The author has granted a non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of this thesis in microform, paper or electronic formats.

The author retains ownership of the copyright in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author’s permission.

L’auteur a accordé une licence non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de cette thèse sous la forme de microfiche/film, de reproduction sur papier ou sur format électronique.

L’auteur conserve la propriété du droit d’auteur qui protège cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

0-612-45872-5
Abstract

A numerical analysis is presented in this study for solving high-speed fluid flows with strong shocks and discontinuities. This analysis includes an assessment of a number of first- and higher-order numerical methods, based on their accuracy and computational effort they require to solve unsteady, compressible one-dimensional Euler equations. The effect of an adaptive grid refinement procedure on the discontinuity capturing is evaluated. The Godunov's first-order time-marching method with dimensional operator splitting and an exact Riemann solver is used to solve high-speed flows governed by two-dimensional, unsteady, compressible Euler equations. Its performance, both with and without grid adaptation, is compared to that of the CLAWPACK software. The comparison of numerical results produced by various methods indicates that while Godunov's method's performance is superior to that of hibrid higher-order schemes, a truly second-order accurate numerical scheme of CLAWPACK provides a more accurate solution with the reduced computational effort.
Acknowledgements

I would like to thank the following individuals who made significant contributions to this research:

- Professor Gottlieb for his excellent guidance and supervision throughout my research.
- Doctor Gonor for his discussions/lectures on physics.
- Luis Manzano for help with CFD and LaTex.
- People in Gasdynamics and Structures groups for their friendship, sense of humour and support.
- My parents and my girlfriend for putting up with and supporting me.

Igor Chterental

University of Toronto
September 29, 1999
Nomenclature

Alphanumeric Symbols

\( a \), speed of sound
\( c_v \), specific heat at constant volume
\( A \), area of the passage
\( \dot{A} \), flux-Jacobian matrix
\( e \), total energy
\( \hat{F} \), \( x \)-direction flux matrix
\( \hat{F}^* \), Roe flux matrix
\( g \), limiter function
\( \hat{G} \), \( y \)-direction flux matrix
\( h \), enthalpy
\( \dot{H} \), modified numerical flux
\( p \), pressure
\( R \), gas constant
\( \dot{S} \), source matrix
\( t \), time
\( T \), temperature
\( \Delta t \), time step
\( u \), \( x \)-component of velocity
\( \dot{U} \), state matrix
\( v \), \( y \)-component of velocity
\( x \), horizontal coordinate of physical reference frame
\( \Delta x \), spatial step
\( y \), vertical coordinate of physical reference frame
Greek Symbols

\(\delta\)  entropy correction parameter
\(\gamma\)  specific heat ratio
\(\lambda\)  eigenvalues of flux-Jacobian matrix
\(\rho\)  density
\(\psi\)  coefficient of numerical viscosity
\(\Psi\)  flux-limiter function
\(\delta w\)  wave speed
Contents

Abstract ii

Acknowledgments iii

Nomenclature iv

Contents vi

1 Introduction 1
  1.1 Development of Numerical Methods ................................ 1
  1.2 Shock-Capturing Methods ............................................ 2
    1.2.1 First-Order Methods ........................................... 2
    1.2.2 Second-Order Methods ........................................ 3
  1.3 Motivation for the Present Study .................................. 5

2 One-Dimensional Unsteady Compressible Flow Model 6
  2.1 Flow Equations .................................................... 6

3 Procedure for Solving One-Dimensional Euler Equations 9
  3.1 Introductory Remarks .............................................. 9
  3.2 Godunov's First-Order Method with Exact Riemann Solver .......... 10
  3.3 Roe TVD Scheme .................................................. 13
    3.3.1 Roe Approximate One-Dimensional Riemann Solver ............. 14
  3.4 Yee-Harten TVD Scheme .......................................... 17
  3.5 ARC1D .......................................................... 18
  3.6 FLOMG .......................................................... 20
  3.7 Comparison of the Numerical Methods ................................ 21

4 Adaptation 26
4.1 Introductory Remarks .................................................. 26
4.2 Adaptation of Ess ..................................................... 27
4.2.1 Adaptation Function .............................................. 27
4.2.2 Initial Movement ................................................... 28
4.2.3 Movement Limitation .............................................. 29
4.2.4 Movement Correction ............................................. 30
4.2.5 Applications to One-Dimensional Problems .................. 32

5 Two-Dimensional Unsteady Compressible Flow Model .......... 33
5.1 Introductory Remarks ................................................ 33
5.2 Two-Dimensional Problems ........................................ 34
5.2.1 Shock-Tube Problem ............................................. 34
5.2.2 Modified Shock-Tube Problem .................................. 35
5.2.1 Riemann Problem Across a Circular Interface ............... 36

6 Concluding Discussion .................................................. 38
6.1 Summary and Conclusions .......................................... 38
6.2 Recommendations for Future Work ............................... 39

References ........................................................................ 40
Chapter 1

Introduction

1.1 Development of Numerical Methods

Since the early 1950's, a great number of numerical schemes have been devised for the computer simulation of compressible gas dynamics problems. A major difficulty that has motivated research in this area of physics is the problem of representing shocks and contact discontinuities which arise in these simulations. The first solution to this problem was proposed as early as 1950 by [von Neumann and Richtmyer, 1950]. However, certain disadvantages of their approach, such as severe smearing of discontinuities, have geared the research towards more accurate and/or stable and simple solutions.

Computational techniques for solving hyperbolic conservation laws which govern compressible gas dynamics, can be classified into two categories [Anderson, 1984], [Groth, 1992]. One group of methods can be called shock-capturing methods whereas another one is usually identified as shock-tracking methods. The essential difference between these two categories lies in how they handle the discontinuities. In shock-fitting methods, discontinuities solution jumps are satisfied exactly by applying Rankine-Hugoniot jump conditions, hence, allowing to obtain solutions on relatively coarse numerical grids. Therefore, by employing less nodes in the discretization of the flow-field, shock-fitting requires a relatively small CPU time to resolve a given problem. Such advantage, however, is only noticeable for one-dimensional and relatively simple multi-dimensional problems. As the number of discontinuities and the degree of difficulty of the problem increases, the book-keeping and CPU memory requirements
associated with the numerical implementation of shock-fitting methods, grows considerably and the methods become uneffective [Anderson, 1984], [Glimm, 1985]. On the other hand, in shock-capturing methods, the conservation laws are solved without any special treatment of the discontinuities which makes these algorithms much easier to implement. However, the discontinuities and shocks are smeared out, especially if the first-order time-marching methods are applied. In order to compensate for this effect and achieve a reasonable accuracy, a finer numerical grid is required to resolve the solutions, especially for two-dimensional problems with interacting shocks and discontinuities.

Although both methods have been successfully used to solve various compressible gas dynamics problems, the shock-capturing techniques have generally been more popular, thanks to the greater conceptual and coding simplicity as well as more efficient computer performance. Currently, most of the algorithm development is concentrated in the area of improving the existing shock-capturing techniques [Woodward, Collela, 1984], [Beam, Warming, 1976], etc. Several shock-capturing methods are discussed in the following section from the historical perspective.

1.2 Shock-Capturing Methods

1.2.1 First-Order Methods

Von Neumann and Richtmyer invented the first and for a long time most commonly used approach to represent discontinuities. Their method was based on the fact that the real flow had no discontinuities, but rather contained very thin regions of extremely steep gradients. By adding an artificial viscosity to the conservation equations, they ensured that the sharp discontinuities in the solutions of these PDE's would no longer develop, having been replaced by smooth shock transitions. As a result, transition zones would be smeared over several computational cells, decreasing the resolution of the solution and the accuracy of the method.

In 1954, Lax proposed a new explicit numerical scheme for calculating one-dimensional, time-dependent compressible flows containing shocks [Lax, 1954]. It was simpler than von Neumann and Richtmyer's approach in that it did not introduce artificial viscosity explicitly. Instead, in order to stabilize the numerical procedure, Lax introduced an average value of
source terms into a forward time-differencing scheme, which acted as a dissipative term. Lax's method, however, was still characterized by the large smearing of the shocks and discontinuities over several mesh points.

An entirely different, more mathematically accurate and elegant approach was proposed by Godunov in [1959]. In Godunov's approach the nonlinearity is explicitly introduced into the computational method. Unlike von Neumann and Richtmayer, Godunov did not force a physically inappropriate large artificial dissipation at the discontinuity by adding viscous terms to the conservation equations. Neither did he introduce dissipation into the differencing scheme, like Lax proposed. Instead, his method employed an exact solution to an initial value Riemann problem, posed between adjacent nodes in order to calculate numerical fluxes through the cell boundaries. Such an approach enables Godunov's numerical scheme to recognize the shocks, contact surfaces and rarefaction waves and converge to a unique solution. Compared to less sophisticated von Neumann and Richtmayer's and Lax's methods, Godunov's explicit upwind time-marching method reduces the spread of discontinuities over several computational cells in the general solutions to unsteady, compressible fluid dynamics problems.

1.2.2 Second-Order Methods

In an attempt to improve accuracy of the time-marching methods, many implicit and explicit second-order accurate algorithms for solving hyperbolic conservation PDEs have been developed over the past 30 years. The first explicit second-order method was formulated by Lax and Wendroff [1960], followed by the explicit upwind scheme of Warming and Beam [1976]. Although these methods have reduced the smearing effect present in the solutions obtained by first-order methods, they have introduced another major drawback - pre-shock and post-shock oscillations could be observed in the solutions. This phenomenon often leads to the occurrence of physically impossible negative pressures and densities in the vicinity of strong shocks.

In the 70's and 80's, there has been a considerable development of more accurate, oscillation-free upwind-type second-order numerical methods for solving conservation laws. It was found that by adding an implicit or explicit artificial dissipation terms to the central or forward difference algorithm it was possible to suppress the oscillations. One of the first successful implementations of this approach was reported by Pulliam [1986] who developed an im-
plicit second-order unsplit difference scheme with artificial dissipation based on Warming and Beam's method - ARC1D, a one-dimensional version and ARC2D, a two-dimensional one.

A different approach to the eliminations of the dispersive oscillations near the shocks has been devised in the middle of the 70's. It was proposed to seek nonlinear, high-order, solution-dependent schemes that produced monotonic solutions of first-order schemes while retaining the accuracy of the higher-order methods. In 1974, Van Leer developed such a scheme, based on the combination of Lax-Wendroff and Warming-Beam upwind algorithms with the flux limiting function. In the next ten years several other similar high-order Godunov-type methods followed: MUSCL by Van Leer [1979], more accurate generalized Riemann Problem method by Ben-Artzi and Falcovitz [1984], piecewise parabolic method of Colella and Woodward [1984], very accurate total variation diminishing (TVD) schemes of Roe [1981], Harten [1984]. All of these algorithms required exact or approximate Riemann solvers to evaluate fluxes at the cell boundaries.

Although all of the first and second order methods described above have been developed for solving one-dimensional conservation laws (with the exception of ARC2D method), the extension of these schemes to multi-dimensions is possible by using dimensional splitting whereby solutions for each dimension are obtained separately and then combined at each time step to produce solution for the entire flowfield. Some of the most commonly used dimension splitting techniques that are presently in use include Godunov's splitting in which the solution is determined for each dimension once per time step, and Strang's [1968] splitting that requires the solution to be evaluated in x-dimension at half time step, followed by a full time step in y-direction and another half step in the x-direction. LeVeque [1995] has successfully used both dimension splitting techniques while implementing CLAWPACK - a multi-dimensional solver for the hyperbolic conservation equations that was based on the second-order upwind Godunov-type flux-limiting [Sweeby, 1985] scheme. He was able to demonstrate that the solution obtained by using the second-order scheme was more accurate than the one produced by the standard Godunov first-order method with the exception of some small regions in the solution domain.

Many high-order, solution-dependent algorithms for solving conservation laws of gas dynamics have been investigated and developed over the past three decades, but there is no
single algorithm that can be considered the ultimate one. The numerical scheme is chosen for a particular type of application individually, on the basis of simplicity, accuracy and stability.

1.3 Motivation for the Present Study

The present growing interest in the multi-dimensional conservation equations stems from demands for the more sophisticated models, capable of predicting flowfields, containing strong shocks and/or vorticies, which is the case with the flows around supersonic aircraft, space shuttles, detonation and stellar shock waves, and many more applications.

The long-term goal of this project is to develop a two-dimensional code capable of calculating the flowfield around the spherical particle which is struck by the shock wave, propagating through the liqueuos explosive. Detonation codes capable of performing similar calculations have already been developed at Los Alamos Scientific Laboratory [Mader, 1979], using a first-order upwind method, with first-order explicit artificial viscosity added to the numerical scheme for stability considerations. The purpose of the present study is to implement a more accurate first-order Godunov method, employing the exact Riemann solver to evaluate numerical fluxes at the cell boundaries while using Godunov's dimensional splitting to solve the problem in two dimensions. An important aspect of the algorithm implementation will be to compliment the computer code with the exact Riemann solver for the Tait's equation of state which will be used to model the explosive. In this procedure, the temperature of the flow will have to be determined implicitly, from second law of thermodynamics, combined with the two-dimensional energy equation. Then, the CFD code can be used as a tool for investigating the structure of this high-speed flow with reflecting and interacting shocks in trans- and supersonic regimes. A key aspect of this study will be to accurately determine and explain the location of the pressure peak and a so-called 'hot-spot' in the temperature distribution.
Chapter 2

One-Dimensional Unsteady Compressible Flow Model

2.1 Flow Equations

The most general equations governing the non-stationary, compressible gas dynamics are hyperbolic, three-dimensional conservation PDEs, which are called Navier-Stokes equations. The system of these highly non-linear equations includes real-gas and viscous effects, and the heat transfer. By introducing an appropriate equation of state, the system of Navier-Stokes equations can be closed and theoretically solved to yield thermodynamic variables as functions of three space variables and time. However, currently, there exists no exact mathematical solution to this system for any equation of state. There are numerical methods that allow scientists to solve Navier-Stokes equations, but only for certain boundary conditions and relatively simple equations of states [Hirsch, 1990]. Either way, the numerical methods are complex and the general solution is extremely computationally and financially expensive, requiring the fastest computers. Fortunately, for many practical applications it is possible to use simplifying assumptions and obtain reliable, accurate solutions by representing the physical situation with the two-dimensional model with truncated or no source terms. The detonation problem of this project, for example, can be mathematically described with two-dimensional conservation laws, so-called Euler equations, without source terms, combined with the Tait's equation of state.
Since the CFD code of this project will be based on the dimensional splitting, it is essential to first develop a code capable of solving unsteady compressible one-dimensional gas dynamics problems, such as interaction of gases in the shock tube. The system of equations, governing this type of problems, is called one-dimensional Euler equations. It is well documented and has been used in practical engineering applications such as flows within ducts and pipes for many decades [Zucrow and Hoffman, 1976].

The equations of conservation of mass, momentum and energy for general one-dimensional, unsteady, compressible flows of fluid can be expressed in the standard matrix notation as:

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{F}}{\partial x} = \mathbf{S},$$

where $\mathbf{U}$, $\mathbf{F}$ and $\mathbf{S}$ are the column matrices containing state, flux and source terms. The $\mathbf{U}$ and $\mathbf{F}$ matrices are composed of homogeneous terms and are defined by

$$\mathbf{U} = \begin{bmatrix} \rho \\ \rho u \\ \rho (e + \frac{1}{2} u^2) \end{bmatrix},$$

$$\mathbf{F} = \begin{bmatrix} \rho u \\ \rho u^2 + p \\ u (\rho e + \frac{1}{2} \rho u^2 + p) \end{bmatrix},$$

where $p$, $\rho$, $u$, $e$, $x$ and $t$ denote the static pressure, density, flow velocity, internal energy per unit mass, position in the flow and time, respectively. The $\mathbf{S}$ column matrix includes inhomogeneous source associated with the potentials arising from area change, friction, head transfer and losses. It is represented by

$$\mathbf{S} = \begin{bmatrix} -\rho u \frac{1}{A} \frac{dA}{dx} \\ -\rho u^2 \frac{1}{A} \frac{dA}{dx} + F_f + F_h \\ -u (\rho e + \frac{1}{2} \rho u^2 + p) \frac{1}{A} \frac{dA}{dx} + q \end{bmatrix},$$

where $A$, $F_f$, $q$ and $F_h$ represent the cross-sectional area of the passage as the function of $x$, body force per unit volume due to friction, heat transfer per unit volume and the body force due to head loss per unit volume, respectively.
In order to solve for the intensive variables, the system of one-dimensional Euler equations must be closed by introducing an equation of state. In this study, two equations of state are employed. A well-known and most often used in engineering applications ideal gas law that relates the temperature and density of polytropic gases will be used to solve test problems in both one and two dimensions. This equations of state is expressed as

$$p = \rho RT, \quad (2.5)$$

where $T$ is the temperature and $R$ is the gas constant. The internal energy, $e$, of polytropic gases can be related to the pressure and density by the following relation

$$e = \frac{p}{\rho (\gamma - 1)} = c_v T, \quad (2.6)$$

where $\gamma$ is the specific heat ratio and $c_v$ is the specific heat at constant volume. The speed of sound, which is important in the determination of the size of the time step in the Godunov's time-marching method, is related to other intensive properties as follows

$$a = \sqrt{\gamma RT} = \sqrt{\frac{p}{\rho}}, \quad (2.7)$$

$a$ being the speed of sound.

The Tait equation of state is valid for liquids and is more complicated to use than the ideal gas law because it directly relates only pressure and density, while the temperature can be calculated from the second law of thermodynamics and the energy equation. Here is the expression for this equation of state

$$p = B \left( \frac{\rho}{\rho_0} \right)^\gamma - B, \quad (2.8)$$

where $B = \frac{p_0 c_0^2}{\gamma}$, and $\gamma, \rho_0$ and $c_0$ are the parameters specific to a given liquid. The internal energy can be calculated from

$$e = \frac{p - c_0^2 (\rho - \rho_0)}{(\gamma - 1) \rho} \quad (2.9)$$

The speed of sound is related to intensive properties by

$$a = \sqrt{\frac{dp}{d\rho}} = \sqrt{\frac{\gamma (p + B)}{\rho}} \quad (2.10)$$

The Tait equation of state has been successfully used in studying flows of water, in particular, underwater explosions [Flores and Holt, 1981].
Chapter 3

Procedure for Numerically Solving One-Dimensional Euler Equations

3.1 Introductory Remarks

The model for the shock tube problem requires a numerical method that can identify the discontinuities such as interacting shocks and contact surfaces which are contained in the general solution to non-linear conservation PDEs. Therefore, the numerical method must produce weak solutions that satisfy the conservation laws and are differentiable at least everywhere except at the discontinuities. Another requirement to the numerical scheme is that it is 'shock-capturing' as opposed to 'shock-tracking' in order to avoid an additional effort needed for any special treatment of the discontinuities.

In this study, four different schemes are implemented in order to solve the shock tube problem. They are compared based on both qualitative and quantitative measures such as simplicity of implementation, accuracy and CPU time required to converge to the solution. The four procedures that were used to solve the shock tube problem are the explicit, first-order classic Godunov’s method with the exact Riemann solver, the explicit first-order Euler method in TVD scheme with approximate Riemann solver of Roe, explicit first-order Euler method in TVD scheme with approximate Riemann solver of Yee and Harten and second-order five-stage predictor-corrector Runge-Kutta method (RK5) designed specifically for multi-grid methods. All of these methods have been widely used to solve gas dynamics problems and are well-
documented. In addition, implicit second-order Beam-Warming method [1976] in the ARC1D algorithm is used to solve the subsonic/transonic converging/diverging nozzle problem.

In the following sections various numerical algorithms that were employed to solve one-dimensional Euler equations are briefly discussed and compared.

3.2 Godunov's First-Order Method with Exact Riemann Solver

In 1959, Godunov proposed an upwind scheme that was drastically different from the previous schemes of von Neumann and Richtmyer [1950] and Lax [1954]. These schemes did not consider the physical propagation of perturbations along characteristics, typical of hyperbolic PDEs. Godunov, on the other hand, introduced all the physical properties of the flow into the scheme. In Godunov's method, the conservative variables are treated as piecewise constant over the computational cells at each time step and the solution is advanced in time by exactly solving the Riemann problem at each cell boundary. Therefore, physical quantities obtained from the exact local solution to the Euler equations are introduced into the discretization.

Discretization of the initial solution at time \( t = n\Delta t \) in the piecewise manner guarantees that the spatial error will be of order \( \Delta x \) and so, the scheme will be spatially first-order accurate. Note that this approximation is a one-dimensional analogue of the finite-volume, cell-centered approach where the discrete values represent average values of the state variables over each cell.

Consider one-dimensional Euler equations with source terms given by Eq. (2.1). Integration of this equation in time and space over the \( n - i \) cell will allow to determine the state variables, represented by the matrix \( \hat{U} \) as a function of convective fluxes, \( \hat{F} \) and source terms \( \hat{S} \).

\[
\int_{t_{n}}^{t_{n+\Delta t_n}} \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \left( \frac{\partial \hat{U}}{\partial t} \right) dx dt + \int_{t_{n}}^{t_{n+\Delta t_n}} \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \left( \frac{\partial \hat{F}}{\partial x} \right) dx dt = \int_{t_{n}}^{t_{n+\Delta t_n}} \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \hat{S} dt dx. \tag{3.1}
\]

First two integrals can be simplified immediately by applying a fundamental theorem of Cal-
culus and re-written in the following form:

\[
\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \left[ \dot{U} (x, t_n + \Delta t_n) - \dot{U} (x, t_n) \right] dx + \int_{t_n}^{t_n + \Delta t_n} \left[ \tilde{F} (x_{i+\frac{1}{2}}, t) - \tilde{F} (x_{i-\frac{1}{2}}, t) \right] dt = \\
\int_{t_n}^{t_n + \Delta t_n} \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \tilde{S} dt dx. \quad (3.2)
\]

Since the initial solution was assumed to be spatially piecewise constant (i.e., it could be written as \( \hat{U}^n_i \) in each cell between boundaries located at \( x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}} \)), it will produce a V-shaped wave pattern at each cell boundary as the time progresses from \( t_n \) to \( t_{n+1} \).

The wave patterns consist of rightward and leftward moving waves with the contact discontinuity between them. These wave patterns depend only on \( \xi_i \) for homogeneous Euler equations. Each wave composing a pattern can be either a rarefaction or a shock wave. The shock line and the rarefaction characteristics will be straight lines in the \( x - t \) plane [Gottlieb, 1998]. However, this is not true for the Euler equations with the source terms. The solutions associated with the wave patterns are not \textit{self-similar} (not dependent on \( \xi_i \)) and the shock as well as characteristics of rarefaction wave can no longer be represented by straight lines. However, Godunov assumed that the wave patterns of inhomogeneous equations are straight lines, thus approximating them by the wave patterns of homogeneous equations. This assumption is the first-order accurate which corresponds to the global order of accuracy of the entire scheme.

It is possible to prevent the wave patterns arising from each discontinuity from crossing by controlling the size of the time step with the suitable CFL condition. Each boundary at \( x_{i-\frac{1}{2}} \) can now be viewed as an initial value Riemann problem with the left and right states defined by \( \hat{U}_{i-1} \) and \( \hat{U}_i \) respectively. The solution of the Riemann problem allows to determine the types of waves (shock or rarefaction) emanating from the discontinuity and the flow properties in the different regions formed between the waves and the contact surface. The solution of Riemann problem for the one-dimensional homogeneous Euler equations coupled with the ideal gas law contains five distinct regions. It could be calculated from a set of these highly non-linear equations only by iteration. The algorithm for this procedure, called Riemann solver, is fully discussed in [Groth and Gottlieb, 1988].

State and flux matrices \( \hat{U} \) and \( \hat{F} \) are constant along the straight wave patterns since all of the flow properties are constant there. This implies that the fluxes do not change in time along
the interfaces of the \( n - i \) cell. Therefore, the second integral in Eq. 3.2 can be evaluated. Moreover, the \( \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \hat{U}(x, t_n) \) term in the first integral of Eq. 3.2 is just a solution matrix \( \hat{U} \) in the cell \( n - i \) at initial time level \( t_n \). Since it was assumed to be constant, it can be explicitly written as \( U^n_i \). Then, Eq. 3.2 becomes

\[
\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \hat{U}(x, t_n + \Delta t_n) \, dx = \Delta x \hat{U}^n_i - \Delta t_n \left( \hat{F}^n_{i+\frac{1}{2}} - \hat{F}^n_{i-\frac{1}{2}} \right) + \int_{t_n}^{t_n+\Delta t_n} \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \hat{S} \, dt \, dx. \tag{3.3}
\]

Extending the assumption of piecewise constant solution to the next time level \( t_{n+\Delta t} \), Godunov proposed that the solution matrix \( \hat{U}(x, t_{n+\Delta t}) \) is also constant. The integral on the left-hand side of Eq. 3.3 can now be evaluated to yield the following expression:

\[
\Delta x \hat{U}^{n+1}_i = \Delta x \hat{U}^n_i - \Delta t_n \left( \hat{F}^n_{i+\frac{1}{2}} - \hat{F}^n_{i-\frac{1}{2}} \right) + \int_{t_n}^{t_{n+\Delta t_n}} \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \hat{S} \, dt \, dx. \tag{3.4}
\]

In order to evaluate the integral containing the source terms, Godunov assumed that they do not vary with time within the \( n - i \) cell, i.e. they can be represented by \( \hat{S}^n_i \) and taken out from under the integral sign. The double integral \( \int_{t_n}^{t_{n+\Delta t_n}} \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \hat{S} \, dt \, dx \) is just an area of the \( n - i \) cell, which, for rectangular cell, is equal to \( \Delta x \Delta t_n \). Substituting this into Eq. 3.4 and dividing by \( \Delta x \) we recover

\[
\hat{U}^{n+1}_i = \hat{U}^n_i - \frac{\Delta t_n}{\Delta x} \left( \hat{F}^n_{i+\frac{1}{2}} - \hat{F}^n_{i-\frac{1}{2}} \right) + \Delta t_n \hat{S}^n_i, \tag{3.5}
\]

Godunov's first-order explicit time marching method.

The size of \( \frac{\Delta t_n}{\Delta x} \) is usually restricted in order to either prevent the wave patterns from crossing or to ensure that if they do cross, their portions that travel into the cell \( n - i \) do not leave the cell. The Courant-Frederichs-Lewy solution-dependent criterion is used to enforce the above restrictions:

\[
\frac{\Delta t_n}{\Delta x} \leq \frac{1}{\max \{|u^n_i| + a_i^n\}}. \tag{3.6}
\]

Godunov's method gave rise to the family of upwind methods that are referred to as flux difference splitting methods, whereby the localized solution to the Riemann problem is obtained with approximate, or linearized Riemann solver. Two of such methods are presented in the following sections.
3.3 Roe TVD Scheme

This scheme belongs to a family of higher-order upwind numerical schemes that are second-order accurate in space, whose time accuracy depends on the specifically chosen time-integration method. It is based on the concept of total variation diminishing (TVD) which states that the total variation of any physically admissible solution

\[ TV = \int |\frac{\partial u}{\partial x}| \, dx \]  

(3.7)
does not increase in time [Hirsch, 1990]. All TVD schemes are automatically monotonicity preserving, which for linear TVD schemes implies that they are only first-order accurate. However, this restriction does not apply to the non-linear TVD schemes, which can be second-order accurate. One of such schemes was developed by Roe and it is presented below (with explicit first-order Euler time-integration method).

\[ \tilde{U}_{i+1}^{n+1} = \tilde{U}_i^n - \frac{\Delta t_n}{\Delta x} \left( 1 + \frac{1}{2} \Psi \left( \frac{r^+_{i+\frac{1}{2}}}{r^-_{i+\frac{1}{2}}} \right) \right) \left( \tilde{F}_i - \tilde{F}_{i-\frac{1}{2}}^* \right) \]  

(3.8)

\[ \frac{1}{\Delta x} \left( 1 + \frac{1}{2} \Psi \left( \frac{r^-_{i+\frac{1}{2}}}{r^+_{i+\frac{1}{2}}} \right) \right) \left( \tilde{F}_{i+\frac{1}{2}}^* - \tilde{F}_i \right) \]

\[ r^+_{i+\frac{1}{2}} = \frac{F_{i+1} - F^*_{i+\frac{1}{2}}}{F_{i+1} - F_{i+\frac{1}{2}}} \]

\[ r^-_{i+\frac{1}{2}} = \frac{F_{i-1} - F^*_{i+\frac{1}{2}}}{F_{i-1} - F_{i+\frac{1}{2}}} \]

where \( \Psi(r) \) is the limiter function, \( F \) is the convective flux, \( F^* \) is the Roe-averaged flux, and \( r^\prime s \) are the flux-difference ratios. The non-limited form of this scheme with \( \Psi = 1 \) is linearly, unconditionally unstable while the non-linear, limited TVD form is conditionally stable, since the boundedness of the total variation scheme ensures the stability [Hirsch, 1990]. This demonstrates the essential role of flux-limiters. After the numerical flux of the spatially first-order accurate scheme has been extended to second-order accuracy, non-linear flux-limiters restrict the amplitude of the gradients as to ensure TVD conditions. Several effective flux-limiters have been developed up to date. Here are, respectively, minmod,
superbee and Van Leer limiters - three most-commonly used ones that were included into the computer-code of this study.

\[
\psi = \begin{cases} 
0 & r \leq 0 \\
\min(r, 1) & r > 0
\end{cases}
\]

\[
\psi = \max(0, \min(2r, 1), \min(r, 2))
\]

(3.9)

\[
\psi = \frac{r + |r|}{1 + r}
\]

The standard CFL criterion was used with this scheme to restrict the size of the \( \Delta t_n \) time-step to prevent the wave-patterns from crossing:

\[
\frac{\Delta t_n}{\Delta x} \leq \frac{1}{\max_i (|a_i^r| + u_i^n)}.
\]

(3.10)

where \( a_i^r \) and \( u_i^n \) are the local speed of sound and velocity of the discretized solution. The convective flux \( F \), given in 2.1, is evaluated at the computational mesh points, whereas Roe-averaged flux, \( F^* \) is calculated between the nodes, at the specified cells’ boundaries, employing Roe approximate Riemann solver, which is given in the full detail below.

### 3.3.1 Roe Approximate One-Dimensional Riemann Solver

In order to develop an approximate Riemann Solver for the Euler equations, the equations are first written in a quasi-linear form. The original system without source terms

\[
\frac{\partial \tilde{U}}{\partial t} + \frac{\partial \tilde{F}}{\partial x} = 0
\]

(3.11)

becomes

\[
\frac{\partial \tilde{U}}{\partial t} + \tilde{F} \frac{\partial \tilde{U}}{\partial x} = 0,
\]

(3.12)

or

\[
\frac{\partial \tilde{U}}{\partial t} + \tilde{A} \frac{\partial \tilde{U}}{\partial x} = 0
\]

(3.13)

The matrix \( \tilde{A} = \frac{\partial \tilde{F}}{\partial \tilde{U}} \) is called a flux-Jacobian matrix, and it is approximated by a locally constant, linearized matrix \( \tilde{A}^* \) that depends on the local average state \( \tilde{U}^* \).

\[
\frac{\partial \tilde{U}}{\partial t} + \tilde{A}^* (\tilde{U}^*) \frac{\partial \tilde{U}}{\partial x} = 0
\]

(3.14)
The average state is a function of the neighboring states $\bar{U}_i$ and $\bar{U}_r$. The determination of the mean value Jacobian matrix $\tilde{A}^*$, the average state $\bar{U}^*$, and, ultimately, the Roe-averaged flux is the goal of the approximate Riemann solver.

Matrix $\tilde{A}^*$ has to satisfy the following properties [Hirsch, 1990]:

1. $\tilde{A}^* = \tilde{A} (\bar{U}^*)$,

2. its eigenvalues, $\lambda$, are all real and linearly independent,

3. $\tilde{F}_r - \tilde{F}_i = \lambda (\bar{U}_r - \bar{U}_i)$,

4. for $\bar{U}_i = \bar{U}_r = \bar{U}$ the matrix $\tilde{A}^* = \tilde{A}$.

Roe [1981] has been able to define a scheme for constructing such a matrix and evaluating averaged-fluxes $F^*$. The scheme can be given in full details in the following form.

1. For each cell $i$, the averaged values of intensive properties are calculated:

$$\rho_{i+\frac{1}{2}}^* = \sqrt{\rho_{i+1}} = R_{i+\frac{1}{2}} \rho_i,$$

where $R_{i+\frac{1}{2}} = \sqrt{\frac{\rho_{i+1}}{\rho_i}}$.

$$u_{i+\frac{1}{2}}^* = \frac{(u \sqrt{\rho})_{i+1} + (u \sqrt{\rho})_i}{\sqrt{\rho_{i+1}} + \sqrt{\rho_i}} = \frac{R_{i+\frac{1}{2}} u_{i+1} + u_i}{R_{i+\frac{1}{2}} + 1}$$  \hspace{1cm} (3.15)

$$h_{i+\frac{1}{2}}^* = \frac{(h \sqrt{\rho})_{i+1} + (h \sqrt{\rho})_i}{\sqrt{\rho_{i+1}} + \sqrt{\rho_i}} = \frac{R_{i+\frac{1}{2}} h_{i+1} + h_i}{R_{i+\frac{1}{2}} + 1}$$

$$(a_{i+\frac{1}{2}}^*)^2 = (\gamma - 1) \left( h_{i+\frac{1}{2}}^* - \left( \frac{u_{i+\frac{1}{2}}^*}{2} \right)^2 \right).$$

2. Eigenvalues of $\tilde{A}^*$, which are the wavespeeds associated with the Euler equations and eigenvectors, which represent the waves, are determined for each cell $i$ from

$$\begin{pmatrix} \lambda_{i+\frac{1}{2}}^* \\ \lambda_{i+\frac{1}{2}}^* \\ \lambda_{i+\frac{1}{2}}^* \end{pmatrix}_1 = u_{i+\frac{1}{2}}^* \quad \begin{pmatrix} \lambda_{i+\frac{1}{2}}^* \\ \lambda_{i+\frac{1}{2}}^* \end{pmatrix}_2 = u_{i+\frac{1}{2}}^* + a_{i+\frac{1}{2}}^* \quad \begin{pmatrix} \lambda_{i+\frac{1}{2}}^* \end{pmatrix}_3 = u_{i+\frac{1}{2}}^* - a_{i+\frac{1}{2}}^*$$  \hspace{1cm} (3.16)
with the corresponding eigenvectors

\[
\begin{align*}
    r_1^* &= \begin{bmatrix} 1 \\ u^* \\ \langle w^* \rangle^2 \end{bmatrix}, \\
    r_2^* &= \frac{\tilde{\delta}^*}{2a^*} \begin{bmatrix} 1 \\ u^* + a^* \\ h^* + u^*a^* \end{bmatrix}, \\
    r_3^* &= \frac{\tilde{\delta}^*}{2a^*} \begin{bmatrix} 1 \\ u^* - a^* \\ h^* - u^*a^* \end{bmatrix},
\end{align*}
\]  

(3.17)

where all starred variables have a subscript \( i + \frac{1}{2} \), i.e. they are all determined for cell \( i \), between nodes \( i \) and \( i + 1 \).

3. Wave amplitudes and density, pressure and velocity differences are calculated next, according to the following expressions.

\[
\begin{align*}
    (\delta w_{i+\frac{1}{2}}) &= \delta \rho_{i+\frac{1}{2}} - \frac{\delta p}{(a^*)^2}, \\
    (\delta w_{i+\frac{1}{2}}) &= \delta u_{i+\frac{1}{2}} + \frac{\delta p}{\rho a^*}, \\
    (\delta w_{i+\frac{1}{2}}) &= \delta u_{i+\frac{1}{2}} - \frac{\delta p}{\rho a^*}, \\
    \delta z_{i+\frac{1}{2}} &= z_{i+1} - z_i,
\end{align*}
\]

(3.18)

(3.19)

where \( z \) is \( p, \rho, u \).

4. Finally, the Roe-averaged numerical flux between states \( i \) and \( i + 1 \) can be evaluated using the following formula:

\[
\tilde{F}_{i+\frac{1}{2}}^* = \frac{1}{2} \left( \tilde{F}_{i+1} + \tilde{F}_i \right) - \frac{1}{2} \sum_{j=1}^{3} \left| \left( \lambda_{i+\frac{1}{2}}^* \right)_j \right| \left( \delta w_{i+\frac{1}{2}} \right)_j \left( r_{i+\frac{1}{2}}^* \right)_j,
\]

(3.20)

where \( \delta w_j, r_j^* \) and \( \left( \lambda_{i+\frac{1}{2}}^* \right)_j \) are as defined in the previous steps.

The Roe-averaged flux of Eq. (3.15), combined with time-marching scheme of Eq. (3.5) with flux-limiters defined as in Eq. (3.9) form the complete numerical solution algorithm for solving one-dimensional Euler equations.

### 3.4 Yee-Harten TVD Scheme

In 1984, Harten and Yee have devised an easy to implement, quickly-converging TVD upwind second-order scheme, a so-called modified-flux approach. The modified-flux scheme is a technique that allows to design a second-order accurate TVD scheme by starting with a first-order
TVD scheme and incorporating a modified flux into it. The modified flux is chosen so that
the scheme is second-order in smooth regions and first-order accurate at points of extrema.
The smearing of discontinuities which is present in the first-order Godunov's method is consi-
dered less when Yee-Harten approach is used. The updated solution on the time level \( t_{n+1} \)
is a function of the modified flux, \( \tilde{\mathcal{H}} \)

\[
\tilde{U}_{i+1}^{n+1} = \tilde{U}_{i}^{n} - \Delta t^{n} \frac{\partial}{\partial x} \left( \tilde{H}_{i+\frac{1}{2}}^{n} - \tilde{H}_{i-\frac{1}{2}}^{n} \right),
\tag{3.21}
\]

where \( \tilde{H}_{i+\frac{1}{2}}^{n} \) and \( \tilde{H}_{i-\frac{1}{2}}^{n} \) are the modified numerical fluxes at the cell interfaces. The modified
numerical flux is calculated from

\[
\tilde{H}_{i+\frac{1}{2}}^{n} = \frac{1}{2} \left( \tilde{F}_{i+1}^{n} + \tilde{F}_{i+1}^{n} + \tilde{R}_{i+\frac{1}{2}}^{n} \tilde{\phi}_{i+\frac{1}{2}}^{n} \right),
\tag{3.22}
\]

where \( \tilde{R}_{i+\frac{1}{2}}^{n} \) is the matrix whose columns are the eigenvectors of the flux-Jacobian, given in
3.17. The elements of \( \tilde{\phi}_{i+\frac{1}{2}}^{n} \) are calculated from

\[
\phi_{i+\frac{1}{2}} = \sigma \left( \lambda_{i+\frac{1}{2}} \right) (g_{i+1} + g_{i}) - \psi \left( \lambda_{i+\frac{1}{2}} + \gamma_{i+\frac{1}{2}} \right) \alpha_{i+\frac{1}{2}},
\tag{3.23}
\]

where \( \lambda_{i+\frac{1}{2}} \) are characteristic speeds, eigenvalues of the flux-Jacobian matrix, and \( \sigma \left( \lambda_{i+\frac{1}{2}} \right) \) is equal to \( \sigma_{i+\frac{1}{2}} \) for time-accurate calculations and is defined by

\[
\sigma(x) = \frac{1}{2} \psi(x)
\tag{3.24}
\]

The function \( \psi \left( \lambda_{i+\frac{1}{2}} \right) \) is the coefficient of numerical viscosity. It is well-known that \( \psi \) term
has to be modified in order to satisfy entropy condition [Yee, 1987]. In its entropy-corrected
form \( \psi(x) \) is defined as

\[
\psi(x) = \begin{cases} 
|x| & |x| \geq \delta_1 \\
(x^2 + \delta_1^2) / 2\delta_1 & |x| < \delta_1
\end{cases}
\tag{3.25}
\]

The parameter \( \delta_1 \) is a small positive entropy correction parameter that is set to zero in most
computations for the simple unsteady problems containing shocks, such as a the shock-tube
problem. For steady problems with strong shock waves, the parameter is strongly dependent
on the Mach number and geometry of the physical problem. The size of \( \delta_1 \) can be viewed as a
measure of the amount of numerical dissipation for the modified numerical flux. The smaller
the \( \delta_1 \), the less dissipative the scheme is.
The function $\gamma$ is defined by

$$
\gamma_{i+\frac{1}{2}} = \begin{cases} 
\frac{1}{2} \psi_{i+\frac{1}{2}} (g_{i+1} - g_i) / \alpha_{i+\frac{1}{2}} & \alpha_{i+\frac{1}{2}} \neq 0 \\
0 & \text{otherwise}
\end{cases}
$$

Here $\alpha_{i+\frac{1}{2}}$ are elements of

$$
\alpha_{i+\frac{1}{2}} = \tilde{R}_{i+\frac{1}{2}} (\tilde{U}_{i+1} - \tilde{U}_{i+1})
$$

In the above equation $g$ is the limiter function that can be defined by any of the limiters discussed in the previous section or by an original limiter designed by Yee

$$
g_i = \frac{\alpha_{i-\frac{1}{2}} (\alpha_{i+\frac{1}{2}}^2 + \delta_2) + \alpha_{i+\frac{1}{2}} (\alpha_{i-\frac{1}{2}}^2 + \delta_2)}{\alpha_{i-\frac{1}{2}}^2 + \alpha_{i+\frac{1}{2}}^2 + 2\delta_2}
$$

where $\delta_2 = 10^{-7}$.

### 3.5 ARClD

ARClD is an implicit second-order accurate in space approximate factorization finite difference scheme which utilizes central finite differencing for spatial discretization. Original scheme uses second-order accurate three point Beam-Warming implicit time-marching method. In this study, implicit first-order accurate Euler method was employed instead. It is the only implicit method out of five time-marching methods that were employed in this study. Such methods appeared in the late 70’s. Generally, explicit methods require less computational work and are conceptually simpler in both derivation and application. Implicit methods, on the other hand, have much higher stability bounds. In fact, classical Fourier stability analysis demonstrates unconditional stability, but in practical non-linear problem bounds are still present.

Implicit numerical algorithms are chosen whenever it is desired to obtain solutions that require fine grid resolution because the implicit methods do not restrict the size of the time step as severely as conditionally stable explicit methods. The extra work required for an implicit scheme typically is offset by the advantages resulting from the higher stability limits, and implicit schemes have been successfully used for a variety of inviscid and viscous flow calculations [Groth, 1988], [Pulliam, 1986].

ARClD can be employed for both steady and unsteady problems, although it works considerably better for the steady cases. For unsteady problems, it is desired to achieve as high
temporal accuracy as possible, initializing the flow with some physically admissable state and integrating in time with time steps consistent with the transient phenomena that is being investigated. Although both implicit and explicit methods are capable of doing that, an implicit method might be superior in handling numerous strong shock interactions, where the stability of the solution is concerned. In steady state problems, the only requirement to the numerical method is that it can integrate to asymptotic solution in the least amount of time and with the least computational expenses. Such time non-accurate, but convergence increasing techniques as matrix preconditioning and large time steps can be used both with implicit and explicit time-marching methods as long as they do not produce inaccurate solutions.

In ARC1D, time linearizations are applied locally to the non-linear terms, followed by an approximate factorization of the two-dimensional operator into two implicit one-dimensional block trydiagonal operators. The solution is now obtained by consecutive one-dimensional sweeps in x- and y-directions whereby a tridiagonal matrix is inverted once per sweep using lower-upper decomposition which is considerably faster and requires less computer storage than when using un facto red algorithm. The use of factorization gives the scheme an improved robustness and an increased rate of convergence for the steady cases. Explicit and implicit second- and forth-order artificial dissipation terms are added to the factored form of the algorithm in order to prevent oscillations at the shocks.

Both one- and two-dimensional versions of this algorithm, which is presented in detail in [Pulliam, 1986], have been successfully used at NASA Ames Research Center to predict flowfields around airfoils at various angles of attack as well as in this study for the nozzle problem.

### 3.6 FLOMG

This multi-grid (four grids) explicit cell-centered, finite-volume (for more than one dimension) method uses RK5 time-marching method to advance the solution to the next time level. This numerical scheme is second-order accurate in time and first-order accurate in space [D.W.Zingg, 1998].

Artificial dissipation model utilized in FLOMG is a non-linear combination of second-
and forth-order differences. Dissipation terms with adaptive coefficients provide an increased stability, faster convergence and an improved shock-capturing. These terms are added to the governing conservation equations after they have been discretized instead of the original equations. Such approach has been proven to effectively prevent oscillations near shock waves and stagnation points. Moreover, the introduction of appropriate dissipation in the vicinity of shocks permits to satisfy the entropy condition, and, therefore, guarantees the uniqueness of weak solutions. A small point about artificial dissipation should be made here, although it does not greatly affect the shock tube problem. It should be noted that it is tempting not to include the artificial dissipation terms when dealing with viscous flows since the dissipative properties are automatically present due to diffusive source terms. However, the physical dissipation may often not be sufficient to guarantee enough stability, especially in the case of very steep gradients. Thus, for the guaranteed best performance of this numerical scheme it is necessary to include artificial dissipation regardless of the type of flow in the problem.

The system of discretized conservation equations is advanced in time with the RK5 method which is expressed mathematically as

\[
\tilde{U}_i^{n+1} = \tilde{U}_i^0 - \alpha_{m+1} \frac{\Delta t}{\Delta x} \left( \mathcal{L}_C \tilde{U}_i^m + \mathcal{L}_D \tilde{U}_i^0 - \sum_{n=0}^{m} \gamma_{mn} \mathcal{L}_{AD} \tilde{U}_i^n \right),
\]

where \( Q_i^{n+1} = Q_i^5 \) is the solution at the new time level, \( n + 1 \), \( Q_i^n = Q_i^0 \) is the discrete solution at time level \( n \), \( \alpha_{m+1} \) are the coefficients of the scheme, subscripts \( AD \), \( C \) and \( D \) refer to artificial dissipation, convectio and diffusion, and \( \gamma_{mn} \) are the weighting factors of the dissipation model. The coefficients \( \alpha_{m+1} \) are determined so that the numerical scheme possesses the highest possible stability limit, and are given by

\[
\alpha_1 = \frac{1}{4}, \quad \alpha_2 = \frac{1}{6}, \quad \alpha_3 = \frac{3}{8}, \quad \alpha_4 = \frac{1}{2}, \quad \alpha_5 = 1
\]

This scheme has good high-frequency oscillations damping properties, which is essential for a rapidly converging multi-grid method. This is ensured by a suitable choice of dissipative weighting factors \( \gamma_{mn} \) which are given in [Zingg, 1998], where a full description of this method can be found.

FLOMG's performance has been compared to that of ARClD which used an implicit Euler method and while the solutions produced by two methods were very similar, the multigrid method was considerably faster.
3.7 Comparison of the Numerical Methods

Five numerical methods for solving unsteady and steady compressible Euler equations have been described in the previous sections. In this section their performances are analyzed and compared in order to determine the method that is computationally least expensive and provides the highest solution accuracy. Such a method would be the preferable choice for solving multi-dimensional problems where the high grid resolution is essential to obtaining correct solution. In multi-dimensional problems, even a relatively small number of nodes in each direction requires a significant computational effort and it is important to minimize the number of nodes necessary for the convergence to the exact solution. The assessment of the numerical methods was conducted based on approximate solutions they produced for a simple one-dimensional shock-tube problem whose analytic solution is well-known.

The shock tube problem addresses many of the important flow aspects that might be encountered in the more realistic situation. It is an initial value problem, where two fluids with different parameters are separated by a thin diaphragm which is broken at time \( t = 0 \). As a result, a shock wave, a rarefaction wave and a contact surface start propagating with the strength and in the directions determined by initial pressure, temperature and density ratios. The analytic solution to this problem is easily obtained for gases obeying the ideal gas law by using Rankine-Hugoniot relations and Riemann invariants. In this particular case, gases are placed in the 10 m long duct, with the diaphragm positioned in the middle. Both left and right state gases are represented by air. The pressures and densities are 100 Pa, 10 Pa and \( \frac{1}{3}, \frac{1}{6}, \frac{1}{9}, 0.125 \frac{1}{10} \) for the left and right states respectively. Gases are initially at rest. Once the diaphragm is removed, a shock wave and a contact surface form and move to the right whereas the rarefaction wave starts moving to the left. The solution was calculated at time \( t = 6.1 \text{ ms} \), which is before the waves reached the ends of the duct, and so, were easily distinguishable.

The shock tube problem was solved using Godunov, Roe, Yee-Harten and FLOMG methods. The pressure, density, temperature and velocity distributions are depicted in Fig. 1, 2, 3 and 4, respectively. The exact solution is represented by the solid line while the numerical solutions are plotted with points. Equal node spacing was used for all methods.

Clearly, all methods have been able to correctly predict the structure of the resulting
flowfield as well as calculate, with varying degrees of accuracy, the spatial distribution of flow variables.

The Godunov explicit finite-difference method produced an oscillation-free solution which is, however, highly smeared at the contact surface, the shock wave and corners of the rarefaction wave. The shock wave is spread over 4 grid cells, while the dissipation at the contact surface is even more drastic - 13 cells. The position of the rarefaction wave is predicted relatively accurately, however, its upper corner is smeared over 8 cells. Such behaviour is expected from this very dissipative scheme. It is Godunov's method dissipative nature that propelled the research towards more accurate schemes in the first place.

Figure 2 shows the results obtained with the Roe TVD method, where explicit Euler time-marching method was used with the Roe approximate Riemann solver to calculate numerical fluxes and Roe "superbee" flux-limiter was applied to limit them. The method captures the rarefaction wave slightly better, while the smearing is reduced by one computational cell through the contact surface and the shock. The position of the shock is determined more accurately than with the Godunov method. Even if other, less effective, flux-limiters, such as minmod and van Leer limiters, are used the shock capturing is superior to that of Godunov method. However, when minmod limiter is used, small oscillations are present at the corners of the rarefaction wave and at the shock.

Explicit TVD method of Yee-Harten produces an approximate solution depicted in Fig. 3. This method offers an excellent shock capturing, by far superior to both Roe and Godunov methods. Smearing is drastically reduced to 3 cells at one of the corners of rarefaction wave. The dissipation through the contact surface is only 6 cells. Small oscillations are present at the shock and the lower corner of rarefaction wave which disappear as the number of nodes is increased.

The results obtained using FLOMG method are presented in Fig.4. This multigrid algorithm contains 4 grids of various coarseness and requires number of nodes to be a power of 2. The lowest possible number of nodes is 32 or 2^5. The graphs demonstrate that the method is capable of capturing the rarefaction wave almost perfectly, while the contact surface and the shock are smeared over 9 and 4 cells respectively. Considering that the nodal spacing is half that for all other methods, it can be concluded that FLOMG method is as good as the
Yee-Harten TVD scheme. However, solution obtained by FLOMG contains a lot of oscillations that have the highest amplitude at the shock and contact surface, and quickly (over a distance of approximately 10 cells), die out. It should be pointed out that the number and amplitude of these Gibbs-like oscillations is much higher if the implicit and explicit artificial dissipation terms are not used. The fluctuations become less apparent as the number of nodes increases, just like with the Yee-Harten method.

In order to evaluate these four numerical schemes quantitatively CPU time required for each method to obtain the solution at desired time was calculated for different number of nodes. For Godunov's, Roe's and Yee-Harten's method 64, 100, 128, 200, 256, 512 and 1028 nodes were used. For FLOMG CPU times were computed for 64, 128, 256, 512 and 990 nodes. The computations were performed on an SGI Origin 2000 with 8 250MHZ R10000 processors and 2 gigabytes of memory, running IRIX 6.4, connected to via Apollo Series 400 computer.

Figure 13 shows the CPU time required by each method to reach the solution to the shocktube problem versus number of nodes. The standard FORTRAN 77 compiler was used without any optimization of the code or compilation. The graph shows that the computational cost associated with all methods quickly increases non-linearly with the increasing number of nodes. This implies that while solving potentially computationally expensive problems, it is important to perform spacial discretization carefully in order to obtain solution of the desired accuracy within reasonable time limits. The curves clearly demonstrate that the first order explicit Godunov's time marching method that employs the exact Riemann solver for flux calculations is by far faster than the three second-order time-marching methods. Moreover, as the grid becomes finer, the difference in CPU time between Godunov's method and second-order methods grows. Therefore, Godunov's method is superior to the Yee-Harten's, Roe's and FLOMG methods, where the CPU time consumption is concerned.

Figure 9 shows the solution to the shocktube problem for 990 nodes. All four methods provide a much more accurate solution than they did for the smaller number of nodes. The Godunov's method is still the least accurate of all while each of the higher order methods shows some advantages and disadvantages as Figures 10-12 demonstrate. Figure 14 indicates that the numerical error produced by each method decreases as greater number of nodes is employed, but the difference between the errors (for the same number of nodes) associated
with Godunov’s method and higher order methods of Yee-Harten and FLOMG has grown, as expected. It follows that the Godunov’s method, while being considerably faster than second-order methods used in this study, produces a less accurate solution. However, it is possible to achieve the same accuracy with Godunov’s method as with the higher order methods presented here by using more nodes. For example, FLOMG requires 0.37 s of CPU time to obtain the solution to the shocktube problem, in the process producing a 0.0121 \( \frac{kg}{m^3} \) error, when 128 nodes are used. Godunov’s method solves the same problem in 0.33 s, producing a numerical error of only 0.00933 \( \frac{kg}{m^3} \), while employing 256 nodes.

The discussion of the numerical methods for solving Euler equations can be summarized as follows:

- The analysis of the graphical representation of the solution proves that Godunov’s method is the least accurate of the four methods considered here. However, none of the methods can be said to provide clearly the “best” accuracy. The FLOMG method appears to be able to follow the exact solution better than other schemes in the smooth regions, but it produces a lot of oscillations when it encounters the shock and the contact surface. Yee-Harten’s method exhibits, to a smaller degree, same kind of oscillations, while following shock better than any of the methods. Roe’s method is as good as FLOMG at the rarefaction wave, but is slightly worse than both Yee-Harten’s method and FLOMG at the discontinuities.

- Godunov’s method is without dispute the fastest of these four methods, with FLOMG and Roe’s method being distant second and third with almost identical performance. Yee-Harten’s method is the slowest scheme being almost 15 times slower than Godunov’s method at 990 nodes.

- Godunov’s method can be used for quick, relatively accurate estimates of the solution whereas second order methods should be employed if the higher accuracy is required. However, if the memory capabilities of the computer allow to use a lot of nodes, Godunov’s method performs better than higher-order methods discussed above, both in terms of accuracy and computational effort.

- In addition, Godunov’s method is conceptually the easiest and clearest of four methods
presented here. It is considerably faster to implement than FLOMG and slightly faster than both TVD schemes.

Godunov's time-marching scheme with the exact Riemann solver was chosen to solve multi-dimensional Euler equations because of its low CPU time requirements and robustness that allow to use finer grids and, hence, achieve the required accuracy faster than higher-order multigrid and TVD schemes discussed in this paper.
Figure 1: Solution to shocktube problem using Godunov's method.
Figure 2: Solution to shocktube problem using Roe method.
Figure 3: Solution to shocktube problem using Xee-Harten method.
Figure 4: Solution to shocktube problem using FLOMG.
Figure 5: Solution to subsonic problem using ARC1D method.
Figure 6: Solution to transonic nozzle problem using ARC1D method.
Figure 7: Solution to subsonic problem using FLOMG method.
Figure 8: Solution to transonic nozzle problem using FLOMG method.
Figure 9: Shocktube problem. 990 nodes. Density distribution.
Figure 10: Shocktube problem. 990 nodes. Density distribution through shock.
Figure 11: Shocktube problem. 990 nodes. Density distribution through contact surface.
Figure 12: Shocktube problem. 990 nodes. Density distribution through rarefaction wave.
Figure 13: CPU time versus number of nodes for the shocktube problem.
Figure 14: Numerical error in density versus number of nodes for the hocktube problem.
Chapter 4

Adaptation

4.1 Introductory Remarks

A numerical algorithm for solving gas dynamics problems is required to be as efficient and accurate as possible. However, this is, to a large extent, contradiction in terms, since for a discrete solution to converge to an exact one, the mesh has to be progressively refined, detrimenting the efficiency.

Since fluid flows exhibit large gradients in the vicinity of discontinuities, uniform grids have to be deformed to fit the geometry of physical problem as well as have an ability to reform themselves dynamically to the evolving solution.

In numerical computations, where computer power is one of the limiting factors, efficiency constrains mesh size and hence the level of accuracy that can be reached. Therefore, instead of arbitrarily increasing mesh size, a different approach is employed. Given the relatively coarse mesh, the nodes are redistributed based on the solution obtained from an initial arbitrary (usually uniform) mesh. The redistribution process is based on minimization of the numerical error associated with the particular grid.

There exists a variety of adaptation schemes, such as a so-called torsion spring method [Nakahashi, Deiwert, 1986], adaptive mesh refinement (AMR) method, reported in [Bell et al., 1994], and adaptive grid procedure of [Ess, 1997]. The latter is used in this study to achieve more effective shock capturing in both one- and two-dimensional problems.
4.2 Adaptation of Ess

Following the variational approach of [Nakahashi, Deiwert, 1986], total numerical error is reduced when the grid points are distributed such that

$$\int_{x_i}^{x_{i+1}} w(x) \, dx = \text{const}, \quad (4.1)$$

where $w(x)$ is a positive adaptation function. Ideally, $w(x)$ would be the magnitude of the truncation error that results from the discretized solution. Practically, this error is often impossible to accurately determine, and it is generally approximated by the gradients of the solution itself. Such use of the gradients of the solution as a driving mechanism for grid adaptation is allowed because the largest numerical errors are found in regions where the solution changes most rapidly. Based on the above variational statement, [Agrawal et al., 1990] have developed a "self-adaptive" scheme that is given below.

They defined an adaptation cycle which consists of four distinct parts: adaptation function calculation, initial nodal movement calculation and movement limitation followed by movement correction.

4.2.1 Adaptation Function

Adaptation function is calculated at each node based on the gradients of an intensive variable, such as pressure or density. It is defined as

$$w(x, y) = \frac{\partial^n}{\partial s^n} = \sqrt{\sum_{k=0}^{n} \left( \frac{1}{k!(n-k)!} \frac{\partial^n}{\partial x^k \partial y^{n-k}} \Delta x_{\text{max}} \Delta y_{\text{max}} \right)^2} \quad (4.2)$$

Here, $n$ is taken to be 1, and $\Delta x_{\text{max}}$, $\Delta y_{\text{max}}$ are the maximum distances from each node $i$ to its cloud points, $P_j^i$. They are defined as

$$\Delta x_{\text{max}} = \max_{j=1}^{C} (x_j^i - x_i), \quad (4.3)$$

$$\Delta y_{\text{max}} = \max_{j=1}^{C} (y_j^i - y_i), \quad (4.4)$$

where $x_j^i$ and $y_j^i$ are, respectively, $x$ and $y$ coordinates of each of the cloud points $j$ associated with the node $i$. The number of cloud points, $C$, varies depending on the grid. In this study where the quadrilateral grid was used, $C$ is equal to 4.
The adaptation function is composed of partial derivatives of an intensive variable, with respect to \( x \) and \( y \), which can be approximated by central differences for the interior points and forward and backward differences at the boundaries. However, the commonly used differencing formula for interior points

\[
\left( \frac{\partial f}{\partial x} \right)_i = \frac{1}{2\Delta x} (f_{i+1} - f_{i-1}),
\]

is valid only for a uniformly distributed nodes. In case of the non-uniform grid, the following formula should be used

\[
\left( \frac{\partial f}{\partial x} \right)_i = \alpha f_{i-1} + \beta f_i + \gamma f_{i+1},
\]

where

\[
\alpha = \frac{x_{i+1} - x_i}{(x_{i-1} - x_i)(x_{i+1} - x_{i-1})},
\]

\[
\gamma = \frac{x_{i-1} - x_i}{(x_{i+1} - x_i)(x_{i+1} - x_{i-1})}
\]

and

\[
\beta = \frac{2x_i - (x_{i-1} + x_{i+1})}{(x_{i-1} - x_i)(x_{i+1} - x_i)}.
\]

At the boundaries, usual forward and backward differencing can be applied regardless of the uniformity of the grid.

\[
\left( \frac{\partial f}{\partial x} \right)_{i=last} = \frac{1}{\Delta x} (f_i - f_{i-1})
\]

and

\[
\left( \frac{\partial f}{\partial x} \right)_{i=1} = \frac{1}{\Delta x} (f_{i+1} - f_i)
\]

### 4.2.2 Initial Movement

The initial movement of nodes is calculated as a weighted Laplacian function expression for which was derived from the initial variational statement

\[
\Delta \vec{r}_{i,\text{next}} = s \sum_{j=1}^{C} \frac{(\vec{r}_{j,\text{next}} - \vec{r}_{i,\text{prev}})}{\sum_{j=1}^{C} w_j} w_j,
\]

where \( s \) is the relaxation parameter, typically equal to 0.1 in one-dimensional unsteady problems, \( \vec{r}_j \) and \( \vec{r}_i \) are, respectively, a position vector of \( j^{th} \) cloud point of the node \( i \) and position vector of the node \( i \) itself. The superscripts prev and next refer to subsequent iteration levels of the inner adaptation cycle whose length could be determined either empirically, by an experimentally (trial-and-error) determined number, or by some measure of cell's skewiness.
Figures 1-7 show how the nodes are redistributed around the curves of higher gradients.

### 4.2.3 Movement Limitation

Movement limitation consists of two parts: all boundary points can be moved only tangentially to the physical boundary and the interior points movement may have to be adjusted if the initial movement procedure results in the unnecessary nodal clustering.

In the boundary point movement, the tangential movement vector can be calculated for each boundary node from

$$
\Delta \vec{r}_{i} = \frac{\hat{t} \Delta \vec{r}}{\hat{t} \cdot \hat{t}},
$$

(4.13)

where $\hat{t}$ is the vector tangential to the surface.

The interior points' movement is limited by the set of the following equations. First, for each node $i$ the minimum distance to its cloud points is calculated by

$$
d_{\text{min}_p,i} = \min_{j=1}^{c} |\vec{r}_j - \vec{r}_i|,
$$

(4.14)

where the smaller the value of $d_{\text{min}_p,i}$ is, the greater the limitation of the movement will be. Once the $d_{\text{min}_p,i}$ is obtained, the parameter $\xi$ can be calculated

$$
\xi_i = \frac{d_{\text{min}_p,i} - d_{\text{ming}}}{d_{\text{max}_g} - d_{\text{ming}}},
$$

(4.15)

for the use in the so-called blending function

$$
f_i = 1 - 3\xi^2 + 2\xi^3.
$$

(4.16)

The parameters $d_{\text{ming}}$ and $d_{\text{max}_g}$, are, respectively, the minimum and the maximum globally allowed distances from node $i$ to its cloud points. The parameter $\xi$ indicates how close to the node the cloud points are relative to the globally allowed distance. Next, the critical value, $d_{ci}$, is determined based on the blending function by

$$
d_{ci} = c_{\text{max}} d_{\text{min}_p,i} \left(1 - f_i^\lambda\right),
$$

(4.17)

where parameters $\lambda$ and $c_{\text{max}}$ help limit $d_{ci}$ and hence $\Delta \vec{r}$. Finally, the limited value of the displacement vector is given by

$$
\Delta \vec{r}_{\text{lim},i} = \begin{cases} 
\Delta \vec{r}_i & \text{for } |\Delta \vec{r}_i| \leq d_{ci} \\
\Delta \vec{r}_i \frac{d_{ci}}{|\Delta \vec{r}_i|} & \text{for } |\Delta \vec{r}_i| > d_{ci}
\end{cases}
$$

(4.18)
4.2.4 Movement Correction

The movement correction procedure ensures that the movement of nodes which violate specified geometric rules is reduced. In addition, the movement of all the neighbours of such a node is similarly decreased. When the node $i$ is present in the calculation of the adaptation function at that node (as is the case for the non-uniform formulation), it is restricted considerably more than its neighbours, if it happens to be in the region of higher solution gradients. Therefore, in order to avoid the situation where some points are allowed to move much more than the others, thus creating highly irregular meshes, an additional smoothing is introduced. It consists of three stages: nodal movement correction, simultaneous cloud point correction and correction smoothing. Ultimately, the correction is achieved through multiplying the displacement vector $\Delta \tilde{r}_{lim}$, by movement correction coefficient $c_{\text{move}}$, which is a small positive number, less than or equal to 1. Note that $\Delta \tilde{r}_{lim}$ is equal to $\Delta \tilde{r}_i$ if no limitation is applied. The corrected displacement is defined as

$$\Delta \tilde{r}_{\text{cor}, i} = c_{\text{move}} \Delta \tilde{r}_{lim}, \tag{4.19}$$

The coefficient $c_{\text{move}}$ that is initially set to 1 for all points, is a scaling factor that ranges from 0 to 1. At each stage of the inner adaptation cycle it is modified until the final $\Delta \tilde{r}_i$ is established.

In order to calculate final values of $c_{\text{move}}$ it is first rescaled for each node and all of the node's cloud points by

$$c_{\text{move},i} = c_{\text{move}} c_{\text{cor}}, \tag{4.20}$$

$$c_{\text{move},j} = c_{\text{move}} c_{\text{cor}}, \tag{4.21}$$

where $c_{\text{move},j}$ is the coefficient for the cloud points $j$ of the node $i$ and $c_{\text{correct}}$ is a small positive parameter less than 1. The smaller $c_{\text{cor}}$ is, the more rapidly the movement of the nodes is decreased. The correction smoothing coefficient $c_{\text{move},i}$ is finally determined from

$$c_{\text{move},i} = \frac{1}{2} \left( c_{\text{move},i} + \sum_{j=1}^{C} \zeta_i^j c_{\text{move},i} \right), \tag{4.22}$$

with the node to node distance influence factor $\zeta_i^j$ given by

$$\zeta_i^j = \frac{1}{d_i^2} \frac{1}{\sum_j C d_j^2}, \tag{4.23}$$
where $d_i^j$ is the distance from node $i$ to each of its cloud points.

The correction procedure is applied over several correction cycles until a proper distribution of nodes is achieved. The termination of the correction procedure is dictated by the set of rules that put bounds on various parameters, such as $d_{\text{minp}_i}$. The purpose of these rules is to achieve higher concentration of nodes in the regions of the large flow gradients without producing highly irregular meshes in the process. The rules include a minimum and maximum distance rules and an angular difference rule.

A minimum distance rule places a constraint on the $d_{\text{minp}_i}$ such that it can not be greater than the specified minimum globally allowed distance. Any node that violates the rule will be subject to the movement correction procedure. Mathematically, the rule is expressed as

$$d_{\text{minp}_i} = \max \left( \frac{C}{\text{min}} |\tilde{r}_j - \tilde{r}_i|, d_{\text{ming}} \right),$$  \hspace{1cm} (4.24)

for all nodes.

A maximum distance rule requires that the distance between node $i$ and any of its cloud points does not exceed the globally maximum allowed distance, $d_{\text{maxg}}$

$$d_{\text{maxp}_i} = \min \left( \frac{C}{\text{max}} |\tilde{r}_j - \tilde{r}_i|, d_{\text{maxg}} \right),$$  \hspace{1cm} (4.25)

for all nodes.

The angular difference rule states that the angle between two lines connecting node $i$ with two neighbouring cloud points should be no less than a specified globally allowed minimum angle. This rule prevents two cloud points from moving too close to each other, ensuring that the grid is not extremely distorted at any given time in the inner adaptation cycle.

$$\alpha_{\text{minp}_i} = \max \left( \frac{C}{\text{min}} \alpha_j, \alpha_{\text{ming}} \right)$$  \hspace{1cm} (4.26)

for all nodes.

Application of the above rules in the inner adaptation cycle ensures that while the adopted grid provides the improved resolution in the large flow gradients regions, it is not highly distorted. Comparing Figures 2 and 5 one can see that the correction procedure restricted movement of the nodes in the region of high gradients, thus ensuring a smoother nodal distribution.
4.2.5 Applications to One-Dimensional Problems

The adaptation procedure described in the previous subsections was coupled with each of four methods used to solve the shock tube problem. Figures 7-10 indicate that it was possible to improve the accuracy of the solution in each case by 10 to 20 per cent. The computational costs associated with the adaptation procedure proved to be negligible compared to the overall CPU time requirement for each method. For example, for Godunov's method, it required a total of 0.18 s to calculate the solution when 100 nodes were used, 0.03 s up from the non-adapted case. The adaptation worked the best for the oscillations-free Godunov method, where it allowed to achieve almost perfect shock-capturing. It did not work quite as well for Yee-Harten method. While improving overall accuracy and providing excellent shock and contact surface capturing, it tended to amplify the oscillations at the corner of rarefaction wave and at the shock. However, when adaptation was applied to the oscillatory FLOMG method, it improved its already high accuracy and, unlike with the Yee-Harten method, did not amplify the oscillations. The changes of gradients here were relatively small in amplitude and confined to such a small area that the entire cluster of nodes was moved as opposed to several particular nodes being redistributed individually. As a result, the shock was positioned slightly to the left of its exact location, which was still better than the original non-adapted version had it.

The adaptation procedure was also applied to solve steady problems. A problem of transonic flow through the changing-area nozzle has been solved using ARC1D and FLOMG methods. As Figures 12 and 13 indicate, adaptation reduced the numerical error in both cases. It worked especially well with ARC1D method, improving the accuracy of the solution, while preserving its shape at the shock.
Figure 1: Adaptation on the spherical grid around forth circle.
Figure 2: Adaptation on the spherical grid around seventh radius.
Figure 3: Adaptation on the spherical grid around forth circle and seventh radius.
Figure 4: Adaptation on the spherical grid around seventh and twelfth radii.
Figure 5: Adaptation on the spherical grid around seventh radius with correction.
Figure 6: Adaptation on the spherical grid around spiral.
Figure 7: Solution to shocktube problem using Godunov method on adapted grid.
Figure 8: Solution to shocktube problem using Roe method on adapted grid.
Figure 9: Solution to shocktube problem using Yee-Harten method on adapted grid.
Figure 10: Solution to shocktube problem using FLOMG method on adapted grid.
Figure 11: Solution to transonic nozzle problem using FLOMG method on adapted grid.
Figure 12: Solution to transonic nozzle problem using ARC1D method on adapted grid.
Chapter 5

Two-Dimensional Unsteady Compressible Flow Model

5.1 Introductory Remarks

The equations of conservation of mass, momentum and energy without source terms for general two-dimensional, unsteady, compressible flows of fluid can be expressed in the standard matrix notation as:

\[
\frac{\partial \bar{U}}{\partial t} + \frac{\partial \bar{F}}{\partial x} + \frac{\partial \bar{G}}{\partial y} = 0,
\]

where \( \bar{U}, \bar{F}, \bar{G} \) are the column matrices containing state and flux terms. The \( \bar{U}, \bar{F} \) and \( \bar{G} \) matrices are composed of homogeneous terms and are defined by

\[
\bar{U} = \begin{bmatrix}
\rho \\
\rho u \\
\rho v \\
\rho (e + \frac{1}{2} (u^2 + v^2))
\end{bmatrix},
\]

(5.2)

\[
\bar{F} = \begin{bmatrix}
\rho u \\
\rho u^2 + p \\
\rho uv \\
u (\rho u + \frac{1}{2} \rho (u^2 + v^2) + p)
\end{bmatrix},
\]

(5.3)
where $p, \rho, u, e, x$ and $t$ denote the static pressure, density, flow velocity, internal energy per unit mass, position in the flow and time, respectively.

\section*{5.2 Two-Dimensional Problems}

Godunov's method was applied to solve several unsteady problems in two-dimensions on the cartesian grid. In the two-dimensional form it is written as

\begin{equation}
\tilde{\mathbf{G}} = \begin{bmatrix}
\rho v \\
\rho uv \\
\rho v^2 + p \\
v \left( \rho e + \frac{1}{2} \rho (u^2 + v^2) + p \right)
\end{bmatrix},
\end{equation}

where $\rho, u, e, x$ and $t$ denote the density, velocity, internal energy, position in the flow and time, respectively.

The solution is first sought in $x$-direction, solving a quasi one-dimensional system of Eq. 5.1 without the $\tilde{\mathbf{G}}$ matrix, to determine numerical fluxes which are parallel to the $x$-axis. The procedure is repeated in $y$-direction, now removing $\tilde{\mathbf{F}}$ matrix from the original equation. Finally, the solution is updated according to Eq. 5.5, using calculated numerical fluxes in $x$- and $y$-directions.

Solutions obtained with the Godunov's method were compared to those produced by CLAWPACK software [LeVeque, 1995], which uses Godunov’s time splitting with LeVeque’s flux-splitting and Roe approximate Riemann solver. The method is second-order accurate in both time and space, unlike Roe and Yee-Harten methods discussed previously which were second-order accurate in space while using a first-order accurate Euler time-marching method.

\subsection*{5.2.1 Shock Tube Problem}

First, a two-dimensional version of the shock-tube problem was solved. On a 10 m by 10 m plane, gases of two different states are initially separated by a thin diaphragm positioned along $x$-axis at $y = 5$. As the diaphragm is removed and the gases are allowed to interact, we expect the formation of the same pattern as in one-dimensional case, along each $x$-location. Figures 1 and 2 show density distributions calculated with Godunov’s method and CLAWPACK,
respectively. CLAWPACK clearly has much better shock, contact surface and rarefaction wave representation, while Godunov's method produces usual smeared discontinuities. Figure 3 shows a solution produced on a 10x100 grid using Godunov's method. It is clear that since the flow evolves only in y-direction, changing number of nodes in x-direction has no effect on the accuracy of the solution. In fact, in this problem it would be possible to have only two nodes in x-direction at each y-location, thus specifying boundaries of the computational plane. Comparing Figures 1 and 3, it is clear that the solution on Figure 3 (1000 nodes) is more accurate than that depicted on Figure 1 (2500 nodes). Therefore, using some prior knowledge of the probable evolution of the flow, it is possible to achieve higher solution accuracy just by cleverly positioning nodes in the first place. However, even the solution on a 10x100 grid is still inferior to the one obtained by using CLAWPACK.

Figures 4-7 show that by using adaptation on 50x50 grid it was possible to substantially improve accuracy of the Godunov's scheme and make it comparable to that of LeVeque's software. However, CLAWPACK has proved to be the faster one of two codes, requiring only 6.5 s to obtain the solution as opposed to 17.56 s that the original software took to accomplish this task.

5.2.2 Modified Shock Tube Problem

In this problem, it is desired to determine wave patterns that result from the interaction between four different gas states separated by diaphragms on the 10 m by 10 m plane. Initially, gas is at rest in all four quadrants, with pressure and density defined as follows:

\[
p = \begin{cases} 
1.0 \times 10^5 & x > 5, y > 5 \\
1.1 \times 10^5 & x < 5, y > 5 \\
1.0 \times 10^4 & x < 5, y < 5 \\
1.0 \times 10^4 & x > 5, y < 5 
\end{cases},
\]

\[
\rho = \begin{cases} 
1.0 & x > 5, y > 5 \\
1.5 & x < 5, y > 5 \\
0.625 & x < 5, y < 5 \\
0.125 & x > 5, y < 5 
\end{cases},
\]
where $p$ and $\rho$ represent, respectively, pressure in Pa and density in $\text{kg/m}^3$. Figures 8-10 depict the pressure distribution at time $t = 0.006s$. There is a shock and a rarefaction waves moving, respectively to the right and to the left, along the x-axis above $y = 5$ (first and second quadrants). Similar behaviour is exhibited by the flow along y-axis, with shock waves moving in the negative y-direction. The waves in y-direction have much larger amplitudes than those in x-direction because the initial pressure and density differences were much larger along y-axis. In the CLAWPACK solution, a pressure peak is present near the common corner of four quadrants. This phenomena was reported by [Schulz-Rinne, Collins, Glaz, 1993] and is partially a result of CLAWPACK's inability to reduce the oscillations at this "quadruple" discontinuity. The original program that employed Godunov's method produced a highly smeared peak, as expected. Similarly to the previous problem, CLAWPACK provided a more accurate solution than the original program. Moreover, CLAWPACK took only 8.03 s to compute it to 19.4 s required by the original program to perform the calculation. By using the adaptation in the original program it was possible to obtain a solution with better shock resolution in all regions except for the region of intersection of four different initial gases. However, CPU time spent to get to the desired timelevel was increased to 20.7 s.

5.2.3 Riemann problem across a circular interface

All of the problems discussed to this point had initial states separated by a diaphragm perpendicular to one(or both) coordinate axis. In this section a different problem is posed. Initially, gas has density of $10 \text{ kg/m}^3$ and is under pressure of 10 Pa everywhere on a 10 m by 10 m plane, except for the region inside a circle where density and pressure are, respectively, $1 \text{ kg/m}^3$ and 1 Pa. The center of the circle is located at point (5,5), and its radius is 1 m. Moreover, the gas is initially at rest everywhere. It is again desired to compute the properties of the flow at some time $t_n$. The key to solving this problem is to determine a solution to Riemann Problem across the circular interface, when the velocities are no longer perpendicular to it. LeVeque [1995] proposed to determine, for each rectangular cell through which the interface cuts, the fraction of the cell which is inside the circle. The state variable $\hat{U}_i$ for these cells at $t = 0$ is then determined by the relation:

$$
\hat{U}_i = b\hat{U}_{in} + (1 - b)\hat{U}_{out},
$$

(5.8)
where \( b \) is the fraction of the cell which is inside the circle, \( \tilde{U}_{\text{in}} \) and \( \tilde{U}_{\text{out}} \) are the initial state matrices inside and outside the circle, respectively. After this adjustment has been made, the solution is obtained using the same dimensional splitting approach as was successfully applied to the “two-dimensional shocktube” problem.

The density distribution obtained on a 200 by 200 grid is shown in Figure 11. It represents an inward moving shock wave and an outward moving rarefaction wave around the circular interface. Physically, this may crudely represent a bubble of light gas, say, helium, surrounded by air. At some point, the bubble bursts under the higher outside pressure exerted on it by the surrounding medium, and the shock wave propagates through the lighter medium. As expected, there is a fairly high degree of the shock smearing present in the solution. CLAWPACK produced a more accurate solution depicted on Figure 12 significantly faster on a 100 by 100 grid.

The second-order accurate method of CLAWPACK has proved to be both faster and more accurate than first-order accurate Godunov approach while modeling two-dimensional high speed flows.
Density at time $t = 0.0060191$

Figure 1: Solution to 2-D shocktube problem using Godunov method. 50x50 grid
Density at time $t = 0.006$

Figure 2: Solution to 2-D shocktube problem using CLAWPACK. 50x50 grid
Figure 3: Solution to 2-D shocktube problem using Godunov method. 10x100 grid
Pressure at time $t = 0.0060191$

Figure 4: Solution to 2-D shocktube problem using Godunov method. 50x50 grid
Figure 5: Solution to 2-D shocktube problem using Godunov method. 50x50 adapted grid
Figure 6: Solution to 2-D shocktube problem using CLAWPACK. 50x50 grid
Figure 7: Solution to 2-D shocktube problem at $x=5$ m.
Figure 8: Solution to modified 2-D shocktube problem. Godunov method. 50x50 grid
Figure 9: Solution of modified 2-D shocktube problem. CLAWPACK. 50x50 grid
Figure 10: Solution of modified 2-D shocktube problem. Godunov method. 50x50 adapted grid
Figure 11: Solution of Riemann problem across circular interface. Godunov method. 200x200 grid
Figure 12: Solution of Riemann problem across circular interface. CLAWPACK. 100x100 grid
Chapter 6

Concluding Discussion

6.1 Summary and Conclusions

Computational fluid dynamics remains an essential and often the only tool to investigate the behaviour of high speed flow with interacting nonlinear discontinuities, which is present in detonation problems. To predict flows with strong shocks an accurate, efficient and robust numerical algorithms are required to solve system of hyperbolic conservation laws.

In this thesis, a number of such algorithms has been used to solve various one- and two-dimensional problems and their performance has been assessed. Second-order space accurate upwind TVD scheme of Yee-Harten that used first-order accurate time marching Euler method has proved to be slightly more accurate than the first-order space and time accurate Godunov's scheme. Same conclusions apply to a second order accurate FLOMG method. However, Godunov's scheme appeared to be less CPU time-consuming for one-dimensional problems. Combination of Godunov's method with the grid adaptation technique has improved the scheme's accuracy by around 20 per cent while negligibly increasing the CPU time requirements. Increasing number of nodes in the uniform discretization enabled to achieve even greater accuracy. Godunov's scheme allows to provide better accuracy while using less CPU time than TVD schemes that combine a higher-order space accuracy with the first-order temporal accuracy. However, second-order space and time accurate scheme of CLAWPACK is both significantly faster and more accurate than Godunov's scheme, both with and without the adaptation.
6.2 Recommendations for Future Work

Although Godunov's scheme provided satisfactory results for the variety of one- and two-dimensional problems, it requires a more efficient grid adaptation method, something like Adaptive Mesh Refinement (AMR) [Bell, et al., 1994] in order to improve its discontinuity resolution.

It is necessary to investigate whether the exact Riemann solver in the Godunov's scheme can be used in curvilinear coordinates since problems with complicated boundaries or with wave/object interactions are best solved in such coordinate system.

Considering that the second-order accurate scheme of CLAWPACK has proved to be more efficient than the Godunov's scheme, it is recommended that either original software should employ a second-order accurate Generalized Riemann Problem (GRP) scheme of [Ben-Artzi, Falcovitz, 1984] or CLAWPACK should be adapted to solve the detonation problem.
Chapter 7

References


