50
108

A Cartesian Finite-Volume Method for the Euler Equations

by

Sang Keun Choi

Dissertation submitted to the Faculty of the

Virginia Polytechnic Institute and State University

in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

in

Aerospace Engineering

APPROVED:

Dr. B. Grossman, Chairman

Dr. J. A. Schetz

Dr. D. T. Mook

Dr. W. L. Neu

Dr. R. W. Walters

May, 1987

Blacksburg, Virginia

A Cartesian Finite-Volume Method for the Euler Equations

by

Sang Keun Choi

Dr. B. Grossman, Chairman

Aerospace Engineering

(ABSTRACT)

A numerical procedure has been developed for the computation of inviscid flows over arbitrary, complex two-dimensional geometries. The Euler equations are solved using a finite-volume method with a non-body-fitted Cartesian grid. A new numerical formulation for complicated body geometries is developed in conjunction with implicit flux-splitting schemes. A variety of numerical computations have been performed to validate the numerical methodologies developed. Computations for supersonic flow over a flat plate with an impinging shock wave are used to verify the numerical algorithm, without geometric considerations. The supersonic flow over a blunt body is utilized to show the accuracy of the non-body-fitted Cartesian grid, along with the shock resolution of flux-vector splitting scheme. Geometric complexities are illustrated with the flow through a two-dimensional supersonic inlet with and without an open bleed door. The ability of the method to deal with subsonic and transonic flows is illustrated by computations over a non-lifting NACA 0012 airfoil. The method is shown to be accurate, efficient and robust and should prove to be particulary useful in a preliminary design mode, where flows past a wide variety of complex geometries can be computed without complicated grid generation procedures.

Acknowledgements

The author wishes to express his deep appreciation to his advisor, Dr. B. Grossman who gave extensive guidance, advice and ecouragement to the last moment of this study. Without his continuous support, it would have been impossible to finish this work.

The author wishes to thank Dr. J. A. Schetz, Dr. D. T. Mook, Dr. W. L. Neu and Dr. R. W. Walters for serving on the Advisory Committee and for their helpful comments, discussions and suggestions on this study.

Very special thanks are given to his wife, , and two daughters and ., for their support and patience throughout this study.

Table of Contents

List of Illustrations

List of Symbols

A .....Jacobian matrix, $\frac{\partial \vec{F}}{\partial \vec{Q}}$

a .....speed of sound

a_{11} - a_{44}.....elements of A

B.....Jacobian matrix, $\frac{\partial \vec{G}}{\partial \vec{Q}}$

b .....characteristic speed

b_{11} - b_{44}.....elements of B

C_1 - C_9 .....coefficients of a linear system

c_v.....specific heat per unit mass at constant volume

D_b .....matrix defined in eq. (19)

D_l.....matrix defined in eq. (28)

E.....total energy per unit mass

$\vec{F}$.....flux vector defined in eq. (3)

f_1 - f_4 .....elements of $\vec{F}$

$\vec{G}$ .....flux vector defined in eq. (4)

g_1 - g_4.....elements of $\vec{G}$

H.....total entalpy per unit mass

I.....identity matrix

$\hat{i}, \hat{j}$ .....unit vectors in the x,y direction,respectively

i,j.....grid cell notations

$J^{\pm}$.....Riemann invariants defined in eq. (67)

k_x, k_y.....constants in eqs. (56)-(59)

L.....reference length

l_1 - l_4.....defined in eqs. (26),(27)

M .....Mach number, also denotes coefficient matrix

$\vec{N}$.....steady state residual in eq. (64)

$\vec{P}_b$.....vector defined in eq. (17)

p.....pressure

$\vec{Q}$.....vector of conserved variables defined in eq. (2)

q_1 - q_4.....conserved variables in eq. (2)

$\vec{R}$ .....steady state residual

S.....grid cell area

$\vec{S}_b$.....body surface vector defined in eq. (12)

s.....non-dimensional entropy per unit mass

s^*.....entropy per unit mass

T.....transformation matrix

t.....time

U_∞ .....free stream velocity

u, v .....velocity components in Cartesian coordinates

V .....volume

x,y .....Cartesian coordinates

Greek symbols

α .....angle of attack

γ .....ratio of specific heats

ε.....parameter in eq. (39)

θ .....body slope angle

Λ .....diagonal matrix of eigenvalues

λ .....eigenvalue of Jacobian matrix

ρ .....density

φ .....limiter

Operator

δ, Δ, ∇.....difference operators

Subscripts

b .....body surface conditions

c.....denotes extrapolated quantities from computational domain

f.....far-field conditions

i,j.....spatial indices

∞.....free stream conditions

Superscripts

n.....time level index

+,-.....denotes positive and negative flux and eigenvalue contributions

*.....denotes dimensional quantity

1.0 Introduction

The computation of flow fields about geometrically complex configurations still represents a formidable challenge. Most existing finite-difference or finite-volume methods require a body-conforming, nearly orthogonal system of grid lines in order to retain formal accuracy. This grid generation process, whether using numerical, algebraic or analytical methods, is far from automatic. A thorough descripton of numerical grid generation appears in the book by Thompson, Warsi and Mastin (Ref. 1). Comprehensive descriptions of algebraic and analytic methods appear in the review articles by Eiseman (Ref. 2) and Moretti (Ref. 3), respectively. Some very complex grids have been developed recently, using grid patching methods (e.g. Refs. 4-6). The main drawback of these methods is that the topological nature of the geometry must be built into the flow code, i.e. decisions regarding the number of zones in the flow must be made *a priori* and utilized within the basic framework of the resulting computer codes. Although, this is not a large drawback if the same classes of geometries are to be

computed many times, it may be a problem in a preliminary design mode where many possible configurations are to be analyzed.

One procedure which may overcome this problem is to use non-body-conforming grids. This approach was recently promoted by Wedan and South (Ref. 7), who utilized a Cartesian grid overlayed on the flowfield encompassing the body. Within a non-linear potential approach and a finite-volume formulation, accurate transonic flow solutions were obtained, without any grid generation. Only the intersections of the body geometry with the rectangular grid lines were needed to enforce the surface boundary condition, a much simpler task than developing a body-conforming grid. This procedure, which follows from the work of Purvis and Burkhalter (Ref. 8), is ideally suited for potential flows. The method was used in a supersonic potential flow marching code by Grossman and Whitaker (Ref. 9) where procedures for handling arbitrary thin fins were developed. The method was also seen to work well for the Euler solutions using a central-difference, explicit artificial viscosity formulation by Clarke, Hassan and Salas (Ref. 10). This was recently extended to three-dimensions by Gaffney, Hassan and Salas (Ref.11).

In addition to problems associated with geometric complexities, high-speed flow calculations must be able to accurately compute shock waves. Recently developed methods in the general category of "upwind" schemes, (e.g. see the survey articles by Harten, Lax and Van Leer (Ref. 12) and Roe (Ref. 13)), are extremly robust and are well-suited for computations about geometrically complex configurations. These schemes utilize spatial differences which, in some

sense, are biased in the direction determined by the sign of the characteristic speeds, in order to directly simulate the signal-propagation features of governing hyperbolic systems. This approach, as discussed in Ref. 13, is seen to stem from the early methods of Courant, Issacson and Rees (Ref. 14), and Godunov (Ref. 15).

Courant, Issacson and Rees proposed an explicit upwind difference scheme based on the characteristic form of the linear hyperbolic system,

$$u_t + b\,u_x = 0$$

such that

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} + b\frac{u_j^n - u_{j-1}^n}{\Delta x} = 0 \qquad if\ b > 0$$

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} + b\frac{u_{j+1}^n - u_j^n}{\Delta x} = 0 \qquad if\ b < 0$$

where Δx and Δt are, respectively, the spatial mesh size and the time step. In this scheme, the spatial derivative was approximated by a first-order backward difference for a positive characteristic speed b or by a first-order forward difference for a negative characteristic speed b. Hence this scheme is first-order accurate in space and time. But any other choice in spatial derivative (involving u_{j-1} , u_j, u_{j+1}) will be unstable.

Godunov (Ref. 15) extended this approach to the Euler equations in conservation form by using the exact solutions of local Rieman problems to obtain an upsteam-difference scheme.

Moretti (Ref. 16) developed a non-conservative scheme based explicitly on the method of characteristics for the Euler equations. This scheme is called the " λ scheme ", and is seen to have some favorable features regarding accuracy and implementation of boundary conditions, but does require shock fitting procedures.

Steger and Warming (Ref. 17) developed a characteristic-based scheme for the inviscid gas dynamic equations in conservation form. Taking advantage of the homogeneous property of the flux vectors for the Euler equations, they split the flux vectors into two groups , each member of a group having eigenvalues (characteristic speeds) of the same sign. This approach is referred to as " flux-vector splitting ". A refined version of this approach has been developed by Van Leer (Ref. 18). It has been shown in Ref. 19 that Van Leer splitting is continuous at sonic lines and stagnation points whereas Steger-Warming splitting is not. Details of the flux-vector splitting are included in Chap. 2.

Other "upwind" schemes are being developed. One class of these schemes, based on approximations to the local Rieman problem, is in the methods of Roe (Ref. 20) and Osher (Ref. 21). These schemes are usually referred to as flux-difference splitting. Some recent schemes which attempt to mathematically generalize these approaches are the so-called TVD schemes (Total-Variation-Diminishing), by Harten and his co-workers (Ref. 22) and the higher-order accuracy ENO schemes (Essentially-Non-Oscillatory) by Osher and his co-workers (Ref. 23).

The objective of the present study is to develop a computational procedure for solving the Euler equations using flux-vector splitting with a non-body conforming grid. In particular, we develop an implementation of the Van Leer scheme (Ref. 18) in concert with the Cartesian finite-volume method. Special consideration is given to the flux splitting in the vicinity of the boundaries. These schemes, when applied to body-fitted grids have yielded accurate and efficient solutions (Refs. 24-25) to the Euler equations, particulary at high Mach numbers. The natural dissipation in the schemes captures discontinuities without separate explicit spatial dissipation terms.

In Section 2-1, we derive a semi-discrete delta form of the Euler equations from the non-dimensional conservative form of the two-dimensional Euler equations using Euler-implicit time integration and linearization through local Taylor expansions. Computational grid generation near the body is explained in Section 2-2. In Section 2-3, the numerical equations are derived in the context of Cartesian finite-volume methods. Combining the governing equations with surface boundary conditions, new numerical, governing equations, which can be applied to arbitrary complex geometries without any mapping, are derived. Flux-vector splitting by Steger and Warming (Ref. 17) and by Van Leer (Ref. 18) is discussed in Section 2-4, and Van Leer flux-vector splitting is employed to obtain the final numerical equations. The numerical solution of the governing equations by relaxation techniques is also briefly explained. In Chapter 3, the numerical methods are tested with a variety of geometries and flow problems such as supersonic shock reflections on a flat plate, supersonic blunt body flows

with embedded subsonic regions, scram-jet inlet flows and subsonic non-lifting airfoil flows. Computational results are compared with exact solutions and existing solutions obtained with other numerical methods. The grid distribution of the non-body-fitted Cartesian grid is far from optimal because of the wasted grids areas inside the body , and the inefficient Cartesian stretching to the far-field. Nonetheless, the numerical results, with this procedure, indicate that relatively accurate and efficient results are obtained. In the final chapter, we present an overview of our method and discuss the limitations along with recommendations for future research. It is our opinion that this approach, particulary when extended to three dimensions, will serve as a useful design tool.

2.0 Formulation

2.1 Governing Equations

The non-dimensional vector conservation form of the two-dimensional Euler equations, without body forces and heat transfer, can be written in Cartesian coordinates as:

$$\frac{\partial \vec{Q}}{\partial t} + \frac{\partial \vec{F}}{\partial x} + \frac{\partial \vec{G}}{\partial y} = 0 \tag{1}$$

where

$$\vec{Q} = \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ \rho E \end{bmatrix} \equiv \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{bmatrix} \tag{2}$$

$$\vec{F} = \begin{bmatrix} \rho u \\ \rho u^2 + p \\ \rho u v \\ (\rho E + p)u \end{bmatrix} \tag{3}$$

$$\vec{G} = \begin{bmatrix} \rho v \\ \rho u v \\ \rho v^2 + p \\ (\rho E + p)v \end{bmatrix} \tag{4}$$

The quantities p, ρ, u, v are the pressure, density and velocity components in the x and y directions, respectively. E is the total energy per unit mass. The Cartesian coordinate lengths x and y are non-dimensionalized by a reference length L, all velocities and the speed of sound by U_∞, the density by ρ_∞, the pressure by $\rho_\infty U_\infty^2$, the total energy per unit mass by U_∞^2. Here, subscript ∞ denotes free-stream quantities.

To complete the set of governing equations, we use the equation of state for a perfect gas, i.e.

$$p = (\gamma - 1)\rho[E - \frac{1}{2}(u^2 + v^2)] \tag{5}$$

where γ is the ratio of specific heats.

Since we are interested in generating steady-state solutions, the specific form of the time integration is not important. It has been shown that implicit relaxation procedures can be an efficient means for solving the steady Euler equations (Refs. 26, 27). The simplest implicit time integration procedure is

first-order Euler-implicit. In this scheme, the value of $\vec{Q}$ at the advanced time level, n + 1, is given by

$$\vec{Q}^{n+1} = \vec{Q}^{n} + \Delta t \left(\frac{\partial \vec{Q}}{\partial t}\right)^{n+1} + O(\Delta t^2)$$

or

$$\vec{Q}^{n+1} = \vec{Q}^{n} - \Delta t \left(\frac{\partial \vec{F}}{\partial x} + \frac{\partial \vec{G}}{\partial y}\right)^{n+1} + O(\Delta t^2) \tag{6}$$

where

$$\Delta t \equiv t^{n+1} - t^{n}$$

A local Taylor expansion about $\vec{Q}^n$ yields linearizations of $\vec{F}$ and $\vec{G}$, which are homogeneous functions of degree one in $\vec{Q}$, as

$$\vec{F}^{n+1} = \vec{F}^{n} + A^{n}(\vec{Q}^{n+1} - \vec{Q}^{n}) + O(\Delta t^2) \tag{7}$$

$$\vec{G}^{n+1} = \vec{G}^{n} + B^{n}(\vec{Q}^{n+1} - \vec{Q}^{n}) + O(\Delta t^2) \tag{8}$$

where A and B are the Jacobian matrices

$$\begin{aligned} A &\equiv \frac{\partial \vec{F}}{\partial \vec{Q}} \\ &= \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix} \end{aligned} \tag{9}$$

here

$$a_{11} = 0\,, \qquad a_{12} = 1\,, \qquad a_{13} = 0\,, \qquad a_{14} = 0$$

$$a_{21} = -\frac{q_2^2}{q_1^2} + \frac{(\gamma - 1)}{2}\frac{q_2^2 + q_3^2}{q_1^2}$$

$$a_{22} = (3 - \gamma)\frac{q_2}{q_1}\,, \qquad a_{23} = -(\gamma - 1)\frac{q_3}{q_1}\,, \qquad a_{24} = -(\gamma - 1)$$

$$a_{31} = -\frac{q_2 q_3}{q_1^2}\,, \qquad a_{32} = \frac{q_3}{q_1}\,, \qquad a_{33} = \frac{q_2}{q_1}\,, \qquad a_{34} = 0$$

$$a_{41} = -\gamma\frac{q_2 q_4}{q_1^2} + (\gamma - 1)\frac{q_2(q_2^2 + q_3^2)}{q_1^3}$$

$$a_{42} = \gamma\frac{q_4}{q_1} - \frac{(\gamma - 1)}{2}\frac{3q_2^2 + q_3^2}{q_1^2}$$

$$a_{43} = -(\gamma - 1)\frac{q_2 q_3}{q_1^2}$$

$$a_{44} = \gamma\frac{q_2}{q_1}$$

and

$$B \equiv \frac{\partial \vec{G}}{\partial \vec{Q}}$$

$$= \begin{bmatrix} b_{11} & b_{12} & b_{13} & b_{14} \\ b_{21} & b_{22} & b_{23} & b_{24} \\ b_{31} & b_{32} & b_{33} & b_{34} \\ b_{41} & b_{42} & b_{43} & b_{44} \end{bmatrix} \tag{10}$$

here

$$b_{11} = 0\,, \qquad b_{12} = 0\,, \qquad b_{13} = 1\,, \qquad b_{14} = 0$$

$$b_{21} = -\frac{q_2 q_3}{q_1^2}\,, \qquad b_{22} = \frac{q_3}{q_1}\,, \qquad b_{23} = \frac{q_2}{q_1}\,, \qquad b_{24} = 0$$

$$b_{31} = -\frac{q_3^2}{q_1^2} + \frac{(\gamma - 1)}{2}\frac{q_2^2 + q_3^2}{q_1^2}$$

$$b_{32} = -(\gamma - 1)\frac{q_2}{q_1}\,, \qquad b_{33} = (3 - \gamma)\frac{q_3}{q_1}\,, \qquad b_{34} = -(\gamma - 1)$$

$$b_{41} = -\gamma\frac{q_3 q_4}{q_1^2} + (\gamma - 1)\frac{q_3(q_2^2 + q_3^2)}{q_1^3}$$

$$b_{42} = -(\gamma - 1)\frac{q_2 q_3}{q_1^2}$$

$$b_{43} = \gamma\frac{q_4}{q_1} - \frac{(\gamma - 1)}{2}\frac{q_2^2 + 3q_3^2}{q_1^2}$$

$$b_{44} = \gamma \frac{q_3}{q_1}$$

When the linearizations given by eqs. (7) and (8) are applied to eq. (6), a linear system for $\vec{Q}^{n+1}$ results and may be written in delta form as

$$[\frac{I}{\Delta t} + \frac{\partial A^n}{\partial x} + \frac{\partial B^n}{\partial y}]\, \delta\vec{Q} = -\,(\frac{\partial \vec{F}}{\partial x} + \frac{\partial \vec{G}}{\partial y})^n \tag{11}$$

where I is the identity matrix, and $\delta\vec{Q} \equiv \vec{Q}^{n+1} - \vec{Q}^n$. This form of the governing equations is sometimes referred to as a semi-discrete form, since the time derivative has been discretized.

2.2 *Grid Generation*

Typical grid cells near an arbitrary body are shown in Figs. 1 and 2. In Fig. 1, grid cells are generated simply by overlaying a rectangular grid system on a body and erasing the portions contained within the body. To avoid ill-conditioned coefficient matrices of the linearized systems, we merge small body cells (usually less than 10% of normal cell size) into adjacent interior cells. The direction of merging, ± x or ± y direction, is decided by the slope of the body surface in the small body cell. Final computation grid cells near the body are shown in Fig. 2.

An interior cell and a body cell are shown in Fig. 3. We denote cell sides $\Delta y_{i-\frac{1}{2},j}$, $\Delta y_{i+\frac{1}{2},j}$, $\Delta x_{i,j-\frac{1}{2}}$, and $\Delta x_{i,j+\frac{1}{2}}$ for the (i, j) grid cell as shown in Fig. 3. Then, we define a body surface vector, $\vec{S}_b$, of the cell as

$$\begin{aligned}\vec{S}_b &\equiv (\Delta y_{i-\frac{1}{2},j} - \Delta y_{i+\frac{1}{2},j})\,\hat{i} + (\Delta x_{i,j-\frac{1}{2}} - \Delta x_{i,j+\frac{1}{2}})\,\hat{j} \\ &= \Delta y_b\,\hat{i} + \Delta x_b\,\hat{j}\end{aligned} \tag{12}$$

where

$$\Delta y_b \equiv \Delta y_{i-\frac{1}{2},j} - \Delta y_{i+\frac{1}{2},j},$$

$$\Delta x_b \equiv \Delta x_{i,j-\frac{1}{2}} - \Delta x_{i,j+\frac{1}{2}}$$

and $\hat{i}, \hat{j}$ are the unit vectors of x axis and y axis, respectively. Eq. (12) is valid for arbitrary cells; $\vec{S}_b$ becomes zero for interior cells from the definitions of Δy_b and Δx_b.

In general, we have a discontinuous grid near a body surface due to small cell merging (Fig. 2). The problem of a discontinuous grid is solved simply by satisfying local flux continuity (e.g. Ref. 5). For example, referring to Fig. 2, we require the fluxes through the left side of cell 3 to be equal to the sum of the fluxes through the right side of cells 1 and 2. This idea may be directly applied, if necessary, to create locally dense grid regions in future work.

Using the above representation of cell sides, we now derive a unique set of difference equations within the context of a finite-volume formulation.

2.3 *Numerical Discretization and Surface Boundary Conditions*

The semi-discrete form of the governing equations, eq. (11) may be put into finite-volume form by performing the integration about an arbitrary control volume as :

$$\iiint\{[\frac{I}{\Delta t} + \frac{\partial A^n}{\partial x} + \frac{\partial B^n}{\partial y}]\delta\vec{Q}\}dV = -\iiint(\frac{\partial \vec{F}^n}{\partial x} + \frac{\partial \vec{G}^n}{\partial y})dV \tag{13}$$

Applying the divergence theorem to the eq. (13), we get

$$\iiint\frac{\delta\vec{Q}}{\Delta t}dV + \iint(A^n\delta\vec{Q}\hat{i} + B^n\delta\vec{Q}\hat{j})\bullet d\vec{S} = -\iint(\vec{F}^n\hat{i} + \vec{G}^n\hat{j})\bullet d\vec{S} \tag{14}$$

If we consider the control volume to consist of the cell area which includes an intersection with the body, as illustrated in Fig. 3-b, we obtain :

$$
\begin{aligned}
&(\frac{\delta\vec{Q}}{\Delta t}S)_{i,j} + (A^n\delta\vec{Q}\Delta y)_{i+\frac{1}{2},j} - (A^n\delta\vec{Q}\Delta y)_{i-\frac{1}{2},j} \\
&+ (B^n\delta\vec{Q}\Delta x)_{i,j+\frac{1}{2}} - (B^n\delta\vec{Q}\Delta x)_{i,j-\frac{1}{2}} + (A^n\delta\vec{Q}\Delta y + B^n\delta\vec{Q}\Delta x)_b \\
&= -\left[(\vec{F}^n\Delta y)_{i+\frac{1}{2},j} - (\vec{F}^n\Delta y)_{i-\frac{1}{2},j} + (\vec{G}^n\Delta x)_{i,j+\frac{1}{2}} - (\vec{G}^n\Delta x)_{i,j-\frac{1}{2}}\right] \\
&- (\vec{F}^n\Delta y + \vec{G}^n\Delta x)_b
\end{aligned}
\tag{15}
$$

where $(\delta\vec{Q})_{i,j}$ represents the cell-area average of the change in $\vec{Q}$ for the cell associated with the index i, j. The terms $(\vec{F})_{i\pm\frac{1}{2},j}$ and $(\vec{Q})_{i,j\pm\frac{1}{2}}$ are the cell-side averages of the fluxes $\vec{F}$ and $\vec{G}$ and the terms $(A\delta\vec{Q})_{i\pm\frac{1}{2},j}$, and $(B\delta\vec{Q})_{i,j\pm\frac{1}{2}}$ are the cell-side averages of the linearized changes in the fluxes. The terms with the subscript b correspond to the side of the cell at the body surface. The term $S_{i,j}$ corresponds to the cross-sectional area of the cell (i, j).

Eq. (15) represents a cell-centered finite-volume formulation of the Euler equations applied to the cell volume described in Fig. 3-b. An Euler-implicit time integration scheme has been implemented and the equation is written in delta form. The derivation of this equation , starting from the semi-discrete finite-difference form, eq. (11), shows clearly the duality between the finite-difference and finite-volume approaches. The finite-volume form, eq. (15), could have been developed directly from the integral conservation equations and then put in semi-discrete form using an Euler-implicit time integration and local linearization. The identical equation would result. Of course, the direct development, would not require the use of the divergence theorem at all, so that the equations may clearly be applied across shock waves. The steady-state

accuracy of eq. (15) will depend upon the approximations used to determine the cell-side values.

We can simplify the above system by noting that on the body surface,

$$\vec{V} \bullet \hat{n} = 0$$

or

$$(u\Delta y)_b + (v\Delta x)_b = 0 \tag{16}$$

where $\vec{V} = u\,\hat{i} + v\,\hat{j}$, and $\hat{n}$ is a surface normal unit vector. From eqs. (3), (4), and (16) the body surface quantities in eq. (15) become

$$\begin{aligned}(\vec{F}\Delta y + \vec{G}\Delta x)_b &= \begin{bmatrix} \rho(u\Delta y + v\Delta x) \\ \rho u(u\Delta y + v\Delta x) + p\Delta y \\ \rho v(u\Delta y + v\Delta x) + p\Delta x \\ (\rho E + p)(u\Delta y + v\Delta x) \end{bmatrix}_b \\ &= \begin{bmatrix} 0 \\ p\Delta y \\ p\Delta x \\ 0 \end{bmatrix}_b \\ &\equiv \vec{P}_b \end{aligned} \tag{17}$$

$$(A\Delta y + B\Delta x)_b \delta\vec{Q}_b$$

$$= \begin{bmatrix} 0 & \Delta y & \Delta x & 0 \\ 0 & u\Delta y & u\Delta x & 0 \\ 0 & v\Delta y & v\Delta x & 0 \\ 0 & [\gamma E - (\gamma - 1)/2(u^2 + v^2)]\Delta y & [\gamma E - (\gamma - 1)/2(u^2 + v^2)]\Delta x & 0 \end{bmatrix}_b \delta\vec{Q}_b$$

$$+ \begin{bmatrix} 0 & 0 & 0 & 0 \\ (\gamma - 1)/2(u^2 + v^2)\Delta y & -(\gamma - 1)u\Delta y & -(\gamma - 1)v\Delta y & (\gamma - 1)\Delta y \\ (\gamma - 1)/2(u^2 + v^2)\Delta x & -(\gamma - 1)u\Delta x & -(\gamma - 1)v\Delta x & (\gamma - 1)\Delta x \\ 0 & 0 & 0 & 0 \end{bmatrix}_b \delta\vec{Q}_b$$

Since

$$(\rho u\Delta y + \rho v\Delta x)_b = 0$$

then

$$(\delta q_2\Delta y + \delta q_3\Delta x)_b = 0$$

So that the multiplication of the first matrix in the above equation by $\delta\vec{Q}_b$ will be identically zero. The second matrix is precisely $\dfrac{\partial\vec{P}_b}{\partial\vec{Q}_b}$. Hence, we may write

$$(A\Delta y + B\Delta x)_b \delta\vec{Q}_b = \frac{\partial\vec{P}_b}{\partial\vec{Q}_b}\delta\vec{Q}_b = D_b\delta\vec{Q}_b \tag{18}$$

where

$$D_b \equiv \frac{\partial \vec{P}_b}{\partial \vec{Q}_b} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ \frac{\partial p}{\partial q_1}\Delta y & \frac{\partial p}{\partial q_2}\Delta y & \frac{\partial p}{\partial q_3}\Delta y & \frac{\partial p}{\partial q_4}\Delta y \\ \frac{\partial p}{\partial q_1}\Delta x & \frac{\partial p}{\partial q_2}\Delta x & \frac{\partial p}{\partial q_3}\Delta x & \frac{\partial p}{\partial q_4}\Delta x \\ 0 & 0 & 0 & 0 \end{bmatrix}_b \tag{19}$$

with

$$\frac{\partial p}{\partial q_1} = \frac{(\gamma - 1)}{2}(u^2 + v^2)$$

$$\frac{\partial p}{\partial q_2} = -(\gamma - 1)u$$

$$\frac{\partial p}{\partial q_3} = -(\gamma - 1)v$$

$$\frac{\partial p}{\partial q_4} = (\gamma - 1)$$

The body surface pressure, p_b can be obtained from either the normal momentum equation or by extrapolation of cell-centerd values (Ref. 10). Other quantities on the surface such as ρ_b and H_b can be also obtained, in terms of cell- centered values, through extrapolation from the computational domain (Ref. 24). The total entalpy, H can be written as

$$H = E + \frac{p}{\rho}$$
$$= \gamma E - \frac{\gamma - 1}{2}(u^2 + v^2) \tag{20}$$

Then, E_b and u_b , v_b are determined from eqs. (16) and (20) from the following:

$$E_b = H_b - \frac{p_b}{\rho_b}$$

$$u_b = \sqrt{\frac{2}{\gamma - 1}(\gamma E_b - H_b)} \cos\theta \tag{21}$$

$$v_b = \sqrt{\frac{2}{\gamma - 1}(\gamma E_b - H_b)} \sin\theta$$

where

$$\theta = \tan^{-1}\left(-\frac{\Delta y_b}{\Delta x_b}\right)$$

Since the accuracy of the unsteady terms in eq. (15) is not critical for the steady-state solutions, only the first-order extrapolations are used for ρ_b , H_b and p_b in eqs. (17) and (18) to simplify the linearization of $\delta\vec{Q}_b$ in terms of cell-centered values. Now, we have

$$\rho_b = \rho_{i,j}$$

$$p_b = p_{i,j}$$

$$H_b = H_{i,j} \tag{22}$$

$$E_b = E_{i,j}$$

$$u_b = \sqrt{u_{i,j}^2 + v_{i,j}^2}\cos\theta$$

$$v_b = \sqrt{u_{i,j}^2 + v_{i,j}^2}\sin\theta$$

The $\delta\vec{Q}_b$ in eq. (15) becomes

$$\delta\vec{Q}_b = \begin{bmatrix} \delta q_1 \\ \delta q_2 \\ \delta q_3 \\ \delta q_4 \end{bmatrix}_b = \begin{bmatrix} \delta\rho \\ \delta(\rho u) \\ \delta(\rho v) \\ \delta(\rho E) \end{bmatrix}_b$$

$$= \begin{bmatrix} \delta\rho \\ \delta(\rho\sqrt{u^2 + v^2}\cos\theta) \\ \delta(\rho\sqrt{u^2 + v^2}\sin\theta) \\ \delta(\rho E) \end{bmatrix}_{i,j} \tag{23}$$

$$= \begin{bmatrix} \delta q_1 \\ \delta(\sqrt{q_2^2 + q_3^2}\cos\theta) \\ \delta(\sqrt{q_2^2 + q_3^2}\sin\theta) \\ \delta q_4 \end{bmatrix}_{i,j}$$

The linearization of δq_{2_b} and δq_{3_b} are explained as follows. From the definition of $\delta\vec{Q}$, it follows that

$$q^{n+1} = q^n + \delta q \tag{24}$$

From eq. (22), it follows that

$$q_{2_b}^2 = (q_{2_{i,j}}^2 + q_{3_{i,j}}^2)\cos^2\theta \tag{25}$$

Substituting eq. (24) into eq. (25) yields

$$(q_{2_b}^n + \delta q_{2_b})^2 = [(q_{2_{i,j}}^n + \delta q_{2_{i,j}})^2 + (q_{3_{i,j}}^n + \delta q_{3_{i,j}})^2]\cos^2\theta$$

Neglecting the higher-order terms, we have

$$\begin{aligned}\delta q_{2_b} &= \frac{q_{2_{i,j}}^n \delta q_{2_{i,j}} + q_{3_{i,j}}^n \delta q_{3_{i,j}}}{q_{2_b}^n}\cos^2\theta \\ &= l_1 \delta q_{2_{i,j}} + l_2 \delta q_{3_{i,j}}\end{aligned} \tag{26}$$

where

$$l_1 = \left[\frac{q_{2_{i,j}}^n \cos\theta}{\sqrt{(q_{2_{i,j}}^n)^2 + (q_{3_{i,j}}^n)^2}}\right]$$

$$l_2 = \left[\frac{q_{3_{i,j}}^n \cos\theta}{\sqrt{(q_{2_{i,j}}^n)^2 + (q_{3_{i,j}}^{)\ 2}}}\right]$$

Similary,

$$\delta q_{3_b} = l_3 \delta q_{2_{i,j}} + l_4 \delta q_{3_{i,j}} \tag{27}$$

where

$$l_3 = \left[\frac{q^n_{2_{i,j}} \sin\theta}{\sqrt{(q^n_{2_{i,j}})^2 + (q^n_{3_{i,j}})^2}} \right]$$

$$l_4 = \left[\frac{q_{3_{i,j}} \sin\theta}{\sqrt{(q^n_{2_{i,j}})^2 + (q^n_{3_{i,j}})^2}} \right]$$

As a result $\delta\vec{Q}_b$ can be written in terms of cell-centered values as

$$\delta\vec{Q}_b = D_l \; \delta\vec{Q}_{i,j} \tag{28}$$

where

$$D_l = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & l_1 & l_2 & 0 \\ 0 & l_3 & l_4 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

The body surface pressure, p_b, in eq. (17), which appears on the right-hand side of eq. (15), is obtained using the second-order extrapolation in the present analysis.

From the definitions in eqs. (17), (18) and (28) the governing equations can be written as

$$
\begin{aligned}
&(\frac{\delta\vec{Q}}{\Delta t}S)_{i,j} + (A^n\delta\vec{Q}\Delta y)_{i+\frac{1}{2},j} - (A^n\delta\vec{Q}\Delta y)_{i-\frac{1}{2},j} \\
&+ (B^n\delta\vec{Q}\Delta x)_{i,j+\frac{1}{2}} - (B^n\delta\vec{Q}\Delta x)_{i,j-\frac{1}{2}} + D_b D_l \delta\vec{Q}_{i,j} \\
&= -\left[(\vec{F}^n\Delta y)_{i+\frac{1}{2},j} - (\vec{F}^n\Delta y)_{i-\frac{1}{2},j} + (\vec{G}^n\Delta x)_{i,j+\frac{1}{2}} - (\vec{G}^n\Delta x)_{i,j-\frac{1}{2}}\right] + \vec{P}_b
\end{aligned}
\tag{29}
$$

As shown above, eq. (29) is actually a combination of governing equations and surface boundary conditions. From the definition of our grid cell geometries, $\vec{P}_b$ and D_b disappear at the interior cells.

2.4 *Flux-Vector Splitting and Relaxation Algorithm*

Recently, a host of computational methods, usually referred to as "upwind methods", and loosely based on the method of characteristics, have been developed. The methods may be generally categorized as FVS, flux-vector splitting (Refs. 17-19), FDS, flux-difference splitting (Refs. 20, 21, 28), characteristic (Refs. 16, 29), and TVD, total variation diminishing (Refs. 22, 30) schemes. As discussed in the surveys in Refs. 12 and 13, these approaches generally stem from the early methods of Courant, Isaacson and Rees (Ref. 14) and Godunov (Ref. 15). The main advantage of upwind methods over the central difference schemes is that these methods are naturally dissipative, so they do not require additional explicit spatial dissipation terms. Furthermore, it has been

shown (Ref. 31) that schemes based on upwind methods are much more robust than those based on central difference approaches. For the conservative form of the Euler equations, the flux-vector splitting scheme has shown many accurate and efficient applications with body-fitted grids, particulary at high Mach numbers (Refs. 24, 25).

We can develop a flux-vector splitting following Steger and Warming (Ref. 17), by diagonalizing the Jacobian matrix A as

$$\Lambda = T^{-1}AT \tag{30}$$

where Λ is a diagonal matrix containing the eigenvalues of A and T is a transformation matrix, whose columns are composed of the eigenvectors of A, defined in Ref. 17. For the two-dimensional Euler equations in Cartesian coordinates, eq. (1), we can obtain

$$\Lambda = \begin{bmatrix} \lambda_1 & 0 & 0 & 0 \\ 0 & \lambda_2 & 0 & 0 \\ 0 & 0 & \lambda_3 & 0 \\ 0 & 0 & 0 & \lambda_4 \end{bmatrix} \tag{31}$$

where $\lambda_1 = u$, $\lambda_2 = u$, $\lambda_3 = u + a$, $\lambda_4 = u - a$. Any eigenvalues λ_i (i = 1,2,3,4) can be expressed as

$$\lambda_i = \lambda_i^+ + \lambda_i^- \tag{32}$$

where λ_i^+ is always non-negative and λ_i^- is always non-positive. This is accomplished through

$$\lambda_i^+ = \frac{\lambda_i + |\lambda_i|}{2} \quad , \qquad \lambda_i^- = \frac{\lambda_i - |\lambda_i|}{2} \tag{33}$$

Then the diagonal matrix Λ can be split into a non-negative diagonal matrix and a non-positive diagonal matrix as

$$\Lambda = \Lambda^+ + \Lambda^- \tag{34}$$

where Λ^+ and Λ^- have as diagonal elements λ_i^+ and λ_i^-, respectively. The flux vectors $\vec{F}$ and $\vec{G}$ of the Euler equations have the interesting homogeneous property that

$$\vec{F} = A\vec{Q} \quad , \quad \vec{G} = B\vec{Q} \tag{35}$$

which can be verified by simply multiplying the indicated matrices. From eqs. (30) and (35), we may split $\vec{F}$ as follows

$$\begin{aligned} \vec{F} &= A\vec{Q} = T\Lambda T^-\vec{Q} = T(\Lambda^+ + \Lambda^-)T^-\vec{Q} \\ &= A^+\vec{Q} + A^-\vec{Q} \\ &= \vec{F}^+ + \vec{F}^- \end{aligned} \tag{36}$$

The split fluxes $\vec{F}^\pm$ are directly found for this case as

$$\vec{F}^\pm = [f_1^\pm, f_2^\pm, f_3^\pm, f_4^\pm]^T \tag{37}$$

with

$$f_1^{\pm} = \frac{\rho}{2\gamma}[2(\gamma - 1)\lambda_1^{\pm} + \lambda_3^{\pm} + \lambda_4^{\pm}]$$

$$f_2^{\pm} = uf_1^{\pm} + \frac{\rho a}{2\gamma}(\lambda_3^{\pm} - \lambda_4^{\pm})$$

$$f_3^{\pm} = vf_1^{\pm}$$

$$f_4^{\pm} = \frac{1}{2}(u^2 + v^2)f_1^{\pm} + \frac{\rho}{2\gamma}[\frac{a^2}{\gamma - 1}(\lambda_3^{\pm} + \lambda_4^{\pm}) + ua(\lambda_3^{\pm} - \lambda_4^{\pm})]$$

Similary we can split $\vec{G}$ as

$$\vec{G} = \vec{G}^{+} + \vec{G}^{-} \tag{38}$$

based on the eigenvalues and eigenvectors of B, eq. (10). The flux splitting derived above has a discontinuous first derivative when the eigenvalues change sign (i.e. sonic and stagnation points). This discontinuous property causes small oscillations at sonic points so that Steger redifined the eigenvalue splitting as

$$\lambda_i^{\pm} = \frac{\lambda_i \pm \sqrt{\lambda_i^2 + \varepsilon^2}}{2} \tag{39}$$

where ε is a small parameter to insure a smooth transition when the eigenvalues change sign. By properly choosing ε, the oscillation across the sonic point can be reduced (Ref. 19).

The flux splitting in eq. (36) is not unique. Van Leer (Ref. 18) suggested a different flux splitting with smooth transition when the eigenvalues change sign,

i.e. near sonic and stagnation points. It has been shown in Ref. 19 that this form of flux-vector splitting has smoothly varying flux contributions at the sonic and stagnation points. We utilize the Van Leer flux splitting for our numerical algorithm.

Following Van Leer splitting, $\vec{F}^{\pm}$ and $\vec{G}^{\pm}$ are given in terms of the local Mach number $M_x \equiv \frac{u}{a}$ and $M_y \equiv \frac{v}{a}$, where we consider the Mach number to have the sign of u and v, such that

$$\vec{F}^{+} = \vec{F}, \quad \vec{F}^{-} = 0 \qquad for \;\; M_x \geq 1 \tag{40}$$

$$\vec{F}^{+} = 0, \quad \vec{F}^{-} = \vec{F} \qquad for \;\; M_x \leq -1 \tag{41}$$

$$\vec{F}^{\pm} = \begin{bmatrix} \pm \rho a[\frac{1}{2}(M_x \pm 1)]^2 \\ \pm \rho a[\frac{1}{2}(M_x \pm 1)]^2[(\gamma - 1)u \pm 2a]/\gamma \\ \pm \rho a[\frac{1}{2}(M_x \pm 1)]^2 v \\ \pm \rho a[\frac{1}{2}(M_x \pm 1)]^2\{[(\gamma - 1)u \pm 2a]^2/[2(\gamma^2 - 1)] + v^2/2\} \end{bmatrix} \equiv \begin{bmatrix} f_1^{\pm} \\ f_2^{\pm} \\ f_3^{\pm} \\ f_4^{\pm} \end{bmatrix} \qquad for \;\; |M_x| < 1 \tag{42}$$

and

$$\vec{G}^{+} = \vec{G}, \quad \vec{G}^{-} = 0 \qquad for \quad M_y \geq 1 \tag{43}$$

$$\vec{G}^{+} = 0, \quad \vec{G}^{-} = \vec{G} \qquad for \quad M_y \leq -1 \tag{44}$$

$$\vec{G}^{\pm} = \begin{bmatrix} \pm \rho a[\frac{1}{2}(M_y \pm 1)]^2 \\ \pm \rho a[\frac{1}{2}(M_y \pm 1)]^2 u \\ \pm \rho a[\frac{1}{2}(M_y \pm 1)]^2[(\gamma - 1)v \pm 2a]/\gamma \\ \pm \rho a[\frac{1}{2}(M_y \pm 1)]^2\{[(\gamma - 1)v \pm 2a]/[2(\gamma^2 - 1)] + u^2/2\} \end{bmatrix} \equiv \begin{bmatrix} g_1^{\pm} \\ g_2^{\pm} \\ g_3^{\pm} \\ g_4^{\pm} \end{bmatrix} \qquad for \ |M_y| < 1 \tag{45}$$

where a is the local speed of sound. The Jacobian matrices A and B can be also split as

$$A = A^{+} + A^{-} = \frac{\partial \vec{F}^{+}}{\partial \vec{Q}} + \frac{\partial \vec{F}^{-}}{\partial \vec{Q}} \tag{46}$$

$$B = B^{+} + B^{-} = \frac{\partial \vec{G}^{+}}{\partial \vec{Q}} + \frac{\partial \vec{G}^{-}}{\partial \vec{Q}} \tag{47}$$

From eqs. (40) - (45), $A^{\pm}$ and $B^{\pm}$ can be written as

$$A^{+} = A, \quad A^{-} = 0 \qquad for\ M_x \geq 1 \tag{48}$$

$$A^{+} = 0, \quad A^{-} = \vec{A} \qquad for\ M_x \leq -1 \tag{49}$$

$$A^{\pm} = \frac{\partial \vec{F}^{\pm}}{\partial \vec{Q}} = \begin{bmatrix} \frac{\partial f_1^{\pm}}{\partial q_1} & \frac{\partial f_1^{\pm}}{\partial q_2} & \frac{\partial f_1^{\pm}}{\partial q_3} & \frac{\partial f_1^{\pm}}{\partial q_4} \\ \frac{\partial f_2^{\pm}}{\partial q_1} & \frac{\partial f_2^{\pm}}{\partial q_2} & \frac{\partial f_2^{\pm}}{\partial q_3} & \frac{\partial f_2^{\pm}}{\partial q_4} \\ \frac{\partial f_3^{\pm}}{\partial q_1} & \frac{\partial f_3^{\pm}}{\partial q_2} & \frac{\partial f_3^{\pm}}{\partial q_3} & \frac{\partial f_3^{\pm}}{\partial q_4} \\ \frac{\partial f_4^{\pm}}{\partial q_1} & \frac{\partial f_4^{\pm}}{\partial q_2} & \frac{\partial f_4^{\pm}}{\partial q_3} & \frac{\partial f_4^{\pm}}{\partial q_4} \end{bmatrix} \tag{50}$$

$$for\ |M_x| < 1$$

where

$$\frac{\partial f_1^{\pm}}{\partial q_1} = f_1^{\pm} \frac{1}{\rho} \frac{-M_x \pm 1}{M_x \pm 1} [1 + \frac{1}{2a^2} \gamma(\gamma - 1)(-E + u^2 + v^2)]$$

$$\frac{\partial f_1^{\pm}}{\partial q_2} = f_1^{\pm} \frac{1}{\rho a} \frac{1}{M_x \pm 1} [2 - \frac{1}{2} \gamma(\gamma - 1) M_x(-M_x \pm 1)]$$

$$\frac{\partial f_1^{\pm}}{\partial q_3} = f_1^{\pm} \frac{v}{2\rho a^2} \gamma(1 - \gamma) \frac{-M_x \pm 1}{M_x \pm 1}$$

$$\frac{\partial f_1^{\pm}}{\partial q_4} = f_1^{\pm}\frac{1}{2\rho a^2}\gamma(\gamma - 1)\frac{-M_x \pm 1}{M_x \pm 1}$$

$$\frac{\partial f_2^{\pm}}{\partial q_1} = \frac{\partial f_1^{\pm}}{\partial q_1}\frac{f_2^{\pm}}{f_1^{\pm}} + f_1^{\pm}\frac{(\gamma - 1)}{\rho\gamma}\left[-u \pm \frac{\gamma}{a}(-E + u^2 + v^2)\right]$$

$$\frac{\partial f_2^{\pm}}{\partial q_2} = \frac{\partial f_1^{\pm}}{\partial q_2}\frac{f_2^{\pm}}{f_1^{\pm}} - f_1^{\pm}\frac{\gamma - 1}{\rho\gamma}(-1 \pm \gamma M_x)$$

$$\frac{\partial f_2^{\pm}}{\partial q_3} = \frac{\partial f_1^{\pm}}{\partial q_3}\frac{f_2^{\pm}}{f_1^{\pm}} \pm f_1^{\pm}\frac{1 - \gamma}{\rho}\frac{v}{a}$$

$$\frac{\partial f_2^{\pm}}{\partial q_4} = \frac{\partial f_1^{\pm}}{\partial q_4}\frac{f_2^{\pm}}{f_1^{\pm}} \pm f_1^{\pm}\frac{\gamma - 1}{\rho a}$$

$$\frac{\partial f_3^{\pm}}{\partial q_1} = \frac{\partial f_1^{\pm}}{\partial q_1}v - f_1^{\pm}\frac{v}{\rho}$$

$$\frac{\partial f_3^{\pm}}{\partial q_2} = \frac{\partial f_1^{\pm}}{\partial q_2}v$$

$$\frac{\partial f_3^{\pm}}{\partial q_3} = \frac{\partial f_1^{\pm}}{\partial q_3}v + f_1^{\pm}\frac{1}{\rho}$$

$$\frac{\partial f_3^{\pm}}{\partial q_4} = \frac{\partial f_1^{\pm}}{\partial q_4}v$$

$$\frac{\partial f_4^{\pm}}{\partial q_1} = \frac{\partial f_1^{\pm}}{\partial q_1} \frac{f_4^{\pm}}{f_1^{\pm}}$$

$$+ f_1^{\pm} \frac{1}{\rho} \{ \frac{1}{\gamma + 1} [(\gamma - 1)u \pm 2a][- u \pm \frac{\gamma}{a}(- E + u^2 + v^2)] - v^2 \}$$

$$\frac{\partial f_4^{\pm}}{\partial q_2} = \frac{\partial f_1^{\pm}}{\partial q_2} \frac{f_4^{\pm}}{f_1^{\pm}} - f_1^{\pm} \frac{1}{\rho(\gamma + 1)} [(\gamma - 1)u \pm 2a][-1 \pm \gamma M_x]$$

$$\frac{\partial f_4^{\pm}}{\partial q_3} = \frac{\partial f_1^{\pm}}{\partial q_3} \frac{f_4^{\pm}}{f_1^{\pm}} - f_1^{\pm} \frac{v}{\rho} \{ \pm \frac{\gamma}{\gamma + 1} \frac{1}{a} [(\gamma - 1)u \pm 2a] + 1 \}$$

$$\frac{\partial f_4^{\pm}}{\partial q_4} = \frac{\partial f_1^{\pm}}{\partial q_4} \frac{f_4^{\pm}}{f_1^{\pm}} \pm f_1^{\pm} \frac{1}{\rho a} \frac{\gamma}{\gamma + 1} [(\gamma - 1)u \pm 2a]$$

and

$$B^+ = B, \quad B^- = 0 \qquad for\ M_y \geq 1 \tag{51}$$

$$B^+ = 0, \quad B^- = \vec{B} \qquad for\ M_y \leq -1 \tag{52}$$

$$B^{\pm} = \frac{\partial \vec{G}^{\pm}}{\partial \vec{Q}}$$

$$= \begin{bmatrix} \frac{\partial g_1^{\pm}}{\partial q_1} & \frac{\partial g_1^{\pm}}{\partial q_2} & \frac{\partial g_1^{\pm}}{\partial q_3} & \frac{\partial g_1^{\pm}}{\partial q_4} \\ \frac{\partial g_2^{\pm}}{\partial q_1} & \frac{\partial g_2^{\pm}}{\partial q_2} & \frac{\partial g_2^{\pm}}{\partial q_3} & \frac{\partial g_2^{\pm}}{\partial q_4} \\ \frac{\partial g_3^{\pm}}{\partial q_1} & \frac{\partial g_3^{\pm}}{\partial q_2} & \frac{\partial g_3^{\pm}}{\partial q_3} & \frac{\partial g_3^{\pm}}{\partial q_4} \\ \frac{\partial g_4^{\pm}}{\partial q_1} & \frac{\partial g_4^{\pm}}{\partial q_2} & \frac{\partial g_4^{\pm}}{\partial q_3} & \frac{\partial g_4^{\pm}}{\partial q_4} \end{bmatrix} \tag{53}$$

$$for \ |M_y| < 1$$

where

$$\frac{\partial g_1^{\pm}}{\partial q_1} = g_1^{\pm} \frac{1}{\rho} \frac{-M_y \pm 1}{M_y \pm 1} [1 + \frac{1}{2a^2} \gamma(\gamma - 1)(-E + u^2 + v^2)]$$

$$\frac{\partial g_1^{\pm}}{\partial q_2} = g_1^{\pm} \frac{u}{2\rho a^2} \gamma(1 - \gamma) \frac{-M_y \pm 1}{M_y \pm 1}$$

$$\frac{\partial g_1^{\pm}}{\partial q_3} = g_1^{\pm} \frac{1}{\rho a} \frac{1}{M_y \pm 1} [2 - \frac{1}{2} \gamma(\gamma - 1) M_y(-M_y \pm 1)]$$

$$\frac{\partial g_1^{\pm}}{\partial q_4} = g_1^{\pm} \frac{1}{2\rho a^2} \gamma(\gamma - 1) \frac{-M_y \pm 1}{M_y \pm 1}$$

$$\frac{\partial g_2^{\pm}}{\partial q_1} = \frac{\partial g_1^{\pm}}{\partial q_1} u - g_1^{\pm} \frac{u}{\rho}$$

$$\frac{\partial g_2^{\pm}}{\partial q_2} = \frac{\partial g_1^{\pm}}{\partial q_2} u + g_1^{\pm} \frac{1}{\rho}$$

$$\frac{\partial g_2^{\pm}}{\partial q_3} = \frac{\partial g_1^{\pm}}{\partial q_3} u$$

$$\frac{\partial g_2^{\pm}}{\partial q_4} = \frac{\partial g_1^{\pm}}{\partial q_4} u$$

$$\frac{\partial g_3^{\pm}}{\partial q_1} = \frac{\partial g_1^{\pm}}{\partial q_1} \frac{g_3^{\pm}}{g_1^{\pm}} + g_1^{\pm} \frac{(\gamma - 1)}{\rho\gamma} \left[- v \pm \frac{\gamma}{a} (- E + u^2 + v^2) \right]$$

$$\frac{\partial g_3^{\pm}}{\partial q_2} = \frac{\partial g_1^{\pm}}{\partial q_2} \frac{g_3^{\pm}}{g_1^{\pm}} \pm g_1^{\pm} \frac{1 - \gamma}{\rho} \frac{u}{a}$$

$$\frac{\partial g_3^{\pm}}{\partial q_3} = \frac{\partial g_1^{\pm}}{\partial q_3} \frac{g_3^{\pm}}{g_1^{\pm}} - g_1^{\pm} \frac{\gamma - 1}{\rho\gamma} (- 1 \pm \gamma M_y)$$

$$\frac{\partial g_3^{\pm}}{\partial q_4} = \frac{\partial g_1^{\pm}}{\partial q_4} \frac{g_3^{\pm}}{g_1^{\pm}} \pm g_1^{\pm} \frac{\gamma - 1}{\rho a}$$

$$\frac{\partial g_4^{\pm}}{\partial q_1} = \frac{\partial g_1^{\pm}}{\partial q_1} \frac{g_4^{\pm}}{g_1^{\pm}}$$
$$+ g_1^{\pm} \frac{1}{\rho} \left\{ \frac{1}{\gamma + 1} [(\gamma - 1)v \pm 2a] \left[- v \pm \frac{\gamma}{a} (- E + u^2 + v^2) \right] - u^2 \right\}$$

$$\frac{\partial g_4^{\pm}}{\partial q_2} = \frac{\partial g_1^{\pm}}{\partial q_2} \frac{g_4^{\pm}}{g_1^{\pm}} - g_1^{\pm} \frac{u}{\rho} \left\{ \pm \frac{\gamma}{\gamma + 1} \frac{1}{a} [(\gamma - 1)v \pm 2a] + 1 \right\}$$

$$\frac{\partial g_4^{\pm}}{\partial q_3} = \frac{\partial g_1^{\pm}}{\partial q_3}\frac{g_4^{\pm}}{g_1^{\pm}} - g_1^{\pm}\frac{1}{\rho(\gamma+1)}[(\gamma-1)v \pm 2a][-1 \pm \gamma M_y]$$

$$\frac{\partial g_4^{\pm}}{\partial q_4} = \frac{\partial g_1^{\pm}}{\partial q_4}\frac{g_4^{\pm}}{g_1^{\pm}} \pm g_1^{\pm}\frac{1}{\rho a}\frac{\gamma}{\gamma+1}[(\gamma-1)v \pm 2a]$$

In standard flux differencing, cell-side fluxes are obtained through the extrapolation of the cell-centered values of the fluxes. The fluxes are extrapolated in an "upwind" direction based on the signs of the appropriate eigenvalues. For example, $\vec{F}^{+}$ must be determined on the basis of information coming from the negative x (upwind) direction. Correspondingly $\vec{F}^{-}$ is determined on the basis of information coming from the positive x direction.

Van Leer, Ref. 32, suggested a variation to this approach, which is now called " MUSCL " differencing (based on Monotone Upwind Scalar Conservation Law). In this method, cell-side values of $\vec{Q}$ are obtained first from the extrapolation of the cell-centered values of $\vec{Q}$. The cell-side fluxes are then calculated from these extrapolated values of $\vec{Q}$. The extrapolations are of course done in an "upwind" sense. The MUSCL differencing has been shown in Ref. 19 to have superior shock-capturing properties, when applied to Steger and Warming, and Van Leer flux-vector splittings. If we apply this approach to the grid cell (i, j) , we can get

$$\vec{F}_{i\pm\frac{1}{2},j} = \vec{F}^{+}_{i\pm\frac{1}{2},j}(\vec{Q}^{-}_{i\pm\frac{1}{2},j}) + \vec{F}^{-}_{i\pm\frac{1}{2},j}(\vec{Q}^{+}_{i\pm\frac{1}{2},j}) \tag{54}$$

$$\vec{G}_{i,j\pm\frac{1}{2}} = \vec{G}^{+}_{i,j\pm\frac{1}{2}}(\vec{Q}^{-}_{i,j\pm\frac{1}{2}}) + \vec{G}^{-}_{i,j\pm\frac{1}{2}}(\vec{Q}^{+}_{i,j\pm\frac{1}{2}}) \tag{55}$$

Here, the cell interface values of $\vec{Q}^{\pm}$ are obtained through the extrapolation of cell-centered values, with $\vec{Q}^{-}$ extrapolated from the positive x direction and $\vec{Q}^{+}$ extrapolated from the negative x direction. Then, the cell side fluxes are determined from these extrapolated values. The extrapolation of $\vec{Q}$ can be written in operator notation (e.g., see Refs. 24,25) as

$$\vec{Q}^{-}_{i+\frac{1}{2}j} = \vec{Q}_{ij} + \frac{1}{4}\{\varphi^{-}[(1 - k_x)\Delta_x + (1 + k_x)\nabla_x]\}Q_{ij} \tag{56}$$

$$\vec{Q}^{+}_{i+\frac{1}{2}j} = \vec{Q}_{i+1j} - \frac{1}{4}\{\varphi^{+}[(1 - k_x)\nabla_x + (1 + k_x)\Delta_x]\}Q_{i+1j} \tag{57}$$

where

$$\Delta_x Q_{ij} \equiv \vec{Q}_{ij} - \vec{Q}_{i-1j}$$
$$\nabla_x Q_{ij} \equiv \vec{Q}_{i+1j} - \vec{Q}_{ij}$$

and,

$$\vec{Q}^{-}_{i,j+\frac{1}{2}} = \vec{Q}_{ij} + \frac{1}{4}\{\varphi^{-}[(1 - k_y)\Delta_y + (1 + k_y)\nabla_y]\}Q_{ij} \tag{58}$$

$$\vec{Q}^{+}_{i,j+\frac{1}{2}} = \vec{Q}_{ij+1} - \frac{1}{4}\{\varphi^{+}[(1 - k_y)\nabla_y + (1 + k_y)\Delta_y]\}Q_{ij+1} \tag{59}$$

where

$$\Delta_y Q_{i,j} \equiv \vec{Q}_{i,j} - \vec{Q}_{i,j-1}$$

$$\nabla_y Q_{i,j} \equiv \vec{Q}_{i,j+1} - \vec{Q}_{i,j}$$

The order of accuracy of the extrapolation is determined by the value of the switching function $\varphi^{\pm}$. For $\varphi^{\pm} = 0$, the approximation is first-order accurate in space. For $\varphi^{\pm} = 1$, k_x and k_y control the spatial accuracy. We have the fully upwind second-order scheme with $k_x = k_y = -1$. Values of $k_x = k_y = 1/3$ leads to a third-order upwind biased scheme. Spatial variation of $\varphi^{\pm}$ allows switching between the first-order and higher-order formulas, which is generally required to reduce oscillations near discontinuities, such as shock waves. The quantity $\varphi^{\pm}$ is called a limiter. A general discussion of limiters appears in Refs. 33, 12, 13. Several types of limiters were tested in Ref. 19. In this study, we only use a so-called the "catastrophic limiter", which was used in Ref. 24. With this limiter, wherever the higher-order approximation formulas result in a negative value of the square of the speed of sound, $\varphi^{\pm}$ is switched to 0 resulting in a first-order approximation at that point.

Applying the flux-vector splitting to our governing eq. (29), we have

$$
\begin{aligned}
&(\frac{\delta\vec{Q}}{\Delta t}S)_{ij} + [(A^+ + A^-)^n\delta\vec{Q}\Delta y]_{i+\frac{1}{2}j} - [(A^+ + A^-)^n\delta\vec{Q}\Delta y]_{i-\frac{1}{2}j} \\
&+ [(B^+ + B^-)^n\delta\vec{Q}\Delta x]_{ij+\frac{1}{2}} - [(B^+ + B^-)^n\delta\vec{Q}\Delta x]_{ij-\frac{1}{2}} + D_b D_l \delta\vec{Q}_{ij} \\
&= -[(\vec{F}^+ + \vec{F}^-)^n\Delta y]_{i+\frac{1}{2}j} - [(\vec{F}^+ + \vec{F}^-)^n\Delta y]_{i-\frac{1}{2}j} \\
&\quad - [(\vec{G}^+ + \vec{G}^-)^n\Delta x]_{ij+\frac{1}{2}} + [(\vec{G}^+ + \vec{G}^-)^n\Delta x]_{ij-\frac{1}{2}} \\
&\quad + \vec{P}_b
\end{aligned}
\tag{60}
$$

It may be noted that all the flux vectors and Jacobian matrices in eq. (60) on the cell sides $i \pm \frac{1}{2}, j \pm \frac{1}{2}$ are in the flux vector split form. The only remaining terms, $\vec{P}_b$ and $\vec{D}_b = \frac{\partial P_b}{\partial \vec{Q}_b}$ are the corresponding flux vector and Jacobian matrix on the body surface, incorporating the surface boundary condition. It remains an open question as whether these terms should be split also. In this work, both the split and unsplit forms will be evaluated. From Van Leer flux-vector splitting, p_b in the x-momentum equation can be split, for $|M_x| < 1$, such as

$$
p_b = p_b^+ + p_b^- \tag{61}
$$

where

$$
p_b^+ = \frac{\rho a^2}{4\gamma}(M_x + 1)^2(2 - M_x)
$$

$$
p_b^- = \frac{\rho a^2}{4\gamma}(M_x - 1)^2(2 + M_x)
$$

The pressure p_b in y-momentum equation can be also split, for $|M_y| < 1$, such as

$$p_b = p_b^+ + p_b^- \tag{62}$$

where

$$p_b^+ = \frac{\rho a^2}{4\gamma}(M_y + 1)^2(2 - M_y)$$

$$p_b^- = \frac{\rho a^2}{4\gamma}(M_y - 1)^2(2 + M_y)$$

One value of p_b, either p_b^+ or p_b^-, will depend on local body shape, and maybe obtained using second-order extrapolations of pressure, density and entalphy from the computational domain to the body surface. The other one (p_b^- or p_b^+) should be calculated from outside the computational domain (inside of the body). But there is no characteristic coming from inside the body, hence this should be prescribed on the boundary. In this study , we use the body cell-centered pressure.

Applying the delta form of eqs. (54) - (59) to eq. (60), the final form of our numerical equations for the (i, j) cell can be written as

$$\begin{aligned} &C_1\delta\vec{Q}_{i,j-2} + C_2\delta\vec{Q}_{i,j-1} + C_3\delta\vec{Q}_{i,j} + C_4\delta\vec{Q}_{i,j+1} + C_5\delta\vec{Q}_{i,j+2} \\ &C_6\delta\vec{Q}_{i-2,j} + C_7\delta\vec{Q}_{i-1,j} + C_8\delta\vec{Q}_{i+1,j} + C_9\delta\vec{Q}_{i+2,j} \\ &= \vec{R}_{i,j} \end{aligned} \tag{63}$$

where

$$C_1 = [\frac{\varphi}{4}(1 - k_x)A^+\Delta y]_{i-\frac{1}{2}j}$$

$$C_2 = -[\frac{\varphi}{4}(1 - k_x)A^+\Delta y]_{i+\frac{1}{2}j}$$
$$- \{[(1 - \frac{\varphi}{2}k_x)A^+ + \frac{\varphi}{4}(1 + k_x)A^-]\Delta y\}_{i-\frac{1}{2}j}$$

$$C_3 = \frac{S_{ij}}{\Delta t} + D_b D_l$$
$$+ \{[(1 - \frac{\varphi}{2}k_x)A^+ + \frac{\varphi}{4}(1 + k_x)A^-]\Delta y\}_{i+\frac{1}{2}j}$$
$$- \{[\frac{\varphi}{4}(1 + k_x)A^+ + (1 - \frac{\varphi}{2}k_x)A^-]\Delta y\}_{i-\frac{1}{2}j}$$
$$+ \{[(1 - \frac{\varphi}{2}k_y)B^+ + \frac{\varphi}{4}(1 - k_y)B^-]\Delta x\}_{ij+\frac{1}{2}}$$
$$- \{[\frac{\varphi}{4}(1 - k_y)B^+ + (1 - \frac{\varphi}{2}k_y)B^-]\Delta x\}_{ij-\frac{1}{2}}$$

$$C_4 = \{[\frac{\varphi}{4}(1 + k_x)A^+ + (1 - \frac{\varphi}{2}k_x)A^-]\Delta y\}_{i+\frac{1}{2}j}$$
$$+ [\frac{\varphi}{4}(1 - k_x)A^-\Delta y]_{i-\frac{1}{2}j}$$

$$C_5 = -[\frac{\varphi}{4}(1 - k_x)A^-\Delta y]_{i+\frac{1}{2}j}$$

$$C_6 = [\frac{\varphi}{4}(1 + k_y)B^+\Delta x]_{ij-\frac{1}{2}}$$

$$C_7 = -\left[\frac{\varphi}{4}(1 + k_y)B^+\Delta x\right]_{i,j+\frac{1}{2}}$$
$$- \left\{\left[(1 - \frac{\varphi}{2}k_y)B^+ + \frac{\varphi}{4}(1 - k_y)B^-\right]\Delta x\right\}_{i,j-\frac{1}{2}}$$

$$C_8 = \left\{\left[\frac{\varphi}{4}(1 - k_y)B^+ + (1 - \frac{\varphi}{2}k_y)B^-\right]\Delta x\right\}_{i,j+\frac{1}{2}}$$
$$+ \left[\frac{\varphi}{4}(1 + k_y)B^-\Delta x\right]_{i,j-\frac{1}{2}}$$

$$C_9 = -\left[\frac{\varphi}{4}(1 + k_y)B^-\Delta x\right]_{i,j+\frac{1}{2}}$$

$$\vec{R}_{i,j} = -\left[(\vec{F}^+ + \vec{F}^-)\Delta y\right]_{i+\frac{1}{2},j} + \left[(\vec{F}^+ + \vec{F}^-)\Delta y\right]_{i-\frac{1}{2},j}$$
$$- \left[(\vec{G}^+ + \vec{G}^-)\Delta x\right]_{i,j+\frac{1}{2}} + \left[(\vec{G}^+ + \vec{G}^-)\Delta x\right]_{i,j-\frac{1}{2}}$$
$$+ \vec{P}_b$$

Each of the C_1 - C_9 coefficients is a 4 X 4 matrix.

Applying eq. (63) over the entire set of grid cells, we have a linear system

$$M\,\delta\vec{Q} = \vec{N} \tag{64}$$

where M is a banded, block coefficient matrix and $\vec{N}$ is the right-hand side of eq. (14). Direct solution of eq. (64) is not efficient in general. Discussion of the iterative solution of these equations is found in Refs. 24 - 27. Vertical line Gauss - Seidel relaxation is applied in the present study. For subsonic and transonic flows, alternate sweeping direction is employed to have unconditional stability for

higher-order approximations (Refs. 19, 25). For a forward sweep, eq. (63) can be written as

$$C_1\delta\vec{Q}_{i,j-2} + C_2\delta\vec{Q}_{i,j-1} + C_3\delta\vec{Q}_{i,j} + C_4\delta\vec{Q}_{i,j+1} + C_5\delta\vec{Q}_{i,j+2} + \\ = \vec{R}_{i,j} - C_6\delta\vec{Q}_{i-2,j} - C_7\delta\vec{Q}_{i-1,j} \tag{65}$$

and for a backward sweep as

$$C_1\delta\vec{Q}_{i,j-2} + C_2\delta\vec{Q}_{i,j-1} + C_3\delta\vec{Q}_{i,j} + C_4\delta\vec{Q}_{i,j+1} + C_5\delta\vec{Q}_{i,j+2} + \\ = \vec{R}_{i,j} - C_8\delta\vec{Q}_{i+1,j} - C_9\delta\vec{Q}_{i+2,j} \tag{66}$$

The coefficient matrix M in eq. (64) becomes a block penta-diagonal matrix when we use eq. (65) or (66). Furthermore, M becomes a block tri-diagonal matrix for the first-order approximation. Either block penta-diagonal or tri-diagonal system can be solved effectively using LU decomposition methods. For a supersonic flow, a local iteration technique, Ref. 24, was found to be much more efficient than global iteration. In this procedure, when the flow is wholly supersonic, no downstream terms (i+1 or i+2) appear with the first- or second-order flux splitting. Hence the steady-state solution may be obtained by "local" iteration on the non-linear terms before moving to the next downstream line.

2.5 *Far-field Boundary Conditions*

For the conservation form of the Euler equation presented earlier, it can be shown that the characteristic speeds (eigenvalues of Jacobian matrices A and B) are u, u, $u + a$, $u - a$ from A and v, v, $v + a$, $v - a$ from B. Assuming locally isentropic flow near a far-field boundary, we can show that

$$\frac{dJ^{\pm}}{dt} = 0$$

along the characteristic paths

$$\frac{d\bar{x}^{\pm}}{dt} = \bar{u} \pm a$$

where $J^{\pm}$ is the Riemann invariant defined as

$$J^{\pm} \equiv \bar{u} \pm \frac{2a}{\gamma - 1} \tag{67}$$

where $\bar{x}$ is a local coordinate normal to the far-field boundary (defined positive going outward to the far-field), $\bar{u}$ is a local velocity along the $\bar{x}$ axis and a is a local speed of sound.

For supersonic inflows, all the four characteristics come from the known far-field free-steams. Hence all the $q_1 - q_4$ can be prescribed. For supersonic outflows, all the characteristics come from the computational domain, and hence, all the necessary quantities are extrapolated from the computational domain. In

this flow regime, one does not need long distances between the body and the far-field boundary in general. Thus, one can have very good grid resolution near the body even with a uniform Cartesian grid.

For subsonic flows, to use known far-field free stream values at the far-field boundary, a large distance from the body is required. For example, one needs typically at least 50 chord lengths from the body for subsonic or transonic airfoils in general. In a body-fitted, mapped grid system this is not difficult with a moderate number of grid points and a strong grid streching (though it may result in low accuracy). But with our non-mapped, non-body-fitted, Cartesian grid , one may expect a grid resolution problem near the body even with grid streching. To reduce the problem in subsonic or transonic flow regimes, the characteristic type of far-field boundary conditions (Ref. 34) which have been shown to be accurate with less than a 10 chord lengths distance between the body and the far-field for subsonic and transonic lifting airfoils (Ref. 35) can be applied.

For a subsonic inflow, three characteristics come into the computational domain from the far-field and one characteristic leaves the computational domain. For a subsonic outflow, one characteristic comes into the domain from the far-field and three characteristics leave the domain. The Riemann invariant J^- , which propagates along with the characteristic coming into the domain, can be determined from known far-field free-stream quantities. The other Riemann invariant J^+, which propagates along with the characteristic leaving the domain, can be extrapolated from the computed quantities. Assuming that $\frac{\partial(\ )}{\partial \bar{y}} = 0$ along the far-field boundary, one can prescribe the entropy s and the velocity

component tangent to the boundary $\bar{v}$, from known free-stream values (inflow) or extrapolate them from computed values (outflow). Here $\bar{y}$ is the local coordinate tangent to the boundary. The non-dimensional entropy s is defined as

$$s \equiv e^{\frac{s^* - s^*_\infty}{c_v}}$$

where s^* is the dimensional entropy and c_v is the specific heat per unit mass at constant volume. Using the present non-dimensional variables, s can be written as

$$s = \gamma M_\infty^2 \frac{p}{\rho^\gamma} \tag{68}$$

From these four quantities, we can determined $q_1 - q_4$. The detailed procedure is explained in the following, with the left inflow boundary taken as an example. At the left inflow boundary , we have $\bar{x} = -x$, $\bar{u} = -u$. Two Riemann invariants $J^\pm$ are determined as following. Along the incoming characteristics to the computational domain we have

$$\begin{aligned} J^- &= \bar{u}_f - \frac{2a_f}{\gamma - 1} \\ &= -u_\infty - \frac{2a_\infty}{\gamma - 1} \\ &= -u_\infty - \frac{2}{(\gamma - 1)M_\infty} \end{aligned} \tag{69}$$

where the non-dimensional $a_\infty = 1/M_\infty$. Following the outgoing characteristics from the domain, we determine

$$
\begin{aligned}
J^+ &= \bar{u}_f + \frac{2a_f}{\gamma - 1} \\
&= -u_c + \frac{2a_c}{\gamma - 1}
\end{aligned}
\tag{70}
$$

where subscript f denotes the quantities at the far-field boundary grid points and c denotes the extrapolated quantities from the computational domain. By adding and substracting J^+ and J^-, $u_f(= -\bar{u}_f)$ and a_f are determined such as

$$u_f = -\frac{1}{2}(J^+ + J^-) \tag{71}$$

$$a_f = \frac{\gamma - 1}{4}(J^+ - J^-) \tag{72}$$

v_f and s_f are prescribed from free-stream values such as

$$v_f = v_\infty \tag{73}$$

and

$$
\begin{aligned}
s_f = s_\infty &= \gamma M_\infty^2 \frac{p_\infty}{\rho_\infty^\gamma} \\
&= \gamma M_\infty^2 \frac{p_f}{\rho_f^\gamma} \\
&= M_\infty^2 \frac{a_f^2}{\rho_f^{\gamma-1}}
\end{aligned}
\tag{74}
$$

From eq. (74), we get

$$
\rho_f = \left(\frac{M_\infty^2 a_f^2}{s_f}\right)^{\frac{1}{\gamma-1}} \tag{75}
$$

Finally from eqs. (71), (72), and (73), the total energy at the boundary is determined as

$$
E_f = \frac{a_f^2}{\gamma(\gamma-1)} + \frac{1}{2}(u_f^2 + v_f^2) \tag{76}
$$

3.0 Results and Discussion

In order to verify the procedures developed here, several calculations of two-dimensional flows over a variety of geometries have been performed.

The first problem considered is the inviscid shock reflection on a flat plate to validate the numerical algorithm without additional geometric considerations. An incident shock wave impinges on a flat plate at an angle of 29° and free stream Mach number, M_∞ , of 2.9. Extensive computational tests have been done with this geometrically simple , but complicated flow (near the reflection point). The effects of accuracy, which is controlled by the order of the approximation used to calculate the cell-side fluxes, is tested with and without body surface-pressure splitting (explained in section 2-4). A density contour plot in Fig. 4 is obtained using a first-order spatial approximation with a 41 by 41 uniform grid. Incident and reflected shocks are captured reasonably well, but they are abnormally deflected (toward the inflow direction), as seen in the *isopycnics* (constant density lines) near the reflection point. The constant density

lines should be normal to the body surface for this flow. To illustrate grid size effects, a denser 81 by 41 uniform grid was utilized. The resulting density contours are shown in Fig. 5. Although the improved shock capturing is obvious, the constant density lines near the reflection point are again deflected. This same pattern was also found in a body-fitted upwind calculation (Ref. 36) and a finite element calculation (Ref. 37). The calculation was repeated, but this time using second-order spatial accuracy on the coarser, 41 by 41 grid. The resulting density contour plot, Fig. 6, shows excellent shock capturing results. Furthermore, the abnormally deflected constant density lines have disappeared. The effects of the order of approximation employed to calculate the cell-side fluxes on the accuracy of the steady-state solution are clearly evident. We believe the deflected contour lines near the reflection point are due to the large numerical dissipation of the first-order spatial approximation. Next, the first set of calculations were repeated, this time with body surface-pressure splitting. The resulting density contour plots are shown in Figs. 7,8 and 9. The contour plot in Fig. 7 corresponds to a first-order spatial approximation on a 41 by 41 grid. The deflected constant density lines, in the Fig. 4, are obviously improved compare to those without surface pressure splitting. A similar tendency is also observed in Fig. 8 on the finer 81 by 41 grid. However, the contour plot for second-order spatial approximation is almost the same for both the unsplit (Fig. 6) and split (Fig. 9) body surface pressure. The density contours in Fig. 10 from the second-order spatial approximation on the dense 81 by 41 grid show excellent shock capturing ability of the employed flux-vector splitting scheme. The effects of the accuracy

of the spatial approximationson on the body surface pressure distributions are shown in Figs. 11, 12 and 13 and compared with exact solution. The body surface pressure distributions with and without surface-pressure splitting (on a 41 by 41 grid) , in Fig. 11, again show big differences between the first- and second-order spatial approximations. These results illustrate the oscillation-free shock capturing of the first-order spatial approximation. As mentioned earlier, the "catastrophic" limiter , when employed with a second-order spatial approximation, reduces oscillations across the shocks, but is not totally oscillation-free near shocks. This limiter is employed here, not only because of its simplicity, but also because it requires less computer storage. With more computer storage, stronger limiters (Refs. 12, 13, 19, 33) can be utilized. The effect of surface-pressure splitting on the body surface pressure distributions is negligible for the first-order calculations (Fig. 12). In the second-order calulations, Fig. 13 , the surface-pressure splitting reduces the oscillations noticeably. The agreement between exact values and computed values is very good. Typical convergence histories of the local iteration technique are shown in Fig. 14 for both unsplit and split surface pressure, p_b terms. With the unsplit p_b terms, the convergence is quadratic. This is to be expected, since for supersonic flows, the upwind relaxation scheme, with Euler implicit time differencing, becomes a Newton-method (in the limit of the time step approaching infinity). However, with the split p_b terms, the convergence is reduced to being almost linear. Apparently, the body surface pressure splitting results in a deviation from conservation form for the body cells, which is found to slow convergence. The

required computation time to achive convergence (L_2norm of the residual to machine zero) for unsplit and split p_b terms were 36 and 58 cpu seconds, respectively. All the computations which will be reported here, have been performed on an IBM 3090 computer. According to the results shown above, the pressure splitting is desirable with the present limiter to reduce the oscillations across the shock wave, but definitely requires more computation time. The surface-pressure splitting formulation have been utilized with the remaining computation, discussed below.

The second test case is a supersonic flow, $M_\infty = 6.57$, over a blunt body. These calculations are utilized in order to investigate the accuracy of the non-body-fitted cartesian grid and the shock resolution of the flux-vector splitting scheme. The configuration of the blunt body and a part of the computational grid are shown in Fig. 15. A 76 by 101 uniform grid is used to form the computational grid. The density contours from the first-order calculation are shown in Fig. 16. The second-order calculation in Fig. 17, again shows much better accuracy. The pressure distributions, from the first- and second-order calculations, along the symmetry plane are shown in Fig. 18. Both the first- and second-order calculations show excellent shock-capturing with one grid difference in shock locations on the symmetry plane. We believe this difference is due to the combined effects of different numerical dissipations and the present limiter. The normal shock on the symmetry plane is well captured, with only one grid within the discontinuous region, for both the first- and second-order calculations. The pressure distributions, from the first- and second-order approximations, along the

outflow plane are shown in Fig. 19. The bow shock is oblique at this station and is captured within 9 grid points with the first-order calculation and 7 grid points with the second-order calculation. The body surface pressure distribution of the second-order calculation is given in Fig. 20 and compared with a method of lines, shock-fitting calculation with a body-fitted grid (Ref. 38). Although the grid cells near the body are irregular and discontinuous, as illustrated in Fig. 15, the resulting pressure distribution along the body surface is smooth and shows good agreement with the body-fitted grid calculation. This calculation served as a good test problem for the grid generation method and pressure extrapolation technique. The results verify the grid merging method (merging small body cells into adjacent large cells in $\pm$ x or $\pm$ y direction, depending on the local body slope in the small cells) and body surface pressure extrapolation method (extrapolating either $\pm$ x or $\pm$ y direction, depending on the body slope of the computational cells). Due to the embedded subsonic region, a global iteration technique was used. This problem required approximately 8 seconds of cpu per iteration, along with 300 iterations for the first-order approximation to achive convergence.

The third problem set considered is the supersonic flow through a two-dimensional scramjet inlet, shown in Fig. 21 a. This set of calculations was performed in order to illustrate the ability of the method to calculate complex shock reflections. This problem set was also utilized to show the geometric flexibility of the technique, by performing the calculation both with and without a bleed door. The details of the grids used are in Figs. 21 b and c. The first

computation was for an inflow Mach number of 2.2 on the smooth (closed door) contour with second-order spatial accuracy on a uniform 55 by 59 grid. The formation of Mach reflections are seen in the pressure contour plot of this flow field in Fig. 22 a. These results are qualitatively similar to those of Ref. 39, which used a Navier-Stokes computation, and are reproduced here in Fig. 22 b. Of course, we would expect the flow fields from our inviscid results to be similar to viscous calculations, only when there is no significant flow separation.

Next, the effect of reducing the inflow Mach number to 1.9 is considered. The resulting pressure contours, shown in Fig. 23 a, indicate aerodynamic choking. The same trend is seen in the viscous calculation from Ref. 39, which is reproduced as Fig. 23 b.

We then considered several inlet calculations with the bleed door opened. The resulting pressure contours for inflow Mach numbers of 2.2 and 1.9 are shown in Figs. 24 a and b. It should be noted that the Mach 2.2 calculation was performed with second-order spatial accuracy, whereas, the Mach 1.9 case utilized first-order accuracy. The motivation for using first-order method was only to reduce computational cost, not because of any other difficulties. For example, in the inlet calculations, a typical first-order calculation required approximately 200 iterations (15 min. cpu time) for convergence, whereas the second-order calculations required approximately 300 iterations (30 min. cpu time). These calculations not only suggest a possible way to control aerodynamic choking and formation of Mach reflection, but also illustrate the ability of the present method to handle very complex geometries. Changing to a multi-connected domain was

very simple in the present code. This would have been, in general, a very significant change for a body-fitted grid.

All the computations described so far, are for supersonic flow problems. The final problem set considered is the subsonic and transonic non-lifting flow over a NACA 0012 airfoil. A 41 by 31 uniform inner grid is used to wrap the airfoil and to generate body cells. From the outer-edge of the inner grid , grid stretching is used to the far-field boundary points with 5 % and 11 % continuous increment of grid size in x and y directions, respectively. With overall 119 by 109 grid points, there are about 3 chord lengths between the leading edge and left far-field boundary and about 2 chord lengths between the camber line and top far-field boundary. A sketch of the computational domain is shown in Fig. 25. Although the far-field boundary is applied very close to the airfoil, the characteristic far-field boundary conditions, explained previously , result in very accurate results. In Fig. 26, the body surface pressure coefficient distribution for $M_\infty = 0.5$ is compared with the potential calulation on a Cartesian grid in Ref. 7 , which reported having good agreement with other body-fitted potential code results. The body surface-pressure-coefficient distribution for a transonic flow, $M_\infty = 0.8$, is shown in Fig. 27 along with the results from two other Euler solutions. The Euler solutions utilize a central difference, explicit artificial viscosity algorithm, one with a body-fitted grid (FLO52S, similar to that described in Ref. 40), and the other with a Cartesian grid (Ref. 10), similar to that used here. Comparing the two Cartesian grid calculations in Fig. 27, shows our result to have a slightly sharper shock compared to those in Ref. 10. This is

likely due to the upwind difference scheme used here. However other slight discrepancies in shock location may be due different grid spacing (their grid was uniform, whereas we utilized far-field stretching). Aside from the minor shock location differences, the three calculations are in excellent agreement. This case required approximately 600 iterations (180 min. cpu time) to achieve convergence.

For the cases discussed here, the new numerical method performs reasonably well and shows good agreement with exact solutions and other methods with body-fitted grids. The blunt body results compare favorably with the shock-fitted Euler calculation (Ref. 38), and the supersonic inlet calculation shows a very similar flow structure with viscous calculations (Ref. 39). The subsonic and transonic airfoil calculations agree quite well with other Cartesian grid methods, such as the potential solution by Wedan and South (Ref. 7), and the central-difference, Runge-Kutta time-stepped Euler solution by Clark et al. (Ref. 10).

4.0 Concluding Remarks

The objective of the present study was to develop a computational procedure for solving the Euler equations with an "upwind" scheme using a non-body conforming Cartesian grid. The procedure was required to be accurate, efficient and applicable to a wide variety of complex geometries.

To achieve this goal, an implementation of the Van Leer flux-vector splitting scheme with the Cartesian finite-volume method was developed. This approach leads to a single vector set of governing difference equations which contained the surface boundary conditions. The grid was generated using a simple Cartesian grid, without any complicated grid generation procedures. An efficient iterative relaxation procedure was used to obtain steady-state solutions.

The resulting method was easy to implement for a wide variety of geometries. No new grid generation was required, even for multiply connected domains. Accurate solutions were obtained for several transonic and supersonic test cases.

The results compared favorably the other numerical methods which required body-fitted grids.

Some drawbacks of the present method include the use of a non-optimal number of grids points. Many grid cells are wasted inside the body, and if (one-dimensional) grid stretching is used, fine grids will unnecessarily extend to the far-field. In extreme cases, perhaps as much as 20 - 40 % more grid points (or cell volumes) must be utilized, as compared to a body-fitted grid. We feel that the simplicity of the grid generation more than compensates for this disadvantage.

A more severe problem with this approach is that it is not easily extended to viscous flows. The fact that a single coordinate line does not follow the body surface contour, (as in a body-fitted grid), means that one cannot efficiently stretch the grid in a direction normal to the surface in order to resolve viscous boundary layers. Also, not having a coordinate line along the body makes it very difficult to to use simplifying "thin-layer" approximations.

Nonetheless, the approach utilized in this study will still be very useful for inviscid Euler solutions. The method has been shown to be accurate, efficient and robust. Upon extension to three dimensions, this approach should be particulary useful in a preliminary design mode, where flows past a wide variety of complex geometries can be computed without complicated grid generation procedures.

References

1. Thompson, J.F., Warsi, Z.U.A. and Mastin, C.W., *Numerical Grid Generation,* Elsevier Science Publishing Co., Inc., NewYork 1985.

2. Eiseman, P.R., "Grid Generation for Fluid Mechanics Computation," *Ann. Rev. Fluid Mech.,* Vol. 17, 1985, pp.487-522.

3. Moretti, G., "Grid Generation Using Classical Techniques," in "Numerical Grid Generation Techniques," NASA CP 2166, 1980, pp. 1-35.

4. Rai, M.M., "A Conservative Treatment of Zonal Boundaries for Euler Equation Calculations," *J. Comput. Phys.,* Vol. 62, 1986, pp. 472-503.

5. Walters, R.W., Thomas, J.L. and Switzer, G.F., "Aspects and Applications of Patched Grid Schemes," AIAA Paper 86-1063, May 1986.

6. Sorenson, R.L., "Three-Dimensional Elliptic Grid Generation About Fighter Aircraft for Zonal Finite Difference Computations," AIAA Paper 86-0429, January 1986.

7. Wedan, B. and South, J.C., "A Method for Solving the Transonic Full-Potential Equation for General Configurations," AIAA Computational Fluid Dynamics Conference Proceedings, July 1983, pp. 515-526.

8. Purvis, J.W. and Burkhalter, J.E., "Prediction of Critical Mach Number for Store Configurations," *AIAA J.,* Vol. 17, November 1979, pp. 1170-1177.

9. Grossman, B. and Whitaker, D.L., "Supersonic Flow Computations Using a Rectangular-Coordinate Finite-Volume Method," AIAA Paper 86-0442, January 1986.

10. Clarke, D.K., Hassan, H.A. and Salas, M.D., "Euler Calculations for Multielement Airfoils Using Cartesian Grids," *AIAA J.*, Vol. 24, No. 3, 1986, pp. 353-358.

11. Gaffney, R.L., Hassan, H.A. and Salas, M.D., "Euler Calculations for Wings Using Cartesian Grids," AIAA Paper 87-0356, January 1987.

12. Harten, A., Lax, P.D. and Van Leer, B., "On Upstream Differencing and Godnunov-Type Schemes for Hyperbolic Conservation Laws," *SIAM Review,* Vol. 25, No. 1, January 1983, pp. 35-61.

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

14. Courant, R., Isaacson, E. and Rees, M., "On the solution of Non-linear Hyperbolic Differential Equations by Finite Differences," *Communications Pure and Applied Math.,* Vol. 5, 1952, pp. 243-249.

15. Godunov, S.K., "A Finite Difference Method for the Numerical Computation of Discontinuous Solutions of the Equations of Fluid Dynamics," *Mat. sb.,* Vol. 47, 1959, pp. 357-393.

16. Moretti, G., "The λ - Scheme," *Computers and Fluids,* Vol. 7, 1979, pp. 191-205.

17. Steger, J.L. and Warming, R.F., "Flux Vector Splitting of the Inviscid Gasdynamic Equations with Application to Finite-Difference Methods," *J. Comput. Phys.,* Vol. 40, 1981, pp. 263-293.

18. Van Leer, B., "Flux-Vector Splitting for the Euler Equations," *Lecture Notes in Physics,* Vol. 170, 1982, pp. 507-512. (also ICASE Rept. 82-30, September 1982)

19. Anderson, W.K., Thomas, J.L. and Van Leer, B. "A Comparison of Finite Volume Flux Vector Splitting for the Euler Equations," *AIAA J.,* Vol. 24, No. 9, 1986, pp. 1453-1460.

20. Roe, P.L., "Approximate Riemann Solvers, Parameter Vectors, and Difference Schemes," *J. Comput. Phys.,* Vol. 43, 1981, pp. 357-372.

21. Osher, S., "Riemann Solvers, the Entropy Condition and Difference Approximations," *SIAM J. Numer. Anal.,* Vol. 21, 1984, pp. 217-235.

22. Harten, A., "High Resolution Schemes for Hyperbolic Conservation Laws," *J. Comput. Phys.,* Vol. 49, 1983, pp. 357-393.

23. Carkravarthy, S.R., Harten, A. and Osher, S., "Essentially Non-Oscillatory Shock Capturing Schemes of Uniformly Very High Accuracy," AIAA Paper 86-0339, January 1986.

24. Walters, R.W. and Dwoyer, D.L., "An Efficient Iteration Strategy Based on Upwind/Relaxation Schemes for the Euler Equations," AIAA Paper 85-1529-CP, July 1985.

25. Thomas, J.L., Van Leer, B. and Walters, R.W., "Implicit Flux-Split Schemes for the Euler Equations," AIAA Paper 85-1680, July 1985.

26. Van Leer, B. and Mulder, W.A., "Relaxation Methods for Hyperbolic Equations," Report 84-20, Delft University of Technology, 1984.

27. Chakravarthy, S.R., "Relaxation Methods for Unfactored Implicit Schemes," AIAA Paper 84-0165, January 1984.

28. Roe, P.L. and Pike, J., "Efficient Construction and Utilization of Approximate Riemann Solutions," *Computing Methods in Applied Sciences and Engineering, VI,* Elsevier Science Publishers, INRIA, 1984.

29. Moretti, G. and Zannetti, L., "A New, Improved Computational Technique for Two-Dimensional, Unsteady compressible Flows," *AIAA J.,* Vol. 22, 1984, pp. 758-766.

30. Yee, H.C., Warming, R.F. and Harten, A., "Implicit Total-Variation-Diminishing (TVD) Schemes for Steady State Calculations," *J. Comput. Phys.,* Vol. 57, 1985, pp.327-360.

31. Van Leer, B., "Upwind-Difference Methods for Aerodynamic Problems Governed by the Euler Equations," *Lectures in Appl. Math.,* Vol. 22, 1985, pp. 327-336.

32. Van Leer, B., "Towards the Ultimate Conservative Difference Scheme, II., Monotonicity and Conservation Combined in a Second-Order Scheme," *J. Comput. Phys.,* Vol. 14, 1974, pp. 361-376.

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

34. Salas, M.D., Jameson, A. and Melnik, R.E., "A Comparative Study of Non-Uniqueness of the Potential Equation," AIAA Paper 83-1888, 1983.

35. Thomas, J.L. and Salas, M.D., "Far-Field Boundary Conditions for Transonic Lifting Solutions to the Euler Equations," *AIAA J.*, Vol. 24, No. 7, 1986, pp. 1074-1080.

36. Walters, R.W., Virginia Polytechnic Institute and State University, private communication.

37. Lohner, R., Morgan, K., Peraire, J. and Zienkiewicz, O.C., "Finite Element Methods for High Speeds Flows," AIAA Paper 85-1531, July 1985.

38. Hamilton III, H.H., "Solution of Axisymmetric and Two-Dimensional Inviscid Flow over Blunt Bodies by the Method of Lines," NASA TP 1154, April 1978.

39. Kumar, A., "Mach Reflection and Aerodynamic Choking in Two-Dimensional Ducted Flow," *AIAA J.*, Vol. 24, April 1986, pp. 695-697.

40. Jameson, A., Schmit, W. and Turkel, E., "Numerical Solutions of the Euler Equations Using Runge-Kutta Time-Stepping Schemes," AIAA Paper 81-1259, 1981.

Figures

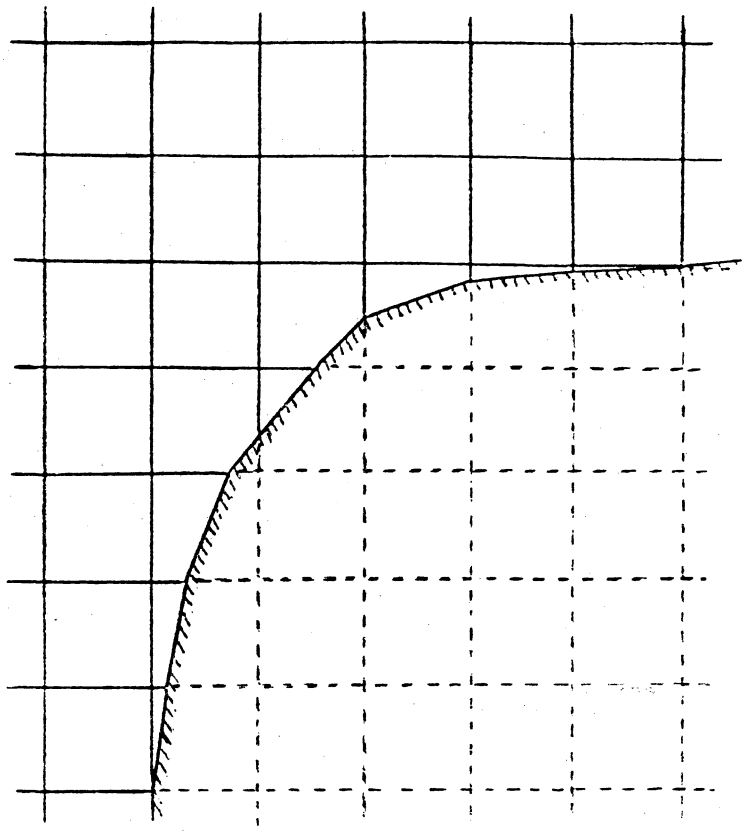


Figure 1. Intermediate Grid Cells near a Blunt Body.

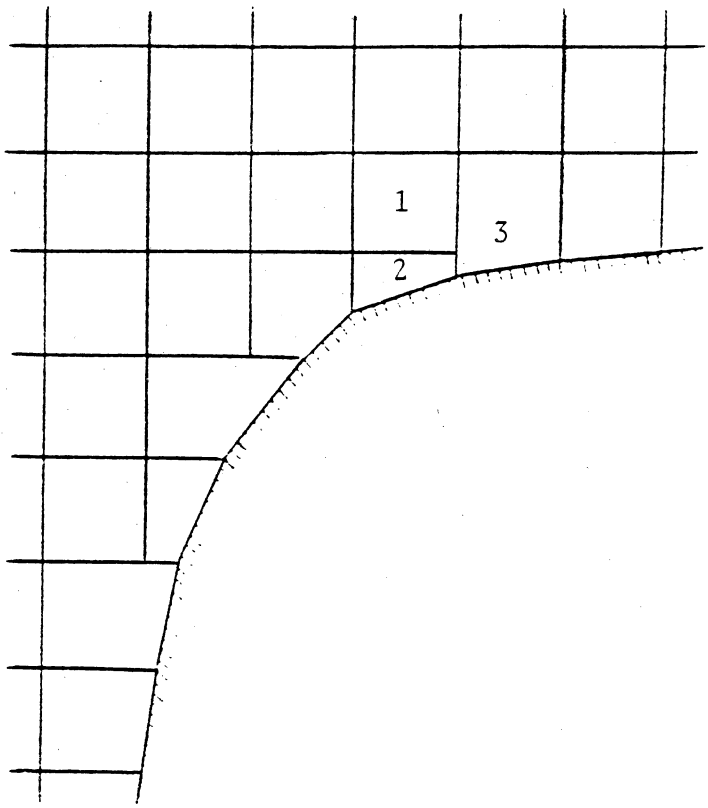
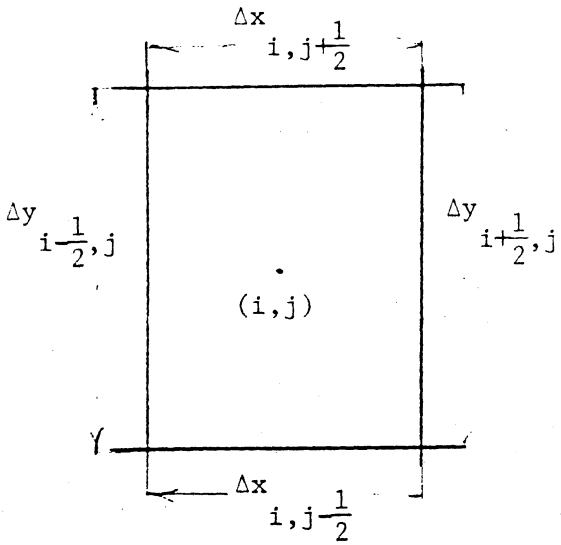
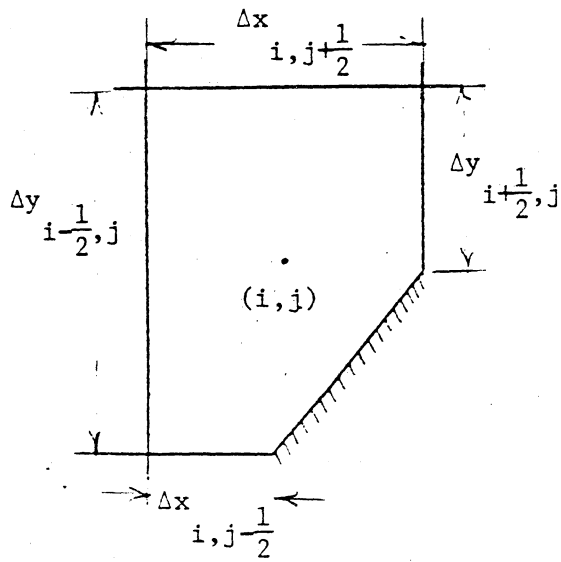


Figure 2. Computational Grid Cells near a Blunt Body.



a) Interior cell



b) Body cell

Figure 3. Grid Cell Definitions.

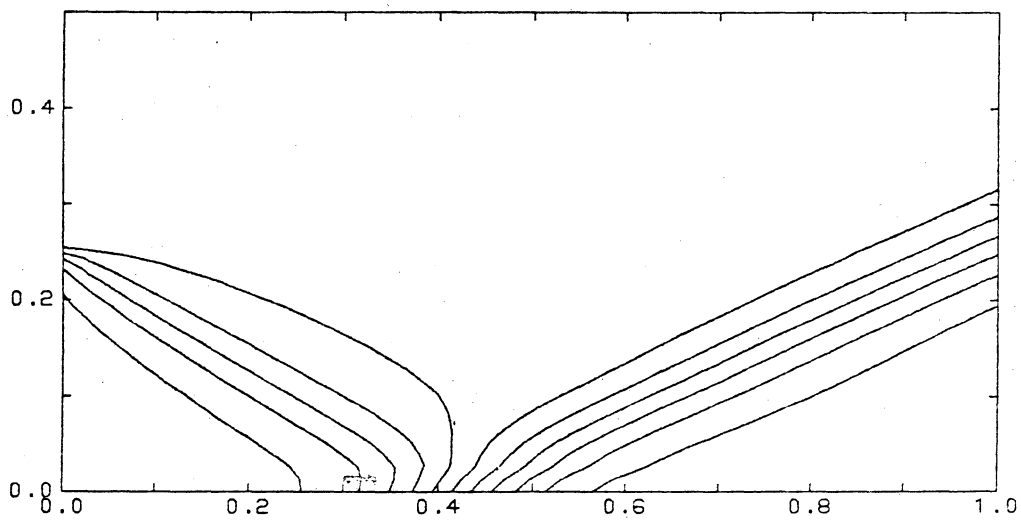


Figure 4. Density Contours - Shock Reflection of a Flat Plate: $M_\infty = 2.9$, 1st-order Upwind with Unsplit p_b, 41 by 41 Grid.

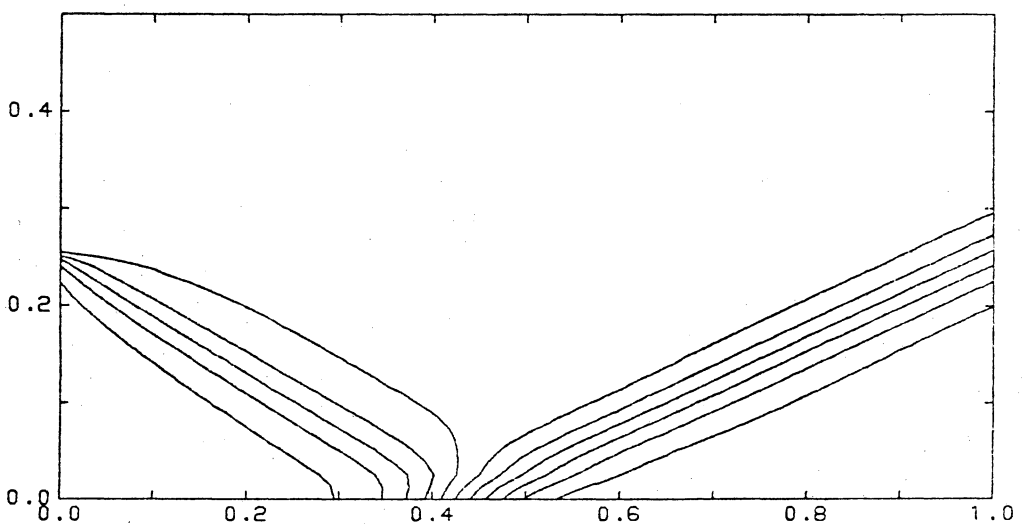


Figure 5. Density Contours - Shock Reflection of a Flat Plate: $M_\infty = 2.9$, 1st-order Upwind with Unsplit p_h, 81 by 41 Grid.

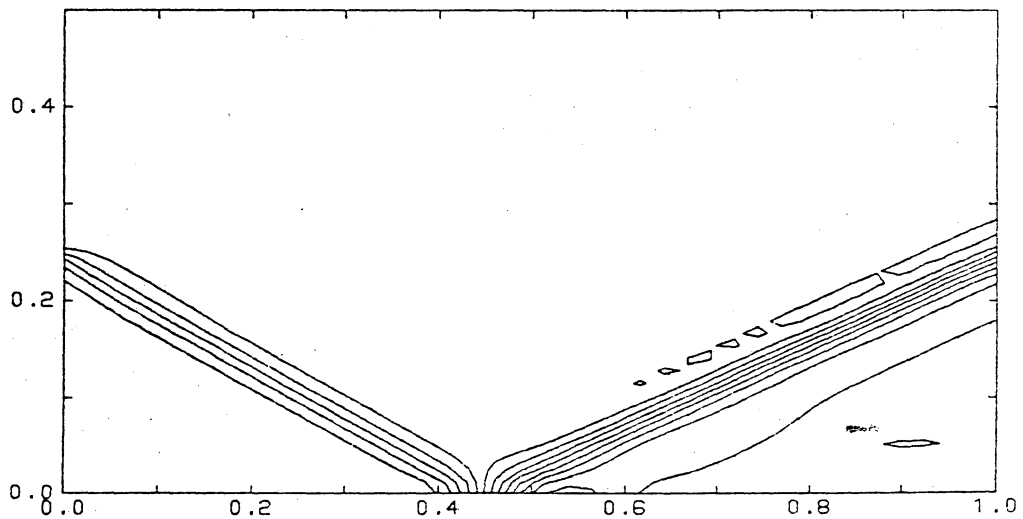


Figure 6. Density Contours - Shock Reflection of a Flat Plate: $M_\infty = 2.9$, 2nd-order Upwind with Unsplit p_b, 41 by 41 Grid.

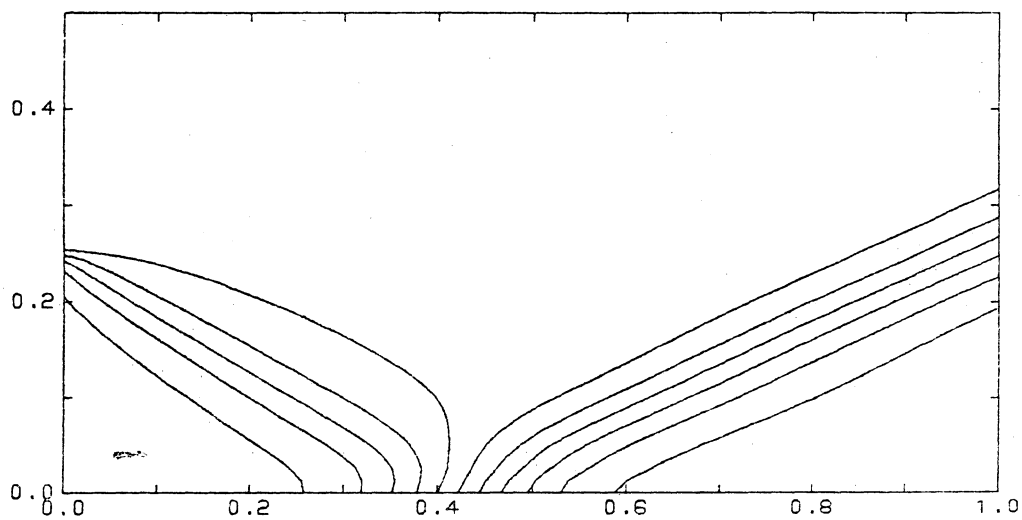


Figure 7. Density Contours - Shock Reflection of a Flat Plate: $M_\infty = 2.9$, 1st-order Upwind with Split p_b, 41 by 41 Grid.

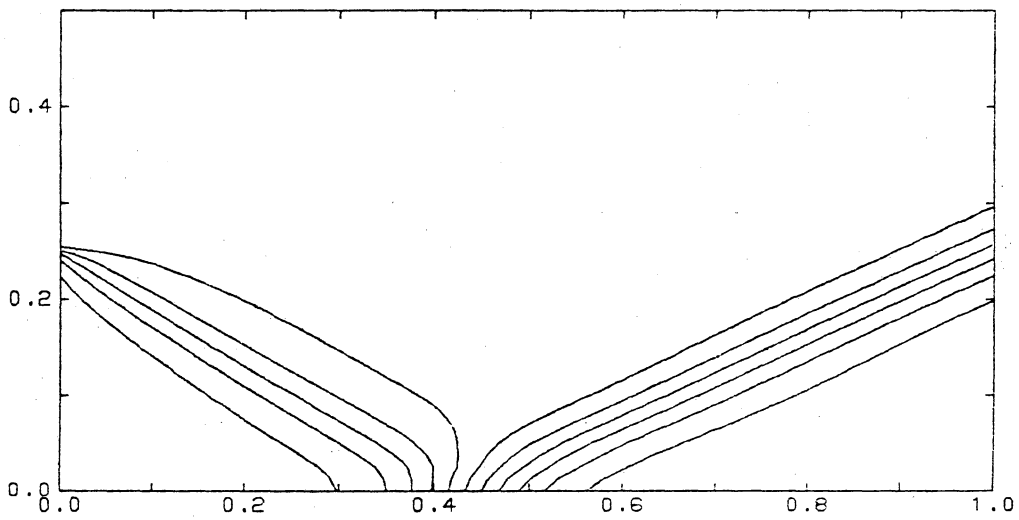


Figure 8. Density Contours - Shock Reflection of a Flat Plate: $M_\infty = 2.9$, 1st-order Upwind with Split p_b, 81 by 41 Grid.

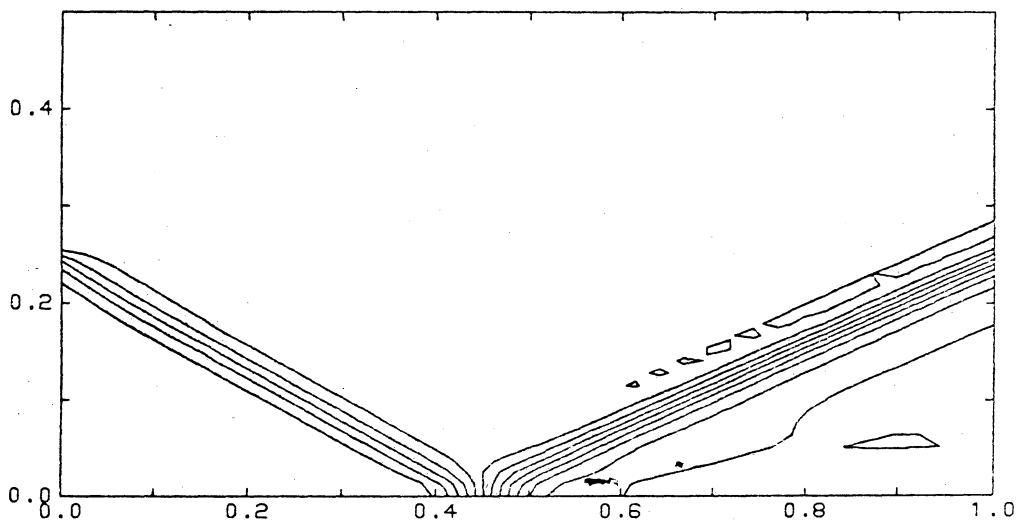


Figure 9. Density Contours - Shock Reflection of a Flat Plate: M_∞ = 2.9, 2nd-order Upwind with Split p_h, 41 by 41 Grid.

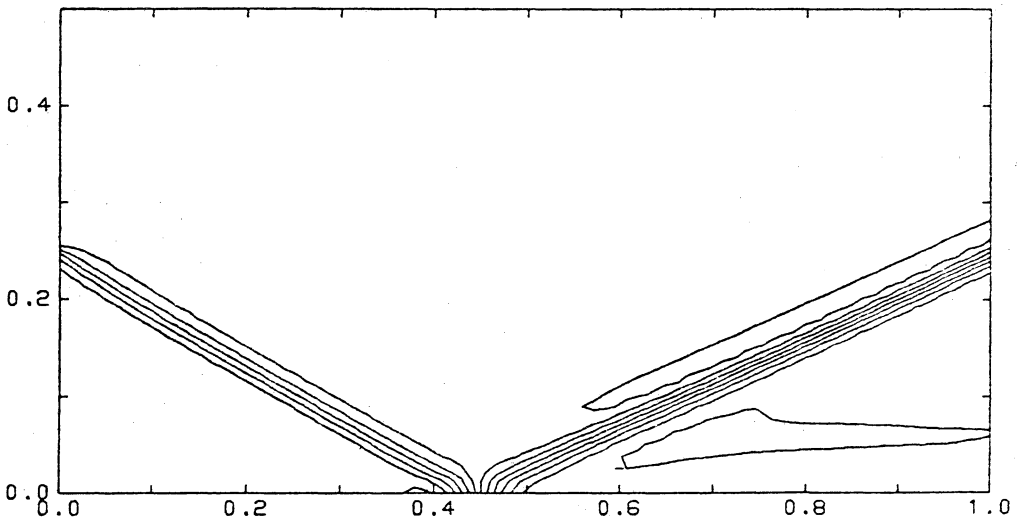
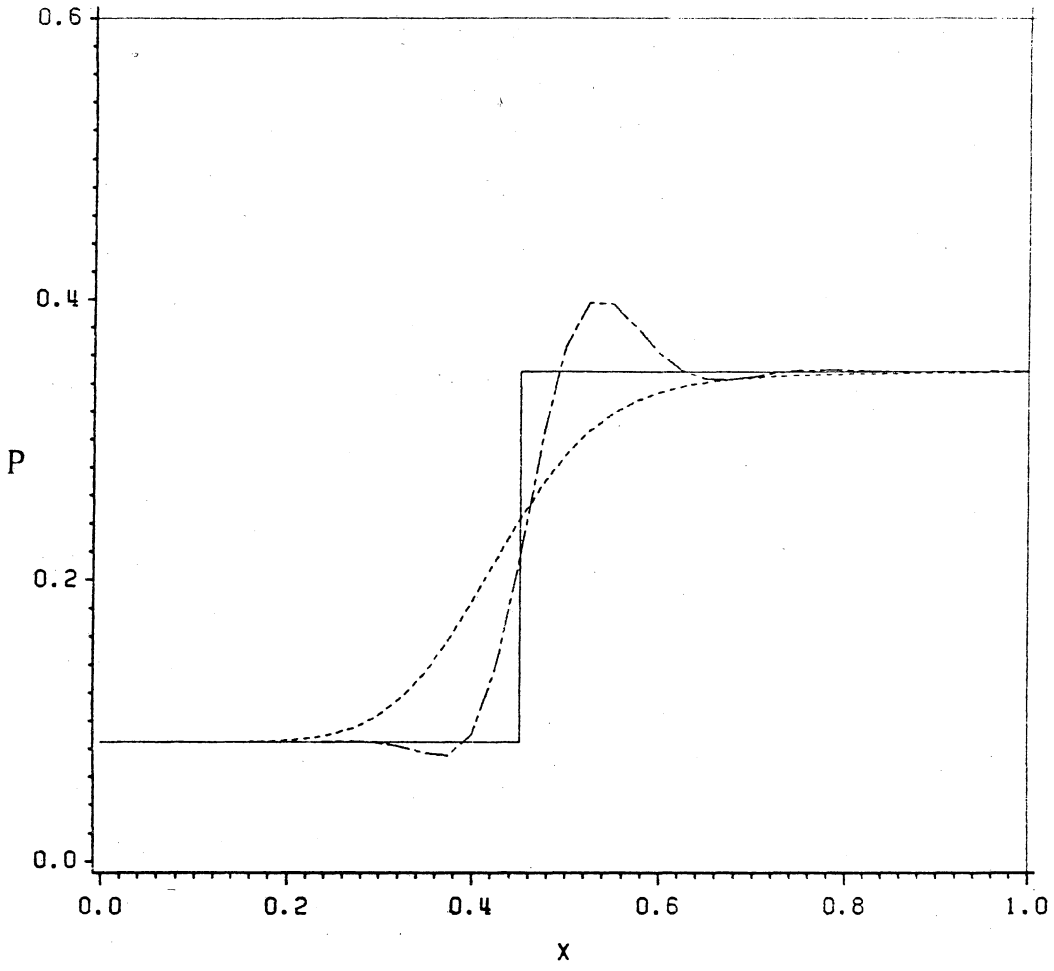


Figure 10. Density Contours - Shock Reflection of a Flat Plate: $M_\infty = 2.9$, 2nd-order Upwind with Split p_b, 81 by 41 Grid.



exact solution

first order

second order

Figure 11. Surface Pressure Distribution on a Flat Plate: $M_\infty = 2.9$, Unsplit p_b, 41 by 41 Grid.

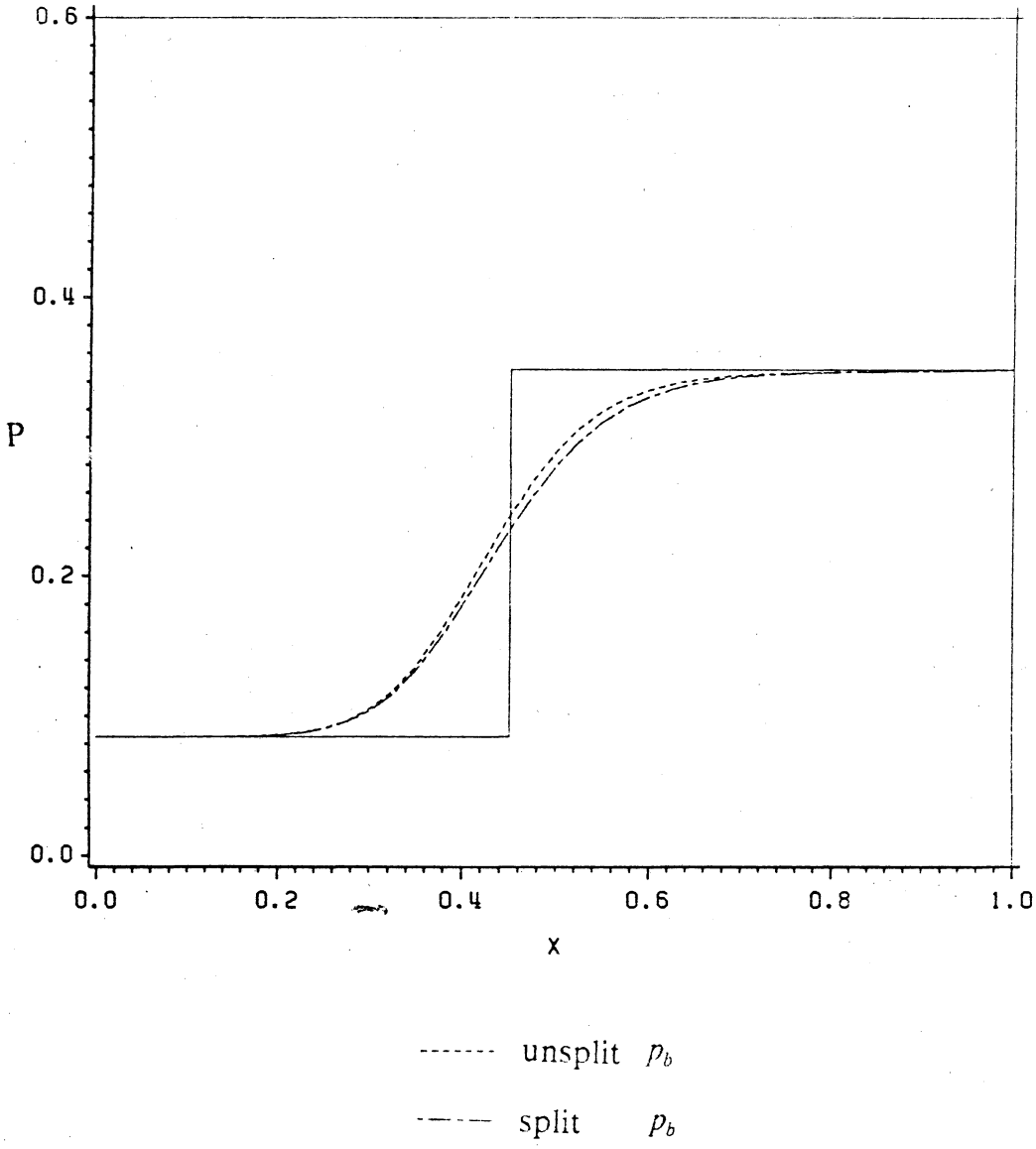


Figure 12. Surface Pressure Distribution on a Flat Plate: $M_\infty = 2.9$, 1st-order Upwind, 41 by 41 Grid.

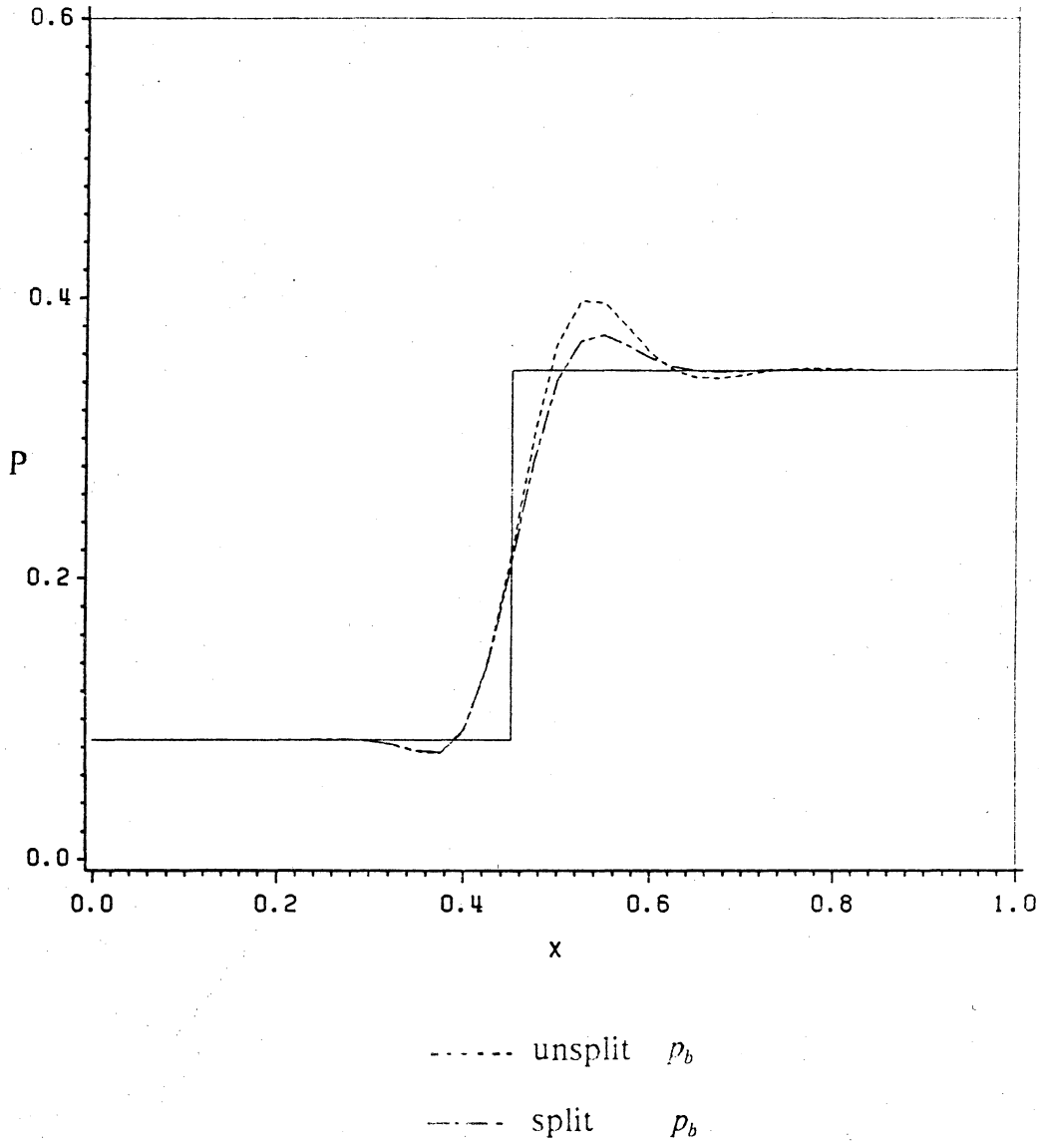


Figure 13. Surface Pressure Distribution on a Flat Plate: $M_\infty = 2.9$, 2nd-order Upwind, 41 by 41 Grid.

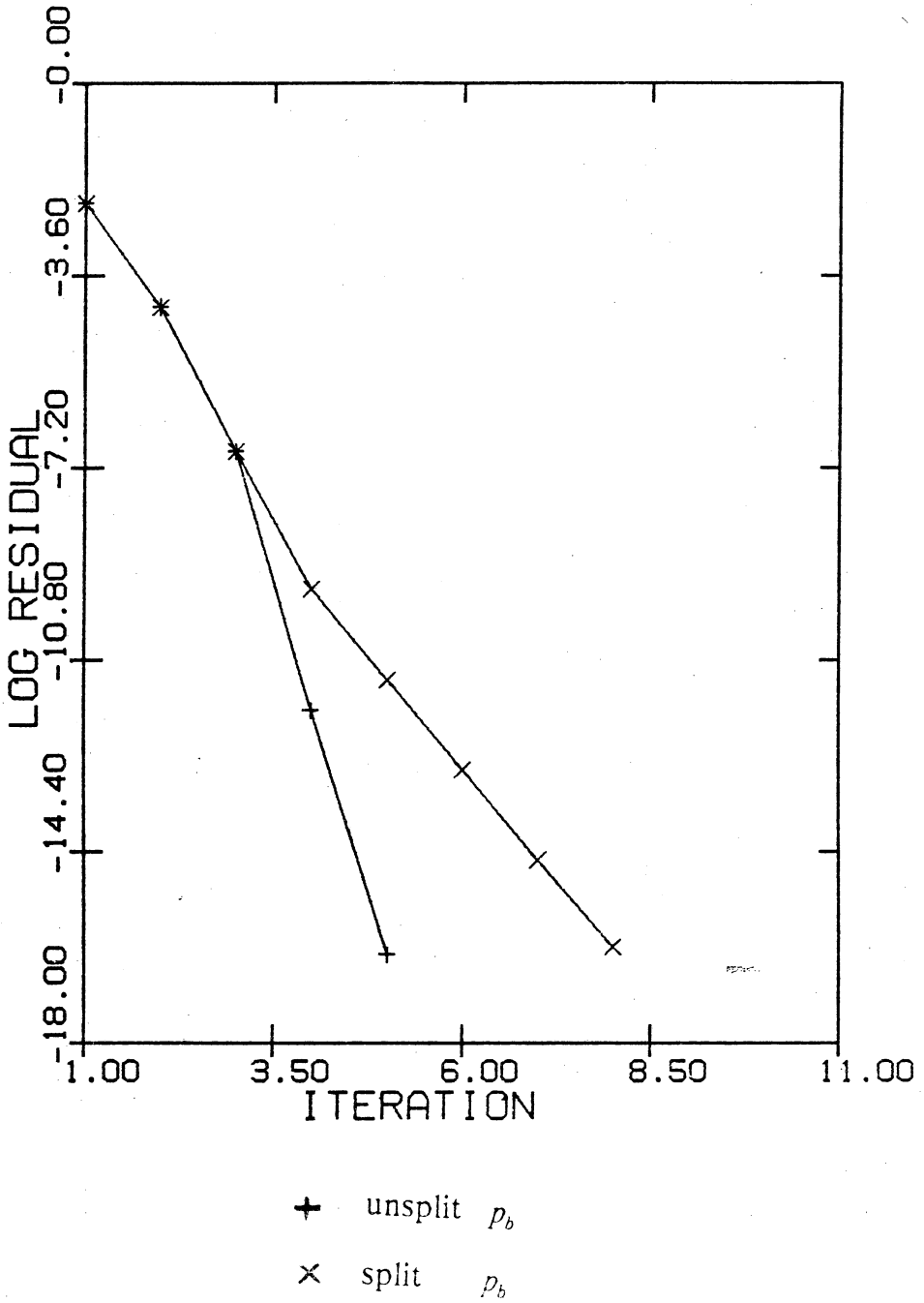
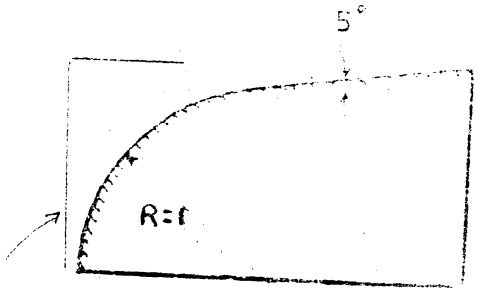


Figure 14. Typical Convergence History of Local Iteration.

$\rightarrow \quad M_\infty = 6.57$



region where detailed grid depicted

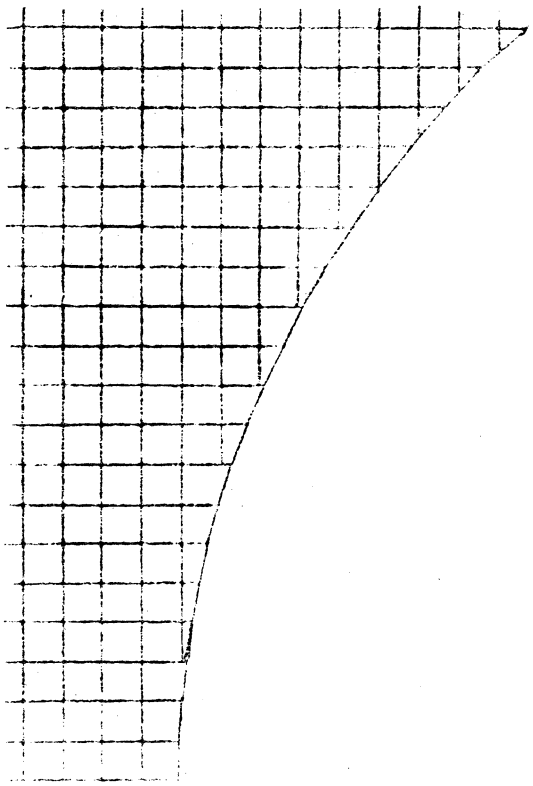


Figure 15. Configuration and grid - Supersonic Flow over a Blunt Body

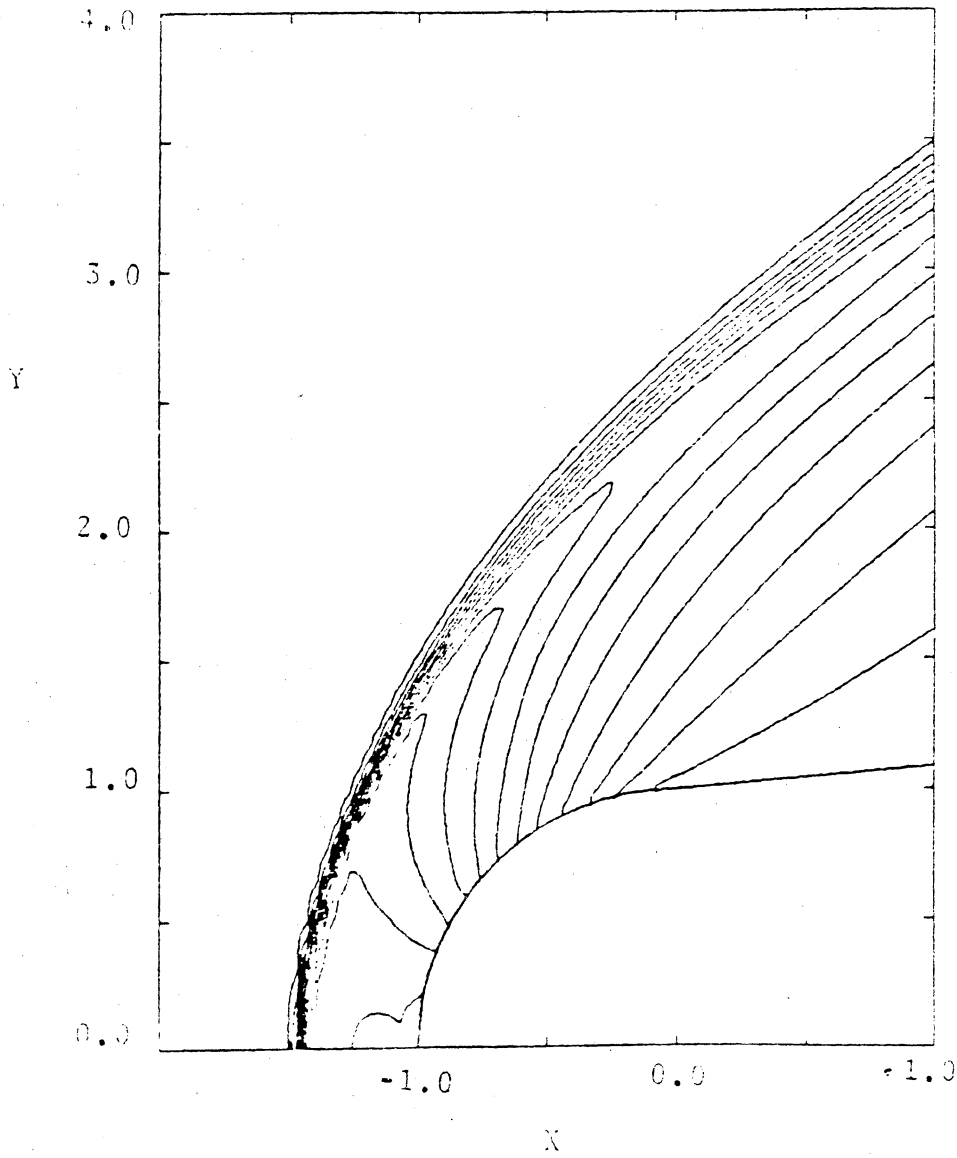


Figure 16. Density Contours - Supersonic Flow over a Blunt Body: $M_\infty = 6.57$, 1st-order Upwind.

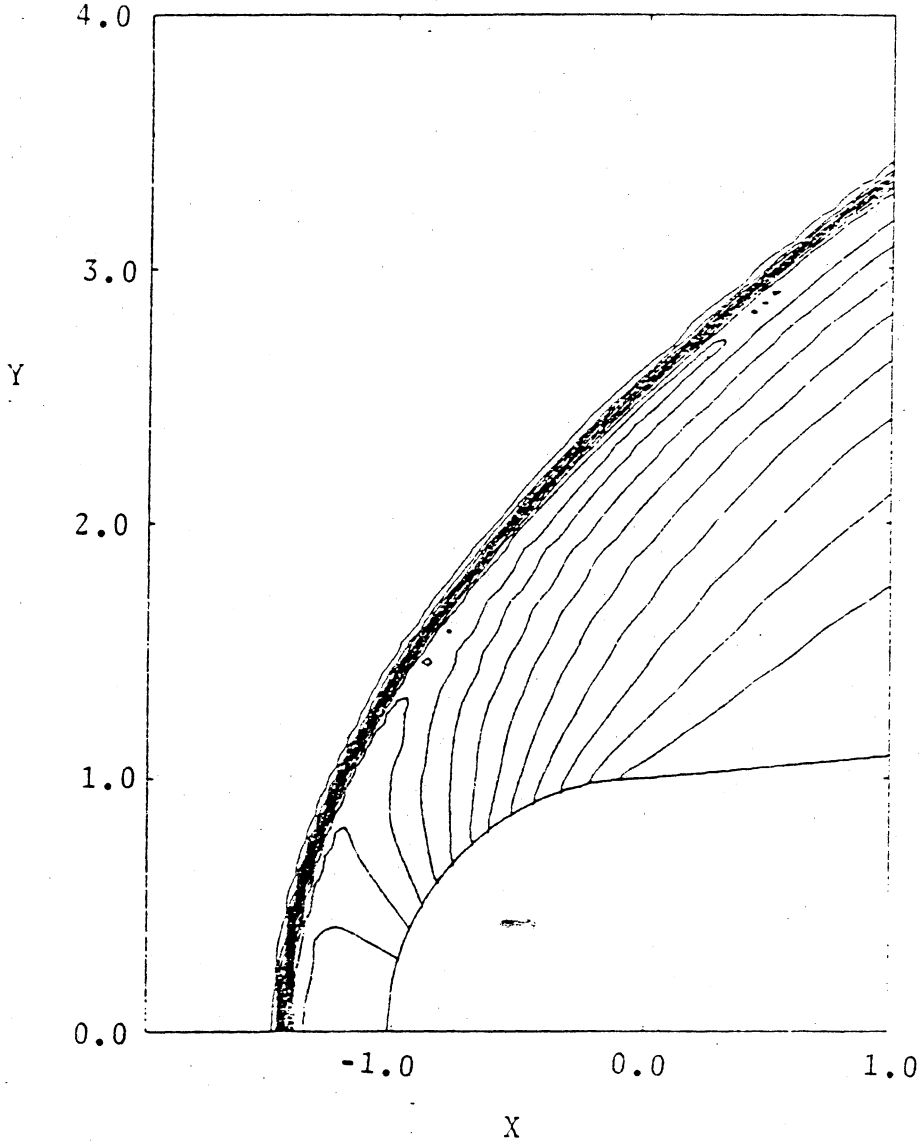


Figure 17. Density Contours - Supersonic Flow over a Blunt Body: M_∞ = 6.57, 2nd-order Upwind.

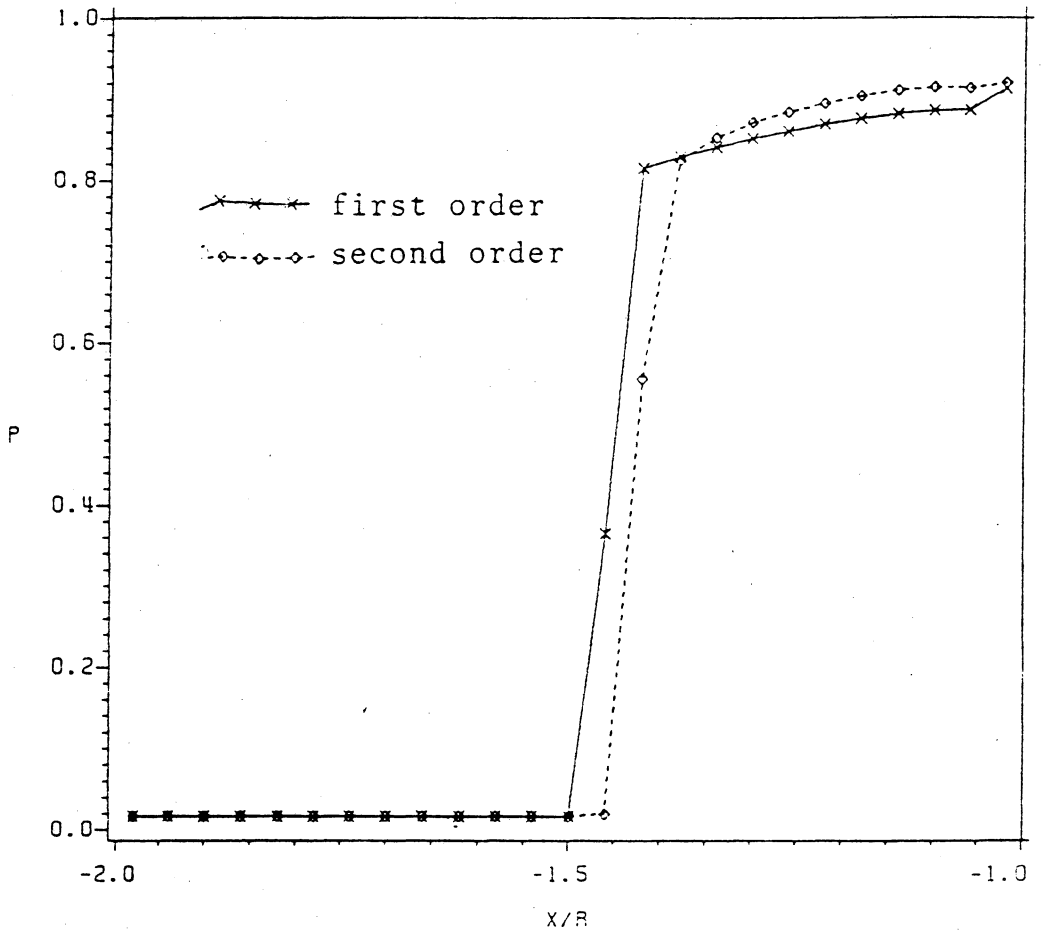


Figure 18. Pressure at the Symmetry Plane - Supersonic Flow over a Blunt Body: $M_\infty = 6.57$.

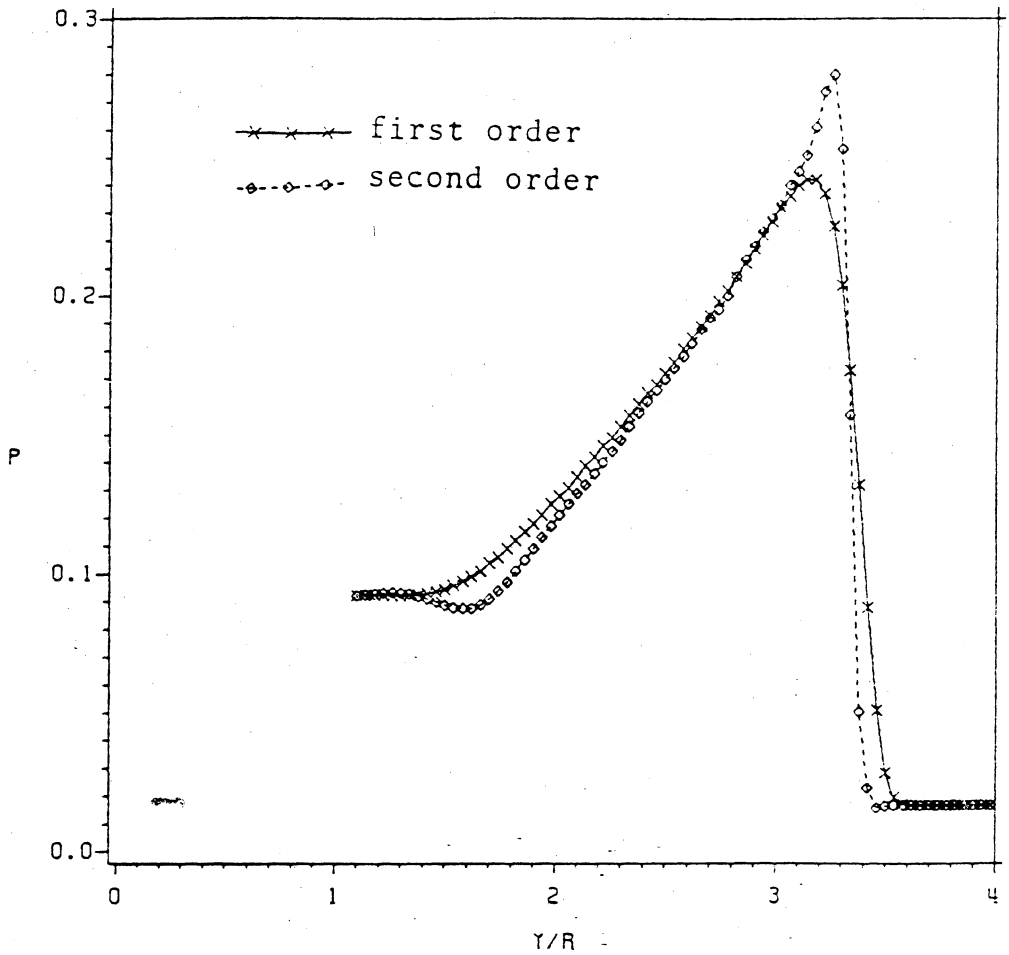


Figure 19. Pressure at the Outflow Plane - Supersonic Flow over a Blunt Body: $M_\infty = 6.57$.

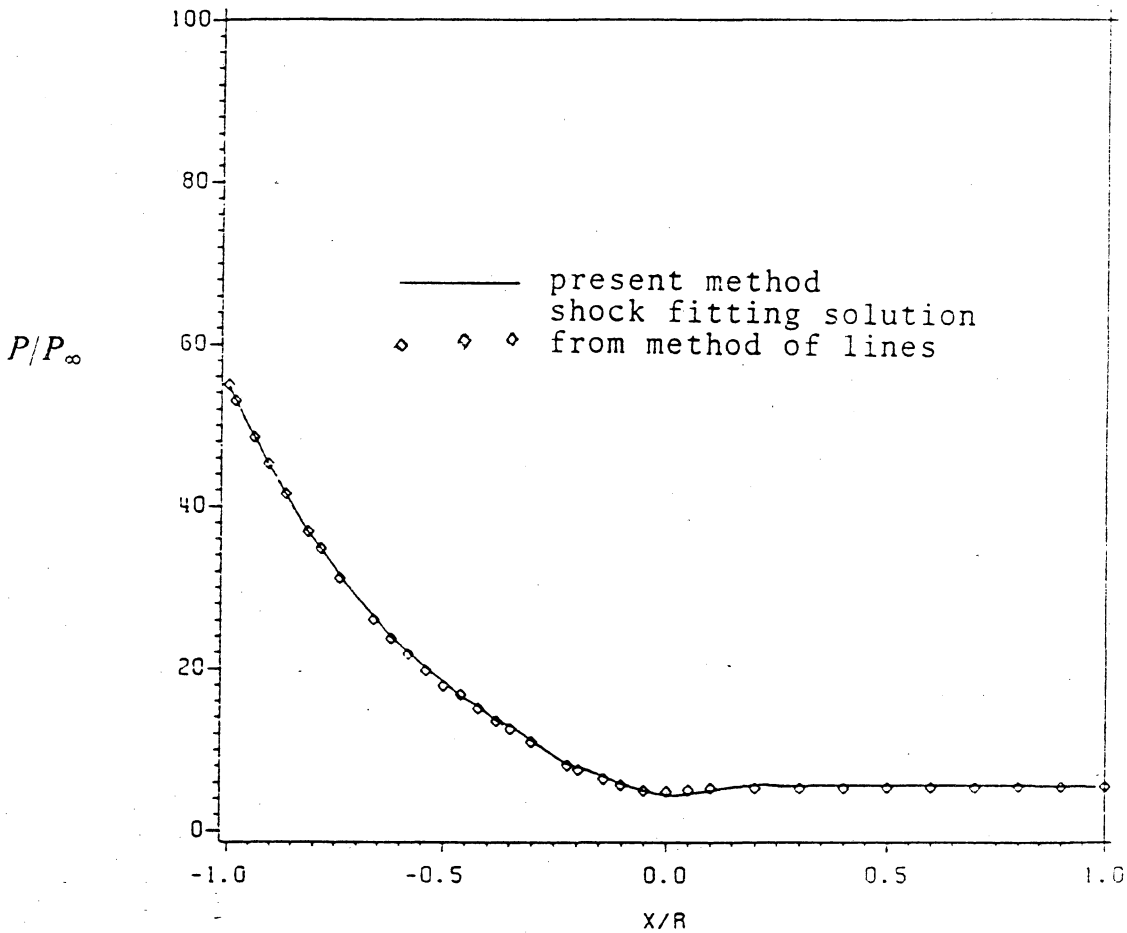
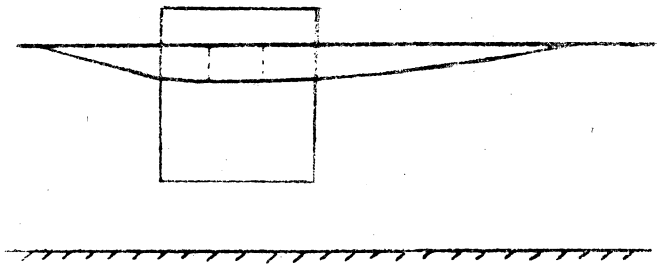
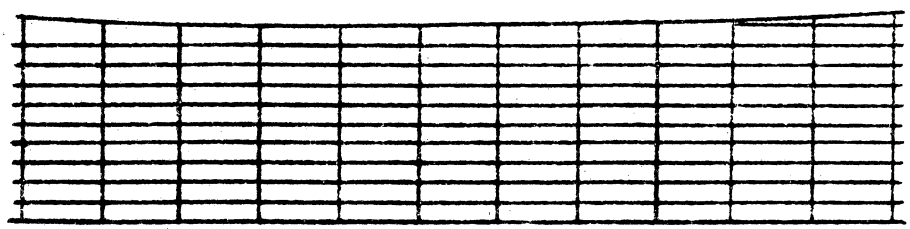


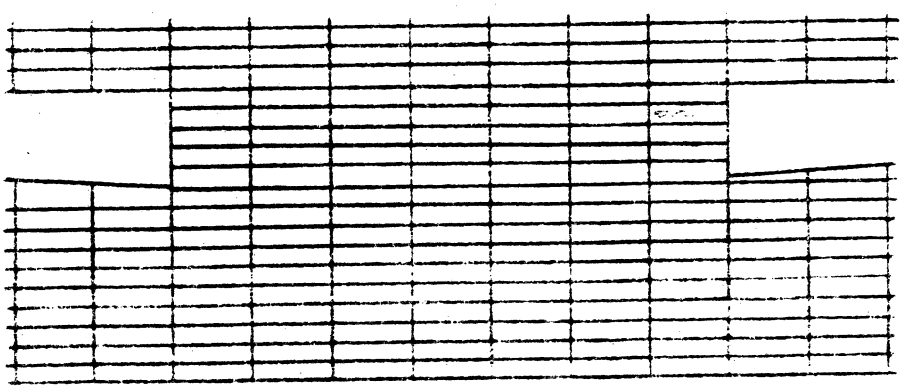
Figure 20. Pressure Distribution on a Blunt Body: $M_\infty = 6.57$.



a) region where detailed grid depicted

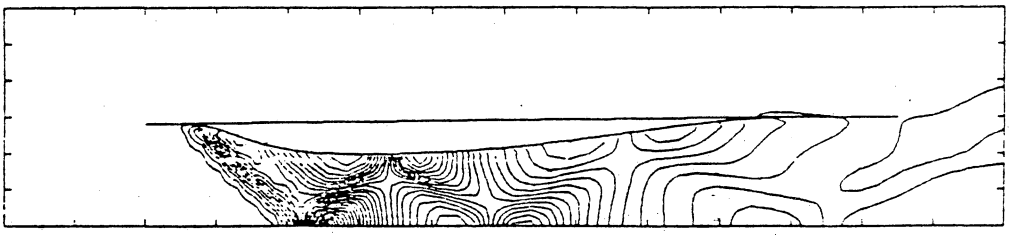


b) grid near throat (without door)

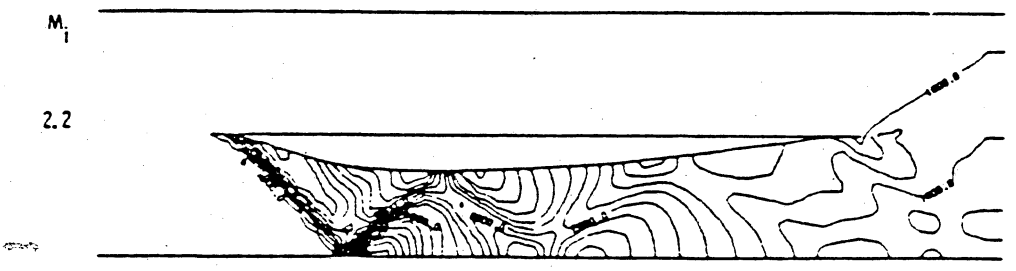


c) grid near throat (with door)

Figure 21. Configuration and grid - Scramjet Inlet Flows

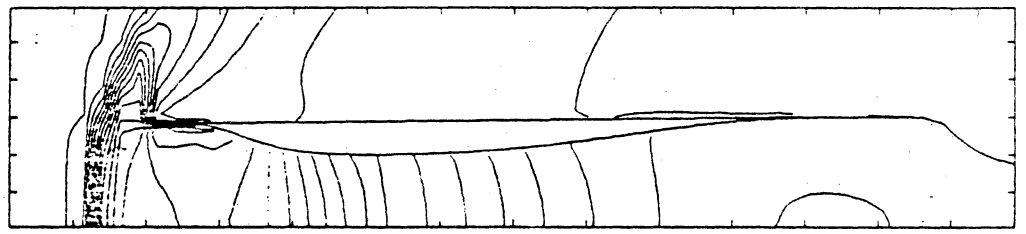


a) present inviscid calculation

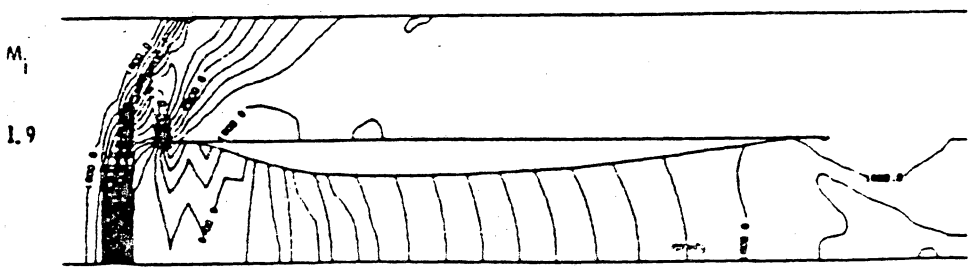


b) Navier-Stokes calculation (Ref. 39)

Figure 22. Pressure Contours - Scramjet Inlet Flows (Mach Reflection): $M_i = 2.2$.

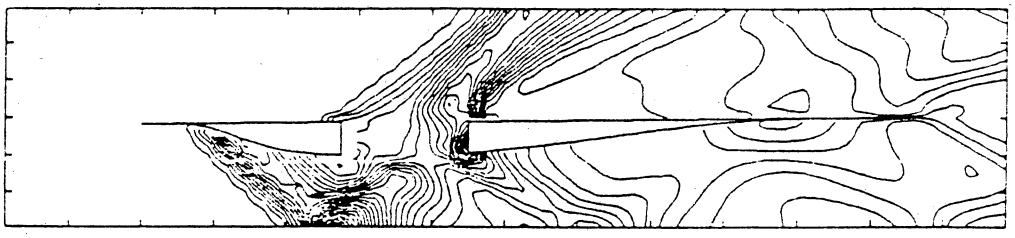


a) present inviscid calculation

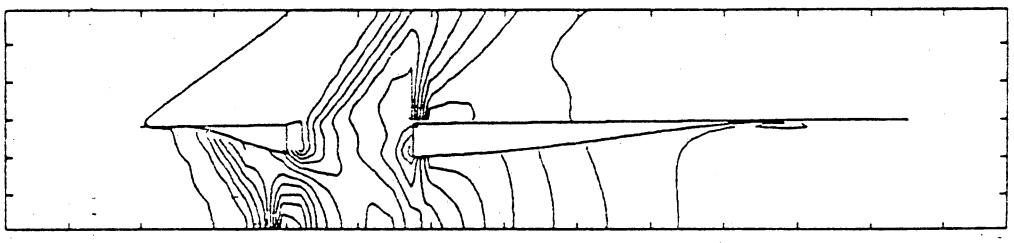


b) Navier-Stokes calculation (Ref. 39)

Figure 23. Pressure Contours - Scramjet Inlet Flows (Aerodynamic Choking): $M_i = 2.2$.



a) $M_i = 2.2$



b) $M_i = 1.9$ (first-order)

Figure 24. Pressure Contours - Scramjet Inlet Flows with Opening on the Upper Wall.

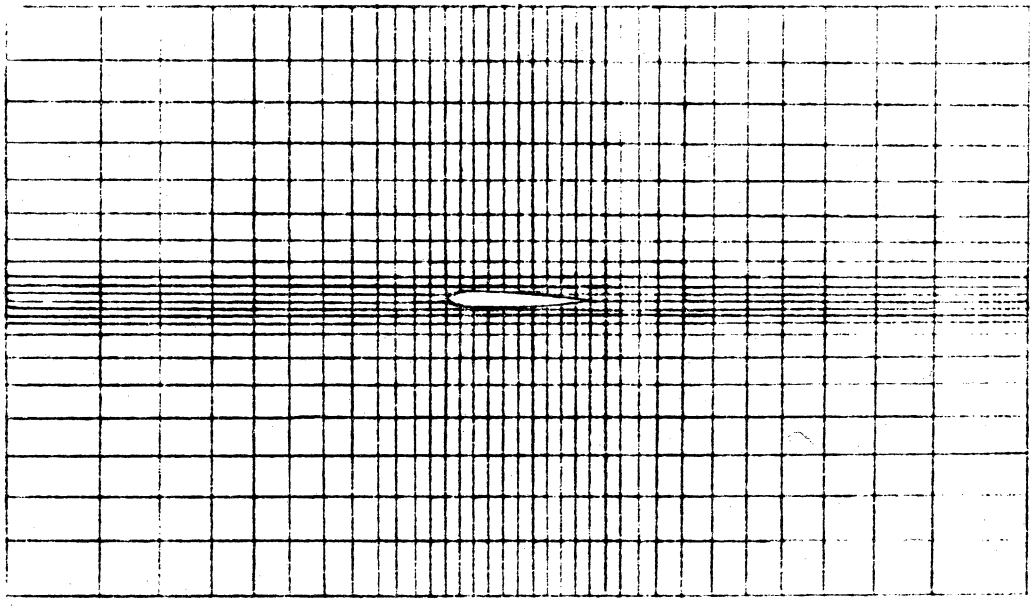
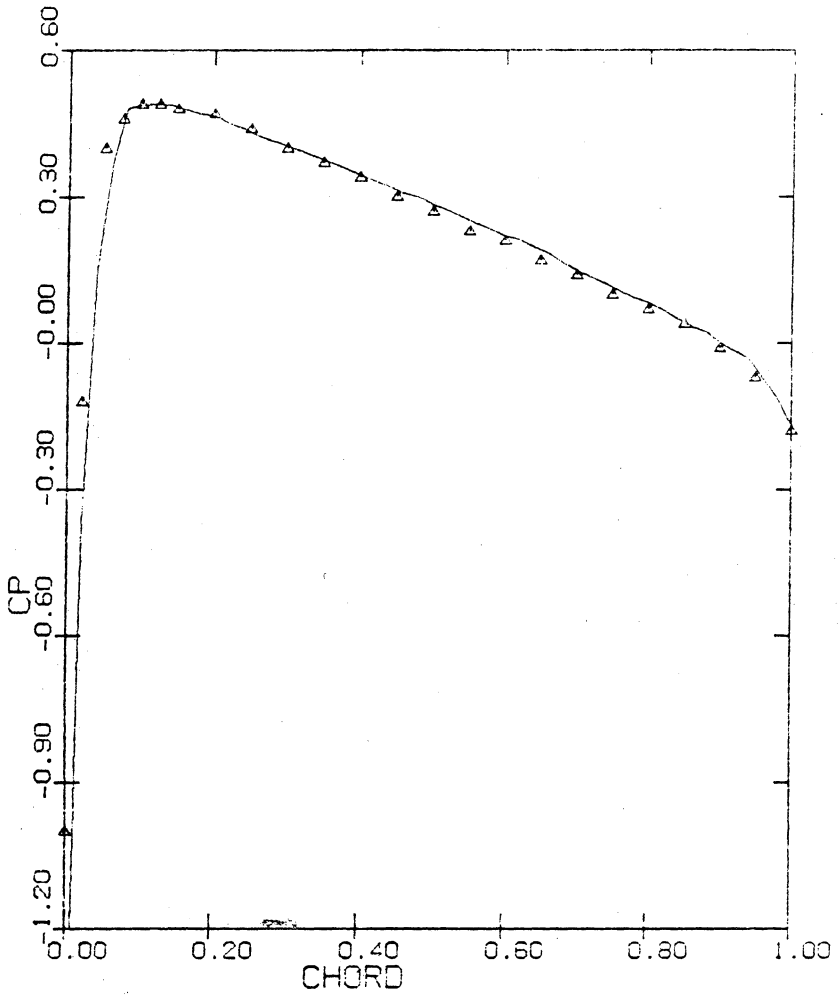


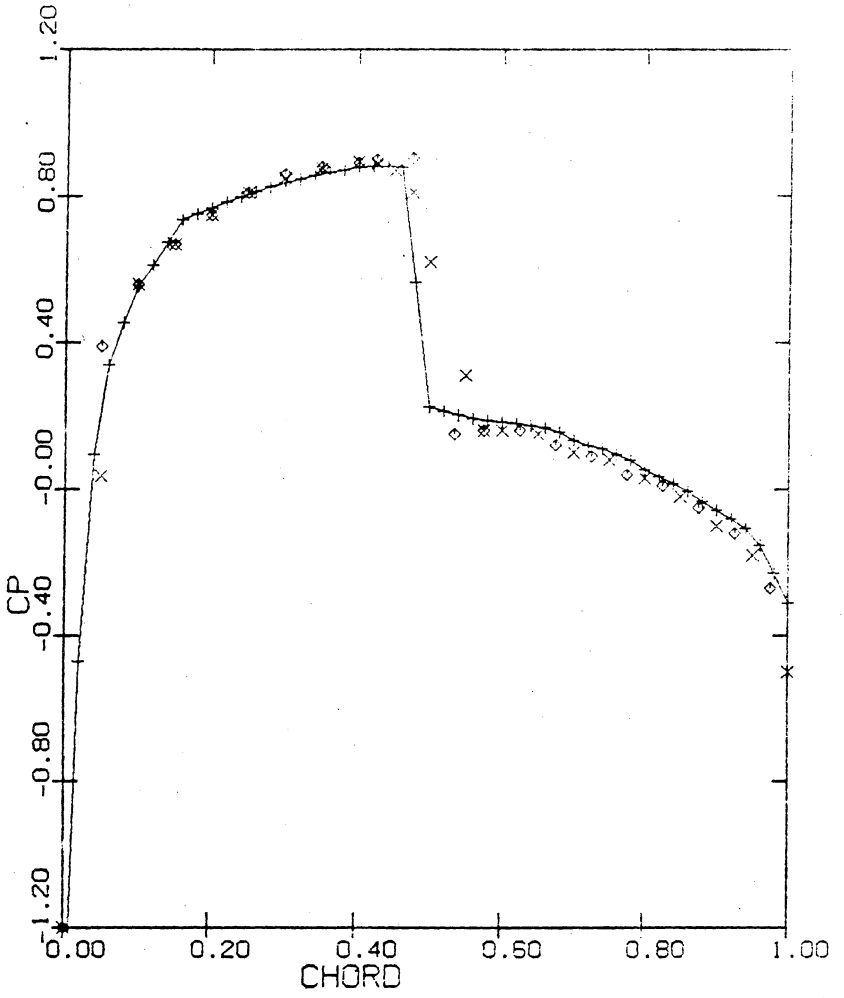
Figure 25. Sketch of Grid - NACA 0012 Airfoil



—— present method

△ △ Wedan and South (Ref. 7)

Figure 26. C_p Distributions over a NACA 0012 Airfoil: $M_\infty = 0.5$, $\alpha = 0^\circ$.



× × Clark et al. (Ref. 10)

◊ ◊ FLO52S Jameson et al. (Ref. 40)

—+ +— present method

Figure 27. C_p Distributions over a NACA 0012 Airfoil: $M_\infty = 0.8$, $\alpha = 0^\circ$.

The vita has been removed from the scanned document