Development of a General-Purpose Compressible Flow AnuPravaha Based Solver using an Implicit Method

A Thesis Submitted
in Partial Fulfilment of the Requirements
for the Degree of
Master of Technology

by
Sutrisha Majumder
Roll No.- ME13M1034
M.Tech II Yr.-Thermofluids

भारतीय प्रौद्योगिकी संस्थान हैदराबाद
Indian Institute of Technology Hyderabad

Department Of Mechanical Engineering
Indian Institute Of Technology Hyderabad

JULY 2015

Declaration

I declare that this written submission represents my ideas in my own words, and where ideas and 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 misinterpreted 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.

Sutrisha Majumder

(Signature)

SUTRISHA MAJUMDER

(Student Name)

ME13M1034

(Roll No.)

Approval Sheet

This thesis entitled “Development of a General-Purpose Compressible Flow AnuPravaha Based Solver using Implicit Method ” by Sutrisha Majumder is approved for the degree of Master of Technology from IIT Hyderabad.

Narasimha Mangadoddy (Examiner)
Dept. of Chemical Engg.
IITH

Venkatasubbaiah K (Examiner)
Dept. of Mechanical Engg.
IITH

Prof. Vinayak Eswaran (Adviser)
Dept. of Mechanical Engg.
IITH

To my family ...

ACKNOWLEDGMENT

I have taken efforts in this project. However, it would not have been possible without the kind support and help of many individuals and organizations. I would like to extend my sincere thanks to all of them.

I would like to thank my thesis adviser Prof. Vinayak Eswaran for giving me an opportunity to work with him and for his exemplary guidance, monitoring and constant encouragement throughout the course of this thesis. His interest and confidence in me has helped immensely for the successful completion of this work.

I would also like to thank Nikhil Kalkote and Ashwani Assam for guiding me through the basics of the Anupravaha solver and constantly helping and clearing my doubts. I would also like to thank Praveen Throvagunta for his friendly support and encouragement. A special note of thanks to all my classmates from Thermal Engineering for inspiring me to do my best always. I would like to make a special mention of the excellent computational facilities provided by Professor Eswaran. I would also like to thank Mr. Madhu Pandicheri and Srikanth Vootla for his timely support in the CAE and HT lab.

Last but not the least, I would like to pay high regards to my parents for their encouragement, inspiration and lifting me uphill this phase of life.

Sutrisha Majumder

Abstract

The design of aerospace vehicles requires a detailed knowledge of the interactions between flight surfaces and the ambient atmosphere through which they pass. The resulting aerodynamic pressure as well as heating loads are among the critical parameters inherentto a successful design. Insight into these conditions is obtained through a mix of analytical, computational and ground based experimental techniques. The recent paradigm shift embraced by the motto faster, better, cheaper calls for continual improvement in these areas through a greater understanding of the physical processes involved.

In this thesis We aim to develop a general-purpose and robust compressible flow solver with implicit MacCormack scheme in finite volume formulation. A system of unsteady Euler equations is integrated to a steady state solution utilizing MacCormack's implicit numerical scheme. A new implicit boundary treatment was introduced in the MacCormack implicit scheme. The scheme is unconditionally stable and does not require solution of large systems of linear equations. Moreover, the upgrade from explicit MacCormack scheme to implicit one is very simple and straightforward.

Several computational results for 2D and 3D flows over profiles and wings are presented for the case of inviscid flows.

Contents

List of Figures

List of Tables

Nomenclature

ρ	Density of gas
α	Angle of attack
γ	Ratio of specific heat
δ	Implicit operator
$\triangle$	Explicit operator
$\triangle_+, \triangle_-$	forward and backward finite differencing operator respectively
β	γ - 1
A	inviscid Jacobian, $\frac{\partial F_x}{\partial W}$
B	inviscid Jacobian, $\frac{\partial F_y}{\partial W}$
C	inviscid Jacobian, $\frac{\partial F_z}{\partial W}$
c	Speed of sound
C_v	Specific heat of gas at constant volume
C_p	Specific heat of gas at constant pressure
D_A, D_B, D_C	Characteristic Diagonal matrix
e	Total internal energy
F_x	x direction flux vector
F_y	y direction flux vector
F_z	z direction flux vector
I	Identity Matrix
M	Mach no.
P	Pressure(N/m^2)
R	Universal gas constant

Sx	Right eigen vector for x direction
Sx^{-1}	left eigen vector for x direction
Sy	Right eigen vector for y direction
Sy^{-1}	left eigen vector for y direction
Sz	Right eigen vector for z direction
Sz^{-1}	left eigen vector for z direction
T	Absolute Temperature
u	x component of velocity
v	y component of velocity
V_p	Volume of cell p
w	z component of velocity
W	Conservative vector
x	Cartesian coordinate
y	Cartesian coordinate
z	Cartesian coordinate

Chapter 1

Introduction

The basic mathematical model of fluid flow takes the form of partial differential equations which express the laws of conservation of mass, momentum and energy. While analytical solutions to these equations are possible for a few simple cases, in most cases, specially for complex geometry, the only alternative is to obtain approximate numerical solutions. Computational Fluid Dynamics (CFD, in short) is a powerful bridge between the calculus describing flow physics and highspeed computing. CFD methodology has matured over the years to an extent that it has found its way into most fluid flow research applications, notably in the aerospace industry.

Computational Fluid Dynamics (CFD) methods must satisfy stringent constraints because of the wide range of scales and frequencies in the target flows. To deal with those requirements, higher order, low dispersion and low dissipation schemes are needed. However, these schemes are also more sensitive to spurious waves generated by numerical boundary conditions.

In aerodynamics, the compressibility of a fluid is a very important factor. In nature, all the fluids are detectably compressible, but we define incompressible fluids for our convenience of study. To understand what compressible fluids is one must first understand what compressibility is. The compressibility of a fluid is the reduction of the volume of the fluid due to external pressures acting on it. A compressible fluid will reduce its volume in the presence of an external pressure and/or temperature. 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 effects are observed in practical applications like high speed aerodynamics, missile and rocket propulsion, high speed turbo compressors, steam and gas turbines, etc.

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 discontinuities such as shock waves. So for compressible flows special attention is required for solution methods which will accurately capture these discontinuities.

A major difference between solution methods for compressible flow and incompressible flow lies in the boundary conditions that are imposed. In compressible flow, 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.

For over a dicade our research group has been continuously developing and modifying a CFD software called IITK-DAE ANUPRAVAHA, a genaral purpose CFD solver. An compressible finite volume method with structured grid arrangement has been used in the present work to solve unsteady Euler Equations. However, using an unsteady solver to obtain steady-state solutions by the false transient method is inefficient, especially if explicit time stepping with time-step constraints due to numerical stability, is used. To achieve fast convergence to the steady state, an implicit time marching scheme is much better to solve the Euler equations. This thesis implements a scheme is based on the MacCormack implicit scheme [22].

1.1 IITK-DAE ANUPRAVAHA Compressible Solver

The ANUPRAVAHA Compressible solver is separated from the original ANUPRAVAHA incompressible solver to cater to aerospace applications exclusively. In this solver, flow equations have been previously solved using explicit schemes: the MacCormack scheme and AUSM+ scheme.The explicit MacCormack scheme with artificial viscosity terms proved to have very good accuracy and efficiency. It has been applied successfully for calculations of subsonic, transonic and supersonic flows over profiles and wings.

The main drawback of the explicit scheme is its time-step limitation due to the numerical stability condition. It becomes inefficient for unsteady flows where the global time-scale (e.g. period of oscillation of a wing) can be much larger than the time-step, and for the high-Reynolds viscous flows, where the mesh refinement in boundary layers results in extremely small time-steps. A computation with an explicit scheme requires substantial computer time.

Some Implicit schemes have the advantage of being unconditionally stable without CFL restrictions. Since the convergence to steady-state depends on the propagation speed of the error waves, large CFL numbers accelerate the convergence to steady state. The implicit MacCormack scheme therefore implements in this thesis to facilitate future studied of unsteady and steady compressible flows.

1.2 Literature review

Hirsch (2007) 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. The second volume of Hirsch (2007) discusses almost all basic numerical schemes. such as central, upwinding and high-resolution schemes pertaining to Euler and Navier-Stokes equations. Euler equations are solved in conservative form but require imposition of boundary condition in primitive form. In Chapter 19 Hirsch discusses the implementation of boundary conditions (both physical and numerical) from characteristic extrapolation for conservative and primitive variables, along with different extrapolation methods.

Implicit and semi-implicit schemes require a very powerful linear solver since the jacobians usually lack diagonal dominance at least at high CFL numbers. This has an adverse effect on the convergence of many iterative solvers. Implicit solvers are still rarely used for the computation of stationary solutions to the Euler equations. However, their development has been pursued by several groups [[14], [15], [12],[22]]. Many existing schemes employ linearizable/differentiable limiters, and are conditionally stable, and the rate of steady-state convergence deteriorates if the CFL number exceeds a certain upper bound. The scheme presented here converges for arbitrary CFL numbers despite oscillatory correction factors and the rate of steady-state convergence does not deteriorate for large CFL numbers. In this work the implicit approach circumvents the computationally expensive nonlinear iterations.

The development of robust and accurate boundary conditions is of primary importance, and sufficient care must be taken in the numerical implementation. The accuracy, robustness, stability, and convergence of an implicit solver are strongly influenced by the boundary treatment. Strongly imposed boundary conditions may inhibit convergence to a steady state. Thus, it is worthwhile to use flux boundary conditions of Neumann type. The weak type of boundary conditions turns out to be much more stable and flexible than its strong counterpart.When boundary conditions are prescribed in a weak sense, only the boundary integral of the weak formulation is affected by the boundary conditions, while the volume integrals remain unchanged. This is similar to the boundary treatment, which is usually implemented in finite volume schemes. In the finite volume framework the boundary fluxes are directly overwritten by the imposed boundary conditions.

The Neumann type of boundary conditions, based on the weak formulation, can be treated implicitly and incorporated into the matrix in a physical way. It improves the convergence rates and does not affect the matrix properties or give rise to stability restrictions in contrast to the strong type of boundary conditions. According to [29], [27] a stability restriction of CFL number of 0.6 applies for an explicit implementation of weak wall boundary conditions, while the stability is significantly improved for a semi-implicit version up to a CFL number of 100. This emphasizes the importance of an implicit treatment of boundary conditions for the numerical performance, which is presented in this study. We recommend a boundary Riemann solver to compute the boundary fluxes in the boundary integrals to avoid unphysical effects particularly at large CFL numbers. To define a boundary Riemann problem the concept of ghost nodes is introduced. We show that a suitable treatment of boundary conditions makes it possible to achieve unconditional stability.

In the following chapters, the design procedure of an unconditionally stable finite volume schemes for the Euler equations are addressed. In the Euler equations, the treatment of boundary conditions based on a boundary Riemann solver is described, and the implicit solver is presented. Furthermor, the design procedure of implicit solver for Eulers equations is described. Finally, the numerical performance and accuracy of the proposed scheme are analyzed.

1.3 Objective of present work

The objectives of this thesis are threefold:

- To investigate ways of modeling finite volume implicit MacCormack methodology in a three dimensional simulation code to solve the system of Euler equations.

- To develop a new CFD code that implements solution of Euler equations numerically with significant reduction in computation time.

- To validate this solver for three different regimes of flow i.e. in subsonic, transonic and in supersonic regime.

Chapter 2

Governing Equations

2.1 The Flow and its Mathematical Description

Fluid dynamics is defined as the investigation of the interactive motion of a large number of individual particles (molecules or atoms). So, we can assume the density of the fluid is high enough and it can be approximated as a continuum. It means, even an infinitesimally small (in the sense of differential calculus) element of the fluid contains a sufficient number of particles, in terms of molecule or atoms, for which we can specify mean velocity and mean kinetic energy. In this way, we are able to define velocity, pressure, temperature, density and other important quantities at each point of the fluid.

The derivation of the principal equations of fluid dynamics depends upon the dynamical behaviour of a fluid, is determined by the following conservation laws:

1. The conservation of mass.
2. The conservation of momentum.
3. The conservation of energy.

The conservation of a certain flow quantity is based on the total variation of flow quantity inside an arbitrary volume and the net effect of the amount of the quantity being transported across the boundary due to any internal forces and sources and/or the external forces acting on the volume. The amount of the quantity crossing the boundary is called *Flux.* The flux can be divided into two different parts: one due to the convective transport and the other one due to the molecular motion present in the fluid at rest.

Consider a general flow field as represented by streamlines in Fig. 2.1. An arbitrary finite region of the flow, bounded by the closed surface ∂v and fixed in space, defines the control volume v. We also consider a surface element dS and its associated, outward pointing unit normal vector $\vec{n}$ of the control surface which enclose the control volume v.

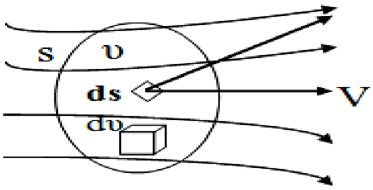


Figure 2.1: Definition of a finite control volume (fixed in space)

Let the conservation law applied to an scalar quantity per unit volume ϕ. Its variation in time within ∂v can be written as,

$$\frac{\partial}{\partial t}\int_v \phi \, dv$$

This is equal to the sum of the contributions due to the convective flux which is the amount of the quantity ϕ entering the control volume through the boundary with the velocity $\vec{u}$.

$$-\oint_{\partial v} \phi(\vec{u}.\vec{n})dS$$

The integral formulation of the conservation law is given by

$$\frac{\partial}{\partial t}\int_v \phi \, dv \; + \; \oint_{\partial v} \phi(\vec{u}.\vec{n})dS \; = \; 0 \tag{2.1}$$

2.1.1 Continuity Equation

If we consider only single-phase fluids, the law of mass conservation expresses as: mass cannot be created in such a fluid system, nor it can disappear. For the continuity equation, the conserved quantity ϕ is the density ρ. According to the general formulation of Eqn. 2.1, we can write the continuity equation as:

$$\frac{\partial}{\partial t}\int_v \rho \, dv \; + \; \oint_{\partial v} \rho(\vec{u}.\vec{n}) \, dS \; = \; 0$$

2.1.2 Momentum Equation

The derivation of the momentum equation is based on the particular form of Newton's second law which states that the variation of momentum is caused by the net force acting on an mass element. The momentum of an infinitesimally small portion of the control volume υ given by $\rho\vec{u}\,d\upsilon$. The variation in time of momentum within the control volume equals

$$\frac{\partial}{\partial t}\int_{\upsilon}\rho\vec{u}\,d\upsilon$$

Here $\rho\vec{u} = [\rho u \;\; \rho v \;\; \rho w]^T$, where u, v, w are the x component, y component and z component velocities respectively.

In the conservation of momentum, the contribution of the convective tensor is given by

$$-\oint_{\partial\upsilon}\rho\vec{u}(\vec{u}.\vec{n})\,dS$$

Two types of forces act on the control volume: external volume or body forces and surface forces. Surface forces result from only two sources:

a) The pressure distribution, imposed by the outside fluid surrounding the volume.

b) The shear and normal stresses, resulting from the friction between the fluid and the surface of the volume.

Now sum up all the above contributions according to the general conservation law (Eqn. 2.1), and finally obtain the expression for momentum conservation equation

$$\frac{\partial}{\partial t}\int_{\upsilon}\rho\vec{u}\,d\upsilon + \oint_{\partial\upsilon}\rho\vec{u}(\vec{u}.\vec{n})\,dS \;=\; \int_{\upsilon}\rho\vec{f_e} \;-\; \oint_{\partial\upsilon}p\vec{n}\,dS \;+\; \oint_{\partial\upsilon}(\vec{\tau}.\vec{n})\,dS$$

where $\vec{f_e}$ body force per unit mass, p is the static pressure, τ is the stress tensor.

2.1.3 Energy Equation

The energy equation is based on the first law of thermodynamics. It states that the rate of change in the total energy inside the volume is equal to the rate of work of forces acting on the volume and by the net heat flux into it. The total energy per unit mass is defined E and we can write:

$$E \;=\; e + \frac{u^2 + v^2 + w^2}{2}$$

where e is internal energy per unit mass.

Similar to the momentum conservation equation we can write a conservative equation for the heat energy by accordingly for the rate of heat addition by conduction and volumetric heating and the work done by surface and body forces. The energy conservation equation according to the general conservation law (Eq. 2.1) is

$$\frac{\partial}{\partial t}\int_v \rho E \, dv + \oint_{\partial v} \rho E(\vec{u}.\vec{n}) \, dS \;=\; \oint_{\partial v} k(\nabla T.\vec{n}) \, dS \;+\; \int_v (\rho \vec{f_e}.\vec{u} + \dot{q_h}) - \oint_{\partial v} p(\vec{u}.\vec{n}) \, dS \;+\; \oint_{\partial v} (\vec{\tau}.\vec{u}).\vec{n} \, dS$$

where $\dot{q_h}$ is the rate of heat addition per unit volume and k is the thermal conduction of the fluid.

2.2 Euler Equations

The most general flow configuration for a non-viscous, non-heat conducting fluid is described by the set of Euler equations, obtained from the Navier Stokes equations by neglecting all shear stresses and heat conduction terms. If we collect the conservation laws of mass, momentum and energy into one system of equations neglecting the body forces and stress forces, we obtain the Euler Equations. The time-dependent Euler equations, in conservation form and in an absolute frame of reference, for the conservative variables U is:

$$\frac{\partial}{\partial t}\int_v U \, dv \;+\; \oint_{\partial v} \nabla.\vec{F} \, dv \;=\; 0 \tag{2.2}$$

which form a system of first order hyperbolic partial differential equations, where U is the solution vector

$$U = \begin{bmatrix} \rho \\ u \\ v \\ w \\ E \end{bmatrix}$$

and the flux vector F has the Cartesian components (f , g, h) given by equation 2.2

$$f = \begin{bmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ \rho uw \\ (e+p)u \end{bmatrix} g = \begin{bmatrix} \rho v \\ \rho uv \\ \rho v^2 + p \\ \rho vw \\ (e+p)v \end{bmatrix} h \ = \begin{bmatrix} \rho w \\ \rho uw \\ \rho vw \\ \rho w^2 + p \\ (e+p)w \end{bmatrix}$$

Assuming the Control Volume (CV) is fixed in space, the governing integral equation can be written as,

$$\oint_{\partial v} (\frac{\partial U}{\partial t} + \nabla.\vec{F})\, dv \ = \ 0 \tag{2.3}$$

and further, as the CV is arbitrary, we can write,

$$\frac{\partial U}{\partial t} + \nabla.\vec{F} \ = \ 0 \tag{2.4}$$

2.3 Discretization Techniques and Grid Generation

In mathematics, discretization concerns the process of transferring continuous functions, models and equations into discrete counterparts. This process is usually carried out as a first step toward making them suitable for numerical evaluation and implementation on digital computers The discretization techniques use grids in order to discretize the governing equations 2.2, 2.4. Basically, there are two different types of grids:

- Structured Grids: each grid point (vertex, node) is uniquely identified by the indices i, j, k and the corresponding Cartesian coordinates $x_{i,j,k}, y_{i,j,k}, andz_{i,j,k}$. The grid cells are quadrilaterals in 2D and hexahedral in 3D.

- Unstructured Grids: grid cells as well as grid points have no particular ordering, i.e., neighbouring cells or grid points cannot be directly identified by their indices. In the past, the grid cells were triangles in 2D and tetrahedral in 3D. Nowadays unstructured grids usually consist of a mix of quadrilaterals and triangles in 2D and of hexahedral, tetrahedral, prisms and pyramids in 3D.

Here we use structured grids to solve the governing equations. The main advantage of structured grids is that the indices i, j, k represent a linear address space, since it directly corresponds to how the flow variables are stored in the computer memory. This property allows it to access the neighbours of a grid point very quickly and easily, just by adding or subtracting an integer value to or from the corresponding index (e.g. (i+1), (j-3), etc . see Fig. 2.2). The evaluation of gradients, fluxes, and also the treatment of boundary conditions is simplified by this feature. The same holds for the implementation of an implicit scheme, because of the well-ordered, banded flux Jacobian matrix.

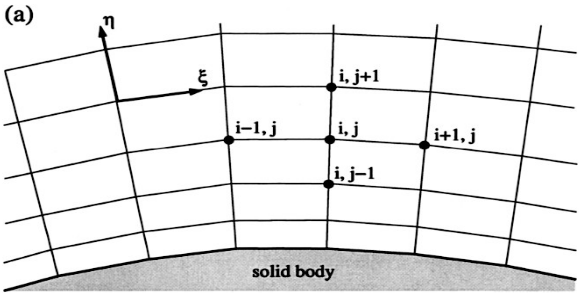


Figure 2.2: Structured, body-fitted grid approach (in two dimensions)

But there is also a disadvantage. The disadvantage is the generation of structured grids for complex geometries. To overcome this disadvantage we can divide the physical space into a number of topologically simpler parts or blocks (see 2.3), which can be more easily meshed. This is called the multiblock mesh. In this thesis we use multiblock approach to generate the mesh. The advantage of this approach is, the number of grid lines can be chosen separately for each block as required to be close to rectangular, or orthogonal, which increase numerical accuracy and convergence. The another advantage of the multiblock methodology is that it allows for the possibility of using parallel computation by means of domain decomposition.

The discretization schemes can be divided into the following three main categories:

- Finite Difference Method: which can be applied to rectangular/structured mesh configurations.

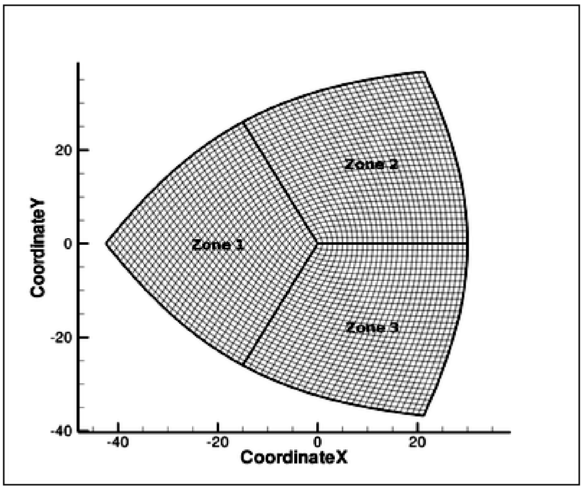


Figure 2.3: Structured, multiblock grid

- Finite Volume Method: which can be applied to both structured and unstructured mesh configuration.
- Finite Element Method: which is the common method in solid mechanics, but is also applicable to fluid mechanics, which is applied to unstructured grid.

2.3.1 Finite Difference Method

The finite difference method was the first approaches applied to the numerical solution of differential equations. It was first utilized by Leonhard Euler in 1768 [17]. This method is directly applied to the differential form of the governing equations 2.4.

For a function U(x), the Taylor series expansion of $U_{x_0+\Delta x}$ in x can be written as

$$U_{(x_0+\Delta x)} = U_{(x_0)} + \Delta x \left(\frac{\partial U}{\partial x}\right)_{x_0} + \frac{\Delta x^2}{2}\left(\frac{\partial^2 U}{\partial x^2}\right)_{x_0} + ...$$

From the above equation, the first derivative of U can be approximated as

$$\left(\frac{\partial U}{\partial x}\right)_{x_0} = \frac{U_{x_0+\Delta x} - U_{x_0}}{\Delta x} + \bigcirc(\Delta x) \qquad (2.5)$$

The above approximation is of first order, since the truncation error (abbreviated as $\bigcirc(\Delta x)$), which is proportional to the largest term of the remainder, goes to zero with the first power of Δx.

To apply this general definition 2.5, we consider an one-dimensional space, the x-axis, and the space discretization is done with N discrete mesh points x_i, i = 0,...,N (Figure 2.4). Let U_i is the value of the function U_{x_0} at the point x_i , i.e. $U_i = U_{x_i}$ and the spacing between the discrete points is constant and equal to Δx. Applying the above relation 2.5 at point i, we obtain the following finite difference approximation

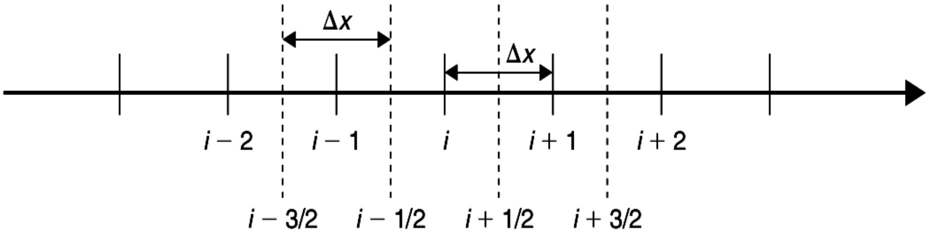


Figure 2.4: One-dimensional uniform FDM grid on the x-axis[17]

$$(U_x)_i = \left(\frac{\partial U}{\partial x}\right)_i = \frac{U_{i+1} - U_i}{\Delta x} \underbrace{- \frac{\Delta x}{2}\left(\frac{\partial^2 U}{\partial x^2}\right)_i - \frac{\Delta x^2}{6}\left(\frac{\partial^3 U}{\partial x^3}\right)_i + ...}_{\text{Truncation error}} \tag{2.6}$$

$$= \frac{U_{i+1} - U_i}{\Delta x} + \bigcirc(\Delta x) \tag{2.7}$$

As this formula involves the point (i+1) to the right of point i, it is called the first order forward difference for the first derivative U_{x_i}.

Now if Δx is replaced by $-\Delta x$, then the finite difference approximation is

$$(U_x)_i = \left(\frac{\partial U}{\partial x}\right)_i = \frac{U_i - U_{i-1}}{\Delta x} \underbrace{+ \frac{\Delta x}{2}\left(\frac{\partial^2 U}{\partial x^2}\right)_i - \frac{\Delta x^2}{6}\left(\frac{\partial^3 U}{\partial x^3}\right)_i + ...}_{\text{Truncation error}} \tag{2.8}$$

$$= \frac{U_i - U_{i-1}}{\Delta x} + \bigcirc(\Delta x) \tag{2.9}$$

This formula is called the first order backward difference for the derivative U_{x_i} as it involves the point (i-1) to the left of point i. If we add this two equations (Equations 2.6 and 2.8), we obtain a second order approximation

$$(U_x)_i = \frac{U_{i+1} - U_{i-1}}{2\Delta x} - \frac{\Delta x^2}{6}\left(\frac{\partial^3 U}{\partial x^3}\right)_i + ... \tag{2.10}$$

$$= \frac{U_{i+1} - U_{i-1}}{2\Delta x} + \bigcirc(\Delta x^2) \tag{2.11}$$

Equation 2.10 involves the points to the left and to the right of point i, is therefore called a central difference formula.

The important advantages of the finite difference methodology are its simplicity and the possibility to obtain high-order approximations easily and to achieve high-order accuracy of the spatial discretization. The main disadvantage of this method is, it requires a structured rectangular grid, so the range of application is restricted. Furthermore, the finite difference method cannot be directly applied in body-fitted i.e curvilinear coordinates. So first we have to transform the governing equations into a rectangular grid system or in other words from the physical to the computational space. Thus, the finite difference method can be applied only to rather simple geometries.

2.3.2 Finite Volume Formulation

The finite volume method directly makes use of the conservation laws, the integral formulation of the Euler equations(Equations 2.2). It was first employed by McDonald for the simulation of 2-D inviscid flows[17]. The finite volume method discretizes the governing equations by first dividing the physical space into a number of arbitrary polyhedral control volumes. The surface integral is Equation 2.2 is then approximated by the sum of the fluxes crossing the individual faces of the control volume. The accuracy of this spatial discretization depends on the particular scheme with which the fluxes are evaluated.

Additionally, complicated boundary conditions for complex flow domains can be implemented in a relatively straight-forward manner. Figure 2.5 shows an

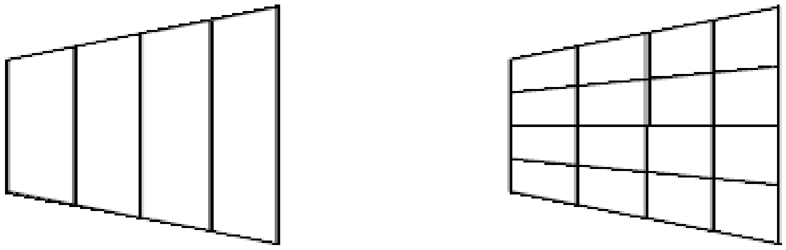


Figure 2.5: 1d (left) and 2d (right) Finite Volume discretization of an expanding domain

example of a 1d and 2d finite volume discretization for an expanding flow domain. Algebraic equations can be obtained for each control volume by approximating the volume and surface integrals using quadrature formulae. Volume integrals can be evaluated with second order accuracy by the product of the mean value of ϕ, assumed to be at the cell centroid, and the cell volume whilst surface integrals are calculated by summation over the sides of the cell. The integral on each face being approximated by the midpoint rule. The semi-discrete form of the governing equations are written for each cell as

$$\frac{\partial U_{cell-centered}}{\partial t} = -\frac{1}{V}\sum_{if} F_{if} A_{if}$$

with A and V being the cell edge interface area and cell volume respectively. The discretized equations applied to each control volume can be advanced in time from an initial solution once a technique for determining the interface fluxes is specified.

There are two basic approaches of defining the shape and position of the control volume with respect to the grid:

- Cell centered scheme (Fig: 2.6(a)): Here the flow quantities are stored at the centroids of the grid cells. So, the control volumes are identical to the grid cells. We use the cell-centered scheme in this thesis.

- Cell vertex scheme (Fig: 2.6(b)): Here the flow variables are stored at the grid points. The control volume can then either be the union of all cells sharing the grid point, or some volume centered around the grid point.

The main advantage of the finite volume method is that the spatial discretization is carried out directly in the physical space. Thus, there are no problems with any transformation between the physical and the computational coordinate system, as in the case of the finite difference method. Another advantage of the finite volume method, compared to the finite difference method is that it is very flexible, and can be rather easily implemented on structured as well as on unstructured grids. This makes the finite volume method particularly suitable for the treatment of flows in complex geometries.

The finite volume method is based on the direct discretization of the conservation laws, mass, momentum and energy, which are also conserved by the

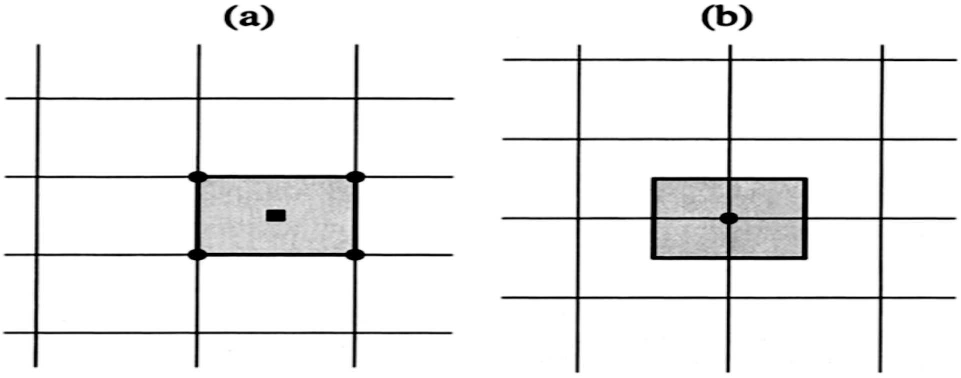


Figure 2.6: Control volume of cell centered (a) and cell vertex (b) scheme

numerical scheme. So it has the ability to compute weak solutions of the governing equations correctly. This is the another important feature of the method, However, one additional condition is needed to be fulfilled in the case of the Euler equations, and this is known as the entropy condition. It is necessary because of the non-uniqueness of the weak solutions. The entropy condition prevents the occurrence of unphysical features like expansion shocks, which violate the second law of thermodynamics (decrease of the entropy).

Under certain conditions, the finite volume method can be shown to be equivalent to the finite difference method, or to a low-order finite element method.

2.4 Time Integration

For a given current flow state, the discretized equations can be advanced in time by selecting an appropriate numerical integration technique. Schemes are classified as being either explicit, implicit or a mixture of the two. Explicit integration uses knowledge of only the current flow state and as such is not very computationally intensive. The equations are advanced in small time steps governed by strict stability criteria. For example, a wave starting at a cell interface should not cross more than half of the cell width during a time step. Implicit integration uses knowledge of both the (known) current flow state and the (unknown) next time step state. So each time step is computationally more expensive than an explicit method because the equations for all cells have to be solved simultaneously. But implicit methods have advantages in stability.

Mathematically, if $Y(t)$ is the current system state and $Y(t+\Delta t)$ is the state at the later time (Δt is a small time step), then, for an explicit method for the PDE

$$\frac{\partial Y}{\partial t} = F(y)$$

is

$$Y(t+\Delta t) \ = Y(t) \ + \ F(Y(t))$$

while for an implicit method one solves an equation

$$Y(t+\Delta t) \ = Y(t) \ + \ F(Y(t+\Delta t))$$

The main drawback of explicit scheme is that the stability requirements can result in very short time steps and correspondingly long computation times. Implicit methods are used because many problems arising in practice are stiff, for which the use of an explicit method requires impractically small time steps Δt to keep the error in the result bounded. For such problems, to achieve given accuracy, it takes much less computational time if we use an implicit method with larger time steps.

For the flows considered in this thesis, we use implicit technique for time integration to reduce the computational time.

2.5 CGNS File Format

CGNS (CFD General Notation System) originated in 1994 as a joint effort between Boeing and NASA, and has since grown to include many other contributing organizations worldwide. It is an effort to standardize CFD input and output, include grid (both structured and unstructured), flow solution, connectivity, boundary conditions (BCs), and auxiliary information. CGNS is also extensible, and allows for file-stamping and user-inserted-commenting. It employs ADF (Advanced Data Format), a system which reads a binary files that are portable across computer platforms. CGNS also includes a second layer of software known as the midlevel library, or API (Application Programming Interface), which eases the implementation of CGNS into existing codes.

CGNS has the capability to replace most of the translator programs now necessary when working between machines and between CFD codes. Also, it eventually may allow for the results from one code to be easily restarted using another code. It will hopefully therefore save a lot of time and money. In particular, it is hoped that future grid- generation softwares will generate grids with all connectivity and BC information included as part of a CGNS database, saving time and avoiding potential costly errors in setting up this information after the fact.

A CGNS file is an entity that is organized (inside the file itself) into a set of nodes in a tree-like structure, in much the same way as directories are organized in the UNIX environment. The top-most node is referred to as the root node. Each node below the root node is defined by both a name and a label, and may or may not contain information data. Each node can also be a parent to one or more child nodes.

CGNS files are binary files and they cannot be viewed by user with standard UNIX ASCII- editing tools. The utility ADF was created to allow users to easily view CGNS files. For more detailed information, readers are suggested to see the CGNS website www.cgns.org (especially the Users Guide Section).

In the present work, CGNS version 2.4 file-format is used to store grids and flow solution. Post processing is done by Tecplot 360, which is also a CGNS 2.4 compatible software.

2.6 Closure

In this chapter we discussed the basic nature of the governing equations for the flow problems and different discretization techniques of solving these governing equations. We also discussed the way of time integration and CGNS file format .

Chapter 3

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:

$$
\begin{aligned}
u\frac{\partial \rho}{\partial x} + v\frac{\partial \rho}{\partial y} + \rho\frac{\partial u}{\partial x} + \rho\frac{\partial v}{\partial y} &= 0 \\
u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} &= -\frac{1}{\rho}\frac{\partial p}{\partial x} \\
u\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y} &= -\frac{1}{\rho}\frac{\partial p}{\partial y}
\end{aligned}
\tag{3.1}
$$

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

$$
\left(1 - \frac{u^2}{c^2}\right)\frac{\partial^2 \phi}{\partial x^2} - \frac{2uv}{c^2}\frac{\partial^2 \phi}{\partial x \partial y} + \left(1 + \frac{u^2}{c^2}\right)\frac{\partial^2 \phi}{\partial y^2} = 0 \tag{3.2}
$$

where ϕ is the potential function defined by,

$$
u = \frac{\partial \phi}{\partial x} \qquad ; \qquad v = \frac{\partial \phi}{\partial y}
$$

These two sets of PDEs 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 3.1 seeming to be convection dominated, whereas the second equation 3.2 seemingly diffusion dominated. Convection and diffusion are two im- portant phenomena in Fluid Mechanics, and relate to very different physical behavior. But, as we know, both 3.1 and 3.2 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 their of physical form, to tell us something about the nature of their solutions.

3.1 Domain of dependance and zone of influence

3.1.1 Single Conservation Law

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

$$\frac{\partial w}{\partial t} + \frac{\partial F}{\partial x} = 0 \qquad w(x,0) \ = \ w_0(x) \tag{3.3}$$

$$\frac{\partial w}{\partial t} + \frac{\partial F}{\partial w}\frac{\partial w}{\partial x} \ = \ 0 \tag{3.4}$$

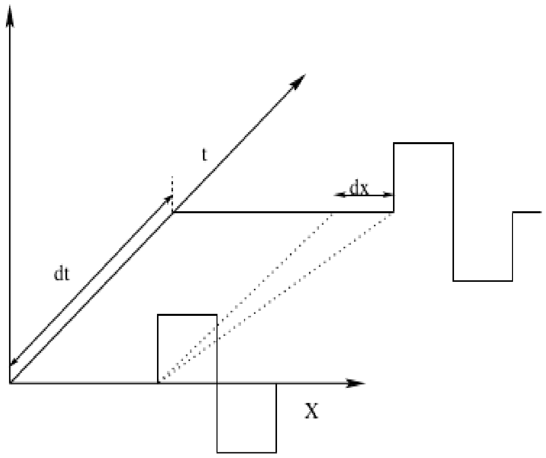
Introducing wave speed as, $\frac{\mathrm{d}F}{\mathrm{d}x} \ = \ \lambda$ in the above equation 3.4 can also be written as,

$$\frac{\partial w}{\partial t} + \lambda\frac{\partial w}{\partial x} \ = \ 0 \tag{3.5}$$

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

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

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 $\frac{\mathrm{d}x}{\mathrm{d}t} \ = \ \lambda$, 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 we can see that, for a single equation, the solution at a point 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 dependance and zone of influence of any point on that characteristic.

3.1.2 System of PDEs

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

$$\frac{\partial w_i}{\partial t} + \frac{\partial F_i}{\partial x} = 0 \tag{3.6}$$

where w_i is a vector containing conservative variables, F_i is a vector containing flux associted with w_i.

The equation 3.6 can be written as,

$$\frac{\partial w_i}{\partial t} + [A]\frac{\partial w_i}{\partial x} = 0 \tag{3.7}$$

where [A] is called the Jacobian (matrix). The Jacobian in equation 3.7 determines the behavior of the solutions, based on nature of its eigenvalues. Diagnolizing the Jacobian in equation 3.7, we can rewrite the equation as:

$$\frac{\partial w_i}{\partial t} + [Q_r]\,[\lambda_i]\,[Q_l]\frac{\partial w_i}{\partial x} = 0 \tag{3.8}$$

where Q_r, Q_l are the left and right eigenvector matrices that satisfy.

For any diagonalizable matrix A we can write,

$$\begin{aligned} [\lambda] &= [Q_l]\,[A]\,[Q_r] \\ [Q_l] &= [Q_r]^{-1} \end{aligned}$$

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 equation 3.8 by $[Q_l]$ and rewriting equation 3.8 we get

$$[Q_l]\,\frac{\partial w_i}{\partial t} + [Q_l]\,[Q_r]\,[\lambda_i]\,[Q_l]\,\frac{\partial w_i}{\partial x} = 0 \tag{3.9}$$

As we know,

$$[Q_l] = [Q_r]^{-1} \tag{3.10}$$

$$[Q_l]\,\frac{\partial w_i}{\partial t} + [\lambda_i]\,[Q_l]\,\frac{\partial w_i}{\partial x} = 0 \tag{3.11}$$

Introducing new set of variables as,

$$\delta v_i = [Q_l]\,\delta w_i \tag{3.12}$$

equation 3.11 can be written as,

$$\frac{\partial v_i}{\partial t} + [\lambda_i]\,\frac{\partial v_i}{\partial x} = 0$$

Now, note the simplicity of equation 3.1.2 in comparison to equation 3.6; unlike in the latter equation, the equations in system 3.1.2 are decoupled, i.e., the solution of any one of them is independent of the solution of others. Thus the component equations of 3.1.2 can thus be solved separately and easily. Then, by inverting equation 3.12 we can obtain the solution to system of equation 3.6. As shown in the figure 3.1 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.

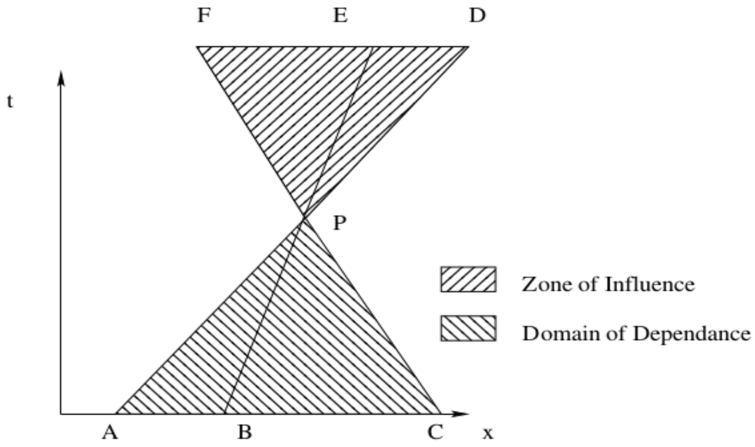


Figure 3.1: Characteristics in 3 equation system

Hence this region is termed as the *domain of dependance*. 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 dependance 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, charac- teristic information may flow from left to right or right to left (from figure 3.1). So the problem to well-posed boundary conditions must be handled carefully. We will discuss this in the next few sections.

3.2 Characteristics and Characteristic Waves

3.2.1 What are characteristics?

The concept of characteristics may be introduced from several points of views [25]

- From the physical point of view, a characteristic curve is defined as the path of propagation of physical disturbances.(e.g. in supersonic field disturbances are propagated along the mach lines in flow). It is been shown that mach lines are characteristics for supersonic flows.

- From the heuristic point of view, a characteristic is defined as a curve along which the governing partial differential equation can be converted to an ordinary differential equation.

- From a more rigorous mathematical point of view, a characteristic is a curve across which derivatives of physical properties may be discontinuous, while its property remains itself continuous.

  Thus regions of flow having continuous properties and derivatives within each region but discontinuities in derivatives at the interface, may be joined together along characteristic lines. For example, the Prandtl mayer expansion fan is joined to uniform upstream and downstream flow regions along mach lines.

- From the most rigorous point of view, a characteristic is defined as a curve along which governing PDE reduces to an interior operator, the interior operator known as a compatibility equation. Accordingly, along a characteristic the dependent variable may not be specified arbitrarily, because it must satisfy the compatibility condition.

These concepts are employed for developing numerical procedures for solving hyperbolic partial differential equations.

3.2.2 Characteristic in a single governing PDE

For simplicity, first we will show how a single governing partial differential equation reduces to a ordinary differential equation along a characteristic line, and show that real information gets propagated along these characteristic lines.

Any first order partial differential equation in two dimensions can be written as

$$a\frac{\partial f}{\partial x} + b\frac{\partial f}{\partial y} + c = 0 \tag{3.13}$$

where a, b, c may be constant or function of x, y. The equation 3.13 can be written as,

$$a\left(\frac{\partial f}{\partial x} + \frac{b}{a}\frac{\partial f}{\partial y} + \frac{c}{a}\right) = 0 \tag{3.14}$$

while the total differential of f can be written as,

$$df = \frac{\partial f}{\partial x}dx + \frac{\partial f}{\partial y}dy \tag{3.15}$$

and thus

$$\frac{\mathrm{d}f}{\mathrm{d}x} = \frac{\partial f}{\partial x} + \frac{\partial f}{\partial y}\frac{\mathrm{d}y}{\mathrm{d}x} \tag{3.16}$$

$$\frac{\partial f}{\partial x} = \frac{\mathrm{d}f}{\mathrm{d}x} - \frac{\partial f}{\partial y}\frac{\mathrm{d}y}{\mathrm{d}x} \tag{3.17}$$

Substituting this 3.17 value in equation 3.14 we get,

$$\frac{\mathrm{d}f}{\mathrm{d}x} - \frac{\partial f}{\partial y}\frac{\mathrm{d}y}{\mathrm{d}x} + \frac{b}{a}\frac{\partial f}{\partial y} + \frac{c}{a} = 0 \tag{3.18}$$

$$\frac{\mathrm{d}f}{\mathrm{d}x} + \left(\frac{b}{a} - \frac{\mathrm{d}y}{\mathrm{d}x}\right)\frac{\partial f}{\partial x} + \frac{c}{a} = 0 \tag{3.19}$$

Equation 3.19 can be written as the ODE

$$\frac{\mathrm{d}f}{\mathrm{d}x} + \frac{c}{a} = 0 \tag{3.20}$$

if

$$\frac{\mathrm{d}y}{\mathrm{d}x} = \frac{b}{a} \tag{3.21}$$

So we can conclude that along this curve (i.e., characteristic) satisfying the condition $\frac{\mathrm{d}y}{\mathrm{d}x} = \frac{b}{a}$ the PDE 3.19 will become the ODE 3.20

Now consider a case of

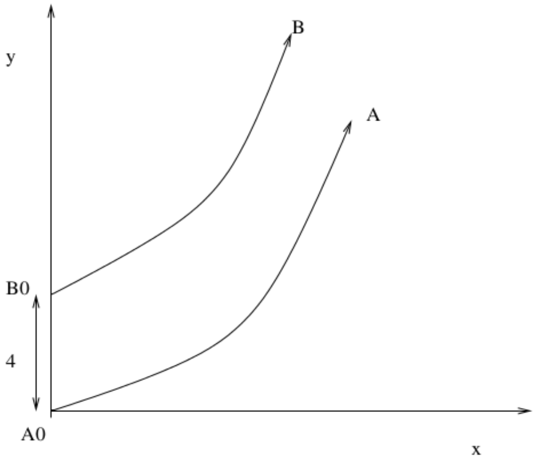
$$f = u \qquad a = 2x \qquad b = -3x^2$$

Then the equation 3.13 becomes,

$$\frac{\partial u}{\partial x} + 2x\frac{\partial u}{\partial x} - 3x^2 = 0 \tag{3.22}$$

and equation 3.19 reduces to,

$$\frac{\mathrm{d}u}{\mathrm{d}x} + \left(2x - \frac{\mathrm{d}y}{\mathrm{d}x}\right)\frac{\partial u}{\partial y} - 3x^2 = 0 \tag{3.23}$$



and can be written as

$$\mathrm{d}u \;=\; \left(\left(2x - \frac{\mathrm{d}y}{\mathrm{d}x}\right)\frac{\partial u}{\partial y} \;-\; 3x^2\right) dx \tag{3.24}$$

Carrying-out integration along lines

$$\begin{aligned} y &= x^2 \\ y &= x^2 \;+ 4 \end{aligned}$$

Note that these curves satisfies the condition (3.21)

Case 1:

$$\begin{aligned} y &= x^2 \\ \frac{\mathrm{d}y}{\mathrm{d}x} &= 2x \end{aligned}$$

and

$$\begin{aligned} \mathrm{d}u &= -\left(0 - 3x^2\right)\mathrm{d}x \\ \int_{u_0}^{u_x} \mathrm{d}u &= \int_0^x 3x^2 \mathrm{d}x \\ u_x \;- u_0 &= x_3 \end{aligned}$$

and,

$$u_x \;=\; u_0 \;+\; 3x^2$$

and thus along line A,

$$u_A \;=\; u_A 0 \;+\; 3x^2$$

and along line B,

$$u_B \;=\; u_B 0 \;+\; 3x^2$$

These expressions show the simplest picture of propagation of initial conditions and boundary conditions into the domain in a hyperbolic system.

3.3 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:

$$[A]\frac{\partial w_i}{\partial x} + [B]\frac{\partial w_i}{\partial y} + [C]\frac{\partial w_i}{\partial z} = 0 \tag{3.25}$$

Unsteady state:

$$[I]\frac{\partial w_i}{\partial t} + [A]\frac{\partial w_i}{\partial x} + [B]\frac{\partial w_i}{\partial y} + [C]\frac{\partial w_i}{\partial z} = 0 \tag{3.26}$$

Diagonalizing the Jacobian matrices, we get:

$$I\frac{\partial w_i}{\partial t} + [Q_a][\lambda_a][Q_a]^{-1}\frac{\partial w_i}{\partial x} + [Q_b][\lambda_b][Q_b]^{-1}\frac{\partial w_i}{\partial y} + [Q_c][\lambda_c][Q_c]^{-1}\frac{\partial w_i}{\partial z} = 0 \tag{3.27}$$

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 (3.25) or (3.26) gets decoupled. Therefore a deeper analysis needs to be done to get eigenvalues and characteristic variables.

3.3.1 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 [6]:

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:

$$\Sigma_j A_j \frac{\partial w}{\partial x_j} = T \qquad j = 1, 2, 3....m+1$$

$$w = \left\{ \begin{array}{c} w_1 \\ w_2 \\ w_3 \\ . \\ . \\ . \\ w_n \end{array} \right\} \tag{3.28}$$

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 $x_j(j = 1, ..., m+1)$, defined by

$$w = \hat{U} e^{i(\vec{n} \cdot \vec{x})} \tag{3.29}$$

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 (3.28) and find the values of n satisfying the resulting equation.

The homogenous part of Eq. (3.28) is written as

$$\Sigma_j A_j \frac{\partial w}{\partial x_j} = 0 \qquad j = 1, 2, 3....m + 1 \tag{3.30}$$

and the function (3.29) is a solution of this system of equations if the homogeneous algebraic system of equations:

$$\left[\Sigma_j A_j n_j\right] \hat{U} = 0 \tag{3.31}$$

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

$$det\left[\Sigma_j A_j n_j\right] \tag{3.32}$$

Eq. (3.32) 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 (3.32) 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 (3.32) 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.

3.3.2 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. (3.32). 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

$$\overrightarrow{n} = \frac{\nabla S}{\|\nabla S\|} \tag{3.33}$$

(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. (3.29)*?*

If Eq. (3.33) is introduced in the plane wave Eq. (3.29), a general representation is defined as,

$$w = \hat{U} e^{i(\overrightarrow{x} \cdot \nabla S)} = \hat{U} e^{i(x_j S_j)} \qquad \text{with } S_j \equiv \frac{\partial S}{\partial x_j} \tag{3.34}$$

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

$$S(x_j) = S(0) + \overrightarrow{x} \cdot \nabla S = S(0) + x_j \frac{\partial S}{\partial x_j} = S(0) + x_j n_j \tag{3.35}$$

we observe that along the constant values of the phase of the wave $\phi = \overrightarrow{x} \cdot \nabla S$, 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 $\overrightarrow{n}$.

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:

$$\mathrm{d}S = \nabla S \cdot \mathrm{d}x = \frac{\partial S}{\partial x}\mathrm{d}x + \frac{\partial S}{\partial y}\mathrm{d}y = 0 \tag{3.36}$$

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

$$\frac{\mathrm{d}y}{\mathrm{d}x} = -\frac{S_x}{S_y} = -\frac{n_x}{n_y} \tag{3.37}$$

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

$$\begin{gathered}
\frac{\partial \rho}{\partial t} + u\frac{\partial \rho}{\partial x} + v\frac{\partial \rho}{\partial y} + w\frac{\partial \rho}{\partial z} = 0 \\
\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} + w\frac{\partial u}{\partial z} = -\frac{1}{\rho}\frac{\partial p}{\partial x} \\
\frac{\partial v}{\partial t} + u\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y} + w\frac{\partial v}{\partial z} = -\frac{1}{\rho}\frac{\partial p}{\partial y} \\
\frac{\partial w}{\partial t} + u\frac{\partial w}{\partial x} + v\frac{\partial w}{\partial y} + w\frac{\partial w}{\partial z} = -\frac{1}{\rho}\frac{\partial p}{\partial z} \\
\frac{\partial E}{\partial t} + u\frac{\partial E}{\partial x} + v\frac{\partial E}{\partial y} + w\frac{\partial E}{\partial z} = -\frac{1}{\rho}\left(\frac{\partial pu}{\partial x} + \frac{\partial pv}{\partial y} + \frac{\partial pw}{\partial z}\right)
\end{gathered} \tag{3.38}$$

By following the analysis given in Sec. 3.3, 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. 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. 3.2. 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. 3.1. 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.

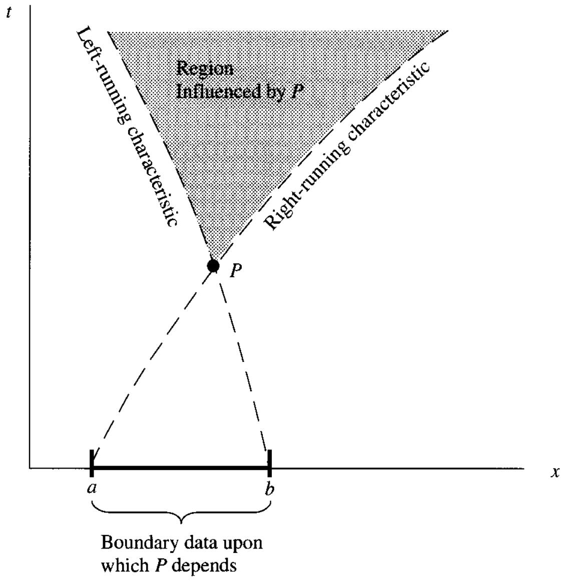


Figure 3.2: Domain and boundaries for the solution of hyperbolic equations. One dimensional unsteady flow. [18]

3.4 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:

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

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 [18].

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.

3.5 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 eigenvlaues of the system, which correspond to the speed of propagation of five characteristic quantities, are given by,

$$\frac{\overrightarrow{u}\cdot\overrightarrow{k}}{k}, \qquad \frac{\overrightarrow{u}\cdot\overrightarrow{k}}{k}, \qquad \frac{\overrightarrow{u}\cdot\overrightarrow{k}}{k}, \qquad \frac{\overrightarrow{u}\cdot\overrightarrow{k}}{k}+c, \qquad \frac{\overrightarrow{u}\cdot\overrightarrow{k}}{k}-c$$

where c is the local sonic speed. [6]

The derivation for the above eigenvalues can be found in the Section of 11.2.1 of [18].

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:

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

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

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

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

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

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 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 $\overrightarrow{n}$, pointing into the computational domain, is defined by the number of characteristics entering the domain.

3.6 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 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 systems.

Chapter 4

Numerical Methodology

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. Another basic requirement for the solution of the Euler equations is to ensure that solution schemes provide an adequate amount of artificial viscosity required for correct and rapid convergence towards a solution. In the present work, the Implicit MacCormack scheme has been chosen to solve the Euler equations, since it is a very robust and tested scheme.

4.1 Governing Equations

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

$$\begin{gathered}
\frac{\partial \rho}{\partial t} + \frac{\partial(\rho u)}{\partial x} + \frac{\partial(\rho v)}{\partial y} + \frac{\partial(\rho w)}{\partial z} = 0 \\
\frac{\partial(\rho u)}{\partial t} + \frac{\partial(\rho u^2 + P)}{\partial x} + \frac{\partial(\rho uv)}{\partial y} + \frac{\partial(\rho uw)}{\partial z} = 0 \\
\frac{\partial(\rho v)}{\partial t} + \frac{\partial(\rho vu)}{\partial x} + \frac{\partial(\rho v^2 + P)}{\partial y} + \frac{\partial(\rho vw)}{\partial z} = 0 \\
\frac{\partial(\rho w)}{\partial t} + \frac{\partial(\rho wu)}{\partial x} + \frac{\partial(\rho wv)}{\partial y} + \frac{\partial(\rho w^2 + P)}{\partial z} = 0 \\
\frac{\partial(\rho E)}{\partial t} + \frac{\partial(\rho uH)}{\partial x} + \frac{\partial(\rho vH)}{\partial y} + \frac{\partial(\rho wH)}{\partial z} = 0
\end{gathered} \tag{4.1}$$

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

4.2 Discretization of Governing Equation

The equations can be written in compact form as

$$\frac{\partial \{W_i\}}{\partial t} + \frac{\partial \{F_{xi}\}}{\partial x} + \frac{\partial \{F_{yi}\}}{\partial y} + \frac{\partial \{F_{zi}\}}{\partial z} = 0 \tag{4.2}$$

$$W \equiv \left\{ \begin{array}{c} \rho \\ \rho u \\ \rho v \\ \rho w \\ \rho E \end{array} \right\}$$

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

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

$$\frac{\partial W_i}{\partial t} + \nabla \cdot \mathbf{F_i} = 0 \tag{4.3}$$

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.

$$\oint_V \left(\frac{\partial W_i}{\partial t} + \nabla \cdot \mathbf{F}_i \right) \mathrm{d}V = 0$$

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

$$\oint_V \frac{\partial W_i}{\partial t} \mathrm{d}V + \oint_S \mathbf{F}_i \cdot \mathrm{d}\overrightarrow{S} \;=\; 0$$

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

$$\frac{\partial}{\partial t} \oint_V W_i \mathrm{d}V + \oint_S \mathbf{F}_i \cdot \mathrm{d}\overrightarrow{S} = 0$$

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.

$$\underbrace{\frac{\partial}{\partial t}\oint_V W_i \mathrm{d}V}_{Temporal\ Part} + \underbrace{\oint_S \mathbf{F}_i \cdot \mathrm{d}\overrightarrow{S}}_{Convective\ part} = 0$$

Temporal term:

The volume averaged value of conservative variable can be written for the pth cell as:

$$\frac{1}{V_p}\oint_{V_p} W \mathrm{d}V = W_p$$

thus,

$$\oint_{V_p} W \mathrm{d}V = V_p W_p$$

where, V_p is the volume of the pth cell, and W_p is the value of its cell center.

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:

There are two methods to calculate the value of convective part at new time level depending upon time value of flux as,

1. Implicit: where the flux variable are taken to be at the new (unknown) time-level.

2. Explicit: where the flux variable are taken to be at the old (known) time-level.

In this study the implicit method is applied to discretize the convective part.

Equation 4.3 is integrated in time by using implicit method and written as,

$$V\,\frac{W^{n+1} - W^n}{\Delta t} + \nabla.(F^{n+1}) = 0 \tag{4.4}$$

with a time step of size Δt. The superscript n refers to current time level and the result is a nonlinear system of algebraic equations, which calls for nonlinear iterations in each time step. But nonlinear iterations are computationally expensive and have poor convergence. To overcome this problem, we assumes sufficient smoothness and linearizes the equations around the current solution W^n by a Taylor series expansion of the fluxes

$$F^{n+1} = F^n + \left(\frac{\partial F}{\partial W}\right)^n (W^{n+1} - W^n) + \bigcirc(\left\|W^{n+1} - W^n\right\|^2) \tag{4.5}$$

Substitution of equation 4.5 into the nonlinear equations 4.4 leads to a linear algebraic system

$$\begin{aligned} V\,\frac{W^{n+1} - W^n}{\Delta t} + \nabla.\left(F^n + \left(\frac{\partial F}{\partial W}\right)^n \left(W^{n+1} - W^n\right)\right) &= 0 \\ \textit{or}\;\; V\,\frac{W^{n+1} - W^n}{\Delta t} + \nabla.\left(\frac{\partial F}{\partial W}\right)^n \left(W^{n+1} - W^n\right) &= -\nabla.F^n \end{aligned}$$

Consider

$$\delta W^{n+1} \equiv W^{n+1} - W^n$$

Substituting δW^{n+1} in main equation,

$$\begin{aligned} V\,\frac{\delta W^{n+1}}{\Delta t} + \nabla.\left(\frac{\partial F}{\partial W}\right)^n \delta W^{n+1} &= -\nabla.F^n \\ \left[I + \frac{\Delta t}{V}\,\nabla.\left(\frac{\partial F}{\partial W}\right)^n\right] \delta W^{n+1} &= -\frac{\Delta t}{V}\,\nabla.F^n \end{aligned}$$

where $\frac{\partial F}{\partial W}$ is the Jacobian of flux F.

In the convective term, the explicit term $\bigtriangledown.F^n$ 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:

$$\oint_S \overrightarrow{F_i} \cdot \mathrm{d}\overrightarrow{S} = \oint_e \overrightarrow{F_i} \cdot \mathrm{d}\overrightarrow{S}_e + \oint_w \overrightarrow{F_i} \cdot \mathrm{d}\overrightarrow{S}_w + \oint_n \overrightarrow{F_i} \cdot \mathrm{d}\overrightarrow{S}_n + \oint_s \overrightarrow{F_i} \cdot \mathrm{d}\overrightarrow{S}_s + \\ \oint_t \overrightarrow{F_i} \cdot \mathrm{d}\overrightarrow{S}_t + \oint_b \overrightarrow{F_i} \cdot \mathrm{d}\overrightarrow{S}_b$$

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

$$\oint_{Sf} \overrightarrow{F_i} \cdot \mathrm{d}\overrightarrow{S}_f = \oint_S F_{ix} \mathrm{d}S_x + \oint_S F_{iy} \mathrm{d}S_y + \oint_S F_{iz} \mathrm{d}S_z$$

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}:

$$\frac{1}{S_f} \oint_{Sf} F_i \mathrm{d}\overrightarrow{S}_f \ = \ F_{if}$$

Therefore we can write,

$$\oint_{Sf} \overrightarrow{F_i} \cdot \mathrm{d}\overrightarrow{S}_f = F_{ix} S_{fx} + F_{iy} S_{fy} + F_{iz} S_{fz}$$

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

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

Now, putting the discretized convective terms together, the Explicit term can be written in discretized form as:

$$\Delta F^n = -\sum_f (F_{fx} S_{fx} + F_{fy} S_{fy} + F_{fz} S_{fz}) \tag{4.6}$$

MacCormack [22] proposed a two-step approach to solve the wave equation, with a finite-difference method. It is known to be a robust scheme that gives stable results with good accuracy when provided with some artificial dissipation. As the scheme is a finite-difference method, we need to modify it for the finite-volume method, which shall be done below. First, however, we will introduce the MacCormack finite-difference scheme for the wave equation, and then extend it to the finite-volume method for Euler equations in the later sections.

4.3 MacCormack Finite Difference Scheme

MacCormack′s scheme solves hyperbolic problems in two steps, popularly known as the predictor-corrector approach. It falls in the category of multi-step central schemes.

Consider a simple one dimensional model initial value problem in 1D:

$$\frac{\partial u}{\partial t} + c\frac{\partial u}{\partial x} = 0 \tag{4.7}$$

with an initial condition u(x,0) = $u_0(x)$ The explicit MacCormack scheme is

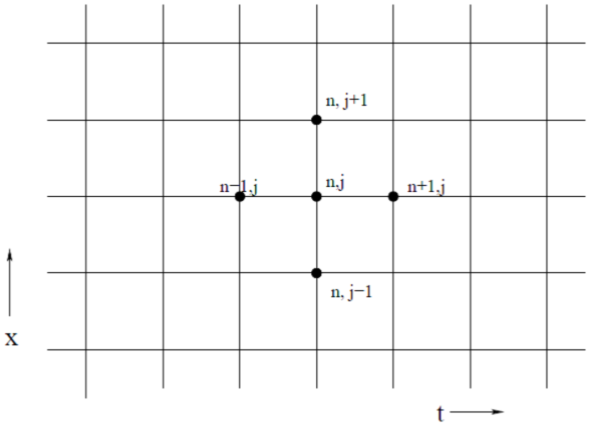


Figure 4.1: Finite-difference grid

realized in two steps:

Predictor:

$$\begin{aligned} \triangle u_i^n &= -\frac{c\triangle t}{\triangle x}\left(u_{i+1}^n - u_i^n\right) \\ u_i^{\overline{n+1}} &= u_i^n + \triangle u_i^n \end{aligned} \tag{4.8}$$

where $u_i^{\overline{n+1}}$ is the so-called predicted value of the solution at the $n+1$ time-level, obtained explicitly in step 1 and $\triangle u_i^n \equiv u_i^{\overline{n+1}} - u_i^n$,

Corrector:

$$\begin{aligned} \triangle u_i^{\overline{n+1}} &= -\frac{c\triangle t}{\triangle x}\left(u_i^{\overline{n+1}} - u_{i-1}^{\overline{n+1}}\right) \\ u_i^{n+1} &= \frac{1}{2}\left(u_i^n + u_i^{\overline{n+1}} + \triangle u_i^{\overline{n+1}}\right) \end{aligned} \tag{4.9}$$

The explicit scheme is stable under the iCFL condition:

$$\triangle t \le \frac{1}{(c/\triangle x)}$$

The implicit scheme is obtained by replacing one-sided differences in convective terms in predictor by:

$$(1-\alpha)\frac{c\triangle t}{\triangle x}\left(u_{i+1}^{n} - u_i^{n}\right) + \alpha\frac{c\triangle t}{\triangle x}\left(\overline{u_{i+1}^{n+1}} - \overline{u_i^{n+1}}\right)$$

or

$$\left(1+\frac{\lambda\triangle t}{\triangle x}\right)\delta\overline{u_{i+1}^{n+1}} + \frac{\lambda\triangle t}{\triangle x}\delta\overline{u_i^{n+1}}$$

where

$$\delta\overline{u_i^{n+1}} \equiv \overline{u_i^{n+1}} - u_i^{n}, \qquad \delta\overline{u_{i+1}^{n+1}} \equiv \overline{u_{i+1}^{n+1}} - u_{i+1}^{n}, \qquad \lambda = \alpha\left|c\right|$$

and similarly for corrector

$$(1-\alpha)\frac{c\triangle t}{\triangle x}\left(\overline{u_{i+1}^{n+1}} - \overline{u_i^{n+1}}\right) + \alpha\frac{c\triangle t}{\triangle x}\left(u_{i+1}^{n+1} - u_i^{n+1}\right)$$

or

$$\left(1+\frac{\lambda\triangle t}{\triangle x}\right)\delta u_{i+1}^{n+1} + \frac{\lambda\triangle t}{\triangle x}\delta u_i^{n+1}$$

where

$$\delta u_i^{n+1} \equiv u_i^{n+1} - \overline{u_i^{n+1}}, \qquad \delta u_{i+1}^{n+1} \equiv u_{i+1}^{n+1} - \overline{u_{i+1}^{n+1}}, \qquad \lambda = \alpha\left|c\right|$$

The final implicit finite-difference scheme is

Predictor:

$$\begin{aligned} \triangle u_i^{n} &= -\frac{a\triangle t}{\triangle x}\left(u_{i+1}^{n} - u_i^{n}\right) \\ \left(1+\lambda\frac{\triangle t}{\triangle x}\right)\delta\overline{u_i^{n+1}} &= \triangle u_i^{n} + \lambda\frac{\triangle t}{\triangle x}\delta\overline{u_{i+1}^{n+1}} \\ \overline{u_i^{n+1}} &= u_i^{n} + \delta\overline{u_i^{n+1}} \end{aligned} \tag{4.10}$$

Corrector:

$$\begin{aligned} \triangle\overline{u_i^{n+1}} &= -\frac{a\triangle t}{\triangle x}\left(\overline{u_i^{n+1}} - \overline{u_{i-1}^{n+1}}\right) \\ \left(1+\lambda\frac{\triangle t}{\triangle x}\right)\delta u_i^{n+1} &= \triangle\overline{u_i^{n+1}} + \lambda\frac{\triangle t}{\triangle x}\delta u_{i-1}^{n+1} \\ u_i^{n+1} &= \frac{1}{2}\left(u_i^{n} + \overline{u_i^{n+1}} + \delta u_i^{n+1}\right) \end{aligned} \tag{4.11}$$

The predictor step is evaluated starting at the greatest index i using an appropriate boundary condition and going to lowest index. The corrector step is evaluated in the similar manner starting with boundary condition for lowest index and going to greatest one.

The linear scheme is unconditionally stable provided that the implicit blending parameter λ is chosen such that

$$\lambda \geq \frac{1}{2} max \left(|c| - \frac{\triangle x}{\triangle t}, 0 \right) \tag{4.12}$$

All three steps in predictor can be evaluated together during one backward sweep through the mesh, i.e. it is not necessary to solve any system of linear equations. The same is valid for the corrector, which can be again realized by one forward sweep.

4.4 Implicit MacCormack scheme in FVM

In this section we will see how to apply the MacCormack scheme in the finite volume methodology. Since the MacCormack scheme is second order accurate in space and time, oscillations are observed in solution having abrupt step-changes in value.

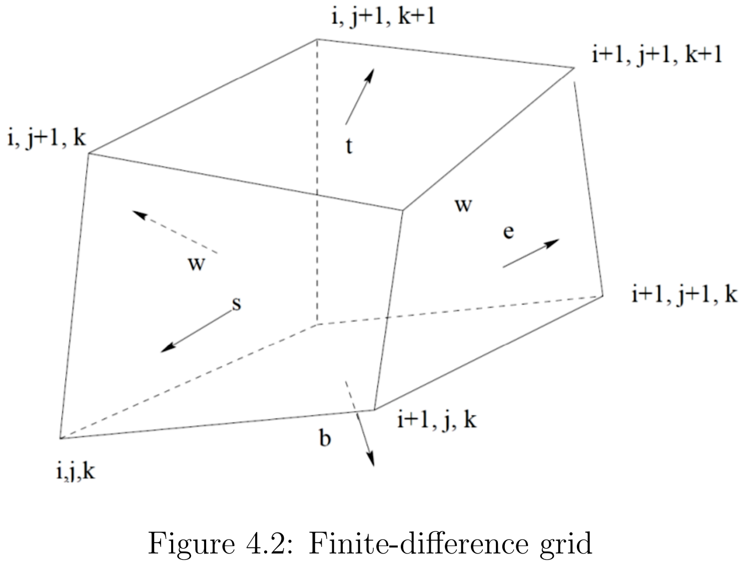


Figure 4.2: Finite-difference grid

The implicit MacCormack scheme in finite volume formulation is

Predictor:

$$\triangle W^{n}_{i,j,k} \;=\; -\Delta t\left(\frac{\Delta_{+}F^{n}_{x_{i,j,k}}}{\Delta x}+\frac{\Delta_{+}F^{n}_{y_{i,j,k}}}{\Delta y}+\frac{\Delta_{+}F^{n}_{z_{i,j,k}}}{\Delta z}\right) \tag{4.13}$$

$$\left[I-\frac{\triangle t}{\triangle x}D^{+}_{I}|A|^{n}_{i,j,k}\right]\left[I-\frac{\triangle t}{\triangle y}D^{+}_{J}|B|^{n}_{i,j,k}\right]\left[I-\frac{\triangle t}{\triangle z}D^{+}_{K}|C|^{n}_{i,j,k}\right]\delta W^{n}_{i,j,k}=\triangle W^{n}_{i,j,k} \tag{4.14}$$

$$W^{\overline{n+1}}_{i,j,k}=W^{n}_{i,j,k}+\delta W^{\overline{n+1}}_{i,j,k} \tag{4.15}$$

Corrector:

$$\triangle W^{\overline{n+1}}_{i,j,k} \;=\; -\Delta t\left(\frac{\Delta_{-}F^{\overline{n+1}}_{x_{i,j,k}}}{\Delta x}+\frac{\Delta_{-}F^{\overline{n+1}}_{y_{i,j,k}}}{\Delta y}+\frac{\Delta_{-}F^{\overline{n+1}}_{z_{i,j,k}}}{\Delta z}\right) \tag{4.16}$$

$$\left[I+\frac{\triangle t}{\triangle x}D^{-}_{I}|A|^{\overline{n+1}}_{i,j,k}\right]\left[I+\frac{\triangle t}{\triangle x}D^{-}_{J}|B|^{\overline{n+1}}_{i,j,k}\right]\left[I+\frac{\triangle t}{\triangle z}D^{-}_{K}|C|^{\overline{n+1}}_{i,j,k}\right]\delta W^{\overline{n+1}}_{i,j,k} \;=\; \triangle W^{\overline{n+1}}_{i,j,k} \tag{4.17}$$

$$W^{n+1}_{i,j,k}=(W^{n}_{i,j,k}+W^{\overline{n+1}}_{i,j,k}+\delta W^{\overline{n+1}}_{i,j,k})/2 \tag{4.18}$$

Where the operators δ and Δ denote the implicit and explicit temporal difference operators, respectively. Operators D^{+}_{I}, D^{-}_{I}, $D_{J}+$, D^{-}_{J}, D^{+}_{K}, D^{-}_{K} are one-sided forward and backward differences in each index dimension. $|A|$, $|B|\; and\; |C|$ are diagonalized jacobian matrices. All these operators are explained later.

The values of $\frac{\Delta_{+}}{\Delta x}$, $\frac{\Delta_{-}}{\Delta x}$, $\frac{\Delta_{+}}{\Delta y}$, $\frac{\Delta_{-}}{\Delta y}$, $\frac{\Delta_{+}}{\Delta z}$, $\frac{\Delta_{-}}{\Delta z}$ operators are:

$$\begin{aligned}
\frac{\Delta_{+}F_{x_{i,j,k}}}{\Delta x} &= \frac{F_{x_{i+1,j,k}}-F_{x_{i,j,k}}}{\Delta x}\\
\frac{\Delta_{-}F_{x_{i,j,k}}}{\Delta x} &= \frac{F_{x_{i,j,k}}-F_{x_{i-1,j,k}}}{\Delta x}\\
\frac{\Delta_{+}F_{y_{i,j,k}}}{\Delta y} &= \frac{F_{y_{i,j+1,k}}-F_{y_{i,j,k}}}{\Delta y}\\
\frac{\Delta_{-}F_{y_{i,j,k}}}{\Delta y} &= \frac{F_{y_{i,j,k}}-F_{x_{i,j-1,k}}}{\Delta y}\\
\frac{\Delta_{+}F_{x_{i,j,k}}}{\Delta z} &= \frac{F_{z_{i,j,k+1}}-F_{z_{i,j,k}}}{\Delta z}\\
\frac{\Delta_{-}F_{x_{i,j,k}}}{\Delta z} &= \frac{F_{z_{i,j,k}}-F_{z_{i,j,k-1}}}{\Delta z}
\end{aligned}$$

The values of D_I^+, D_I^-, D_J+, D_J^-, D_K^+, D_K^- operators are:

$$\begin{aligned}
D_I^+ Z_{i,j,k} &= \frac{|A|_{i+1,j,k} - |A|_{i,j,k}}{\Delta x} \\
D_I^- Z_{i,j,k} &= \frac{|A|_{i,j,k} - |A|_{i-1,j,k}}{\Delta x} \\
D_J^+ Z_{i,j,k} &= \frac{|B|_{i,j+1,k} - |B|_{i,j,k}}{\Delta y} \\
D_J^- Z_{i,j,k} &= \frac{|B|_{i,j,k} - |B|_{i,j-1,k}}{\Delta y} \\
D_K^+ Z_{i,j,k} &= \frac{|C|_{i,j,k+1} - |C|_{i,j,k}}{\Delta z} \\
D_K^- Z_{i,j,k} &= \frac{|C|_{i,j,k} - |C|_{i,j,k-1}}{\Delta z}
\end{aligned}$$

Here the Jacobians of flux F is written as matrices A, B, C so that

$$\frac{\partial F_x}{\partial W} = A \qquad \frac{\partial F_y}{\partial W} = B \qquad \frac{\partial F_z}{\partial W} = C$$

Matrices $|A|$, $|B|$ *and* $|C|$ have positive eigenvalues and are related to the Jacobians A, B and C.

The inviscid jacobians A, B and C can be diagonalize by S_x, S_y, S_z. The matrices A, B, C can be witten as,

$$A = S_x^{-1} \Lambda_A S_x \qquad B = S_y^{-1} \Lambda_B S_y \qquad C = S_z^{-1} \Lambda_C S_z$$

The matrices S_x, S_y *and* S_z are each expressed as the product of two matrices.

They can be written as,

$$S_x = \begin{pmatrix} 1 & 0 & 0 & 0 & \frac{-1}{c^2} \\ 0 & \rho c & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & -\rho c & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ \frac{-u}{\rho} & \frac{1}{\rho} & 0 & 0 & 0 \\ \frac{-v}{\rho} & 0 & \frac{1}{\rho} & 0 & 0 \\ \frac{-w}{\rho} & 0 & \frac{1}{\rho} & 0 & 0 \\ \alpha\beta & -u\beta & -v\beta & -w\beta & \beta \end{pmatrix} \tag{4.19}$$

$$S_y = \begin{pmatrix} 1 & 0 & 0 & 0 & \frac{-1}{c^2} \\ 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & \rho c & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & -\rho c & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ \frac{-u}{\rho} & \frac{1}{\rho} & 0 & 0 & 0 \\ \frac{-v}{\rho} & 0 & \frac{1}{\rho} & 0 & 0 \\ \frac{-w}{\rho} & 0 & \frac{1}{\rho} & 0 & 0 \\ \alpha\beta & -u\beta & -v\beta & -w\beta & \beta \end{pmatrix} \tag{4.20}$$

$$S_z = \begin{pmatrix} 1 & 0 & 0 & 0 & \frac{-1}{c^2} \\ 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & \rho c & 0 \\ 0 & 0 & 0 & -\rho c & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ \frac{-u}{\rho} & \frac{1}{\rho} & 0 & 0 & 0 \\ \frac{-v}{\rho} & 0 & \frac{1}{\rho} & 0 & 0 \\ \frac{-w}{\rho} & 0 & \frac{1}{\rho} & 0 & 0 \\ \alpha\beta & -u\beta & -v\beta & -w\beta & \beta \end{pmatrix} \tag{4.21}$$

$$\Lambda_A = \begin{pmatrix} u & 0 & 0 & 0 & 0 \\ 0 & u+c & 0 & 0 & 0 \\ 0 & 0 & u & 0 & 0 \\ 0 & 0 & 0 & u & 0 \\ 0 & 0 & 0 & 0 & u-c \end{pmatrix}, \quad \Lambda_B = \begin{pmatrix} v & 0 & 0 & 0 & 0 \\ 0 & u & 0 & 0 & 0 \\ 0 & 0 & v+c & 0 & 0 \\ 0 & 0 & 0 & v & 0 \\ 0 & 0 & 0 & 0 & v-c \end{pmatrix} \tag{4.22}$$

$$\Lambda_C = \begin{pmatrix} w & 0 & 0 & 0 & 0 \\ 0 & w & 0 & 0 & 0 \\ 0 & 0 & w & 0 & 0 \\ 0 & 0 & 0 & w+c & 0 \\ 0 & 0 & 0 & 0 & w-c \end{pmatrix} \tag{4.23}$$

and where $c = \sqrt{\gamma p/\rho}$ is the speed of sound, $\alpha = \frac{1}{2}\left(u^2 + v^2 + w^2\right) and \beta = \gamma - 1$.

The inverses S_x^{-1}, $S_y^{-1}\, and\, S_z^{-1}$ are simply the inverse matrix of S_x, $S_y\, and\, S_z$ respectively.

The matrices $|A|\, and\, |B|$ are defined by

$$|A| = S_x^{-1} D_A S_x \qquad |B| = S_y^{-1} D_B S_y \qquad |C| = S_z^{-1} D_C S_z$$

where $D_A,\ D_B\ and\ D_C$ are diagonal matrices defined by

$$D_A = \begin{pmatrix} \lambda_{A1} & 0 & 0 & 0 & 0 \\ 0 & \lambda_{A2} & 0 & 0 & 0 \\ 0 & 0 & \lambda_{A3} & 0 & 0 \\ 0 & 0 & 0 & \lambda_{A4} & 0 \\ 0 & 0 & 0 & 0 & \lambda_{A5} \end{pmatrix} \tag{4.24}$$

$$D_B = \begin{pmatrix} \lambda_{B1} & 0 & 0 & 0 & 0 \\ 0 & \lambda_{B2} & 0 & 0 & 0 \\ 0 & 0 & \lambda_{B3} & 0 & 0 \\ 0 & 0 & 0 & \lambda_{B4} & 0 \\ 0 & 0 & 0 & 0 & \lambda_{B5} \end{pmatrix} \tag{4.25}$$

$$D_C = \begin{pmatrix} \lambda_{C1} & 0 & 0 & 0 & 0 \\ 0 & \lambda_{C2} & 0 & 0 & 0 \\ 0 & 0 & \lambda_{C3} & 0 & 0 \\ 0 & 0 & 0 & \lambda_{C4} & 0 \\ 0 & 0 & 0 & 0 & \lambda_{C5} \end{pmatrix} \tag{4.26}$$

and

$$\begin{aligned} \lambda_{A1} &= max\left\{|u| - \frac{1}{2}\frac{\Delta x}{\Delta t},\ 0\right\} \\ \lambda_{A2} &= max\left\{|u+c| - \frac{1}{2}\frac{\Delta x}{\Delta t},\ 0\right\} \\ \lambda_{A3} &= max\left\{|u| - \frac{1}{2}\frac{\Delta x}{\Delta t},\ 0\right\} \\ \lambda_{A4} &= max\left\{|u| - \frac{1}{2}\frac{\Delta x}{\Delta t},\ 0\right\} \\ \lambda_{A5} &= max\left\{|u-c| - \frac{1}{2}\frac{\Delta x}{\Delta t},\ 0\right\} \end{aligned}$$

$$\begin{aligned} \lambda_{B1} &= max\left\{|v| - \frac{1}{2}\frac{\Delta y}{\Delta t},\ 0\right\} \\ \lambda_{B2} &= max\left\{|v| - \frac{1}{2}\frac{\Delta y}{\Delta t},\ 0\right\} \\ \lambda_{B3} &= max\left\{|v+c| - \frac{1}{2}\frac{\Delta y}{\Delta t},\ 0\right\} \\ \lambda_{B4} &= max\left\{|v| - \frac{1}{2}\frac{\Delta y}{\Delta t},\ 0\right\} \\ \lambda_{B5} &= max\left\{|v-c| - \frac{1}{2}\frac{\Delta y}{\Delta t},\ 0\right\} \end{aligned}$$

$$\begin{aligned}
\lambda_{C1} &= max\left\{|w| - \frac{1}{2}\frac{\Delta z}{\Delta t},\, 0\right\} \\
\lambda_{C2} &= max\left\{|w| - \frac{1}{2}\frac{\Delta z}{\Delta t},\, 0\right\} \\
\lambda_{C3} &= max\left\{|w| - \frac{1}{2}\frac{\Delta z}{\Delta t},\, 0\right\} \\
\lambda_{C4} &= max\left\{|w+c| - \frac{1}{2}\frac{\Delta z}{\Delta t},\, 0\right\} \\
\lambda_{C5} &= max\left\{|w-c| - \frac{1}{2}\frac{\Delta z}{\Delta t},\, 0\right\}
\end{aligned}$$

The Jacobian matrix formulation is explained in Appendix A.

For regions of the flow in which Δt satisfies the following explicit stability conditions

$$\Delta t \leq \frac{1}{2}\frac{\Delta x}{(|u|+c)} \qquad \Delta t \leq \frac{1}{2}\frac{\Delta y}{(|v|+c)} \qquad \Delta t \leq \frac{1}{2}\frac{\Delta z}{(|w|+c)} \tag{4.27}$$

all λ_A, $\lambda_B\, and\, \lambda_C$ vanish and the set of Implicit equations reduces to the explicit equations with simple solution. For other regions in which neither relation is satisfied, the resulting difference equations are either upper or lower block bidiagonal equations with fairly straightforward solutions.

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

$$\frac{\partial u}{\partial t} + a\frac{\partial u}{\partial x} = 0 \tag{4.28}$$

is shown below

$$\begin{aligned}
\frac{\partial u}{\partial t} + a\frac{\partial u}{\partial x} = \frac{a\Delta x}{2}(1-\nu)\frac{\partial^2 u}{\partial x^2} + \frac{a(\Delta x)_2}{6}(3\nu - 2\nu_2 - 1)\frac{\partial^3 u}{\partial x^3} \\
+O[(\Delta t)^3, (\Delta t)^2(\Delta x), (\Delta t)(\Delta x)^2, (\Delta x)^3]
\end{aligned} \tag{4.29}$$

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 B.

4.5 Solution Procedure

Predictor: Let us solve first the predictor step of Eqs. 4.13 and 4.14 assuming At satisfies neither of Eqs. 4.27.

$\triangle W^n_{i,j,k}$ value can be found explicitly using 4.6.

$$\begin{aligned}\triangle W^n_{i,j,k} &= F^n_{ex}S_{ex} + F^n_{ey}S_{ey} + F^n_{ez}S_{ez} + F^n_{wx}S_{wx} + F^n_{wy}S_{wy} + F^n_{wz}S_{wz}\\ &\quad + F^n_{nx}S_{nx} + F^n_{ny}S_{ny} + F^n_{nz}S_{nz} + F^n_{sx}S_{sx} + F^n_{sy}S_{sy} + F^n_{sz}S_{sz}\\ &\quad + F^n_{tx}S_{tx} + F^n_{ty}S_{ty} + F^n_{tz}S_{tz} + F^n_{bx}S_{bx} + F^n_{by}S_{by} + F^n_{bz}S_{bz}\end{aligned}$$

To solve the equation 4.14 consider

$$\delta W^*_{i,j,k} = \left(I - \frac{\Delta t}{\Delta y} D^+_J \left|B\right|_{i,j,k}\right)\left(I - \frac{\Delta t}{\Delta z} D^+_K \left|C\right|_{i,j,k}\right)\delta W^{\overline{n+1}}_{i,j,k} \qquad (4.30)$$

Substituting eqn. 4.30 value to 4.14 we get,

$$\left[I - \frac{\Delta t}{\triangle x} D^+_I |A|^n_{i,j,k}\right] \delta W^*_{i,j,k} = \triangle W^n_{i,j,k} \qquad (4.31)$$

Now substituting D^+_I value in this equation we get

$$\left[I - \frac{\triangle t}{\triangle x}\left(|A|^n_{i+1,j,k} - |A|^n_{i,j,k}\right)\right] \delta W^*_{i,j,k} = \triangle W^n_{i,j,k} \qquad (4.32)$$

Rearranging equation 4.32 we can write

$$\left(I + \frac{\triangle t}{\triangle x}|A|^n_{i,j,k}\right) \delta W^*_{i,j,k} = \triangle W^n_{i,j,k} + + \frac{\triangle t}{\triangle x}|A|^n_{i+1,j,k}\,\delta W^*_{i+1,j,k} \qquad (4.33)$$

It is an upper bidiagonal equation. The solution for $\delta W^*_{i,j,k}$ can be obtained for each j and k by sweeping in the decreasing i direction.

After obtaining $\delta W^{*}_{i,j,k}$ for all i, j, k then substituting this value in Eqn. 4.30 and we get

$$\left(I \;-\; \frac{\Delta t}{\Delta y} D^{+}_{J}\,|B|_{i,j,k}\right)\,\left(I \;-\; \frac{\Delta t}{\Delta z} D^{+}_{K}\,|C|_{i,j,k}\right)\,\delta W^{\overline{n+1}}_{i,j,k} \;=\; \delta W^{*}_{i,j,k} \qquad (4.34)$$

Let us consider

$$\delta W^{**}_{i,j,k} \;=\; \left(I \;-\; \frac{\Delta t}{\Delta z} D^{+}_{K}\,|C|_{i,j,k}\right)\,\delta W^{\overline{n+1}}_{i,j,k} \qquad (4.35)$$

Substituting this value into Eqn. 4.34 we get,

$$\left(I \;-\; \frac{\Delta t}{\Delta y} D^{+}_{J}\,|B|_{i,j,k}\right)\,\delta W^{**}_{i,j,k} \;=\; \delta W^{*}_{i,j,k} \qquad (4.36)$$

Substituting D^{+}_{J} value in this equation we get

$$\left[I - \frac{\Delta t}{\Delta y}\left(|B|^{n}_{i,j+1,k} \;-\; |B|^{n}_{i,j,k}\right)\right]\,\delta W^{**}_{i,j,k} = \triangle W^{*}_{i,j,k} \qquad (4.37)$$

Rearranging equation 4.37 we can write

$$\left(I + \frac{\Delta t}{\Delta y}|B|^{n}_{i,j,k}\right)\,\delta W^{**}_{i,j,k} = \triangle W^{*}_{i,j,k} \;+\; + \frac{\Delta t}{\Delta y}|B|^{n}_{i,j+1,k}\,\delta W^{**}_{i,j+1,k} \qquad (4.38)$$

We can get the solution for $\delta W^{**}_{i,j,k}$ for each i and k by sweeping in the decreasing j direction.

After obtaining $\delta W^{**}_{i,j,k}$ for all i, j, k then substituting this value in Eqn. 4.35 and we get

$$\left(I \;-\; \frac{\Delta t}{\Delta z} D^{+}_{K}\,|C|_{i,j,k}\right)\,\delta W^{\overline{n+1}}_{i,j,k} \;=\; \delta W^{**}_{i,j,k} \qquad (4.39)$$

Substituting D^{+}_{K} value in this equation we get

$$\left[I - \frac{\Delta t}{\Delta z}\left(|C|^{n}_{i,j,k+1} \;-\; |C|^{n}_{i,j,k}\right)\right]\,\delta W^{\overline{n+1}}_{i,j,k} \;=\; \triangle W^{**}_{i,j,k} \qquad (4.40)$$

Rearranging equation 4.40 we can write

$$\left(I + \frac{\Delta t}{\Delta z}|C|^{n}_{i,j,k}\right)\,\delta W^{\overline{n+1}}_{i,j,k} = \triangle W^{**}_{i,j,k} \;+\; + \frac{\Delta t}{\Delta z}|C|^{n}_{i,j,k+1}\,\delta W^{\overline{n+1}}_{i,j,k+1} \qquad (4.41)$$

We can get the solution for $\delta W^{\overline{n+1}}_{i,j,k}$ for each i and j by sweeping in the decreasing k direction.

Then we can go to the third setp and calculate

$$W^{\overline{n+1}}_{i,j} = W^{n}_{i,j} + \delta W^{\overline{n+1}}_{i,j}$$

In the above procedure, the solution of the block bidiagonal system is carried out making use of the known decomposition of $|A|$, $|B|$, $|C|$ which reduces the computation in the inversion of the block matrices. For example, to solve Eq. 4.33 in the predictor, the equation is rewritten as

$$Sx^{-1}_{i,j,k}\left(I + \frac{\triangle t}{\triangle x}|D_A|^n_{i,j,k}\right) Sx_{i,j,k}\ \delta W^*_{i,j,k} = \triangle W^n_{i,j,k} + \frac{\triangle t}{\triangle x}|A|^n_{i+1,j,k}\ \delta W^*_{i+1,j,k}$$

and can be easily solved as

$$\delta W^*_{i,j,k} = Sx^{-1}_{i,j,k}\left(I + \frac{\triangle t}{\triangle x}|D_A|^n_{i,j,k}\right)^{-1} Sx_{i,j,k}\left[\triangle W^n_{i,j,k} + \frac{\triangle t}{\triangle x}|A|^n_{i+1,j,k}\ \delta W^*_{i+1,j,k}\right]$$

Note that the block matrix inversion is trivial because $Sx^{-1}_{i,j,k} and Sx_{i,j,k}$ are known and $\left(I + \frac{\triangle t}{\triangle x}|D_A|^n_{i,j,k}\right)$ is diagonal. This in fact means that a block bidiagonal matrix inversion is reduced to a scalar bidiagonal matrix inversion.

The procedure to solve this equation 4.33 is as follows:

For each j, k and for i = I, I-1, I-2..... 2, 1

1. $W = \triangle W^n_{i,j,k} + \frac{\triangle t}{\triangle x}|A|^n_{i+1,j,k}\ \delta W^*_{i+1,j,k}$
2. $X = Sx\, W$
3. D_A is calculated using 4.24
4. $Y = \left(I + \frac{\triangle t}{\triangle x}|D_A|^n_{i,j,k}\right)^{-1} X$
5. $\delta W^*_{i,j,k} = Sx^{-1}\, Y$
6. $Z = D_A\, Y$
7. $|A|_{i,j,k}\delta W^*_{i,j,k} = Sx^{-1}\, Z$

Each of the above seven steps requires to calculate $\delta W^*_{i,j,k}$ for each i, j, k. The matrix inversion of step 4 is trivial because the matrix is diagonal. So we use the inversion of a diagonal matrix formula. Let D is a diagonal matrix and

$$D = \begin{pmatrix} a_{11} & 0 & 0 & 0 & 0 \\ 0 & a_{22} & 0 & 0 & 0 \\ 0 & 0 & a_{33} & 0 & 0 \\ 0 & 0 & 0 & a_{44} & 0 \\ 0 & 0 & 0 & 0 & a_{55} \end{pmatrix}$$

Then according to this formula its inverse is given by:

$$D^{-1} = \begin{pmatrix} \frac{1}{a_{11}} & 0 & 0 & 0 & 0 \\ 0 & \frac{1}{a_{22}} & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{a_{33}} & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{a_{44}} & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{a_{55}} \end{pmatrix}$$

Note that the solution $\delta W^*_{i,j,k}$ at grid point i,j,k is obtained at step 5. The flux $|A|_{i,j,k}\delta W^*_{i,j,k}$ to be used in the calculation at grid point i-1, j, k is obtained at step 7.

To start the above procedure, the value of $|A_{I,j,k}|^n \delta W^*_{I,j,k}$ has to be known. So in addition to the conventional boundary conditions, boundary conditions are also required for the implicit operator $\delta W^*_{I,j,k}$. It is called Implicit operator boundary condition and discussed in the next chapter.

4.6 Closure

In this chapter we have seen the detailed formulation of the implicit MacCormack scheme which can be used for the study of compressible flows.

Chapter 5

Boundary Conditions

5.1 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. 3.5 and referring the literature [6] 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.

The second question would be answered in the following subsection.

5.1.1 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

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

Table 5.1: No of boundary condition to be fixed on boundary in Euler system of equation

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 [8]. 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. 5.1.

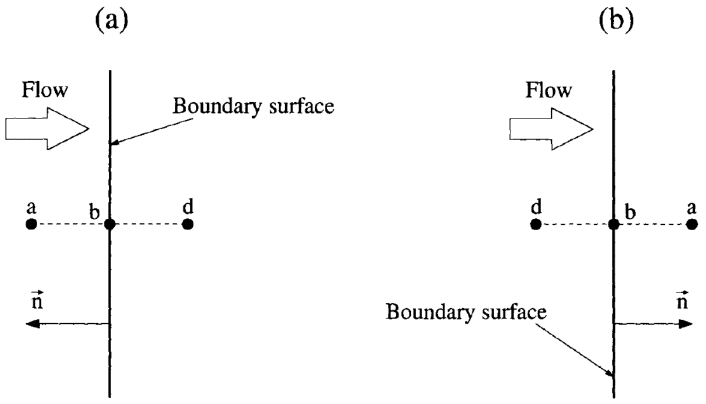


Figure 5.1: Flow Situation at boundary: inflow (a) and outflow (b) situation. Position a is outside, b on the boundary, and the position d is inside the physical domain. The unit normal vector $\vec{n} = [n_x, n_y, n_z]^T$ points out of the domain.[17]

5.1.2 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) ([6], 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 [17].

$$\begin{aligned}
p_b &= \frac{1}{2}\{p_a + p_d - \rho_o c_o[n_x(u_a - u_d) + n_y(v_a - v_d) + n_z(w_a - w_d)]\} \\
\rho_b &= \rho_a + \frac{(p_b - p_a)}{c_o^2} \\
u_b &= u_a - n_x\frac{(p_a - p_b)}{\rho_o c_o} \\
v_b &= v_a - n_y\frac{(p_a - p_b)}{\rho_o c_o} \\
w_b &= w_a - n_z\frac{(p_a - p_b)}{\rho_o c_o}
\end{aligned} \tag{5.1}$$

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. 5.1). 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 [17]. 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.

$$\begin{aligned} p_o &= p(1+\frac{\gamma-1}{2}M^2)^{\frac{\gamma}{\gamma-1}} \\ \rho_o &= \rho(1+\frac{\gamma-1}{2}M^2)^{\frac{1}{\gamma-1}} \end{aligned} \tag{5.2}$$

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. 5.1) are determined by freestream values only.

5.1.3 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

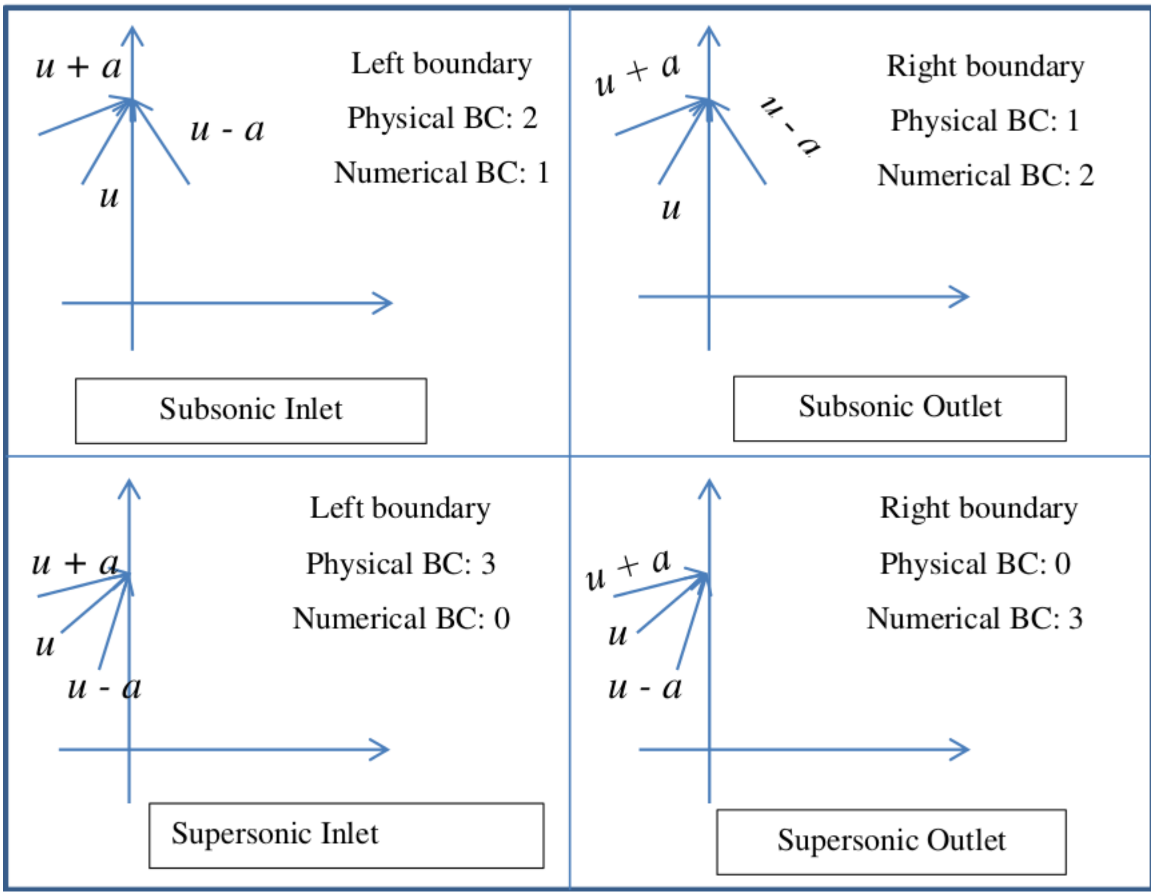


Figure 5.2: Speed regimes and characteristic variables entering and leaving domain [1]

problems. The following numerical formulation is used:

$$\begin{aligned} p_b &= p_a \\ \rho_b &= \rho_d + \frac{(p_b - p_a)}{c_o^2} \\ u_b &= u_d + n_x \frac{(p_d - p_b)}{\rho_o c_o} \\ v_b &= v_d + n_y \frac{(p_d - p_b)}{\rho_o c_o} \\ w_b &= w_d + n_z \frac{(p_d - p_b)}{\rho_o c_o} \end{aligned} \tag{5.3}$$

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 [4], [11] 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 [7] to implement the non-reflecting boundary condition. It has the following form.

$$\frac{\partial u}{\partial t} - \frac{1}{\rho_b a_b}\frac{p_b^{n+1} - p_b^n}{\Delta t} - \frac{\alpha}{\rho a}(p_b^{n+1} - p_b^*) = 0 \tag{5.4}$$

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.

5.1.4 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,

$$F_x = \begin{Bmatrix} \rho u \\ \rho u^2 + p \\ \rho v u \\ \rho u w \\ \rho u H \end{Bmatrix} \quad F_y = \begin{Bmatrix} \rho v \\ \rho v u \\ \rho v^2 + p \\ \rho v w \\ \rho v H \end{Bmatrix} \quad F_z = \begin{Bmatrix} \rho w \\ \rho w u \\ \rho w u \\ \rho w^2 + p \\ \rho w H \end{Bmatrix} \tag{5.5}$$

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

$$(F_x S_{fwx} + F_y S_{fwy} + F_z S_{fwz})_{wall} = \left\{ \begin{array}{c} \rho u S_{fwx} + \rho v S_{fwy} + \rho w S_{fwz} \\ u_w \left(\rho u S_{fwx} + \rho v S_{fwy} + \rho w S_{fwz}\right)_w + p_w S_{fwx} \\ v_w \left(\rho u S_{fwx} + \rho v S_{fwy} + \rho w S_{fwz}\right)_w + p S_{fwy} \\ w_w \left(\rho u S_{fwx} + \rho v S_{fwy} + \rho w S_{fwz}\right)_w + p S_{fwz} \\ H_w \left(\rho u S_{fwx} + \rho v S_{fwy} + \rho w S_{fwz}\right)_w \end{array} \right\} \tag{5.6}$$

Since at the wall

$$\rho u S_{fwx} + \rho v S_{fwy} + \rho w S_{fwz} = 0$$

$$F_x S_{fx} + F_y S_{fy} + F_z S_{fz} = \left\{ \begin{array}{c} 0 \\ p \\ 0 \\ 0 \\ 0 \end{array} \right\} S_{fx} + \left\{ \begin{array}{c} 0 \\ 0 \\ p \\ 0 \\ 0 \end{array} \right\} S_{fy} + \left\{ \begin{array}{c} 0 \\ 0 \\ 0 \\ p \\ 0 \end{array} \right\} S_{fz} \tag{5.7}$$

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

$$V_p \frac{w_p^{n+1} - w_p^n}{\Delta t} = -\sum_f F_{fx} S_{fx} + F_{fy} S_{fy} + F_{fz} S_{fz} \tag{5.8}$$

so whenever for cell p surface f corresponds to the solid boundary, then flux will be calculated from Eq. (5.7). 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.

5.1.5 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. (5.7). 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 center of the associated cell. This is a zero-order extrapolations. We apply the volume extrapolation using the inside two cells. This is second-order and thus supposedly more accurate. *Through numerical experiments we have found that the second order extrapolation esp. at wall and symmetry plane can lead to divergence in the solution.*

5.2 Implicit Operator Boundary Condition

Design of implicit operator boundary treatment is typically guided by characteristic theory. We can write the implicit operator as PDE with respect to time i.e

$$\delta W_{i,j,k} = \frac{\partial W}{\partial t}$$

5.2.1 Inflow BC

Subsonic Inflow

For subsonic inflow, two characteristic entering into domain i.e u, $u+c$ and one characteristic leaving domain $u-c$ (see Fig. 5.2). We can get δW values by interpolating between the inflow plane and the first interior points.

$$\delta W_{boundary} = \delta W_{domain} \tag{5.9}$$

Velocity-Driven Flows:

For velocity driven flows the velocities remains same with respect to time i.e

$$\begin{aligned} \delta u_{i,j,k} &= \frac{\partial u}{\partial t} = 0 \\ \delta v_{i,j,k} &= \frac{\partial v}{\partial t} = 0 \\ \delta w_{i,j,k} &= \frac{\partial w}{\partial t} = 0 \end{aligned}$$

So this velocity condition is updated in Eqn. 5.9.

Pressure-Driven Flows:

For velocity driven flows pressure is always remains same with respect to time i.e

$$\delta p_{i,j,k} = \frac{\partial p}{\partial t} = 0$$

So this pressure condition is updated in Eqn. 5.9.

Supersonic Inflow

For supersonic outflow all eigenvalues are positive and all the characteristic entering the domain (see Fig 5.2). So we set δW values to zero.

5.2.2 Outflow BC

Subsonic Outflow

For subsonic inflow, one characteristic entering into domain i.e $u - c$ and one characteristic leaving domain u, $u + c$ (see Fig. 5.2). We can get δW values by interpolating between the outflow plane and the first interior points.

$$\delta W_{boundary} = \delta W_{domain}$$

But subsonic outflow pressure we need to specify as conventional boundary condition. So this pressure value is constant with respect to time i.e

$$\delta p_{i,j,k} = \frac{\partial p}{\partial t} = 0$$

We add this pressure condition in the calculation of $\delta W_{i,j,k}$.

Supersonic Outflow

For supersonic outflow all eigenvalues are positive and all the characteristic leaving the domain (see Fig 5.2). So we set δW values to zero.

5.2.3 Wall Boundary

The eigenvalues of the Jacobian matrix for the full Euler system are

$$u_n,\ u_n,\ ,u_n,\ ,u_n + c,\ u_n - c$$

For wall boundary only one characteristic enters the domain i.e $u_n + c$, because normal velocity is zero at a non-permeable wall. But for wall boundary no net mass or energy fluxes should transmitted across the wall.

So, the implicit operator is active at the wall in the normal direction. When the operator passes information away from the wall, incoming values of δW are set equal to zero. When the operator passes information toward the wall, the outgoing flux is mirrored about the wall plane and propagated back into the flow using the inward operator. Reflections are used at centerlines.

For example, at the YZ plane the predictor step boundary condition is $\delta W^{*}_{I,j,k} = 0$.

For the corrector step the computed end flux terms from predictor step $\delta W^{*}_{I,j,k}$ are saved to be used as a boundary condition for the corrector step that sweeps away from this boundary in the increasing j direction. According to the usual rules of reflection, the starting flux of the corrector step is given by

$$\delta W^{\overline{n+1}}_{I,j,k} = E\,\delta W^{n}_{I-1,j,k}$$

where

$$\mathrm{E} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Similarly the boundary conditions can be found for XZ and XY planes.

For XZ plane the value of E is

$$\mathrm{E} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

For XY plane the value of E is

$$\mathrm{E} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

5.2.4 Symmetry Boundary

For symmetry boundary also only one characteristic enters the domain i.e $u_n + c$. Symmetry boundary condition can also be treated more approximately by taking

$$\delta W^{\overline{n+1}}_{I,j,k} = E\,\delta W^{n}_{I,j,k}$$

E is same as in the wall boundary condition.

5.3 Closure

In this chapter the boundary conditions described and have been implemented.

Chapter 6

Results and Discussion

In this section we will discuss the results for four different test cases for all the models selected in this work. The test cases are:

- Shocktube Problem
- Supersonic flow over a wedge
- Subsonic flow over a circular bump
- Subsonic flow over a airfoil
- Flow over re-entry capsule

The results are all compared with standard benchmark and/or analytical results. The explicit MacCormack scheme with artificial viscosity (specify as MacCormack), with TVD implementation and Implicit MacCormack scheme (specify as ImpMacCormack) have been tested. The aim is to study the performance of these schemes 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.

6.1 Shock-tube Problem

This problem ([5], (pg-352),) 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 [25]. 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 Δx= 0.1m . A Courant number of 1.1 has been used.

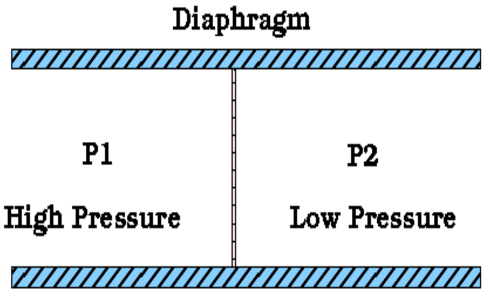


Figure 6.1: Shocktube

It contains two zones, first zone supports high pressure fluid and second zone supports low pressure fluid. The details of the geometry are:

- four slip walls
- two symmetric boundary surface

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

Boundary Conditions

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 [25]. 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. 6.1). The explicit and Implicit MacCormack schemes are compared. The advantage of using an implicit scheme compared to a explicit scheme is the computation time. The expansion shock occurring on the left has been captured accurately as in explicit one with less computation time. The results from Implicit MacCormack and Explicit MacCormack scheme (with artificial viscosity) are presented.

Results

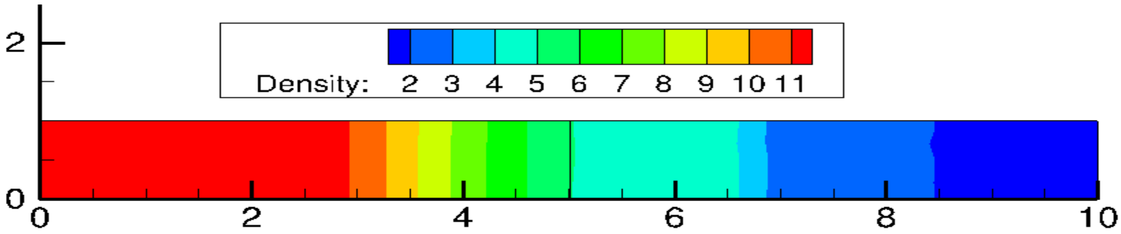


Figure 6.2: Density Contour at 6.1 ms with pressure ratio of 10 with constant Courant No = 1.1

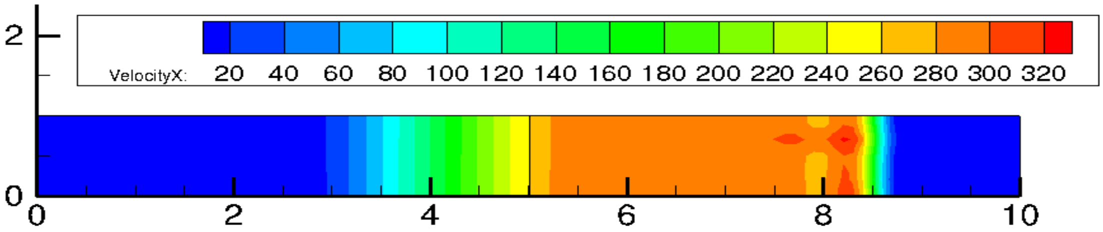


Figure 6.3: Velocity Contour at 6.1 ms with pressure ratio of 10 with constant Courant No = 1.1

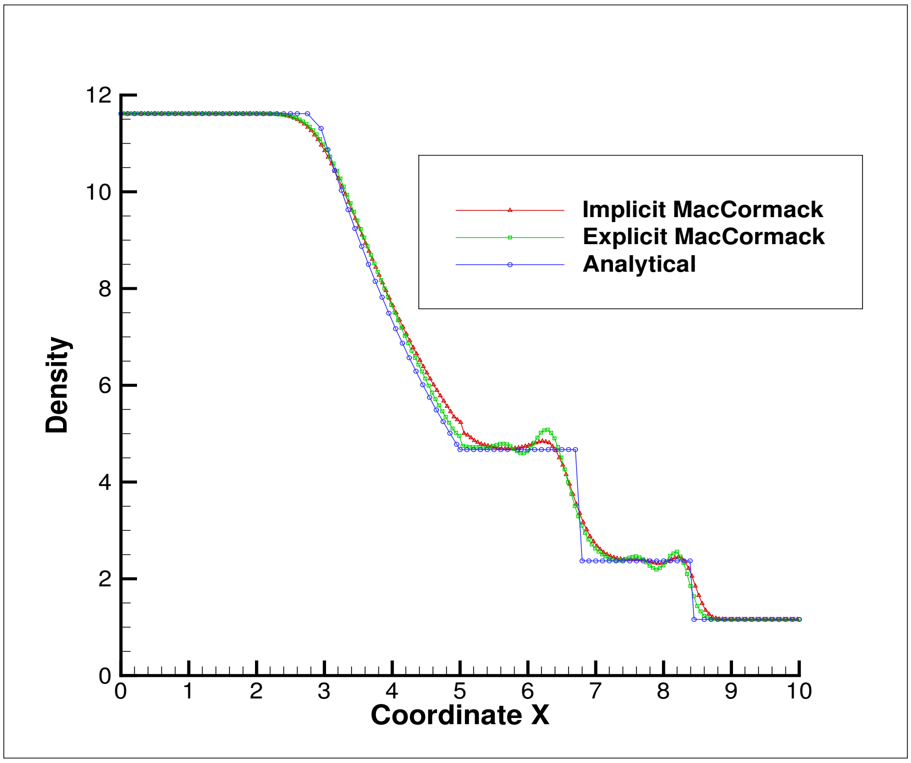


Figure 6.4: Density Plot at 6.1 ms

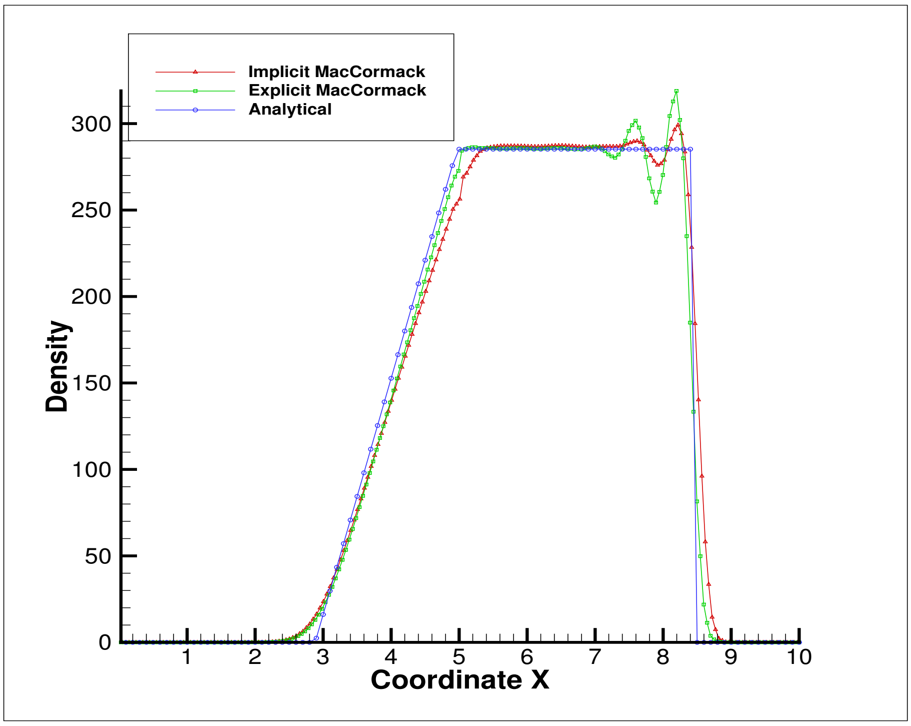


Figure 6.5: U Velocity Plot at 6.1 ms

Computational Time:

Method	CFL	Toatl CPU Time	Time Steps	Time step size
Explicit MacCormack	0.75	23.5 s	121	0.00005
Implicit MacCormack	1.1	16.28 s	83	0.0001

Table 6.1: Computational time comparison

6.2 Supersonic Flow Over a Wedge

We now consider the supersonic flow over a 2-D wedge with wedge angle 15° , as shown in Fig. 6.6. The inflow conditions are summarized in Table 6.2 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 [25]). Courant number of 0.3 and 1.1 are used for Explicit and Implicit MacCormack, respectively.

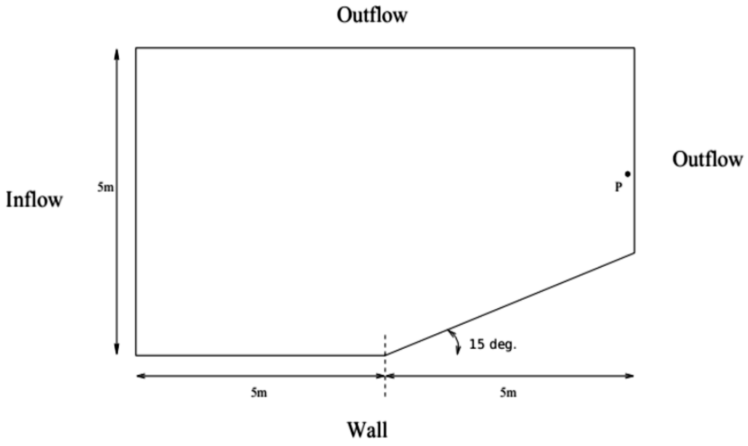


Figure 6.6: Computational domain

Boundary Conditions

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

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

Table 6.2: Boundary conditions for supersonic wedge

The steady-state contours of Mach number and static pressure using Implicit MacCormack Scheme shown in Fig. 6.7 and 6.9. Under the same flow condition the contours obtained by numerical computation done in Hirschs book [6] is also shown in Fig. 5.4. The results downstream of the shock has been tabulated in Table 6.3 where, P_2/P_1 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. Pressure, density, mach and temperature values are extracted along x = 1.2 line for both the schemes and plotted along Y axis. The plots are compared for explicit and implicit MacCormack schemes in figures 6.12, 6.2, 6.14, 6.15. Both the Explicit MacCorMack and Implicit MacCormack gives similar results and both have high accuracy, corresponding with the analytical results. Implicit MacCormack scheme however gives a solution within less computational time.

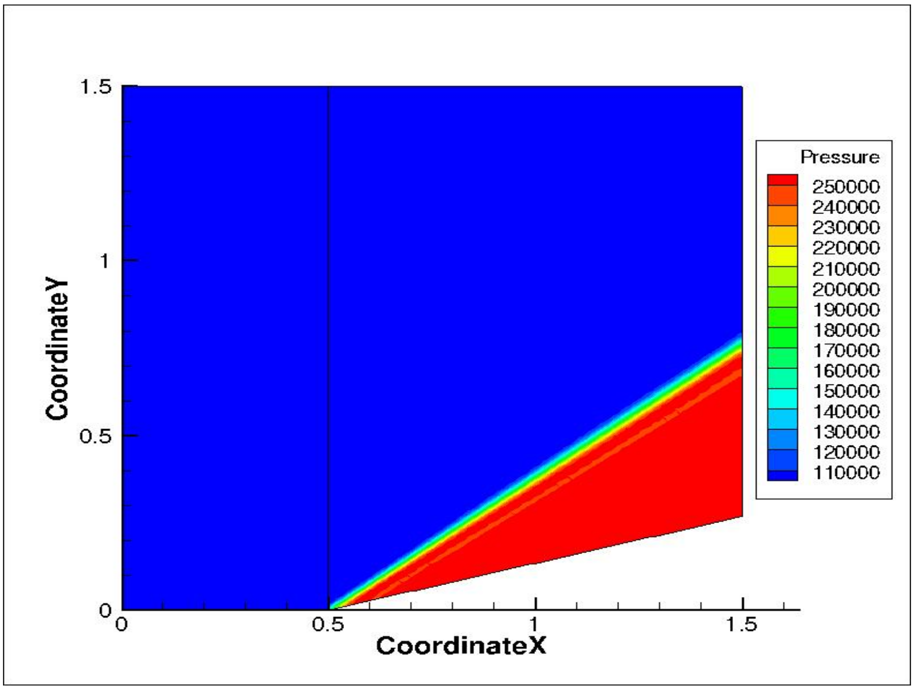


Figure 6.7: Pressure Contour with Implicit MacCormack Scheme

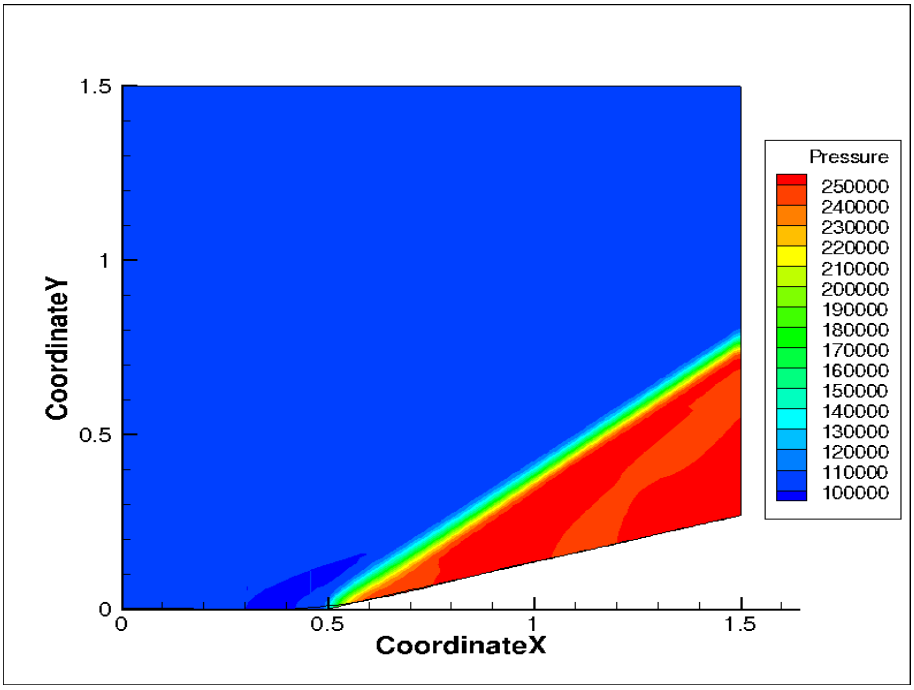


Figure 6.8: Pressure Contour with Explicit MacCormack Scheme

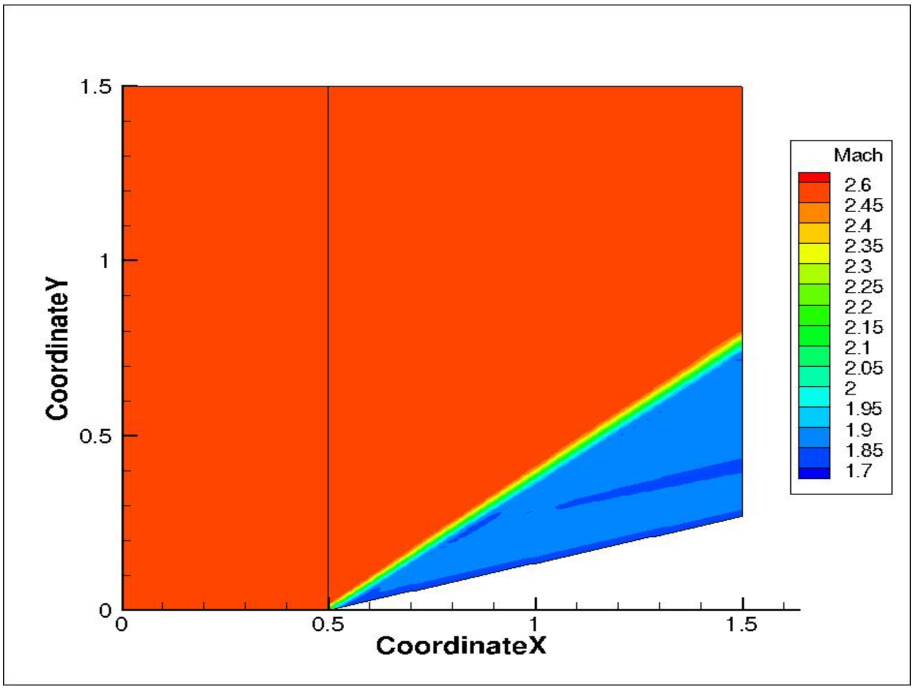


Figure 6.9: Mach Contour with Implicit MacCormack Scheme

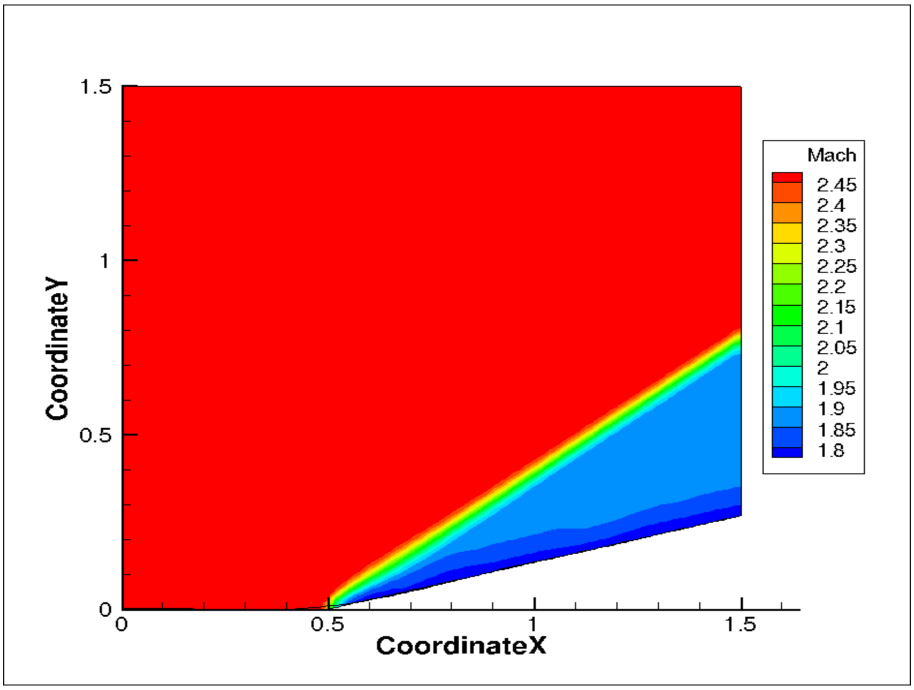


Figure 6.10: Mach Contour with Explicit MacCormack Scheme

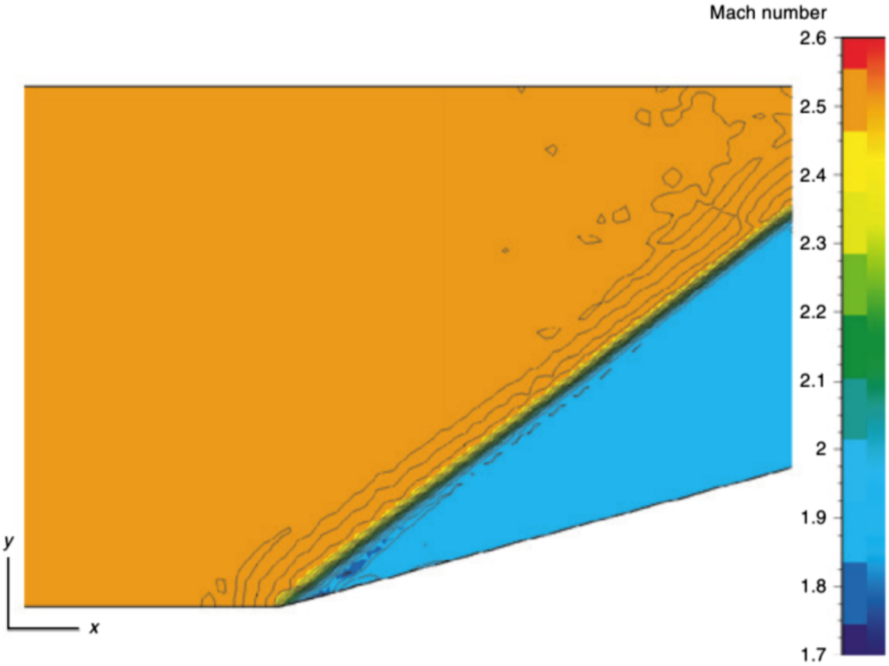


Figure 6.11: Mach Contour from Reference Hirsch [6]

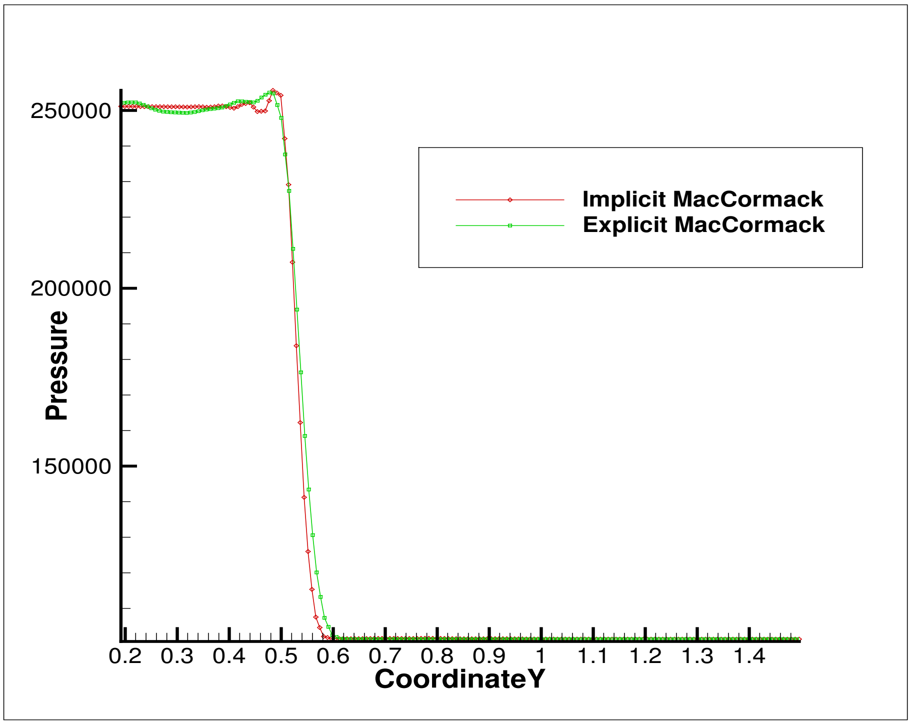


Figure 6.12: Variation of pressure along Y co-ordinate

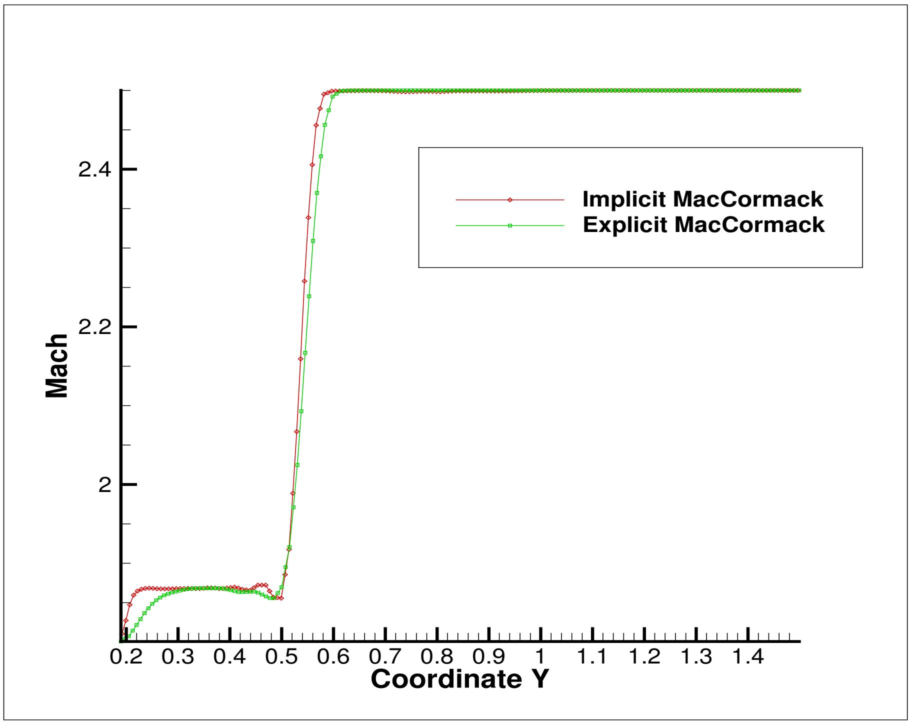


Figure 6.13: Variation of Mach Number along Y co-ordinate

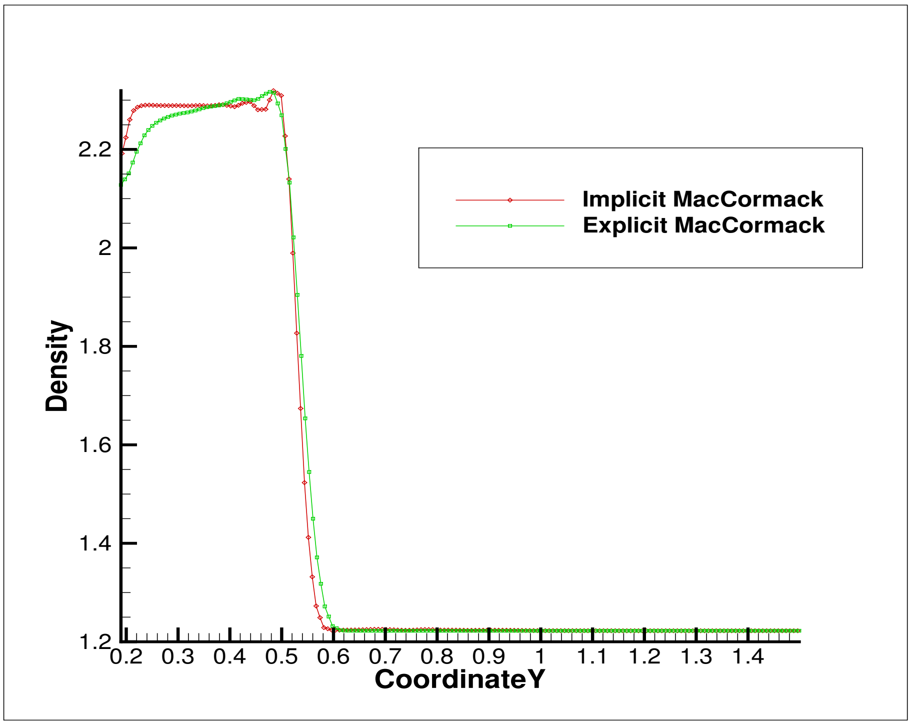


Figure 6.14: Variation of Density along Y co-ordinate

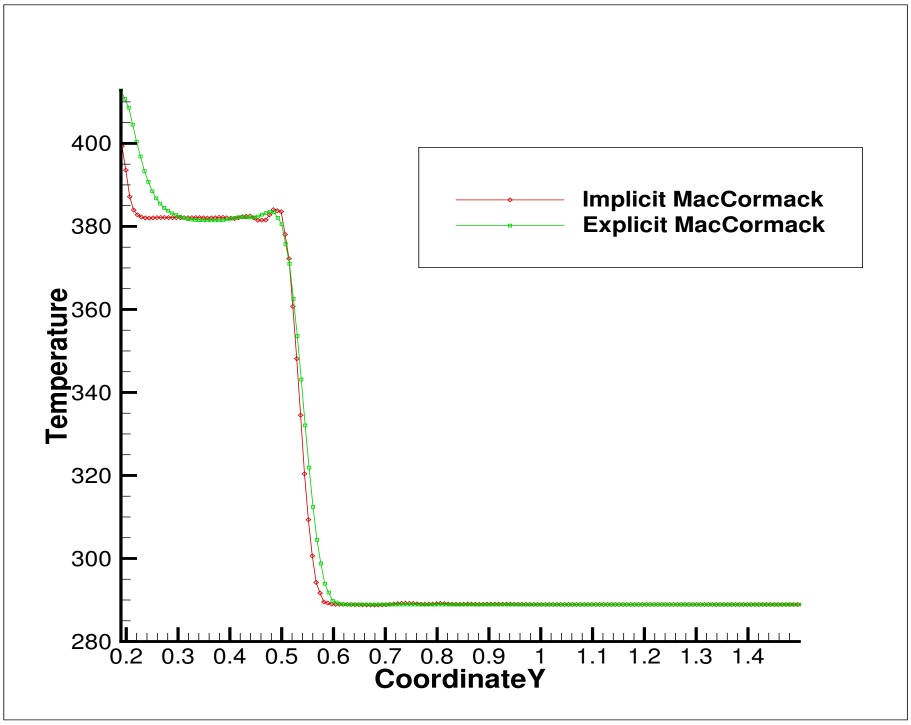


Figure 6.15: Variation of Temperature along Y co-ordinate

Validation:

Ratio's	Analytical	Implicit MacCormack	Explicit MacCormack
P_2/P_1	2.468	2.477	2.468
T_2/T_1	1.322	1.325	1.322
ρ_2/ρ_1	1.867	1.869	1.866
Mach	1.874	1.864	1.873
Shock angle(in degree)	36.945	37.954	38.66

Table 6.3: Validation with analytical solution

Computational Time:

Method	CFL	Toatl CPU Time	Time Steps
Explicit MacCormack	0.3	551274 s (154 hrs)	632573
Implicit MacCormack	1.1	997.3 s (17 min)	1075

Table 6.4: Computational time comparison

Once again the figures show the results of the Implicit scheme are close to that of the explicit scheme, where table 6.4 shows the Implicit scheme taken only $\frac{1}{500}$th the computational time.

6.3 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. 6.19. For the subsonic case 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 the numerical algorithm. We used a Courant number of 1.1 for cases below. The implicit MacCormack has convergence difficulty for the subsonic case if we want to use residual which is less than 10^{-8}. So we are using the convergence criteria upto 10^{-6}.

6.3.1 Subsonic Case

The inlet Mach number is chosen equal to 0.5. We provide total pressure and total temperature at inlet with respect to the static condition, so as to get inlet Mach Number equal to 0.5. This is as per our boundary condition discussion is Sec. 5.1.2 under pressure-driven flow section. At outflow we use the free-stream condition (static condition). The boundary conditions are summarized in Table

6.5 and the solver results have been compared with the study done by Rincon and Elder et al. [20].

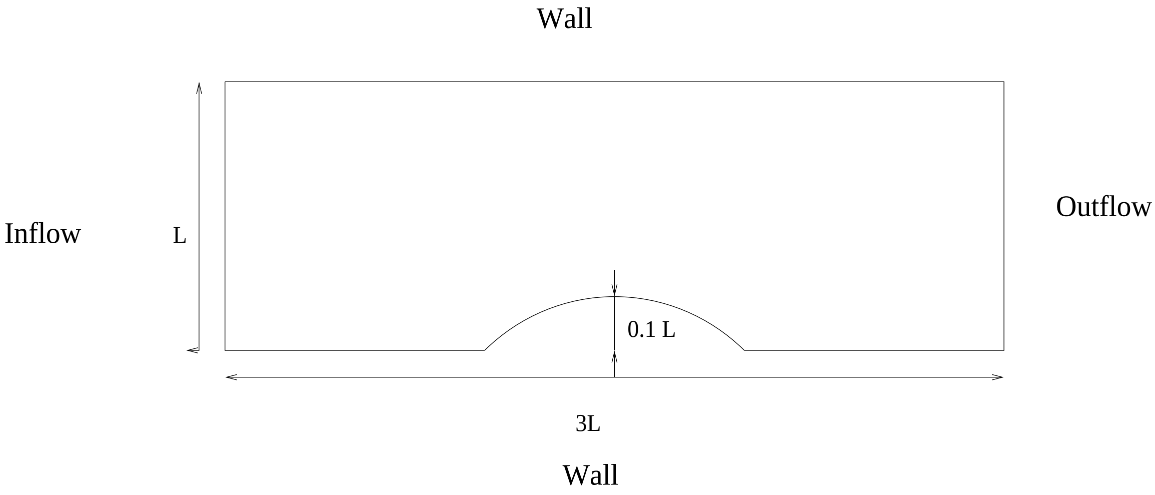


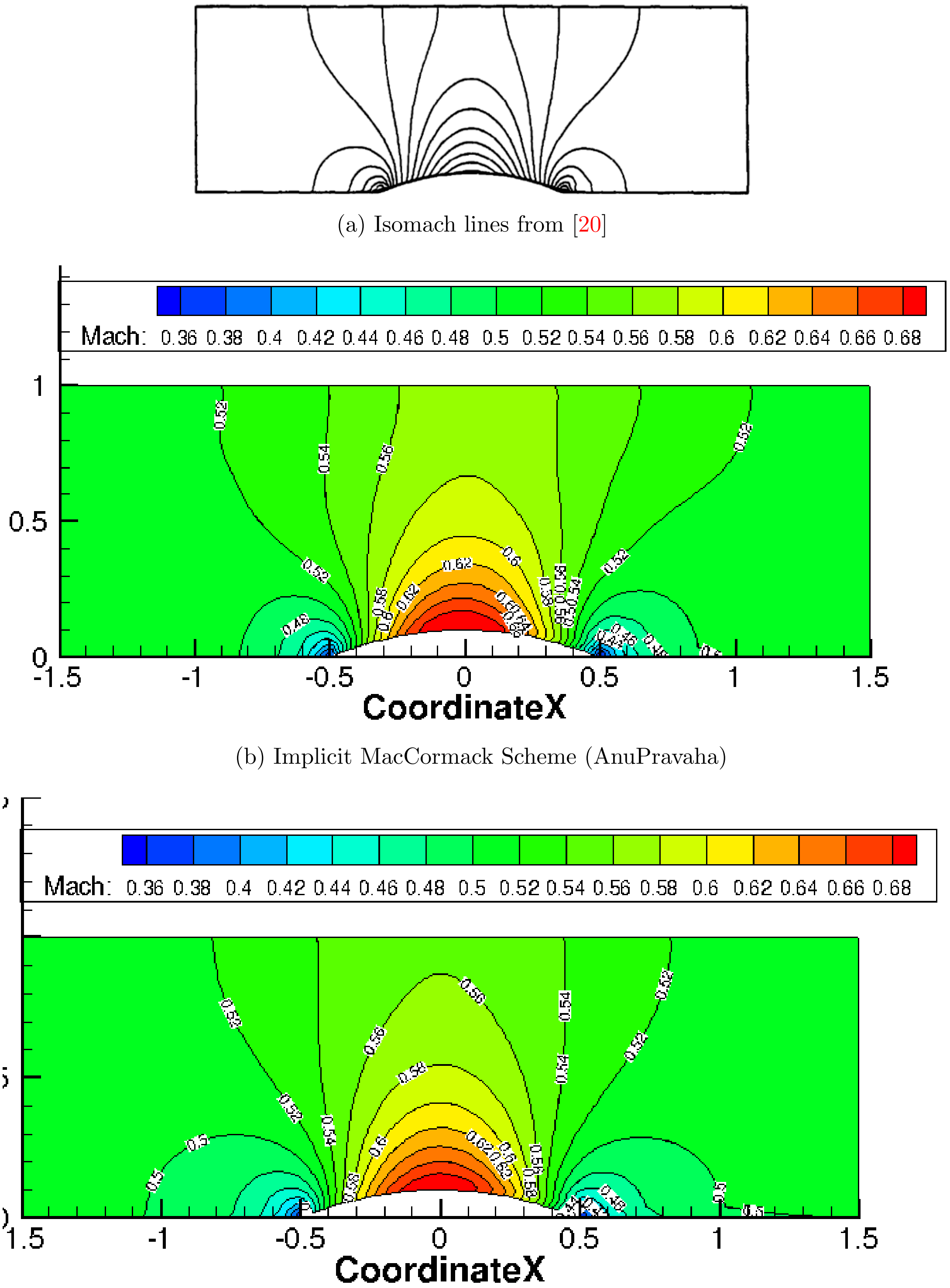
Figure 6.16: Computational domain [2]

Boundary Conditions

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

Table 6.5: Boundary Condition for subsonic bump

The comparison for Mach contours for Explicit MacCormack, Implicit MacCormack and the reference is shown in Fig. 6.17. Fig. 6.18 shows the variation in Mach number along the upper and lower walls. Comparison of computational times are shown in Table 6.6.



(a) Isomach lines from [20]

(b) Implicit MacCormack Scheme (AnuPravaha)

(c) Explicit MacCormack Scheme (AnuPravaha)

Figure 6.17: Mach Contour for M = 0.5

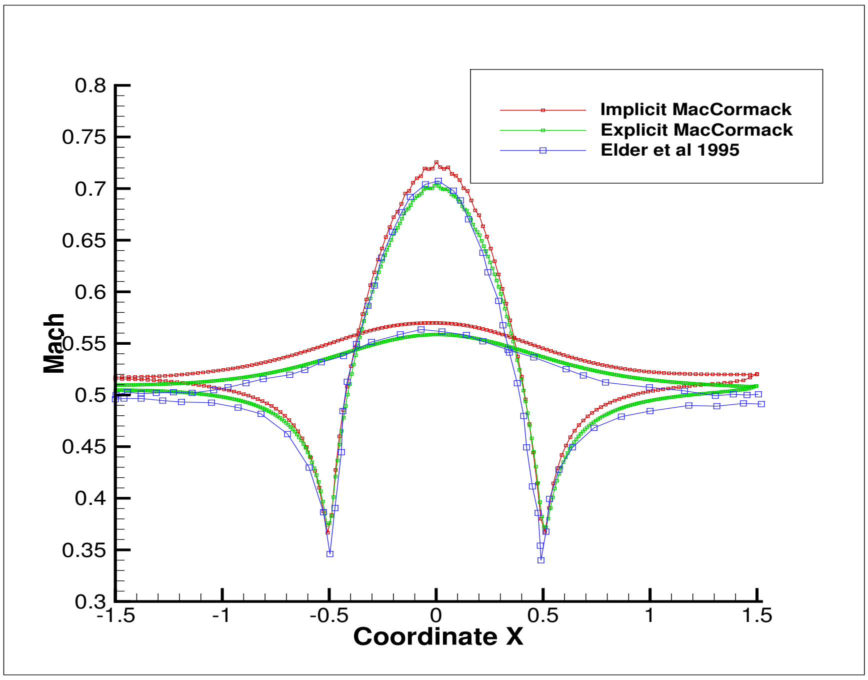


Figure 6.18: Variation of Mach number along lower and upper wall (M = 0.5)

Computational Time:

Method	CFL	Toatl Time	Time Steps	Residual
Explicit MacCormack	0.4	313308 s (87.03 hrs)	396418	$3.3368e^{-6}$
Implicit MacCormack	1.1	16750.75 s (4.6 hrs)	31518	$3.3368e^{-6}$

Table 6.6: Computational time comparison

The figures show the results of the Implicit scheme for bump case are close to that of the explicit scheme, where Table 6.6 shows the Implicit scheme taken $\frac{1}{21}$th the computational time.

6.4 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. The computational domain for the NACA Aerofoil considered is shown below.

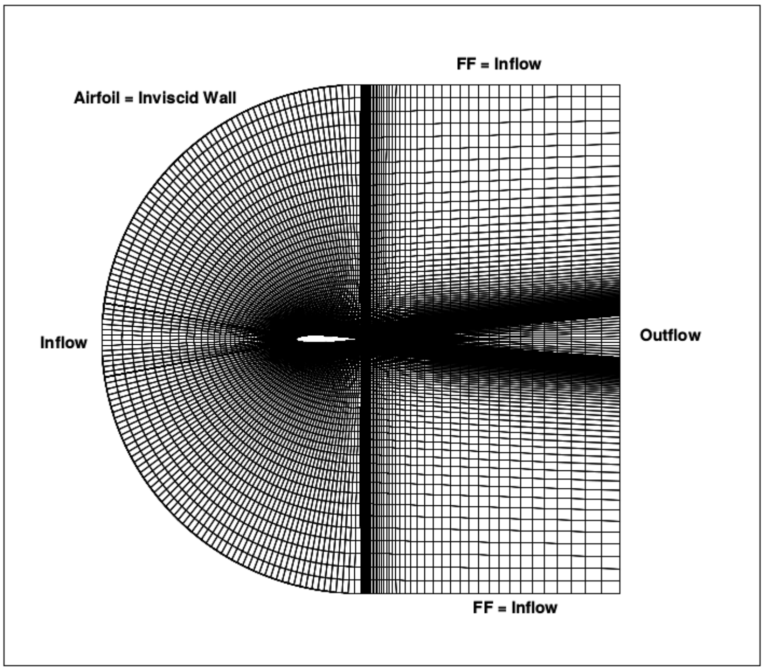


Figure 6.19: Computational domain [2]

6.4.1 Subsonic Case:Mach 0.5, Angle of Attack ($\alpha = 0^\circ$)

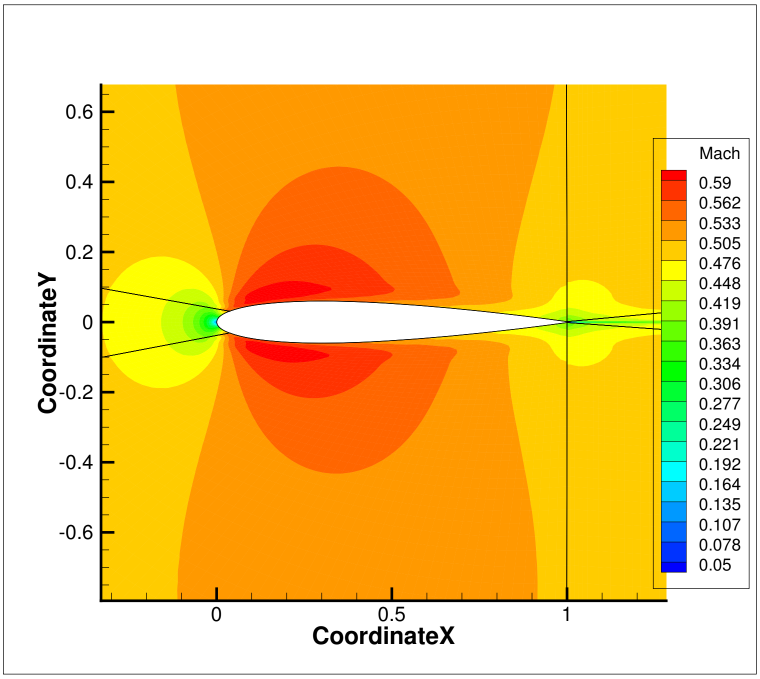
This is a subsonic case involving external flow. We have used velocity-driven boundary condition for inlet and far-field. The boundary conditions are tabulated in Table 6.7

Boundary Conditions

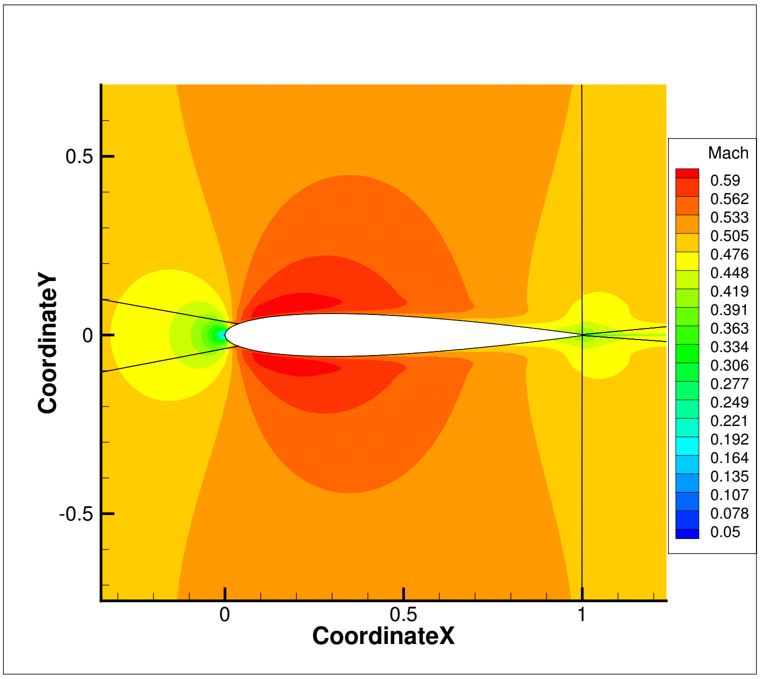
Quantity	Inflow	Outflow
Pressure	100000 Pa	100000 Pa
Temperature	300 K	288 K
U velocity	173.594	-
V velocity	0	-
W velocity	0	-

Table 6.7: Boundary conditions for NACA 0012 M = 0.5, $\alpha = 0^\circ$

The comparison for Mach and pressure contours for Explicit MacCormack, Implicit MacCormack and the reference is shown in Fig. 6.20 and 6.21 respectively.

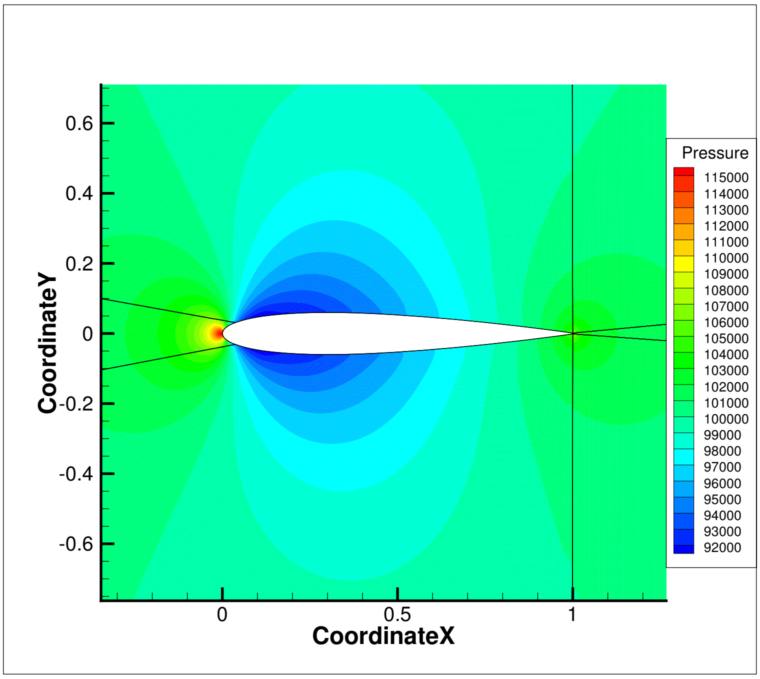


(a) Implicit MacCormack

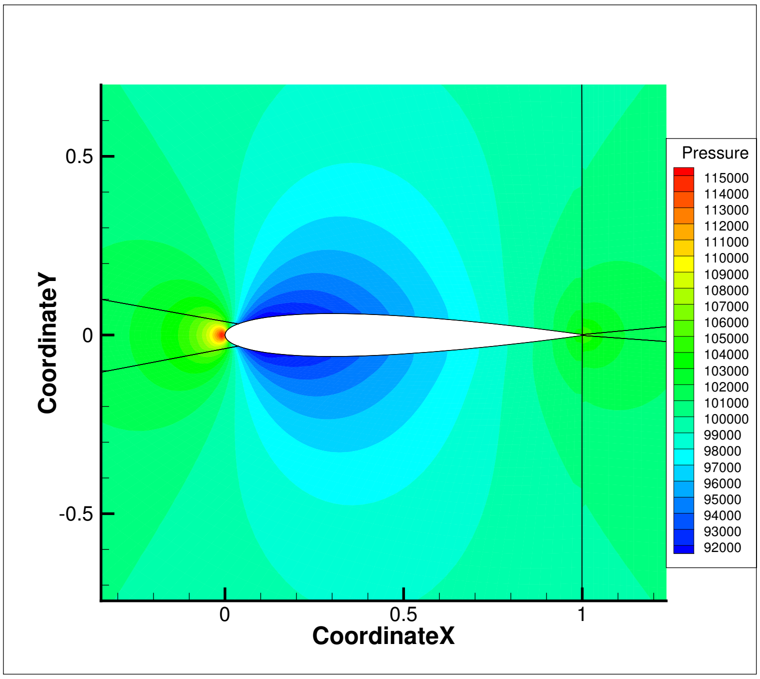


(b) Explicit MacCormack

Figure 6.20: Mach Contour for M=0.5



(a) Implicit MacCormack



(b) Explicit MacCormack

Figure 6.21: Pressure Contour for M=0.5

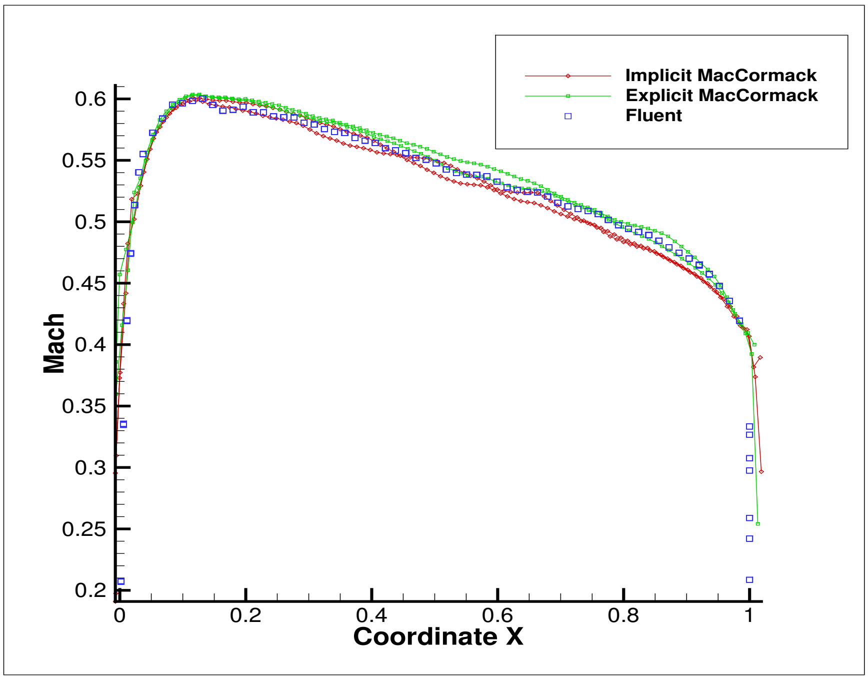


Figure 6.22: Variation of Mach number along airfoil wall (M = 0.5)

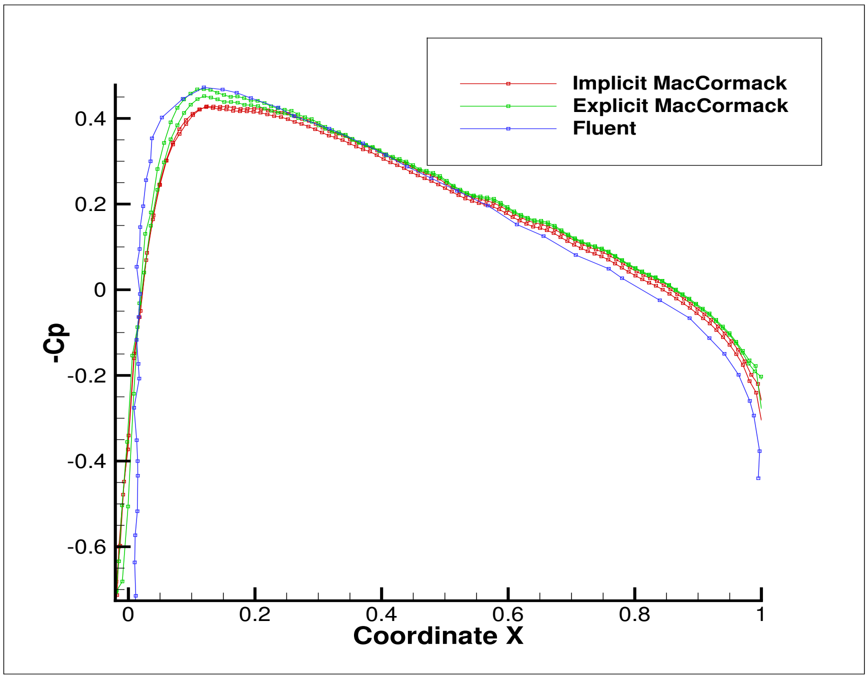


Figure 6.23: Variation of coefficient of pressure along airfoil wall (M = 0.5)

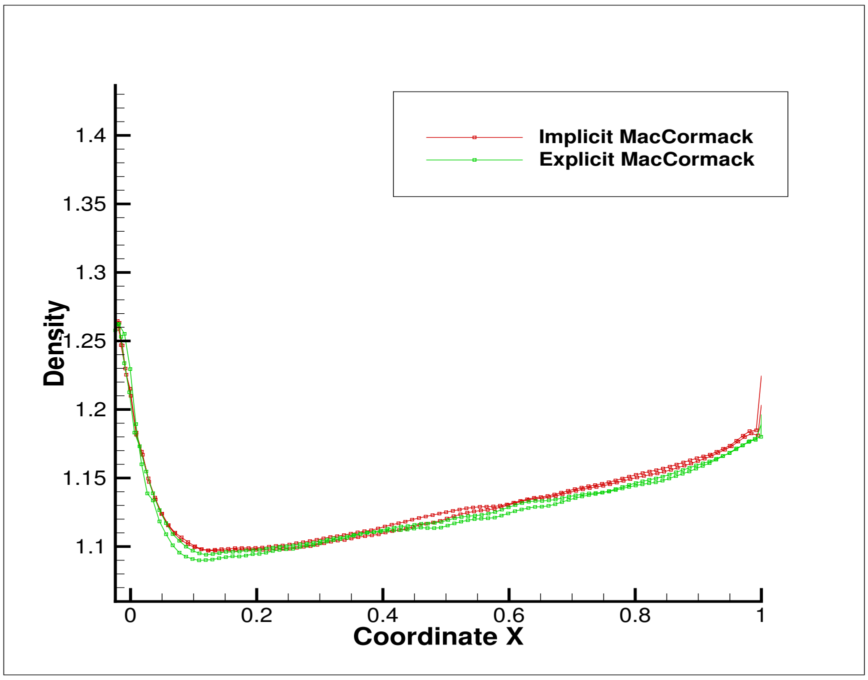


Figure 6.24: Variation of density number along airfoil wall (M = 0.5)

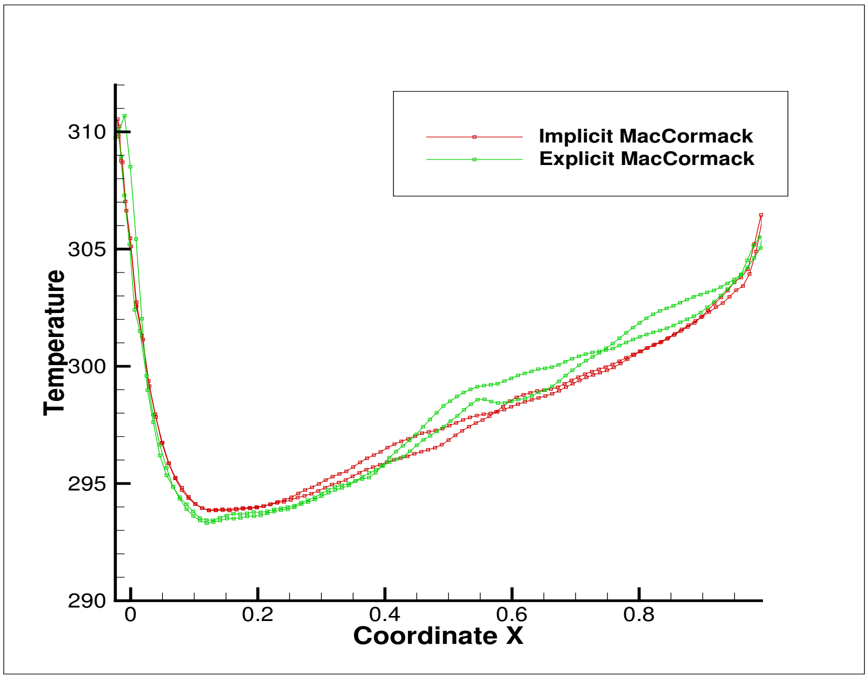


Figure 6.25: Variation of temperature number along airfoil wall (M = 0.5)

Computational Time:

Method	CFL	Toatl Time	Time Steps
Explicit MacCormack	0.3	413815 s (114.9 hrs)	597819
Implicit MacCormack	1.1	21522.67 s (5.7 hrs)	41728

Table 6.8: Computational time comparison

The figures show the results of the Implicit scheme for bump case are close to that of the explicit scheme, where table 6.6 shows the Implicit scheme taken $\frac{1}{20}$th the computational time.

6.5 3D Case: Flow over Re-entry Capsule

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. 6.26. The mesh and the computational domain are shown in Fig 6.27. 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 6.9. We validate the result with the study done by [30]. In this study, the wind tunnel data [28] has been used for validation. We have also compare the two MacCormak scheme results.

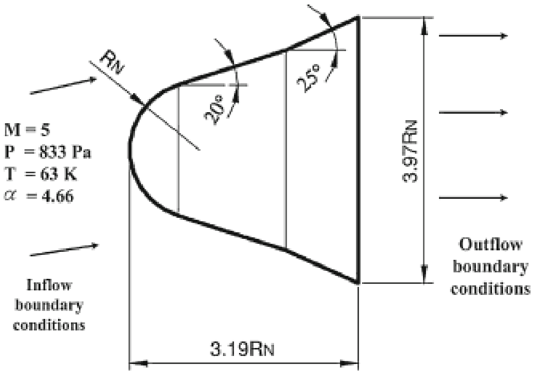


Figure 6.26: Re-entry vehicle model dimensions

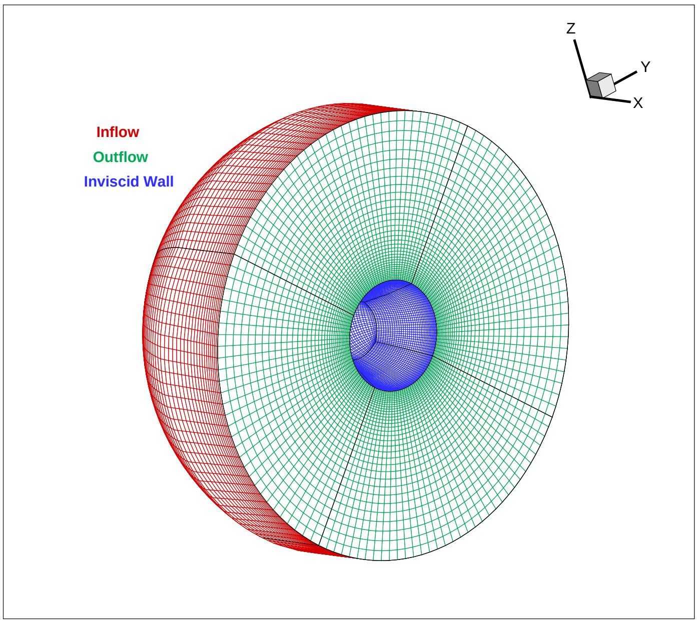


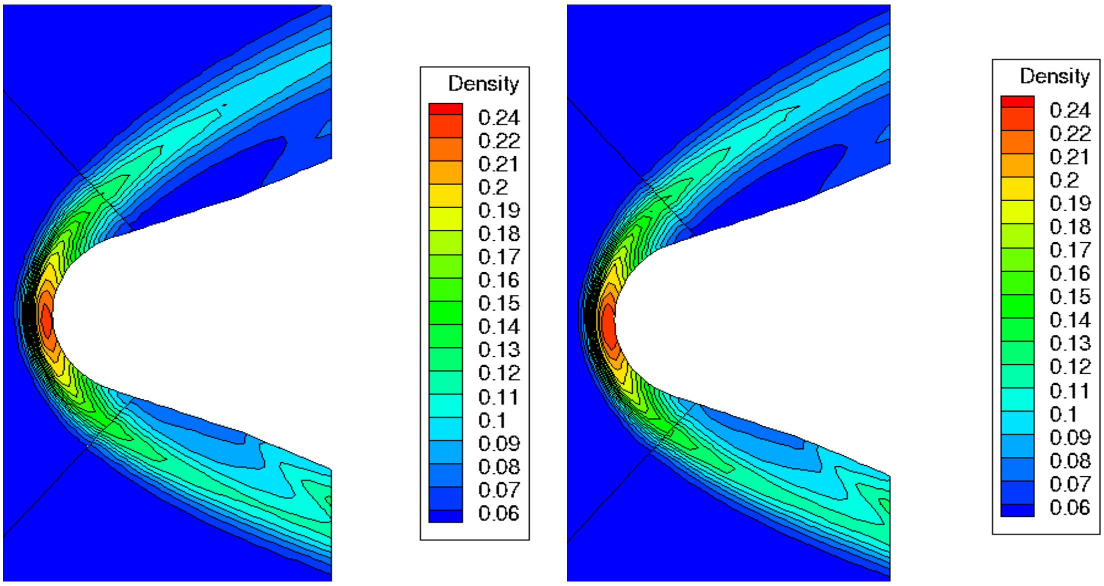
Figure 6.27: Computational domain and mesh [2]

Boundary Conditions

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

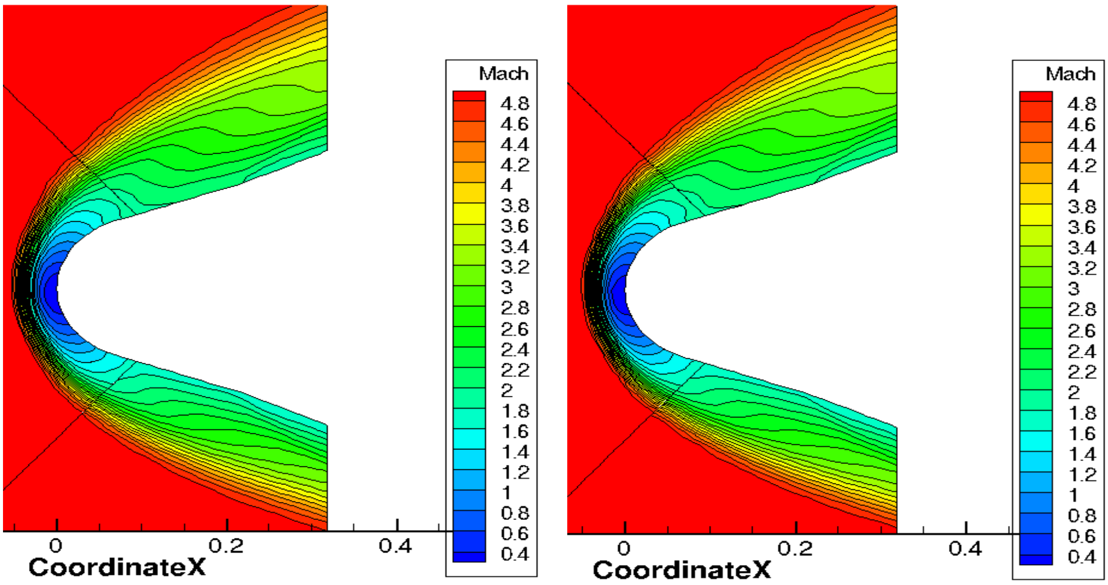
Table 6.9: Boundary Condition

Figs. 6.33 and 6.29 show the density contours and mach countours respectively. We can see from the contours the presence of bow shock. On the windward side the formation of second shock is more pronounced in comparison to leeward side. The plot of Cp distribution along the capsule wall is shown in Fig. 6.30. 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 along the surface of capsule till the second shock. The plot of density, mach and distribution along the capsule wall is shown in figures with respect to x-axis. The two MacCormack schemes are compared in these plots.



(a) (b)

Figure 6.28: Density Contours using (a) Implicit MacCormack (b) Explicit MacCormack



(a) (b)

Figure 6.29: Mach Contours using (a) Implicit MacCormack (b) Explicit MacCormack

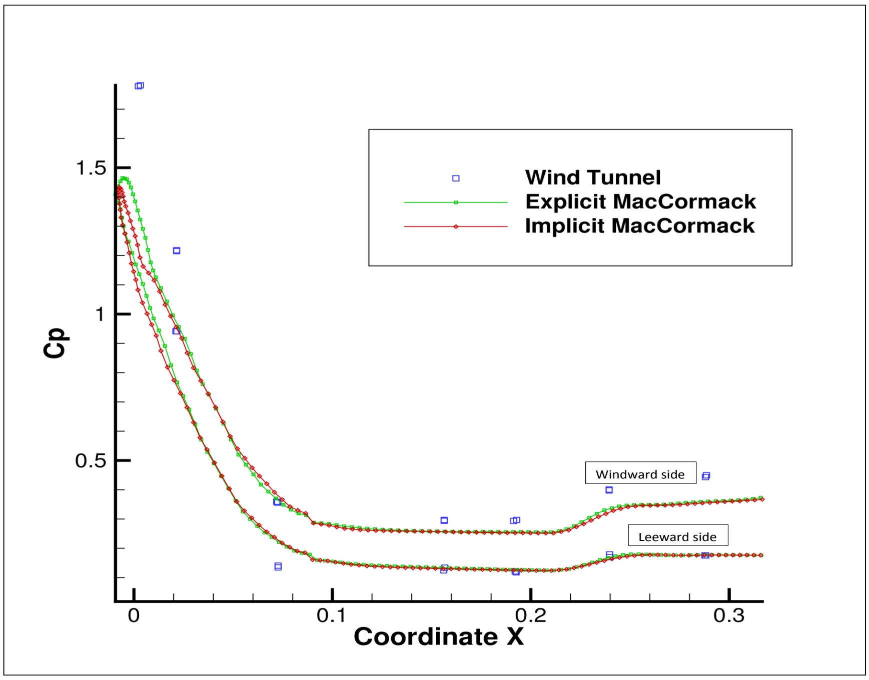


Figure 6.30: Variation of coefficient of pressure along the capsule wall

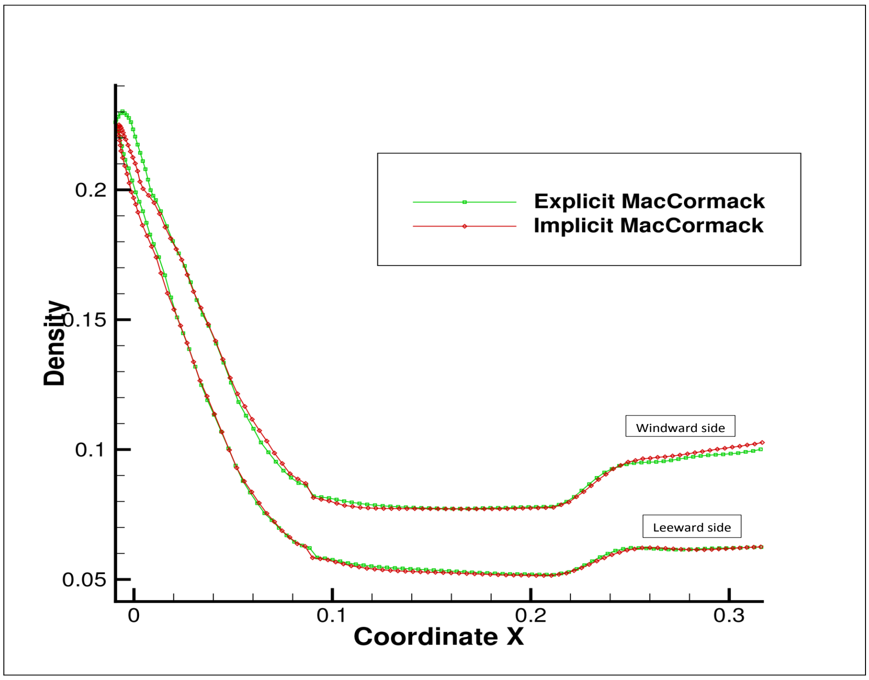


Figure 6.31: Variation of density along the capsule wall

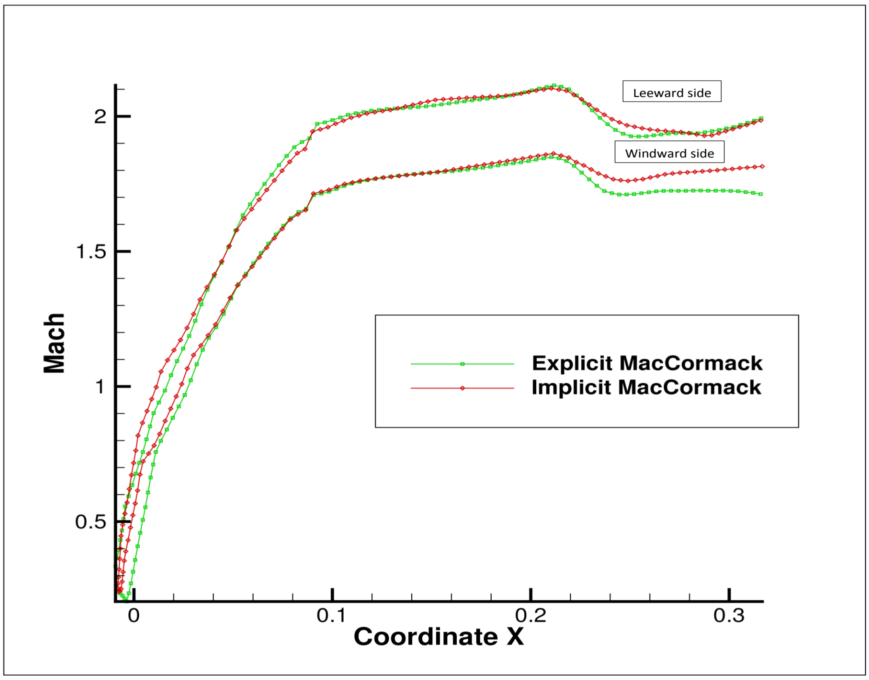


Figure 6.32: Variation of mach number along the capsule wall

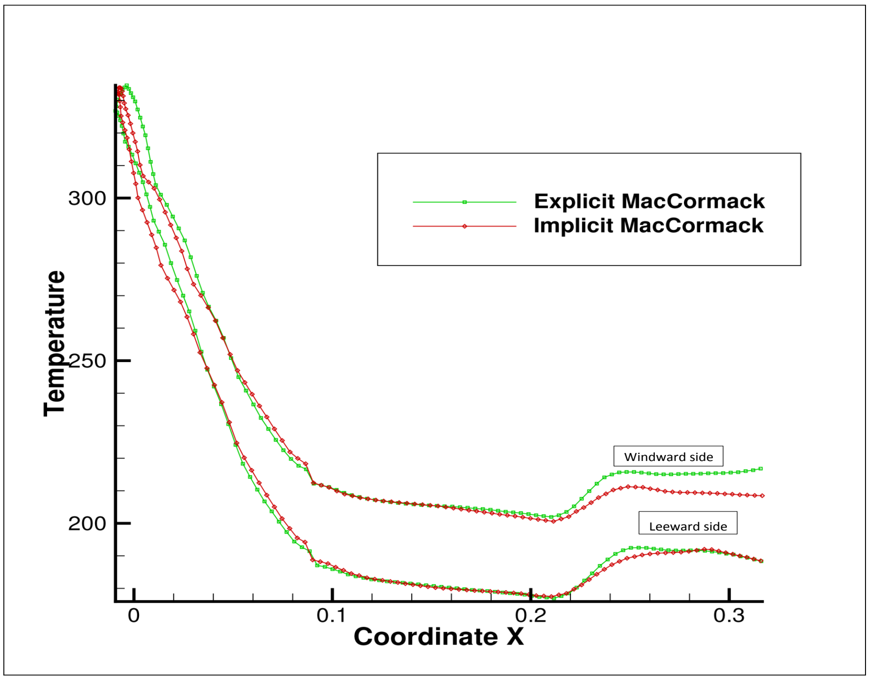


Figure 6.33: Variation of temperature along the capsule wall

Computational Time:

Method	CFL	Toatl Time	Time Steps	End Time
Explicit MacCormack	0.4	313408 s (87 hrs)	396418	1 s
Implicit MacCormack	1.1	10038.25 s (2.8 hrs)	26617	0.003934 s

Table 6.10: Computational time comparison

The results show that the implicit scheme gives solution that are accurate as those of the explicit scheme, while taking just $\frac{1}{30}$th of the computational time.

Chapter 7

Conclusions and future work

Building on the earlier work of Nikhil Kalkote [21] and Ashwani Assam [2], we have created a stand-alone version of the AnuPravaha Solver for computational flows using implicit schemes. An compressible flow module was created for the general-purpose CFD solver ANUPRAVHA, which uses the Implicit MacCormack scheme with artificial viscosity in finite-volume form to solve the Euler system (continuity, momentum and energy) of equations on a structured non-orthogonal multi-block grid.

The above method is unconditionally stable, and is second order accurate in both space and time.

In addition to this, the following features of this scheme should be pointed out.

a) For regions of the flow satisfying explicit stability criteria, the implicit method reduces to the corresponding explicit method and therefore no more computing time than the explicit scheme is needed in these regions. Due to this feature, the implicit MacCormack scheme is also called explicit-implicit or hybrid in some literature.

b) Viscous effects are included in the implicit operator in an approximate and very simple way to enhance the stability for viscous flows. Therefore the computation of the implicit operator and its inversion can be done with the help of the knowledge of the inviscid Jacobians. Two block bidiagonal matrix inversions are reduced to two scalar bidiagonal matrix inversions, a fact which greatly reduces the computation.

c) Although the scheme is unconditionally stable in von Neumann's sense, $\triangle t$ is still limited in practical computation, which is considered to be mainly due to the error created by approximate factorization taken in the procedure and the approximate linearization.

d) An intrinsic property of the two- step MacCormack type schemes, explicit or implicit, is the time step dependence of the steady state solution. Thus, convergent steady state solutions may only be reliable with sufficiently small $\triangle t$. Therefore one measure to achieve spatial accuracy is to reduce time step towards the end of the marching until variation of the solution with this reduction diminishes. This is obviously a disadvantage of the scheme for steady state solutions.

Future work can be in the direction of

1. Further validation of the present code can be made for more complex geometries and for subsonic and transonic flow regimes.

2. Naiver-Stokes version of compressible flow with Turbulence models should be implemented.

Appendix A

Jacobian Matrix Formulation

The inviscid Jacobians $|A|$, $|B|$ and $|C|$ can be diagonized by Sx, Sy and Sz respectively. i.e.

$$A \;=\; S_x^{-1} D_A S_x \qquad B \;=\; S_y^{-1} D_B S_y \qquad C \;=\; S_z^{-1} D_C S_z$$

$$S_x \;=\; \begin{pmatrix} 1 & 0 & 0 & 0 & \frac{-1}{c^2} \\ 0 & \rho c & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & -\rho c & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ \frac{-u}{\rho} & \frac{1}{\rho} & 0 & 0 & 0 \\ \frac{-v}{\rho} & 0 & \frac{1}{\rho} & 0 & 0 \\ \frac{-w}{\rho} & 0 & \frac{1}{\rho} & 0 & 0 \\ \alpha\beta & -u\beta & -v\beta & -w\beta & \beta \end{pmatrix}$$

We use Mathematica to calculate the multiplication and inverse of the matricies.

So,

$$S_x \;=\; \begin{bmatrix} \left(1 - \frac{\alpha\beta}{c^2}\right) & \frac{u\beta}{c^2} & \frac{v\beta}{c^2} & \frac{w\beta}{c^2} & \frac{-\beta}{c^2} \\ -uc + \alpha\beta & c - u\beta & -v\beta & -w\beta & \beta \\ \frac{-v}{\rho} & 0 & \frac{1}{\rho} & 0 & 0 \\ \frac{-w}{\rho} & 0 & 0 & \frac{1}{\rho} & 0 \\ uc + \alpha\beta & -c - u\beta & -v\beta & -w\beta & \beta \end{bmatrix}$$

and

$$S_x^{-1} \;=\; \begin{bmatrix} 1 & \frac{1}{2c^2} & 0 & 0 & \frac{1}{2c^2} \\ u & \frac{c+u}{2c^2} & 0 & 0 & \frac{u-c}{2c^2} \\ v & \frac{v}{2c^2} & 0 & 0 & \frac{v}{2c^2} \\ w & \frac{w}{2c^2} & 0 & \rho & \frac{w}{2c^2} \\ a_{51} & a_{52} & v\rho & w\rho & a_{55} \end{bmatrix}$$

where

$$\begin{aligned}
a_{51} &= u^2 \,+\, v^2 \,+\, w^2 \,-\alpha, \\
a_{52} &= \frac{1}{2\beta} + \frac{u}{2c} + \frac{u^2 \,+\, v^2 \,+\, w^2 \,-\, \alpha}{2c^2} \\
a_{55} &= \frac{1}{2\beta} - \frac{u}{2c} + \frac{u^2 \,+\, v^2 \,+\, w^2 \,-\, \alpha}{2c^2}
\end{aligned}$$

$$S_y = \begin{pmatrix} 1 & 0 & 0 & 0 & \frac{-1}{c^2} \\ 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & \rho c & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & -\rho c & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ \frac{-u}{\rho} & \frac{1}{\rho} & 0 & 0 & 0 \\ \frac{-v}{\rho} & 0 & \frac{1}{\rho} & 0 & 0 \\ \frac{-w}{\rho} & 0 & \frac{1}{\rho} & 0 & 0 \\ \alpha\beta & -u\beta & -v\beta & -w\beta & \beta \end{pmatrix}$$

$$= \begin{bmatrix} \left(1 - \frac{\alpha\beta}{c^2}\right) & \frac{u\beta}{c^2} & \frac{v\beta}{c^2} & \frac{w\beta}{c^2} & \frac{-\beta}{c^2} \\ \frac{-u}{\rho} & \frac{1}{\rho} & 0 & 0 & 0 \\ -vc + \alpha\beta & -u\beta & c - v\beta & -w\beta & \beta \\ \frac{-w}{\rho} & 0 & 0 & \frac{1}{\rho} & 0 \\ vc + \alpha\beta & -u\beta & -c - v\beta & -w\beta & \beta \end{bmatrix}$$

and

$$S_y^{-1} = \begin{bmatrix} 1 & 0 & \frac{1}{2c^2} & 0 & \frac{1}{2c^2} \\ u & \rho & \frac{u}{2c^2} & 0 & \frac{u}{2c^2} \\ v & 0 & \frac{v+c}{2c^2} & 0 & \frac{v-c}{2c^2} \\ w & 0 & \frac{w}{2c^2} & \rho & \frac{w}{2c^2} \\ a_{51} & u\rho & a_{53} & w\rho & a_{55} \end{bmatrix}$$

where

$$\begin{aligned}
a_{51} &= u^2 \,+\, v^2 \,+\, w^2 \,-\alpha, \\
a_{52} &= \frac{1}{2\beta} + \frac{v}{2c} + \frac{u^2 \,+\, v^2 \,+\, w^2 \,-\, \alpha}{2c^2} \\
a_{55} &= \frac{1}{2\beta} - \frac{v}{2c} + \frac{u^2 \,+\, v^2 \,+\, w^2 \,-\, \alpha}{2c^2}
\end{aligned}$$

$$S_z = \begin{pmatrix} 1 & 0 & 0 & 0 & \frac{-1}{c^2} \\ 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & \rho c & 0 \\ 0 & 0 & 0 & -\rho c & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ \frac{-u}{\rho} & \frac{1}{\rho} & 0 & 0 & 0 \\ \frac{-v}{\rho} & 0 & \frac{1}{\rho} & 0 & 0 \\ \frac{-w}{\rho} & 0 & \frac{1}{\rho} & 0 & 0 \\ \alpha\beta & -u\beta & -v\beta & -w\beta & \beta \end{pmatrix}$$

$$= \begin{bmatrix} \left(1 - \frac{\alpha\beta}{c^2}\right) & \frac{u\beta}{c^2} & \frac{v\beta}{c^2} & \frac{w\beta}{c^2} & \frac{-\beta}{c^2} \\ \frac{-u}{\rho} & \frac{1}{\rho} & 0 & 0 & 0 \\ \frac{-v}{\rho} & 0 & \frac{1}{\rho} & 0 & 0 \\ -wc + \alpha\beta & -u\beta & -v\beta & c - w\beta & \beta \\ vc + \alpha\beta & -u\beta & -v\beta & -c - w\beta & \beta \end{bmatrix}$$

and

$$S_z^{-1} = \begin{bmatrix} 1 & 0 & 0 & \frac{1}{2c^2} & \frac{1}{2c^2} \\ u & \rho & 0 & \frac{u}{2c^2} & \frac{u}{2c^2} \\ v & 0 & \rho & \frac{v}{2c^2} & \frac{v}{2c^2} \\ w & 0 & 0 & \frac{w+c}{2c^2} & \frac{w-c}{2c^2} \\ a_{51} & u\rho & v\rho & a_{54} & a_{55} \end{bmatrix}$$

where

$$\begin{aligned} a_{51} &= u^2 + v^2 + w^2 - \alpha, \\ a_{52} &= \frac{1}{2\beta} + \frac{w}{2c} + \frac{u^2 + v^2 + w^2 - \alpha}{2c^2} \\ a_{55} &= \frac{1}{2\beta} - \frac{w}{2c} + \frac{u^2 + v^2 + w^2 - \alpha}{2c^2} \end{aligned}$$

Appendix B

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.

$$\frac{\partial U}{\partial t} = -\frac{\partial U}{\partial x} - \frac{G}{y} + J \tag{B.1}$$

where U is the solution vector, $U = \begin{bmatrix} \rho & \rho u & \rho v & \rho(e + V^2/2) \end{bmatrix}$.

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

$$\begin{aligned} S^t_{i,j} = {} & C_x \frac{\left|p^t_{i+1,j} - 2p^t_{i,j} + p^t_{i-1,j}\right|}{p^t_{i+1,j} - 2p^t_{i,j} + p^t_{i-1,j}} (U^t_{i+1,j} - 2U^t_{i,j} + U^t_{i-1,j}) \\ & + C_y \frac{\left|p^t_{i,j+1} - 2p^t_{i,j} + p^t_{i,j-1}\right|}{p^t_{i,j+1} - 2p^t_{i,j} + p^t_{i,j-1}} (U^t_{i,j+1} - 2U^t_{i,j} + U^t_{i,j-1}) \end{aligned} \tag{B.2}$$

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

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

$$\begin{aligned} \bar{S}^t_{i,j} = {} & C_x \frac{\left|\bar{p}^t_{i+1,j} - 2\bar{p}^t_{i,j} + \bar{p}^t_{i-1,j}\right|}{\bar{p}^t_{i+1,j} - 2\bar{p}^t_{i,j} + \bar{p}^t_{i-1,j}} (\bar{U}^t_{i+1,j} - 2\bar{U}^t_{i,j} + \bar{U}^t_{i-1,j}) \\ & + C_y \frac{\left|\bar{p}^t_{i,j+1} - 2\bar{p}^t_{i,j} + \bar{p}^t_{i,j-1}\right|}{\bar{p}^t_{i,j+1} - 2\bar{p}^t_{i,j} + p^t_{i,j-1}} (\bar{U}^t_{i,j+1} - 2\bar{U}^t_{i,j} + \bar{U}^t_{i,j-1}) \end{aligned} \tag{B.3}$$

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

The value of $S^t_{i,j}$ and $\bar{S}^t_{i,j}$ 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,

$$\bar{\rho}^{t+\Delta t}_{i,j} = \rho^t_{i,j} + \left(\frac{\partial \rho}{\partial t}\right)^t_{i,j} \Delta t + S^t_{i,j} \tag{B.4}$$

On the corrector step,

$$\rho^{t+\Delta t}_{i,j} = \rho^t_{i,j} + \left(\frac{\partial \rho}{\partial t}\right)_a v\Delta t + \bar{S}^{t+\Delta t}_{i,j} \tag{B.5}$$

References

[1] Amit Shivaji Dighe, "Numerical Computation of Compressible Fluid Flows", Thesis for the Degree of Master of Technology, *Department of Mechanical Engineering*, Indian Institute of Technology Hyderabad, 2012.

[2] Ashwani Assam, "Development of a General-Purpose Compressible Flow AnuPravaha Based Solver", Thesis for the Degree of Master of Technology, *Department of Mechanical Engineering*, Indian Institute of Technology Hyderabad, 2014.

[3] Axel Rohde, "Eigenvalues and Eigenvectors of the Euler Equations in General Geometries", *AIAA journal*, Vol. 20, number 9 pp. 2001-2609.

[4] B. Engquist and A. Majda, "Radiation boundary conditions for acoustic and elastic wave calculations.", *Communications on Pure and Applied Mathematics 32*, pp- 313357, 1979.

[5] C. B. Laney, " Computational gasdynamics", *Univ. Press, Cambridge u.a*, 1998

[6] C. Hirsch., "Numerical computation of internal and external flows fundamentals of computational fluid dynamics.", *Elsevier, Amsterdam*,2007.

[7] D. H. Rudy and J. C. Strikwerda, "A nonreflecting outflow boundary condition for subsonic navier-stokes calculations", *Journal of Computational Physics 36*, pp- 5570. Cited by 0234, 1980.

[8] D. L. Whitfield and J. M. Janus, "Three-dimensional unsteady Euler equations solution using flux vector splitting", *00121*, 1984.

[9] Eswaran V. and Prakash S., “A finite volume method for Navier Stokes equations”, *Proc. Third Asian CFD Conference*,Vol. 1, pp. 127-136, Bangalore, India, July 1998.

[10] Eswaran V. et al., “Development of a General Purpose Robust CFD Solver”, Project Report No. 1, *Department of Mechanical Engineering*, Indian Institute of Technology Kanpur, December, 2005.

[11] G. Hedstrom, “Nonreflecting boundary conditions for nonlinear hyperbolic systems”, *Journal of Computational Physics 30*, pp- 222237, 1979.

[12] Gurris, Marcel and Kuzmin, Dmitri and Turek, Stefan, “Implicit finite element schemes for the stationary compressible Euler equations”, *International Journal for Numerical Methods in Fluids*, vol-69, No-1, pp=1-28, Wiley Online Library, 2012

[13] Harten Ami, “Numerical Solution of the Navier–Stokes Equations by Semi–Implicit Schemes”, *Journal of Computational Physics*, vol. 49, pp. 357393

[14] Hozman, J, “High resolution schemes for hyperbolic conservation laws”, *Charles University, Faculty of Mathematics and Physics, Prague, Czech Republic. Wds*, vol. 6, pp. 59-64

[15] Feistauer, Miloslav and CESENEK, JAN, “ON NUMERICAL SIMULATION OF AIRFOIL VIBRATIONS INDUCED BY COMPRESSIBLE FLOW”, Thesis for the Degree of Master of Technology, *Proceedings of ALGORITMY*, pp. 22-31, 2012.

[16] Fürst, J and Furmánek, P, “An implicit MacCormack scheme for unsteady flow calculations”, *Computers & Fluids*, Vol. 46,number 1, pp. 231–236, 2011.

[17] J. Blazek., “Computational fluid dynamics principles and applications.”, *Elsevier, Amsterdam; San Diego*, 2005.

[18] J. D. Anderson., “Computational fluid dynamics: the basics with applications”, *McGraw-Hill, Boston, Mass*, 2003.

[19] J. D. Anderson., "Modern compressible flow: with historical perspective", *McGrawHill, New York*, 1995.

[20] J. Rincon and R. Elder, "A high-resolution pressure-based method for compressible flows", *Computers & fluids 26*, pp-217231, 1997.

[21] Kalkote Nikhil Narayan, "Development of Compressible Flow Module In AnuPravaha Solver", Thesis for the Degree of Master of Technology, *Department of Mechanical Engineering*, Indian Institute of Technology Hyderabad, 2013.

[22] MacCormack, Robert William, "A numerical method for solving the equations of compressible viscous flow", *AIAA journal*, Vol. 20, number 9 pp. 1275-1281, 1982.

[23] MacCormack, K. R. Kneile, "Implicit solution of the 3-D compressible Navier-Stokes equations for internal flows", *Springer-Verlag*, Ninth International Conference on Numerical Methods in Fluid Dynamics, 1985.

[24] N. S. Madhavan, V. Swaminathan, "Implicit numerical solution of unsteady Euler equations for Transonic flows", *Indian J. pure appl. Math*, vol. 17(9),pp: 1164-1173,September 1986.

[25] Maurice J Zucrow and Joe D Hoffmann, "Gas dynamics", *Vol 1*, 1796.

[26] Rhie C.M. and Chow W.L., "Three-dimensional unsteady Euler equations solution using flux vector splitting", *00121*,1983.

[27] Richtmyer, R.D. and Morton, K.W., "Difference methods for initial-value problems", *Interscience Publishers*, 1969.

[28] R. Kalimuthu., "Surface Pressure Measurement Results on the SRE (biconic) Configuration at Mach = 5.",

[29] Sommer, Tomáš and Helmich, Martin, "Mathematical modeling of fluid flow using the numerical scheme with artificial viscosity",

[30] S. Nagdewe, G. Shevare, and H.-D. Kim., "Study on the numerical schemes for hypersonic flow simulation.", *Shock Waves 19,*, pp. 433442, 2009.

[31] Qin, Ning, "Towards numerical simulation of hypersonic flow around space-plane shapes", *PhD thesis*, The Univrsity of Glasgow, 1987.