A higher order numerical method for transonic flows

Chen, Bo

Bo Chen
Supervised by Dr. Li He
A Thesis for the Degree of Master of Science

A Higher Order Numerical Method for Transonic Flows

by

Bo Chen

A Thesis submitted in partial fulfilment
of the requirements for the degree of
Master of Science

School of Engineering

The University of Durham
1997

Contents

Abstract

This thesis presents and verifies a numerical method for solving the compressible Euler equations. The method is based on a finite volume method with an upwind type TVD dissipation terms originally developed by Harten (1983) for scalar hyperbolic conservation law and extended to Euler equations by using Roe's approximate Riemann solver. The present method has second-order accurate in smooth region of the solution and intelligently switchs the scheme to first-order accurate in the vicinity of shocks to presents a sharp and smooth shock wave profile. The present method contains no user-dependent and problem-dependent parameters. An explicit multistage Runge-Kutta time stepping is used to integrate the system. A multigrid method is employed in the present method to accelerate to convergence. Meanwhile a fully implicit time integration scheme is also investigated and adopted in this method. The explicit multistage time stepping with the multigrid acceleration is modified to solve the fully implicit system.

The present method is programmed in two-dimensions for the Euler equations aiming at the application to internal flows. Numerical experiments are carried out to test the accuracy and the efficiency of the present method. Results compare well with exact solutions and perform better than some well-documented results. The desired efficiency is obtained.

The connection between central difference and upwind difference is investigated. It is found that the widely used Jameson's central differencing plus explicit adaptive artificial viscosity can be interpreted as a hybrid scheme by a weighted average of a first order upwind scheme and a second order upwind scheme.

Acknowledgements

I would like to deeply thank my supervisor, Dr. Li He, who gave me this opportunity to study this graduate program at the University of Durham. His continuous guidance and encouragement in both academic and personal life are highly appreciated.

The author is very much grateful to Mr. John Northridge and Mr. Paul Jarram, Rolls-Royce, Derby. At my hard time they have given me warm and friendly help, without which this study is impossible.

Many thanks will also go to my colleagues, especially to Mr. Z. Fan, School of Engineering, University of Durham, who helped the author in his preparation of this report using TEX.

I express my deep thanks to my parents for their lasting support during my life.

Chapter 1. Introduction

Computer simulations provide a powerful tool for investigations of fluid flows and can give more and more detailed information. Previously these knowledges can only be obtained by experiments which are difficult, expensive and sometimes dangerous for researchers. Meanwhile the reliability and efficiency of computer simulations heavily depend on computers' ability and human's knowledge and understanding of physics and the mathematical models they used in these simulations.

As the evolution of computer, researchers may adopt more sophisticated mathematical and numerical models which govern the fluid flows, and employ finer grids, in another word, more numerical probes in a numerical wind tunnel. At the same time, deeper insight has been reached into the field of numerical method and its application to fluid dynamics. The interplay between these two fields gave the birth of a new science *Computational Fluid Dynamics* (CFD).

Normally the first step to solve a PDE numerically is to discretize the PDE and to obtain an algebraic model, which is more or less an approximation of the PDE and can be input into a digital computer.

The most widely used discretization methods can be fall into three categories: finite difference, finite element and finite volume method.

Finite difference method is the oldest and simplest one. Its idea can be traced back to Euler in 18th century. This method is based on the Taylor expansion and can be applied to PDEs straightforwardly. One shortcoming of finite difference method is that it heavily depends on the structure of computational mesh. In another word, it needs the directions of the spatial derivatives (Fig. 1.1) .

The invention of finite element method is attributed to some structural engineers. In this method the space domain is discretized into a finite number of small, nonoverlapped elements with more arbitrary shape and size (Fig. 1.2). Normally the numerical unknowns are stored at the vertex (or nodes) of these elements. In each element the field variables are approximated by linear combinations of local interpolation functions using the nodal values. These small elements can be assembled by a variational principle or weighted residuals method, in the cases a general variational principle does not exist, to form an algebraic system, in the jargon of structural engineers, called stiffness matrix and mass matrix.

The most physically intuitive one is the finite volume method, in which the physical space is divided into a finite number of control volumes, or cells, and the integral form of conservation law is applied directly to these control volumes. This assures such quantities as mass, momentum, energy conserved at discrete level. According to the location of numerical unknowns inside the control volumes this method usually falls into two categories: cell centered and cell vertex (Fig. 1.3). The great advantage of both finite element and finite volume method is their flexibility of dealing with complex geometry while the finite volume method involves much smaller bandwidth matrix computation.

It is well known that the PDEs governing the fluid flows is always nonlinear and a direct reflection of nonlinearity in the solutions is the discontinuities. In the case of gas dynamics, there are shock waves in transonic and supersonic flows. Researchers face the question of how to simulate the discontinuity numerically using the so-called *shock capturing method.* The nonphysical oscillation and the unstability of the solution have forced them to resort to *artificial viscosity.* The advent of the concept of upwind differencing is a result of the interplay of discretization method and the theory of characteristics. Godunov's scheme is the most successful conservative upwind scheme and became the building block of most modern upwind schemes. Unfortunately, these schemes are at most first order accurate and the discontinuities are oversmeared. In a long period after the publication of Godunov's scheme most researchers still enjoy the second order central scheme with an explicit artificial dissipation although it contains user-dependent or problem-dependent free parameters and sometimes probably selects a entropy-violation solution. In the early 80s some deep understanding was reached in the field of hyperbolic PDEs and stimulated the new design principles of higher-order oscillation-free schemes using the concept of *Total Variation Diminishing.* In recent years TVD scheme becomes more and more widely used in engineering simulations. But applications of these new schemes to internal flows seems lagged far behind those to external flows. In the present study, the author intends to incorporate Harten's second order TVD scheme, which is originally designed for finite difference method to a cell-centered finite volume method and apply it to the internal flow calculations.

To solve evolution PDEs we normally integrate the system alone a time-like direction. This leads to generally two time-discretization methods: explicit method and implicit method. Furthermore, if the spatial operator is first approximated by a discrete scheme, the equations then become a system of coupled ordinary differential equations (ODEs) with the flow variables defined at a set of discrete points. This method is frequently cited as the *Method of Lines* which assures the final steady solution is independent of the time step. Some existing techniques developed for ODEs can be applied to this system. Jameson and his coworkers (1981) adopted an explicit multistage Runge-Kutta time stepping scheme while Beam and Warming (1978) employed an implicit linear multistep method.

It is well known that the explicit integration methods are highly restricted by the Courant-Freidrichs-Lewy condition and thus slow in convergence. To reach convergence as quickly as possible some acceleration techniques have been sought and incorporated into existing explicit schemes. Multiple grid method is the most powerful one. It is emphasised that when applied to the cases with complex geometries it performs well almost without any loss of efficiency. Another choice of relaxing the limit of time step is the adoption of implicit scheme, in which the ODE system is globally coupled and the region of dependence of a point is enlarged. Thus much larger time step can be used. Since the requirement of inversion of banded matrixes or solution of large nonlinear system in implicit method more work should be paid in every time step.

The presentation is organized in the following order.

In chapter 2, we first explore the connection between central and upwind difference scheme, and give the result that the widely used Jameson's central scheme with explicit artificial viscosity can be reinterpreted as a hybrid scheme composed of a weighted average of a first order upwind scheme and a second order upwind scheme. Secondly the concept of TVD and design principle for TVD schemes are reviewed, and a second order accurate TVD scheme for scalar hyperbolic conservation law is constructed following the way of Harten (1983). Thirdly, Godunov's exact Riemann solver and Roe's approximate Riemann solver are discussed. In the final section of this chapter, a 2D compressible Euler solver is constructed using the TVD concept, Roe's approximate Riemann solver, the cell-centered finite volume discretization and multistage Runge-Kutta time stepping.

In chapter 3, We discuss the multigrid method, one of most powerful acceleration technique in numerical computations. First, a brief review of the philosophy and development of multigrid method will be given. Then the detailed algorithm and implementation of Jameson's multigrid method for steady Euler equation will be reported.

In chapter 4, we concentrate on implicit schemes for time integration. First, we review the some widely used implicit schemes, most of which are linerized versions, or called *Delta form*. Then a fully implicit time integration scheme using multigrid acceleration, which is reported in chapter 3, is presented. Thirdly, application of Newton-Raphson subiteration technique to solve the fully implicit system is emphasised.

Chapter 5 demonstrates the implementation details and computational results using the method presented. Four different test cases are showed with further discussions.

A conclusion is reached in the last chapter.

Chapter 2. Spatial discretization and TVD schemes

§2.1 Why Artificial Viscosity and Upwinding

For simplicity, we first consider a linear convective equation with a constant convection speed a

$$\frac{\partial u}{\partial t} + a\frac{\partial u}{\partial x} = 0 \tag{2.1.1}$$

A central second-order difference formula can be used for the discretization of the space derivative u_x about the mesh point i. This leads to a semi-discrete scheme as follows:

$$\frac{\partial u_i}{\partial t} = -\frac{a}{2\Delta x}(u_{i+1} - u_{i-1}) \tag{2.1.2}$$

It is seen, from the Taylor's expansion about the mesh point i, that the derivatives of even orders are canceled and dispersion is dominate in the scheme. Further more, if we use explicit Euler method, for example, to discretize the time derivative u_t, anti-viscousity will be introduced, which causes the numerical procedure absolutely unstable.

To remedy this, in 1950s, von Neuman and Rychtmyer first introduced the concept of artificial viscosity to stablize the computation and treat shocks based upon the research of Becker in 1920s (Rychtmyer and Morton 1967). Becker showed in his article that there are typically two kinds of dissipation mechanism in physical shocks. One is heat conduction and the other is viscosity. In their calculations they found that when the shock is strong the artificial heat conduction fails to capture the shock but artificial viscosity is always valid no matter how strong the shock wave is. When the artificial viscosity is used in the calculation the shock wave is captured within 3-4 grid intervals. Different types of artificial viscosity are thoroughly reviewed by Hirsch (1990).

One alternative is the upwind difference based on the characteristic theory of hyperbolic partial differential equations. The basic idea of upwinding is to introduce some physical mechanism of signal propagation. Following the notation of Steger and Warming (1981), the semi-discrete scheme, with a first-order approximation for the space derivative, can be obtained as:

$$\frac{\partial u_i}{\partial t} = -\frac{a+|a|}{2\Delta x}(u_i - u_{i-1}) - \frac{a-|a|}{2\Delta x}(u_{i+1} - u_i) \qquad (2.1.3)$$

,which can be rewritten as:

$$\frac{\partial u_i}{\partial t} = -\frac{a}{2\Delta x}(u_{i+1} - u_{i-1}) + \frac{|a|}{2\Delta x}(u_{i+1} - 2u_i + u_{i-1}) \qquad (2.1.3a)$$

The second term of the RHS of eq (2.1.3a) is the approximation of the spatial derivative u_{xx}. So it can be seen that the first order accurate upwind scheme is equivalent to a second order accurate central scheme plus a second order numerical viscosity. Due to the truncation error introduced by the numerical viscosity the scheme is only first order accurate in space.

To improve the spatial accuracy we may choose one-sided second-order difference to approximate the space derivative. The semi-discrete scheme can be written as:

$$\frac{\partial u_i}{\partial t} = -\frac{a+|a|}{4\Delta x}(3u_i - 4u_{i-1} + u_{i-2}) - \frac{a-|a|}{4\Delta x}(-3u_i + 4u_{i+1} - u_{i+2}) \qquad (2.1.4)$$

,which can be rewritten as:

$$\begin{aligned}\frac{\partial u_i}{\partial t} =& -\frac{a}{4\Delta x}[4(u_{i+1}-u_{i-1})-(u_{i+2}-u_{i-2})]\\ &-\frac{|a|}{4\Delta x}(u_{i+2}-4u_{i+1}+6u_i-4u_{i-1}+u_{i-2})\end{aligned} \tag{2.1.4a}$$

It is also seen that the second order upwind scheme is equivalent to a second order central scheme plus a forth order numerical viscosity u_{xxxx} . But numerical experiment showed the oscillation in the vicinity of discontinuity, such as shocks, can not be damped out sufficiently (Steger and Warming, 1981).

Based upon above observation of the first order and second order upwind schemes, it is a natural choice to construct a composite scheme using a weighted average of these two schemes, as follows:

$$\begin{aligned}\frac{\partial u_i}{\partial t} =&(1-\theta)[-\frac{a}{2\Delta x}(u_{i+1}-u_{i-1})+\frac{|a|}{2\Delta x}(u_{i+1}-2u_i+u_{i-1})]\\ &\theta\{-\frac{a}{4\Delta x}[4(u_{i+1}-u_{i-1})-(u_{i+2}-u_{i-2})]\\ &-\frac{|a|}{4\Delta x}(u_{i+2}-4u_{i+1}+6u_i-4u_{i-1}+u_{i-2})\}\end{aligned} \tag{2.1.5}$$

where $0 \leq \theta \leq 1$, following Anderson et. al, (1986), can be called *limiter*, which is a grid function and specially designed to conservatively switch the scheme from second order to first order in the vicinity of shock. This strategy will give a smooth profile of shock while keeping the solution second order accurate elsewhere.

Furthermore, if we rewrite the new scheme in its viscous form, approximately, as follows:

$$\begin{aligned}\frac{\partial u_i}{\partial t} =& -[(1+2\theta)\frac{a}{4\Delta x}(u_{i+1}-u_{i-1})-\theta\frac{a}{4\Delta x}(u_{i+2}-u_{i-2})]\\ &+[(1-\theta)\frac{|a|}{2}\frac{\partial^2 u}{\partial x^2}\Delta x-\theta\frac{|a|}{4}\frac{\partial^4 u}{\partial x^4}\Delta x^3]\end{aligned} \tag{2.1.6}$$

It can be seen that this composite scheme can be interpreted as a central difference scheme plus an adaptive numerical viscosity. Actually, it is the central idea of Jameson's scheme, in which a blend of second and forth order artificial viscosity is explicitly added to a second order accurate central difference scheme. The complete numerical recipe of Jameson and his coworkers can be found in Jameson et. al., 1981 .

§2.1 The TVD Schemes.

Now we consider the nonlinear scalar conservation law as follows,

$$\frac{\partial u}{\partial t} + \frac{\partial f}{\partial x} = 0 \qquad (2.2.1a)$$

with initial condition

$$u(x,0) = \phi(x) \qquad -\infty < x < \infty \qquad (2.2.1b)$$

$a(u) = df/du$ is the characteristic speed.

To approximate conservation law (2.2.1a) The numerical schemes can be written in the *conservative form* as follows:

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} + \frac{\bar{f}_{i+1/2}^n - \bar{f}_{i-1/2}^n}{\Delta x} = 0 \qquad (2.2.2)$$

or

$$u_i^{n+1} = u_i^n - \lambda[\bar{f}_{i+1/2}^n - \bar{f}_{i-1/2}^n] \qquad (2.2.3)$$

where $\lambda = \Delta t/\Delta x$.

It is noticed that the numerical flux can be written in the following form:

$$\bar{f}_{i+1/2} = \frac{1}{2}[f_i + f_{i+1} - Q(\bar{a}_{i+1/2})\Delta_{i+1/2}u] \qquad (2.2.3a)$$

where $f_i = f(u_i)$, $\Delta_{i+1/2}u = u_{i+1} - u_i$.

$$\begin{aligned} \bar{a}_{i+1/2} &= (f_{i+1} - f_i)/\Delta_{i+1/2}u \qquad & \Delta_{i+1/2}u \neq 0 \\ &= a(u_i) & \Delta_{i+1/2}u = 0 \end{aligned} \tag{2.2.3b}$$

where Q is a function of $\bar{a}_{i+1/2}$ and λ, and is sometimes referred to as the coefficient of numerical viscosity.

To get some insight of numerical schemes, here we consider three classical schemes (Yee et. al. 1983)

a) A form of Lax-Wendroff (L-W) scheme with

$$\bar{f}_{i+1/2} = \frac{1}{2}[f_i + f_{i+1} - \lambda(\bar{a}_{i+1/2})^2\Delta_{i+1/2}u] \tag{2.2.4a}$$

b) Lax-Friedrichs (L-F) scheme with

$$\bar{f}_{i+1/2} = \frac{1}{2}[f_i + f_{i+1} - \frac{1}{\lambda}\Delta_{i+1/2}u] \tag{2.2.4b}$$

c)A generalization of the Courant-Issacson-Rees (GCIR) scheme with

$$\bar{f}_{i+1/2} = \frac{1}{2}[f_i + f_{i+1} - |\bar{a}_{i+1/2}|\Delta_{i+1/2}u] \tag{2.2.4c}$$

It is seen that the only difference among these three schemes is the coefficient of numerical viscosity. The coefficient of numerical viscosity of L-F scheme is a constant $1/\lambda$ and it is the most dissipative scheme among the three. The GCIR scheme, which sometimes referred to as Roe's scheme, has the coefficient of numerical viscosity $|\bar{a}_{i+1/2}|$, which, due to the restriction of CFL condition, is always less than that of L-F scheme. This makes the GCIR scheme is less dissipative than L-F scheme. Although they both belong to the class of first order scheme GCIR scheme is always preferred. One shortcoming of GCIR scheme is that it allows expansion shock, which violates the entropy condition and leads to a nonphysical solution. The L-W scheme, with $Q(\bar{a}_{i+1/2}) = \lambda(\bar{a}_{i+1/2})^2$, is a second order scheme and is fourth-order dissipative since its second order viscosity is canceled by the anti-viscosity introduced by the time discretization. Although it can give better resolution than the first order schemes the oscillation of the solution in the neighborhood of shock can not be damped out by the scheme itself.

It is known from Becker's research that the width of shock wave heavily depends on the amount of dissipation. The larger the dissipation is the wider the shock wave. In the numerical computations, the hefty amount of numerical dissipation in first order scheme or the artificial viscosity added to the centered scheme usually smears the shock and causes the loss of accuracy. To remedy this, some nonlinear, feedback mechanism should be introduced into our numerical schemes to control the amount of dissipation accurately and to represent shock sharply. The requirement of our new schemes are ,

a) At least second order accuracy on smooth solution, and also in smooth regions of a solution even when there are discontinuities elsewhere.

b) Sharp resolution of discontinuities without over-smearing and the absence of spurious oscillations in the computed solution.

c) A discrete form of the entropy condition which assure the computation converge to the correct physical solution.

Some theoretical studies have been carried out since 1950s. Godunov first introduced the concept of *monotonicity* and showed that monotonic scheme at most first order accurate. It is apparent that the requirement of monotonicity is too severe. Harten (1983) relaxed this by introducing the notion of *Total Variation Nonincreasing*, or roughly, *Total Variation Decreasing*.

The *Total Variation* of the function u can be defined as:

$$TV(u) = \int_{-\infty}^{\infty} |\frac{\partial u}{\partial x}| dx \qquad (2.2.5a)$$

For a finite-difference scheme we have

$$TV(u) = \sum_{i=0}^{i_{max}-1} |\Delta_{i+1/2}u| \tag{2.2.5b}$$

where u is a mesh function

The concept of *Total Variation Nonincreasing* can be interpreted as (Fig 2.1) .

$$TV(u^{n+1}) \leq TV(u^n) \tag{2.2.6}$$

Harten obtained a set of sufficient condition to construct TVD scheme for 1D scalar hyperbolic PDEs.

Rewrite the finite-difference scheme as follows:

$$u_i^{n+1} = u_i^n + C_{+,i+1/2}\Delta_{i+1/2}u - C_{-,i-1/2}\Delta_{i-1/2}u \tag{2.2.7}$$

$$C_{+,i+1/2} = C_+(u_{i-1}, u_i, u_{i+1}, u_{i+2}), \quad C_{-,i-1/2} = C_-(u_{i-2}, u_{i-1}, u_i, u_{i+1}) \tag{2.2.7a}$$

Harten proved that the scheme is TVD if the coefficients C_+, C_- in (2.2.7) satisfy

$$C_{-,i+1/2} \geq 0, \qquad C_{+,i+1/2} \geq 0 \tag{2.2.8a}$$

$$C_{-,i+1/2} + C_{+,i+1/2} \leq 1 \tag{2.2.8b}$$

Reconsider (2.2.4a) - (2.2.4c). It is seen that L-F scheme and GCIR scheme are TVD scheme and L-W scheme is not. There is a further distinction between L-F scheme and GCIR scheme that L-F scheme is consistent with entropy inequality whereas GCIR is not. To remedy this, the coefficient of numerical viscosity can be slightly modified as

$$\begin{aligned} Q(z) &= |z| & |z| \leq \epsilon \\ &= (z^2 + \epsilon^2)/2\epsilon & |z| < \epsilon \end{aligned} \tag{2.2.9}$$

We now turn to construct second order TVD scheme. It is noticed that in the past decades several second order accurate schemes have been designed based on different principles and methodologies. Van Leer's second order Godunov scheme is a good example. In the present study we employ Harten's modified flux TVD scheme and extend it formally to 2D Euler Eq. using the cell-centered finite-volume method.

We start with the first-order TVD scheme (2.2.3) with numerical flux (2.2.3a) and a modified coefficient of numerical viscosity, function $Q(z)$, (2.2.9) and applied it to a modified flux $f + g$.

The new numerical flux function $\bar{f}_{i+1/2}$ depends on $(f + g)$ instead f along. The coefficient of the numerical viscosity term Q is a function of a modified characteristic speed $\bar{a} + \bar{\gamma}$ instead of the characteristic speed $\bar{a}$ along, and the new numerical flux $\bar{f}_{i+1/2}$ can be rewritten as :

$$\bar{f}_{i+1/2} = \frac{1}{2}[(f_i + g_i) + (f_{i+1} + g_{i+1}) - Q(\bar{a}_{i+1/2} + \bar{\gamma}_{i+1/2})\Delta_{i+1/2}u] \qquad (2.2.13)$$

where g_i is an appropriately chosen function of the $\bar{a}_{i+1/2}$ and $\Delta_{i+1/2}u$, and

$$\begin{aligned} \bar{\gamma}_{i+1/2} &= (g_{i+1} - g_i)/\Delta_{i+1/2}u \qquad & \Delta_{i+1/2}u \neq 0 \\ &= 0 & \Delta_{i+1/2}u = 0 \end{aligned} \qquad (2.2.14)$$

The requirements on g are: (i) The function g should have a bounded γ in (2.2.14), so that the scheme is TVD with respect to the modified flux $(f + g)$. (ii) The modified scheme should be second-order accurate (except at points of extrema). One recipe for g (Harten, 1983) which satisfies the above requirements can be written as:

$$g_i = S \cdot max[0, min(\sigma_{i+1/2}|\Delta_{i+1/2}u|, S \cdot \sigma_{i-1/2}\Delta_{i-1/2}u)] \qquad (2.2.15)$$

$$S = sign(\Delta_{i+1/2}u) \qquad (2.2.15a)$$

with $\sigma_{i+1/2} = \sigma(\bar{a}_{i+1/2})$, and we choose

$$\begin{aligned} \sigma(x) &= \frac{1}{2}[Q(x) - Q_{LW}(x)] \\ &= \frac{1}{2}[Q(x) - \lambda x^2] \geq 0 \end{aligned} \tag{2.2.16}$$

for time accurate calculations. With this choice of $\sigma(x)$, the scheme is second-order accurate in both time and space (Harten 1983). It is noticed that this scheme is actually a nonlinear 5-point scheme even when it is applied to linear PDEs.

It is an interesting observation that at points of extrema we have, for example,

$$u_{i-1} \leq u_i = u_{i+1} \geq u_{i+2}$$

This leads to $g_i = g_{i+1} = 0$ and the numerical flux (2.2.13) becomes identical to that of the original first-order scheme. This shows that the scheme is second order accurate at smooth region of the solution and first-order accurate at points of extrema, and the specially designed function g, with the property of switching the second-order scheme into first-order in the vicinity of shock, is sometimes referred to as *limiter*. A more general description of limiters can be found in Sweby (1984).

§2.3 Riemann Solver

The concept of upwind can be traced back to Courant when he and his colleagues devised their scheme in 1950's making use of the characteristics. They proposed solving certain equations along the characteristics going back from the point (x_i, t_{n+1}). To evaluate the characteristic variables at time t_n, this method uses interpolation based on the two nearest grid values, which are (u_{j-1}^n, u_j^n) or (u_j^n, u_{j+1}^n) depending on whether the corresponding characteristic speed is positive or negative. This method is not a good method for a problem involving shocks, since it is not in conservation form; and it is still first order accurate.

Godunov devised a method based on solving a set of Riemann problems. His Riemann solver forms the basis of most modern shock-capturing method because it can be cast into conservative form while keeping the characteristic structure. The details of Godunov method can be found in Rychtmyer and Morton (1967), Holt (1977), Roe (1986).

The Riemann problem for a set of conservation laws is that the initial data are prescribed as two semi-infinite states ($u = u_L$ for $x < 0, u = u_R$ for $x > 0$). Consider the solution of the initial value problem

$$\frac{\partial u}{\partial t} + \frac{\partial f}{\partial x} = 0 \tag{2.3.1}$$

$$u(x,0) = u_L \qquad -\infty < x < 0$$

$$u(x,0) = u_R \qquad 0 < x < \infty \tag{2.3.1a}$$

There are three waves for the Euler equations; the inner one is a contact discontinuity separating states at different temperature, and the outer ones may be shock waves or rarefaction fans. The exact solution, called *similarity solution*, of this problem involves only algebraic equations. The basic idea of Godunov is to introduce the characteristic information by successive sequence of local exact solutions (Fig. 2.2).

The first step of Godunov method is an average step. Let U_i^n be the average state over grid interval between $(i + 1/2)\Delta x$ and $(i - 1/2)\Delta x$ at $n\Delta t$, in which the state is uniform using

$$U_i^n = \frac{1}{\Delta x} \int_{(i-1/2)\Delta x}^{(i+1/2)\Delta x} u(x, n\Delta t) dx \tag{2.3.2}$$

For each interface $(i + 1/2)\Delta x$ we can solve the Riemann problem with $u_L = U_i^n, u_R = U_{i+1}^n$. This step is an evolution step and this gives an exact solution to the approximate problem from $n\Delta t$ to $n + 1\Delta t$, provided Δt small enough that the waves from neighboring interfaces do not intersect. Using the exact solution of state variables at the interface $(i + 1/2)\Delta x$ and $(i - 1/2)\Delta x$ we can integrate the Euler equation using the following conservation law,

$$\int_a^b u(x,t_2)dx = \int_a^b u(x,t_1)dx - [\int_{t_1}^{t_2} f(u(b,t))dt - \int_{t_1}^{t_2} f(u(a,t))dt] \qquad (2.3.3)$$

where

$$a = (i-1/2)\Delta x, \quad b = (i+1/2)\Delta x$$

$$t_1 = n\Delta t, \quad t_2 = (n+1)\Delta t$$

The solution at $(n+1)\Delta t$ can be again approximated by a piecewise-uniform distribution using (2.3.2). So (2.3.3) can be written as

$$U_i^{n+1} = U_i^n - \frac{\Delta t}{\Delta x}[\bar{F}(U_i^n, U_{i+1}^n) - \bar{F}(U_{i-1}^n, U_i^n)] \qquad (2.3.4)$$

where

$$\bar{F}(U_i^n, U_{i+1}^n) = \frac{1}{\Delta t}\int_{t_1}^{t_2} f(u(x_{i+1/2}, t))dt \qquad (2.3.4a)$$

This shows that the Godunov method can be written in conservative form. In practical computation the time step Δt should be small enough to keep the neighboring Riemann problems from interacting because the exact solution of interacting Riemann problem is very complicated. This is also assured by the CFL condition. Then the process can be repeated.

The *upwind* property of Godunov method can be seen that, for example, if the characteristic speeds at the interface $x_{i+1/2}$ are all positive (or negative) then the Godunov flux $\bar{F}(U_i^n, U_{i+1}^n)$ becomes $f(U_i^n)$ (or $f(U_{i+1}^n)$). In the case of mixture characteristic speeds, more work should be done to distinguish the direction of the *wind*. In practical calculations Godunov's *exact Riemann solver* is relatively expensive because of the need of iteration to obtain the pressure at interface of cells. Further more, some information will be lost due to the average step of (2.3.2). This makes the exact solution seeming unnecessary. Some simplifications have been introduced to reduce the work. This theory is called *approximate Riemann solver*. The most widely used approximate Riemann solvers are due to Roe (1981) and Osher (1981). Here we only consider the Roe's scheme for 1-D Euler equations.

Consider 1D Euler equations in matrix form as,

$$\frac{\partial u}{\partial t} + A\frac{\partial u}{\partial x} = 0 \tag{2.3.5}$$

with the initial value of

$$u(x,0) = u_L \qquad -\infty < x < 0$$

$$u(x,0) = u_R \qquad 0 < x < +\infty \tag{2.3.5.a}$$

where A is *Jacobian* $A = \partial f/\partial u$

Roe suggested a linearization technique using a constant *Jacobian* $\tilde{A}$ instead of *Jacobian* A. To construct the constant *Jacobian* $\tilde{A}(u_L,\ u_R)$ we require it possess following properties ,which is called, by Roe, *property U*.

(1) $\tilde{A}(u,\ u) = A(u)$

(2) $\tilde{A}(u_L,\ u_R)(u_R - u_L) = f(u_R) - f(u_L)$

(3) $\tilde{A}(u_L,\ u_R)$ has real eigenvalues with linearly independent eigenvectors.

Roe explained that (2) is a sufficient condition for the algorithm to be conservative. (2) and (3) are both necessary and sufficient conditions for the algorithm to 'recognize' a shock wave. It is seen that if $u_L,\ u_R$ satisfy the jump condition

$$f_R - f_L = S(u_R - u_L)$$

for some scalar S, then by (2) S is an eigenvalue of $\tilde{A}$. A projection of $(u_L,\ u_R)$ onto the eigenvectors of $\tilde{A}$ will (because of (3)) be solely onto the eigenvector which corresponds to S. In this special case, the solution of the Riemann problem will be exact. It is seen that (1) guarantees that the method behaves well on smooth solutions.

The potential candidates for $\tilde{A}$ are

$$\tilde{A} = \frac{1}{2}(A_L + A_R) \tag{2.3.6a}$$

or

$$\tilde{A} = A(\frac{1}{2}(u_L + u_R)) \tag{2.3.6b}$$

But (2.3.6) can not completely satisfy the *property U*. Roe constructed it by introducing a parameter vector w based on an observation that the column vectors u and f can be expressed as quadratic functions of the variable w. The detailed derivation can be found in Roe's original paper (1981), or Hirsch (1990)

It is found that matrix $\tilde{A}$ is identical to to the local Jacobian given by eq. (2.3.5), when expressed as a function of the variables v and H, if these variables are replaced by an average called *Roe's Average*.

$$\tilde{\rho} = \sqrt{\rho_L \rho_R} \tag{2.3.7a}$$

$$\tilde{v} = \frac{(v_L + v_R\sqrt{\rho_R/\rho_L})}{1 + \sqrt{\rho_R/\rho_L}} \tag{2.3.7b}$$

$$\tilde{h}_0 = \frac{(h_{0L} + h_{0R}\sqrt{\rho_R/\rho_L})}{1 + \sqrt{\rho_R/\rho_L}} \tag{2.3.7c}$$

$$\tilde{a}^2 = (\gamma - 1)(\tilde{h}_0 - 0.5\tilde{v}^2) \tag{2.3.7d}$$

After the construction of $\tilde{A}$ Roe solves a piecewise linear problem

$$\frac{\partial u}{\partial t} + \tilde{A}\frac{\partial u}{\partial x} = 0 \tag{2.3.8}$$

within each cell $(i,\ i+1)$, with $\tilde{A} = \tilde{A}(U_i, U_{i+1})$. The numerical flux for this local approximate Riemann problem is easily constructed using linear eigenvalue expansion

$$\bar{F}_{i+1/2} = \frac{1}{2}(f_i + f_{i+1}) - \frac{1}{2}|\tilde{A}|(U_{i+1} - U_i) \tag{2.3.9}$$

§2.4 Construction of 2nd Order TVD Scheme for 2D Euler Equations

We now present the method to construct a second order accurate TVD scheme to solve the Euler equations on arbitrary quadrilateral mesh.

The Inviscid Gas dynamics Equation in two spatial dimensions has the integral form:

$$\frac{d}{dt}\iint_{\Omega} U ds + \oint_{\partial\Omega} \vec{F}(U)\cdot\vec{n}dl = 0 \tag{2.4.1}$$

where

$$\vec{F}(U) = f(U)\vec{i} + g(U)\vec{j} \qquad \vec{n} = n_x\vec{i} + n_y\vec{j} \tag{2.4.1a}$$

where U is the vector of dependent variables, and $\vec{F}$ is the convective flux vector.

$$U = \begin{pmatrix} \rho \\ \rho u \\ \rho v \\ \rho E \end{pmatrix} \qquad f(U) = \begin{pmatrix} \rho u \\ \rho uu + p \\ \rho uv \\ \rho uH \end{pmatrix} \qquad g(U) = \begin{pmatrix} \rho v \\ \rho vu \\ \rho vv + p \\ \rho vH \end{pmatrix} \tag{2.4.1b}$$

$$E = \frac{1}{\gamma - 1}\frac{p}{\rho} + \frac{1}{2}(uu + vv), \qquad H = E + \frac{p}{\rho} \tag{2.4.1c}$$

$$\begin{aligned} F(U) &= \vec{F}(U)\cdot\vec{n} \\ &= f(U)n_x + g(U)n_y \end{aligned} \tag{2.4.1d}$$

Equation (2.4.1) is descretized in space using the cell-centered finite-volume method suggested by Jameson & Mavriplis (Jameson and Mavriplis, 1986)

For the x momentum equation, it yields

$$\frac{\partial}{\partial t}(\Omega_{i,j}(\rho u)_{i,j}) + \sum_{k=1}^{4}[V_k(\rho u)_k + pn_x]\Delta l_k = 0 \tag{2.4.2}$$

where

$$\begin{aligned} V &= \vec{V}\cdot\vec{n} \\ &= un_x + vn_y \end{aligned} \tag{2.4.2a}$$

$$n_x = \frac{\Delta y_k}{\Delta l_k}, \qquad n_y = -\frac{\Delta x_k}{\Delta l_k} \tag{2.4.2b}$$

$$\Delta l_k = \sqrt{\Delta x^2 + \Delta y^2} \tag{2.4.2c}$$

It can be seen that the velocity V is normal to the cell face.

$$\begin{aligned} A &= \frac{\partial F}{\partial U} \\ &= \frac{\partial f}{\partial U}n_x + \frac{\partial g}{\partial U}n_y \end{aligned} \tag{2.4.3}$$

The eigenvalue of A are

$$\begin{aligned}\lambda_{1,2} &= V \\ \lambda_3 &= V + c\sqrt{n_x^2 + n_y^2} \\ \lambda_4 &= V - c\sqrt{n_x^2 + n_y^2}\end{aligned} \tag{2.4.4}$$

Because most existing upwind schemes for multidimensional problems assume that the conserved quantities are exchanged between two cells is by means of traveling waves normal to the interface (Roe, 1986) we can solve a 1D Riemann problem in the normal direction of the interface and the velocities parallel to the interface are neglected (Fig. 2.3). So the numerical flux for the Roe's approximate Riemann solver can be written in the following form

$$\bar{F}_{i+1/2} = \frac{1}{2}(F_i + F_{i+1}) - \frac{1}{2}|\tilde{A}|(U_{i+1} - U_i) \tag{2.4.5}$$

where F and A are defined by (2.4.1d) and (2.4.3).

Now we extend Harten's 2nd order TVD scheme to Euler equations using Roe's technique that applying the scalar scheme to each of the characteristic variables.

Due to eigenvalues of A are all real a similarity transformation exist.

$$A = T\Lambda T^{-1} \tag{2.4.6}$$

where T, T^{-1} and Λ are all 4×4 matrixes for 2D case (and 5×5 matrixes for 3D case). The columns of T are the right eigenvectors of the Jacobian matrix A as

$$T = (R^1,\ R^2,\ R^3,\ R^4) \tag{2.4.6a}$$

The rows of T^{-1} are the left eigenvectors of the Jacobian matrix A.

$$T^{-1} = \begin{pmatrix} L^1 \\ L^2 \\ L^3 \\ L^4 \end{pmatrix} \tag{2.4.6b}$$

The detailed description of these matrixes can be found in appendix 1.

Then the characteristic variables W are obtained by

$$W = T^{-1}U, \qquad w^k = L^kU, \qquad 1 \leq k \leq 4 \tag{2.4.7a}$$

or

$$U = TW, \qquad U = \sum_{k=1}^{4} w^k R^k \tag{2.4.7b}$$

Because the Jacobian $\tilde{A}$ is locally constant the system of (2.3.8) can be decoupled into 4 scalar characteristic equations.

$$\frac{\partial w^k}{\partial t} + a^k \frac{\partial w^k}{\partial x} = 0 \qquad a^k = const \tag{2.4.8}$$

Let $U_{i+1/2} = U(U_i,\ U_{i+1})$ be an average of U_i and U_{i+1}, such as Roe's average. Yee and Harten (1987) showed that arithmetic average can be used in practical calculation to reduce the work. And let $a^k_{i+1/2}$, $R^k_{i+1/2}$, $L^k_{i+1/2}$ denote the respective quantities of related to $\tilde{A}$. We define

$$\Delta_{i+1/2}U = U_{i+1} - U_i, \qquad \alpha^k_{i+1/2} = w^k_{i+1} - w^k_i \tag{2.4.9}$$

so we have

$$\Delta_{i+1/2}U = T_{i+1/2}\alpha_{i+1/2}, \qquad \alpha_{i+1/2} = T^{-1}\Delta_{i+1/2}U \tag{2.4.10}$$

Using these notations, the numerical flux can be evaluated as following,

$$\bar{F}_{i+1/2} = \frac{1}{2}(F_i + F_{i+1} + T_{i+1/2}\Phi_{i+1/2}) \tag{2.4.11}$$

$$\phi^k_{i+1/2} = \frac{1}{2}Q(a^k_{i+1/2})(g^k_i + g^k_{i+1}) - Q(a^k_{i+1/2} + \gamma^k_{i+1/2})\alpha^k_{i+1/2} \tag{2.4.12}$$

$$g^k_i = S \cdot max[0,\ min(|\alpha^k_{i+1/2}|,\ S \cdot \alpha^k_{i-1/2})] \tag{2.4.13}$$

$$S = sign(\alpha^k_{i+1/2})$$

$$\gamma^k_{i+1/2} = \begin{cases} 1/2(\bar{a}^k_{i+1/2})(g^k_{i+1} - g^k_i)/\alpha^k_{i+1/2}, & \alpha^k_{i+1/2} \neq 0; \\ 0, & \alpha^k_{i+1/2} = 0. \end{cases} \tag{2.4.14}$$

With above numerical fluxes the residual vector can be evaluated using

$$\begin{aligned} R_{i,j} &= \bar{F}_{i+1/2,j}\Delta l_{i+1/2,j} - \bar{F}_{i-1/2,j}\Delta l_{i-1/2,j} \\ &+ \bar{F}_{i,j+1/2}\Delta l_{i,j+1/2} - \bar{F}_{i,j-1/2}\Delta l_{i,j-1/2} \end{aligned} \tag{2.4.15}$$

From above formulae it is seen that present TVD scheme can be viewed as a centered scheme plus a TVD dissipation term as follows.

$$R_{i,j} = R^{physics}_{i,j} + D^{TVD}_{i,j} \tag{2.4.16}$$

This provide some convenience for our implementation in that the physical flux and dissipation term can be calculated separately using different subroutine.

After the spatial discretization a multistage Runge-Kutta time-stepping scheme is used to integrate the discretized system in time.

$$
\begin{aligned}
U^{(0)} &= U^{n} \\
U^{(1)} &= U^{(0)} - \alpha^{(1)} \quad \Delta t R^{(0)} \\
&\vdots \\
U^{(q-1)} &= U^{(0)} - \alpha^{(q-1)} \Delta t R^{(q-2)} \\
U^{(q)} &= U^{(0)} - \qquad\quad \Delta t R^{(q-1)} \\
U^{n+1} &= U^{(q)}
\end{aligned}
\tag{2.4.17}
$$

Chapter 3. Multigrid Methods

The original idea of multigrid (MG) can be traced back to 1940s when Southwell used the solution on a coarse grid to provide a better initial guess for a fine grid iteration to reduce computation work. This most simple multigrid, sometimes called *sequential grid method*, is still used by some researchers (e.g. Pulliam and Steger, 1985). This method is very simple but does not establish an interaction between coarse grid and fine grid. In 1960s Fedorenko devised the first multigrid method to solve elliptic partial differential equation (PDE) numerically based on the observation that the iteration on coarse grid has a high convergence rate, while on a fine grid the convergence rate is slower and more work should be done, but small truncation error is retained.

In 1974, Brandt (1974) published his landmark paper to establish a mathematical theory of multigrid method. Using a Fourier analysis he showed that, if a point iteration is employed, on a fine grid the high frequency components of the solution can be damped efficiently but the efficiency of damping low frequency components is very poor.

On the other hand, the low frequency components on a fine grid are relatively high frequency components on a coarse grid (Fig. 3.1 and 3.2), on which they can be damped rapidly when using the same iterative scheme on the fine grid. In a practical computation a sequence of successively coarser grids, on each of which approximations of solution of the PDE are defined, are introduced. These different approximations to the exact solution of the PDE interplay during the iteration procedure to reach a high convergence rate, reduce the computational work and keep the small truncation error. After the publication of Brandt's work, multigrid method became widely used in fluid flow calculations, in 1970s and 1980s, mainly to solve the potential equation (Jameson, 1979).

Although Brandt's theory depends on the assumption of ellipticity of the partial differential equation, from the point of view of signal propagation it can be seen that when we solve a hyperbolic equation numerically, under the same Courant number, the signals on a fine grid travel slower than those on a coarse grid within one time step. This made Ni (1981) to apply multigrid to hyperbolic Euler calculation to accelerate his cell-vertex finite volume Euler solver. Then Jameson (1983) incorporated multigrid to his cell-centered finite volume scheme. Denton (1983) also devised a multigrid method for his well-known time marching finite volume scheme. Davis et. al. (1986) extended Ni's multigrid method to Navier-Stokes calculation while Martinelli (1989) successfully validated Jameson's MG technique in viscous computation to fulfill his Ph.D. dissertation. Multigrid method has also been applied to 3D problems by many researchers for both external and internal, inviscid and viscous flows (Jameson et. al., 1986, Ni and Bogagian., 1989, Vasta and Wendan, 1990, Arnone et. al., 1993). In the past decade multigrid method gradually became a standard technique in engineering calculation of steady flows. In 1993 He (1993) successfully extended Denton's multigrid to unsteady viscous flow calculations.

In the present study, Jameson's multigrid method was implemented in our cell-centered finite volume TVD code to accelerate to convergence.

First, a sequence of successively coarser grids numbered from 1 to M are introduced. A coarser grid of number m, $(1 < m \leq M)$ is obtained by doubling the grid lines of a finer grid of level $(m-1)$. A numerical solution is defined on the finest grid. After some Euler calculations on the finest grid the dependent variables are transferred to coarser grid using an area-weighted average as follows,

$$U_{m+1} = \frac{\Sigma U_m \Omega_m}{\Sigma \Omega_m} \tag{3.1}$$

This makes the conservation of mass, momentum and energy. The flux residuals on coarser grid are also collected from the finer grid using the formula as follows,

$$R_{m+1} = \Sigma R_m \tag{3.2}$$

Suggested by Jameson (1983) and Jameson & Mavripolis (1986), a forcing function is defined as,

$$P_{m+1} = \Sigma R_m - R_{m+1}(U^{(0)}_{m+1}) \tag{3.3}$$

Then the modified residuals for the multistage scheme are,

$$R^{(q)}_{m+1} = R_{m+1}(U^{(q)}_{m+1}) + P_{m+1} \tag{3.4}$$

This modified residual should be collected in practical calculation. So the Euler calculation can be performed on a coarse grid using a multistage time stepping scheme.

$$\begin{aligned}
U^{(0)}_{m+1} &= U^{n}_{m+1} \\
U^{(1)}_{m+1} &= U^{(0)}_{m+1} - \alpha^{(1)} \quad \Delta t_{m+1} R^{(0)}_{m+1} \\
&\vdots \\
U^{(q-1)}_{m+1} &= U^{(0)}_{m+1} - \alpha^{(q-1)} \Delta t_{m+1} R^{(q-2)}_{m+1} \\
U^{(q)}_{m+1} &= U^{(0)}_{m+1} - \qquad \Delta t_{m+1} R^{(q-1)}_{m+1} \\
U^{n+1}_{m+1} &= U^{(q)}_{m+1}
\end{aligned}$$

It can be seen that in the first stage of the scheme the residual of $R^{(0)}_{m+1}$ is replaced by ΣR_m. This result means that the evolution on the coarse grid is driven by the residuals on the fine grid.

The above process is repeated until the coarsest grid level is reached. The correction calculated on each grid is passed back to the next fine grid using the following formula:

$$U^{new}_{m} = U^{old}_{m} + I^{m+1}_{m}(U^{new}_{m+1} - U^{(0)}_{m+1}) \tag{3.5}$$

where U^{old}_m is the solution on grid m after the time step and before the transfer from grid $(m+1)$ and I^{m+1}_m is an interpolation operator. In the present study both piecewise constant interpolation and bilinear interpolation are employed.

The boundary condition on each grid can be calculated in the same way as that on the finest grid. Suggested by Jameson (1983), since the evolution on coarse grid is driven by the residuals collected from the next finer grid, the final solution on the finest grid is independent of the choice of boundary condition on the coarse grid. So the boundary condition on coarse grid can be transferred from the finer grid.

It is important to choose a multistage scheme with a superior high-frequency damping property. In the present study a 3-stage time stepping scheme (Martinelli, 1987) is employed with coefficient of 0.6, 0.6, 1.0 and the dissipation is evaluated at each stage.

A simple sawtooth cycle (Fig. 3.3) is used and more efficient W-cycle (Fig. 3.4) will be tried in the future studies.

It is can be seen from the above description, under the assumption of fixed Courant number on all grid levels and the sawtooth cycle strategy, that in each cycle the total time that the solution evolves is

$$\Delta t_{tot} = (1+2+4+8)\Delta t_f = 15\Delta t_f$$

where Δt_f is the time step on the finest grid, and the total computational work is

$$\begin{aligned} W_{tot} &= (1+\frac{1}{4}+\frac{1}{16}+\frac{1}{64})W_f \\ &= \frac{85}{64}W_f \end{aligned}$$

where W_f is the work to advance the solution on the finest grid using Δt_f. In addition to a little more computational work of transferring dependent variables and residuals from coarse grid to fine grid and the interpolation of the correction from fine grid to coarse grid, it is seen that a lot of computational work is saved when the multigrid method is employed.

Chapter 4. Implicit Multigrid Method

Although explicit schemes have been widely used in time accurate calculations, such as multistage Runge-Kutta (e.g. He 1989) and Lax-Wendroff (e.g. Giles 1987), it is well known that explicit stepping is restricted by the stable condition which is well known as CFL condition. Especially in the viscous calculations the time step is extremely limited by the smallest scale of computational grid in the thin shear layer. He (1993) devised a two grid marching method to accelerate the calculation but the time accuracy is deteriorated due to the use of coarse mesh.

On the other hand, when an implicit time discretization is used the stability restriction is relaxed and large time step can be used.

We start from the differential form of the Euler equation

$$\frac{\partial U}{\partial t} + \frac{\partial F}{\partial x} + \frac{\partial G}{\partial y} = 0 \tag{4.1}$$

For simplicity, we use the first order backward Euler scheme to discretize the time derivative. Then (4.1) becomes to

$$\frac{U^{n+1} - U^n}{\Delta t} + R^{n+1} = 0 \tag{4.2}$$

where R is the residual vector defined as follows,

$$R = \frac{\partial F}{\partial x} + \frac{\partial G}{\partial y} \tag{4.3}$$

This implicit time discretization leads to a large nonlinear system (4.2) and R^{n+1} can not be evaluated explicitly using U^n.

The dominant technique to solve this system in the past decades is the local linearization as follows

$$F^{n+1} = F^n + (\frac{\partial F}{\partial U})^n \Delta U + O(\Delta t^2) \qquad (4.4a)$$

$$G^{n+1} = G^n + (\frac{\partial G}{\partial U})^n \Delta U + O(\Delta t^2) \qquad (4.4b)$$

where $\Delta U = U^{n+1} - U^n$. and $A,\ B$ is called *Jacobian* defined as,

$$A = \frac{\partial F}{\partial U}, \quad B = \frac{\partial G}{\partial U} \qquad (4.5)$$

Substitute (4.4) into (4.2) and drop the small terms then we obtain

$$[I + \Delta t(\partial_x A + \partial_y B)^n]\Delta U = -\Delta t(\partial_x F + \partial_y G)^n = -\Delta t R^n \qquad (4.6)$$

Eq. (4.6), referred as to *Deltaform*, is preferred by CFD community because it possesses some apparent physical significance. The right hand side, called *physical part* by MacCormack (1985), is the residual vector which is evaluated explicitly. When a large time step is used the explicit time stepping is unstable. So the implicit operator $\partial_x A + \partial_y B$, without which (4.6) reduce to an explicit scheme, is introduced in the left hand side to stablize the calculation.

The strategy for solving the linerized system (4.6) normally falls into two categories, one is iteration method, for example, Gauss-Siedel (MacCormack, 1985) and Newton method (Bailey and Beam, 1991), the other is factorization, such as the ADI method (Beam and Warming, 1978) and LU decomposition due to Jameson and Turkel (1981). The Newton method (when the time step tends to infinity $\Delta t \to \infty$) seems to lack time accuracy and is not suitable for unsteady calculations. The noniterative factorization simplifies the linearized system (4.6) into some standard structure such as triblock and lower-upper matrixes which can be inverted by some well-documented techniques. But when large time step is used the time accuracy will be lost due to the factorization error.

Another natural choice is to solve the nonlinear system (4.2) using some iterative techniques. Jameson (1991) devised an iterative scheme by introducing a pseudo time variable.

Now we consider the semi-discretized integral form Euler equation,

$$\frac{\partial(\Omega U)}{\partial t} + R^{n+1} = 0 \tag{4.7}$$

where R is the residual vector after the spatial discretization.

In the present study, a second order accurate, unconditionally stable, 3-point backward difference is used to discretize the time derivative. So (4.7) becomes

$$\begin{aligned} \frac{3}{2\Delta t}(\Omega_{i,j}U_{i,j})^{n+1} + R_{i,j}^{n+1} &= \frac{1}{2\Delta t}[\ 4(\Omega_{i,j}U_{i,j})^{n} - (\Omega_{i,j}U_{i,j})^{n-1}\ ] \\ &= S(U_{i,j}^{n},\ U_{i,j}^{n-1}) \end{aligned} \tag{4.8}$$

Since eq. (4.8) is a large nonlinear system an iteration must be devised to obtain the conservative variables and the residual vector simultaneously. Following the suggestion of Jameson (1991) a pseudo time is introduced and the governing equation is reformulated as

$$\frac{\partial U}{\partial t^*} + R^*(U) = 0 \tag{4.9}$$

where the modified residual R^* is

$$R^*(U) = \frac{3}{2\Delta t}U + \frac{1}{\Omega_{i,j}^{n+1}}[R(U) - S(U^n,\ U^{n-1})] \tag{4.10}$$

A pseudo time marching technique is used here. At each physical time step, for example, $n\Delta t$ the system (4.9) is advanced explicitly in the pseudo time using the multistage time stepping scheme as,

$$\begin{aligned}
U^{(0)} &= U^{n} \\
U^{(1)} &= U^{(0)} - \alpha_1 \quad \Delta t^* R^*(U^{(0)}) \\
&\vdots \\
U^{(q-1)} &= U^{(0)} - \alpha_{q-1} \Delta t^* R^*(U^{(q-2)}) \\
U^{(q)} &= U^{(0)} - \qquad \Delta t^* R^*(U^{(q-1)}) \\
U^{n+1} &= U^{q}
\end{aligned} \tag{4.11}$$

and the conservative variables are updated at each pseudo time step. When steady state or convergence is reached for system (4.9) the implicit system (4.8) at time $(n+1)\Delta t$ is satisfied. It is noted that $S(U^{(n)}, U^{(n-1)})$ is the source term and is fixed in the pseudo time marching from $n\Delta t$ to $(n+1)\Delta t$.

Nevertheless, the above mentioned explicit procedure for solving eq. (4.9) will be extremely expensive unless some efficient acceleration technique can be incorporated. It is can be seen that in the iteration procedure of (4.9) the residual vector $R(U)$ has to be reevaluated in every pseudo time step. This can be compared to the linerized version of (4.6), in which residual is calculated only once. So more work will be done in every physical time step. This will become extremely severe in NS calculation, in which the evaluation of the viscous terms will make the expense increased rapidly. The other reason is that explicit stepping will also suffer from the restriction of time step in the sense that the explicit marching for eq (4.9) can be regarded as point-Jacob iteration and the time step is the relaxation parameter.

Fortunately, the accuracy of the solution of (4.8) is independent of the pseudo time step. So the multigrid method which is widely used in steady Euler and Navier-Stokes calculations can be incorporated and much of CPU time can be saved.

It is also noticed that it is a reasonable choice to march the solution in pseudo time implicitly. Reconsider system (4.2) and define

$$f(U) = \frac{U - U^n}{\Delta t} + \frac{\partial F(U)}{\partial x} + \frac{\partial G(U)}{\partial y} = 0 \tag{4.12}$$

Following the suggestion of MacCormack (1985), we have the derivative of f with respect to dependent variable U

$$f'(U) = I/\Delta t + \partial_x A + \partial_y B \tag{4.13}$$

Then apply the Newton-Raphson iteration to nonlinear system (4.12). We have

$$[I + \Delta t(\partial_x A + \partial_y B)^m](U^{m+1} - U^m) = (U^m - U^n) - \Delta t(\partial_x F + \partial_y G)^m \tag{4.14}$$

here m is the subiteration number.

It can be seen if the left hand side of (4.14) is convergent to zero the (4.14) is satisfied at physical $(n+1)$ time level. It is also found that (4.14) can be cast into the above-mentioned implicit *Delta form*. So some existing methodologies like factorization and relaxation can be applied to invert (4.14). Due to the implicit nature of Newton-Ralphson iteration, a few number of subiteration is needed. Rai (1987) first applied this implicit iteration to unsteady calculation for the flows in turbomachinery. The three-point backward difference formulae can also be used here to reach second order time accuracy (Pulliam, 1993). Pulliam (1993) reported that 6 $\sim$ 7 subiterations is needed to reach the desired accuracy.

Chapter 5. Implementation and Numerical Results

In the present study, some numerical work has been done. Firstly, we developed a basic code using cell-centered finite-volume spatial discretization, explicit multistage time stepping scheme, and TVD dissipation, which are detailed described in chapter 2. Then we incorporated the steady multigrid method, reported in chapter 3, into this code. Thirdly we developed a fully implicit time stepping code with the pseudotime-marching and the multigrid acceleration. Some numerical experiments were carried out to validate our code and numerical experience and results were obtained.

In this chapter, we first report some implementation details, including computational routine, the treatment of boundary conditions. Then some numerical results are showed .

§5.1 Implementation Issues

At first a basic code was written using the cell-centered finite-volume scheme, multistage time stepping and TVD dissipation term.

To evaluate the numerical fluxes, the program is organized in the following way,

(1) compute flux F_i using (2.3.5).

(2) compute quantities on the interface using Roe's average $(2.1.7a) - (2.1.7e)$.

(3) compute eigenvalues using $(a1.3)$.

(4) compute left eigenvectors $T^{-1}_{i+1/2}$ using $(a1.5)$.

(5) compute $\alpha_{i+1/2}$ at interface using (2.3.11).

(6) compute g_i using (2.3.15).

(7) compute $\gamma_{i+1/2}$ using (2.3.16).

(8) compute right eigenvectors $T_{i+1/2}$ using (a1.4).

(8) compute $\phi_{i+1/2}$ using (2.3.14).

(9) compute $\bar{F}_{i+1/2}$ using (2.3.13).

The Euler equations in both its continuous and discrete forms have to be supplemented with boundary condition to fully determine a solution. It is well known that, due to the use of discrete method and finite computational domain, most numerical methods for fluid dynamics problems require more boundary conditions than those required by the governing PDEs. These additional boundary conditions, which are called numerical boundary conditions, should be specified by some numerical techniques.

In 2D cascade calculations, usually four types of boundary conditions are encountered : wall, periodic, inlet, outlet. At the inlet boundary, all of the four dependent variables must be specified for supersonic inlet flows, or three of the four must be specified for subsonic inlet flows. Considering the actual experimental conditions we choose to specify the total temperature, total pressure, which is equivalent to entropy, and the flow angle of the incoming flow. The static pressure is specified or extrapolated from interior points. At the outlet boundary, in the case of supersonic outlet flow, all of the four dependent variables should be extrapolated from interior points, while in the case of subsonic outlet flow, the static pressure is specified and the other three quantities are obtained by extrapolation. Ghost cells are used in the present study to obtain the fluxes at the interface of $1/2,\ j$ and $im+1/2,\ j$. At the solid wall the tangential condition should be satisfied.

$$\vec{V}\cdot\vec{n}=0$$

Since the flow variables are stored at the center of each control volume (see Fig. 1.3), the tangency condition can not be applied directly. A integral form or sometimes referred to as weak form condition is adopted. This leads to the contribution to the fluxes only comes from the pressure on the surface of the solid wall. We simply let the convective flux be zero and the pressure on the wall is obtained by extrapolation based on the assumption that the normal pressure gradient on the wall is zero.

$$\frac{\partial p}{\partial n}_{|wall} = 0$$

The pressure can also be obtained using more sophisticated normal pressure momentum equation (Steger, 1978, Pulliam and Steger, 1985, Jameson and Mavriplis, 1986).

The current TVD scheme is a 5-point scheme in 1D and 9-point scheme in 2D. In the practical calculations, when we evaluate the dissipation terms at interfaces of 3/2 and $im - 1/2$ we can not compute g_1 and g_{im} using the formulae for the interior points. In the present study an extrapolation is used in the following way,

$$g_1 = g_2, \qquad g_{im} = g_{im-1}$$

Our numerical experiments show that this treatment is more stable than others.

After the evaluation of residual vector the system can be advanced in time using the multistage Runge-Kutta scheme. A 3-stage scheme, proposed by Jameson and Mavriplis (1986) is chosed since it has good high frequency damping property. In future study, a 5-stage scheme will also be considered. In the current computations, the disspation terms are evaluated in each stage so that the present code can be applied to not only steady but also unsteady flow problems without the loss of time accuracy.

In implementing the steady multigrid method, the problem is how to store the quantities of different grid levels. It can be seen from the interpolation formula (3.5) that after transferring the conservative variables from fine grid to coarse grid, the updated conservative variables on each grid should be stored for the use of interpolation. So a large 1D array is created in current code and a mapping subroutine is written to map the solution from the normally used 2D array to this

1D array and vise versa. Actually the use of 1D array saves a lot of computer memory in practical computation.

In the implicit program a 3-point backward time discretization is used. It is second order accurate in time and unconditionally stable. It is noticed that it need a special starting procedure. In the current calculation an explicit time stepping is used to advance the solution several steps in time using relatively small time step. Then we can advance the solution using the fully implicit scheme.

§5.2 Numerical Results

Case 1. Shock reflection problem

We first consider a regular shock reflection problem to test the code. Fig. 5.1 is an illustration of this problem and the computational domain with exact pressure solution. The incident shock angle is 29° and the free stream Mach number M_∞ is 2.9. The computation grid is 64 × 32 with uniform grid intervals. Three level multigrid is used. The computation is started from a uniform flow which is set equal to the freestream supersonic inflow values and upper boundary conditions are prescribed to produced the desired shock. The reflection boundary condition is specified on the rigid wall as

$$u_{i,0} = u_{i,1}, \qquad v_{i,0} = -v_{i,1}$$

The solution is assumed to reach steady state when the maximum absolute residual of continuity equation is less than 10^{-5} as

$$|\frac{\Delta\rho}{\Delta t}| \leq 10^{-5}$$

The present calculations are carried out on a HP series 700 work station of Durham University with double precision. The CPU time for this case is 11 min.

Fig.5.2 is the Mach contour. Fig. 5.3 is the pressure coefficient distribution evaluated at $y = 0.5$ and the definition of pressure coefficient is

$$C_p = \frac{2}{\gamma Ma_{\infty}^2}(\frac{p}{p_0} - 1)$$

Fig 5.2 shows that the computation gives the correct positions of both incident shock and reflected shock. The computed pressure solution is compared to the exact pressure solution as

Exact	0.714286	1.52819	2.93398
Computed	0.714286	1.52830	2.93315

Due to the use of the uniform Cartesian coordinate in this case, it is easy to see that the error is tolerated at the level of $(\Delta x^2,\ \Delta y^2)$, the truncation error of the numerical scheme used. It is also seen, from Fig. 5.3, that the shocks is oscillation free and a little smeared because the shock is not aligned with the mesh lines. The results show that our code can capture the shock with second order accuracy. Fig. 5.4 and Fig. 5.5 show the results of Yee et. al. (1983), and Zhang et. al. (1990) using their 2nd-order TVD finite difference methods.

Case 2. Supersonic flow past a bicircular arc cascade

Then a bicircular arc cascade is used to validate our program. The maximum thickness of the cascade is 4% of the chord. A supersonic case with inlet Mach number of 1.4 is adopted. Two H-type grids are used. One is a coarse grid with 64×32 cells. The other is a fine grid with 128×64 cells. To improve the accuracy of the solution, an exponential function is used to cluster the grid points near the rigid wall ,where the gradient of flow quantities are usually larger than that of other regions. The finest grid spacing is about 0.01 for the 64×32 grid and 0.005 for the 128×64 grid. Fig. 5.6 shows the 64×32 grid. On both grids the multigrid solver is tested. Currently the sawtooth cycle (Jameson, 1983) is implemented. One iteration corresponds one multigrid cycle.

The Initially the flow field is uniform with the upstream quantities and the tangency condition is applied on both upper and lower boundary. Then this boundary condition introduces a perturbation into the fluid field and makes the fluid field condition introduces a perturbation into the fluid field and makes the fluid field evolve to a new steady state. In this case, the convergence criterion is the same as that in case 1. Fig. 5.7 and Fig. 5.8 compare the convergence history for the residual of continuity equation with and without multigrid. It is seen from the convergence histories that the desired convergence rate of multigrid reported in chapter 3 has been reached. Since more grid levels can be used on the fine grid, the convergence history on the fine grid with 4 grid levels is almost the same as that on the coarse grid with 3 grid levels and this shows the grid-independen property of multigrid method. The fully implicit code is tested on the 64×32 grid with time step of 100 times of that for the explicit scheme. Convergence is reached in 68 steps. It is observed that at the initial steps quite a large number of pseudo time steps is needed and it becomes fewer and fewer when the steady state is reached. The conclusion is that the number of subiteration within each physical time step depends on the change of flow properties within this physical time step.

Fig. 5.9, Fig. 5.10, Fig. 5.11, Fig. 5.12 are the iso-pressure lines and iso-Mach lines computed using 64×32 grid and 128×64 grid respectively. It can be seen that two oblique shocks are formed at leading-edge and trailing-edge respectively and the convex surface of the cascade leads to the expansion wave in the channel. The reflected shock are presented clearly.

Fig. 5.13, Fig. 5.14, Fig. 5.15 and Fig. 5.16 are Mach number and pressure distribution along the cascade surface. They show that the shocks are captured in 1-2 grid-point. The reflected shock are captured in 3 grid-point because the shock does not align with the grid.

Fig. 5.17 is the result of Struijs et. al.(1989), the work of a joint group led by Deconink at VKI and Roe in Michigan. Fig.5.18 is the surface Mach number distribution by Ni (1981). It is seen from the comparison that the present method gives more sharp shocks than Ni's Lax-Wendroff method with artificial viscousity and has almost the same performance as the upwind method of Struijs et. al.

Case 3. Denton's supersonic wedge cascade

Denton (1982) proposed a supersonic wedge cascade with inlet Mach number of 2.0 to test the shock capturing ability and the exact solution of this problem is found by him. Fig. 5.19 is the illustration of this case and a computational grid with 64×32 cells. This case is designed that the leading edge shock is reflected from the lower blade and 'exactly canceled at the upstream corner, giving a uniform flow between the two parallel surfaces and an expansion off the downstream corner. ' (Denton, 1982).

Fig. 5.20 is the convergence history with 1, 2, 3 grid levels respectively. Fig. 5.21 is the Mach contour using this 64×32 grid. Fig. 5.22 shows the Mach contour on a grid with 128×32 cells. Fig 5.23 is the surface Mach number distribution on the blade with comparison to the exact solution. The box and circle are computed solution on upper and lower surface of the blade, respectively. The solid line is the exact solution. It is seen from Fig. 5.23 that the leading edge shock is captured within 1 grid point with slight post-shock oscillation, which is also found in Denton' result (Fig. 5.24). In Liu's result (Fig.5.25) this shock is smeared by a hefty amount of artificial viscosity. The reflected shock is captured without any oscillation . It is noted that the reflected shock from the lower blade should be canceled out at the corner of the upper blade. But the canceling is not perfect. This leads to the unexpected post-shock overshoot. These overshoot and undershoot are believed to be caused by the grid singularity that the large derivatives of the variables , which play as the coefficients of the truncation errors, lead to the large truncation errors. Recently, perfect canceling and oscillation free solution is reported by using the adaptive unstructured grid method, in which extremely fine grid points are clustered in the vicinity of these corners. It is also observed that the expansion and the trailing edge shock are captured both within one grid point without oscillation. This is much better than both Denton's and Liu's results. In the expansion region, the Mach number is under-predicted. This happened in both Denton's and Liu's results. It is also believed that this can be improved by using much finer grid.

Fig. 5.24 and Fig. 5.25 are the results by Denton himself (1982). and Liu (1991).

Case 4. Shock propagation in a nozzle

This case is an unsteady problem. Initially a supersonic shock propagates in a convergent-divergent nozzle from the left to the right with the shock Mach number

M_s of 3.0. The nozzle contours are formed using the sine curves. Due to the symmetry of the nozzle only the half the geometry was the used in the present computation with 350 × 82 cells. Fig. 5.26 to Fig. 5.40 is a sequence of density contours at different times and they record the evolution the the shocks in the nozzle. In Fig. 5.27 and Fig. 5.28, regular reflection are observed. Fig. 5.29 shows the Mach reflection. In Fig. 5.30–Fig. 5.34 , the reflected bow shocks collide with each other and passing through the each other. Then they proceed further and hit the nozzle walls. The incident Mach shock fronts merge into one propagating shock front and proceed down the nozzle. Due to multiple interactions of various discontinuities, the flow field becomes very complicated (Fig. 5.35–Fig. 5.40). In this case, the abilities of resolving the discontinuities and the accuracy in both space and time of the present method are rigorously tested.

Fig. 5.41 is the results from Yang (1990) using his more complicated UNO3 (Third-order Uniformly Non-Oscillation) scheme. The comparison shows that the evolutions simulated by the author and Yang respectively are almost identical.

Chapter 6. Conclusions

A numerical method for solving two-dimensional Euler equations is presented. The method makes use of cell-centered finite volume method. The dissipation terms is derived from the concept of upwinding and TVD and extended to the Euler equations by using Roe's approximate Riemann solver. The scheme is second-order accurate almost everywhere. The method is capable of giving sharp shock profiles without spurious oscillations and free of user-dependent and problem-dependent parameters. The method uses a multi-stage Runge-Kutta scheme for integration in time. An efficient multigrid method is incorporated to accelerate the convergence to steady state. Meanwhile, a fully implicit time integration method is coded with multigrid acceleration in the pseudotime marching.

The method has been particularly implemented to calculate internal flows and it can be applied to both steady and unsteady flows. Numerical experiments show that the solutions by the present method compare well with the exact solutions, a desired accuracy has been reached and the present method performs better than some existing numerical methods. The finite volume method also provides the author the flexibility to straightforwardly extend the present program to unstructured grid, with which the grid singularity problem encountered in the case of Denton's wedge cascade can be remedied.

In addition, it is found that there is an interesting connection between the central and the upwind differencing and that Jameson's second order central difference scheme plus an explicit adaptive artificial viscosity can be reinterpreted as a hybrid scheme composed of a weighted average of a first order upwind scheme and a second order upwind scheme. The weight is a conservative switch. It is of interest that further work can be done by using this idea.

Reference

Anderson, K. W., Thomas, J. L. and Van Leer, B., 1986, Comparison of Finite-Volume Flux-Vector Splitting for the Euler Equations, AIAA J., 24, pp. 1453-1460.

Arnone, A, Liou, M.-S., and Povinelli, L., A.,1993, Multigrid Calculation of Three-Dimensional Viscous Cascade Flows, J. of Propulsion and Power, Vol. 9, No. 4, pp. 605- 614.

Arnone, A., and Swanson, R., C., 1993, A Navier-Stokes Solver for Turbomachinery Applications, J. of Turbomachinery, Vol. 115, April, pp. 305-313.

Bailey, H., E., and Beam, R., M., 1991, Newton's Method Applied to Finite-Difference Approximations for the Steady-State Compressible Navier-Stokes Equations, J. of Comput. Phys., Vol. 93, pp. 108-127.

Beam, R., M., and Warming, R., F., 1978, An Implicit Factored Scheme for the Compressible Navier-Stokes Equations, AIAA J. , vol. 16, pp. 393-402.

Davis, R., L., Ni, R.-H., and Carter, J., E., 1986, Cascade Viscous Flow Analysis Using the Navier-Stokes Equations, AIAA paper, No. 86-0033.

Denton, J., D., 1975, A Time Marching Method for Two and Three Dimensional Blade to Blade Flows, R&M 3795, Aeronautical Research Council.

Denton, J., D., 1982, An Improved Time Marching Method for Turbomachinery Flow Calculations, in Roe ed., Numerical Methods in Aeronautical Fluids Dynamics, London, Academic Press.

Giles, M, 1988, Calculation of Unsteady Wake Rotor Interaction, J. of Propulsion and Power, Vol. 4, No. 4, pp. 356-362.

Giles, M., 1990, Stator/Rotor Interaction in a Transonic Turbine, J. of Propulsion and Power, Vol. 6, No. 5, pp 621-627.

Godunov, S. K., 1959, Math. Sb., 47, 271.

Harten, A., 1983, High Resolution Schemes for Hyperbolic Conservation Laws, J. of Comput. Phys., 49, 357-393.

He, L, 1990, An Euler Solution for Unsteady Flows Around Oscillating Blades, J. of Turbomachinery, Vol. 112, No. 4, pp. 714-722.

He, L, 1993, New Two-Grid Acceleration Method for Unsteady Navier-Stokes Calculations, J. of Propulsion and Power, Vol. 9, No. 2, pp272-280.

Hirsch, C., 1990, Numerical Computation of Internal and External Flows, Vol. 1: Fundamentals of Numerical Discretization, Chichester, John Wiley& Sons.

Hirsch, C., 1990, Numerical Computation of Internal and External Flows, Vol. 2: Computational Methods for Inviscid and Viscous Flows, Chichester, John Wiley& Sons.

Holt, M., 1977, Numerical Methods in Fluid Dynamics, New York, Springer-Verlag.

Jameson, A., 1979, Acceleration of Transonic Potential Flow Calculations on Arbitrary Meshes by the Multi-Gird Method, AIAA paper No. 79-1458.

Jameson, A., 1983, Solution of the Euler Equations for Two Dimensional Transonic Flow by a Multigrid Method, Applied Math. and Comput., Vol. 13, pp 327-355.

Jameson, A, 1991, Time Dependent Calculations Using Multigrid, with Applications to Unsteady Flows Past Airfoils and Wings, AIAA paper 91-1596.

Jameson, A., and Mavriplis, D, 1986, Finite Volume Solution of the Two-Dimensional Euler Equations on a Regular Triangular Mesh, AIAA J. 24, pp. 611 - 618.

Jameson, A., Schmidt, W., and Turkel, E., 1981, Numerical Solution of the

Euler Equations by Finite-Volume Methods Using Runge-Kutta Time-Stepping Schemes, AIAA paper No. 81-1259.

Jameson, A., and Turkel, E., 1981, Implicit Schemes and LU Decompositions, Math. of Comput., Vol. 37, No. 156, October, 1981, pp.385-397.

Liu, F., 1991, Numerical Calculation of Turbomachinery Cascade Flows, Ph. D. Thesis, MAE Dept. Princeton University.

MacCormack, R., W., 1985, Current Status of Numerical Solutions of the Navier-Stokes Equations, AIAA paper No. 85-0032.

Martinelli, L., 1987, Calculation of Viscous Flows with Multigrid Method. Ph. D. Thesis, MAE Dept. Princeton University.

Ni, R.-H., 1981, A Multiple Grid Scheme for Solving the Euler Equations, AIAA J., Vol. 20, No. 7, pp 1565-1571.

Ni, R.-H., and Bogoian, J, 1989, Prediction of 3-D Multi-Stage Turbine Flow Field Using a Multiple-Grid Euler Solver, AIAA paper No. 89-0203.

Osher, S., 1981, Numerical Solution of Singular Perturbation Problems and Hyperbolic Systems of Conservation Laws, North-Holland Math. Studies, 47, pp. 179-205.

Pulliam, T. H., 1993, Time Accuracy and the Use of Implicit Methods, AIAA paper No. 93-3360.

Pulliam, T. H., and Steger, J., L., 1985, Recent Improvements in Efficiency, Accuracy and Convergence for Implicit Approximate Factorization Algorithms. AIAA paper No. 85-0360.

Rai, M. M, 1987, Naiver-Stokes Simulation of Rotor/stator Interaction Using Patched and Overlaid Grid, J. of Propulsion and Power, Vol. 3, No. 5, pp387-396.

Richtmyer, R., D., and Morton, K.,W., 1967, Difference Methods for Initial Value Problems., 2nd Eds., New York, John Wiley & Sons.

Roe, P. L., 1981, Approximate Riemann Solvers, Parameter Vectors, and Difference Schemes, J. of Compt. Phys., 43, pp. 357-372.

Roe, P. L., 1986, Characteristic-Based Schemes for the Euler Equations, Ann. Rev. Fluid Mech. , 18, pp. 337-365.

Steger, J., L., 1978, Implicit Finite Difference Simulation of Flow about Two-dimensional Geometries, AIAA J., vol. 16, pp. 697-686.

Struijs, R., Deconinck, H., de Plama, P., Roe, P. and Powell, K. G., 1991, Progress on Multidimensional Upwind Euler Solvers for Unstructured Grid, AIAA paper No. 91-1550.

Sweby, P. K., 1984, High Resolution Schemes Using Flux Limiters for Hyperbolic Conservation Laws, SIAM J. Numer. Anal., 21, pp. 995-1011.

Vatsa, V., N., and Wedan, B., W., 1990, Development of A Multigrid Code for 3-D Navier-Stokes Equations and Its Application to a Grid-Refinement Study, Computer & Fluids, Vol. 18, No. 4, pp. 391-403.

Yang, J., Y., 1992, Private collection.

Yee, H., C., Harten, A., 1987, Implicit TVD Schemes for Hyperbolic Conservation Laws in Curvilinear Coordinates, AIAA J. , Vol. 25, No. 2, pp. 266-274.

Yee, H., C., Warming, R., F., Harten, A., 1983, Implicit Total Variation Diminishing Schemes for Steady-State Calculations, AIAA paper No. 83-1902.

Zhang, H., and Zhuang, F, 1991, NND Schemes and Their Applications to Numerical Simulation of Two- and Three-Dimensional Flows, in Advances in Applied Mechanics, vol. 29, pp193-256, Academic Press.

Appendix 1. Construction of Jacobian of Euler Equations

Let f and g are fluxes in x and y direction respectively. Consider a general form of flux as following,

$$\begin{aligned} F &= (fn_x + gn_y)\Delta l \\ &= f\Delta y - g\Delta x \end{aligned} \qquad (a1.1)$$

The Jacobian matrix of F have a set of eigenvalues and a complete distinct set of eigenvectors. Similarity transformation can be used.

$$A = T\Lambda T^{-1}$$

with

$$\Lambda = \begin{pmatrix} V & 0 & 0 & 0 \\ 0 & V & 0 & 0 \\ 0 & 0 & V + c\sqrt{\Delta x^2 + \Delta y^2} & 0 \\ 0 & 0 & 0 & V - c\sqrt{\Delta x^2 + \Delta y^2} \end{pmatrix}$$

where $V = u\Delta y - v\Delta x$, which is normal to the cell surface.

Then we have,

$$T =$$

$$\begin{pmatrix} 1 & 0 & \alpha & \alpha \\ u & \rho n_x & \alpha(u + cn_x) & \alpha(u - cn_x) \\ v & -\rho n_x & \alpha(v + cn_x) & \alpha(v - cn_x) \\ \phi^2/k & \rho(un_y - vn_x) & \alpha[(\phi^2 + c^2)/k + c\theta] & \alpha[(\phi^2 + c^2)/k - c\theta] \end{pmatrix}$$

$$T^{-1} =$$

$$\begin{pmatrix} 1-\theta^2/c^2 & ku/c^2 & kv/c^2 & -k/c^2 \\ -(un_y - vn_x)/\rho & n_y/\rho & -n_x/\rho & 0 \\ \beta(\phi^2 - c\theta) & \beta[cn_x - ku] & \beta[cn_y - kv] & \beta k \\ \beta(\phi^2 + c\theta) & -\beta[cn_x + ku] & -\beta[cn_y + kv] & \beta k \end{pmatrix}$$

with $\alpha = \rho/(\sqrt{2}c)$, $\beta = 1/(\sqrt{2}\rho c)$, $\theta = un_x + vn_y$, $k = \gamma - 1$ and

$$n_x = \frac{\Delta y}{\sqrt{\Delta x^2 + \Delta y^2}}, \quad n_y = \frac{-\Delta x}{\sqrt{\Delta x^2 + \Delta y^2}}$$

$$\phi^2 = \frac{1}{2}(\gamma - 1)(u^2 + v^2)$$

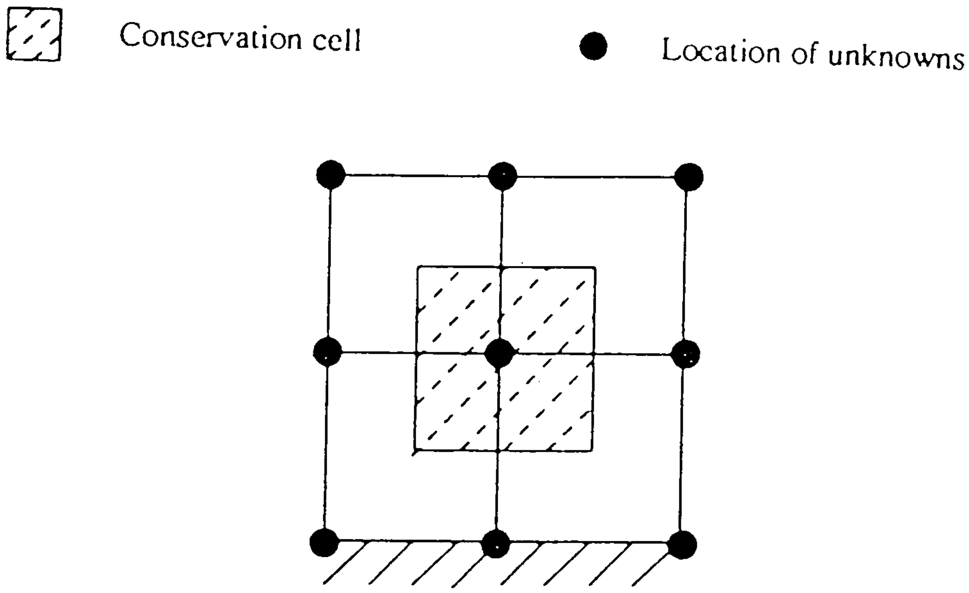


Figure 1.1: Grid for finite diference method.

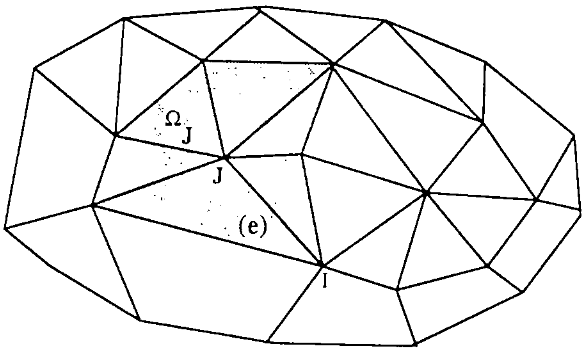


Figure 1.2: Grid for finite element method.

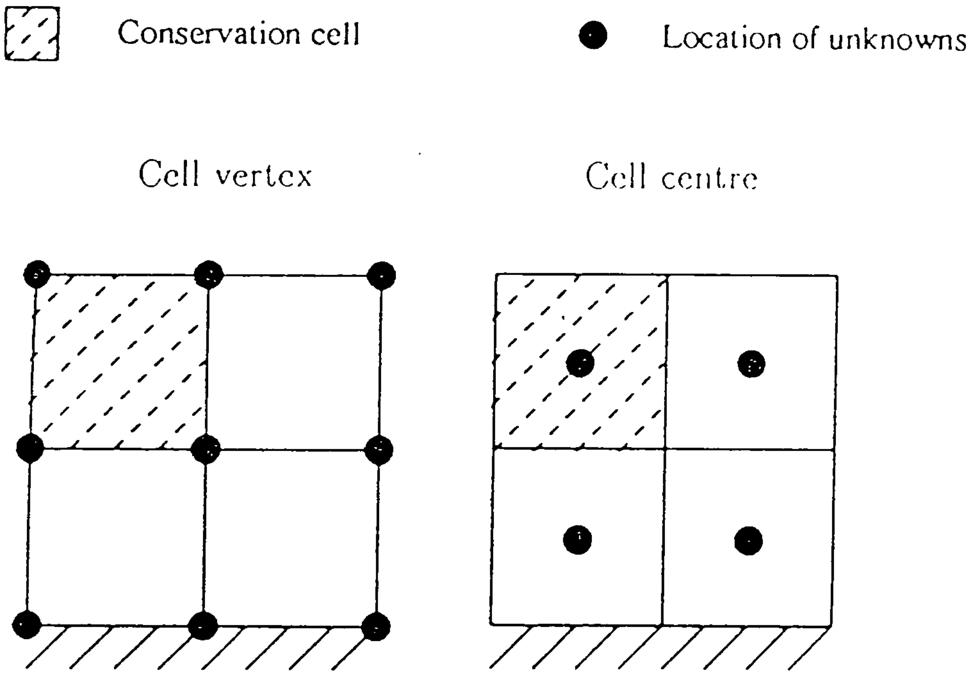


Figure 1.3: Grid for cell-centered and cell-vertex finite volume method.

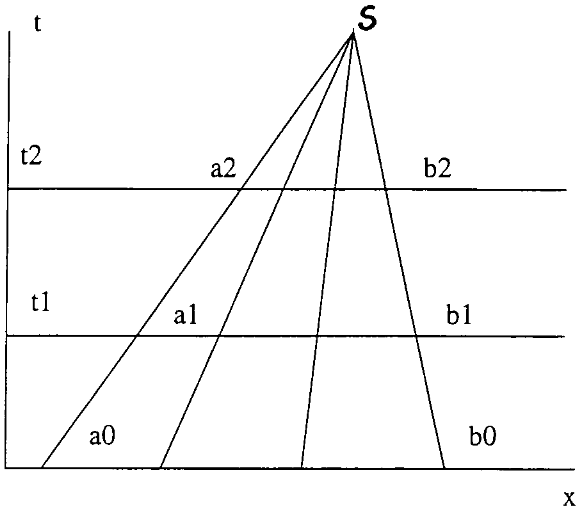


Figure 2.1: Intersection of compression shock discontinuity with characteristics

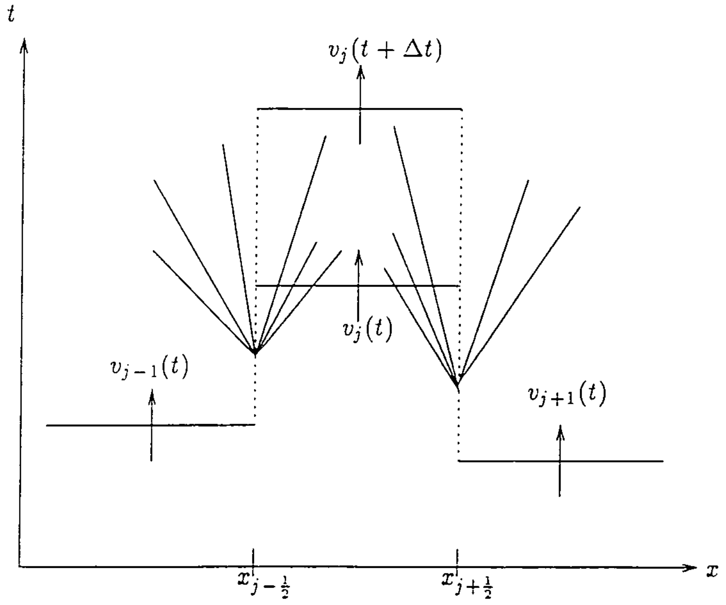


Figure 2.2: Illustration of Godunov's Riemann solver with piecewise constant grid function

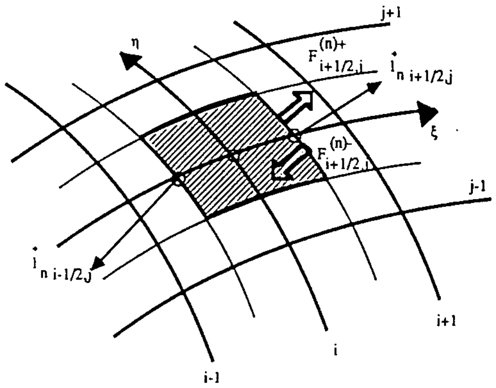


Figure 2.3: Two dimensional cells in arbitrary coordinates

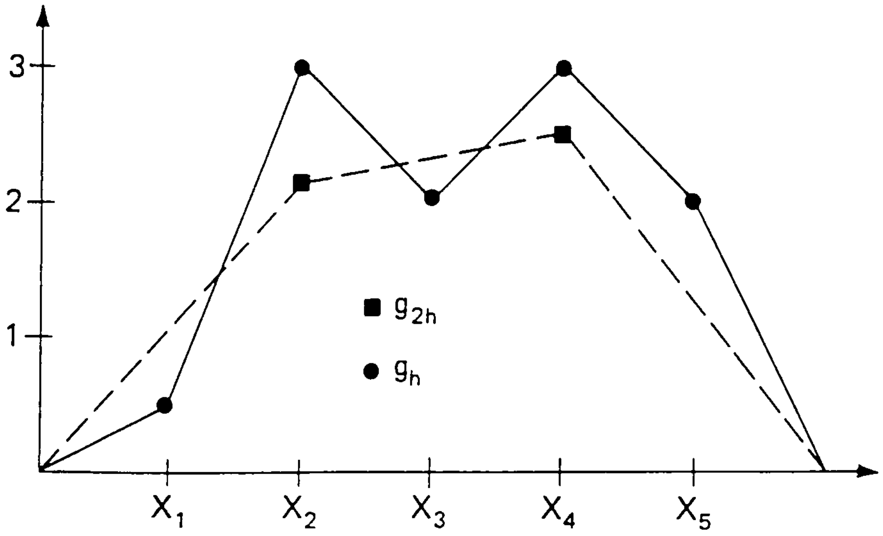


Figure 3.1: Illustation of transferring quantity from fine grid to coarse grid by averaging

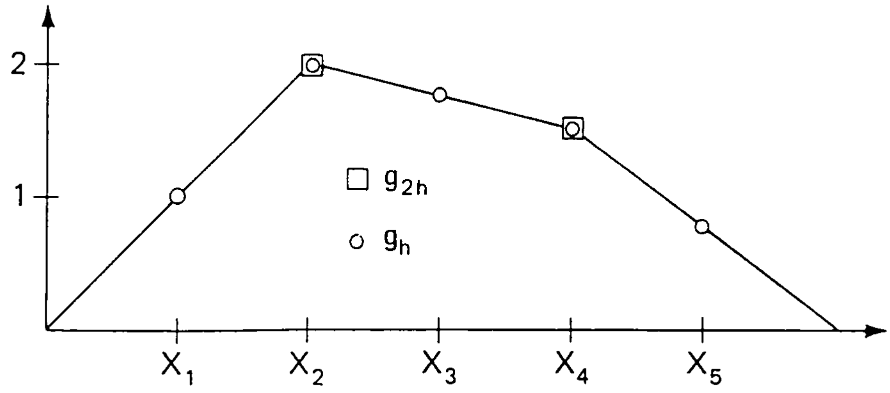


Figure 3.2: Illustration of transferring quantity from coarse grid to fine grid by interpolation

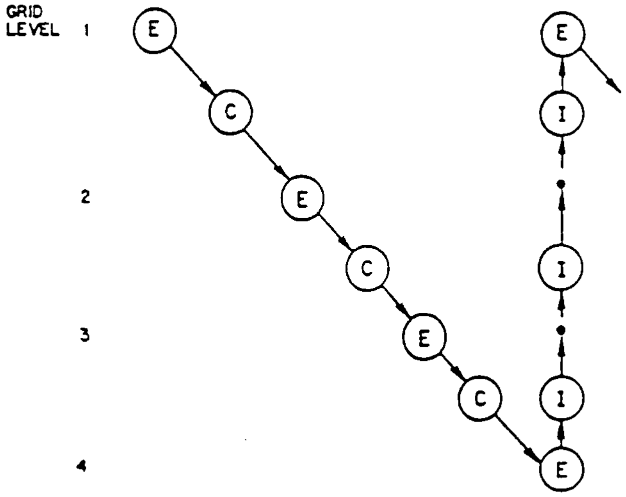


Figure 3.3: Multigrid Sawtooth cycle

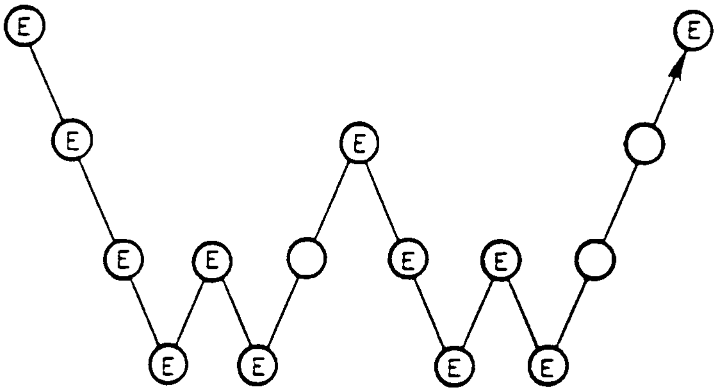


Figure 3.4: Multigrid W cycle

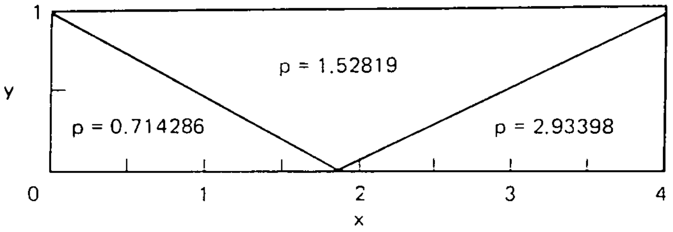


Figure 5.1: Illustration of the shock reflection problem and the exact solution

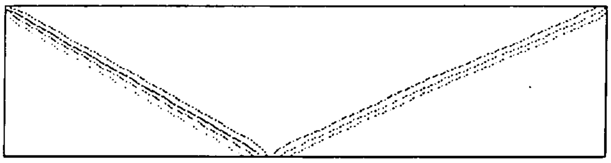


Figure 5.2: Iso-mach lines of the shock reflection problem.

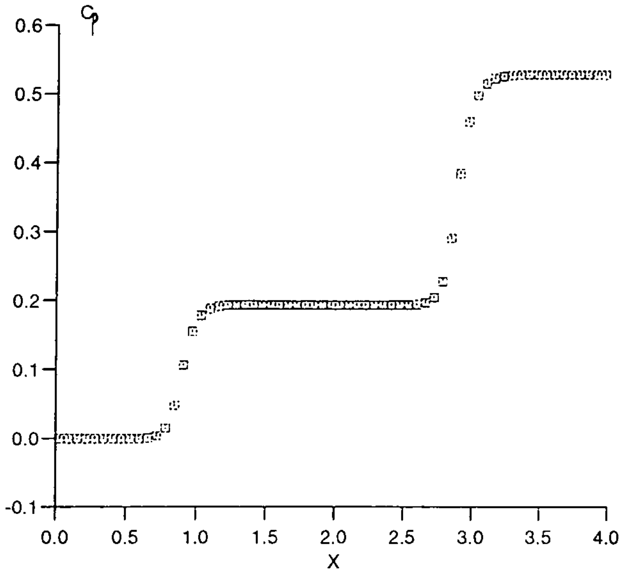


Figure 5.3: Pressure coefficient distribution evaluated at y=0.5

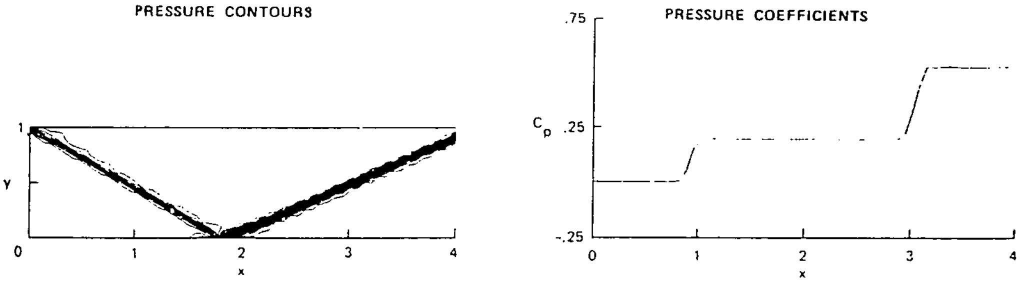


Figure 5.4: Result of the shock reflection problem calculated by Yee et. al., (1983).

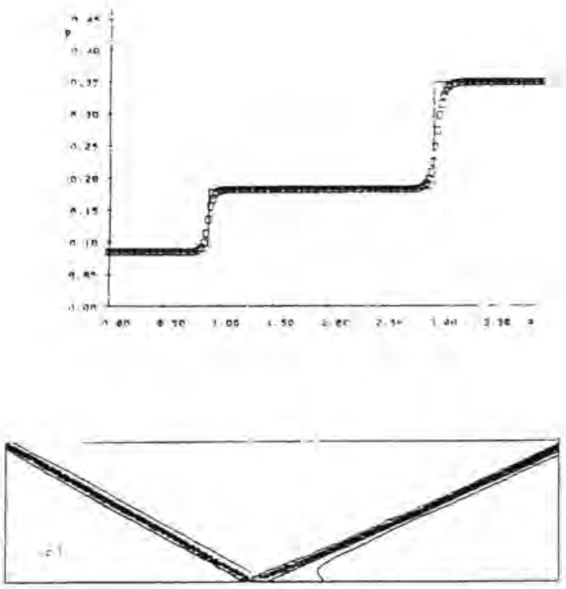


Figure 5.5: Result of the shock reflection problem calculated by Zhang et. al., (1994).

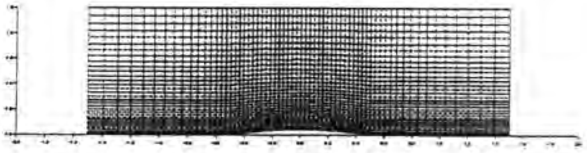


Figure 5.6: Computational grid for the 4% bicircular cascade.

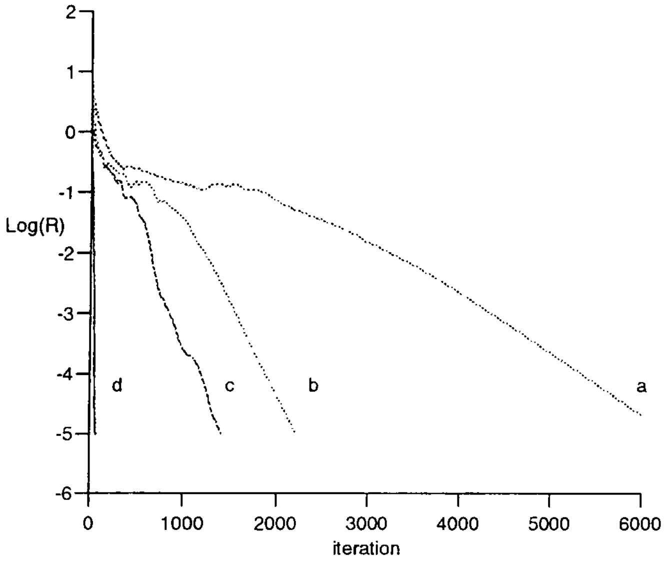


Figure 5.7: Convergence history on the 64 × 32 grid, a – 1 grid level, b – 2 grid level, c – 3 grid level, d – implicit calculation.

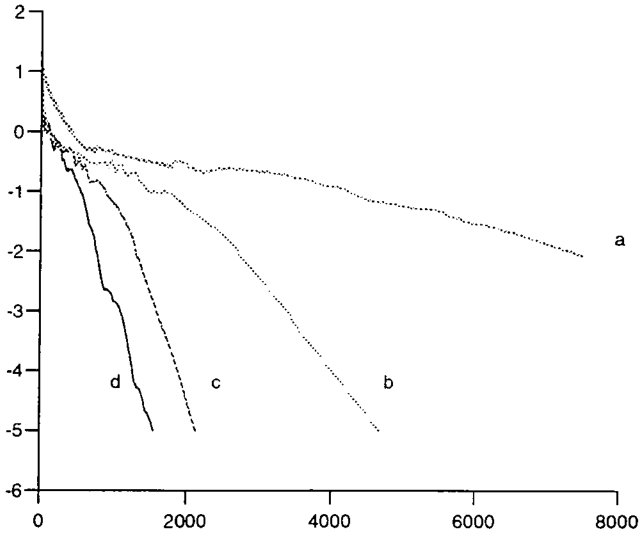


Figure 5.8: Convergence history on the 128 × 64 grid, a – 1 grid level, b – 2 grid level, c – 3 grid level, d – 4 grid level.

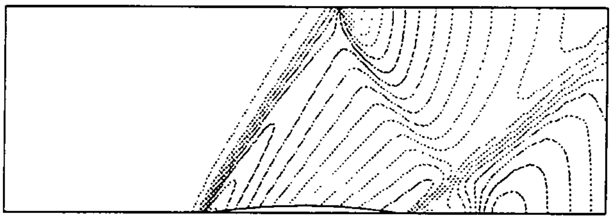


Figure 5.9: Iso-pressure lines on the 64 × 32 grid.

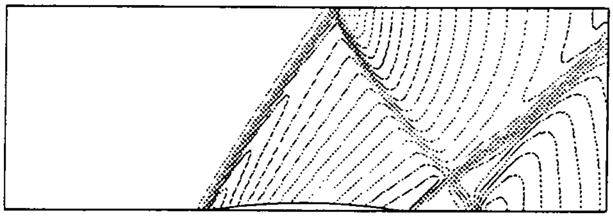


Figure 5.10: Iso-pressure lines on the 128 × 64 grid.

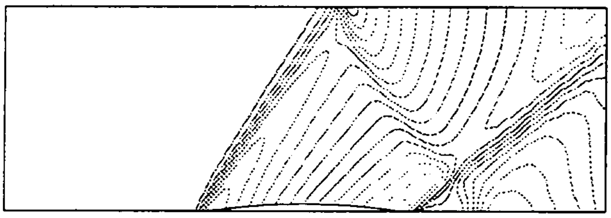


Figure 5.11: Iso-Mach lines on the 64 × 32 grid.

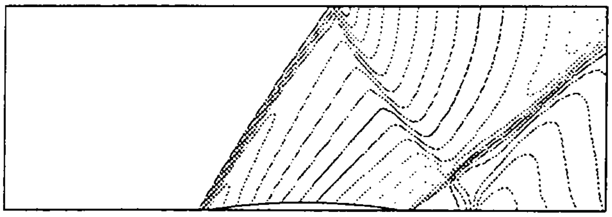


Figure 5.12: Iso-Mach lines on the 128 × 64 grid.

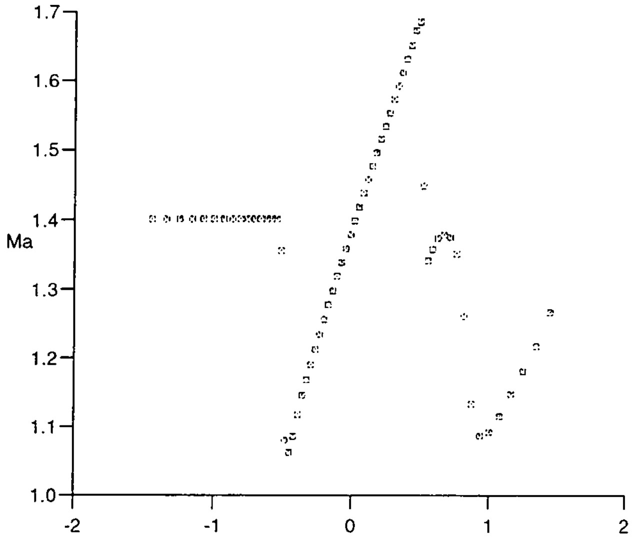


Figure 5.13: Mach number distribution on the wall using the 64 × 32 grid.

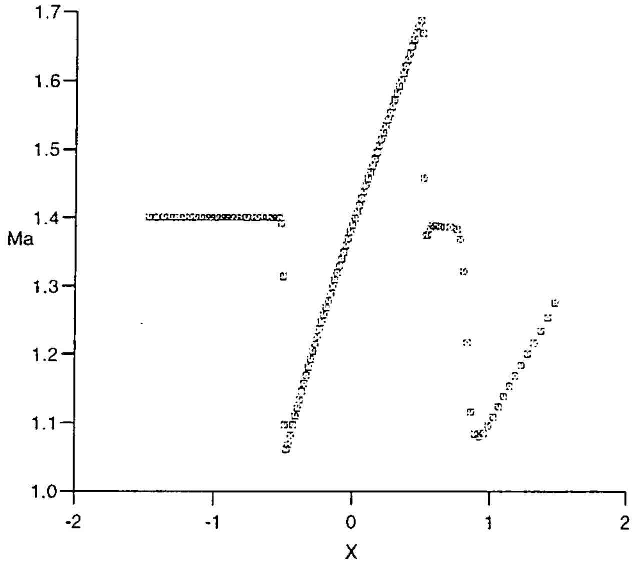


Figure 5.14: Mach number distribution on the wall using the 128 × 64 grid.

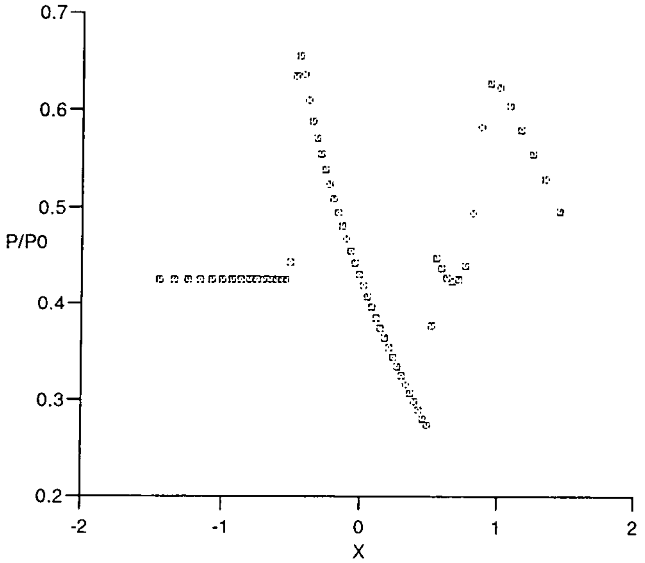


Figure 5.15: Pressure distribution on the wall using the 64 × 32 grid.

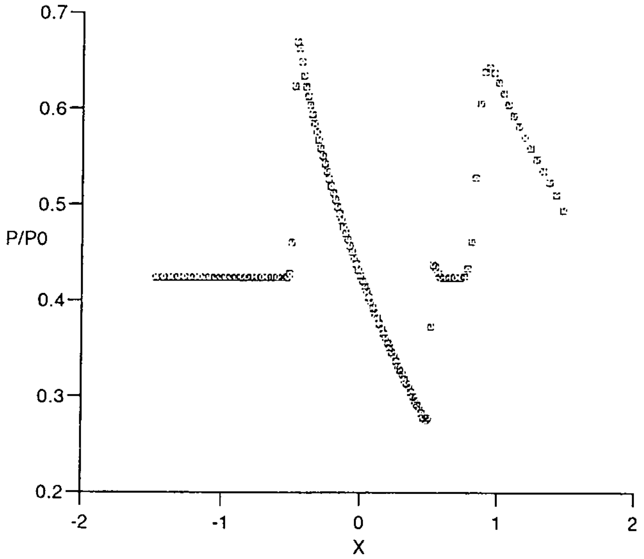


Figure 5.16: Pressure distribution on the wall using the 128 × 64 grid.

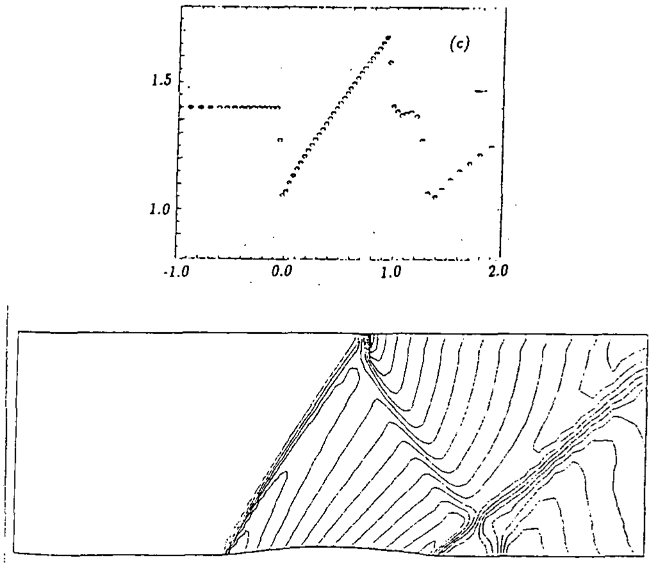
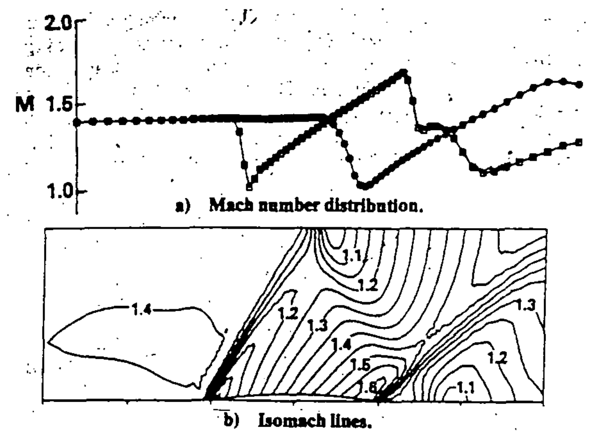


Figure 5.17: Roe's result for the supersonic bicircular cascade



a) Mach number distribution.

b) Isomach lines.

Figure 5.18: Ni's result for the supersonic bicircular cascade

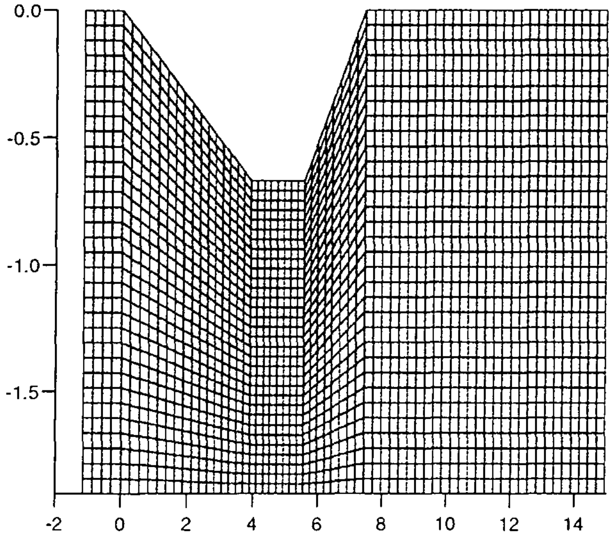


Figure 5.19: Illustration of computation grid of Denton's wedge cascade.

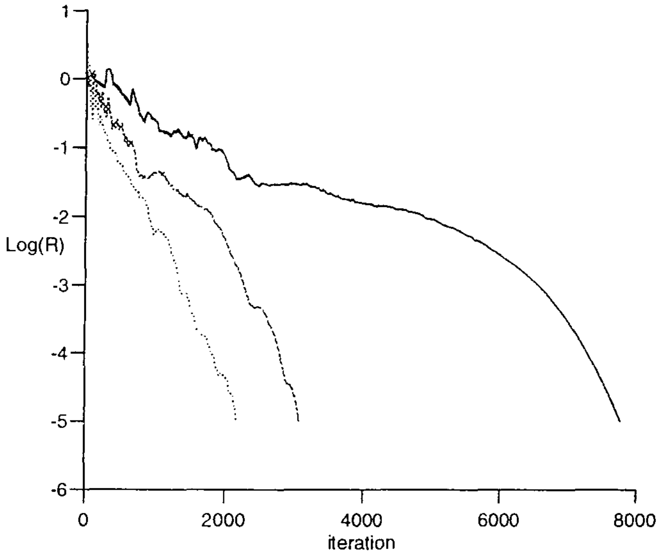


Figure 5.20: Convergence history of Denton' wedge cascade, with 1, 2, 3 level grid respectively

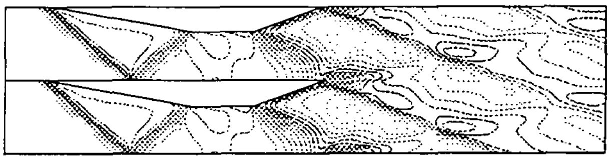


Figure 5.21: Iso-Mach lines of Denton's wedge cascade on the 64 × 32 grid

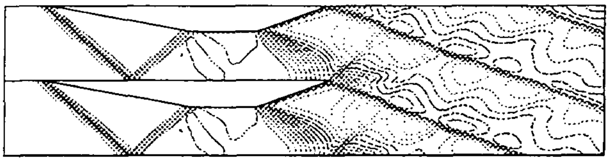


Figure 5.22: Iso-Mach lines of Denton's wedge cascade on the 128 × 32 grid

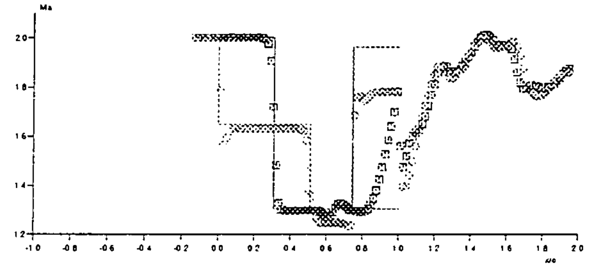
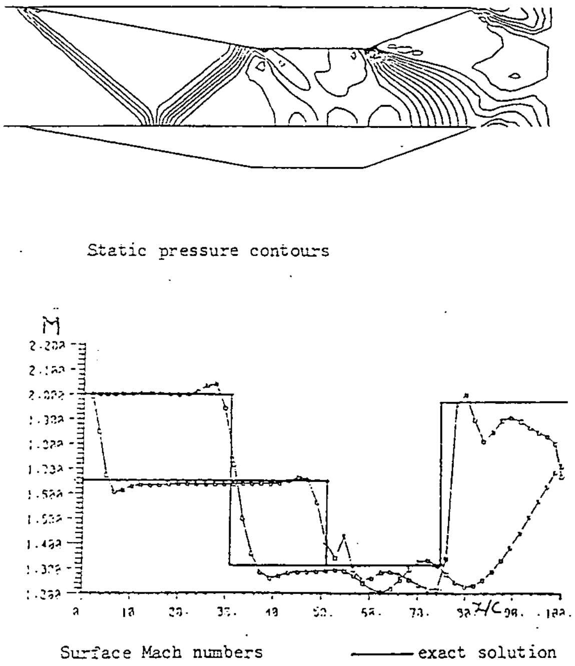


Figure 5.23: Surface Mach number distribution of Denton's wedge cascade.



Static pressure contours

Surface Mach numbers ——— exact solution

Figure 5.24: the result of Denton (1982)

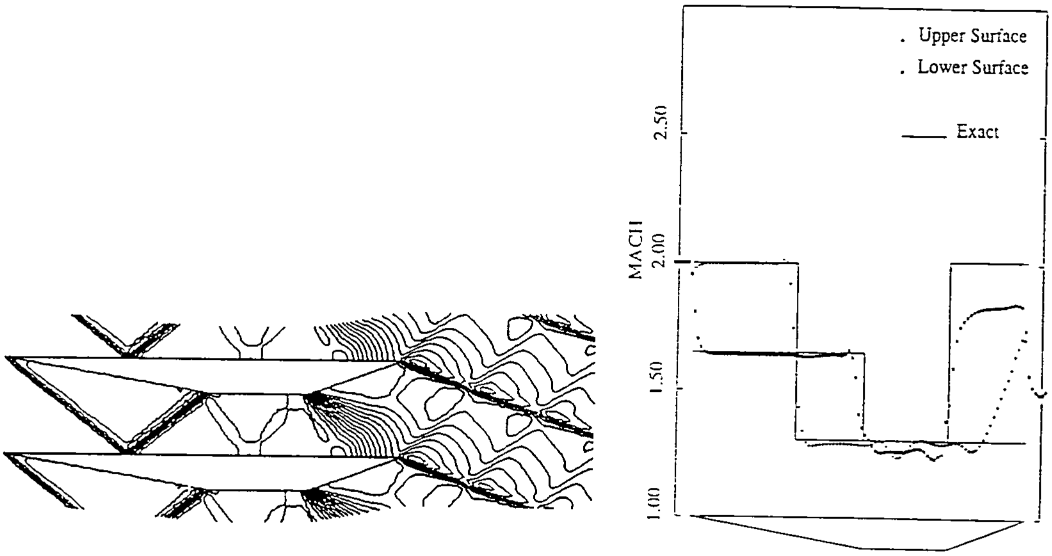


Figure 5.25: the result of Liu(1991)

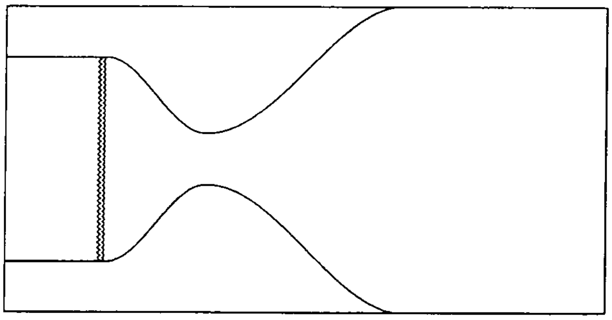


Figure 5.26: Iso-density lines of the shock propagation problem at the beginning of the calculation

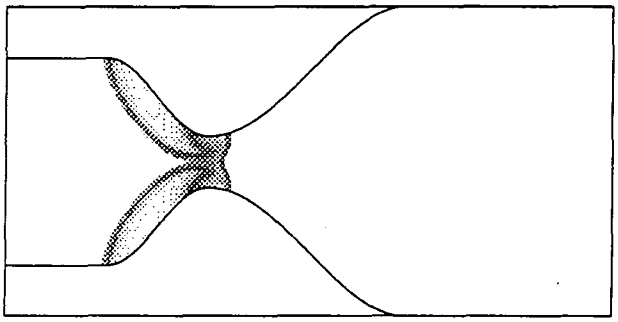


Figure 5.29: Iso-density lines of the shock propagation problem after 1500 time steps.

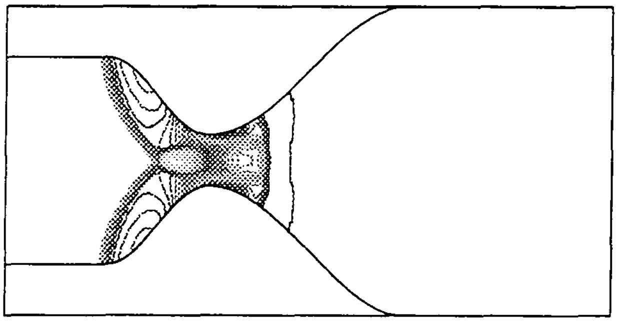


Figure 5.30: Iso-density lines of the shock propagation problem after 2000 time steps.

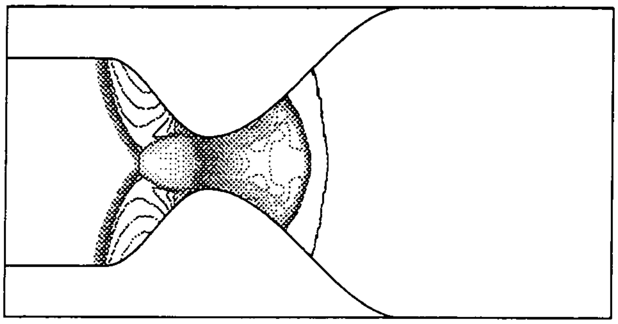


Figure 5.31: Iso-density lines of the shock propagation problem after 2500 time steps.

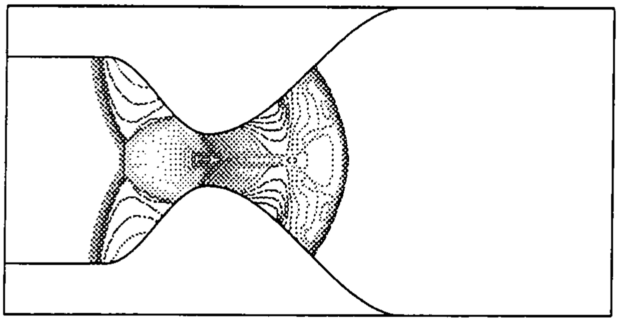


Figure 5.32: Iso-density lines of the shock propagation problem after 3000 time steps.

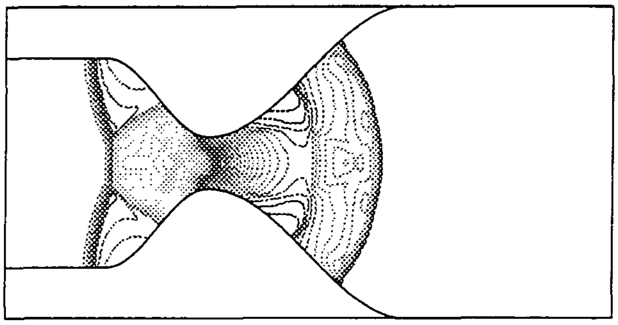


Figure 5.33: Iso-density lines of the shock propagation problem after 3500 time steps.

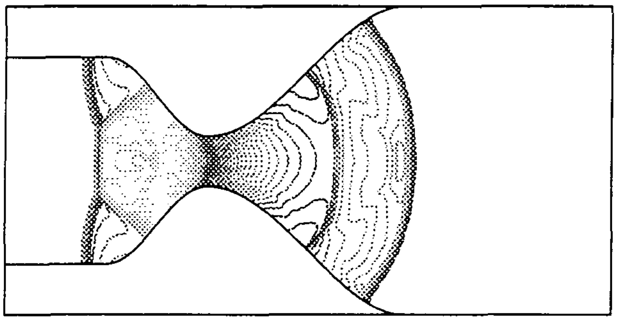


Figure 5.34: Iso-density lines of the shock propagation problem after 4000 time steps.

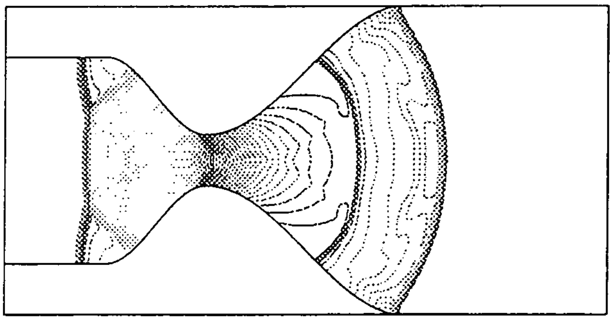


Figure 5.35: Iso-density lines of the shock propagation problem after 4500 time steps.

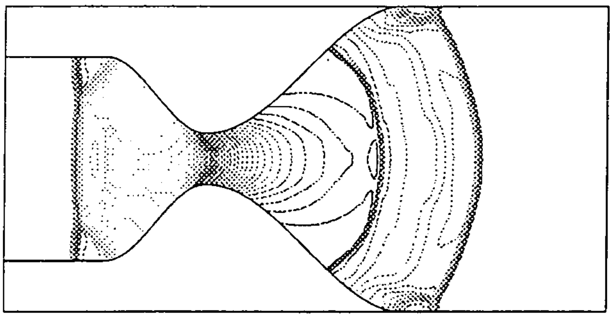


Figure 5.36: Iso-density lines of the shock propagation problem after 5000 time steps.

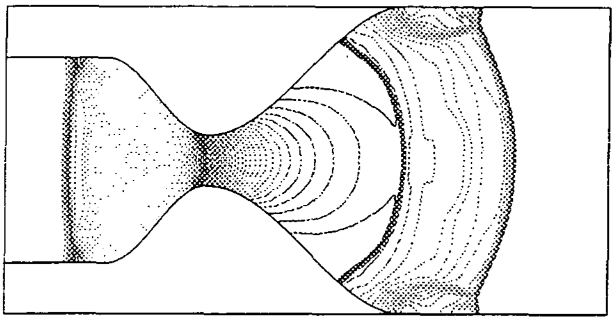


Figure 5.37: Iso-density lines of the shock propagation problem after 5500 time steps.

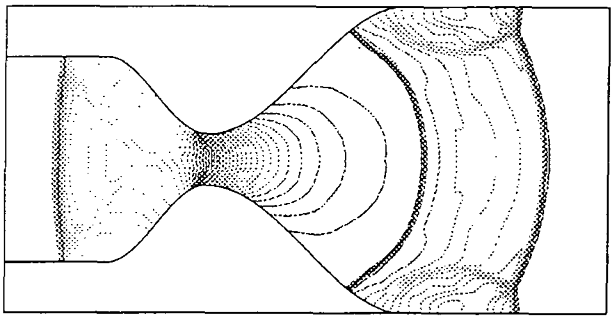


Figure 5.38: Iso-density lines of the shock propagation problem after 6000 time steps.

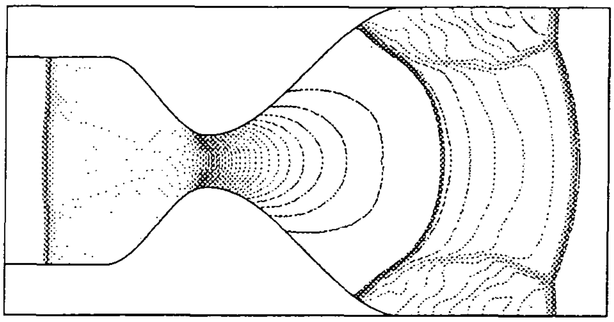


Figure 5.39: Iso-density lines of the shock propagation problem after 6500 time steps.

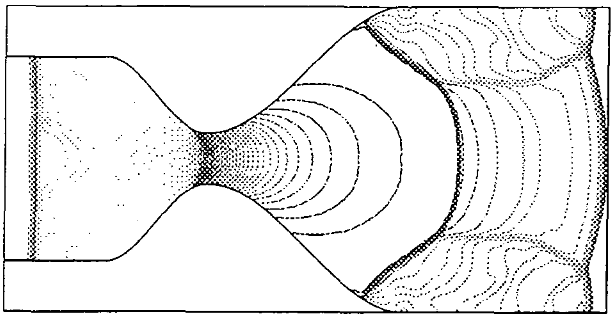


Figure 5.40: Iso-density lines of the shock propagation problem after 7000 time steps.

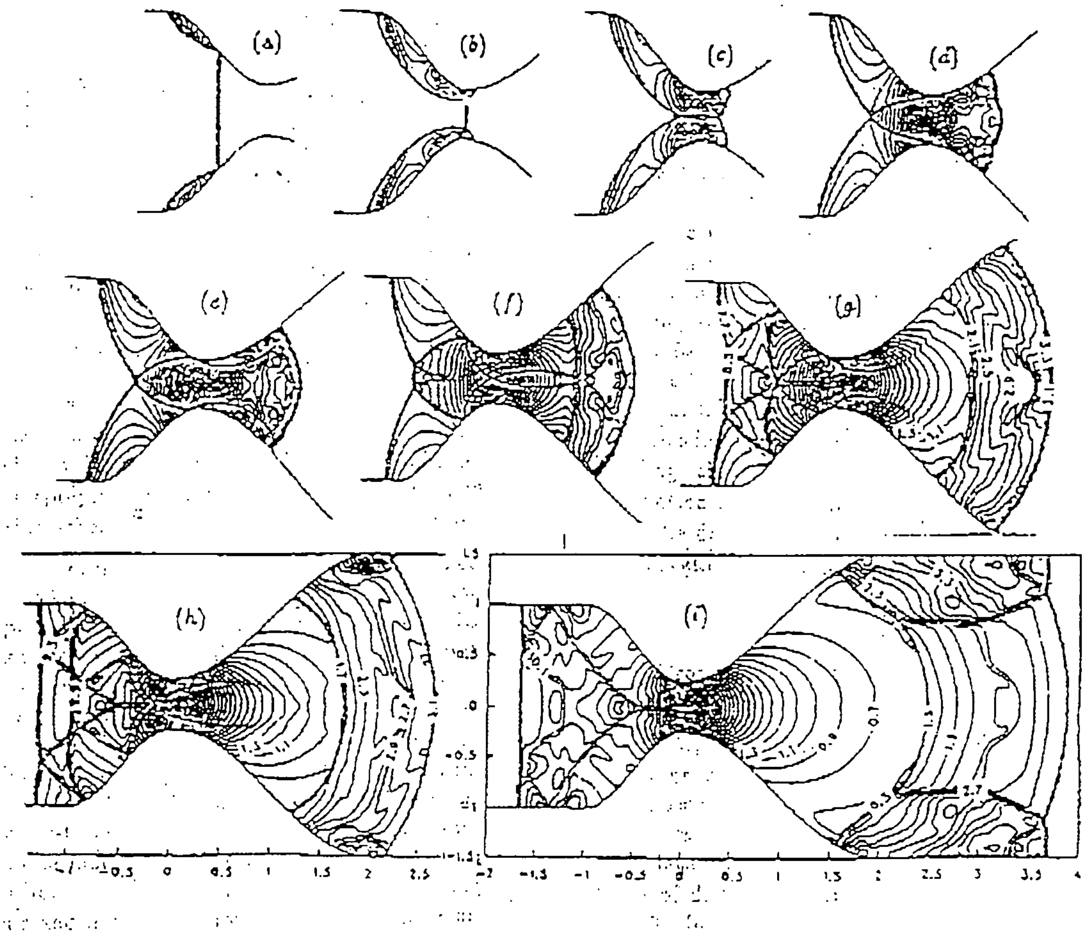


Figure 5.41: Yang's result of the shock propagation problem using the UNO3 scheme