the OpenFOAM that incorporates a new high resolution central scheme based on the work of [18]. The MacCormack method, a central scheme, has been used in this work.

However, for better stability and accuracy, we have successfully tried to implement the upwind-based scheme AUSM+ of [19] with second order accuracy. This scheme was first presented by Liou in 1993 as Advection Upstream Splitting Method (AUSM) [20]. The scheme has went through several revision since then, which can be found in [21].

1.2 Objective of present work

- Optimizing and increasing the robustness of the existing Euler Solver in Anupravaha.
- Incorporating Far-Field Boundary Conditions, and expanding the boundary conditions for curved surfaces.
- To incorporate robust High resolution Scheme into the solver.
- To validate this solver for different regimes of flow i.e. in subsonic, transonic and in supersonic regime and for the 3D cases.
- To separate the density-based solver to cater to aerospace applications exclusively.

Chapter 2

Mathematical Nature Of Equations And Boundary Conditions

Let us consider the system of governing differential equations for inviscid compressible flow in primitive variable form, also known as the Euler equations, describing the conservation of mass, momentum and energy:

$$\begin{aligned}u \frac{\partial \rho}{\partial x} + v \frac{\partial \rho}{\partial y} + \rho \frac{\partial u}{\partial x} + \rho \frac{\partial v}{\partial y} &= 0 \\u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} &= -\frac{1}{\rho} \frac{\partial p}{\partial x} \\u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y} &= -\frac{1}{\rho} \frac{\partial p}{\partial y}\end{aligned}\tag{2.1}$$

However, for the special case of irrotational compressible flow we can equivalently write the potential flow equation as

$$\left(1 - \frac{u^2}{c^2}\right) \frac{\partial^2 \phi}{\partial x^2} - \frac{2uv}{c^2} \frac{\partial^2 \phi}{\partial x \partial y} + \left(1 - \frac{v^2}{c^2}\right) \frac{\partial^2 \phi}{\partial y^2} = 0\tag{2.2}$$

where ϕ is the potential function defined by,

$$u = \frac{\partial \phi}{\partial x} \quad ; \quad v = \frac{\partial \phi}{\partial y}\tag{2.3}$$

These two sets of PDE's describe exactly the same physics for steady irrotational flow. But looking at the forms of the equations, they appear to be quite different, with the first set of equations (2.1) seeming to be convection dominated, whereas the second Eq. (2.2) seemingly diffusion dominated. Convection and diffusion are two important phenomena in fluid mechanics, and relate to very different physical behavior. But as we know, both

Eq. (2.1) and Eq. (2.2) describe the same physics.

So, it is clear from the above discussion that one cannot predict the behavior of the solutions of differential equation by just looking at the form of the equations. It is not the form of the governing equation which decides the behavior of solutions; rather, it is the eigenvalue matrix (discussed in the next section) which determines this, and provides a tool to analyze the nature of governing equations, independent of physical form, to tell us something about the nature of their solutions. Thus, in these chapter we try to learn these tools and others to develop our understanding of hyperbolic equations.

2.1 Domain of dependance and zone of influence

2.1.1 Scalar Conservation Law

To understand the propagation of information in the solutions of governing equations, consider a single equation describing a conservation law:

$$\frac{\partial w}{\partial t} + \frac{\partial F}{\partial x} = 0 \quad w(x, 0) = w_0(x) \quad (2.4)$$

$$\frac{\partial w}{\partial t} + \frac{dF}{dw} \frac{\partial w}{\partial x} = 0 \quad (2.5)$$

Introducing wave speed as, $\frac{dF}{dw} = \lambda$ in the above Eq. (2.5) can also be written as,

$$\frac{\partial w}{\partial t} + \lambda \frac{\partial w}{\partial x} = 0 \quad (2.6)$$

An analytical solution to above equation exists and can be found using the method of separation of variables, and can be written as,

$$w(x, t) = w_0(x - \lambda t) \quad (2.7)$$

The analytical solution of this equations for a constant and positive wave-speed shows that solution is constant along the line satisfying satisfying the condition $\frac{dx}{dt} = \lambda$; this line is called the *characteristic*. For linear PDEs the solution is constant along the characteristic, while for non-linear PDEs the solution may vary along characteristic but it will be purely a function of the curvilinear co-ordinate describing the characteristic. Now from Eq. (2.7), the solution at a point for the hyperbolic systems is dependent on the previous solutions at points lying on the same characteristic, and ultimately on the initial condition at that characteristic at $t=0$. So, for a single first-order hyperbolic equation all the points on a particular characteristic line form the domain of dependence and zone of influence of any point on that characteristic.

2.1.2 System of PDEs

Now, moving one step forward, consider a general 1-D unsteady system of first-order PDEs,

$$\frac{\partial w_i}{\partial t} + \frac{\partial F_i}{\partial x} = 0 \quad (2.8)$$

where w_i is a vector containing conservative variables

F_i is a vector containing flux associated with w_i

The Eq. (2.8) can be written as,

$$\frac{\partial w_i}{\partial t} + [A] \frac{\partial w_i}{\partial x} = 0 \quad (2.9)$$

where $[A]$ is called the *Jacobian* (matrix).

The Jacobian in Eq. (2.9) determines the behavior of the solutions, based on the nature of its eigenvalues. Diagonalizing the Jacobian in Eq. (2.9), we can rewrite the equation as:

$$\frac{\partial w_i}{\partial t} + [Q_r] [\lambda_i] [Q_l] \frac{\partial w_i}{\partial x} = 0 \quad (2.10)$$

where Q_l , Q_r are the left and right eigenvector matrices explained through the following relations.

For any diagonalizable matrix A , Q_r is a matrix whose columns \mathbf{r}_i are *right characteristics vectors* or *right eigenvectors* of A , and Q_l is a matrix whose rows \mathbf{l}_i are *left characteristics vectors* or *left eigenvectors* of A .

The right characteristic vectors are defined as follows:

$$A\mathbf{r}_i = \lambda_i\mathbf{r}_i \quad (2.11)$$

While less familiar, left characteristic vectors are defined in almost the same way as right characteristic vectors, except that left characteristic vectors multiply A on the left than on the right.

$$\mathbf{l}_i A = \lambda_i \mathbf{l}_i \quad (2.12)$$

Also,

$$[\lambda] = [Q_l][A][Q_r]$$

$$[Q_l] = [Q_r]^{-1}$$

As mentioned earlier, the nature of λ_i determines the behavior of the solution. If a full set of real non-zero eigenvalues exists then the system of equations is called *Hyperbolic*, if a

full set of real eigenvalues does not exist then the system is called *Parabolic* and if some of eigenvalues are complex then the system is called *Elliptic*.

Let us now consider a hyperbolic system. By multiplying Eq. (2.10) by $[Q_l]$ and rewriting equation(2.10) we get

$$[Q_l] \frac{\partial w_i}{\partial t} + [Q_l][Q_r][\lambda_i][Q_l] \frac{\partial w_i}{\partial x} = 0 \quad (2.13)$$

As we know,

$$[Q_l] = [Q_r]^{-1} \quad (2.14)$$

$$[Q_l] \frac{\partial w_i}{\partial t} + [\lambda_i][Q_l] \frac{\partial w_i}{\partial x} = 0 \quad (2.15)$$

Introducing new set of variables as,

$$\delta v_i = [Q_l] \delta w_i \quad (2.16)$$

Eq. (2.15) can be written as,

$$\frac{\partial v_i}{\partial t} + [\lambda_i] \frac{\partial v_i}{\partial x} = 0 \quad (2.17)$$

where $[\lambda_i]$ is a diagonal matrix of eigenvalues.

Now, note the simplicity of Eq. (2.17) in comparison to Eq. (2.8); unlike in the latter equation, the equations in system (2.17) are decoupled, i.e., the solution of any one of them is independent of the solution of others. Thus the component equations of (2.17) can thus be solved separately and easily. Then, by inverting Eq. (2.16) we can obtain the solution to system of Eq. (2.8).

As shown in the figure for a system of 3 equations, the information flowing along corresponding characteristic lines passing through a point P determine the solution at P by the superposition of the characteristic information. From this, it is not difficult to show that solution at point P depends only on the solution in region APCBA and it has nothing to do with the solution outside this region. Hence this region is termed as the *domain of dependence*. By extending the characteristic lines beyond point P, we can say that solution at this point is going to affect the solution in the region FPDE, hence this region is termed the *zone of influence*. The point to note here is that a numerical scheme determining the solution (for a time-step) at point P must include only points from the domain of dependence to capture the solution correctly — failing to do this causes serious issues, mostly resulting in the blowing up of the solution. This condition is called *Courant-Friedrichs-Lewy* condition and will be discussed in the next chapter.

Depending upon the nature (positive or negative valued) of the eigenvalues, character-

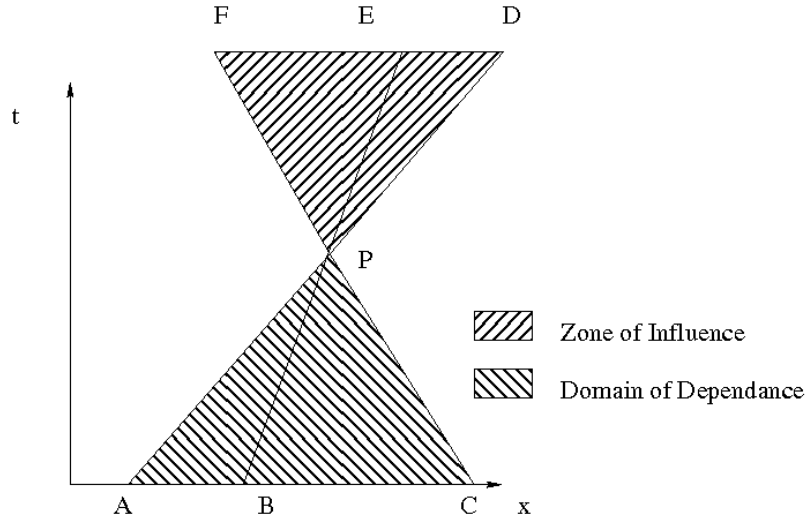


Figure 2.1: Characteristics in 3 equation system

istic information may flow from left to right or right to left (Fig. 2.1). So, the problem to well-posed boundary conditions must be handled carefully. We will discuss this in the next few sections.

Further discussion over characteristics can be found in the ([3], Section 2). Here, in the following section we shall present the method to find the nature of the governing equations for the multi-dimensional cases.

2.2 System of partial differential equation in multi-dimensions

By direct matrix manipulation we can find out the nature of governing equation for 1-D case. We discuss here the general method to find out the nature of governing equations applicable for all the cases. Governing equations defining conservation laws in multi-dimensions can be written as:

Steady state:

$$[A] \frac{\partial w_i}{\partial x} + [B] \frac{\partial w_i}{\partial y} + [C] \frac{\partial w_i}{\partial z} = 0 \quad (2.18)$$

Unsteady state:

$$[I] \frac{\partial w_i}{\partial t} + [A] \frac{\partial w_i}{\partial x} + [B] \frac{\partial w_i}{\partial y} + [C] \frac{\partial w_i}{\partial z} = 0 \quad (2.19)$$

Diagonalizing the Jacobian matrices, we get:

$$I \frac{\partial w_i}{\partial t} + [Q_a][\lambda_a][Q_a]^{-1} \frac{\partial w_i}{\partial x} + [Q_b][\lambda_b][Q_b]^{-1} \frac{\partial w_i}{\partial y} + [Q_c][\lambda_c][Q_c]^{-1} \frac{\partial w_i}{\partial z} = 0 \quad (2.20)$$

Since, $[Q_a] \neq [Q_b] \neq [Q_c]$ we cannot replace, as before the conservative variables with another set such that the equations in the system (2.18) or (2.19) gets decoupled. Therefore a deeper analysis needs to be done to get eigenvalues and characteristic variables.

2.2.1 System of First Order Steady-State PDEs

The following steps define the procedure to identify the nature of a mathematical system. These are taken from [1]:

Step 1: Write the system of PDEs describing the mathematical model as a system of first order PDEs

Suppose we have n unknown variables w_j , in $(m + 1)$ -dimensional space x_J , we can group all the variables w_j in an $(n \times 1)$ vector column w and write the system of first order PDEs under the general form:

$$\sum_j A_j \frac{\partial w}{\partial x_j} = T \quad j = 1, 2, 3, \dots, m + 1$$

$$w = \begin{pmatrix} w_1 \\ w_2 \\ w_3 \\ \cdot \\ \cdot \\ \cdot \\ w_n \end{pmatrix} \quad (2.21)$$

where A_j are $(n \times n)$ matrices and T is a column vector of the non-homogeneous source terms. The matrices A_j and T can depend on x_j and w , but not on the derivatives of w .

Step 2: Consider a plane wave solution of amplitude \hat{U} in the space of the independent variables x with components $x_j (j = 1, \dots, m + 1)$, defined by

$$w = \hat{U} e^{i(\vec{n} \cdot \vec{x})} \quad (2.22)$$

where $i = \sqrt{-1}$, \vec{n} is a vector in the m -dimensional space of the independent variables x_j and \hat{U} is an $(n \times 1)$ column vector.

Step 3: Introduce this solution in the homogeneous part of the system (2.21) and find the values of n satisfying the resulting equation.

The homogenous part of Eq. (2.21) is written as

$$\Sigma_j A_j \frac{\partial w}{\partial x_j} = 0 \quad j = 1, 2, 3 \dots m + 1 \quad (2.23)$$

and the function (2.22) is a solution of this system of equations if the homogeneous algebraic system of equations:

$$[\Sigma_j A_j n_j] \hat{U} = 0 \quad (2.24)$$

has non-vanishing solutions for the amplitude \hat{U} . This will be the case if and only if the determinant of the matrix $\Sigma_j A_j n_j$ vanishes.

Step 4: Find the n solutions of the equation

$$\det [\Sigma_j A_j n_j] \quad (2.25)$$

Eq. (2.25) defines a condition on the normals \vec{n} . This equation can have at most n solutions, and for each of these normals n_i , the system (2.25) has a non-trivial solution.

The system is said to be hyperbolic if all the n characteristic normals n_i are real and if the solutions of the n associated systems of equations (2.25) are linearly independent. If all the characteristics are complex, the system is said to be elliptic. If some are real and other complex the system is classed as hybrid. If the matrix $\Sigma_j [A_j n_j]$ is not of rank n , i.e. there are less than n real characteristic normals then the system is said to be parabolic.

The last case will occur, for instance, when at least one of the variables, say w_1 has derivatives with respect to one coordinate, say x_1 , missing. This implies that the components $A_1 = 0$ for all equations i .

2.2.2 Characteristic and Characteristic Surface in Multi-dimensions

Parabolic and hyperbolic equations play an important role in CFD, due to their association to diffusion and convection phenomena. They are recognized by the existence of real characteristic normals, solutions of Eq. (2.25). Each of these normals n_i defines therefore normal to the surface, which is called the characteristic surface. We will show here the very important consequences of these properties, as they have a significant effect on the whole process of discretization in CFD.

If we define a surface $S(x_j) = 0$, in the $(m + 1)$ -dimensional space of the independent variables x_j , the normal to this surface is defined by the gradient of the function $S(x_j)$, as

$$\vec{n} = \frac{\nabla S}{\|\nabla S\|} \quad (2.26)$$

(Henceforth, the normalizing $\|\nabla S\|$ is to be absorbed into the function S).

What is the significance of this characteristic surface in terms of wave propagation, referring to the plane wave solution Eq. (2.22)?

If Eq. (2.26) is introduced in the plane wave Eq. (2.22), a general representation is defined as,

$$w = \hat{U} e^{i(\vec{x} \cdot \nabla S)} = \hat{U} e^{i(x_j S_j)} \quad \text{with } S_j \equiv \frac{\partial S}{\partial x_j} \quad (2.27)$$

If we consider the tangent plane to the surface $S(x_j) = 0$, defined by

$$S(x_j) = S(0) + \vec{x} \cdot \nabla S = S(0) + x_j \frac{\partial S}{\partial x_j} = S(0) + x_j n_j \quad (2.28)$$

we observe that along the constant values of the phase of the wave $\phi = \vec{x} \cdot \nabla S$, the quantity w is constant.

Hence, we can consider that, the quantity U is propagating at a constant value in the direction of the normal \vec{n} .

The surface S is called a wave-front surface, defined as the surface separating the space domain already influenced by the propagating quantity w from the points not yet reached by the wave.

Observe that in the general case of n unknown flow quantities u_i , we have n characteristic surfaces, for a pure hyperbolic problem.

In a two-dimensional space the characteristic surface reduces to a characteristic line. The properties w are transported along the line $S(x, y) = 0$ and the vectors tangent to the characteristic line are obtained by expressing that along the wavefront:

$$dS = \nabla S \cdot dx = \frac{\partial S}{\partial x} dx + \frac{\partial S}{\partial y} dy = 0 \quad (2.29)$$

Hence, the direction of the characteristic line in two dimensions is given by

$$\frac{dy}{dx} = -\frac{S_x}{S_y} = -\frac{n_x}{n_y} \quad (2.30)$$

In two dimensions, there are two characteristic directions for a hyperbolic equation. Hence out of each point in the (x, y) domain, two characteristics can be defined, along which two quantities propagate. As we have as many unknowns, at each point the solution can be obtained from the characteristic-related quantities that have propagated from the

boundary or initial condition to that point.

To get a physical understanding of the discussion so far we can consider unsteady inviscid flow. The unsteady inviscid flow equation or the unsteady Euler Equation in non-conservation form is written as

$$\frac{\partial \rho}{\partial t} + u \frac{\partial \rho}{\partial x} + v \frac{\partial \rho}{\partial y} + w \frac{\partial \rho}{\partial z} = 0 \quad (2.31)$$

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} + w \frac{\partial u}{\partial z} = -\frac{1}{\rho} \frac{\partial p}{\partial x} \quad (2.32)$$

$$\frac{\partial v}{\partial t} + u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y} + w \frac{\partial v}{\partial z} = -\frac{1}{\rho} \frac{\partial p}{\partial y} \quad (2.33)$$

$$\frac{\partial w}{\partial t} + u \frac{\partial w}{\partial x} + v \frac{\partial w}{\partial y} + w \frac{\partial w}{\partial z} = -\frac{1}{\rho} \frac{\partial p}{\partial z} \quad (2.34)$$

$$\frac{\partial E}{\partial t} + u \frac{\partial E}{\partial x} + v \frac{\partial E}{\partial y} + w \frac{\partial E}{\partial z} = -\frac{1}{\rho} \left(\frac{\partial p u}{\partial x} + \frac{\partial p v}{\partial y} + \frac{\partial p w}{\partial z} \right) \quad (2.35)$$

By following the analysis given in Sec. 2.2, we can find that the above governing equations is *hyperbolic*, no matter whether the flow is locally subsonic or supersonic. More precisely, we say the flows are hyperbolic with respect to time. (The classification of the unsteady Euler Equations as hyperbolic with respect to time is derived in Sec 11.2.1 of [22].) This implies that in such unsteady flows, no matter whether we have one, two, or three spatial directions, the marching direction is always the *time* direction. Let us examine this more closely to understand the marching behavior discussed before for hyperbolic partial differential equations. For one dimensional flow, consider a point P in the xt plane shown in Fig. 2.2. The region influenced by P is the shaded area between the two advancing characteristics through P . The x -axis ($t = 0$) is the initial data line. The interval ab is the only portion of the initial data along the x axis which the solution at P depends. Extending these thoughts for two-dimensional unsteady flow, consider point P in the xyt space as shown in Fig. 2.3. The region influenced by P and the portion of the boundary in the xy plane upon which the solution at P depends are shown in this figure. Starting with known initial data in the xy plane, the solution “marches” forward in time. The same extension can be applied to the 3D case.

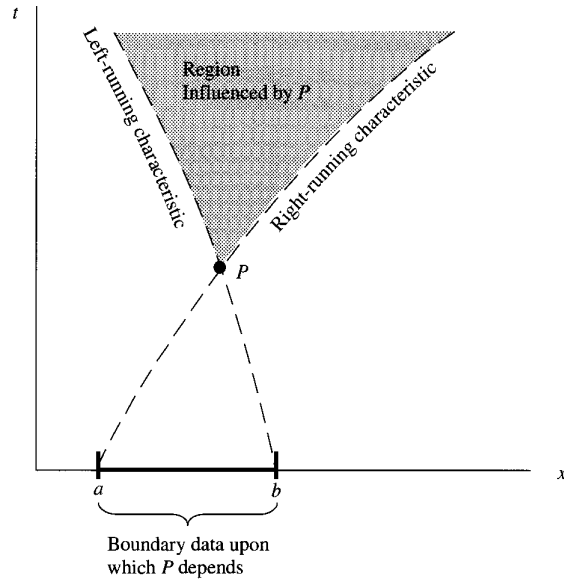


Figure 2.2: Domain and boundaries for the solution of hyperbolic equations. One dimensional unsteady flow. [22]

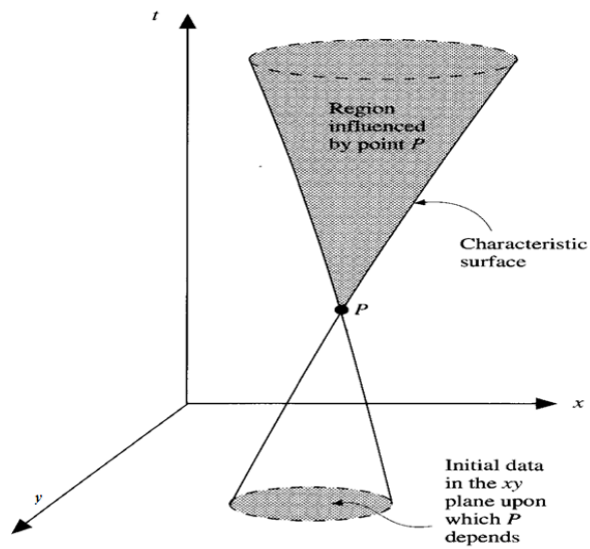


Figure 2.3: Domain and boundaries for the solution of hyperbolic equations. Two dimensional unsteady flow. [22]

2.3 Advantage of Conservation form over the non-conservation form

The conservation and non-conservation form of the continuity equation is shown below.

Conservation form:

$$\frac{\partial \rho}{\partial t} + \vec{\nabla} \cdot (\rho \vec{V}) = 0$$

Non-Conservation form:

$$\frac{D\rho}{Dt} + \rho \vec{\nabla} \cdot (\vec{V}) = 0 \text{ with } \frac{D}{Dt} \equiv \frac{\partial}{\partial t} + \vec{V} \cdot \vec{\nabla}$$

The labelling of the governing equations as either conservation or non-conservation form grew out of modern CFD, as well as concern for which method has to be preferred for a given CFD applications. We shall state here the two perspectives for the advantage of conservation form over the non-conservation form. The detail understanding of these can be found in [22].

1. The conservation form of the governing equations allows to write the system of equation in a general form. Thus, it provides an ease and better organization for numerical and computer programming.
2. Experience has shown that the conservation form of equation is better for shock-capturing method (used in this thesis). For the non-conservation form, the computed flow-field has unsatisfactory results. The reason for this is, the conservation form uses flux variables as the dependent variable and because the changes in these flux variables are either zero or small across a shock wave, the numerical quality of the shock-capturing method will be enhanced. Whereas, the non-conservation form uses the primitive variables as dependent variable, and one would see a large discontinuity in them.

2.4 Boundary Condition Specification in Hyperbolic System

Boundary condition specification is an important part of any CFD problem statement and has to be compatible with physical and numerical properties of problem.

We have already seen that information in a hyperbolic problem propagates in a specific characteristic direction, the eigenvalue spectrum of the Jacobian matrix defines how information is going to propagate. Hence for a hyperbolic problem to be well-posed we cannot specify general boundary conditions on all boundaries. Rather, the following questions have to be answered:

1. How many boundary conditions have to be imposed at a given boundary?
2. What are the boundary conditions that have to be imposed at the boundary?
3. How are the remaining variables (i.e., those without BCs) to be handled at the boundary?

In this section we will discuss only the answer to the first question pertaining to the Euler problem. The five eigenvalues of the system, which correspond to the speed of propagation of five characteristic quantities, are given by,

$$\frac{\vec{u} \cdot \vec{k}}{k}, \quad \frac{\vec{u} \cdot \vec{k}}{k}, \quad \frac{\vec{u} \cdot \vec{k}}{k}, \quad \frac{\vec{u} \cdot \vec{k}}{k} + c, \quad \frac{\vec{u} \cdot \vec{k}}{k} - c$$

where c is the local sonic speed. [1]

The derivation for the above eigenvalues can be found in the Section of 11.2.1 of [22]. Since, the transport properties at a surface are determined by the normal components of the fluxes, the number and type of conditions at a boundary of a multi-dimensional domain will be determined by the propagation of waves with the following speeds:

$$\lambda_1 = \vec{u} \cdot \hat{e}_n = v_n$$

$$\lambda_2 = \vec{u} \cdot \hat{e}_n = v_n$$

$$\lambda_3 = \vec{u} \cdot \hat{e}_n = v_n$$

$$\lambda_4 = \vec{u} \cdot \hat{e}_n + c = v_n + c$$

$$\lambda_5 = \vec{u} \cdot \hat{e}_n - c = v_n - c$$

where v_n is the *inward* normal velocity component at the considered surface, coming into the computational domain. The first three eigenvalues correspond to the entropy and vorticity waves, while the two remaining eigenvalues, are associated with acoustic waves. This defines a locally quasi-one-dimensional propagation of information and we can therefore look at how the propagation behaves at a boundary, from the the sign of these eigenvalues at the boundary.

The key to the understanding of the issue of the number of boundary conditions that are needed at the boundary is that characteristics convey information in the $(n - t)$ space formed by the local normal direction and time. When information is propagated from outside into the computational domain, it means that this information has to be obtained by a

boundary condition; this occurs when the eigenvalue λ is positive, and a physical boundary condition has to be imposed. On the other hand, when the eigenvalue λ is negative and the propagation occurs from the interior of the domain towards the boundary, this means that a boundary condition cannot be imposed from the outside. Such variable will be handled through “numerical boundary conditions”, by extrapolating interior information to the boundary.

In summary, the number of physical conditions to be imposed at a boundary with inward normal vector \vec{n} , pointing into the computational domain, is defined by the number of characteristics entering the domain.

2.5 Closure

In this chapter we saw very basic properties of system of partial differential equations with emphasis on the hyperbolic type. These properties must be understood before implementing boundary conditions, to avoid ill-posedness of system.

We saw how information flows along with characteristic in hyperbolic systems and we used this information to determine the number of variables to be assigned at the boundary, based on direction of characteristic waves, i.e. whether the characteristic is flowing into the domain or out of the domain. Also, we saw the advantage of the conservation form of governing equation over the non-conservation form for hyperbolic system.

Chapter 3

Euler Equations, Boundary Conditions and CFL criterion

3.1 A short description of Euler Equations

The System of Euler equations are first order non-linear coupled Partial Differential Equations (PDEs) governing compressible inviscid flows. They can be formulated in different ways depending upon the choice of flow variables. If the dependent variables are chosen as density, momentum, total energy that directly obey conservation laws, this is called the conservative formulation. However, if the variables are pressure, velocity, temperature, it is called the primitive variable formulation (see Eq. 2.35). As seen in Sec. 2.3 we would use the former one for our numerical methodology. Since, transient Euler equations are hyperbolic in nature, the formulation made using the characteristic variables (variables that flows along characteristic lines) can also be used. These formulation which uses characteristic variables the dependent variables is called the characteristic formulation. We can obtain the system of equations in these various forms from each other by algebraic manipulation. Here, we present the conservative form.

System of Euler equation in conservative form:

$$\frac{\partial \rho}{\partial t} + \frac{\partial(\rho u)}{\partial x} + \frac{\partial(\rho v)}{\partial y} + \frac{\partial(\rho w)}{\partial z} = 0 \quad (3.1)$$

$$\frac{\partial(\rho u)}{\partial t} + \frac{\partial(\rho u^2 + P)}{\partial x} + \frac{\partial(\rho uv)}{\partial y} + \frac{\partial(\rho uw)}{\partial z} = 0 \quad (3.2)$$

$$\frac{\partial(\rho v)}{\partial t} + \frac{\partial(\rho vu)}{\partial x} + \frac{\partial(\rho v^2 + P)}{\partial y} + \frac{\partial(\rho vw)}{\partial z} = 0 \quad (3.3)$$

$$\frac{\partial(\rho w)}{\partial t} + \frac{\partial(\rho wu)}{\partial x} + \frac{\partial(\rho wv)}{\partial y} + \frac{\partial(\rho w^2 + P)}{\partial z} = 0 \quad (3.4)$$

$$\frac{\partial(\rho E)}{\partial t} + \frac{\partial(\rho uH)}{\partial x} + \frac{\partial(\rho vH)}{\partial y} + \frac{\partial(\rho wH)}{\partial z} = 0 \quad (3.5)$$

where, $P = \rho RT$, $E = C_v T + \frac{u^2+v^2+w^2}{2}$, $H = E + \frac{P}{\rho}$

The mathematical nature of the steady and non-steady Euler Equations is stated below. The detailed derivations of the same can be found in [22, 3].

- *the steady state Euler equations show elliptic behaviour when the flow is subsonic, parabolic behaviour when flow is sonic and hyperbolic behaviour when the flow is supersonic.*
- The 3-D unsteady Euler system is *always hyperbolic* in nature.

From the computational point of view, usually we are more interested in steady state than transient solutions. Therefore, while solving the steady state equations we have to check for the sonic condition, as the numerical schemes for each type of PDE are different. Till date no scheme has been developed which can work well for all these types of PDEs. So there need to be completely separate modules to be developed for subsonic and supersonic flows, while for transonic flows it would be even more difficult to obtain solutions since the domain will contain all three types of PDEs.

Thus, by *just the addition of the temporal dimension in this system of equations makes it hyperbolic, independent of the speed of flow.* Therefore, even if we are interested in only in the steady state solution, it is best to solve the transient set of equations to steady state. This is called the false-transient approach and is used in this work.

3.2 Boundary condition treatment in terms of primitive variables

In the preceding chapter we have seen how boundary condition specification is different for hyperbolic problems compared to that of parabolic and elliptic problems, and have seen how the flow of characteristics into or out of the computational domain affects the specification of the boundary conditions.

Extending the thoughts developed in Sec. 2.4 and referring the literature [1, 2] we can present the following table and implementation of boundary condition for Euler equations. This way we answer all the three questions required for the specification of boundary conditions. Namely,

1. How many boundary conditions should be specified?
2. What boundary conditions should be specified?
3. What boundary conditions will have numerical boundary condition?

The answer to the first question depends upon on the number of characteristics that enter into domain at a boundary. Following table summarises the no of physical/numerical B.C. specification in 3-D Euler flows.

Type	Sub-sonic		
	No of <i>+ve</i> Eigen values	No of physical BC	No of Numerical B.C
Inflow	Four	Four	One
Outflow	One	One	Four
	Super-sonic		
	No of <i>+ve</i> Eigen values	No of physical BC	No of Numerical B.C
Inflow	Five	Five	Zero
Outflow	Zero	Zero	Five
	Wall		
	No of <i>+ve</i> Eigen values	No of physical BC	No of Numerical B.C
	One	One	Four

Table 3.1: No of boundary condition to be fixed on boundary in Euler system of equation

The second question would be answered in the following subsection.

3.2.1 Implementation of Boundary Conditions

For implementing the boundary condition for the structured grid arrangement, we use the fictitious cell with zero-volume approach. The value of the fictitious cell is updated using the value calculated at the boundary directly. But this, method has to be reviewed, for inhomogeneous Neumann conditions at the boundary. Under such condition, for non-orthogonal grid, taking fictitious cell-center at the face center will lead to complexity. One has to take into account the cross-diffusion terms also.

The characteristic variables has to be defined in terms of the primitive variables and using them we have to specify the boundary conditions. The detail is very interesting and can be found in [11]. The boundary condition used in this thesis is based on these concepts.

The two basic flow situations at the boundary is sketched in the Fig. 3.1.

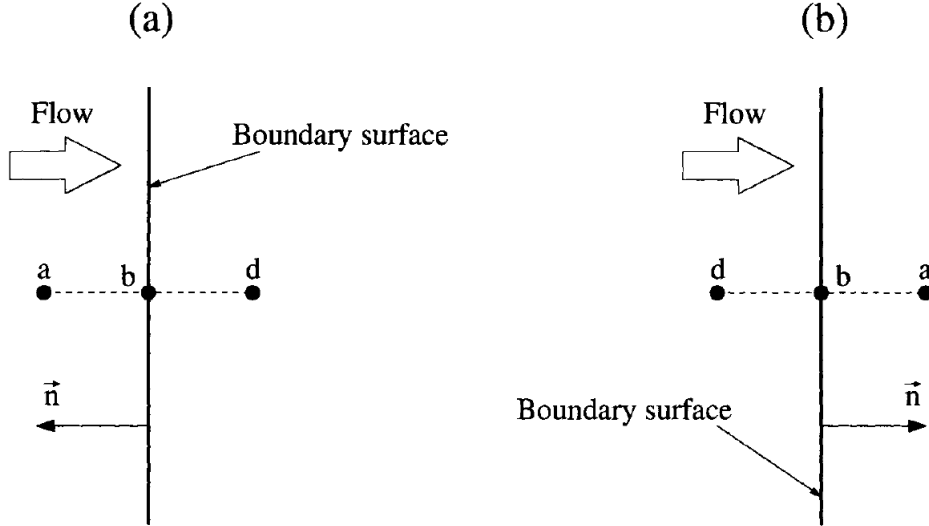


Figure 3.1: Flow Situation at boundary: inflow (a) and outflow (b) situation. Position a is outside, b on the boundary, and the position d is inside the physical domain. The unit normal vector $\vec{n} = [n_x, n_y, n_z]^T$ points out of the domain.[10]

3.2.2 Inflow BC

Subsonic Inflow

For subsonic inflow, we have four physical boundary condition and one numerical boundary condition. All combinations of conservative and primitive variables can be selected as physical boundary conditions, with the exception of the pair (u,p) ([2], Pg. 353). The combination of (u,p) is not well posed as the problem is over-specified. It allows for specifying the outgoing characteristic variables, which is already specified by the interior domain. The combinations such as (ρ, p) or (ρ, u) are well-posed boundary conditions. The former is called the pressure-driven inlet condition useful for internal flows and later is called the velocity-driven inlet condition is useful for the external flow problems. Here, u refers to inlet velocity. The remaining variable will have the “numerical BC”.

Following are the numerical formulation for subsonic inflow/inlet boundary conditions.

Velocity-Driven Flows: These are more suitable for external flow problems, which are velocity driven [10].

$$\begin{aligned}
p_b &= \frac{1}{2} \{p_a + p_d - \rho_o c_o [n_x(u_a - u_d) + n_y(v_a - v_d) + n_z(w_a - w_d)]\} \\
\rho_b &= \rho_a + \frac{(p_b - p_a)}{c_o^2} \\
u_b &= u_a - n_x \frac{(p_a - p_b)}{\rho_o c_o} \\
v_b &= v_a - n_y \frac{(p_a - p_b)}{\rho_o c_o} \\
w_b &= w_a - n_z \frac{(p_a - p_b)}{\rho_o c_o}
\end{aligned} \tag{3.6}$$

where ρ_o and c_o represent a reference state. The reference state is normally set equal to the state at the interior point (point d in Fig. 3.1). The values at point a are determined from the freestream state.

Pressure-Driven Flows: A common procedure consists of the specification of the total pressure, total temperature, and of two flow angles. We unsuccessfully attempted to implement the boundary condition based on the outgoing Riemann invariant, as given in [10]. This could be tried in future again. However, based on the basic of compressible fluid flow we came up with a simpler formulation. It requires specification of total pressure, total temperature and velocity in y and z direction for flow having dominance in x -direction. The value of velocity in x direction is numerically extrapolated from inside the domain. The following are the isentropic relations used for determining the static pressure (p), density (ρ) and temperature (T), which are used in the governing equations.

$$\begin{aligned}
p_o &= p \left(1 + \frac{\gamma - 1}{2} M^2\right)^{\frac{\gamma}{\gamma - 1}} \\
\rho_o &= \rho \left(1 + \frac{\gamma - 1}{2} M^2\right)^{\frac{1}{\gamma - 1}}
\end{aligned} \tag{3.7}$$

The imposed isentropic static-to-stagnation pressure ratio implies the inlet Mach number. Thus, this boundary condition can also be defined in terms of inlet Mach number and the flow angle. This can be incorporated in the future versions of the solver.

Supersonic Inflow

When the flow is supersonic, all boundary conditions are physical. The conservative variables on the boundary (point b in Fig. 3.1) are determined by freestream values only.

3.2.3 Outflow BC

Subsonic Outflow

It requires only one physical boundary conditions, the others have to be numerical boundary conditions. The most appropriate physical condition, particularly for internal flows and corresponding to most experimental situations, consists in fixing the downstream static

pressure. This can also be applied for external flow problems. The following numerical formulation is used:

$$\begin{aligned}
p_b &= p_a \\
\rho_b &= \rho_d + \frac{(p_b - p_a)}{c_o^2} \\
u_b &= u_d + n_x \frac{(p_d - p_b)}{\rho_o c_o} \\
v_b &= v_d + n_y \frac{(p_d - p_b)}{\rho_o c_o} \\
w_b &= w_d + n_z \frac{(p_d - p_b)}{\rho_o c_o}
\end{aligned} \tag{3.8}$$

with p_a being the prescribed static pressure.

A point to be considered is that when imposing a constant pressure at a subsonic exit section, one actually allows perturbation waves to be reflected at the boundaries. The non-reflecting boundary condition [23, 6] expresses the physical boundary condition as the requirement that the local perturbations propagated along incoming characteristics be made to vanish. We use the work of Rudy [9] to implement the non-reflecting boundary condition. It has the following form.

$$\frac{\partial u}{\partial t} - \frac{1}{\rho_b a_b} \frac{p_b^{n+1} - p_b^n}{\Delta t} - \frac{\alpha}{\rho a} (p_b^{n+1} - p_b^*) = 0 \tag{3.9}$$

where,

$$\alpha = \begin{cases} 0.25 & \text{for } M \geq 0.7 \\ 0.6 & \text{for } 0.5 \leq M \leq 0.7 \\ 1 & \text{otherwise} \end{cases}$$

, p_b^* is the constant pressure imposed at the subsonic exit section and a_b is the sonic speed.

Supersonic Outflow

When the flow is supersonic at outflow, all the conservative variables at the boundary must be determined from the solution inside the boundary.

3.2.4 Wall (or Solid) Boundary

Since the Euler equation system describes inviscid flow, we cannot assign a no-slip BC at the wall. Only one physical BC can be imposed.

$$\vec{v} \cdot \hat{n} = 0 \text{ at the solid boundary,}$$

where \hat{n} denotes unit normal vector at the solid boundary.

In numerical calculation with finite volume methodology we are interested in the flux at the surface of a cell than the values of the variable at the wall. Fluxes in the Euler equation can be written as,

$$F_x = \begin{pmatrix} \rho u \\ \rho u^2 + p \\ \rho v u \\ \rho w u \\ \rho u H \end{pmatrix} \quad F_y = \begin{pmatrix} \rho v \\ \rho v u \\ \rho v^2 + p \\ \rho v w \\ \rho v H \end{pmatrix} \quad F_z = \begin{pmatrix} \rho w \\ \rho w u \\ \rho w v \\ \rho w^2 + p \\ \rho w H \end{pmatrix} \quad (3.10)$$

Multiplying with the respective area components at the wall surface segment of the cell, the net flux crossing a surface can be written as,

$$(F_x S_{f_{wx}} + F_y S_{f_{wy}} + F_z S_{f_{wz}})_{wall} = \begin{pmatrix} \rho u S_{f_{wx}} + \rho v S_{f_{wy}} + \rho w S_{f_{wz}} \\ u_w (\rho u S_{f_{wx}} + \rho v S_{f_{wy}} + \rho w S_{f_{wz}})_w + p_w S_{f_{wx}} \\ v_w (\rho u S_{f_{wx}} + \rho v S_{f_{wy}} + \rho w S_{f_{wz}})_w + p S_{f_{wy}} \\ w_w (\rho u S_{f_{wx}} + \rho v S_{f_{wy}} + \rho w S_{f_{wz}})_w + p S_{f_{wz}} \\ H_w (\rho u S_{f_{wx}} + \rho v S_{f_{wy}} + \rho w S_{f_{wz}})_w \end{pmatrix} \quad (3.11)$$

Since at the wall

$$\rho u S_{f_{wx}} + \rho v S_{f_{wy}} + \rho w S_{f_{wz}} = 0$$

$$F_x S_{f_x} + F_y S_{f_y} + F_z S_{f_z} = \begin{pmatrix} 0 \\ p \\ 0 \\ 0 \\ 0 \end{pmatrix} S_{f_x} + \begin{pmatrix} 0 \\ 0 \\ p \\ 0 \\ 0 \end{pmatrix} S_{f_y} + \begin{pmatrix} 0 \\ 0 \\ 0 \\ p \\ 0 \end{pmatrix} S_{f_z} \quad (3.12)$$

The general discretized Euler equation (which we will derive in next chapter) is given as,

$$V_p \frac{w_p^{n+1} - w_p^n}{\Delta t} = - \sum_f F_{fx} S_{f_x} + F_{fy} S_{f_y} + F_{fz} S_{f_z} \quad (3.13)$$

so whenever for cell p surface f corresponds to the solid boundary, then flux will be calculated from Eq. (3.12). Thus, using this method we are actually using at wall the physical Boundary condition as $v_n = 0$ and the remaining variables will have numerical boundary conditions.

3.2.5 Symmetry Boundary Conditions

We apply the Neumann boundary condition on characteristic variables to update value of the boundary slabs corresponding to symmetry boundaries. While for flux calculation we follow the same procedure as for wall described by Eq. (3.12). Since, at symmetry we also have $v_n = 0$.

Numerical boundary conditions: We end this section with the discussion on how to implement the numerical boundary conditions. This is particularly important for solid walls, where we want to determine the pressure variations. The simplest way is to take the value at the cell centre of the associated cell. This is a zero-order extrapolations. We can also apply volume extrapolation using the inside two cells. This is first-order and thus supposedly more accurate. *Through numerical experiments we have found that the linear extrapolation esp. at wall and symmetry plane can lead to divergence in the solution.*

3.3 CFL Condition

3.3.1 Physical interpretation

From our study of hyperbolic partial differential equations, we know that the solution at a point does not depend upon the previous solution in the whole domain, but rather it depends on the previous solution in the *domain of dependence*. This domain is also referred to as the *physical domain of dependence or true domain of dependence*. While determining the solution at a point using a computational technique, we use the previous solution at its neighbouring points; a space formed by these points is known as the *numerical domain of dependence*.

The figure below shows the physical and numerical domain of dependence for explicit schemes.

Fig. 3.2 shows the numerical domain of dependence formed by a scheme which uses solution at points $(n, j - 1)$, (n, j) , $(n, j + 1)$ to calculate the solution at point n, j

Now to avoid numerical instability and to capture the solution correctly, *the numerical domain of dependence must lie within the physical domain of dependence*. This condition is called the *Courant-Fridrichs-Lewy* (CFL) condition. Any numerical method that violates the CFL condition uses information from outside the physical domain of dependence, and has large errors and usually causes the solution to blow-up.

Mathematically, the CFL condition is stated as $\frac{a\Delta t}{\Delta x} \leq 1$, where a is the wave-speed. The dimensionless term $\frac{a\Delta t}{\Delta x}$ known as the *Courant No.* and we can say that CFL condition is followed if and only if $CN \leq 1$.

Extending the concept by analogy to a 3-D rectangular grid, we obtain the CFL condition [3] as

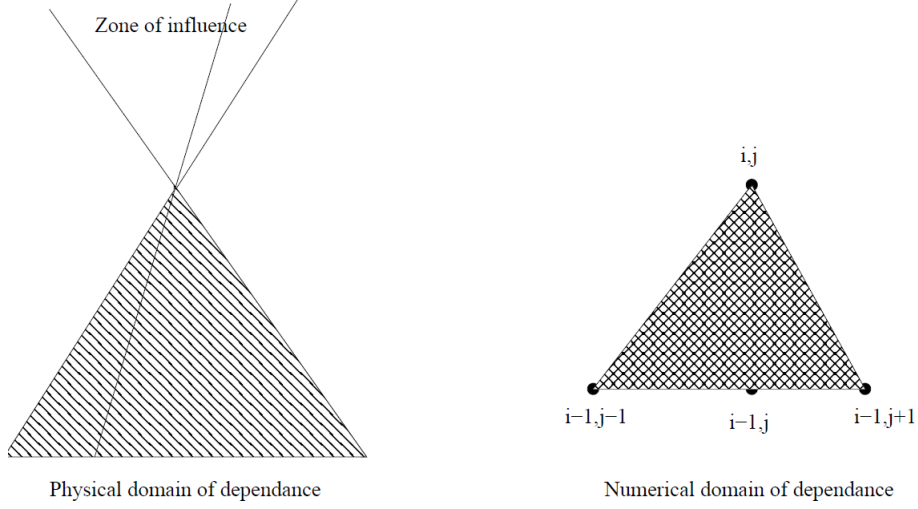


Figure 3.2: Physical and Numerical domain of dependence

$$\Delta t \leq \min \left[\frac{\Delta x}{a_x}, \frac{\Delta y}{a_y}, \frac{\Delta z}{a_z} \right] \quad (3.14)$$

There are many practical difficulties that arise in the implementation of the CFL condition given in expression (3.14) for a finite-volume solver using non-rectangular cells. To overcome these difficulties we will derive the following volume based CFL condition. Consider a general 2-D non-orthogonal grid with local rectangular assumption,



Figure 3.3: Locally rectangular grid assumption

where A, B are cell centres and C is common face centre. Rewriting the CFL condition with AB as the distance of propagation of information and a as the wave speed, we get,

$$\frac{a\Delta t}{AB} = \text{CN} < 1$$

$$\frac{a\Delta t}{\frac{\Delta x_1}{2} + \frac{\Delta x_2}{2}} < 1$$

Then, multiplying and dividing by common surface area S_f

$$\frac{a\Delta t S_f}{\left(\frac{\Delta x_1}{2} + \frac{\Delta x_2}{2}\right) S_f} < 1 \quad S_f = \sqrt{S_{fx}^2 + S_{fy}^2 + S_{fz}^2}$$

$$\frac{a\Delta S_f \times 2}{V_1 + V_2} < 1$$

or in terms of Courant No.,

$$\Delta t = \frac{CN (V_p + V_{nb})}{2 \times c \times S_f}$$

This is the volume based expression for determining size of Δt uses a chosen value of $CN \leq 1$. Now consider all six indices direction, i.e., e, w, n, s, t, b and note the simplicity in expression when cell is near the boundary and $V_{nb} = 0$

$$\Delta t_e = CN \frac{V_p + V_E}{2aS_e} \quad S_e = \sqrt{S_{ex}^2 + S_{ey}^2 + S_{ez}^2}$$

$$\Delta t_w = CN \frac{V_p + V_W}{2aS_w} \quad S_w = \sqrt{S_{wx}^2 + S_{wy}^2 + S_{wz}^2}$$

$$\Delta t_n = CN \frac{V_p + V_N}{2aS_n} \quad S_n = \sqrt{S_{nx}^2 + S_{ny}^2 + S_{nz}^2}$$

$$\Delta t_s = CN \frac{V_p + V_S}{2aS_s} \quad S_s = \sqrt{S_{sx}^2 + S_{sy}^2 + S_{sz}^2}$$

$$\Delta t_t = CN \frac{V_p + V_T}{2aS_t} \quad S_t = \sqrt{S_{tx}^2 + S_{ty}^2 + S_{tz}^2}$$

$$\Delta t_b = CN \frac{V_p + V_B}{2aS_b} \quad S_b = \sqrt{S_{bx}^2 + S_{by}^2 + S_{bz}^2}$$

We implement the CFL condition by choosing the minimum Δt out of six,

$$\Delta t = \min(\Delta t_e, \Delta t_w, \Delta t_n, \Delta t_s, \Delta t_t, \Delta t_b) \quad (3.15)$$

The above section defines timestep using cell-face values. A similar method can be described based on the cell-center values [22]. We follow the method based on cell-center values, though the code also provides function for cell-face values.

3.4 Closure

In this chapter we have set up the basic governing equations we use for the study of inviscid compressible flows. The boundary conditions described in the previous chapter have been implemented. At last we have discussed the criteria for the calculation of the time-step, based on the CFL criterion.

Chapter 4

Numerical Methodology

4.1 Numerical Schemes.

Real flow includes rotational, non-isentropic, and non-isothermal effects. Compressible inviscid flow including such effects requires simultaneous solution of continuity, momentum, and energy equations. Special computational schemes are required to resolve the shock discontinuities encountered in transonic flow. The most basic requirement for the solution of the Euler equations is to assure that solution schemes provide an adequate amount of artificial viscosity required for rapid convergence toward an exact solution.

Numerical schemes to solve Euler equations may be grouped into three major categories: (1) central schemes, (2) first order upwind schemes, and (3) second order upwind schemes and essentially non-oscillatory (ENO) schemes.

1. Central Schemes. These schemes are combined space-time integration schemes.
 - (a) Explicit schemes
 - i. Lax- Friedrichs: first-order scheme
 - ii. Lax- Wendroff: second-order scheme
 - (b) Two-step explicit schemes
 - i. Richtmyer and Morton scheme
 - ii. MacCormack scheme
2. First order upwind
 - (a) Multiple Flux vector splitting method
 - i. Steger and Warming method
 - ii. Van Leer method.
 - (b) Godunov methods.

3. Second order upwind

- (a) Variable extrapolation
- (b) TVD (Total variation diminishing) scheme

In the present work, the MacCormack scheme has been chosen to solve Euler equations, since it is a very robust and tested scheme. We extend the scheme to TVD MacCormack by slight modification in the MacCormack code. We have also implemented the upwind method, by incorporating the AUSM+ scheme in the solver.

4.2 Governing Equations

The Euler equations which describes the inviscid compressible fluid motion can be presented in conservation form as,

$$\frac{\partial \rho}{\partial t} + \frac{\partial(\rho u)}{\partial x} + \frac{\partial(\rho v)}{\partial y} + \frac{\partial(\rho w)}{\partial z} = 0 \quad (4.1)$$

$$\frac{\partial(\rho u)}{\partial t} + \frac{\partial(\rho u^2 + P)}{\partial x} + \frac{\partial(\rho uv)}{\partial y} + \frac{\partial(\rho uw)}{\partial z} = 0 \quad (4.2)$$

$$\frac{\partial(\rho v)}{\partial t} + \frac{\partial(\rho vu)}{\partial x} + \frac{\partial(\rho v^2 + P)}{\partial y} + \frac{\partial(\rho vw)}{\partial z} = 0 \quad (4.3)$$

$$\frac{\partial(\rho w)}{\partial t} + \frac{\partial(\rho wu)}{\partial x} + \frac{\partial(\rho wv)}{\partial y} + \frac{\partial(\rho w^2 + P)}{\partial z} = 0 \quad (4.4)$$

$$\frac{\partial(\rho E)}{\partial t} + \frac{\partial(\rho uH)}{\partial x} + \frac{\partial(\rho vH)}{\partial y} + \frac{\partial(\rho wH)}{\partial z} = 0 \quad (4.5)$$

where $P = \rho RT$, $E = C_v T + \frac{u^2+v^2+w^2}{2}$, $H = E + \frac{P}{\rho}$

4.3 Discretization of Governing Equation

The equations can be written in compact form as

$$\frac{\partial \{w_i\}}{\partial t} + \frac{\partial \{F_{xi}\}}{\partial x} + \frac{\partial \{F_{yi}\}}{\partial y} + \frac{\partial \{F_{zi}\}}{\partial z} = 0 \quad (4.6)$$

$$w \equiv \begin{pmatrix} \rho \\ \rho u \\ \rho v \\ \rho w \\ \rho E \end{pmatrix}$$

$$F_x \equiv \begin{pmatrix} \rho u \\ \rho u^2 + P \\ \rho uv \\ \rho uw \\ \rho uH \end{pmatrix}, F_y \equiv \begin{pmatrix} \rho v \\ \rho uv \\ \rho v^2 + P \\ \rho vw \\ \rho vH \end{pmatrix}, F_z \equiv \begin{pmatrix} \rho w \\ \rho uw \\ \rho vw \\ \rho w^2 + P \\ \rho wH \end{pmatrix}$$

Note that the F_x, F_y, F_z column vectors are used just for notational convenience. Now assuming that \mathbf{F} is an arbitrary vector whose x,y,z components are F_x, F_y, F_z we can write

$$\frac{\partial w_i}{\partial t} + \nabla \cdot \mathbf{F}_i = 0$$

where each row i respectively represents the governing continuity, momentum, energy equation equations.

The finite volume method uses the integral form of the equations while the governing equation above is in differential form. The corresponding integral form of the equation can be obtained by taking the integral of the equation over a control volume.

$$\oint_V \left(\frac{\partial w_i}{\partial t} + \nabla \cdot \mathbf{F}_i \right) dV = 0$$

where V is the fluid domain under analysis. Using the divergence theorem, $\oint_V \nabla \cdot \vec{v} dV = \oint_S \vec{v} \cdot d\vec{S}$ we get

$$\oint_V \frac{\partial w_i}{\partial t} dV + \oint_S \mathbf{F}_i \cdot d\vec{S} = 0$$

Assuming the control volume is not changing with time, the equation can be written as,

$$\frac{\partial}{\partial t} \oint_V w_i dV + \oint_S \mathbf{F}_i \cdot d\vec{S} = 0$$

The equation can be divided into the temporal and convective parts, as shown, and we will now do the finite volume discretization of each part to get the full discretized equation.

$$\underbrace{\frac{\partial}{\partial t} \oint_V w_i dV}_{\text{TemporalPart}} + \underbrace{\oint_S \mathbf{F}_i \cdot d\vec{S}}_{\text{Convectivepart}} = 0$$

The control volume V can be arbitrarily chosen so we use the above equation for each of the finite-volume cells of a chosen grid spanning the entire computational domain. The shape of the finite volume cells are the user's choice. The numerical results obtained from hexahedral elements are more accurate than that obtained from any other element like triangular prism, pyramid or polyhedral. And it is easy to create a structured grid from hexahedral elements rather than use of any other prismatic shaped element. Thus, we use

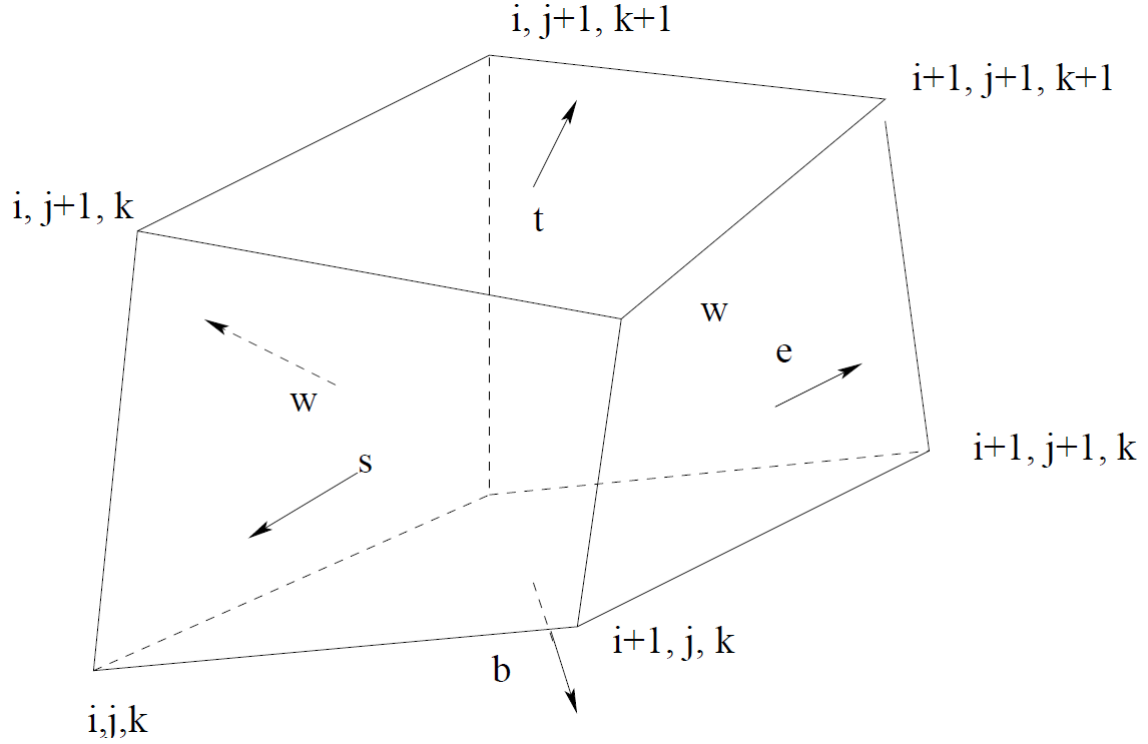


Figure 4.1: Finite volume cell

non-orthogonal hexahedral elements, as shown in the figure, having east, west, north, south, top and bottom faces.

Temporal term: The volume averaged value of conservative variable can be written for the p th cell as:

$$\frac{1}{V_p} \oint_{V_p} w dV = w_p$$

thus,

$$\oint_{V_p} w dV = V_p w_p$$

where, V_p is the volume of the p th cell.

Using this volume averaged value we can get the discretized form of the temporal term as:

$$\frac{\partial w_i}{\partial t} = V_p \frac{w_p^{n+1} - w_p^n}{\Delta t}$$

Convective term: In the convective term, the integral is carried out over the full surface of the control volume, without any approximation it can be divided into six parts over the east(e), west(w), north(n), south(s), top(t) and bottom(b) faces as follows:

$$\oint_S \vec{F}_i \cdot d\vec{S} = \oint_e \vec{F}_i \cdot d\vec{S}_e + \oint_w \vec{F}_i \cdot d\vec{S}_w + \oint_n \vec{F}_i \cdot d\vec{S}_n + \oint_s \vec{F}_i \cdot d\vec{S}_s + \oint_t \vec{F}_i \cdot d\vec{S}_t + \oint_b \vec{F}_i \cdot d\vec{S}_b$$

where each face integral can be divided, without approximation, into 3 scalar parts:

$$\oint_{S_f} \vec{F}_i \cdot d\vec{S}_f = \oint_S F_{ix} dS_x + \oint_S F_{iy} dS_y + \oint_S F_{iz} dS_z$$

The value of flux variable may change over the surface. For each scalar component, we now approximate the surface averaged value of the variable by its face-centroid value F_{if} :

$$\frac{1}{S_f} \oint_{S_f} F_i d\vec{S}_f = F_{if}$$

Therefore we can write,

$$\oint_{S_f} \vec{F}_i \cdot d\vec{S}_f = F_{ix} S_{fx} + F_{iy} S_{fy} + F_{iz} S_{fz}$$

where S_{fi} is the i^{th} component of face vector \vec{S}_f . Repeating the procedure for each of the faces we can write

$$\begin{aligned} \oint_{S_f} \vec{F} \cdot d\vec{S}_f &= F_{ex} S_{ex} + F_{ey} S_{ey} + F_{ez} S_{ez} + F_{wx} S_{wx} + F_{wy} S_{wy} + F_{wz} S_{wz} \\ &+ F_{nx} S_{nx} + F_{ny} S_{ny} + F_{nz} S_{nz} + F_{sx} S_{sx} + F_{sy} S_{sy} + F_{sz} S_{sz} \\ &+ F_{tx} S_{tx} + F_{ty} S_{ty} + F_{tz} S_{tz} + F_{bx} S_{bx} + F_{by} S_{by} + F_{bz} S_{bz} \end{aligned}$$

Now, putting the discretized temporal and convective terms together, the Euler equations can be written in discretized form as:

$$V_p \frac{w_p^{n+1} - w_p^n}{\Delta t} = - \sum_f (F_{fx} S_{fx} + F_{fy} S_{fy} + F_{fz} S_{fz}) \quad (4.7)$$

So, while discretizing the equation the following approximations are considered:

1. The volume-average value of the variable is approximated by its cell-center value.
2. The surface-average value of the variable on a cell-face is approximated by its face-center value.
3. Second-order time variation is neglected.
4. The faces of the cell are taken to be flat.

The approach used here for space discretization (dividing the domain into small finite-volume cells) is the cell-centered approach because of its various advantages for a general-purpose finite-volume solver. Therefore, the data-structure used for the discretized equation stores the cell-values in its solution array. But the discretized equation also requires the surface averaged value at the six faces of each cell. To get the variable face values from cell centre values we need some interpolation function. Since, we are dealing with a convective equation whose nature is quite different from a diffusion equation, the interpolation function should give greater weight to upwind than to downwind values, in estimating the variable value at any given point.

4.4 MacCormack scheme in FVM

The MacCormack scheme [22] in the finite volume methodology is described below. Rewriting

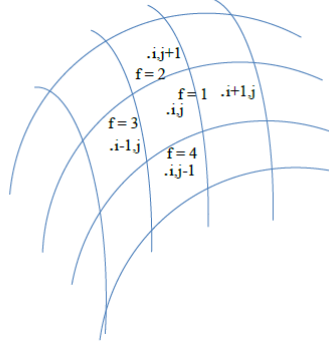


Figure 4.2: Finite volume grid

discretized equation (4.7)

$$V_p \frac{w_p^{n+1} - w_p^n}{\Delta t} = - \sum_f (F_{fx} S_{fx} + F_{fy} S_{fy} + F_{fz} S_{fz}) \quad (4.8)$$

and assuming step 1 as forward differenced and step 2 as backward differenced, the finite volume discretization can be written as,

Step 1: Predictor Step

$$\begin{aligned} W_P^* = w_p^n - \frac{\Delta t}{V_p} & (F_{Ex}^n S_{ex} + F_{Ey}^n S_{ey} + F_{Ez}^n S_{ez} + F_{Px}^n S_{wx} + F_{Py}^n S_{wy} + F_{Pz}^n S_{wz} \\ & + F_{Nx}^n S_{nx} + F_{Ny}^n S_{ny} + F_{Nz}^n S_{nz} + F_{Px}^n S_{sx} + F_{Py}^n S_{sy} + F_{Pz}^n S_{sz} \\ & + F_{Tx}^n S_{tx} + F_{Ty}^n S_{ty} + F_{Tz}^n S_{tz} + F_{Px}^n S_{bx} + F_{Py}^n S_{by} + F_{Pz}^n S_{bz}) \end{aligned} \quad (4.9)$$

Step 2: Corrector Step

$$\begin{aligned}
W_p^{**} = w_p^* - \frac{\Delta t}{V_p} & (F_{P_x}^* S_{ex} + F_{P_y}^* S_{ey} + F_{P_z}^* S_{ez} + F_{W_x}^* S_{wx} + F_{W_y}^* S_{wy} + F_{W_z}^* S_{wz} \\
& + F_{P_x}^* S_{nx} + F_{P_y}^* S_{ny} + F_{P_z}^* S_{nz} + F_{S_x}^* S_{sx} + F_{S_y}^* S_{sy} + F_{S_z}^* S_{sz} \\
& + F_{P_x}^* S_{tx} + F_{P_y}^* S_{ty} + F_{P_z}^* S_{tz} + F_{S_x}^* S_{bx} + F_{S_y}^* S_{by} + F_{S_z}^* S_{bz}) \quad (4.10)
\end{aligned}$$

and the new time-level value can be calculated as:

$$w_P^{n+1} = \frac{w_P^n + w_P^{**}}{2} \quad (4.11)$$

We see above that we have applied forward difference step in the Predictor step and backward difference in the corrector step. We can apply the same in reverse also. The left running waves are better captured by the former one, whereas the right running waves are better captured by the second version. To avoid favouring either left- or right-running waves, the two versions are often combined, reversing the order of FTBS and FTFS after every time-steps ([13], Pg.361). Since the MacCormack scheme is second order accurate in space and time, oscillations are observed in solution having abrupt step-changes in value. The amplitude of oscillation depends upon the magnitude of the change; often this causes the variable to cross the CFL condition and the solution gets blown to infinity. To avoid such oscillations and to get physically acceptable solutions, we need to add a small amount of diffusion in the governing equation which is discussed in the next section.

4.4.1 Artificial Viscosity

The MacCormack method operates satisfactorily in the regions where the variations of properties is smooth. But there is oscillations occurring around discontinuities, i.e. around a shock wave or in the boundary layer. So, *artificial smoothing* terms must be introduced, to damp these oscillations.

From the basic CFD we know that modified equation of a PDE gives us some information on the *behaviour* to be expected of the numerical solution of the difference equation. The modified equation for the one-dimensional wave equation given by

$$\frac{\partial u}{\partial t} + a \frac{\partial u}{\partial x} = 0 \quad (4.12)$$

is shown below

$$\begin{aligned}
\frac{\partial u}{\partial t} + a \frac{\partial u}{\partial x} = \frac{a\Delta x}{2}(1 - \nu) \frac{\partial^2 u}{\partial x^2} + \frac{a(\Delta x)^2}{6}(3\nu - 2\nu_2 - 1) \frac{\partial^3 u}{\partial x^3} \\
+ O[(\Delta t)^3, (\Delta t)^2(\Delta x), (\Delta t)(\Delta x)^2, (\Delta x)^3] \quad (4.13)
\end{aligned}$$

The dissipative term in the above equation, i.e., even-order derivative terms $\frac{\partial^2 u}{\partial x^2}$ is actually

the artificial viscosity term implicitly embedded in the numerical scheme. It prevents the solution from going unstable due to the oscillations caused by the dispersive terms i.e. odd-order derivative terms $\frac{\partial^3 u}{\partial x^3}$. But for variable velocity problems, the MacCormack scheme often does not have enough artificial viscosity implicitly in the algorithm, and the solution will become unstable unless more artificial viscosity is added *explicitly* to the calculation, which makes the solution more inaccurate. Therefore, there is a trade off involved. The artificial viscosity formulation is explained in Appendix A.

4.5 TVD Scheme

Numerical schemes of second and even higher orders of accuracy have oscillatory behavior. This oscillatory behavior creates errors in the solution, which can lead to non-physical values of quantities which are physically bounded. Godunov (1959) introduced an important concept known as monotonicity to characterize numerical schemes. Monotonicity means no new extrema should be created other than those which are already present in the initial solution. That is the maxima in the solution must be non-increasing and minima nondecreasing. Oscillating solutions are non-monotonic. For non-linear equations [24] introduced the concept of bounded total variation and the Total Variation Diminishing criteria. The principal condition of TVD schemes is that the total variation of the solution, defined as

$$TV = \sum_i |U_{i+1} - U_i| \quad (4.14)$$

for a scalar conservation equation, should decrease with time. The TVD property ensures that unwanted oscillations are not generated in the solution and monotonicity is preserved, which allows strong shock waves to be accurately captured without any spurious oscillations of the solution. In TVD schemes limiters or limiter functions prevent unwanted spurious solutions in the region of high gradient. Limiters maintain the original higher order discretization of the numerical scheme in the smooth flow regions, but in the regions of high gradients and /or strong discontinuities the limiter has to reduce the order of the scheme by adding high numerical dissipation to prevent the generation of spurious extrema.

A detail study and implementation work on TVD schemes has been done by Furst [25]. The properties of TVD described by [24] is valid only for the one-dimensional case. In fact Goodman and LeVeque show in [26], that the TVD property in multidimensional case is only a first order accurate. In spite of their result, many methods based on one-dimensional high order TVD methods were constructed for practical problems. While, they are *not* TVD, they usually remain high order for smooth solutions and do not generate oscillations near discontinuities. Coquel and Le Floch derive in [27] weak BV (Bounded Variation) estimates to extend the concept of TVD to multi-dimension case. This approach has been used by [25, 28] in his various works to extend the concept to MacCormack Scheme. The

present implementation of TVD MacCormack scheme has been inspired by their work.

4.5.1 TVD MacCormack Scheme

Total variation diminishing (TVD) is a property of certain discretization schemes used to solve hyperbolic partial differential equations. The concept of TVD was introduced by Ami Harten [24]. We have use a TVD scheme based on the work done in published literature [29, 30, 28]. We describe the pros and cons of this scheme in the following subsection.

After the predictor and corrector steps the MacCormack scheme presented above can be updated to TVD form by appending to the right hand side of Eq. 4.11 the TVD-type viscosity term: We use ijk notation in this section for cell-center.

$$W_{ijk}^{n+1} = w_{ijk}^{n+1} + DW_{ijk}^1 + DW_{ijk}^2 + DW_{ijk}^3 \quad (4.15)$$

DW_{ijk}^1, DW_{ijk}^2 and DW_{ijk}^3 are the part of artificial dissipation in the direction of index i , j and k respectively.

We use the artificial viscosity terms in the form

$$\begin{aligned} DW_{ijk}^1 = & \left[\bar{G}^{1+} \left(r_{ijk}^{1+} \right) + \bar{G}^{1-} \left(r_{i+1jk}^{1-} \right) \right] \left(W_{i+1jk}^n - W_{ijk}^n \right) \\ & - \left[\bar{G}^{1+} \left(r_{i-1jk}^{1+} \right) + \bar{G}^{1-} \left(r_{ijk}^{1-} \right) \right] \left(W_{ijk}^n - W_{i-1jk}^n \right) \end{aligned} \quad (4.16)$$

where,

$$r_{ijk}^{1+} = \frac{W_{ijk}^n - W_{i-1jk}^n}{W_{i+1jk}^n - W_{ijk}^n}, \quad r_{ijk}^{1-} = \frac{W_{i+1jk}^n - W_{ijk}^n}{W_{ijk}^n - W_{i-1jk}^n} \quad (4.17)$$

$$\bar{G}^{1\pm}(r_{ijk}^{1\pm}) = \frac{1}{2} C(\nu_{ijk}) \left[1 - \Phi(r_{ijk}^{1\pm}) \right] \quad (4.18)$$

where,

$$\Phi(r_{ijk}) = \max(0, \min(2r_{ijk}, 1)) \quad (4.19)$$

$$C(\nu_{ijk}) = \begin{cases} \nu_{ijk}(1 - \nu_{ijk}) & \text{for } \nu_{ijk} \leq 0.5 \\ 0.25 & \text{for } \nu_{ijk} > 0.5 \end{cases} \quad (4.20)$$

$$\nu_{ijk} = \frac{\Delta t}{\Delta x} \Psi \left(\min \left(\left| u_{ijk}^{(1)} - c_{ijk} \right|, \left| u_{ijk}^{(1)} \right|, \left| u_{ijk}^{(1)} + c_{ijk} \right| \right) \right), \quad c_{ijk} = \sqrt{\gamma \frac{P_{ijk}}{\rho_{ijk}}} \quad (4.21)$$

$$\Psi(z) = \begin{cases} |z| & \text{for } |z| > \epsilon_e \\ \frac{z^2 + \epsilon_e^2}{2\epsilon_e} & \text{for } |z| \leq \epsilon_e \end{cases}, \quad \epsilon_e = 1e^{-3} \quad (4.22)$$

where Δx_{ijk} is the approximation of the length of the cell ijk in the direction i computed as

$$\Delta x_{ijk} = \frac{2V ol_{ijk}}{\vec{S}_{i+\frac{1}{2}jk} - \vec{S}_{i-\frac{1}{2}jk}} \quad (4.23)$$

and $u_{ijk}^{(1)}$ is the velocity in the direction i evaluated as

$$u_{ijk}^{(1)} = \frac{\left\langle \vec{S}_{i+\frac{1}{2}jk} - \vec{S}_{i-\frac{1}{2}jk}, (u_{ijk}, v_{ijk}, w_{ijk}) \right\rangle}{\left| \vec{S}_{i+\frac{1}{2}jk} - \vec{S}_{i-\frac{1}{2}jk} \right|} \quad (4.24)$$

where, $\langle \cdot, \cdot \rangle$ denotes usual inner product on \mathbb{R}^m .

In terms of letter notation, $\left| \vec{S}_{i+\frac{1}{2}jk} - \vec{S}_{i-\frac{1}{2}jk} \right|$ is same as $|\vec{S}_w| + |\vec{S}_e|$.

For DW_{ijk}^2 and DW_{ijk}^3 , we use similar formulae. The time step can be chosen either as same used for the MacCormack scheme or by using the following so-called CFL conditon

$$\Delta t = CFL \min \frac{1}{\frac{|u_{ijk}^{(1)}| + c_{ijk}}{\Delta x_{ijk}} + \frac{|u_{ijk}^{(2)}| + c_{ijk}}{\Delta y_{ijk}} + \frac{|u_{ijk}^{(3)}| + c_{ijk}}{\Delta z_{ijk}}} \quad (4.25)$$

Analogically, $(u^{(2)}$ and $\Delta y)$ and $(u^{(3)}$ and $\Delta z)$ are the velocity and the length of the cell in the direction of the change of index j and k .

Special Treatment at Boundary: The r calculation at cell-centre ijk require five-grid stencil Eq. (4.17) and (4.16). Thus, for the cells next to boundary or interface slabs, the calculation will require the conservative variable value at grid point which does not exist. Thus, for such r at present an *ad hoc* treatment is done. We take corresponding value of r at nearby cell. This is expected to work well for problems which have no discontinuity at boundary or interface. But the accuracy of the solution is not good. It can be seen through Fig. 4.3.

For the cell next to the boundary/interface, we require modification in TVD formulation (since, we have only one fictious cell). In this thesis, we could not arrive at a formulation which gives accurate results. So, we have used the artificial viscosity formulation discussed in Sec. 4.4.1 and presented in Appendix A at the boundary and interface slabs. This diminishes the purpose of TVD scheme at the boundary and interface. For this case, through numerical experiment we had to determine the correct values for the artificial viscosity parameters C_x, C_y and C_z . It comes out to be $C_x = C_y = C_z = 0.9$. These analysis can be seen through Fig. 4.4. The experiments performed are for the case mentioned in Sec. 5.1.

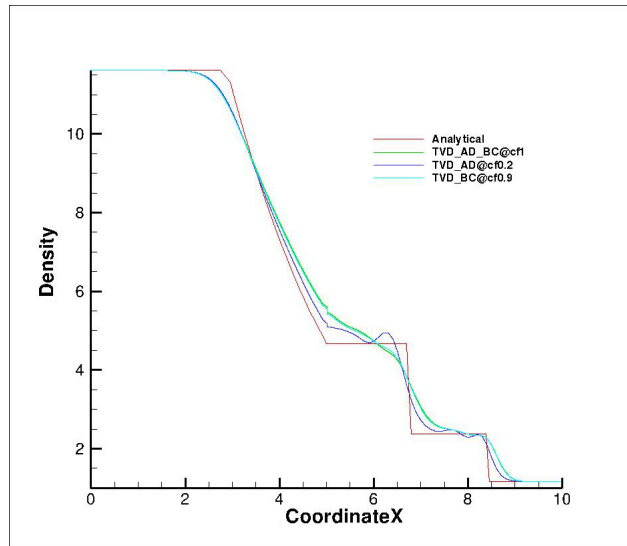


Figure 4.3: Density plot for TVD schemes to decide for better TVD method (Problem Case: 1 [3])

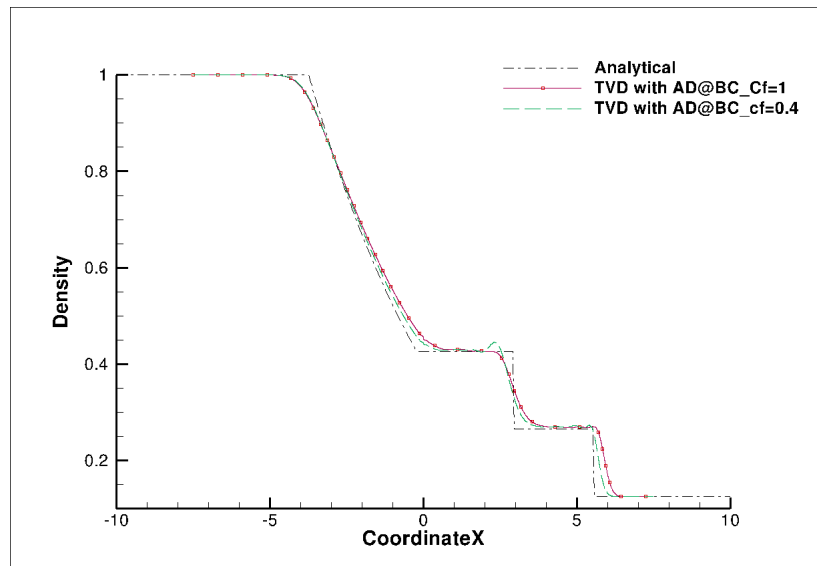


Figure 4.4: Density plot for TVD schemes to decide for correct coefficient for AV (Problem Case: Sec. 5.1)

4.6 Advection Upstream Splitting Method (AUSM)

AUSM is developed as a numerical inviscid flux function for solving a general system of conservation equations and especially it is used to simulate hyperbolic conservation equations. It is based on the upwind concept and was motivated to provide an alternative approach to other upwind methods, such as the Godunov method, flux difference splitting methods by Roe, and Solomon and Osher, flux vector splitting methods by Van Leer, and Steger and Warming. It was first presented by Liou and Steffen [20].

We present here the AUSM⁺-up of Liou (2006) [19]. The token used “-up” will be clear after the scheme formulation is made. The scheme has been formulated in terms of the time-dependent Euler equations and relies on splitting the flux vector \mathbf{F} into a convective component $\mathbf{F}^{(c)}$ and a pressure component $\mathbf{F}^{(p)}$. For the x-split three dimensional flux we have

$$\mathbf{F}(\mathbf{w}) = \begin{Bmatrix} \rho u \\ \rho u^2 + P \\ \rho uv \\ \rho uw \\ \rho u(E + P) \end{Bmatrix} = \begin{Bmatrix} \rho u \\ \rho u^2 \\ \rho uv \\ \rho uw \\ \rho uH \end{Bmatrix} + \begin{Bmatrix} 0 \\ P \\ 0 \\ 0 \\ 0 \end{Bmatrix} \equiv \mathbf{F}^{(c)} + \mathbf{F}^{(P)} \quad (4.26)$$

where, w represents conservative variables.

By introducing the Mach number and enthalpy

$$M = \frac{u}{a}, \quad H = \frac{E+P}{\rho}$$

we write,

$$\mathbf{F}^{(c)} = M \begin{Bmatrix} \rho a \\ \rho au \\ \rho av \\ \rho aw \\ \rho aH \end{Bmatrix} \equiv M \hat{\mathbf{F}}^{(c)} \quad (4.27)$$

The face straddles two neighboring cells labeled by subscripts “**L**” and “**R**”, respectively, namely left and right of the interface. The values of various variable at the face is denoted by subscript “1/2”. In defining the flux at face (say $\mathbf{F}_{1/2}$), the scheme take

$$\mathbf{F}_{1/2} = \mathbf{F}_{1/2}^{(c)} + \mathbf{F}_{1/2}^{(p)} \quad (4.28)$$

where the *convective flux component* is given by

$$\mathbf{F}_{1/2}^{(c)} = M_{1/2} [\hat{\mathbf{F}}^{(c)}]_{1/2} \quad (4.29)$$

with definition

$$[\bullet]_{1/2} = \begin{cases} [\bullet]_L & \text{if } M_{1/2} \geq 0 \\ [\bullet]_R & \text{if } M_{1/2} < 0 \end{cases}$$

The flux vector in Eq. 4.29 is upwinded as per the sign of face Mach number $M_{1/2}$. The convective terms are upstream-biased using the velocity implied in the face Mach Number. For this reason Liou and Steffen call their scheme AUSM, which stands for Advection Upstream Splitting Method.

The face Mach number is given by the splitting

$$M_{1/2} = M_L^+ + M_R^- \quad (4.30)$$

with the positive and negative component yet to be defined. The splitting of the pressure flux components depends on the splitting of the pressure itself, namely

$$P_{1/2} = P_L^+ + P_R^- \quad (4.31)$$

For the splitting of Mach number and pressure we follow the formulation presented in Liou (2006) [19]. The objective of doing this was to make the earlier version of AUSM to be uniformly valid for all speed regimes.

The final algorithm for flux calculation is given as follows *AUSM⁺ - up*.

1. We define

$$M_L = \frac{V_L}{a_{1/2}}, \quad M_R = \frac{V_R}{a_{1/2}}$$

$$\text{where, } V_i = \vec{V}_i \cdot \hat{n}$$

The corresponding speed of sound at the interface is given by

$$a_{1/2} = \frac{a_L + a_R}{2} \quad \text{or, } a_{1/2} = \min(\hat{a}_i, \hat{a}_R)$$

$$\text{where, } \hat{a}_i = \frac{a_L^*}{\max(a_L^*, V_L)}, \quad \hat{a}_R = \frac{a_R^*}{\max(a_R^*, -V_R)}$$

where, a^* is defined as the speed of sound based on total enthalpy(H_t). And is given as $a^* = \sqrt{\frac{2(\gamma-1)}{\gamma+1} H_t}$.

2. $\bar{M}^2 = \frac{V_L^2 + V_R^2}{2a_{1/2}^2}$,

$$M_o^2 = \min(1, \max(\bar{M}^2, M_\infty^2)) \in [0, 1],$$

$$f_a(M_o) = M_o(2 - M_o) \in [0, 1],$$

A pressure diffusion term M_p , is introduced to enhance calculations of low Mach number flow. It is defined as

$$M_p = -\frac{K_p}{f_a} \max(1 - \sigma \bar{M}^2, 0) \frac{P_R - P_L}{(\rho a)_{1/2}},$$

where, $0 \leq K_p \leq 1$, $\sigma \leq 1$ and $\rho_{1/2} = \frac{\rho_L + \rho_R}{2}$

Thus,

$$M_{1/2} = M_{(4)}^+(M_L) + M_{(4)}^-(M_R) + M_p$$

$$\text{where, } M_{(4)}^\pm(M) = \begin{cases} M_{(1)}^\pm & \text{if } |M| \geq 1 \\ M_{(2)}^\pm(1 \mp 16\beta M_{(2)}^\mp) & \text{otherwise} \end{cases}$$

where,

$$M_{(1)}^\pm(M) = \frac{1}{2}(M \pm |M|)$$

$$M_{(2)}^\pm(M) = \pm \frac{1}{4}(M \pm 1)^2$$

$$3. P_{1/2} = P_{(5)}^+(M_L)P_i + P_{(5)}^-(M_R)P_R + P_u$$

$$\text{where, } P_{(5)}^\pm(M) = \begin{cases} \frac{1}{M}M_{(1)}^\pm & \text{if } |M| \geq 1 \\ M_{(2)}^\pm[(\pm 2 - M) \mp 16\alpha MM_{(2)}^\mp] & \text{otherwise} \end{cases}$$

where, P_u is the velocity diffusion term similar to interface Mach number defined in step 2. $P_u = -K_u P_{(5)}^+(M_i)P_{(5)}^-(M_R)(\rho_L + \rho_R)(f_a a_{1/2})(u_R - u_L)$

where, $0 \leq K_u \leq 1$

$$4. \alpha = \frac{3}{16}(-4 + 5f_a^2) \in [-\frac{3}{8} + \frac{3}{16}],$$

$$\beta = \frac{1}{8}, K_p = 0.25, K_u 0.75 \text{ and } \sigma = 1.0$$

5. The mass flux at the interface has the following form

$$\dot{m}_{1/2} = a_{1/2} M_{1/2} \begin{cases} \rho_L & \text{if } M_{1/2} \geq 0 \\ \rho_R & \text{otherwise} \end{cases}$$

6. Finally, the flux vector is written as

$$\mathbf{F}_{1/2}(w) = \dot{m}_{1/2} \begin{cases} w_L & \text{if } w_{1/2} \geq 0 \\ w_R & \text{otherwise} \end{cases} + P_{1/2} \quad (4.32)$$

The value of the conservative variables w_L and w_R are the left and right states at the face. They are obtained using MUSCL extrapolation formula[2]. Considering non-uniform structured grid, we have used second order volume extrapolation. At the cells next to the boundary we use simple zero-order extrapolation, i.e. we have 1st order accuracy at the boundary. For the multi-block approach, we have maintained second-order accuracy at the block interface using center-difference approach. We have also tried with single and double upwind approach. But, the center-difference has been given best results in terms of accuracy and stability.

The scheme is called AUSM⁺-up. The suffix ‘‘u’’ is used to indicate the velocity diffusion term (P_u) included in the pressure split of flux vector and the suffix ‘‘p’’ is used to indicate the pressure diffusion term (M_p) included in the convective split part of the flux vector.

4.7 Closure

In this chapter we have seen the various numerical schemes which can be used for the study of compressible flows. Also, we saw the detail formulation of central scheme such as MacCormack and TVD MacCormack and the upwind based scheme AUSM⁺-up. In the following chapter, we will see the results, these various schemes gives and will also try to derive appropriate conclusions of which scheme does better.

Chapter 5

Results And Discussion

In this chapter we present the results obtained using the solver for different geometric configurations and for different regimes of flow. The results are all compared with standard benchmark and/or analytical results. The MacCormack scheme with artificial viscosity (referred as *MacCormack*), with TVD implementation (referred to as *TVD*) and AUSM⁺-up scheme (referred as *AUSM+*) have been tested. The aim is to study the performance of these scheme and validate the compressible flow solver module of Anupravaha. Since, the solver is 3D-based, for all the 2D test-cases we have given a minimum of 4 cells thickness in the z-direction and symmetry boundary conditions are applied on the two boundaries normal to the z-direction.

5.1 Shock-tube Problem

Set-up

This problem ((P.N.352), [13]) comprises of a tube initially containing two regions of a stationary gas at different pressures, separated by a diaphragm. At $t = 0$, the diaphragm is removed instantaneously so that the pressure imbalance causes a unsteady flow containing a moving expansion fan, shock and contact discontinuity. The problem can be solved analytically as a 1-D case[31]. However, we solve the computational problem as a 2-D case, and compare it with the 1-D analytical solution. The computational results were obtained on a uniform grid of $\Delta x = 0.1\text{m}$. A Courant number of 0.75 has been used.

Initial Condition

IC	Part 1	Part 2
Pressure	100000 Pa	10000 Pa
Temperature	300K	300K
u velocity	0	0
v velocity	0	0
w velocity	0	0

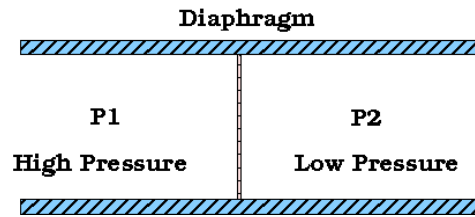
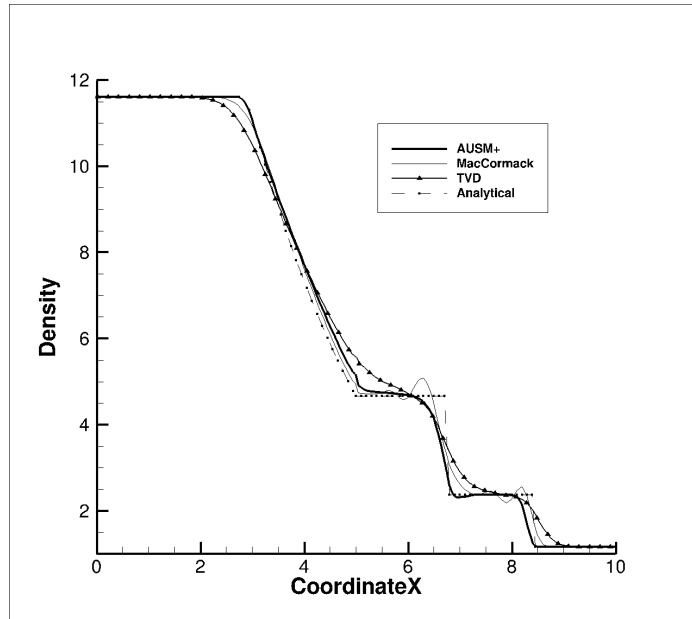


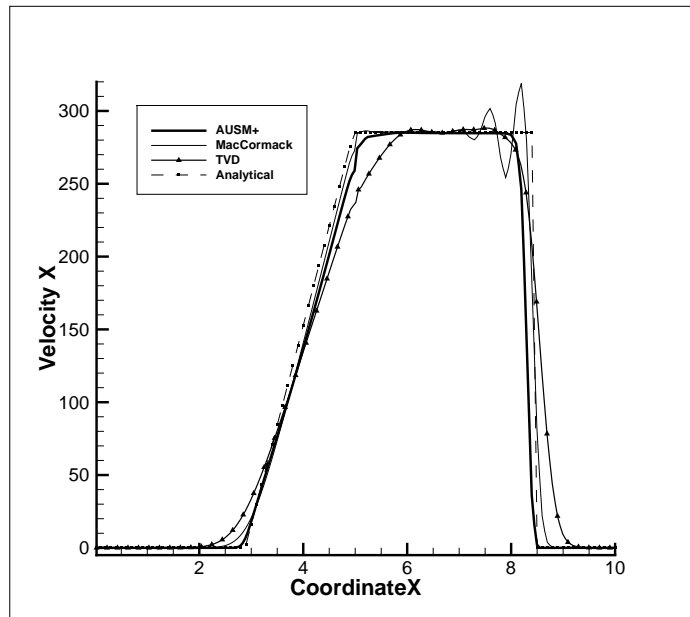
Figure 5.1: Shocktube

Boundary Condition All boundaries are (slip) walls, while symmetry boundary condition are implemented on surfaces on the z-plane.

The calculation was done to compare with analytical results previously derived for the shocktube problem [31]. The analytical solution to the shock-tube problem at $t = 0.0061s$ is compared to the computational result at the centerline of the tube (see Fig. 5.2). It can be seen that AUSM+ scheme has non-oscillatory solutions compared to the MacCormack scheme. The advantage of using an upwind scheme compared to a central scheme is self-evident here. The expansion shock occurring on the left has been captured accurately. However, the AUSM+ scheme is less accurate compared to MacCormack in capturing the position of the compression shock. The MacCormack scheme with TVD has been successful in removing the oscillatory solutions and is too diffusive. It has less accuracy compared to the other schemes. Based on these and other tests, in rest of this thesis only the results from AUSM+ and MacCormack scheme (with artificial viscosity) are presented.



(a) Density



(b) Velocity X

Figure 5.2: Plot along centre-line of shock-tube at 0.0061s

5.2 Supersonic flow over a wedge

We now consider the supersonic flow over a 2-D wedge with wedge angle 15° , as shown in Fig. 5.3. The inflow conditions are summarized in Table 5.2 and the present results have been compared with the analytical solution obtained from the standard $(\theta - \beta - M)$ chart and the analytical oblique shock relationships (Ch. 3 of [31]). Courant number of 0.6 and 0.4 are used for AUSM+ and MacCormack, respectively.

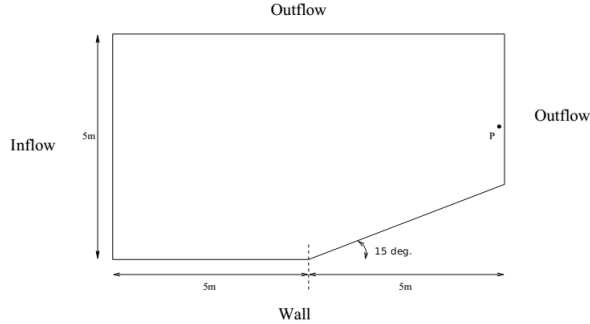


Figure 5.3: Computational domain.

Boundary Conditions

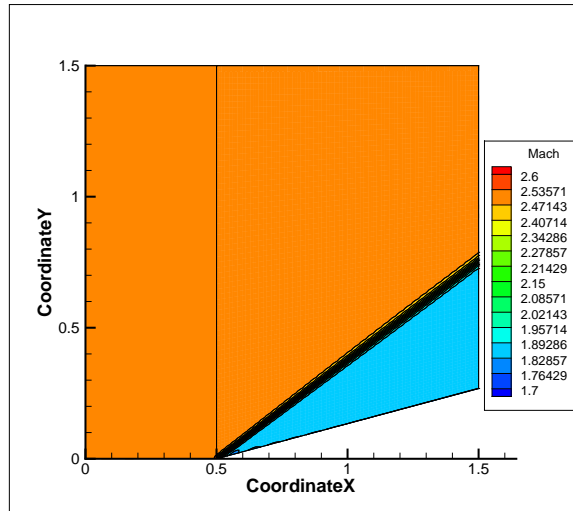
Quantity	Inflow	Outflow
Pressure	101353 Pa	(101353 Pa*)
Temperature	288.9 K	-
U velocity	2.5 Mach	-
V velocity	0	-
W velocity	0	-

Table 5.1: Boundary conditions for supersonic wedge

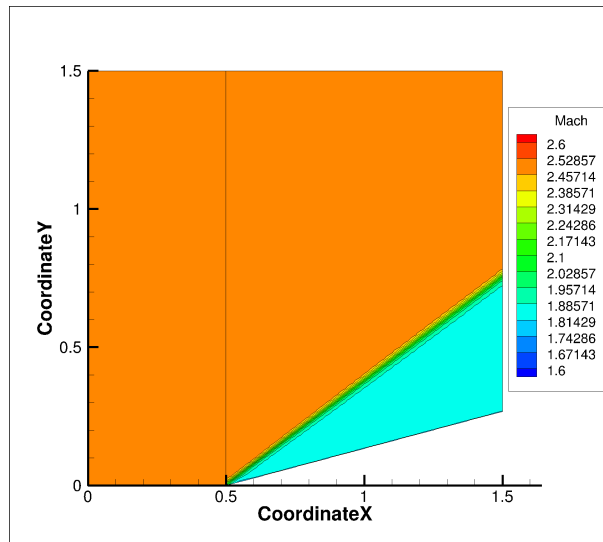
*Note: The solver explicitly asks for an outflow pressure but imposes this condition only if the flow is sub-sonic there.

The steady-state contours of Mach number and static pressure are shown in Fig. 5.4. Under the same flow condition the contours obtained by numerical computation done in Hirsch's book [1] is also shown in Fig. 5.4. The results downstream of the shock has been tabulated in Table 5.2 where, $P2/P1$ corresponds to the downstream and upstream pressure ratio. Point P refers to the point (1.495, 0.3) on the outflow plane. The analytical results are also presented. The pressure contours are shown in Fig. 5.5.

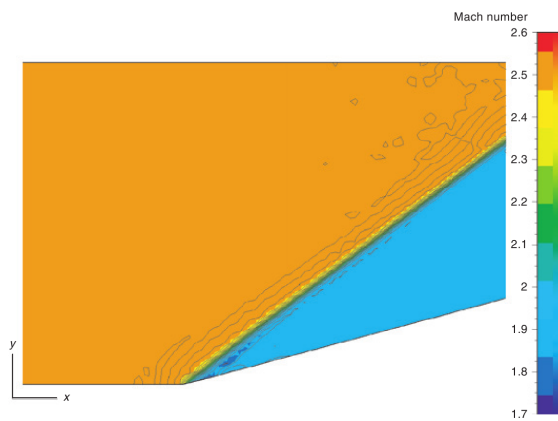
Both the MacCorMack and AUSM+ gives similar results and both have high accuracy, corresponding with the analytical results. AUSM+ scheme however gives a solution with less oscillations.



(a) MacCormack Scheme

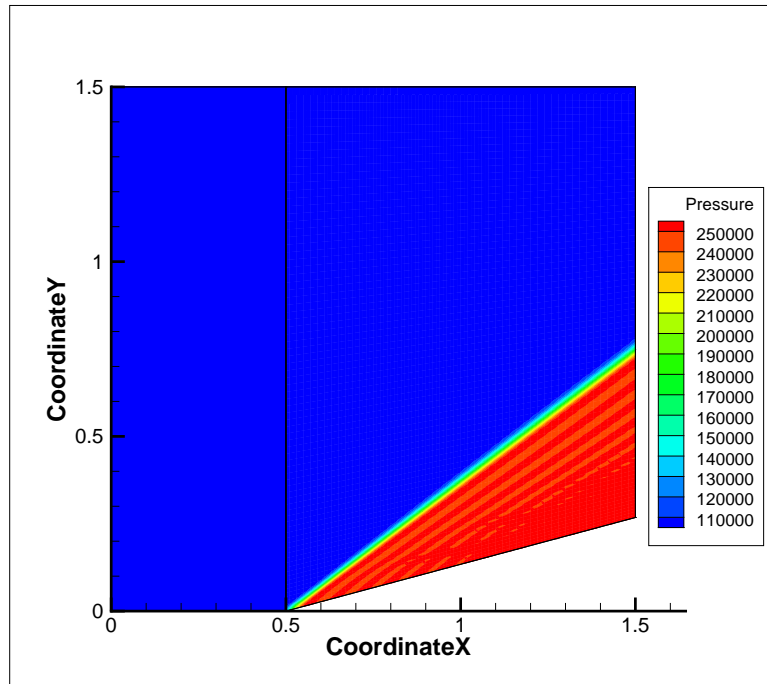


(b) AUSM+ Scheme.

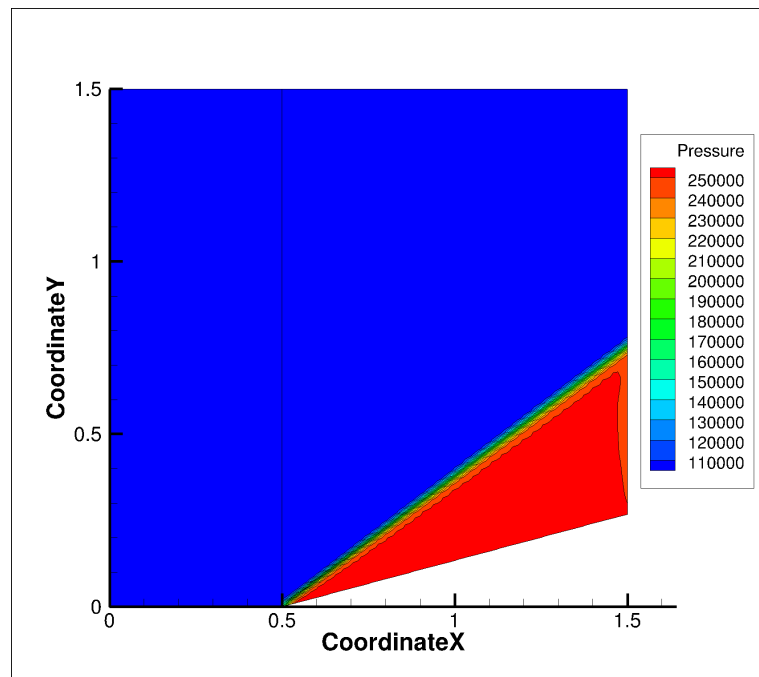


(c) Reference Hirsch [1].

Figure 5.4: Contours for Mach Number



(a) MacCormack Scheme



(b) AUSM+ Scheme.

Figure 5.5: Contours for Pressure

Ratio's	Analytical	MacCormack	AUSM+
P_2/P_1	2.468	2.468	2.468
T_2/T_1	1.322	1.322	1.317
ρ_2/ρ_1	1.867	1.866	1.869
Mach	1.874	1.873	1.877
Shock angle (in deg.)	36.945	35.976 – 38.213	35.85 – 37.95

Table 5.2: Validation with analytical solution

5.3 Steady Shock Reflection Problem

Considered here is the problem of an oblique shock, generated by a supersonic flow over a sharp wedge, and the subsequent reflections between the wedge surface and a flat plate located oblique to the wedge. The problem is taken from the study done by Wang et al. [32]. The computational domain for the problem is shown in Fig. 5.6. The inlet flow has a Mach no. of 2.9.

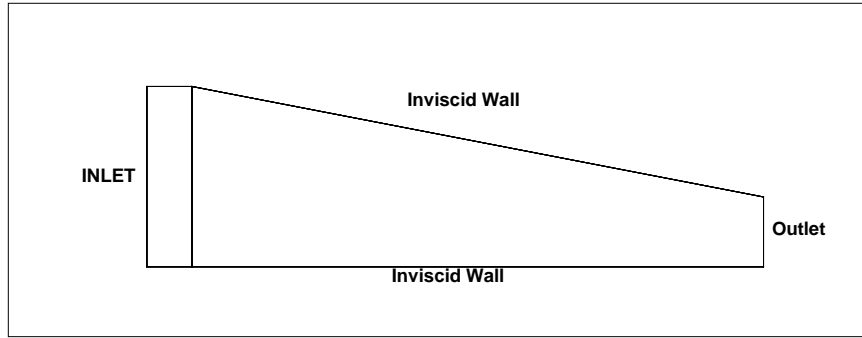


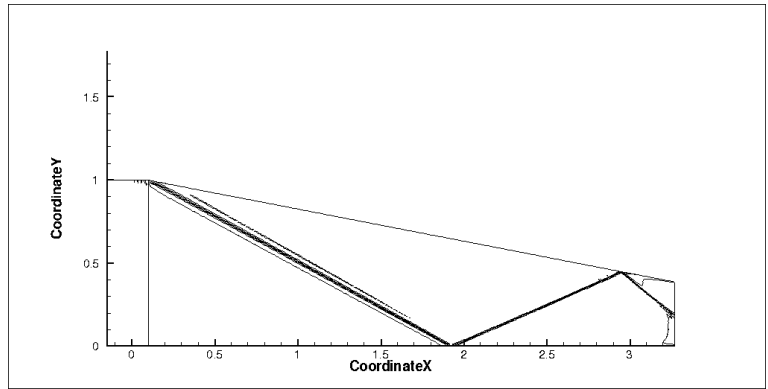
Figure 5.6: Computational domain.

Boundary Conditions

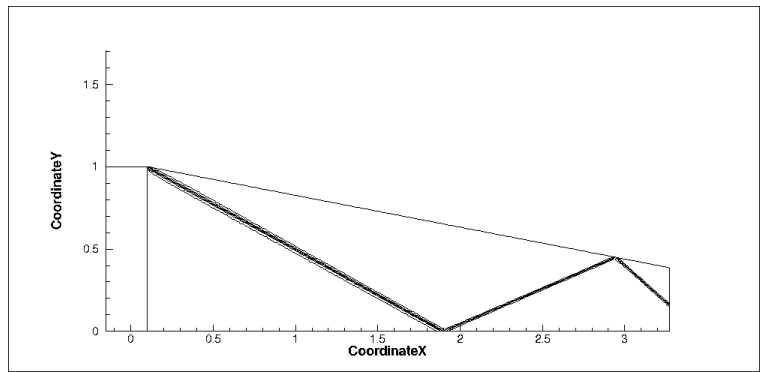
Quantity	Inlet	Outflow
Pressure	0.714 Pa	0.714 Pa
Temperature	0.002488 K	-
U velocity	2.9	-
V velocity	0	-
W velocity	0	-

Table 5.3: Boundary conditions for Shock reflection Problem

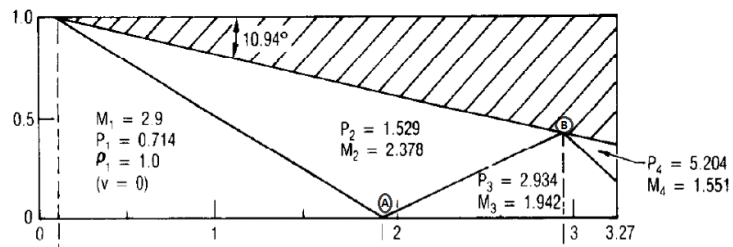
The result is validated with the reference. The Mach contours from the MacCormack and AUSM+ schemes of Anupravaha and the analytical results with second order solution of [32] are shown in Fig. 5.7. The positions of the shock has been accurately predicted by the present calculations. The AUSM+ scheme is less diffusive and non-oscillatory and is seen to be more accurate than the MacCormack Scheme. The values of pressure and Mach number at various points shown in the exact solution inset in Fig. 5.7 are tabulated in Table 5.4. Both the AnuPravaha schemes are seen to be accurate.



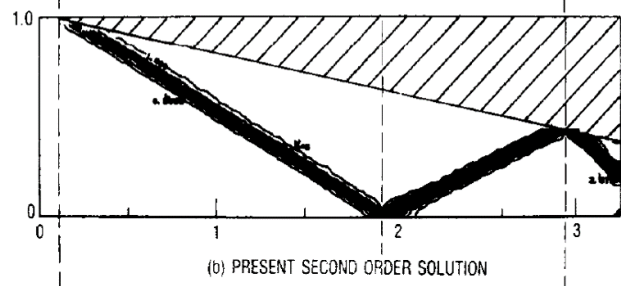
(a) MacCormack Scheme (AnuPravaha)



(b) AUSM+ Scheme. (AnuPravaha)



(a) EXACT SOLUTION



(b) PRESENT SECOND ORDER SOLUTION

(c) Reference Wang et.al [32].

Figure 5.7: Contours for Mach Number

Quantity	MacCormack	AUSM+	Analytical
Region 1	$P = 0.714$ $M = 2.9$	$P = 0.714$ $M = 2.9$	$P = 0.714$ $M = 2.9$
Region 2	$P = 1.528$ $M = 2.378$	$P = 1.528$ $M = 2.378$	$P = 1.529$ $M = 2.357$
Region 3	$P = 2.9333$ $M = 1.943$	$P = 2.933$ $M = 1.943$	$P = 2.934$ $M = 1.942$
Region 4	$P = 5.204$ $M = 1.552$	$P = 5.204$ $M = 1.551$	$P = 5.204$ $M = 1.551$
Loc. A	1.89units	1.9units	1.9units
Loc. B	2.94units	2.95units	2.9units

Table 5.4: Values in various regions : Loc. A refers to position of the 1st reflection, Loc. B refers to position of 2nd reflection

5.4 External Flow over NACA0012 Airfoil

To validate the code for complex geometry, we have taken the case of NACA 0012 airfoil. We study the external flow at Mach Number of 0.5, 0.8 and 1.2. Thus, we try to validate our code against the subsonic, transonic and supersonic regimes of the flow, a major objective of this thesis. The computational domain for the NACA Aerofoil considered is shown below. The following three are the cases considered.

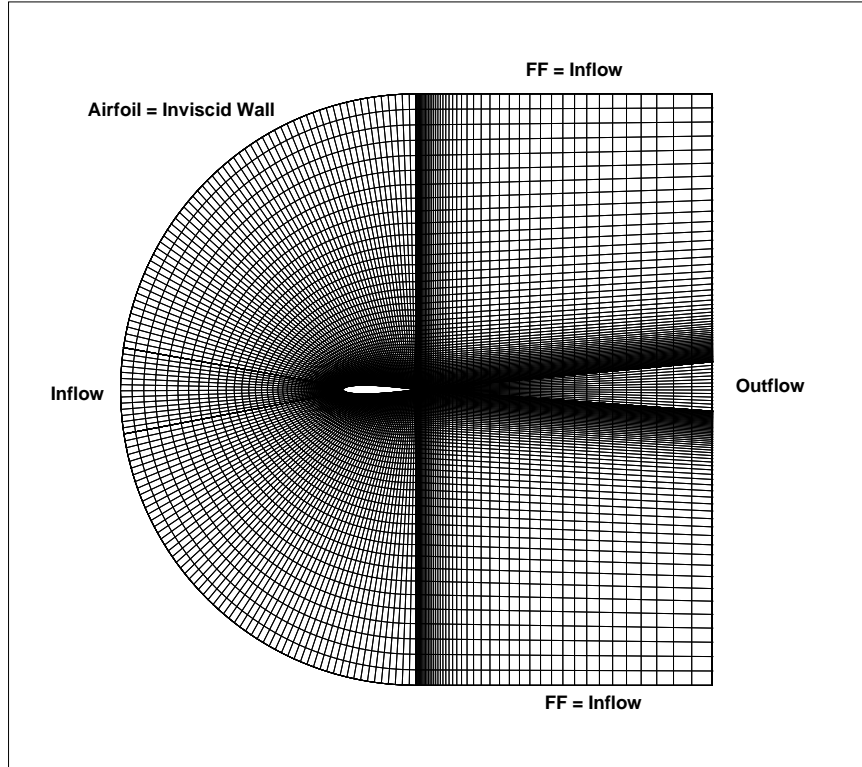


Figure 5.8: Computational domain and Mesh.

5.4.1 Mach 0.5, Angle of Attack ($\alpha = 0^\circ$)

This is a subsonic case involving external flow. We have used velocity-driven boundary condition for inlet and far-field. The boundary conditions are tabulated in Table 5.5.

Quantity	Inflow	Outflow
Pressure	100000 Pa	100000 Pa
Temperature	300 K	-
U velocity	173.594	-
V velocity	0	-
W velocity	0	-

Table 5.5: Boundary conditions for NACA 0012 $M = 0.5$, $\alpha = 0^\circ$

The result is validated against the results obtained by Furmanek [33]. The value of coefficient of pressure along the airfoil surface is plotted and compared with the reference in the Fig. 5.11. The Mach contours are shown in Fig. 5.10. In most airfoil cases, it is easier to accurately capture pressure compared to velocities or Mach numbers, which is a vector. An unrealistic boundary layer occurred at the airfoil surface due to some bugs in the code like improper update of interface slabs value. One such result can be seen for $M = 1.2$ in the Fig. 5.9. The major bug was the index calculation for the north and south face in the compressible code was not matching the data structure notation of AnuPravaha Solver. Thus, in Fig. 5.12, we next compare the Mach Number plot with the results obtained from FLUENT using the implicit version of the AUSM+ scheme. MacCormack scheme is more accurate compared to AUSM+ scheme.

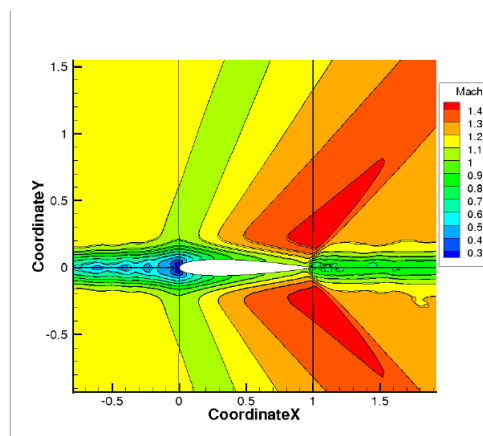
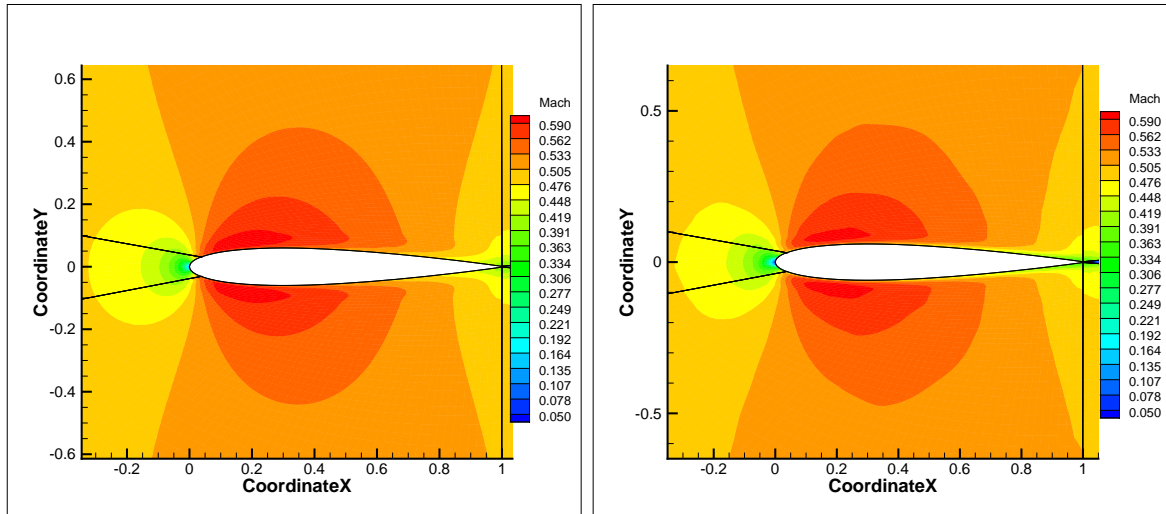


Figure 5.9: Mach Number contours with the bug in the code for $M = 1.2$ and $\alpha = 0^\circ$.



(a) MacCormack Scheme

(b) AUSM+ Scheme.

Figure 5.10: Mach Contour for $M=0.5$

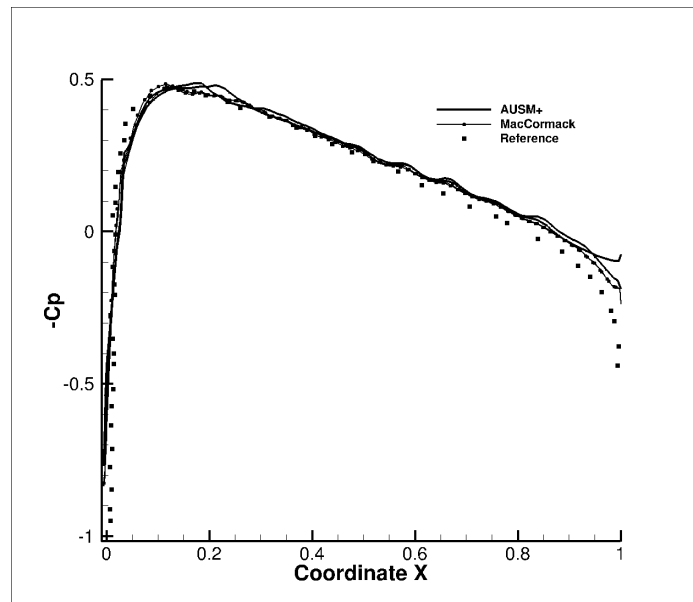


Figure 5.11: Pressure Coefficient along the wall: Comparison for $M = 0.5$ and $\alpha = 0^\circ$.

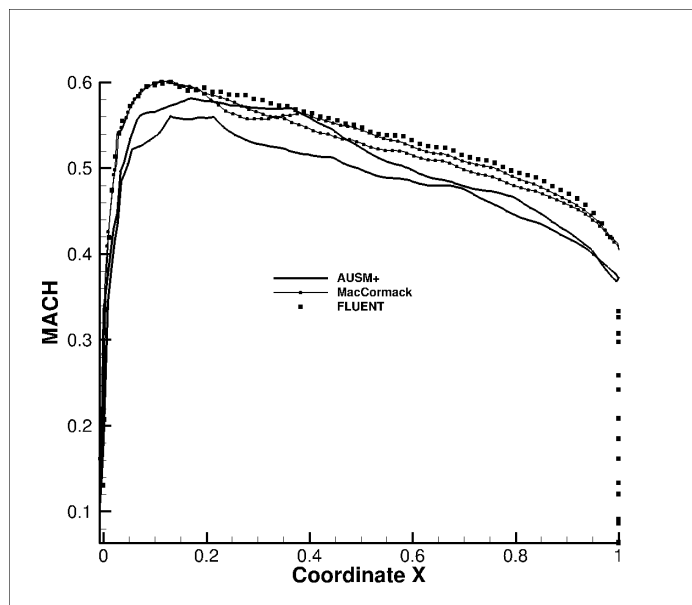


Figure 5.12: Mach Number along the wall for $M = 0.5$ and $\alpha = 0^\circ$.

5.4.2 Mach 0.8, Angle of Attack ($\alpha = 1.25^\circ$)

This is a transonic case of external flow. We have used velocity-driven boundary condition for inlet and far-field. The boundary conditions are tabulated in Table 5.7.

Quantity	Inflow	Outflow
Pressure	101325 Pa	101325 Pa
Temperature	273.15 K	-
U velocity	264.967	-
V velocity	5.782	-
W velocity	0	-

Table 5.6: Boundary conditions for NACA 0012 $M = 0.8$, $\alpha = 1.25^\circ$

The result is validated against the results obtained by Arias et al. [34]. The value of the coefficient of pressure along the wall is plotted and is compared with the reference in Fig. 5.13. The MacCormack scheme failed to give a stable and accurate result. The X-Mach contours are compared with that of the reference in Fig. 5.14 and Fig. 5.15. Our results are in well agreement with the literature.

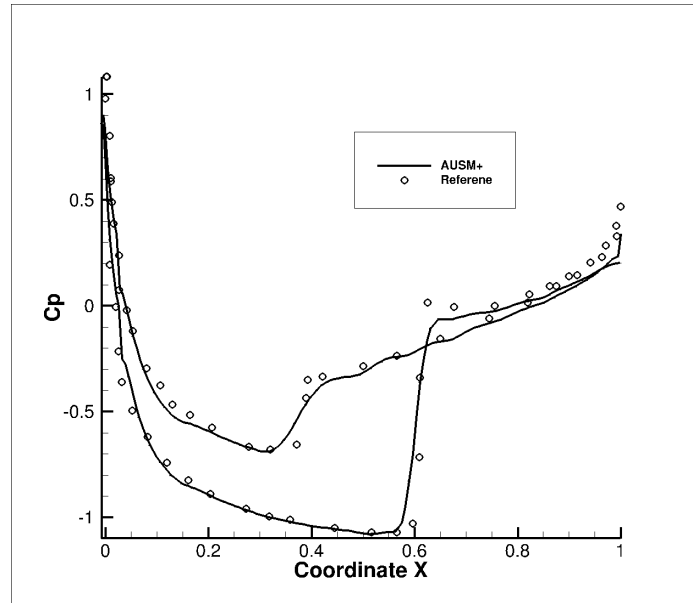


Figure 5.13: Pressure Coefficient along the wall: Comparison for $M = 0.8$ and $\alpha = 1.25^\circ$.

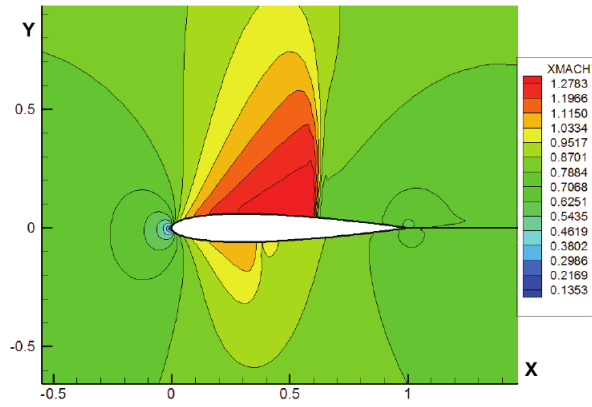


Figure 5.14: Mach Contours in x-direction for $M = 0.8$, $\alpha = 1.25^\circ$ from [34].

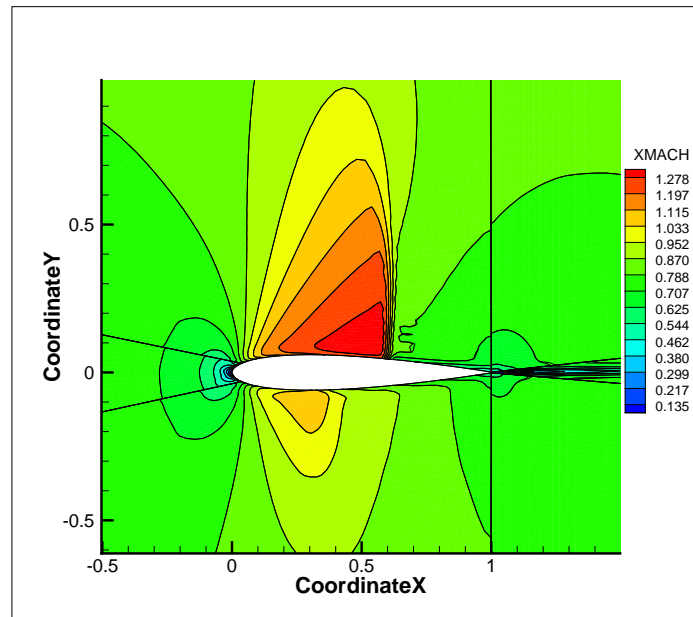


Figure 5.15: Mach Contours in x-direction using AUSM+ Scheme of AnuPravaha.

5.4.3 Mach 1.2, Angle of Attack ($\alpha = 0^\circ$)

This is a supersonic case involving external flow. We have used supersonic inflow boundary condition at the inlet and far-field.

Quantity	Inflow	Outflow
Pressure	100000 Pa	100000 Pa
Temperature	300 K	-
U velocity	416.63	-
V velocity	0	-
W velocity	0	-

Table 5.7: Boundary conditions for NACA 0012 $M = 1.2$, $\alpha = 0^\circ$

The result is validated against the results obtained by Arias et al. [34]. The value of coefficient of pressure along the wall is plotted and is compared with the reference in the Fig. 5.16. The Mach contours from AnuPravaha are shown in Fig. 5.17 and Fig. 5.18 for MacCormack and AUSM+ schemes. In Fig. 5.19, we next compare the Mach Number plot with the results obtained from FLUENT using the implicit version of AUSM+ scheme. We see here the AnuPravaha MacCormack and AUSM+ scheme gives similar results and are close to the FLUENT solutions.

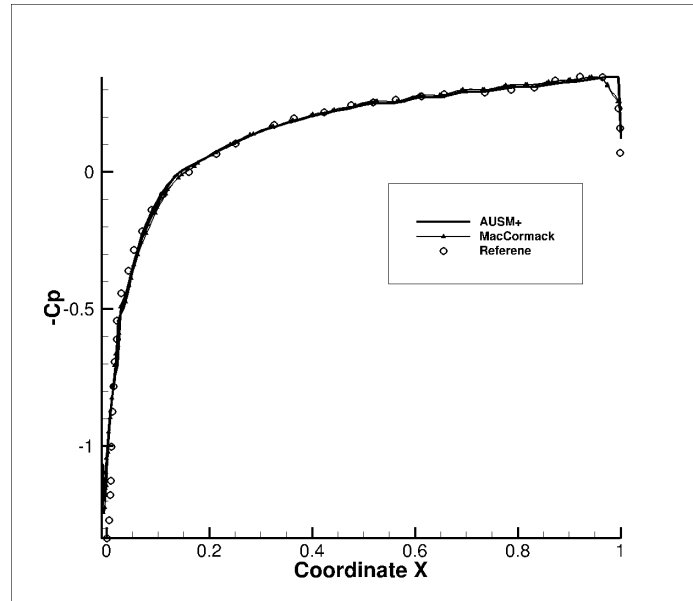


Figure 5.16: Pressure Coefficient along the wall: Comparison for $M = 1.2$ and $\alpha = 0^\circ$.

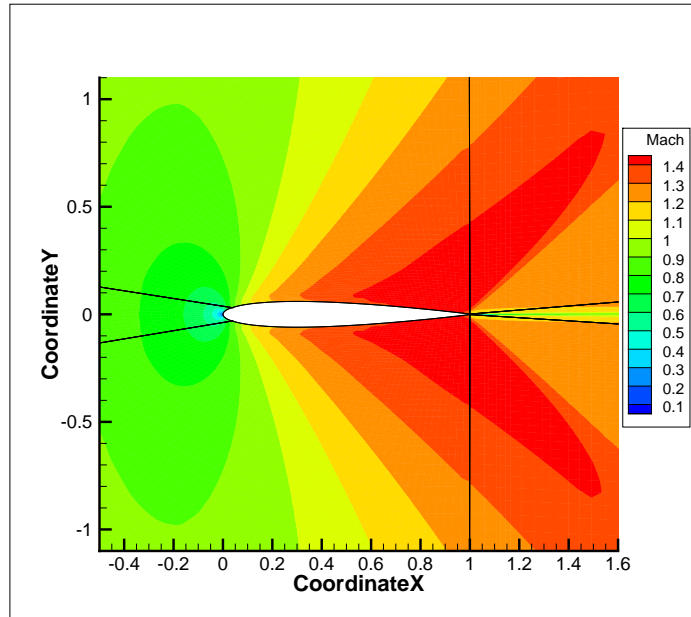


Figure 5.17: Mach Contours for $M = 1.2$ using MacCormack Scheme (AnuPravaha).

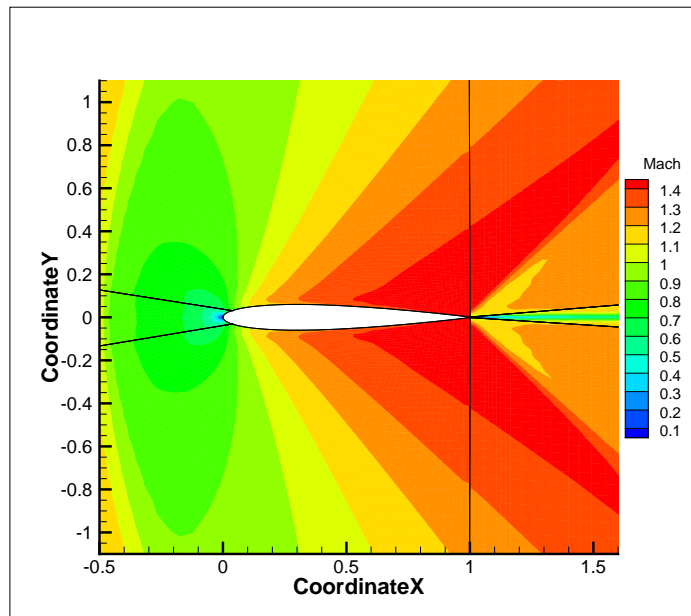


Figure 5.18: Mach Contours for $M = 1.2$ using AUSM+ Scheme (AnuPravaha).

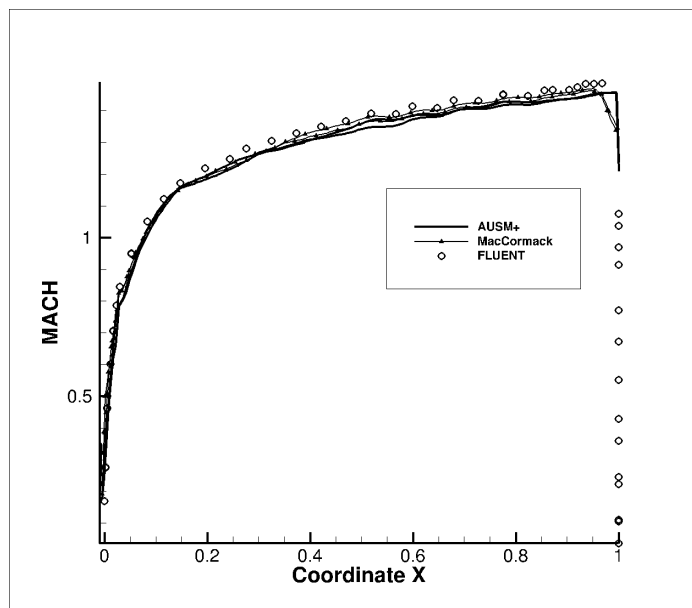


Figure 5.19: Mach Number along the wall for $M = 1.2$ and $\alpha = 0^\circ$.

5.5 Internal flow in a channel with a circular Bump

We now take a case considering internal flow. It consists of a channel of height L and length $3L$, with a circular arc of length L and thickness equal to $0.1L$, along the bottom wall, as shown in Fig. 5.20. For the subsonic and transonic cases we use a pressure-driven inlet boundary condition. For initializing the flow-field, we have used free-stream conditions. The inlet x-velocity is calculated by numerical-extrapolation from the interior domain. Its specification in the problem below is indicative for Mach Number of the flow at the inlet and is used in numerical algorithm. We used a Courant number of 0.5 for cases below. The MacCormack scheme again has convergence difficulty for the transonic case.

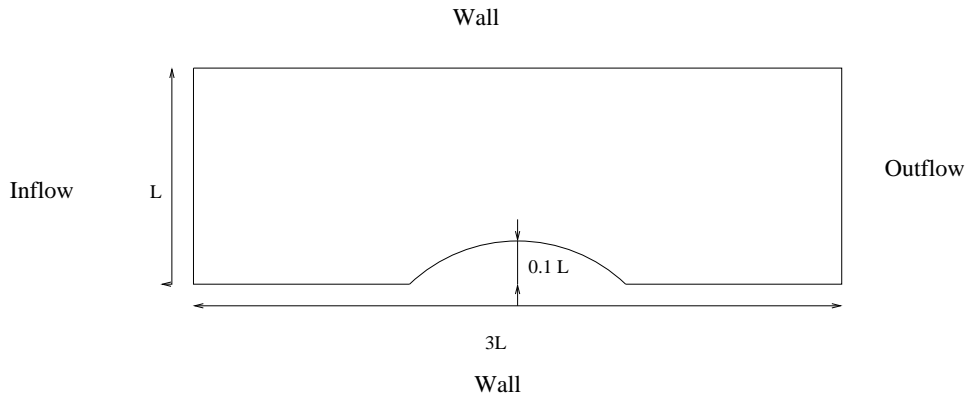


Figure 5.20: Computational Domain.

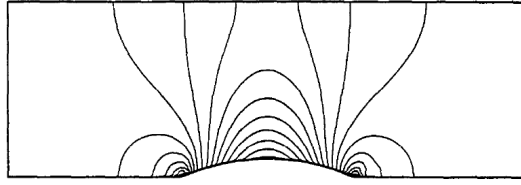
5.5.1 Subsonic Case

The inlet Mach number is chosen equal to 0.5. We provide total pressure and total temperature at inlet w.r.t. the static condition, so as to get inlet Mach Number equal to 0.5. This is as per our boundary condition discussion is Sec.3.2.2 under pressure-driven flow section. At outflow we use the free-stream condition (static condition). The boundary conditions are summarized in Table 5.8 and the solver results have been compared with the study done by Rincon et al. [35].

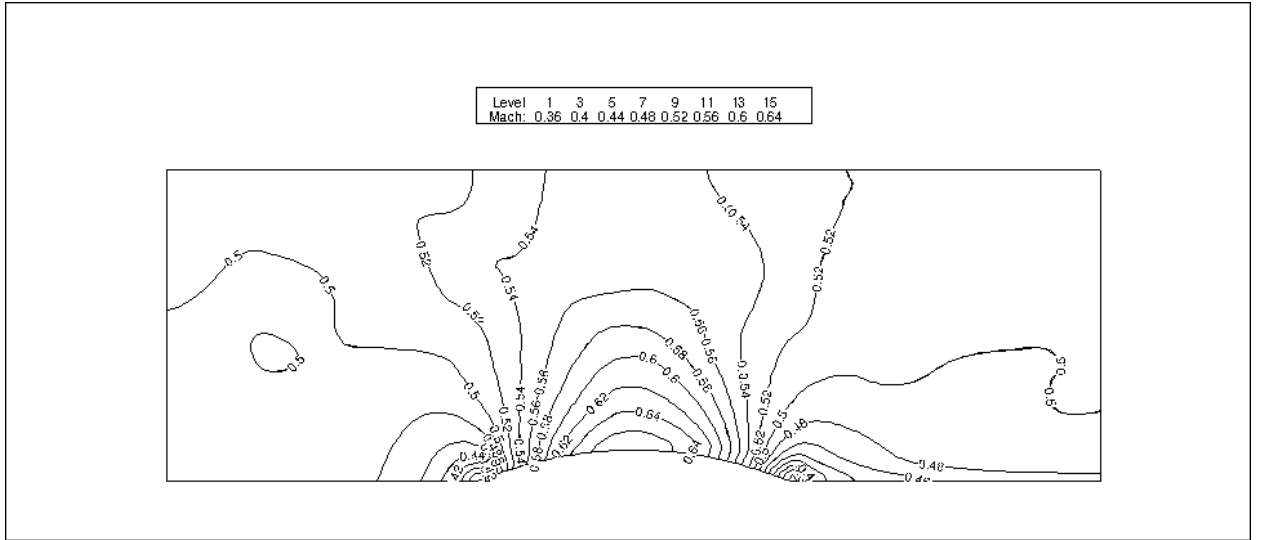
Quantity	Inflow	Outflow
Pressure	120141.8 Pa	101300 Pa
Temperature	302.4 K	288K
U velocity	174.287	-
V velocity	0	-
W velocity	0	-

Table 5.8: Boundary conditions for subsonic bump

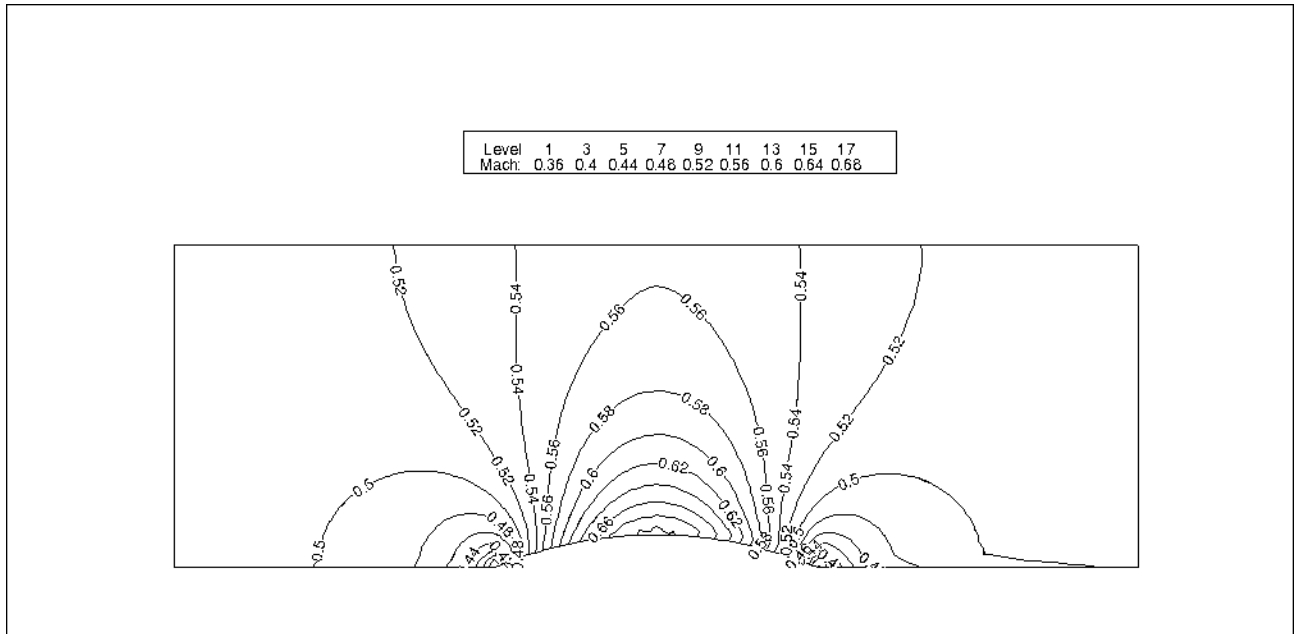
The comparison for Mach contours for MacCormack, AUSM+ and the reference is shown in Fig. 5.21. Fig. 5.22 shows the variation in Mach number along the upper and lower walls.



(a) Isomach lines from [35]



(b) AUSM+ Scheme (AnuPravaha)



(c) MacCormack Scheme (AnuPravaha)

Figure 5.21: Mach Contour for $M=0.5$

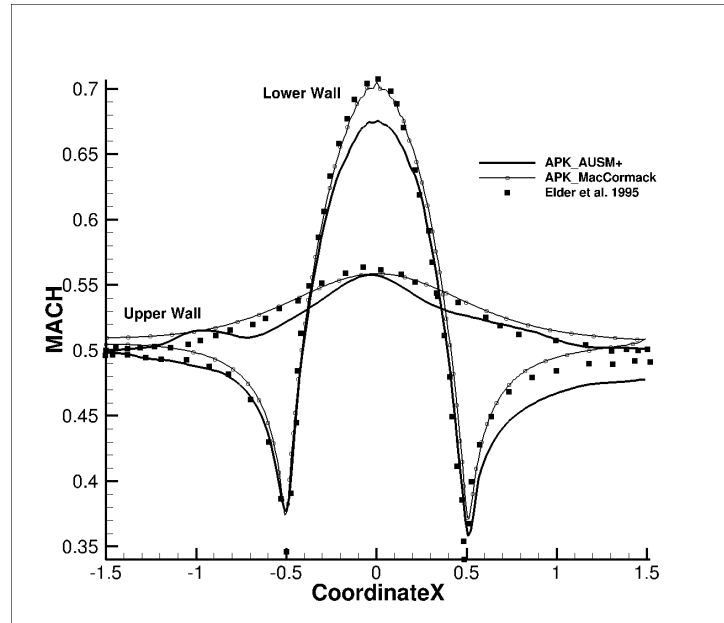


Figure 5.22: Variation of Mach number along lower and upper wall ($M = 0.5$)

As we do not get any shock, the solution is symmetric about the midchord. The result shows that in subsonic regime the MacCormack scheme does better than the AUSM+ scheme. The Courant number used for both the case is 0.5.

5.5.2 Transonic Case

Here, we impose inlet Mach number equal to 0.675. At the outlet the free-stream condition (static condition) is applied. The boundary conditions are summarized in Table 5.9 and the solver results has been compared with Rincon et al. [35].

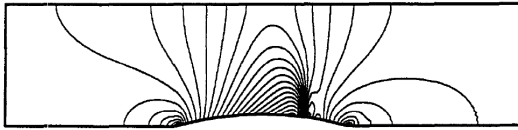
Boundary Conditions

Quantity	Inflow	Outflow
Pressure	137489.425 Pa	101300 Pa
Temperature	314.136 K	288K
U velocity	239.81	-
V velocity	0	-
W velocity	0	-

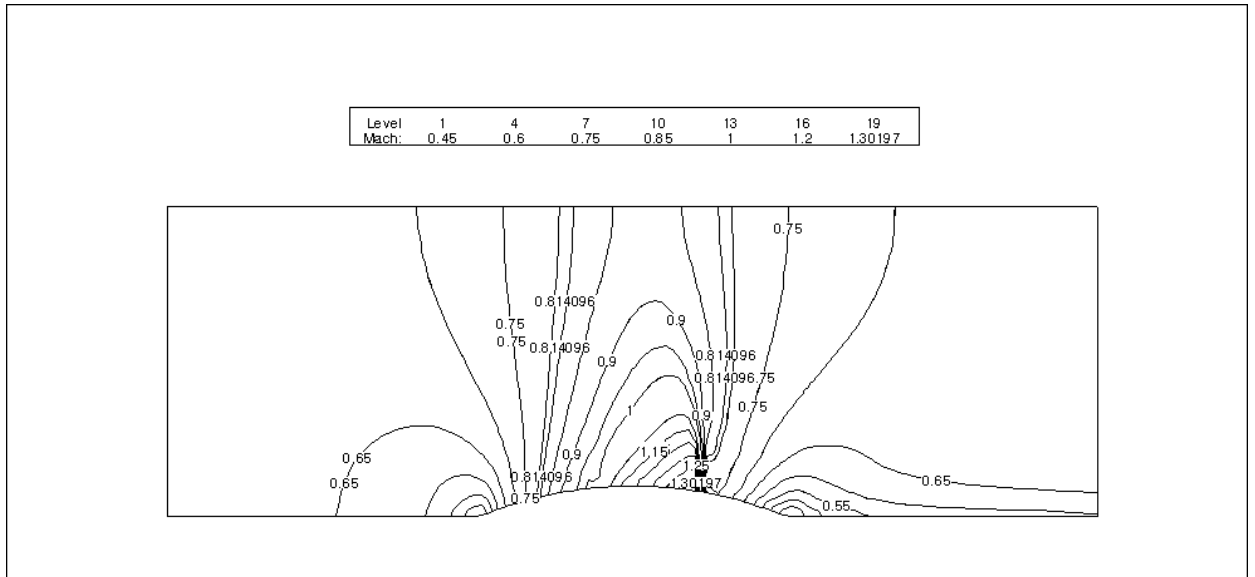
Table 5.9: Boundary conditions for transonic bump

The comparison for Mach contours for MacCormack, AUSM+ and the reference is shown in Fig. 5.23. Fig. 5.24 shows the variation in Mach number along the upper and lower walls. In Fig. 5.23 we can observe that the subsonic inlet flow is first expanded to a supersonic region and then a shock wave is formed to match the subsonic condition at the outlet. The maximum Mach number immediately before the shock is 1.36 and 1.4 for AUSM+ and MacCormack scheme. These results are well in agreement with the reference solutions from literature [35, 36]. The Courant number used for both the case is 0.5. MacCormack scheme is quite unstable in transonic regime, one has to use a lower Courant number.

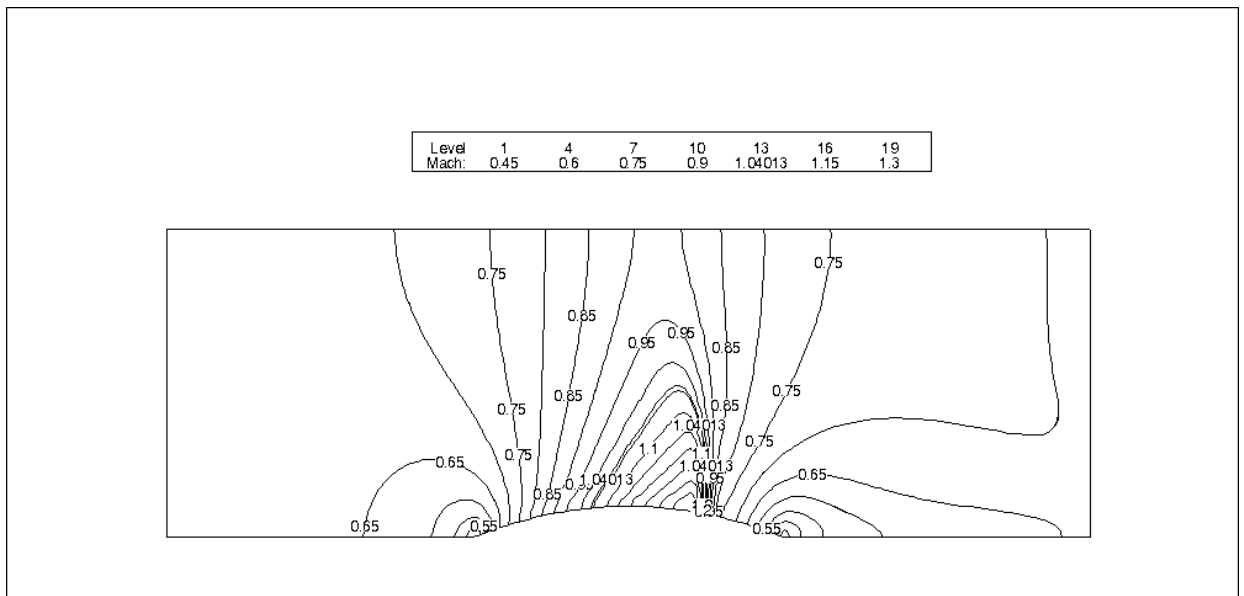
Note: For the Supersonic case we would have a supersonic condition at the exit, and the case would be similar to the steady shock reflection problem validated earlier in Sec. 5.6. So, the case was not done here.



(a) Isomach lines from [35]



(b) AUSM+ Scheme (AnuPravaha)



(c) MacCormack Scheme (AnuPravaha)

Figure 5.23: Mach Contour for $M=0.675$

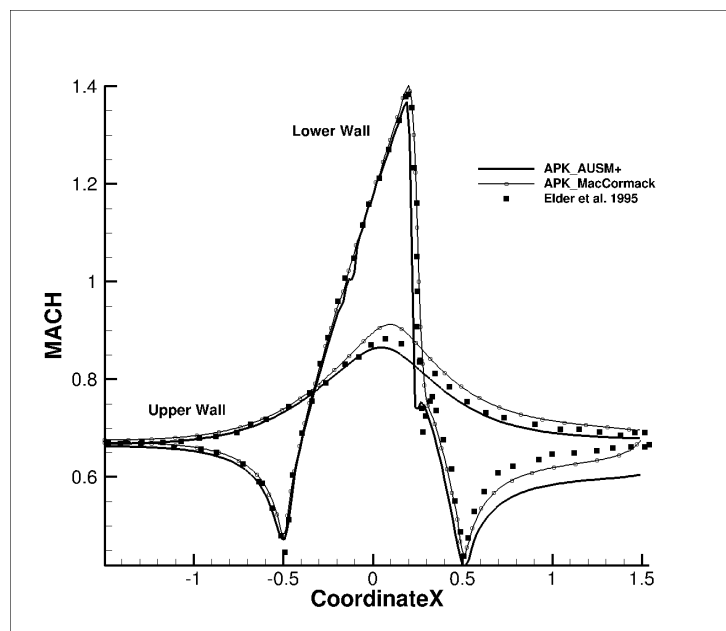


Figure 5.24: Variation of Mach Number along lower and upper wall ($M = 0.675$)

5.5.3 3D Case: Flow over re-entry capsule (Mach no. = 5, AOA = 4.66°)

A ballistic reentry capsule has been considered to validate the solver for a complex geometry. The vehicle consists of a blunt bicone with 20/25 degree cone angles. All the dimensions are shown in Fig. 5.25. The mesh and the computational domain are shown in Fig 5.26. Inlet, outlet and inviscid wall has been shown through red, green and blue colour respectively. The free-stream pressure and temperature are 833Pa and 63K, respectively. Free-stream Mach number is taken as 5.0 with angle of attack of 4.66. We specify free-stream pressure at outflow, which actually has no role to play for a supersonic exit. The boundary conditions based on these are summarized in Table 5.10. We validate the result with the study done by [37]. In this study, the wind tunnel data [38] has been used for validation. We have also included the result of the same test done using FLUENT (explicit AUSM plus scheme).

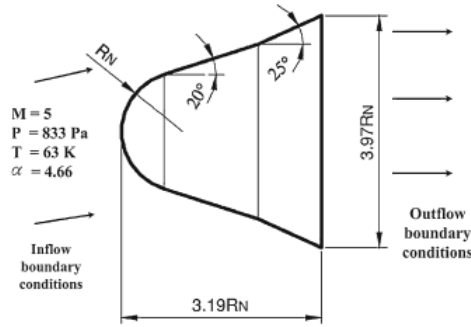


Figure 5.25: Re-entry vehicle model dimensions

Boundary Conditions

Quantity	Inflow	Outflow
Pressure	833 Pa	833 Pa
Temperature	63 K	-
U velocity	792.88	-
V velocity	64.63	-
W velocity	0	-

Table 5.10: Boundary conditions

Fig. 5.27 shows the density contours. We can see from the density contour the presence of bow shock. The numerical diffusion observed for the MacCormack scheme is larger compared to AUSM+ schme. On the windward side the formation of second shock is more pronounced in comparison to leeward side. The plot of C_p distribution along the capsule wall is shown in Fig. 5.28. The x-axis is the Coordinate X of the flow domain along the capsule wall. We can observe the higher pressure plot corresponds to the windward side. At stagnation point, we get the maximum pressure and pressure remains constant

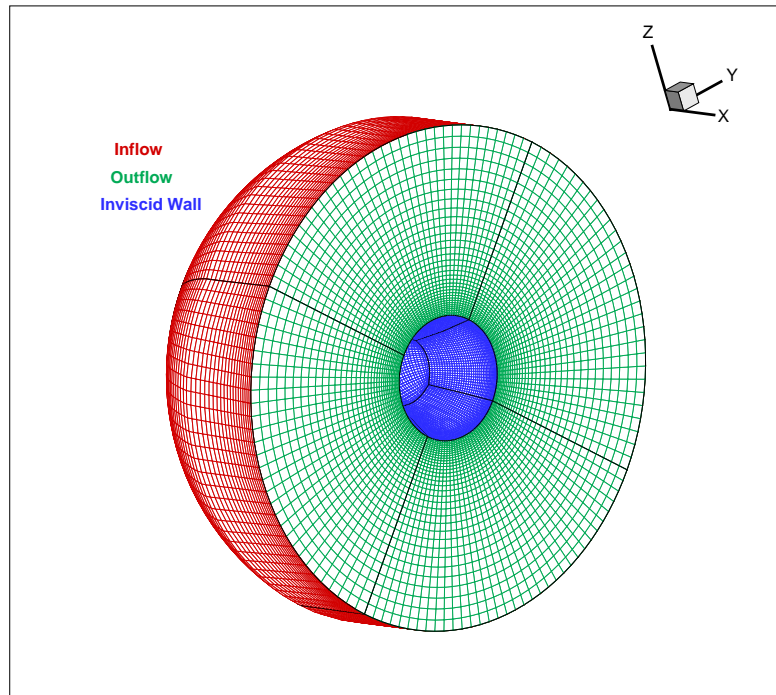
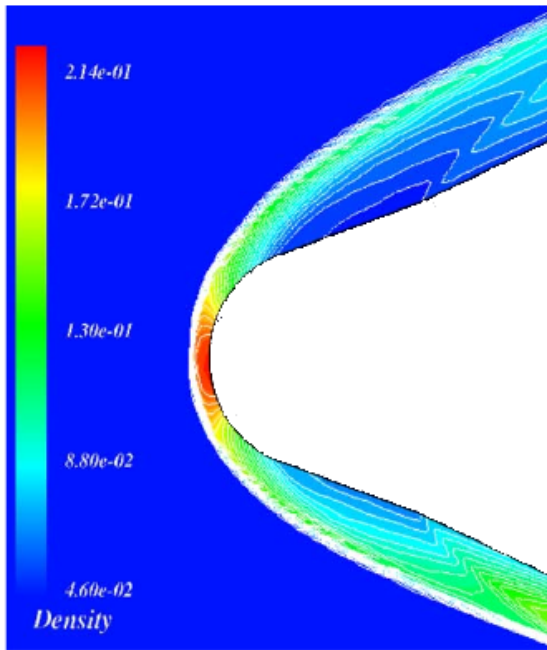
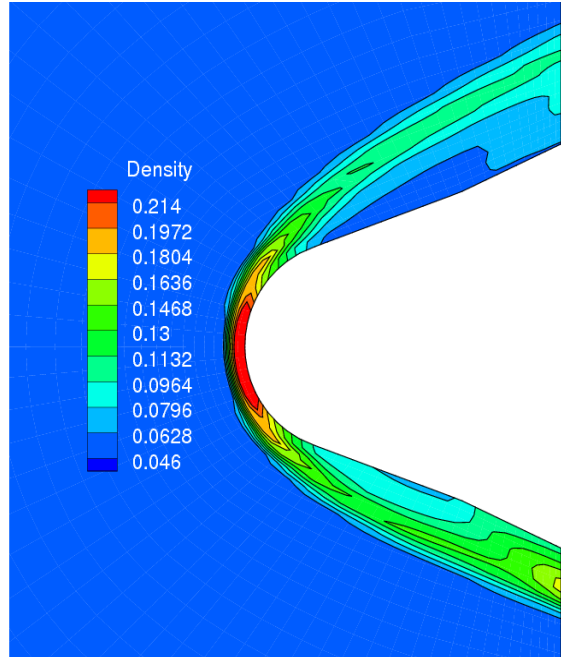


Figure 5.26: Computational domain and mesh

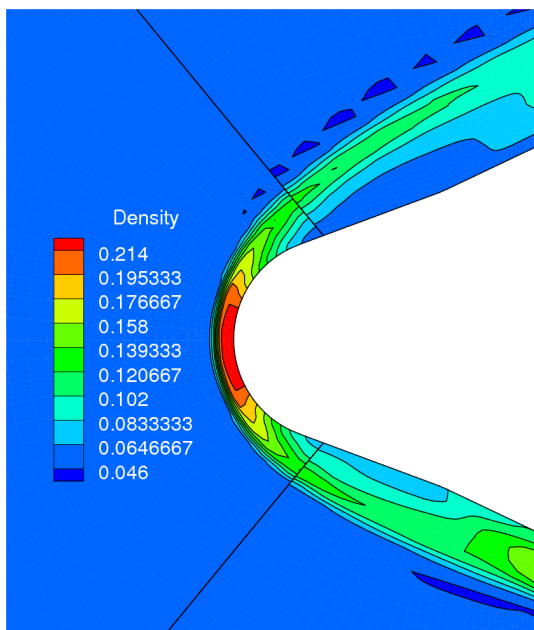
along the surface of capsule till the second shock. The AUSM+ scheme is more accurate. MacCormack with the default artificial viscosity coefficients fails in accuracy especially at the stagnation point.



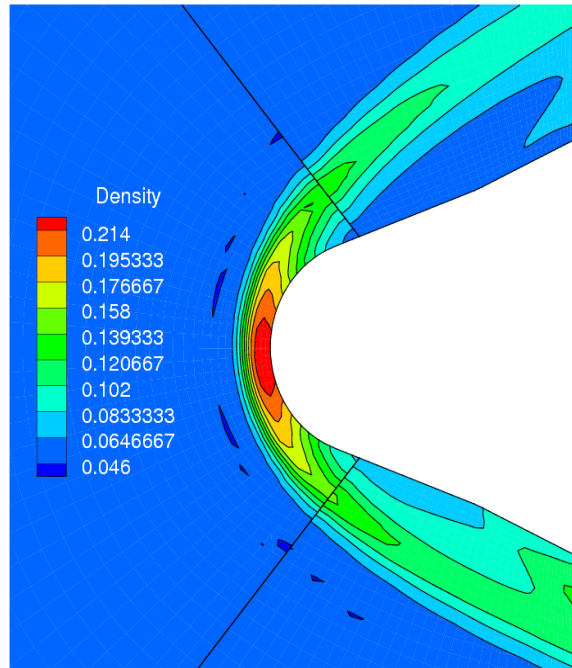
(a) Isodensity lines from Nagdewe et al.(2009) [37]



(b) Isodensity lines from fluent



(c) AUSM+ Scheme (AnuPravaha)



(d) MacCormack Scheme (AnuPravaha)

Figure 5.27: Density Contours

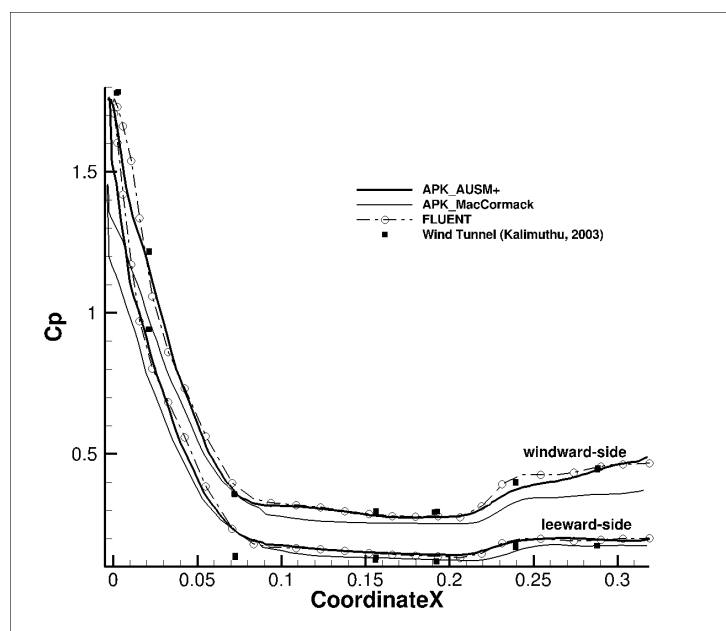


Figure 5.28: Variation of coefficient of pressure along the capsule wall

Chapter 6

Conclusions and future work

Building on the earlier work of Nikhil Kalkote[39], we have created a stand-alone version of the AnuPravaha Solver for computational flows. Several schemes were implemented on the solver, and thoroughly validated for subsonic, transonic and supersonic cases. Boundary conditions for complex and curved boundaries have also been implemented.

In the earlier part of this thesis we have used artificial viscosity in two ways, namely, the traditional artificial viscosity approach and the TVD artificial viscosity approach. The artificial viscosity approach is mainly used for structured meshes and its main advantage is its simplicity and high-speed execution on modern computers. The TVD approach, although theoretically more appealing, does not deliver the same accuracy as the traditional approach, and therefore was not used in the later computations.

In the later part of the thesis, we have presented the upwind-based scheme AUSM+ -up. This scheme proved to be most accurate and stable esp. in the transonic and supersonic regime of the flow. In subsonic regime, the MacCormack scheme with traditional artificial viscosity has done better.

We have incorporated the Far-Field boundary conditions with non-reflecting boundary condition at the outlet and have implemented boundary conditions for curved surfaces.

At the end, we are presenting a standalone density-based solver catering for aerospace applications exclusively.

We list down following points which can be used for future work.

1. AUSM+ -up scheme is more accurate and robust especially for transonic and supersonic regime. It does not have any empirical parameters such as artificial viscosity coefficients.
2. A review of convergence criteria has to be performed. At present, our measure for steady-state has been based on post-processing results.
3. For, turbo-machinery applications, Central schemes does better, thus TVD MacCormack scheme can be revised for future use. Implementing pressure-based subsonic

inlet condition using outgoing Riemann invariants can be considered.

4. Further validation of the present code can be made for more complex geometries and for subsonic and transonic flow regimes.
5. Navier-Stokes version of compressible flow with Turbulence models should be implemented to cater well to real aerospace applications.
6. An implicit version of the code can be developed. Also, a 2D version of the code for faster execution of two-dimensional cases is needed.

Our future goal is to develop an unstructured version of *AnuPravaha* for computation of high-speed flows. A successful implementation of upwind-based AUSM+ scheme has given us the confidence to start working in this direction. We hope to implement the Navier-Stokes equation with turbulence model with the present solver.

Appendices

Appendix A

Artificial Viscosity Formulation

The following explains the artificial viscosity formulation which has been frequently used in connection with the MacCormack technique. We show here the formulation for an unsteady, two-dimensional equation.

$$\frac{\partial U}{\partial t} = -\frac{\partial U}{\partial x} - \frac{G}{y} + J \quad (\text{A.1})$$

where U is the solution vector, $U = \left[\rho \quad \rho u \quad \rho v \quad \rho(e + V^2/2) \right]$.

At each step of the time-marching solution, a small amount of artificial viscosity can be added in the following form:

$$\begin{aligned} S_{i,j}^t = & C_x \frac{|p_{i+1,j}^t - 2p_{i,j}^t + p_{i-1,j}^t|}{p_{i+1,j}^t - 2p_{i,j}^t + p_{i-1,j}^t} (U_{i+1,j}^t - 2U_{i,j}^t + U_{i-1,j}^t) \\ & + C_y \frac{|p_{i,j+1}^t - 2p_{i,j}^t + p_{i,j-1}^t|}{p_{i,j+1}^t - 2p_{i,j}^t + p_{i,j-1}^t} (U_{i,j+1}^t - 2U_{i,j}^t + U_{i,j-1}^t) \end{aligned} \quad (\text{A.2})$$

where we have taken, $C_x = C_y = C_z = 0.12$

Eq. A.2 is a fourth order numerical dissipation expression. On the predictor step $S_{i,j}^t$ is evaluated based on the known quantities at time t . On the corrector step, the corresponding value of $S_{i,j}^t$ is obtained by using the predicted (barred) quantities as $\bar{S}_{i,j}^t$.

$$\begin{aligned} \bar{S}_{i,j}^t = & C_x \frac{|\bar{p}_{i+1,j}^t - 2\bar{p}_{i,j}^t + \bar{p}_{i-1,j}^t|}{\bar{p}_{i+1,j}^t - 2\bar{p}_{i,j}^t + \bar{p}_{i-1,j}^t} (\bar{U}_{i+1,j}^t - 2\bar{U}_{i,j}^t + \bar{U}_{i-1,j}^t) \\ & + C_y \frac{|\bar{p}_{i,j+1}^t - 2\bar{p}_{i,j}^t + \bar{p}_{i,j-1}^t|}{\bar{p}_{i,j+1}^t - 2\bar{p}_{i,j}^t + \bar{p}_{i,j-1}^t} (\bar{U}_{i,j+1}^t - 2\bar{U}_{i,j}^t + \bar{U}_{i,j-1}^t) \end{aligned} \quad (\text{A.3})$$

where we have taken, $C_x = C_y = C_z = 0.12$

The value of $S_{i,j}^t$ and $\bar{S}_{i,j}^t$ are added at various stages of MacCormack scheme as shown below with the help of calculation of density from the continuity equation. For this $U = \rho$.
 On the predictor step,

$$\bar{\rho}_{i,j}^{t+\Delta t} = \rho_{i,j}^t + \left(\frac{\partial \rho}{\partial t} \right)_{i,j}^t \Delta t + S_{i,j}^t \quad (\text{A.4})$$

On the corrector step,

$$\rho_{i,j}^{t+\Delta t} = \rho_{i,j}^t + \left(\frac{\partial \rho}{\partial t} \right)_a v \Delta t + \bar{S}_{i,j}^{t+\Delta t} \quad (\text{A.5})$$

References

- [1] C. Hirsch. Numerical computation of internal and external flows fundamentals of computational fluid dynamics. Elsevier, Amsterdam, 2007.
- [2] C. Hirsch. Numerical computation of internal and external flows vol 2. Wiley, Chichester [England]; New York, 1988.
- [3] N. K. Narayan. Development of Compressible Flow Module In AnuPravaha. Master Thesis, Indian Institute of Technology Hyderabad, India, 2013.
- [4] M. B. Giles. Nonreflecting boundary conditions for Euler equation calculations. *AIAA Journal* 28, (1990) 2050–2058.
- [5] H.-O. Kreiss. Initial boundary value problems for hyperbolic systems. *Communications on Pure and Applied Mathematics* 23, (1970) 277–298.
- [6] G. Hedstrom. Nonreflecting boundary conditions for nonlinear hyperbolic systems. *Journal of Computational Physics* 30, (1979) 222–237.
- [7] B. Engquist and A. Majda. Absorbing boundary conditions for numerical simulation of waves. *Proceedings of the National Academy of Sciences* 74, (1977) 1765–1766. PMID: 16592392.
- [8] B. Gustafsson. Far-Field Boundary Conditions for Time-Dependent Hyperbolic Systems. *SIAM Journal on Scientific and Statistical Computing* 9, (1988) 812–828.
- [9] D. H. Rudy and J. C. Strikwerda. A nonreflecting outflow boundary condition for subsonic navier-stokes calculations. *Journal of Computational Physics* 36, (1980) 55–70. Cited by 0234.
- [10] J. Blazek. Computational fluid dynamics principles and applications. Elsevier, Amsterdam; San Diego, 2005.
- [11] D. L. Whitfield and J. M. Janus. Three-dimensional unsteady Euler equations solution using flux vector splitting 1984. 00121.

- [12] A. Harten, B. Engquist, S. Osher, and S. R. Chakravarthy. Uniformly high order accurate essentially non-oscillatory schemes, {III}. *Journal of Computational Physics* 71, (1987) 231 – 303.
- [13] C. B. Laney. Computational gasdynamics. Cambridge Univ. Press, Cambridge u.a., 1998.
- [14] B. van Leer. Towards the ultimate conservative difference scheme. II. Monotonicity and conservation combined in a second-order scheme. *Journal of Computational Physics* 14, (1974) 361 – 370.
- [15] B. van Leer. Towards the ultimate conservative difference scheme. V. A second-order sequel to Godunov’s method. *Journal of Computational Physics* 32, (1979) 101 – 136.
- [16] B. van Leer. Towards the Ultimate Conservative Difference Scheme. *Journal of Computational Physics* 135, (1997) 229 – 248.
- [17] T. J. Chung. Computational Fluid Dynamics. Cambridge University Press, 2010.
- [18] A. Kurganov and E. Tadmor. New High-Resolution Central Schemes for Nonlinear Conservation Laws and Convection-Diffusion Equations. *J. Comput. Phys* 160, (2000) 241–282.
- [19] M.-S. Liou. A sequel to AUSM, Part II: AUSM-up for all speeds. *Journal of Computational Physics* 214, (2006) 137–170.
- [20] M.-S. Liou and C. J. Steffen Jr. A new flux splitting scheme. *Journal of computational physics* 107, (1993) 23–39.
- [21] M.-S. Liou. Ten Years in the Making: AUSM-family. National Aeronautics and Space Administration, Glenn Research Center, 2001.
- [22] J. D. Anderson. Computational fluid dynamics: the basics with applications. McGraw-Hill, New York, 1995.
- [23] B. Engquist and A. Majda. Radiation boundary conditions for acoustic and elastic wave calculations. *Communications on Pure and Applied Mathematics* 32, (1979) 313–357.
- [24] A. Harten. High resolution schemes for hyperbolic conservation laws. *Journal of Computational Physics* 49, (1983) 357 – 393.
- [25] J. Frst. Numerical Solution of Compressible Flows Using TVD and ENO Finite Volume Methods .
- [26] R. Leveque and H. Yee. A study of numerical methods for hyperbolic conservation laws with stiff source terms. *Journal of Computational Physics* 86, (1990) 187 – 210.

- [27] F. Coquel and P. LeFloch. Convergence of finite difference schemes for conservation laws in several space dimensions: the corrected antidiffusive flux approach. *Mathematics of computation* 57, (1991) 169–210.
- [28] J. Fürst, M. Janda, and K. Kozel. Finite volume solution of 2D and 3D Euler and Navier-Stokes equations. In *Mathematical fluid mechanics*, 173–193. Birkhäuser Basel, 2001.
- [29] S. Davis. A Simplified TVD Finite Difference Scheme via Artificial Viscosity. *SIAM Journal on Scientific and Statistical Computing* 8, (1987) 1–18.
- [30] D. Causon. High resolution finite volume schemes and computational aerodynamics. In *Nonlinear Hyperbolic Equations Theory, Computation Methods, and Applications*, 63–74. Vieweg+ Teubner Verlag, 1989.
- [31] J. D. Anderson. *Modern compressible flow: with historical perspective*. McGraw-Hill, Boston, Mass., 2003.
- [32] J. C. Wang and G. F. Widhopf. A High-resolution TVD Finite Volume Scheme for the Euler Equations in Conservation Form. *J. Comput. Phys.* 84, (1989) 145–173.
- [33] P. Furmnek. Numerical Solution Of Steady And Unsteady Compressible Flow. Ph.D. thesis, Czech Technical University in Prague 2008. 00000.
- [34] O. Arias, O. Falcinelli, N. Fico Jr, and S. Elaskar. FINITE VOLUME SIMULATION OF A FLOW OVER A NACA 0012 USING JAMESON, MacCORMACK, SHU AND TVD ESQUEMES. *Mecánica Computacional* 26, (2007) 3097–3116.
- [35] J. Rincon and R. Elder. A high-resolution pressure-based method for compressible flows. *Computers & fluids* 26, (1997) 217–231.
- [36] A. W. Date. SOLUTION OF NAVIER-STOKES EQUATIONS ON NONSTAGGERED GRID AT ALL SPEEDS. *Numerical Heat Transfer, Part B: Fundamentals* 33, (1998) 451–467.
- [37] S. Nagdewe, G. Shevare, and H.-D. Kim. Study on the numerical schemes for hypersonic flow simulation. *Shock Waves* 19, (2009) 433–442.
- [38] R. Kalimuthu. Surface Pressure Measurement Results on the SRE (biconic) Configuration at Mach = 5.
- [39] K. N. Narayan. Development of Compressible Flow Module In Anupravaha Solver. Ph.D. thesis, Indian Institute of Technology, Hyderabad 2013.