Viscous Solutions for the Navier Stokes Equations

Using an Upwind Finite Volume Technique

by

Curtis R. Mitchell

Thesis submitted to the Faculty of the

Virginia Polytechnic Institute and State University

in partial fulfillment of the requirements for the degree of

Master of Science

in

Mechanical Engineering

APPROVED:

Dr. W. F. Ng, Chairman

Dr. H. L. Moses

Dr. W. F. Brown

May 1988

Blacksburg, Virginia

Viscous Solutions for the Navier Stokes Equations

Using an Upwind Finite Volume Technique

by

Curtis R. Mitchell

Dr. W. F. Ng, Chairman

Mechanical Engineering

(ABSTRACT)

The process of enhancing an upwind finite volume, two-dimensional, thin layer Navier Stokes solver to achieve complete Navier Stokes solutions is described. The shear stress and heat flux contributions are identified and transformed to a generalized coordinate system. The metrics which result from the transformation have a geometrical interpretation in the finite volume formulation and are presented as supporting material. The additional terms which are neglected in the thin-layer approximations, are evaluated and discretized consistently with the finite volume method. Implicit linearizations are applied to the second derivatives tangent to the body surface; however, the cross derivatives are not linearized and are treated conservatively. Validation of the Navier Stokes solver is acquired by comparison to existing computational solutions for a double throat nozzle. Additional viscous solutions for the thin layer and the complete forms of the NS equations are provided for a flat plate shock boundary layer interaction.

Acknowledgements

I would like to express my sincerest thanks to my major professor Dr. Wing Ng for his intuitive insights and helpful advice during the course of my research. His keen ability to focus on the important issues and to provide the proper direction allowed for a successful and enjoyable working relationship.

All of this would not have been possible without the love and patience from my wife, Her continued moral support served as my inspiration and motivation. She participated in my enthusiasm and helped me maintain my drive throughout the duration of my studies.

Special thanks to Dr. Bob Walters for many helpful recommendations and his suggestion to run a double throat nozzle problem. I also feel grateful to for his valuable instruction and shared knowledge during my graduate studies. I would also like to acknowledge for his assistance in preparing this thesis.

Final thanks to my mother and father who have shared in my academic dreams and have truly been a source of encouragement.

Table of Contents

List of Illustrations

List of Tables

Nomenclature

a	speed of sound
CPU	central processing unit
e	total internal energy per unit volume
F,G	pressure and convection vector in cartesian coordinates
j,k	spatial indices
L	reference length
LU	lower-upper
L_2	norm of residual
NS	Navier-Stokes
PNS	Parabolized Navier-Stokes
Pr	Prandtl number
Q	vector of conserved variables
r	component of vector R
r'	component of vector R'
R,S	shear stress and heat flux vector in cartesian coordinates
R',S'	shear stress and heat flux vector in generalized coordinates
Re	freestream Reynolds number (based on reference length)
s	component of vector S
s'	component of vector S'
t	time
t'	component of vector T'
T'	reduced stress and heat flux vector in generalized coordinates
TLNS	Thin Layer Navier-Stokes

u	x component of velocity
v	y component of velocity
x,y	x and y coordinate directions, respectively

GREEK LETTERS

Δ	finite change
η	normal direction in generalized coordinate system
γ	ratio of specific heats
λ	bulk viscosity
μ	coefficient of viscosity, kg/ms
∂	partial derivative
ρ	density
τ	shear stress N/ m^2
ξ	streamwise direction in generalized coordinate system

SUBSCRIPTS

∞	refers to freestream conditions
x	refers to first partial derivative in the x coordinate direction
y	refers to first partial derivative in the y coordinate direction
xx	refers to second partial derivative in the x coordinate direction
yy	refers to second partial derivative in the y coordinate direction
xy	refers to second partial derivative derivative w.r.t. x and y
yx	refers to second partial derivative derivative w.r.t. y and x
1	refers to vector element 1
2	refers to vector element 2
3	refers to vector element 3
4	refers to vector element 4
ξ	refers to 1st partial derivative in the ξ direction
η	refers to 1st partial derivative in the η direction
0	refers to stagnation conditions

1.0 Introduction

A primary goal of fluid dynamicists who are dedicated to analyzing the aerodynamic performance of both external and internal flows, is to numerically solve the differential equations which govern the flow in an efficient and accurate manner. Accomplishing the goal, however, requires continued support and efforts in several areas which contribute to the overall success of computational schemes. Specifically, a great amount of time and effort is spent researching algorithm development, grid generation, zonal methods, and turbulence closure models to ensure accurate analysis for a wide variety of flow problems ranging from subsonic, laminar, inviscid flows to unsteady, hypersonic, turbulent flows including viscous effects. The latter is extremely popular and demanding with the introduction of the National Aero-Space Plane Program and the Trans-Atmospheric Vehicle [1].

The study of unsteady viscous flow problems is a fundamental aspect of classical aerodynamics which does not easily lend itself to computational analysis due to inherent turbulence, massive flow separation, and strong viscous/inviscid interaction [2]. Therefore, there exists an increasing need for accurate numerical simulation of these complex flows by solving the full Navier Stokes (NS) equations to the steady state. However, even with the current rapid advances in the speed and available storage of computers, there is a need for improved algorithms to enable viscous flow

computations to be used routinely in engineering practice [3]. Because of their importance, the viscous terms, identified as the shear stress and heat flux, in the NS equations are the center of concentration within this study. Acknowledging and incorporating these terms into a state of the art solver will be of great value in the understanding of the computational techniques that are dominating the field today and will provide a strong foundation for further research in Computational Fluid Dynamics (CFD).

Until the mid 1970's explicit algorithms, such as MacCormack's [4], were used to numerically solve the NS equations. However, common to explicit methods, the restriction on the maximum time step limited the practicability of the explicit methods when analyzing compressible viscous flows. The maximum admissible time step is established by the CFL (Courant, Friedrichs and Lewy) condition. This concern led to new implicit numerical schemes of Briley and McDonald [5] and Beam and Warming [6] which reduced the CFL limitation but did require linearizations of terms as a result of the implicit formulation. In the early 1980's, MacCormack [7] introduced an explicit-implicit predictor-corrector method which gained popularity for solving an assortment of compressible viscous flows. A comparison of the hybrid MacCormack and the implicit Beam-Warming algorithm can be found in Ref. [8].

The recent development of upwind difference methods for the Euler equations [9], [10], and the thin-layer and full NS equations [3], [11], has provided new opportunities for the development of efficient numerical schemes [12]. Upwind methods identify the correct direction of propagation of information and then upwind difference in that direction. There are several advantages to using this method. First of all, these methods are naturally dissipative and capture strong shocks without oscillation or the addition of explicit artificial viscosity. In addition, the hyperbolic nature of the Euler (inviscid) equations is simulated with upwind differencing.

Furthermore, upwind differencing enables the implementation of alternate solution strategies such as relaxation since the implicit coefficient matrix is more nearly diagonally dominant than its central difference counterpart [12]. Upwind discretization is implemented in the present code used in this thesis. The two techniques that incorporate the upwind concept are Flux Difference Splitting (FDS) and Flux Vector Splitting (FVS). Both of these methods identify the direction of the propagating information by evaluating the signs of the characteristic speeds. The contributions from Steger-Warming and Van Leer dealing with FVS can be found in Refs [13] and [14], respectively. FDS was developed by Roe [15].

Flux splitting and upwinding are two definite major steps in the direction of accurately predicting the physics of high speed flows ; however, the question of correctly capturing shocks and other discontinuities becomes extremely important when analyzing high speed flowfields. Therefore, the objective is to solve the governing equations which can be cast in integral form rather than differential form. This is accomplished through the development of finite volume techniques. The advantage of this approach is that shock waves and contact discontinuities evolve as part of the solution process and, in the limit of a vanishingly small cell size, have the correct strength and location, thus yielding a method applicable to a wide class of problems [16]. Extending the equations to representations of integral conservation laws in a generalized coordinate system requires an evaluation process of the discretization techniques and handling of the metrics terms that result from the transformation. A general overview of the finite volume formulation will, therefore, be presented as supplemental material to gain understanding in this technique. Additionally, the coordinate transformation process will be demonstrated only on the viscous terms that appear in the NS equations, however, extension of the inviscid terms to generalized coordinates is accomplished similarly.

The analysis and evaluation processes described above will be presented ultimately in the form of code modification and validation. Specifically, the major effort involved in this work concentrates on incorporating and validating the capability of Full Navier Stokes solutions using an existing thin-layer Navier Stokes solver as the primary tool. To accomplish this goal, it is necessary to acknowledge the viscous terms which are neglected within the thin-layer approximations and to include them in a manner consistent with the finite volume method. Implicit linearizations are enforced only to the viscous terms that appear as double derivatives ; however, all additional terms must be added in the explicit side of the equations. Implicit treatment of the cross derivatives is usually not enforced. The details of the explicit contributions are thoroughly detailed within this thesis. The remaining implicit terms are duplicated from the thin layer viscous linearizations already provided in the solver and are excluded. Validation of the Full Navier Stokes solver is a secondary effort in this work and therefore limited case studies will be presented.

There is one case study contained in this thesis. The study examines a two-dimensional shock-wave boundary-layer interaction that experiences mild viscous/inviscid interactions. The initially laminar boundary layer separates due to an oblique shock impingement, thus complicating the computation. Success of the study will be based on a comparison between the computational results and the experimental data.

Because this research required a fundamental understanding of the computational techniques as well as personal assistance in the development and benchmarking of the code used in this study, the contents of this thesis will include detailed discussions of the finite volume method with strong emphasis on the viscous development. After summarizing the reduced forms of the Navier Stokes equations used in the flow analyses, an introduction to the newly developed code and its

capabilities will follow. Next will be a summary of the governing equations and an overview of the finite volume formulation including remarks concerning the metrics. The following section will reveal the results of a coordinate transformation and the spatial discretization of the viscous terms in the NS equations. Results of the shock-wave boundary-layer interactions study will then be presented. Conclusions and recommendations will conclude this thesis.

2.0 Forms of the Navier Stokes Equations

The equations of motion for unsteady compressible flow consist of mass, momentum and energy. In their entirety, these equations constitute the well known Navier Stokes equations. However, depending on the complexity of the flows being analyzed, the complete form may be reduced without sacrificing an accurate solution. This section will briefly discuss the implications of solving the following reduced forms of the equations.

1. Parabolized Navier Stokes (PNS)
2. Thin Layer Navier Stokes (TLNS)

2.1 Parabolized Navier Stokes

The Parabolized Navier Stokes equations are solved using an efficient space marching strategy that accurately predicts steady supersonic attached viscous flows. Compared to the Navier Stokes equations which are solved with time iterating techniques, the spatial marching method requires significantly less computational time. There are, however, restrictions on the PNS equations which may cause their solution approach to be unsatisfactory for some flowfields. First of all, the flowfield

in the marching direction must be primarily supersonic, with only 'thin' subsonic regions in the boundary layer. In general, the flowfield must maintain a positive velocity in the marching direction or the algorithm may fail. The major obstacle that may cause trouble is the case of separated flow in this direction. A second major problem with a space marching technique is the handling of the streamwise pressure gradient in the momentum equation. If too much upstream influence is allowed to enter the calculation the problem is not well posed and the solution will feel this influence and grow in error without bound. The streamwise pressure gradient in the subsonic boundary layer must therefore have special treatment to ensure that only a portion of the upstream influence is retained.

The PNS equations, like the thin layer Navier Stokes equations, fully retain the normal pressure gradient term and neglect diffusion terms in the streamwise direction. Due to the restriction on the upstream influence from the downstream pressure gradient, the streamwise pressure gradient term in the streamwise momentum equation can be handled in several ways. The Vigneron technique [17] is utilized in this code.

2.2 *Thin-Layer and Complete Navier Stokes*

The Thin Layer Navier Stokes (TLNS) equations are created as a subset of the full NS equations by neglecting diffusion terms that have derivatives parallel to the body surface. The unsteady boundary layer equations also exhibit this feature; however, also embodied is the assumption of zero normal pressure gradients throughout the boundary layer. For viscous-inviscid regions to be solved simultaneously, the normal

pressure gradient becomes important and is, therefore, fully retained in the three momentum equations (for 3-D flows) within the TLNS and NS equations.

The important advantage of solving the normal momentum equation is the ability to solve separated and reversed flow in a very straight forward manner. Additionally, all flow regimes that exhibit high normal pressure gradients can be computed.

The thin layer equations are hyperbolic partial differential equations in time. The formulation of the thin layer approximations is apparent with an understanding of the physical interpretations of typical high Reynolds number flows. Resolving the normal gradients in the turbulent boundary layers requires a considerable amount of storage and computing time due to the necessity of highly stretched grids. The cost of using such grids is the inability to resolve the streamwise diffusion gradients adequately. Therefore, if it is known that terms are fairly small, then they can be neglected in the computation without sacrificing accuracy in the solution. In contrast, low Reynolds number flows with massive separation will require resolving the streamwise viscous gradients provided in the complete NS equations. The disadvantage, is the need for more dense grids in the streamwise direction which will require additional computer storage and increased computing cost.

The capability of acquiring numerical solutions for the PNS, TLNS and complete NS equations is extremely valuable when analyzing various internal and external flows. The computing code described in the following section has this ability.

3.0 Virginia Tech Navier Stokes Solver (VERSAFLOW)

A cell-centered, finite volume, upwind, two-dimensional, Navier Stokes code has been developed to study the aerodynamic performance of internal and external flows. Incorporated within this code are some of the newest technologies which greatly improve the overall usability and efficiency in obtaining steady state solutions to the Navier Stokes equations. One impressive feature is the ability to utilize an existing PNS, TLNS or NS solution as an initial condition for further computation, provided the mesh distribution remains unchanged.

It is acknowledged that the original algorithm is based on that of Thomas and Walters [3] (code name - ANSERS). However, substantial and important contributions have been added to the code to justify a new code name. The new code is referred to as VERSAFLOW.

3.1 Capabilities of VERSAFLOW

The following list of options is available in VERSAFLOW at the user's option.

- Steady state solutions for the following equations :
  - Euler
  - Parabolized Navier Stokes
  - Thin Layer Navier Stokes
  - Complete Navier Stokes
- Implicit Algorithms :
  - Vertical and Horizontal Line Gauss Seidel
  - Approximate Factorization (time accurate)
- Explicit Algorithms :
  - Multi-Stage Runge Kutta (time accurate)
- Cell Face Interpolation :
  - Upwind difference - first and second order
  - Central difference - second order
  - Upwind biased - third order
- Re-use LU Decomposition/Frozen Jacobian
- Van Leer's Flux Vector Splitting
- Roe's Flux Difference Splitting
- Eddy Viscosity Models :
  - Baldwin Lomax
  - McDonald Camarata (with Bushnell Beckwith correction)
- Van Albeda's and Min-mod limiters
- Higher order spatial discretization of implicit terms within the algorithm

The following options are available for space marching :

- Spatial marching with pressure update (multisweep PNS)
- Vigneron technique for handling streamwise pressure gradient in subsonic region
- Flare approximation for separated flow
- Catastrophic limiting

The specifics detailing the theory and development of the above techniques are readily available in the literature and thus are omitted. However, the governing equations and their implementation will be provided in the following sections.

3.2 Governing Equations

The non-dimensional form of the Navier Stokes equations cast in conservation form in two dimensional cartesian coordinates, is written as

$$\frac{\partial Q}{\partial t} + \frac{\partial F}{\partial x} + \frac{\partial G}{\partial y} = \frac{1}{Re}\left[\frac{\partial R}{\partial x} + \frac{\partial S}{\partial y}\right] \qquad (3.2-1)$$

where the vector of conserved variables is defined as

Time dependent :

$$Q = \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ e \end{bmatrix}$$

The Pressure and convective terms are :

$$F = \begin{bmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ (e + p)u \end{bmatrix} \qquad G = \begin{bmatrix} \rho v \\ \rho uv \\ \rho v^2 + p \\ (e + p)v \end{bmatrix}$$

The Shear stress and heat flux terms are :

$$R = \begin{bmatrix} 0 \\ \tau_{xx} \\ \tau_{xy} \\ r_4 \end{bmatrix} \qquad S = \begin{bmatrix} 0 \\ \tau_{xy} \\ \tau_{yy} \\ s_4 \end{bmatrix}$$

$$\tau_{xx} = 2\mu \frac{\partial u}{\partial x} + \lambda\left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}\right),$$

$$\tau_{yy} = 2\mu \frac{\partial v}{\partial y} + \lambda\left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}\right),$$

$$\tau_{xy} = \mu\left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right),$$

$$r_4 = u\tau_{xx} + v\tau_{xy} + \frac{\mu}{Pr}\frac{1}{(\gamma - 1)}\frac{\partial a^2}{\partial x},$$

$$s_4 = u\tau_{xy} + v\tau_{yy} + \frac{\mu}{Pr}\frac{1}{(\gamma - 1)}\frac{\partial a^2}{\partial y}.$$

The above equations are non-dimensionalized by freestream values of density ρ_∞, viscosity μ_∞ , velocity u_∞, and reference length *L*. The pressure P is calculated using the ideal gas law

$$P = (\gamma - 1)[e - \rho(u^2 + v^2)/2]$$

The molecular viscosity, μ is evaluated using Sutherland's law

$$\mu = \frac{\mu}{\mu_\infty} = \left[\frac{T}{T_\infty}\right]^{\frac{3}{2}}\left[\frac{T_\infty + C}{T + C}\right]$$

where C is the Sutherland's constant equal to 110.4 K. The bulk modulus, λ, is evaluated using Stokes hypothesis ($\lambda = -2\mu/3$).

3.3 *Cell Centered Finite Volume Formulation*

Fundamental to the finite volume method is the integral interpretation of the governing equations. The primary advantage, as compared to the differential form, is the validity of the integral form when discontinuities arise. Consistency of the integral laws will be demonstrated following the derivation of a semi-discrete finite volume representation.

When a coordinate transformation is applied to equation (3.2-1), the governing equations in the (ξ,η) system become

$$\left[\frac{1}{J}\right]\frac{\partial Q}{\partial t} + \frac{\partial F'}{\partial \xi} + \frac{\partial G'}{\partial \eta} = \frac{1}{Re}\left[\frac{\partial R'}{\partial \xi} + \frac{\partial S'}{\partial \eta}\right] \qquad (3.3-1)$$

where

$$F', G' = \frac{K_x F + K_y G}{|\nabla K|} \cdot \frac{|\nabla K|}{J} \qquad K = (\xi, \eta)$$

$$R', S' = \frac{K_x R + K_y S}{|\nabla K|} \cdot \frac{|\nabla K|}{J} \qquad K = (\xi, \eta)$$

$$|\nabla K| = \sqrt{(K_x^2 + K_y^2)}$$

The metric terms that appear as a result of the coordinate transformation are evaluated geometrically, as will be shown later. Figure 1 corresponds to a representative finite volume cell which is used to obtain the following discretized form of equation (3.3-1).

$$\left[\frac{1}{J}\frac{\partial Q}{\partial t}\right]_{j,k} + \left[F' - \frac{1}{Re}R'\right]_{j+\frac{1}{2},k} - \left[F' - \frac{1}{Re}R'\right]_{j-\frac{1}{2},k} + \left[G' - \frac{1}{Re}S'\right]_{j,k+\frac{1}{2}} - \left[G' - \frac{1}{Re}S'\right]_{j,k-\frac{1}{2}} = 0 \qquad (3.3-2)$$

where the fluxes are evaluated at the cell faces and Q is evaluated at the cell center.

This semi-discrete representation can be directly compared to the integral conservation law defined by equation (3.3-3)

$$\frac{\partial}{\partial t}\iiint Q dV = -\iint \vec{F}.nds \qquad (3.3-3)$$

where

$$\vec{F} = \left(F' - \frac{1}{Re}R'\right)_j + \left(G' - \frac{1}{Re}S'\right)_k$$

and n is a unit vector normal to surface element ds.

When the integral on the left hand of equation (3.3-3) is represented as an average value over the entire cell, namely

$$V\frac{\partial Q_{av}}{\partial t} = \frac{\partial}{\partial t}\iiint Q dV$$

and applying the surface integral over the four surfaces of the cell, the integral conservation law yields

$$V\frac{\partial Q_{av}}{\partial t} + \left[(\vec{F}.n)\Delta S\right]_{j+\frac{1}{2}} + \left[(\vec{F}.n)\Delta S\right]_{j-\frac{1}{2}} + \left[(\vec{F}.n)\Delta S\right]_{k+\frac{1}{2}} + \left[(\vec{F}.n)\Delta S\right]_{k-\frac{1}{2}} = 0 \qquad (3.3-4)$$

Focussing attention on the second term, this can be rewritten as follows

$$\left[(\vec{F}.n)\Delta S\right]_{j+\frac{1}{2}} = \left[(\vec{F}.\frac{\nabla\xi}{|\nabla\xi|})\frac{|\nabla\xi|}{J}\right]_{j+\frac{1}{2}} = \left[F' - \frac{1}{Re}R'\right]_{j+\frac{1}{2}}$$

and likewise the remaining terms of this type can also be reduced to their corresponding semi-discrete representation in equation (3.3-2).

By comparing the semi-discrete form of equation (3.3-1) to the equation which resulted from the application of the integral conservation laws (3.3-4), the metrics terms can be evaluated geometrically. For instance the volume V, bounded by constant surfaces ξ and η, corresponds to J^{-1}. In addition, $\xi_x/|\nabla\xi|$, $\xi_y/|\nabla\xi|$, $\eta_x/|\nabla\eta|$ and $\eta_y/|\nabla\eta|$ are simply the direction cosines of the cell faces normal to their respective directions. Finally, $|\nabla\xi|/J$ and $|\nabla\eta|/J$ correspond to the cell face length normal to the direction of the unit vector.

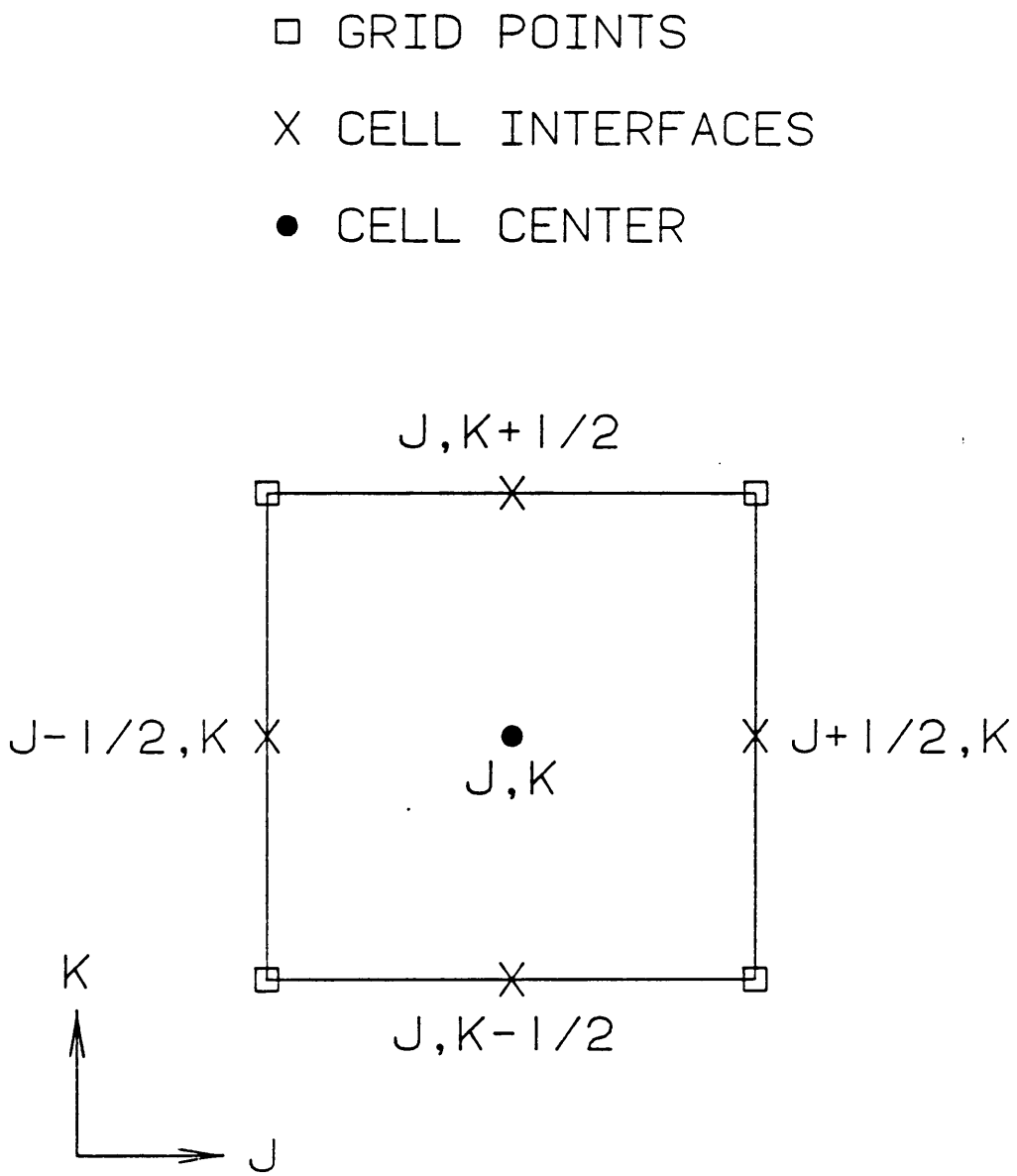


Figure 1. Cell centered finite volume representation

4.0 Shear Stress and Heat Flux Terms in the Navier Stokes Equations

The two dimensional form of the compressible Navier Stokes equations in non-dimensional conservation form is

$$\frac{\partial Q}{\partial t}+\frac{\partial F}{\partial x}+\frac{\partial G}{\partial y}=\frac{1}{Re}\left(\frac{\partial R}{\partial x}+\frac{\partial S}{\partial y}\right) \qquad (4.0-1)$$

Vectors R and S represent the viscous contributions from the following four governing equations: continuity, x momentum, y momentum, and energy. These equations are formulated in an cartesian coordinate system, which requires the solution domain be represented by an cartesian grid. This limitation is, however, not acceptable because most problems are more naturally solved on cartesian grids. In order to relieve the grid dependency problem, a coordinate transformation that maps the cartesian system of (x,y) to a generalized system of (ξ,η) can be performed. The results of the transformation applied to the viscous terms in equation (4.0-1) yield the following new set of equations

$$\frac{\partial R'}{\partial \xi}+\frac{\partial S'}{\partial \eta} \qquad (4.0-2)$$

$$R' = \begin{bmatrix} r'_1 \\ r'_2 \\ r'_3 \\ r'_4 \end{bmatrix}$$

$$r'_1 = 0$$

$$r'_2 = \mu\left[\left(\frac{4}{3}\frac{\eta_x\xi_x}{J} + \frac{\eta_y\xi_y}{J}\right)u_\eta + \left(\frac{\eta_x\xi_y}{J} - \frac{2}{3}\frac{\eta_y\xi_x}{J}\right)v_\eta \right]$$
$$+ \mu\left[\left(\frac{4}{3}\frac{\xi_x^2}{J} + \frac{\xi_y^2}{J}\right)u_\xi + \left(\frac{1}{3}\frac{\xi_x\xi_y}{J}\right)v_\xi \right]$$

$$r'_3 = \mu\left[\left(\frac{1}{3}\frac{\xi_x\xi_y}{J}\right)u_\xi + \left(\frac{\xi_x^2}{J} + \frac{4}{3}\frac{\xi_y^2}{J}\right)v_\xi \right]$$
$$+ \mu\left[\left(\frac{\eta_y\xi_x}{J} - \frac{2}{3}\frac{\eta_x\xi_y}{J}\right)u_\eta + \left(\frac{\xi_x\eta_x}{J} + \frac{4}{3}\frac{\eta_y\xi_y}{J}\right)v_\eta \right]$$

$$r'_4 = \mu\left[\frac{1}{2}\left[\left(\frac{4}{3}\frac{\xi_x^2}{J} + \frac{\xi_y^2}{J}\right)(u^2)_\xi + \left(\frac{4}{3}\frac{\xi_y^2}{J} + \frac{\xi_x^2}{J}\right)(v^2)_\xi \right] \right.$$
$$\left. + (Pr)^{-1}(\gamma-1)^{-1}\gamma\left(\frac{\xi_y^2}{J} + \frac{\xi_x^2}{J}\right)\left(\frac{P}{\rho}\right)_\xi + \left(\frac{1}{3}\frac{\xi_x\xi_y}{J}\right)(uv)_\xi \right]$$
$$+ \mu\left[\frac{1}{2}\left[\left(\frac{4}{3}\frac{\eta_x\xi_x}{J} + \frac{\eta_y\xi_y}{J}\right)(u^2)_\eta + \left(\frac{4}{3}\frac{\eta_y\xi_y}{J} + \frac{\eta_x\xi_x}{J}\right)(v^2)_\eta \right] \right]$$
$$+ \mu\left[\gamma(\gamma-1)^{-1}(Pr)^{-1}\left(\frac{\eta_x\xi_x}{J} + \frac{\eta_y\xi_y}{J}\right)\left(\frac{P}{\rho}\right)_\eta + \left(\frac{\eta_y\xi_x}{J} - \frac{2}{3}\frac{\eta_x\xi_y}{J}\right)(uv)_\eta \right.$$
$$\left. + \frac{5}{3}\left(\frac{\eta_x\xi_y}{J} - \frac{\eta_y\xi_x}{J}\right)uv_\eta \right]$$

$$S' = \begin{bmatrix} s'_1 \\ s'_2 \\ s'_3 \\ s'_4 \end{bmatrix}$$

$$s'_1 = 0$$

$$s'_2 = \mu\left[\left(\frac{4}{3}\frac{\eta_x\xi_x}{J} + \frac{\eta_y\xi_y}{J}\right)u_\xi + \left(\frac{\eta_y\xi_x}{J} - \frac{2}{3}\frac{\eta_x\xi_y}{J}\right)v_\xi\right] + \mu\left[\left(\frac{4}{3}\frac{\eta_x^2}{J} + \frac{\eta_y^2}{J}\right)u_\eta + \left(\frac{1}{3}\frac{\eta_x\eta_y}{J}\right)v_\eta\right]$$

$$s'_3 = \mu\left[\left(\frac{1}{3}\frac{\eta_x\eta_y}{J}\right)u_\eta + \left(\frac{\eta_x^2}{J} + \frac{4}{3}\frac{\eta_y^2}{J}\right)v_\eta\right] + \mu\left[\left(\frac{\eta_x\xi_y}{J} - \frac{2}{3}\frac{\eta_y\xi_x}{J}\right)u_\xi + \left(\frac{\xi_x\eta_x}{J} + \frac{4}{3}\frac{\eta_y\xi_y}{J}\right)v_\xi\right]$$

$$s'_4 = \mu\left[\frac{1}{2}\left[\left(\frac{4}{3}\frac{\eta_x^2}{J} + \frac{\eta_y^2}{J}\right)(u^2)_\eta + \left(\frac{4}{3}\frac{\eta_y^2}{J} + \frac{\eta_x^2}{J}\right)(v^2)_\eta\right] + (Pr)^{-1}(\gamma-1)^{-1}\gamma\left(\frac{\eta_y^2}{J} + \frac{\eta_x^2}{J}\right)\left(\frac{P}{\rho}\right)_\eta + \left(\frac{1}{3}\frac{\eta_x\eta_y}{J}\right)(uv)_\eta\right]$$
$$+ \mu\left[\frac{1}{2}\left[\left(\frac{4}{3}\frac{\eta_x\xi_x}{J} + \frac{\eta_y\xi_y}{J}\right)(u^2)_\xi + \left(\frac{4}{3}\frac{\eta_y\xi_y}{J} + \frac{\eta_x\xi_x}{J}\right)(v^2)_\xi\right]\right]$$
$$+ \mu\left[\gamma(\gamma-1)^{-1}(Pr)^{-1}\left(\frac{\eta_x\xi_x}{J} + \frac{\eta_y\xi_y}{J}\right)\left(\frac{P}{\rho}\right)_\xi + \left(\frac{\eta_x\xi_y}{J} - \frac{2}{3}\frac{\eta_y\xi_x}{J}\right)(uv)_\xi + \frac{5}{3}\left(\frac{\eta_y\xi_x}{J} - \frac{\eta_x\xi_y}{J}\right)uv_\xi\right]$$

The additional terms ξ_x, ξ_y, η_x , and η_y are the metrics which evolve from the transformation to (ξ,η) and are defined as

$$\frac{\xi_x}{J} = y_\eta ; \quad \frac{\xi_y}{J} = -x_\eta ; \quad \frac{\eta_x}{J} = -y_\xi ; \quad \frac{\eta_y}{J} = x_\xi \qquad (4.0-3)$$

where J, the Jacobian of the transformation, is

$$J = [x_\xi y_\eta - x_\eta y_\xi]^{-1}$$

A derivation of the coordinate transformation performed on the Navier Stokes equations to generalized coordinates along with a formal development of the metrics is provided in Appendix A.

4.1 Spatial Differencing of the Viscous Terms

The diffusive nature of the shear stress and heat flux terms allows for first derivatives to be treated using second-order central differences. The full viscous equations, however, involve double derivatives as well as cross derivatives. Treatment of these derivatives, in the finite volume formulation, will require differences across cell interfaces of the first derivative terms.

4.1.1 Double Derivatives in the Streamwise Direction

A typical term representing a double derivative in the streamwise direction is taken from equations (A.2-4) and (A.3-3).

$$\frac{\partial}{\partial \xi}\left[\mu\left(\frac{4}{3}\frac{\xi_x^2}{J} + \frac{\xi_y^2}{J}\right)u_\xi\right] \qquad (4.1-1)$$

A representative portion of the discretized domain in the finite volume method is shown in part A of Figure 2.

When equation (4.1-1) is applied at the central cell j,k, the Finite Difference representation will be as follows,

$$\frac{\left[\mu\left(\frac{4}{3}\frac{\xi_x^2}{J}+\frac{\xi_y^2}{J}\right)u_\xi\right]_{j+\frac{1}{2},k}-\left[\mu\left(\frac{4}{3}\frac{\xi_x^2}{J}+\frac{\xi_y^2}{J}\right)u_\xi\right]_{j-\frac{1}{2},k}}{\Delta\xi}$$

which is merely a flux balance across the vertical cell faces in the ξ direction about cell j,k. The remaining first derivative terms must be evaluated on the half intervals of j + ½ and j-½ along the cell face centers marked □. The cell face center location is used to obtain an average value across the entire face. These derivatives are easily handled as shown below.

$$u_\xi)_{j+\frac{1}{2},k}=\frac{u_{j+1,k}-u_{j,k}}{\Delta\xi}$$

$$u_\xi)_{j-\frac{1}{2},k}=\frac{u_{j,k}-u_{j-1,k}}{\Delta\xi}$$

These metric terms, which are also evaluated on the half interval, as well as all of the metric terms which appear in the equations, are handled geometrically in the Finite Volume Formulation.

4.1.2 Double Derivatives in the Transverse Direction

Double Derivatives in the transverse or η direction are handled very similarly to the preceding section. A typical term is shown below.

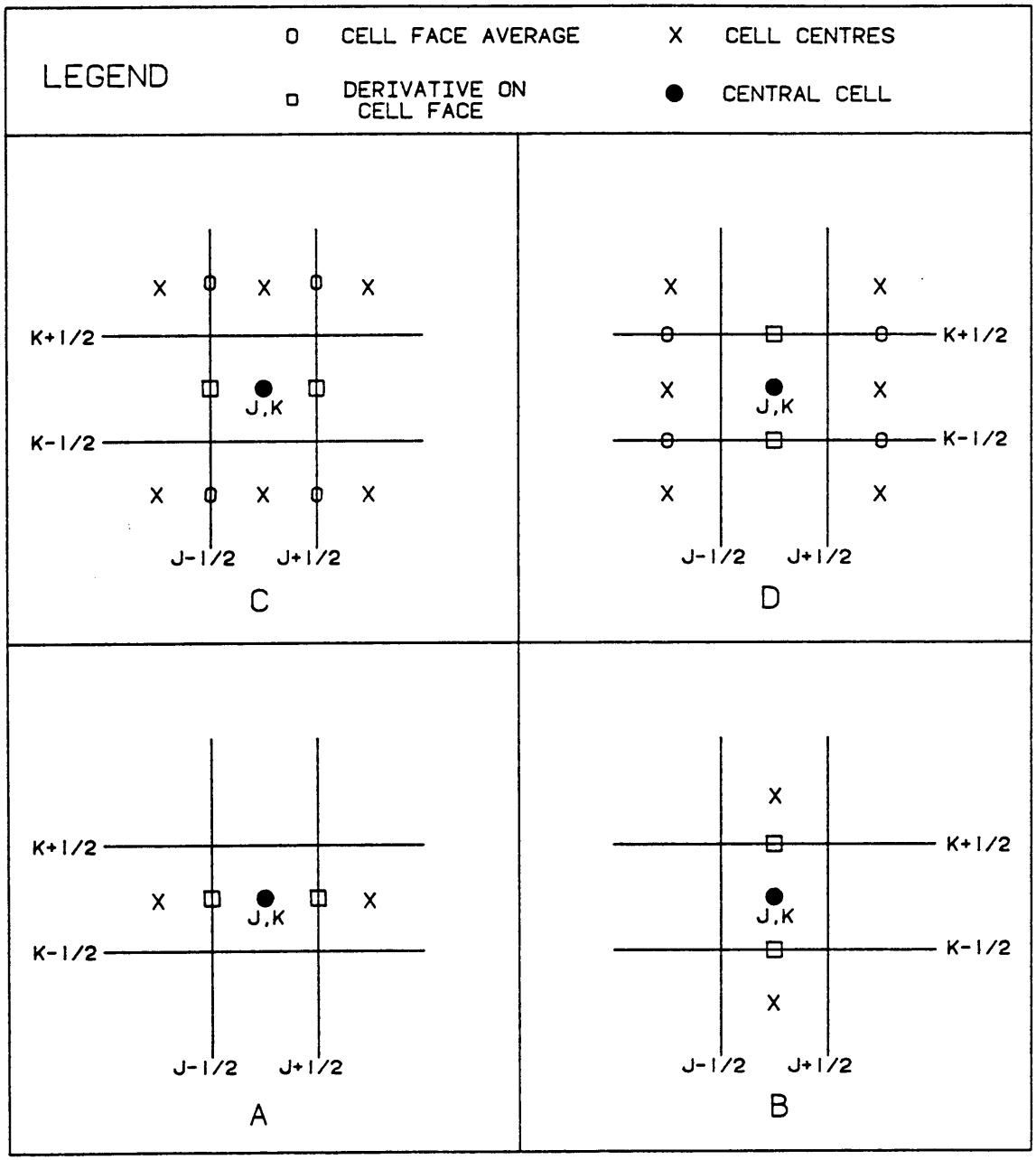


Figure 2. Spatial Differencing of Viscous Terms

$$\frac{\partial}{\partial \eta}\left[\mu\left(\frac{4}{3}\frac{\eta_x^2}{J}+\frac{\eta_y^2}{J}\right)u_\eta\right] \qquad (4.1-2)$$

which again comes from equations (A.2-4) and (A.3-3). The finite difference representations of equation (4.1-2) about cell j,k is

$$\frac{\left[\mu\left(\frac{4}{3}\frac{\eta_x^2}{J}+\frac{\eta_y^2}{J}\right)u_\eta\right]_{j,\,k+\frac{1}{2}}-\left[\mu\left(\frac{4}{3}\frac{\eta_x^2}{J}+\frac{\eta_y^2}{J}\right)u_\eta\right]_{j,\,k-\frac{1}{2}}}{\Delta\eta}$$

which is a flux balance across the horizontal faces in the η direction. The first derivative terms are formed on the cell face centers marked □ in part B of Figure 2, and are differenced as

$$u_\eta)_{j,\,k+\frac{1}{2}}=\frac{u_{j,\,k+1}-u_{j,\,k}}{\Delta\eta}$$

$$u_\eta)_{j,\,k-\frac{1}{2}}=\frac{u_{j,\,k}-u_{j,\,k-1}}{\Delta\eta}$$

4.1.3 Cross Derivatives

The cross derivative terms can be differenced either symmetrically or asymmetrically using second order differences, as shown by Mitchell and Griffiths [20]. Consideration is given to a scalar model equation for viscous flows and the difference representations for both methods is outlined in reference [3]. The analysis shows that the symmetrical treatment results in a loss of diagonal dominance in the coefficient matrix; however, the method is unconditionally stable [23]. On the other hand, the asymmetric treatment can be utilized so that diagonal dominance is

maintained and with assured unconditional stability. The issue of diagonal dominance is extremely important to the convergence rate. Asymmetric treatment will, therefore, converge to the steady state solution faster that the symmetric treatment as long as the conditions for diagonal dominance are met [3].

The cross derivative terms associated with the Navier Stokes equations are handled using either technique. An actual comparison of the benefits or drawbacks between the two methods as applied to the NS equations has not been found. Therefore, for no reason other than programming simplicity, a symmetric treatment of the cross derivative term will be used.

A typical cross derivative terms which appears in equations (A.2-4) and (A.3-3) is

$$\frac{\partial}{\partial \xi}\left[\mu\left(\frac{4}{3}\frac{\eta_x \xi_x}{J}+\frac{\eta_y \xi_y}{J}\right)u_\eta\right] \qquad (4.1-3)$$

which has a finite difference equation

$$\frac{\left[\mu\left(\frac{4}{3}\frac{\eta_x \xi_x}{J}+\frac{\eta_y \xi_y}{J}\right)u_\eta\right]_{j+\frac{1}{2},k}-\left[\mu\left(\frac{4}{3}\frac{\eta_x \xi_x}{J}+\frac{\eta_y \xi_y}{J}\right)u_\eta\right]_{j-\frac{1}{2},k}}{\Delta\xi}$$

This equation is different from the double derivatives in the ξ direction in that the flux balance across the vertical faces in ξ direction involves additional first derivatives in the η direction. Referring to part C of Figure 2, u needs to be defined on the vertical face centers in order to form u_η on the half intervals marked □. This is accomplished by simply averaging the cell centered velocities, shown as X, to obtain the value on the cell interface O, and then forming the derivatives. This is illustrated below.

$$u_\eta)_{j+\frac{1}{2},k} = \frac{u_{j+\frac{1}{2},k+1} - u_{j+\frac{1}{2},k-1}}{2\Delta\eta}$$

where

$$u_{j+\frac{1}{2},k+1} = \frac{u_{j,k+1} + u_{j+1,k+1}}{2}$$

Similar treatment is used in evaluating $u_\eta)_{j-\frac{1}{2},k}$.

The final form of a cross derivative term which will appear in equations (A.2-4) and (A.3-3) is

$$\frac{\partial}{\partial\eta}\left[\mu\left(\frac{\eta_x\xi_y}{J} - \frac{2}{3}\frac{\eta_y\xi_x}{J}\right)u_\xi\right] \qquad (4.1-4)$$

with a Finite Difference representation

$$\frac{\left[\mu\left(\frac{\eta_x\xi_y}{J} - \frac{2}{3}\frac{\eta_y\xi_x}{J}\right)u_\xi\right]_{j,k+\frac{1}{2}} - \left[\mu\left(\frac{\eta_x\xi_y}{J} - \frac{2}{3}\frac{\eta_y\xi_x}{J}\right)u_\xi\right]_{j,k-\frac{1}{2}}}{\Delta\eta}$$

Evaluating the flux across the horizontal cell faces centers, denoted as □ on part D of Figure 2, involves forming first derivatives of u in the ξ direction. Proceeding as before, this is accomplished by first averaging the cell centered velocities marked X and then forming the derivatives. This formulation is shown below.

$$u_\xi)_{j,k+\frac{1}{2}} = \frac{u_{j+1,k+\frac{1}{2}} - u_{j-1,k+\frac{1}{2}}}{2\Delta\xi}$$

where

$$u_{j+1,\,k+\frac{1}{2}} = \frac{u_{j+1,\,k+1} + u_{j+1,\,k}}{2}$$

The spatial differencing for the various types of viscous terms encountered in the Navier Stokes equations has been demonstrated. The additional terms necessary to obtain complete Navier Stokes solutions have been implemented in the computing code. Validation of the method is provided in Appendix B by applying the code to a double throat nozzle and comparing the results to existing NS solutions.

5.0 Two Dimensional Shock Boundary Layer Interaction

The phenomenon of separation due to shock impingement on boundary layers is an extremely complicated flow situation to numerically analyze. Because of the existence of this type of reaction in high speed flows, it is important to obtain accurate computational results for performance studies. During the reaction, the flow first experiences a compression zone as a result from a separating boundary layer. The flow then expands over a separation bubble before a final recompression due to turning at the wall.

5.1 Experimental Background

Extensive experiments have been performed to study the phenomenon of flat plate shock boundary layer interactions. Most recently, Skebe et. al. [21] performed a series of these tests in a Supersonic Wind Tunnel at NASA Lewis Research Center. The range of flow conditions included Mach numbers from 2 to 4 and overall unit Reynolds numbers from 4.72 E06 to 2.95 E07 /m. The investigation encompassed initially laminar, turbulent and transitional boundary layers with the primary objective

of creating a detailed data base for computational and analytical study. A very thorough description of the experimental procedures and results including the tunnel conditions, two-dimensionality, shock-boundary layer development and instrumentation can be found in reference [22].

5.2 *Introduction and Problem Description*

The following computational study examines the flow physics of an initially laminar flat plate shock boundary layer interaction. The configuration of the shock generator and flat plate that produces the phenomenon of two dimensional shock boundary layer interactions is shown in Figure 3. Various shock strengths are obtained by changing the angular orientation of the shock generator to the flat plate. In this study, the Mach 2.0 flow experiences three shock strengths that range from a very weak oblique shock wave interaction to a mild boundary layer separation and reattachment. In all cases presented, however, the flow downstream of the shock impingement point remains laminar and thus transition and turbulence effects are not modeled and do not enter the calculation. The details of the reference conditions for the test cases are provided in Table 1.

5.3 *Boundary Conditions*

The boundary conditions used on this problem are as follows. Since the inflow conditions are supersonic, they are set explicitly at the entrance boundary. Adiabatic, no-slip conditions are applied at the bottom and top surfaces corresponding to the

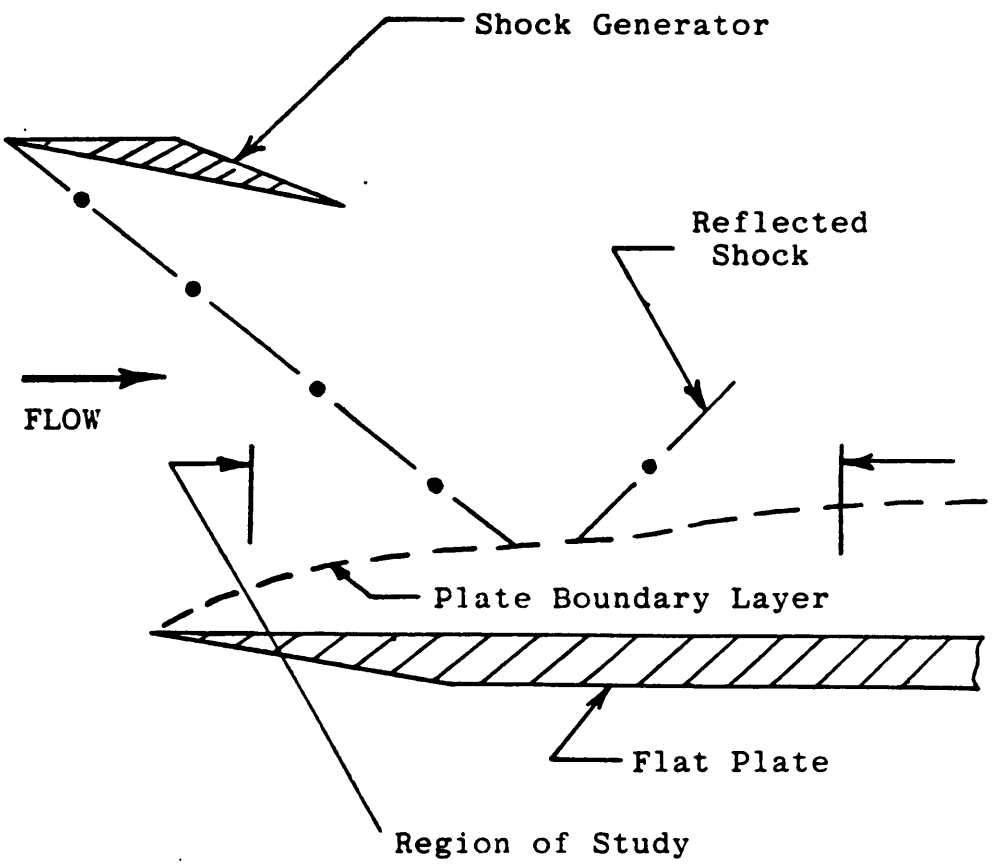


Figure 3. Flat Plate Shock Boundary Layer Configuration from Ref. [22]

Table 1. Flow Conditions For Shock Boundary Layer Problem

Parameter	Case I	Case II	Case III
Shock Generator Angle	0.00	0.56	1.00
Free Stream Mach Number	1.967	1.968	1.966
Free Stream Dynamic Pressure	1.755 E04 Pa	1.755 E04 Pa	1.755 E04 Pa
Reynolds Number per Unit Length	6.0014 E06 /M	6.0447 E06 /M	6.0829 E06 /M
Free Stream Stagnation Pressure	4.8191 E04 Pa	4.8225 E04 Pa	4.8221 E04 Pa

wall and shock generator, respectively. Because the outflow is predominantly supersonic, a direct extrapolation of all state quantities is appropriate.

An alternate method of enforcing a shock wave, as compared to physically modeling the shock generator, is applying the appropriate shock jump conditions. Specifically, this implies that all variables below the impinging shock location are set to the freestream values and all of the variables above this location are set to the jump conditions. The upper surface, in this case, would be overspecified and set to the jump condition. This standard technique is very effective usually resulting in smaller grid domains saving storage and computational time. Specifying the jump conditions to construct the shock wave is preferred ; however, the experimental data indicates a weak oblique shock interaction for a shock generator angle of zero degrees. This effect is caused by aerodynamic loading, shock generator boundary layer, and non-zero leading edge radius of curvature. To produce computational results for the zero degree case would require trial and error jump conditions until the results agreed with the experimental data. In order to apply the correct jump condition for non-zero angles, an additional correction would have to be implemented to account for the increased interaction. Therefore, the geometry of the shock generator inclined at the correct deflection angles will be used as the top boundary alleviating the need for any additional correction factors.

5.4 *Computational Mesh*

The experimental grids used in this study are shown in Figure 4. The coarse grid is composed of 80 equally spaced points in the streamwise direction and 51 points in the normal direction. Algebraic stretching is provided on both surfaces to resolve the

complex viscous flows in the boundary layer. In an attempt to capture more of the flow physics, a very fine grid will be implemented. This grid contains 160 points in the streamwise direction and are vertically packed in the shock impingement interacting regions. Additionally, 91 points are used in the normal direction and are stretched on both walls. This nearly doubles the grid density and maximizes the available computing storage.

5.5 *Computational Results*

Computational results have been obtained for a laminar shock boundary layer interaction with a freestream Mach number of 2.0 and a Reynolds number per unit length of 6.0 E06/m. A vertical line Gauss Seidel relaxation method was used to achieve steady state thin layer and complete Navier-Stokes solutions. Third-order upwind biased differencing for the pressure and convective terms in both directions was implemented using flux vector splitting by Roe [15]. The shear stress and heat flux terms have been central differenced, as described in section 4.0.

The three cases that are examined in this study exhibit increasing pressure gradients corresponding to shock generator angles of 0.00, 0.56, and 1.00 degree. Computational results for each case are presented in graphical form comparing the numerical solutions for static wall pressure and skin friction to the experimental data. The first investigation examines TLNS solutions using two grid densities. The results are displayed in Figures 5 through 7. Figures 8 through 10 compare the TLNS and NS results on the fine mesh. PNS solutions are compared to full NS solutions in Figures 11 through 13. PNS and NS static pressure contours are provided in Figures 14 through 16.

5.6 *Discussion of Results*

5.6.1 Shock angle 0.00

A shock generator angle of 0.00 degrees resulted in a very weak oblique shock interaction on the flat plate. The computational results in Figures 5 and 8 predict this interaction and compare well with the experimental data. The grid sensitivity study reveals improved results for the fine mesh and the full NS solution does not reveal any additional physics in the flow. It is interesting to note that the PNS results in Figures 11 and 14 slightly overpredict the location of the shock impingement point due to suppressing part of the streamwise pressure gradient in the subsonic region of the boundary layer.

5.6.2 Shock Angle 0.56

The oblique shock wave produced from a shock generator inclined at 0.56 degrees results in near incipient separation of the boundary layer. The TLNS solution in Figure 6 predicts this result and then recovers the laminar boundary layer which is consistent with the near zero pressure gradient. The fine mesh provides more accurate results for the flowfield; however, the NS results in Figure 9 continue to match the thin layer solution. The PNS results in Figure 12 do not compare well with the experimental data. Static pressure contours for this shock angle are shown in Figure 15 and demonstrate the incorrect location of the shock wave from the PNS results.

5.6.3 Shock Angle 1.00

A very slight shock induced separation occurs on the flat plate for a 1.0 degree shock generator angle. Notice that the thin-layer results in Figure 7 do not predict the separation using the coarse mesh. In contrast, the finer mesh does resolve a small separated region slightly downstream from the predicted experimental results. The laminar reattachment is evidenced in the near zero pressure gradient. The NS solution in Figure 10 resolves no additional flow features. The PNS skin friction results in Figure 13 do not predict separation and in general do not produce accurate results for this shock angle. Finally, the contour data presented in Figure 16 contrast a PNS and NS static pressure field. Note that the PNS calculation continually overpredicts the shock contact point.

5.7 Code Performance and Convergence Criteria

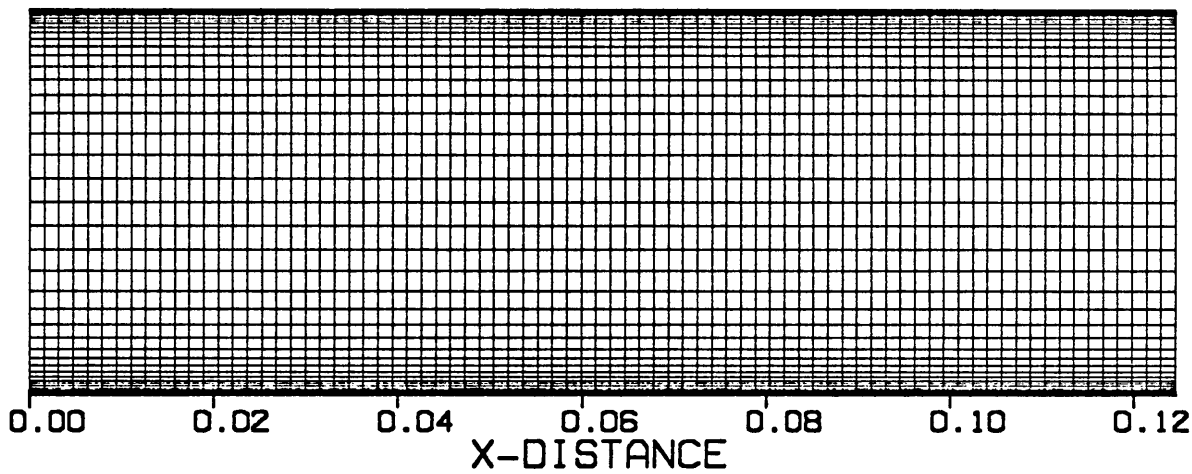
All of the preceding results were computed on an IBM 3090 vector processor. The vector mode of operation increases the computational rate by a factor of seven as compared to scalar.

The thin layer solutions for the shock boundary layer problem were obtained using PNS results as an initial solution. It is estimated that this technique resulted in a computational savings of 50 percent. Increased convergence to the thin layer solutions could have been achieved by utilizing the multisweep PNS option which simply adds that part of the pressure gradient which has been neglected after each sweep across the mesh. Similarly, the complete NS solutions were obtained from the thin-layer solutions.

The convergence criteria for all PNS solutions was defined by a six order reduction of the L_2 norm of the residual. Global solutions (TLNS and NS) were accepted after five orders. Solutions were analyzed where the L_2 norm was reduced to machine zero, (approximately twelve orders of magnitude); however, the results were indistinguishable. The total number of iterations to convergence varied from 300 to 1200 for the thin layer solutions, and an additional 300 to 500 for the NS solutions.

5.8 *Conclusions*

The computational results presented on the shock boundary layer interaction agree well for the three shock generator angles of 0.00, 0.56, and 1.00 degree. At worst, the initially laminar boundary layer experiences a mild separation; however, the physics are captured well using the thin-layer forms of the NS equations. The grid sensitivity study revealed more accurate descriptions of the flow using a finer mesh and the complete NS solutions did not offer additional resolution of the flow. This is expected due to the high Reynolds number and the lack of 'strong' viscous/inviscid interaction. The static pressure contours indicate that shock boundary layer interacting regions are not accurately predicted with the PNS space marching technique due to suppression of the upstream influence from the downstream pressure gradient in the subsonic region.



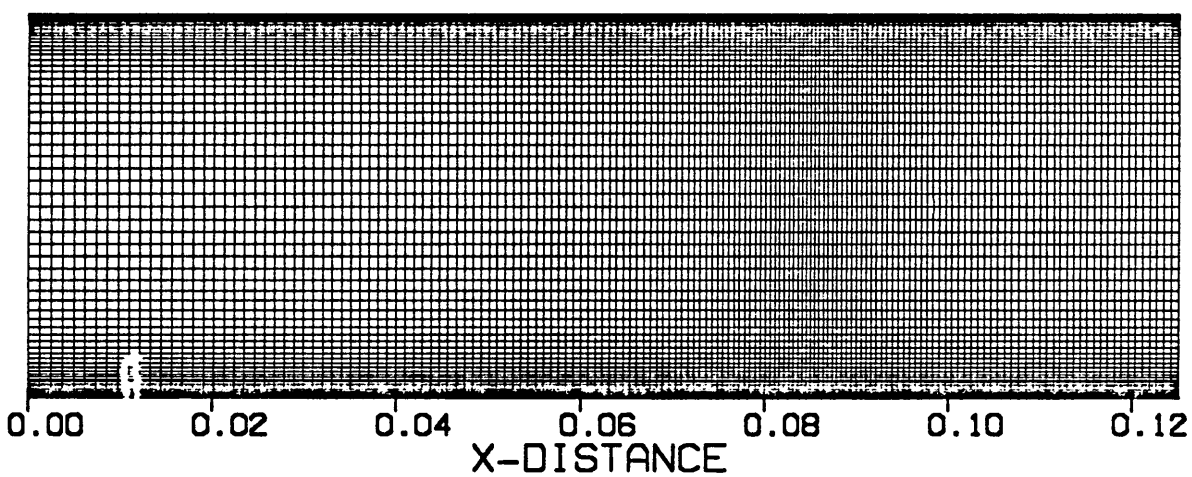


Figure 4. Computational Grids : 160 X 91 ; 80 X 51

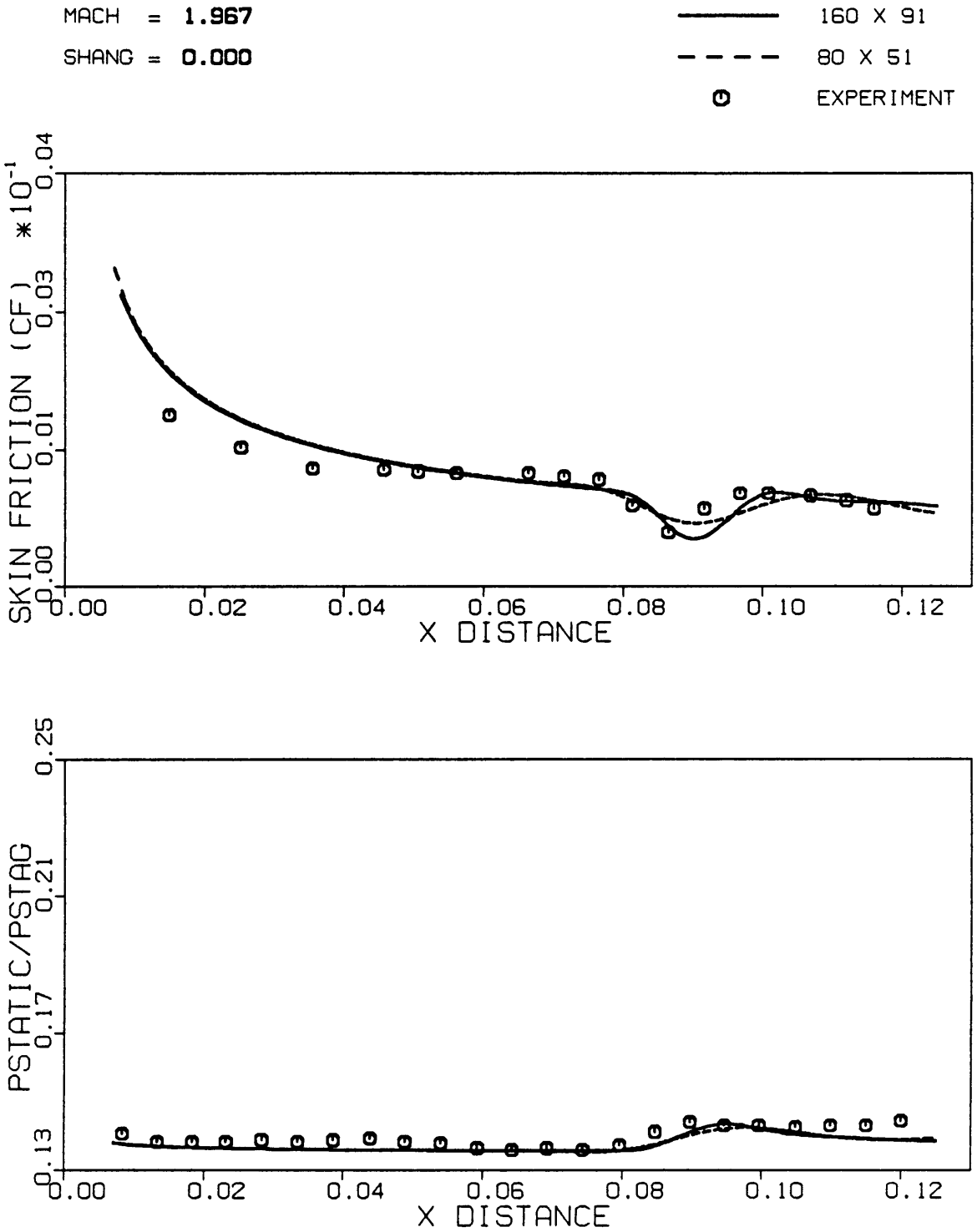


Figure 5. Wall Properties ; TLNS Grid Study ; 0.00 Degrees

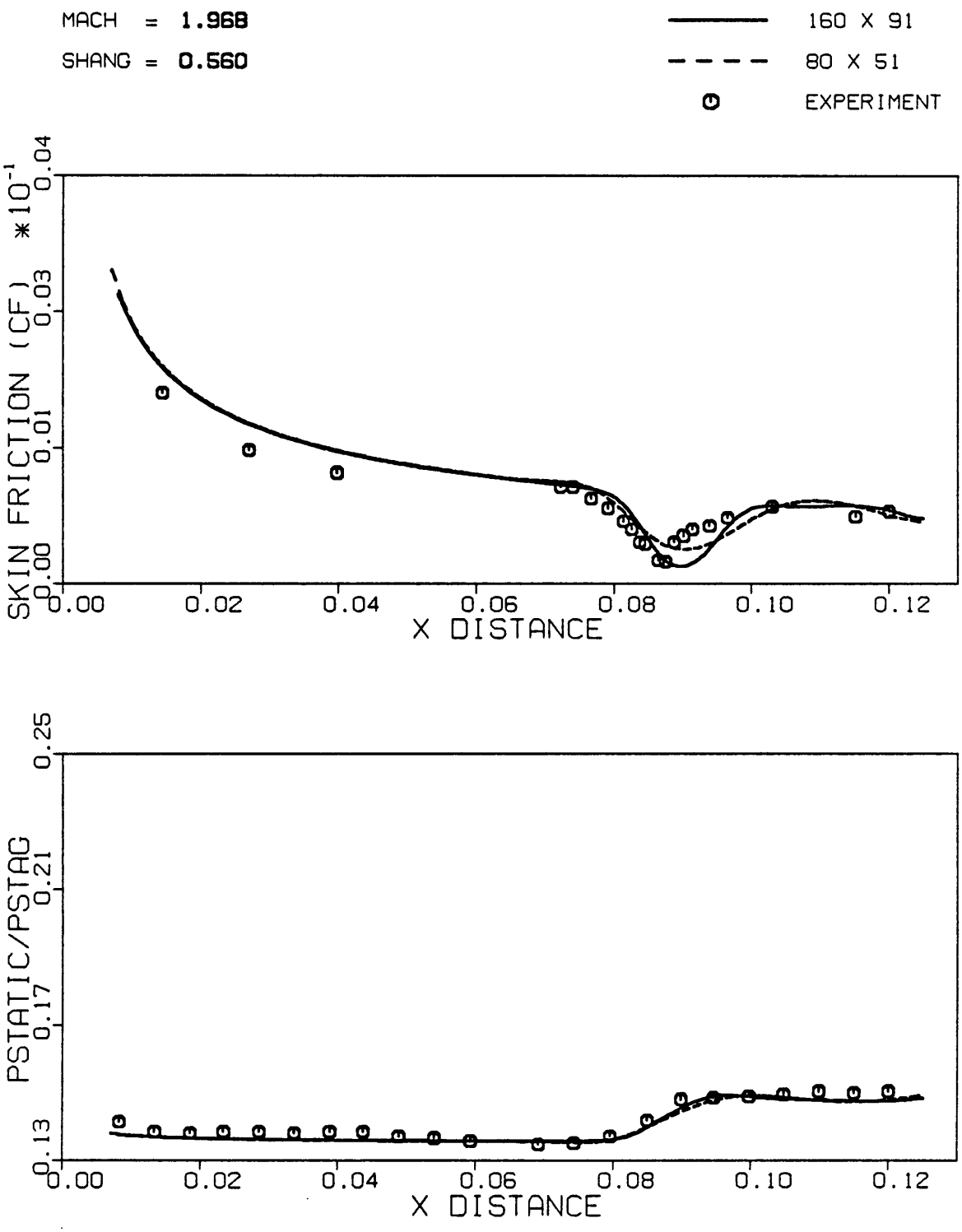


Figure 6. Wall Properties ; TLNS Grid Study ; 0.56 Degrees

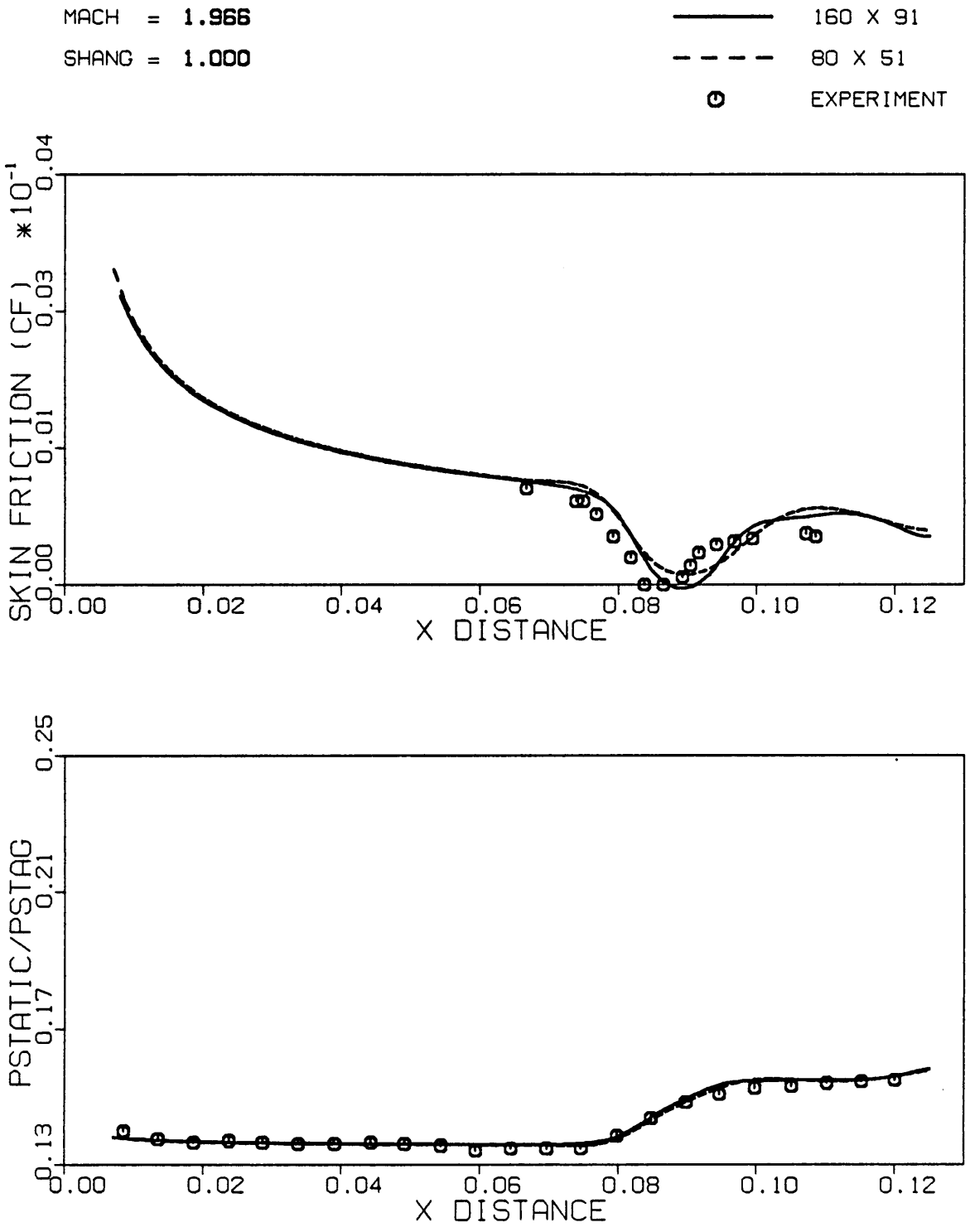


Figure 7. Wall Properties ; TLNS Grid Study ; 1.00 Degrees

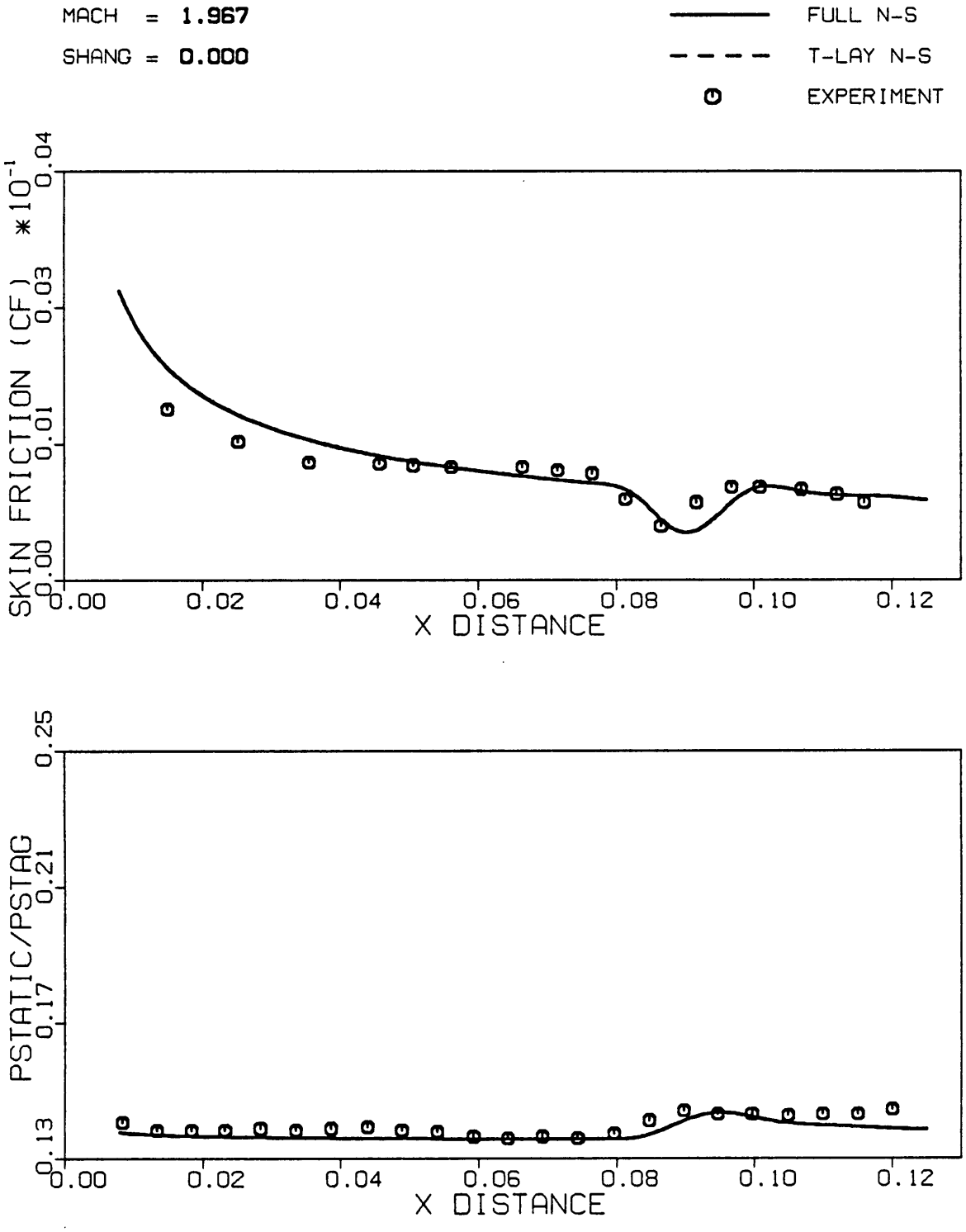


Figure 8. Wall Properties ; NS vs TLNS ; Grid 160 X 91 ; 0.00 Degrees

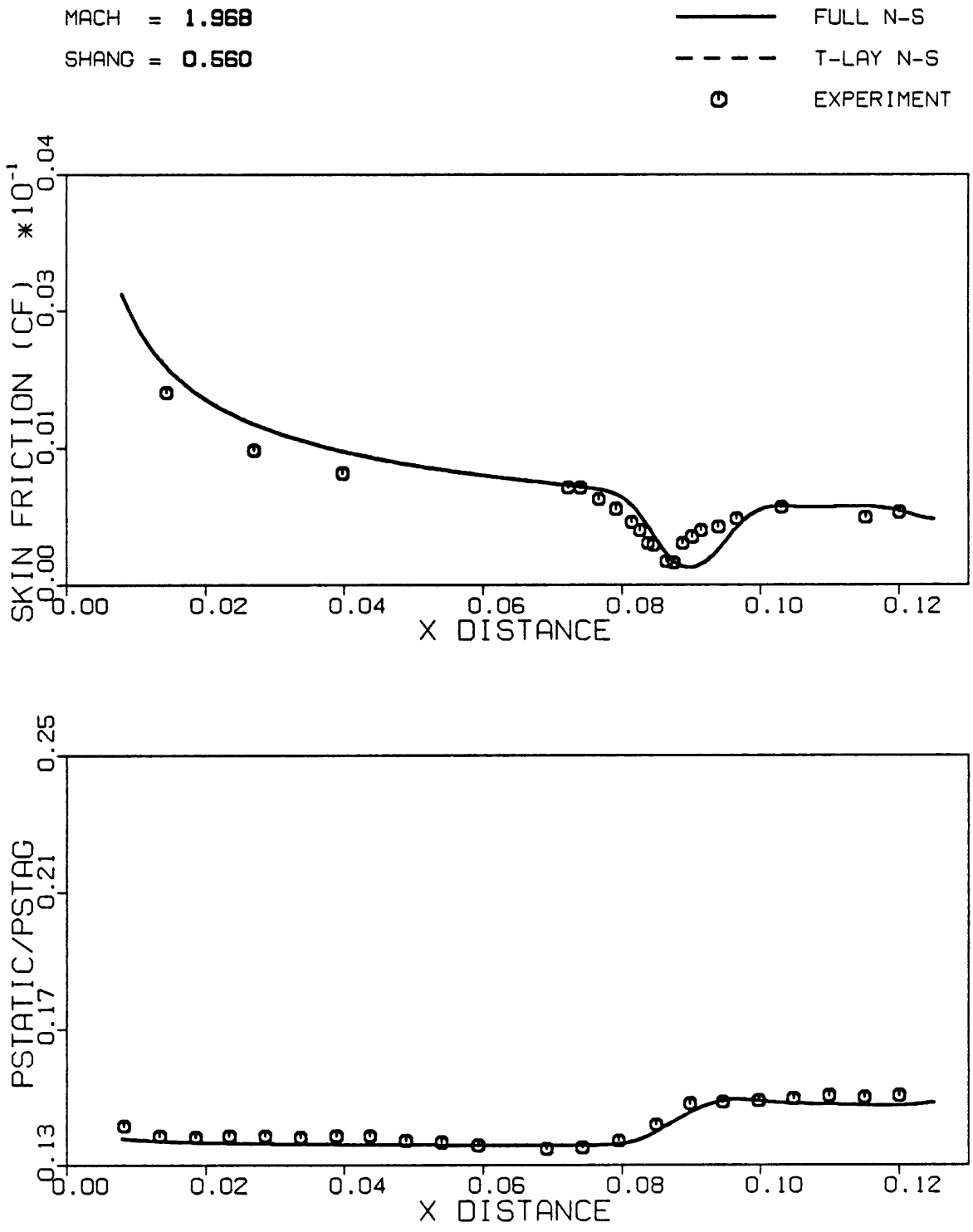


Figure 9. Wall Properties ; NS vs TLNS ; Grid 160 X 91 ; 0.56 Degrees

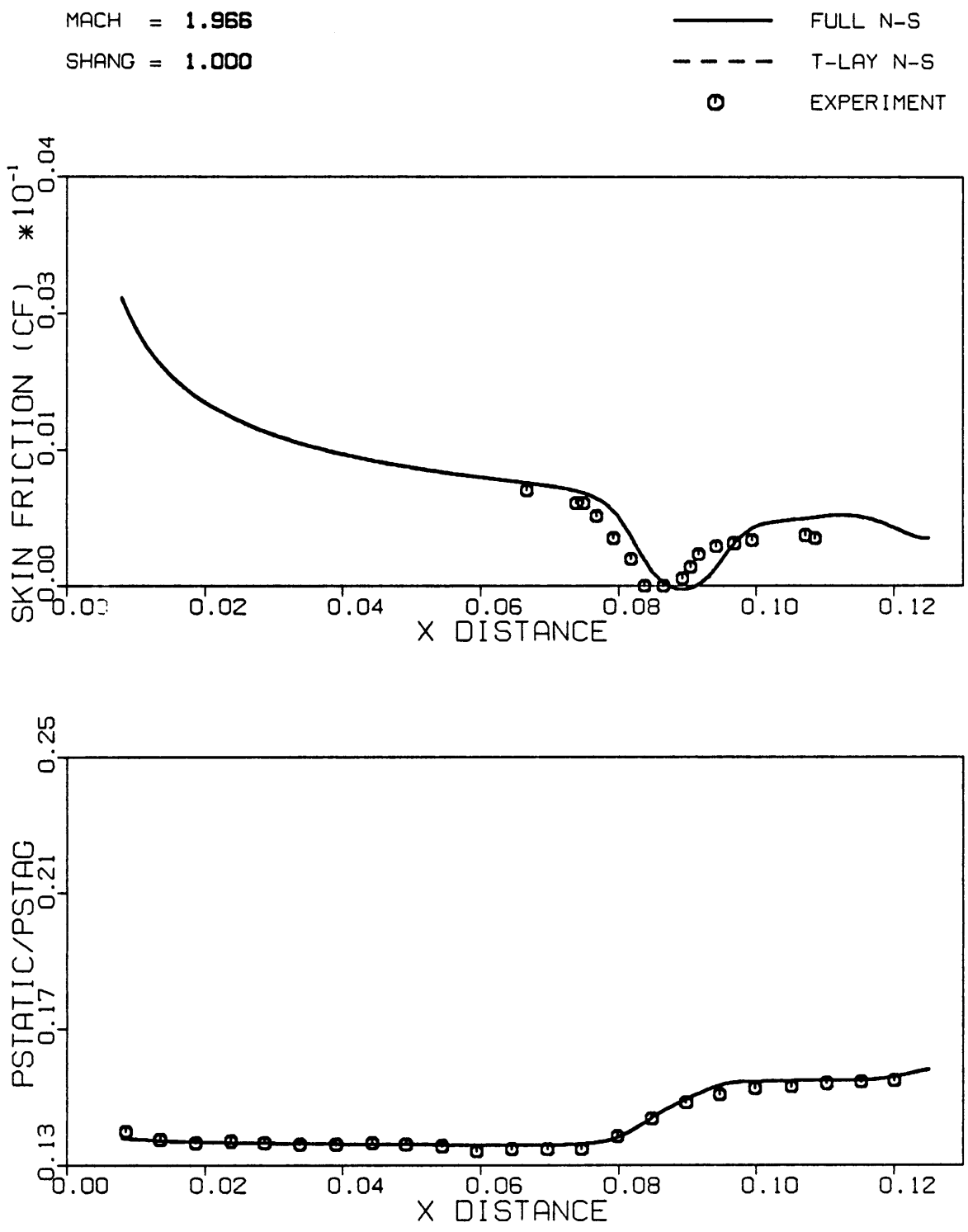


Figure 10. Wall Properties ; NS vs TLNS ; Grid 160 X 91 ; 1.00 Degrees

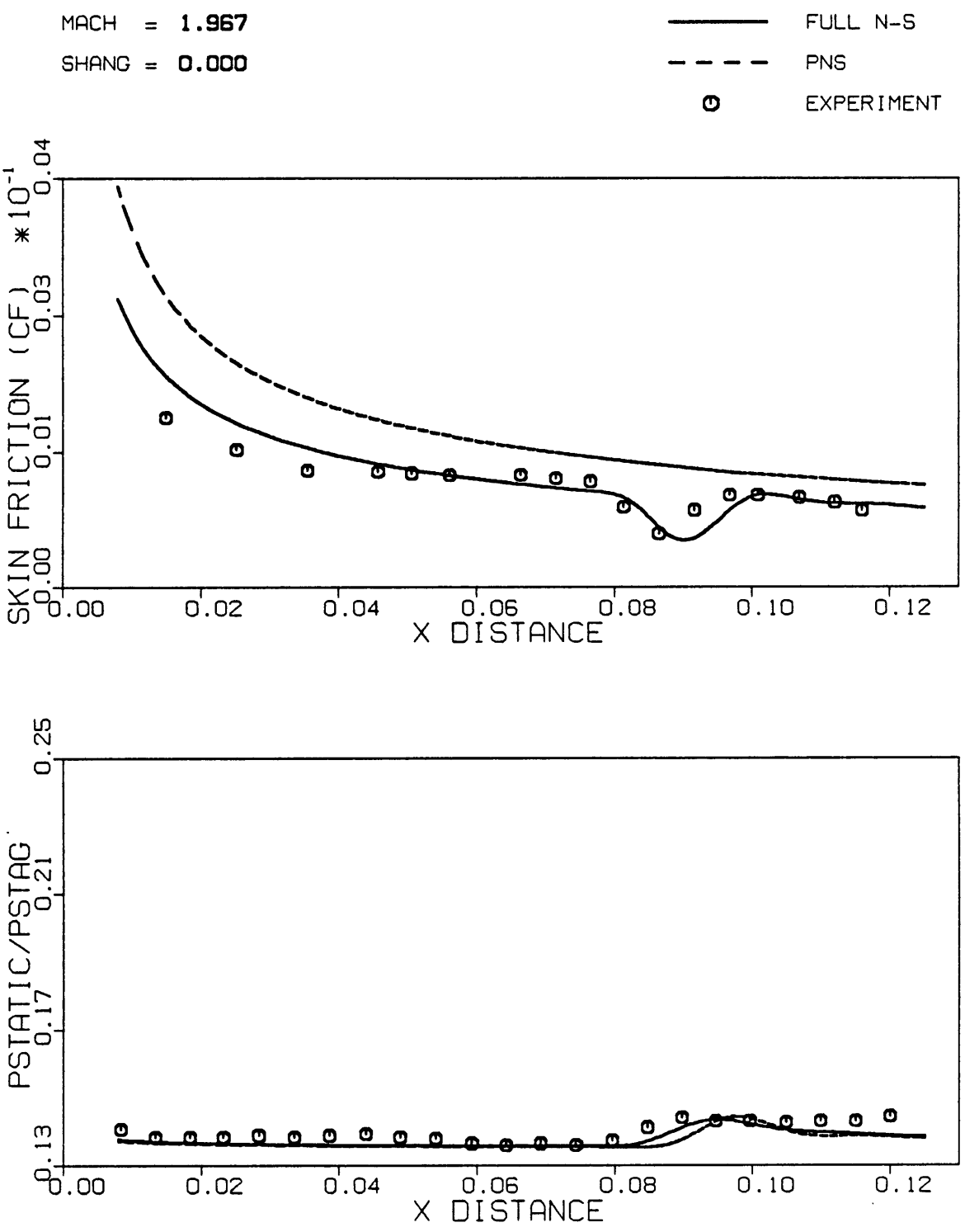


Figure 11. Wall Properties ; NS vs PNS ; Grid 160 X 91 ; 0.00 Degrees

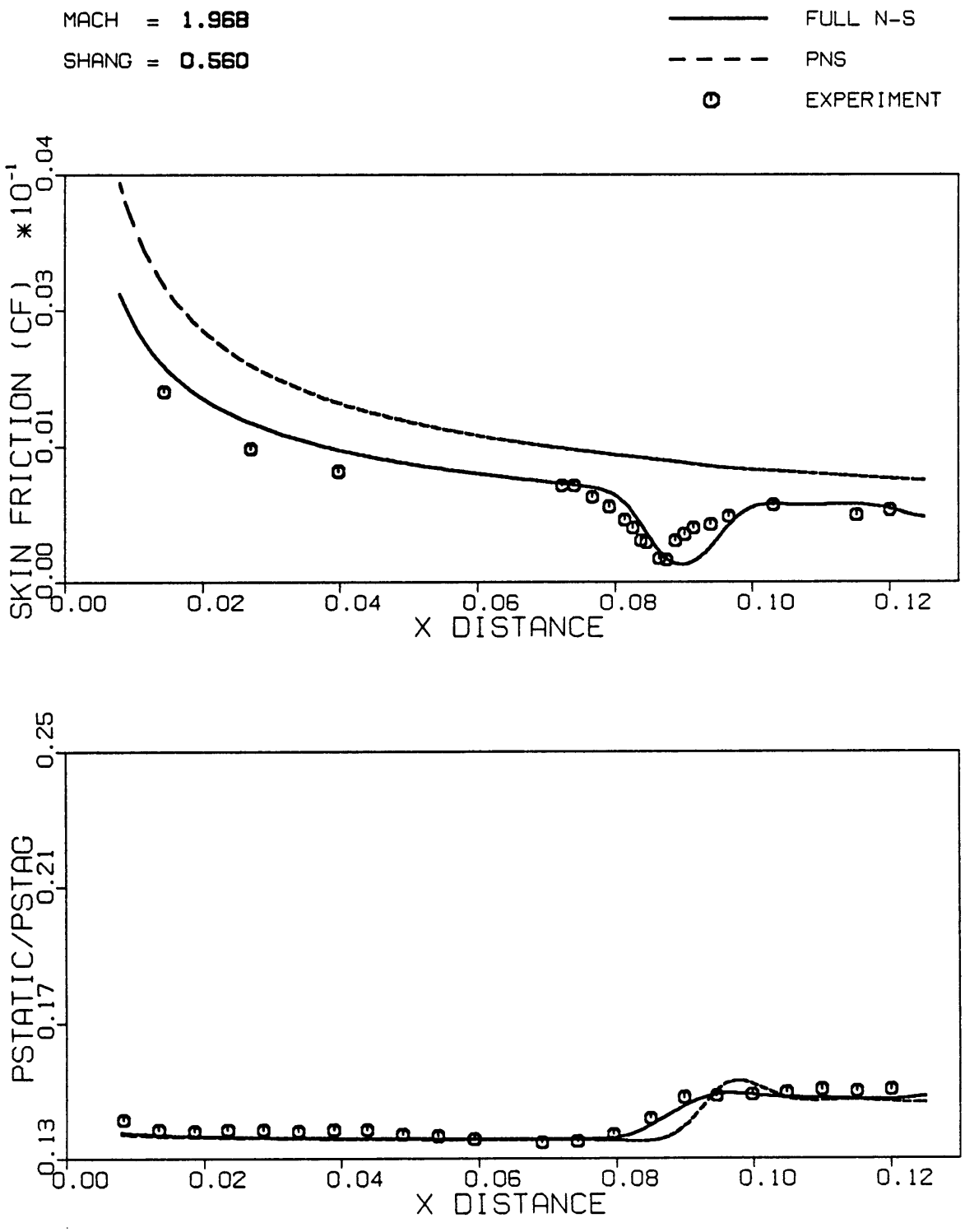


Figure 12. Wall Properties ; NS vs PNS ; Grid 160 X 91 ; 0.56 Degrees

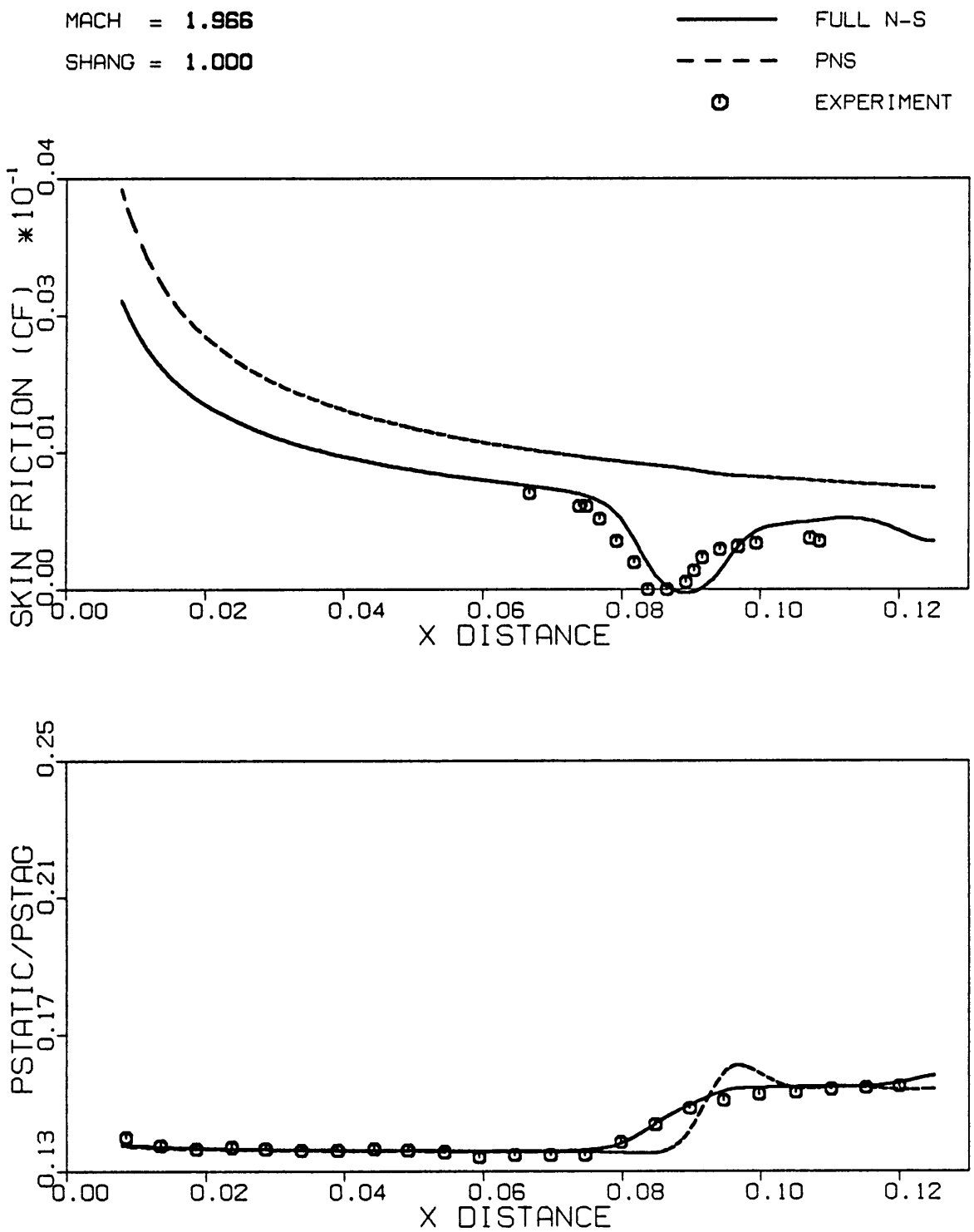


Figure 13. Wall Properties ; NS vs PNS ; Grid 160 X 91 ; 1.00 Degrees

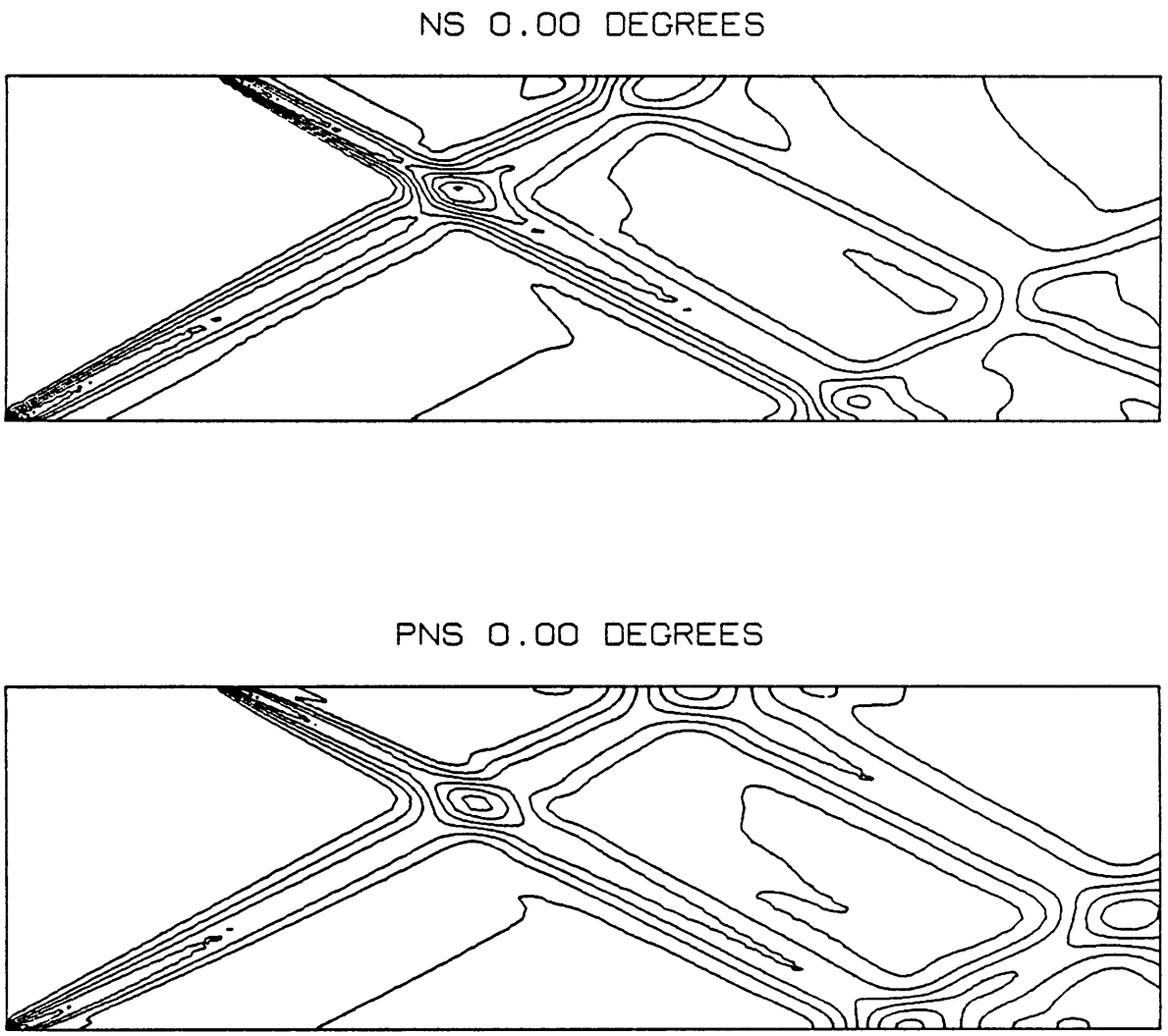


Figure 14. Static Pressure Contours ; Grid 160 x 91 ; 0.00 degrees

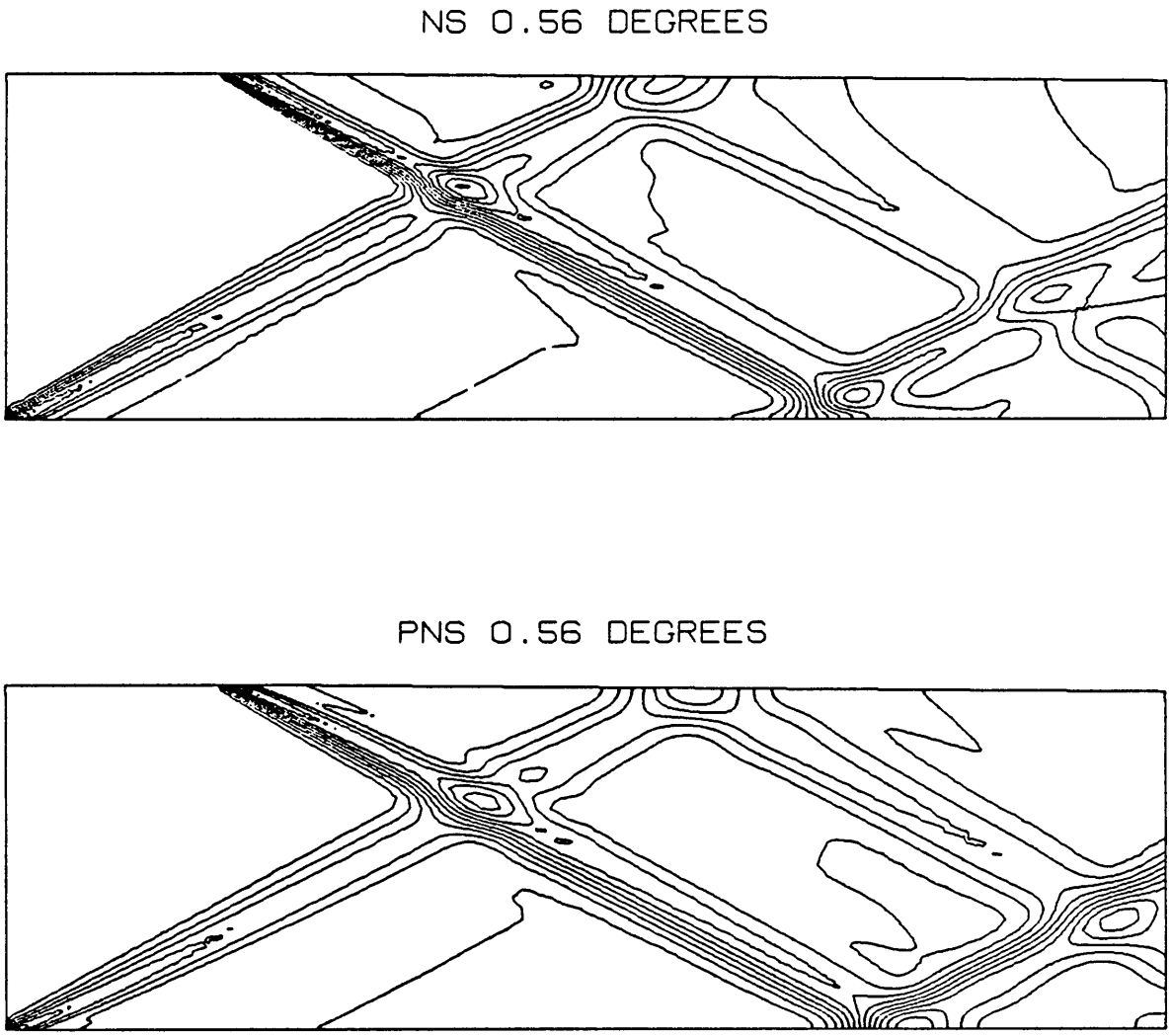


Figure 15. Static Pressure Contours ; Grid 160 x 91 ; 0.56 degrees

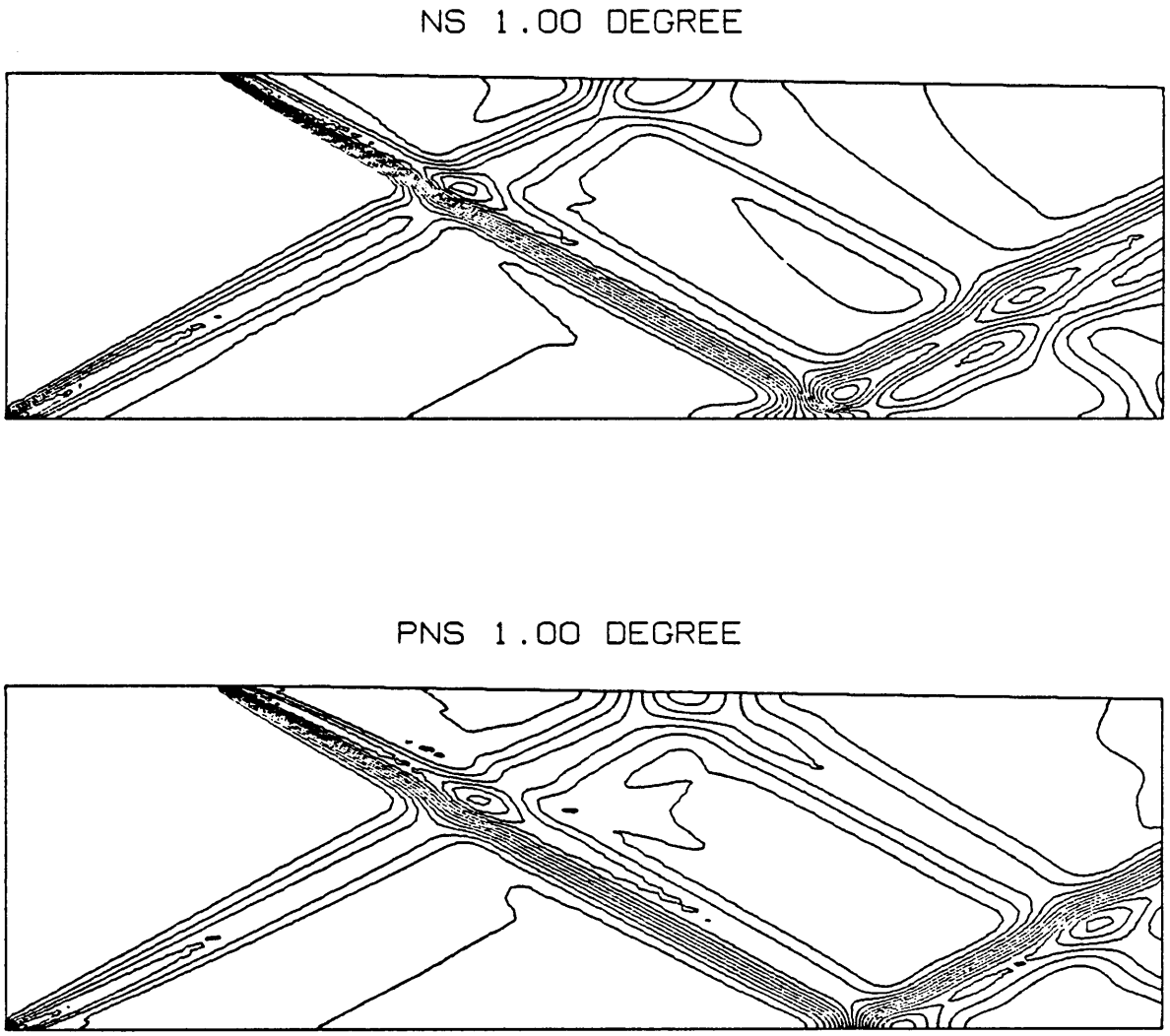


Figure 16. Static Pressure Contours ; Grid 160 x 91 ; 1.00 degrees

6.0 Final Conclusions and Recommendations

Complete Navier-Stokes solutions are obtainable using an upwind finite volume technique. Implementation of the required terms to complete the shear stress and heat flux contributions in the NS equations has been successfully added and validated against existing NS solutions for a double throat nozzle. The transformation processes and discretization techniques have been summarized which concludes the major effort in this thesis. An evaluation and comparison of the conservative and non-conservative handling of the cross derivatives would be recommended for future work. In addition, a study of the computation time versus convergence rate that results from linearizing the cross derivatives would be interesting.

Numerical computations of a laminar shock boundary layer interaction compared well with experimental data. The range of shock angles studied in this investigation do not require solutions to the complete form of the NS equations. In fact, there was no indication of any difference in the thin-layer and full NS solutions. In contrast, PNS solutions do not provide accurate results and consistently over predict the shock boundary layer interface region. Only laminar flow cases were considered in this study; however, transitional and turbulent flows could be investigated for future research.

Appendix A. Transformation Of The Viscous Terms of The Full NS Equations To Generalized Coordinates

A.1 Shear Stress and Heat Flux Terms

The non-dimensional form of the two dimensional time dependent compressible Navier Stokes equations cast in full conservation law form is provided below in Cartesian coordinates

$$\frac{\partial Q}{\partial t} + \frac{\partial F}{\partial x} + \frac{\partial G}{\partial y} = \frac{1}{Re}\left(\frac{\partial R}{\partial x} + \frac{\partial S}{\partial y}\right) \qquad (A.1-1)$$

where the shear stress and heat flux terms are

$$R = \begin{bmatrix} r_1 \\ r_2 \\ r_3 \\ r_4 \end{bmatrix} = \begin{bmatrix} 0 \\ \tau_{xx} \\ \tau_{xy} \\ r_4 \end{bmatrix} \qquad S = \begin{bmatrix} s_1 \\ s_2 \\ s_3 \\ s_4 \end{bmatrix} = \begin{bmatrix} 0 \\ \tau_{xy} \\ \tau_{yy} \\ s_4 \end{bmatrix}$$

$$\tau_{xx} = 2\mu \frac{\partial u}{\partial x} + \lambda\left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}\right),$$

$$\tau_{yy} = 2\mu \frac{\partial v}{\partial y} + \lambda\left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}\right),$$

$$\tau_{xy} = \mu\left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right),$$

$$r_4 = u\tau_{xx} + v\tau_{xy} + \frac{\mu}{Pr}\frac{1}{(\gamma - 1)}\frac{\partial a^2}{\partial x},$$

$$s_4 = u\tau_{xy} + v\tau_{yy} + \frac{\mu}{Pr}\frac{1}{(\gamma - 1)}\frac{\partial a^2}{\partial y}.$$

Implementation of Stokes' hypothesis to the bulk viscosity yields $\lambda = -2\mu/3$. In addition, Re and Pr represent the Reynolds and Prandtl number, respectively. Finally, γ is the ratio of specific heats.

A.2 Transformation to Generalized Coordinates

The two dimensional transformation from Cartesian coordinates (x,y) to generalized coordinates (ξ,η) is accomplished by simple applications of the chain rule in the (ξ,η) coordinate system. Within this system, ξ corresponds to the streamwise direction and η corresponds to the normal direction, thus mapping (ξ,η) into (x,y).

$$\xi = \xi(x,y)$$

$$\eta = \eta(x,y)$$

$$x = x(\xi, \eta)$$

$$y = y(\xi, \eta)$$

Applying the chain rule of differentiation to the viscous terms (right hand side) of equation (A.1-1) yields

$$\frac{\partial R}{\partial x} + \frac{\partial S}{\partial y} = \frac{\partial \xi}{\partial x}\frac{\partial R}{\partial \xi} + \frac{\partial \eta}{\partial x}\frac{\partial R}{\partial \eta} + \frac{\partial \xi}{\partial y}\frac{\partial S}{\partial \xi} + \frac{\partial \eta}{\partial y}\frac{\partial S}{\partial \eta} \qquad (A.2-1)$$

and for convenience the derivatives will be represented as

$$\frac{\partial \xi}{\partial x} = \xi_x,$$

$$\frac{\partial \eta}{\partial x} = \eta_x,$$

$$\frac{\partial R}{\partial \xi} = R_\xi,$$

$$\frac{\partial R}{\partial \eta} = R_\eta, \quad \text{etc.}$$

Using this notation, the right hand side of equation (A.2-1) can be rewritten as

$$\xi_x R_\xi + \eta_x R_\eta + \xi_y S_\xi + \eta_y S_\eta \qquad (A.2-2)$$

The additional terms ξ_x, ξ_y, η_x , η_y, are the metric terms resulting from the coordinate transformation and can be evaluated by solving the linear problem

$$\left[\frac{\partial(x,y)}{\partial(\xi,\eta)}\right]\left[\frac{\partial}{\partial x},\frac{\partial}{\partial y}\right] = \left[\frac{\partial}{\partial \xi},\frac{\partial}{\partial \eta}\right]$$

The results are

$$\frac{\partial}{\partial x} = J\left[\frac{\partial}{\partial \xi}(y_\eta) - \frac{\partial}{\partial \eta}(y_\xi)\right]$$

$$\frac{\partial}{\partial y} = J\left[\frac{\partial}{\partial \xi}(-x_\eta) + \frac{\partial}{\partial \eta}(x_\xi)\right]$$

where the Jacobian of the transformation, J, is given as

$$J = \left|\frac{\partial(\xi,\eta)}{\partial(x,y)}\right| = [x_\xi y_\eta - x_\eta y_\xi]^{-1}$$

and taking derivatives of ξ and η we finally arrive at

$$\frac{\xi_x}{J} = y_\eta;\ \frac{\xi_y}{J} = -x_\eta;\ \frac{\eta_x}{J} = -y_\xi;\ \frac{\eta_y}{J} = x_\xi \qquad (A.2-3)$$

After dividing equation (A.2-2) by the Jacobian J, and collecting terms the result is

$$\frac{\partial}{\partial \xi}\left[\frac{\xi_x R + \xi_y S}{J}\right] + \frac{\partial}{\partial \eta}\left[\frac{\eta_x R + \eta_y S}{J}\right] - R\left[(\frac{\xi_x}{J})_\xi + (\frac{\eta_x}{J})_\eta\right] - S\left[(\frac{\xi_y}{J})_\xi + (\frac{\eta_y}{J})_\eta\right]$$

By taking derivatives of equation (A.2-3), the third and fourth terms in brackets vanish and the shear stress and heat flux terms can be shown to be

$$\frac{\partial}{\partial \xi}\left[\left(\frac{\xi_x}{J}\right)R + \left(\frac{\xi_y}{J}\right)S\right] + \frac{\partial}{\partial \eta}\left[\left(\frac{\eta_x}{J}\right)R + \left(\frac{\eta_y}{J}\right)S\right]$$

and will be redefined as

$$\frac{\partial R'}{\partial \xi} + \frac{\partial S'}{\partial \eta} \qquad (A.2-4)$$

where $R' = \frac{\xi_x}{J}R + \frac{\xi_y}{J}S$

and $S' = \frac{\eta_x}{J}R + \frac{\eta_y}{J}S$

It is interesting to note that the result of the transformation to computational space yields two newly defined vectors R′ , S′ which differ only by the metrics which multiply R and S.

A.3 Shear Stress and Heat Flux Terms in Generalized Coordinates for the Navier Stokes Equations

The transforming of the shear stress and heat flux terms that appear in the full Navier Stokes equations from Cartesian coordinates (x,y) to generalized coordinates (ξ,η) is accomplished via the chain rule of differentiation. The results of the transformation to R and S are shown in equation (A.2-4). It is still remaining to transform the four components of R and S to computational space and finally determine R′ and S′. It can be shown that R transformed to (ξ,η) space yields

$$r_1 = 0 \qquad (A.3-1)$$

$$r_2 = \mu\left[\frac{4}{3}(\xi_x u_\xi + \eta_x u_\eta) - \frac{2}{3}(\xi_y v_\xi + \eta_y v_\eta)\right]$$

$$r_3 = \mu[\xi_y u_\xi + \eta_y u_\eta + \xi_x v_\xi + \eta_x v_\eta]$$

$$\begin{aligned} r_4 = \mu\Big[&\frac{2}{3}(\xi_x(u^2)_\xi + \eta_x(u^2)_\eta) - \frac{2}{3}(\xi_y uv_\xi + \eta_y uv_\eta) \\ &+ (\xi_y vu_\xi + \eta_y vu_\eta + \frac{\xi_x}{2}(v^2)_\xi + \frac{\eta_x}{2}(v^2)_\eta) \\ &+ (Pr)^{-1}(\gamma-1)^{-1}(\xi_x(a^2)_\xi + \eta_x(a^2)_\eta)\Big] \end{aligned}$$

and S transformed to (ξ,η) space yields

$$s_1 = 0 \qquad (A.3-2)$$

$$s_2 = \mu[\xi_y u_\xi + \eta_y u_\eta + \xi_x v_\xi + \eta_x v_\eta]$$

$$s_3 = \mu\left[\frac{4}{3}(\xi_y v_\xi + \eta_y v_\eta) - \frac{2}{3}(\xi_x u_\xi + \eta_x u_\eta)\right]$$

$$\begin{aligned} s_4 = \mu\Big[&\frac{2}{3}(\xi_y(v^2)_\xi + \eta_y(v^2)_\eta) - \frac{2}{3}(\xi_x vu_\xi + \eta_x vu_\eta) \\ &+ (\xi_x uv_\xi + \eta_x uv_\eta + \frac{\xi_y}{2}(u^2)_\xi + \frac{\eta_y}{2}(u^2)_\eta) \\ &+ (Pr)^{-1}(\gamma-1)^{-1}(\xi_y(a^2)_\xi + \eta_y(a^2)_\eta)\Big] \end{aligned}$$

Returning to equation (A.2-4), R′ and S′ can now be defined.

$$R' = \begin{bmatrix} r'_1 \\ r'_2 \\ r'_3 \\ r'_4 \end{bmatrix} \qquad (A.3-3)$$

$$r'_1 = 0$$

$$r'_2 = \mu\left[\left(\frac{4}{3}\frac{\eta_x\xi_x}{J} + \frac{\eta_y\xi_y}{J}\right)u_\eta + \left(\frac{\eta_x\xi_y}{J} - \frac{2}{3}\frac{\eta_y\xi_x}{J}\right)v_\eta\right] + \mu\left[\left(\frac{4}{3}\frac{\xi_x^2}{J} + \frac{\xi_y^2}{J}\right)u_\xi + \left(\frac{1}{3}\frac{\xi_x\xi_y}{J}\right)v_\xi\right]$$

$$r'_3 = \mu\left[\left(\frac{1}{3}\frac{\xi_x\xi_y}{J}\right)u_\xi + \left(\frac{\xi_x^2}{J} + \frac{4}{3}\frac{\xi_y^2}{J}\right)v_\xi\right] + \mu\left[\left(\frac{\eta_y\xi_x}{J} - \frac{2}{3}\frac{\eta_x\xi_y}{J}\right)u_\eta + \left(\frac{\xi_x\eta_x}{J} + \frac{4}{3}\frac{\eta_y\xi_y}{J}\right)v_\eta\right]$$

$$r'_4 = \mu\left[\frac{1}{2}\left[\left(\frac{4}{3}\frac{\xi_x^2}{J} + \frac{\xi_y^2}{J}\right)(u^2)_\xi + \left(\frac{4}{3}\frac{\xi_y^2}{J} + \frac{\xi_x^2}{J}\right)(v^2)_\xi\right] + (Pr)^{-1}(\gamma-1)^{-1}\gamma\left(\frac{\xi_y^2}{J} + \frac{\xi_x^2}{J}\right)\left(\frac{P}{\rho}\right)_\xi + \left(\frac{1}{3}\frac{\xi_x\xi_y}{J}\right)(uv)_\xi\right]$$
$$+ \mu\left[\frac{1}{2}\left[\left(\frac{4}{3}\frac{\eta_x\xi_x}{J} + \frac{\eta_y\xi_y}{J}\right)(u^2)_\eta + \left(\frac{4}{3}\frac{\eta_y\xi_y}{J} + \frac{\eta_x\xi_x}{J}\right)(v^2)_\eta\right]\right]$$
$$+ \mu\left[\gamma(\gamma-1)^{-1}(Pr)^{-1}\left(\frac{\eta_x\xi_x}{J} + \frac{\eta_y\xi_y}{J}\right)\left(\frac{P}{\rho}\right)_\eta + \left(\frac{\eta_y\xi_x}{J} - \frac{2}{3}\frac{\eta_x\xi_y}{J}\right)(uv)_\eta + \frac{5}{3}\left(\frac{\eta_x\xi_y}{J} - \frac{\eta_y\xi_x}{J}\right)uv_\eta\right]$$

$$S' = \begin{bmatrix} s'_1 \\ s'_2 \\ s'_3 \\ s'_4 \end{bmatrix} \qquad (A.3-4)$$

$$s'_1 = 0$$

$$s'_2 = \mu\left[\left(\frac{4}{3}\frac{\eta_x\xi_x}{J} + \frac{\eta_y\xi_y}{J}\right)u_\xi + \left(\frac{\eta_y\xi_x}{J} - \frac{2}{3}\frac{\eta_x\xi_y}{J}\right)v_\xi\right] + \mu\left[\left(\frac{4}{3}\frac{\eta_x^2}{J} + \frac{\eta_y^2}{J}\right)u_\eta + \left(\frac{1}{3}\frac{\eta_x\eta_y}{J}\right)v_\eta\right]$$

$$s'_3 = \mu\left[\left(\frac{1}{3}\frac{\eta_x\eta_y}{J}\right)u_\eta + \left(\frac{\eta_x^2}{J} + \frac{4}{3}\frac{\eta_y^2}{J}\right)v_\eta\right] + \mu\left[\left(\frac{\eta_x\xi_y}{J} - \frac{2}{3}\frac{\eta_y\xi_x}{J}\right)u_\xi + \left(\frac{\xi_x\eta_x}{J} + \frac{4}{3}\frac{\eta_y\xi_y}{J}\right)v_\xi\right]$$

$$s'_4 = \mu\left[\frac{1}{2}\left[\left(\frac{4}{3}\frac{\eta_x^2}{J} + \frac{\eta_y^2}{J}\right)u_\eta^2 + \left(\frac{4}{3}\frac{\eta_y^2}{J} + \frac{\eta_x^2}{J}\right)(v^2)_\eta\right] + (Pr)^{-1}(\gamma-1)^{-1}\gamma\left(\frac{\eta_y^2}{J} + \frac{\eta_x^2}{J}\right)\left(\frac{P}{\rho}\right)_\eta + \left(\frac{1}{3}\frac{\eta_x\eta_y}{J}\right)(uv)_\eta\right]$$
$$+ \mu\left[\frac{1}{2}\left[\left(\frac{4}{3}\frac{\eta_x\xi_x}{J} + \frac{\eta_y\xi_y}{J}\right)(u^2)_\xi + \left(\frac{4}{3}\frac{\eta_y\xi_y}{J} + \frac{\eta_x\xi_x}{J}\right)(v^2)_\xi\right]\right]$$
$$+ \mu\left[\gamma(\gamma-1)^{-1}(Pr)^{-1}\left(\frac{\eta_x\xi_x}{J} + \frac{\eta_y\xi_y}{J}\right)\left(\frac{P}{\rho}\right)_\xi + \left(\frac{\eta_x\xi_y}{J} - \frac{2}{3}\frac{\eta_y\xi_x}{J}\right)(uv)_\xi + \frac{5}{3}\left(\frac{\eta_y\xi_x}{J} - \frac{\eta_x\xi_y}{J}\right)uv_\xi\right]$$

A.4 Shear Stress and Heat Flux Terms in Generalized Coordinates for Thin Layer Navier Stokes Equations

The thin layer approximation neglects all viscous terms associated with derivatives tangent to the body surface. Therefore, the thin layer equations are the same as the equations for the full Navier Stokes excluding all derivatives with respect to ξ. Equation (A.2-4) is reduced to

$$\frac{\partial S'}{\partial \eta}; \quad S' = \frac{\eta_x}{J} R + \frac{\eta_y}{J} S$$

where S′ contains derivatives only in the η direction, specifically (let T′ represent the modified S′ for thin layer equations)

$$T' = \begin{bmatrix} t'_1 \\ t'_2 \\ t'_3 \\ t'_4 \end{bmatrix} \qquad (A.4-1)$$

$$t'_1 = 0$$

$$t'_2 = \mu\left[\left(\frac{4}{3}\frac{\eta_x^2}{J} + \frac{\eta_y^2}{J}\right)u_\eta + \left(\frac{1}{3}\frac{\eta_y\eta_x}{J}\right)v_\eta\right]$$

$$t'_3 = \mu\left[\left(\frac{1}{3}\frac{\eta_x\eta_y}{J}\right)u_\eta + \left(\frac{\eta_x^2}{J} + \frac{4}{3}\frac{\eta_y^2}{J}\right)v_\eta\right]$$

$$t'_4 = \mu\left[\frac{1}{2}\left[\left(\frac{4}{3}\frac{\eta_x^2}{J} + \frac{\eta_y^2}{J}\right)u_\eta^2 + \left(\frac{4}{3}\frac{\eta_y^2}{J} + \frac{\eta_x^2}{J}\right)v_\eta^2\right]\right.$$

$$\left. + \gamma(Pr)^{-1}(\gamma - 1)^{-1}\left(\frac{\eta_y^2}{J} + \frac{\eta_x^2}{J}\right)\left(\frac{P}{\rho}\right)_\eta + \left(\frac{1}{3}\frac{\eta_x\eta_y}{J}\right)(uv)_\eta\right]$$

Appendix B. Computation and Validation of Full Navier Stokes Solver on a Double Throat Nozzle

B.1 Introduction

The double throat nozzle in Figure 17 is an internal flow problem which demonstrates strong viscous interaction phenomena that requires solving the full compressible Navier Stokes equations to resolve the complex flow physics. The nozzle design introduces a wide variety of flow characteristics which complicate the calculation ; however, by comparing the computational results to reference solutions [18], validation of the method will be achieved. The flow starting out subsonic at the inflow will reach supersonic conditions at the first throat (neglecting the subsonic boundary layer region). The flow then travels through a diverging converging section where it experiences compression waves, expansion waves, and separation phenomena. After passing through the second throat, the diverging channel results in a final expansion.

B.2 Computational Mesh

The computational grid used to solve the double throat nozzle was constructed of 171 equally spaced points in the streamwise direction and 38 points in the transverse direction. Algebraic stretching along the solid wall was implemented to resolve the highly complex flow in the viscous boundary layers. The bottom surface is the line of symmetry and thus no stretching is necessary. The symmetrical nozzle is shown in Figure 17 and is composed of five polynomial arcs. All regions of the nozzle exhibit continuity of slope; however, continuity of curvature is not satisfied between interfaces of Regions I and II and Regions IV and V. Comparisons of solutions with varying grid densities is provided in Reference [18] and Reference [19].

B.3 Boundary Conditions

B.3.1 Entrance Condition

The incoming flow at the entrance of the nozzle is created from an infinite reservoir of fluid and the viscous boundary layer thickness at the entrance is taken to be zero. Under these conditions the flow is assumed to be inviscid and isentropic which can correctly be modeled by setting the total enthalpy and the entropy equal to their corresponding reservoir conditions. The flow is also assumed to be parallel ($v = 0$). Consistent with subsonic inflow conditions, it will be necessary to extrapolate one property from the interior domain. In this calculation the u velocity was chosen.

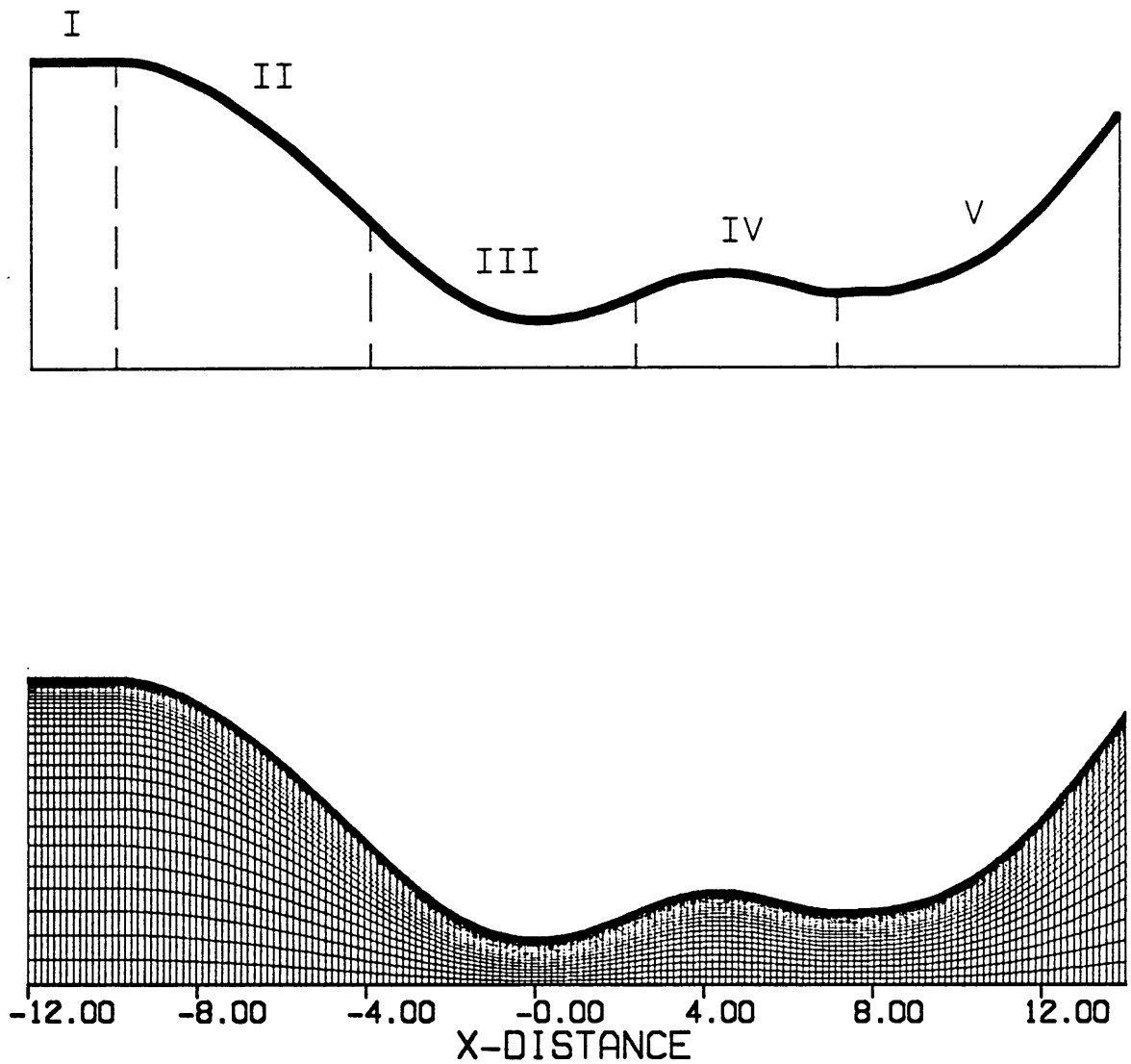


Figure 17. Plane Symmetrical Nozzle and Computational Grid

B.3.2 Wall Condition

The wall boundary conditions to be imposed are no-slip with the temperature set equal to the reservoir stagnation temperature. Extrapolating the pressure from the interior is the remaining condition to be enforced to complete the wall boundary treatment.

B.3.3 Center Line Condition

Along planes of symmetry, such as the center line in the double throat nozzle, a condition of tangency is enforced which zeros the gradients in the normal direction. Specifically gradients of pressure, density, total enthalpy and the velocity normal to the symmetry line were set to zero.

B.3.4 Outflow Condition

The inviscid flow at the exit plane is entirely supersonic, therefore all state properties will be extrapolated. It should be noted, however, that the viscous subsonic regions near the wall could be more accurately modeled by forcing a subsonic outflow condition over that portion of the outflow region where it is applicable. The results of extrapolating all state quantities over the entire downstream boundary is consistent with the reference solution cited earlier and are used in this calculation.

B.4 Computational Results

A full Navier Stokes solution was obtained for the double throat nozzle using a Reynolds number of 100 defined as:

$$Re = \frac{\rho_0 a_0 L}{\mu}.$$

where

ρ_0 is the stagnation density in the reservoir,

a_0 is the speed of sound using the stagnation temperature in the reservoir,

L is the reference length which corresponds to the half height of the first throat,

μ is the molecular viscosity calculated using Sutherland's Law.

A Mach number of 0.1 and a free stream static temperature at standard atmospheric conditions were used to complete the description of the problem.

The implicit Approximation Factorization (AF) algorithm was used to integrate the equations in time to the steady state solution. The calculation was performed using Van Leer's flux vector splitting in generalized coordinates developed in [9] and [10]. A local Courant number of 10 was used along with a third order upwind biased interpolation. Because the interpolation is not fully upwind, limiting will be necessary to avoid oscillations at discontinuities within the solution. Van Albeda's limiting will be used in this calculation.

The results of this test case are provided in Figures 18 through 21. As can be seen, the results compare very well to the reference solutions, providing acceptable assurance that the Navier Stokes solver is performing favorably. Slight discrepancies exist in the properties along the center line due possibly to not enough points near

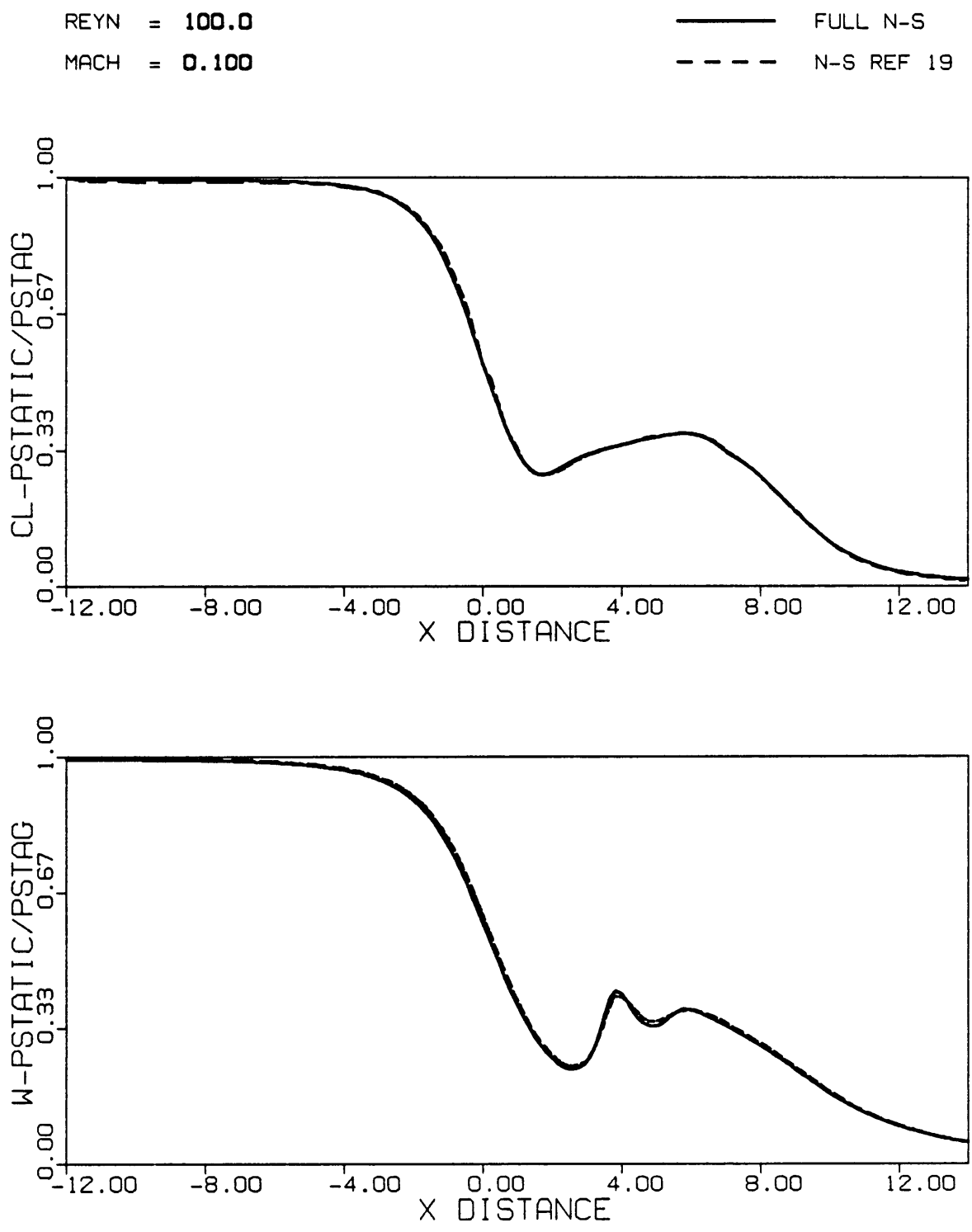


Figure 18. Wall and Centerline Properties ; Re = 100 ; Grid 171 x 38

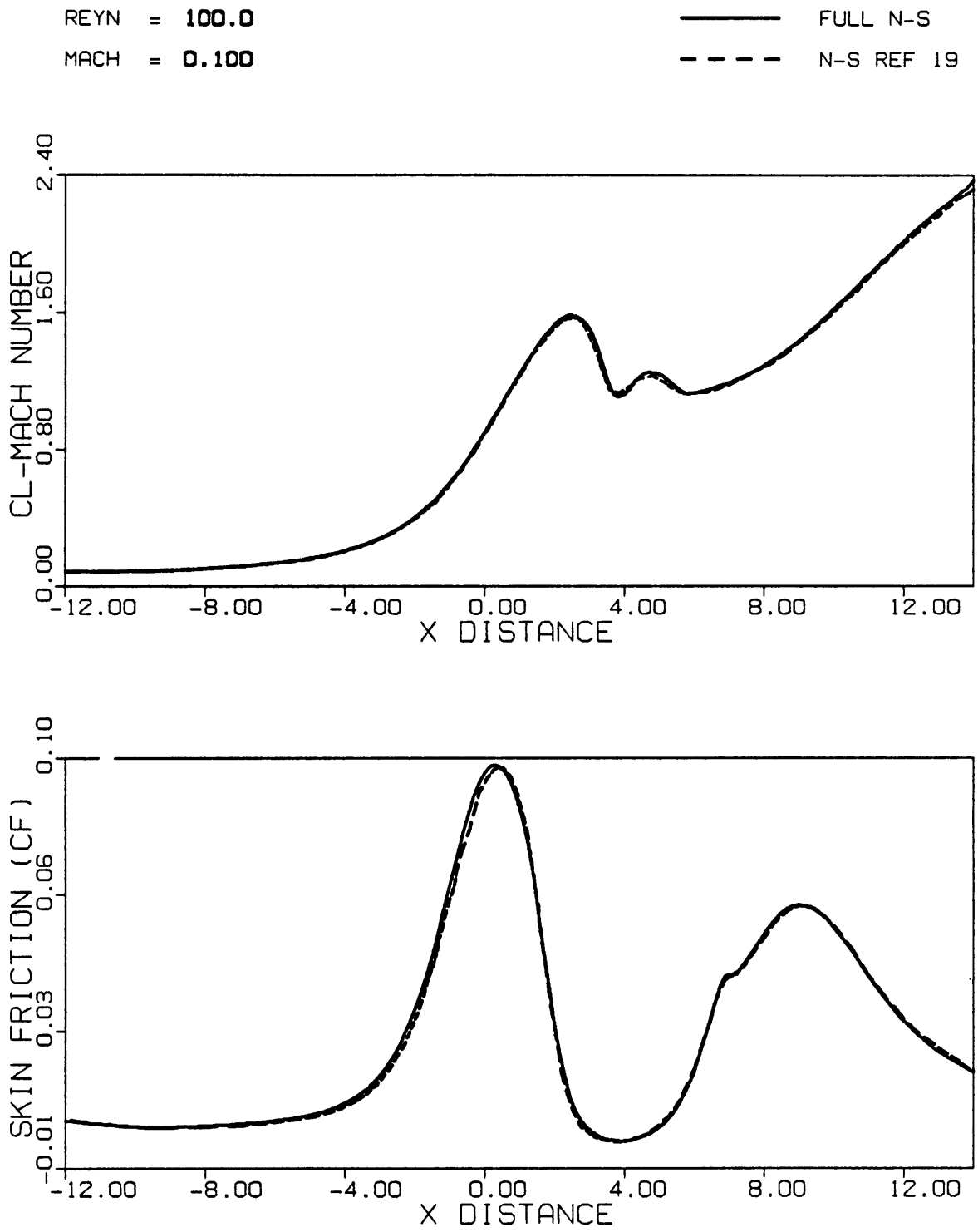


Figure 19. Wall and Centerline Properties ; Re = 100 ; Grid 171 x 38

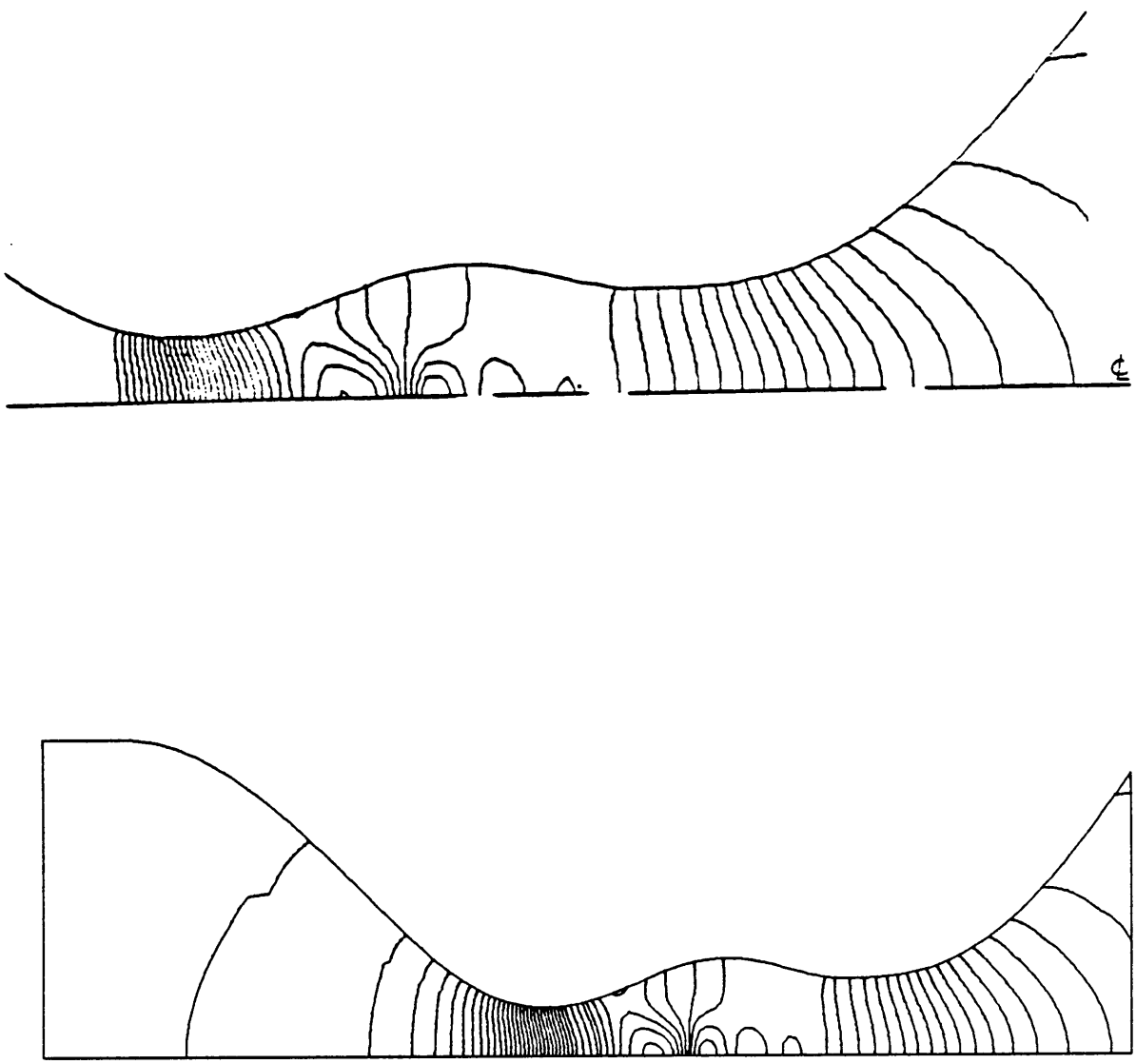


Figure 20. Flowfield Pressure Contours ; Top: Ref 19 ; Grid 171 x 38

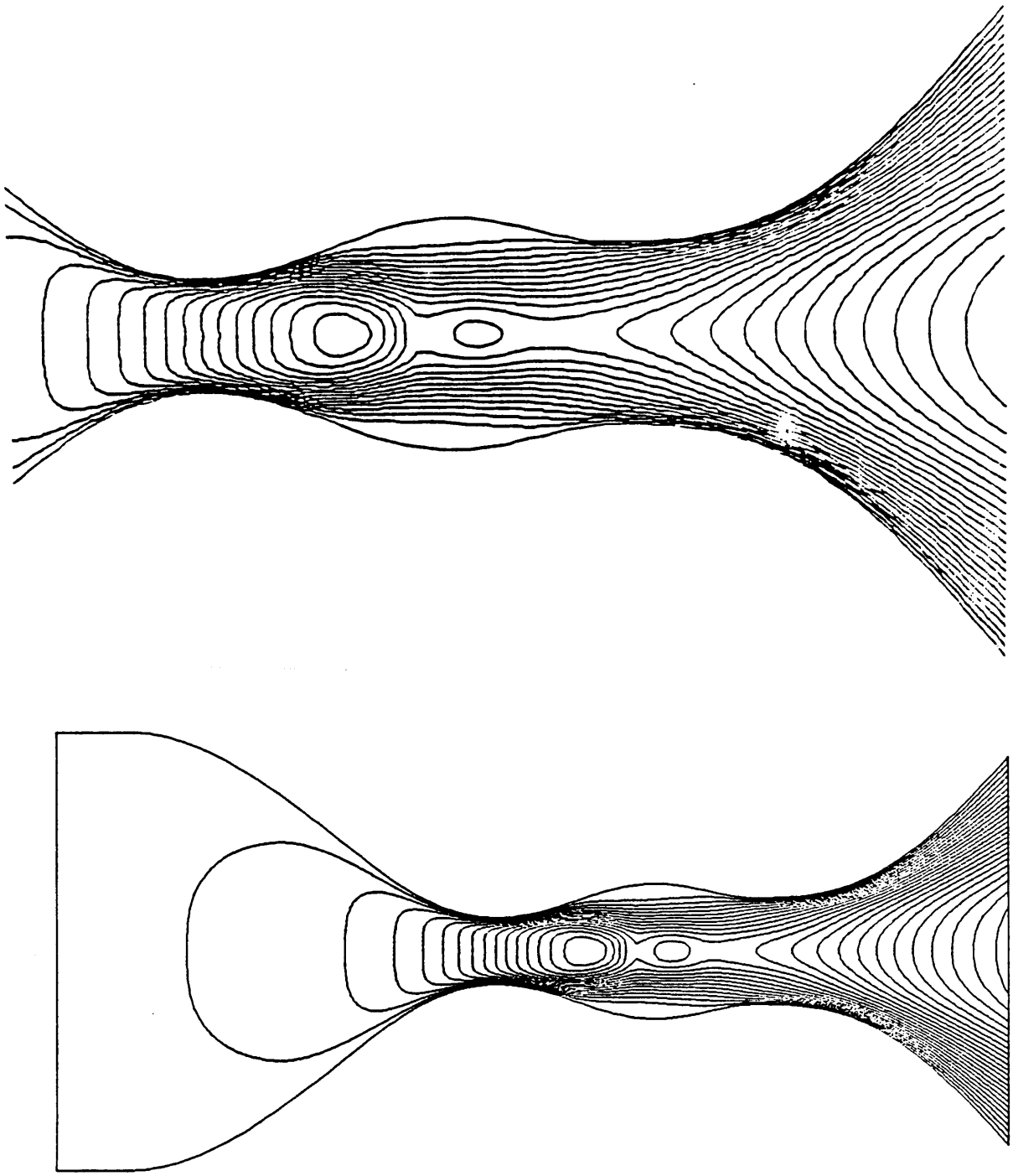


Figure 21. Flowfield Mach Contours ; Top: Ref 19 ; Grid 171 x 38

the surface of symmetry. A comparison of the separation region with Reference [19] is provided in Table 2 and agrees excellently with a maximum variation of 0.016 occurring at the location of separation.

A comparison of the complete Navier Stokes and thin-layer Navier Stokes is provided in Figures 22 and 23. Slight differences between the solutions can be seen in the skin friction coefficient and center line pressure ratio.

B.5 Code Performance and Convergence

The preceding results were computed on an IBM 3090 vector processing computer at Virginia Polytechnic Institute and State University in Blacksburg, Virginia. Because the code is highly vectorizable, it is estimated that the savings in computational time is on the order of seven times as compared to scalar operations. In addition, by reusing the LU decomposition every tenth iteration results in an increase in computational rate by a factor of seven. Table 3 provides the details of the computing resources.

The convergence history to the steady state solution is provided in Figure 24. The L_2 norm of the residual was reduced approximately twelve orders of magnitude and decreased linearly after 600 iterations.

B.6 Conclusions

The purpose of analyzing the flow in the double throat nozzle was to provide sufficient proof that the full Navier Stokes solver is correctly analyzing the physics.

Table 2. Axial Region of Separation

Solution	Region of Separation	Grid
VERSAFLOW	$2.988 \leq x \leq 4.823$	171 X 38
Ref. [19]	$3.004 \leq x \leq 4.820$	131 X 81

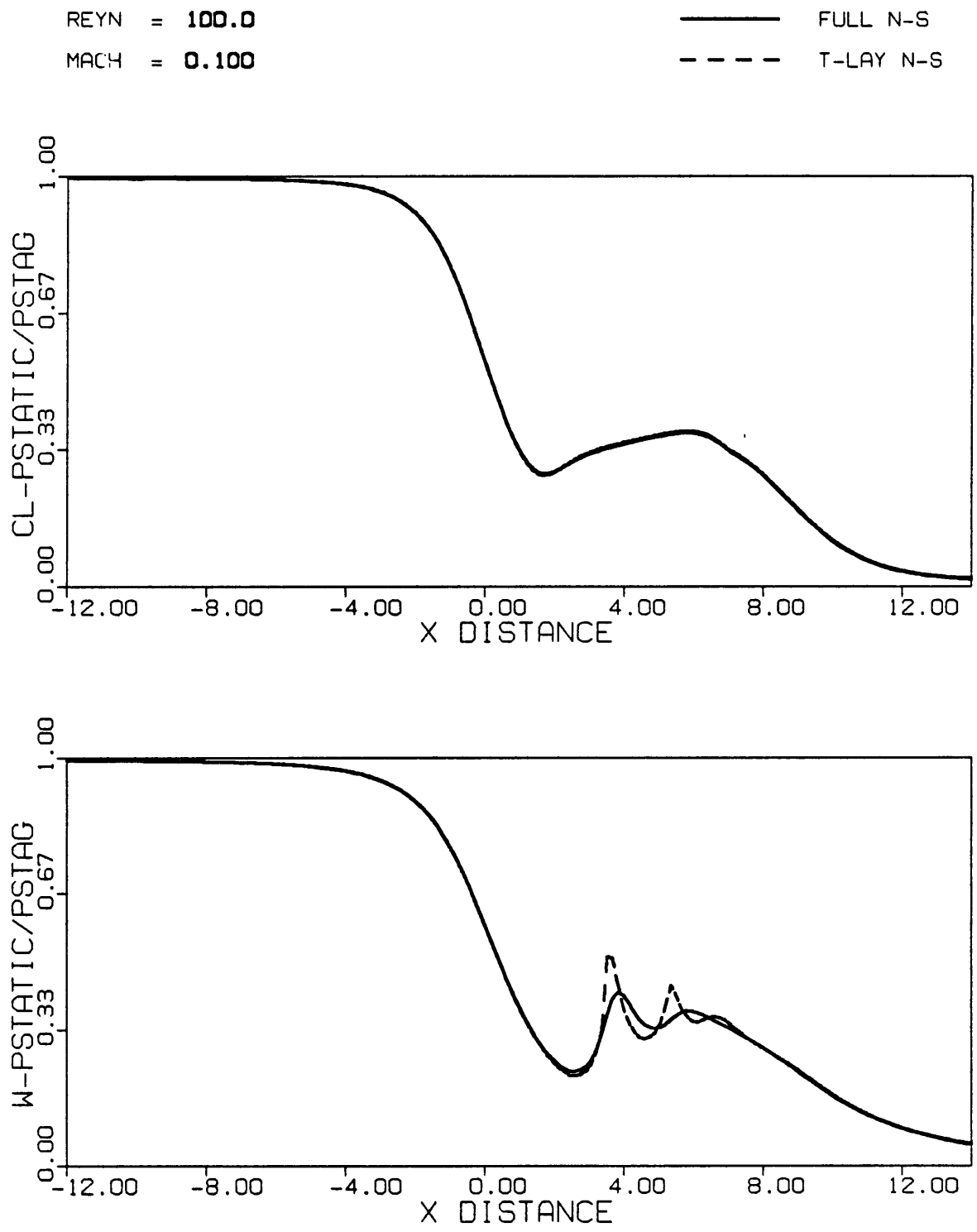


Figure 22. Wall and Centerline Properties ; Re = 100 ; Grid 171 x 38

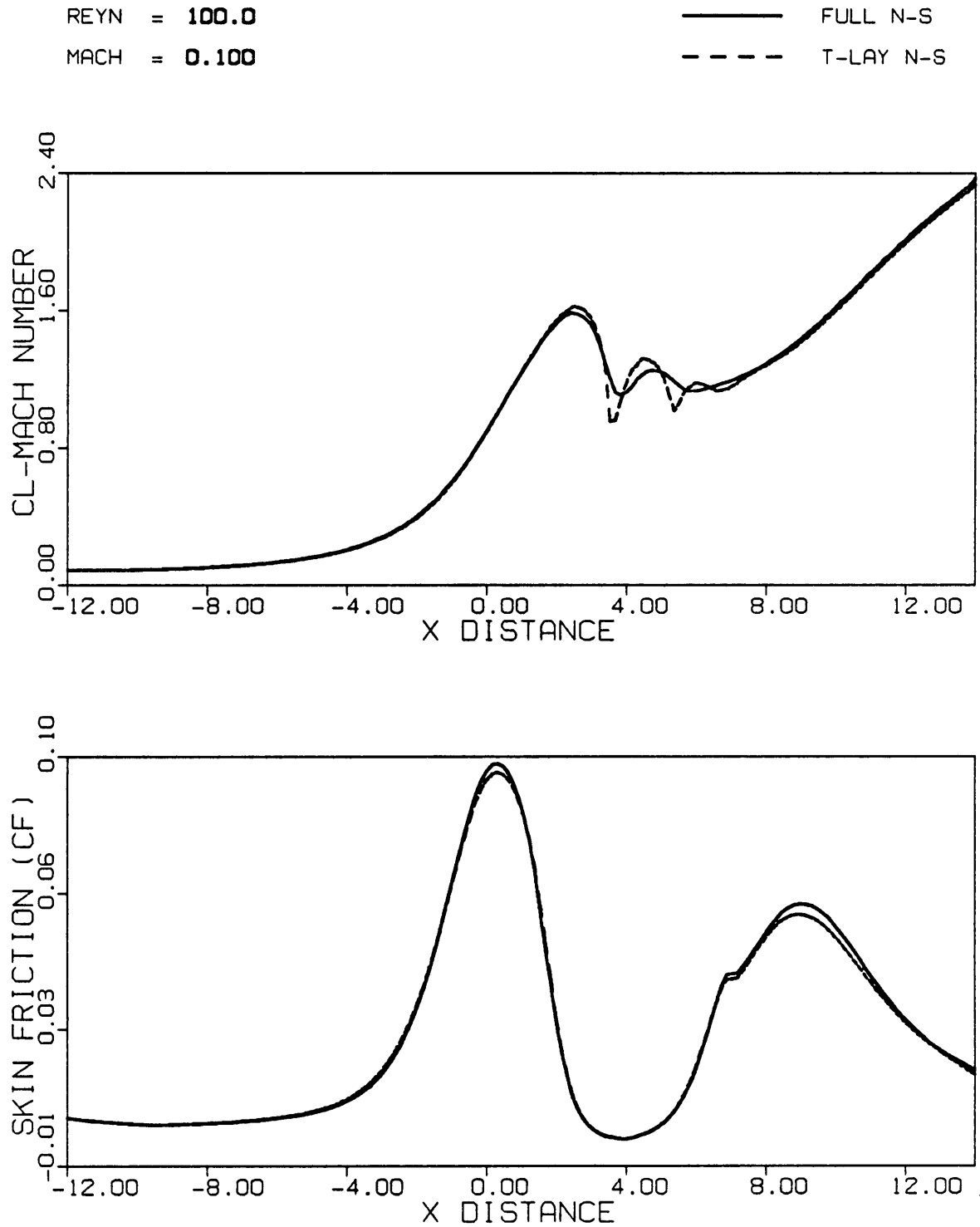


Figure 23. Wall and Centerline Properties ; Re = 100 ; Grid 171 x 38

Table 3. Summary of Computing Resources

LU Decomposition	CPU seconds per iteration	CPU seconds per iteration per mesh point
Reuse	1.117	.1719 E -03
No Reuse	6.971	.1073 E -02
Total Iterations: 2000		Total Time: 2234 sec.

Thorough and accurate comparisons with reference solutions support the conclusion that the method is consistent and can now be used with confidence on other problems. Low Reynolds number problems with strong viscous interactions, along with other classes of problems, will require the full NS equations to resolve the flow field accurately. Comparisons of NS results on the double throat nozzle for Reynolds numbers of 100, 400, and 1600 are provided in Reference [19].

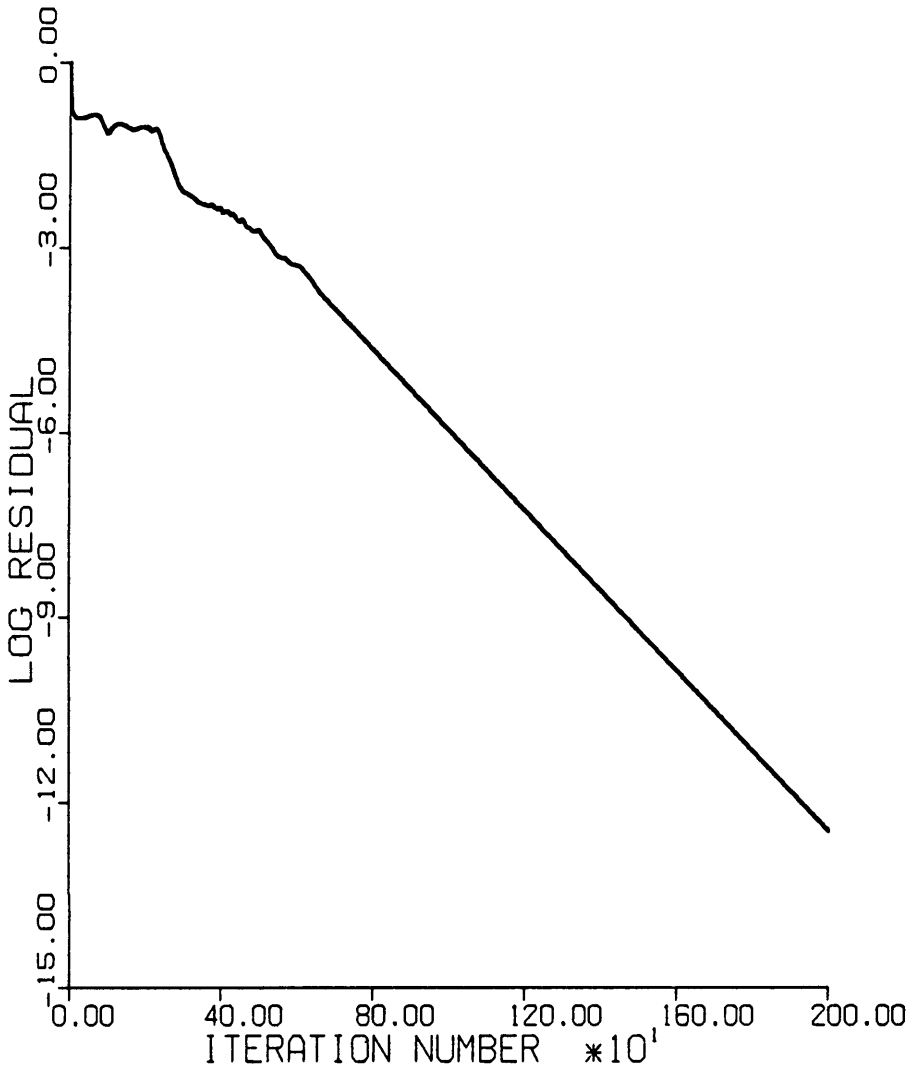


Figure 24. Convergence History for Nozzle ; Grid 171 x 38

```
C
C   INPUT FILE FOR  DOUBLE THROAT NOZZLE
C
   XMACH       RE      TREF      XREF
    0.1  9.94E-06    293.16       1.0
   GAMMA        PR
    1.40       .72
   ISTRT     NITER        DT     DTMAX      ISER
      0       500    -10.00     1.E12         0
   IRELU     NRELU     NSTLU
      0        10         2
    IMPL      IAF      IROE     IVISC      ICNS      NSTG
       1        1         0         1         1         4
    LIMJ      LIMK     XKAPJ     XKAPK
       0        0     .3333     .3333
    LHSJ      LHSK     XLHSJ     XLHSK
       0        0    1.0000    1.0000
  IMARCH     TOLER    IGUESS       SIG     JSTRT      JEND    IFLARE
       0  1.0E-02         1       .95        -1       171         0
ISWEEP(1) ISWEEP(2) ISWEEP(3) ISWEEP(4)
       1        2         1         2
B.C. INPUT     TYPE       BC1       BC2       BC3       BC4      IBND
J=0 BOUNDARY
       1        1
      38        4       1.0       1.0       0.0       1.0         0
J=JDIM BOUNDARY
       1        1
      38        6       0.5       0.0       0.0       0.0         0
K=0 BOUNDARY
       1        1
     171        2      1.00       1.0       0.0       1.0         0
K=KDIM BOUNDARY
       1        1
     171        7     1.002       0.0       0.0       0.0         0
TURBULENCE  MODELING
   ITURB    NSTRIP DELTAB(1) DELTAB(2)
       0        1      0.07      0.07
NPTSTP(I) ISTRTP(I)
     170        4
```

Appendix C. Program Listing

```
C
C    THIS SUBROUTINE NUMERICALLY EVALUATES THE VISCOUS TERMS THAT
C    APPEAR IN THE NAVIER STOKES EQUATIONS.  THE IMPLICIT CONTRIBUTIONS
C    APPEAR FROM THE DOUBLE DERIVATIVES IN BOTH DIRECTIONS.  THE CROSS
C    DERIVATIVES ARE NOT LINEARIZED AND ARE TREATED SYMMETRICALLY
C    AND THUS APPEAR ONLY ON THE RIGHT HAND SIDE.
C
C
      SUBROUTINE VISCUS(JDIM,KDIM,JKDIM,Q,QJ0,QK0,SJ,SK,AJ,BJ,CJ,
     .  AK,BK,CK,VM1,VM2,RES,VOL,T,TT,VISC,VISCJ0,VISCK0,CRJ,CRK)
C
      COMMON/BASE/TITLE(20),XMACH,RE,DT,DTMAX,TREF,PR,XRE,XMUREF
      COMMON/SWITCH/IVISC,IMARCH,IROE,IAF,IMPL,NITER,NSTLU,NSTG,
     .          ISTRT,ISER,NRELU,IRELU,IJAC,ISWP,ICNS,ITURB
      COMMON/FLUID/GAMMA,GAMI
      DIMENSION Q(JDIM,KDIM,4),QJ0(KDIM,4),QK0(JDIM,4)
      DIMENSION T(JKDIM+1,13),TT(JDIM,KDIM,18)
      DIMENSION VOL(JDIM,KDIM)
      DIMENSION AJ(JDIM-1,KDIM+1,4,4)
      DIMENSION BJ(JDIM-1,KDIM+1,4,4)
      DIMENSION CJ(JDIM-1,KDIM+1,4,4)
      DIMENSION AK(KDIM-1,JDIM+1,4,4)
      DIMENSION BK(KDIM-1,JDIM+1,4,4)
      DIMENSION CK(KDIM-1,JDIM+1,4,4)
      DIMENSION VM1(JKDIM,4,4)
      DIMENSION VM2(JKDIM,4,4)
      DIMENSION SJ(JDIM,KDIM,3)
      DIMENSION SK(JDIM,KDIM,3)
      DIMENSION CRJ(JDIM,KDIM,2)
      DIMENSION CRK(JDIM,KDIM,2)
      DIMENSION RES(JDIM,KDIM,4)
      DIMENSION VISC(JDIM,KDIM),VISCJ0(KDIM),VISCK0(JDIM)
C
C  CONSTANTS
C
      JDIMP=JDIM+1
      JDIM1=JDIM-1
      JDIM2=JDIM-2
      KDIMP=KDIM+1
      KDIM1=KDIM-1
      KDIM2=KDIM-2
C
      REI=1.E0/RE
      G1=-8./3.
      G2=3.
      G3=1.
      G4=-4./3.
      G5=1./3.
C
      T0=GAMMA/PR
```

```
      T00=T0/GAMI
C
      DO 2 K=2,KDIM1
      DO 2 J=1,JDIM1
      TT(J,K,1)=SK(J,K,3)*(VISC(J,K)+VISC(J,K-1))/(VOL(J,K)+VOL(J,K-1))
    2 CONTINUE
C
      DO 4 J=1,JDIM1
      TT(J,1,1)=SK(J,1,3)*VISCK0(J)/VOL(J,1)
    4 TT(J,KDIM,1)=SK(J,KDIM,3)*VISC(J,KDIM)/VOL(J,KDIM1)
C
C           THIN-LAYER TERMS
C
      DO 200 J=1,JDIM1
C
C
      DO 10 K=1,KDIM
      T(K,1)=1.+SK(J,K,1)*SK(J,K,1)*G5
      T(K,2)=SK(J,K,1)*SK(J,K,2)*G5
      T(K,3)=1.+SK(J,K,2)*SK(J,K,2)*G5
   10 CONTINUE
C
      T(1,4)=1.E0/QK0(J,1)
      T(1,5)=QK0(J,2)
      T(1,6)=QK0(J,3)
      T(1,10)=QK0(J,4)
C
      DO 20 K=1,KDIM
      T(K+1,4)=1./Q(J,K,1)
      T(K+1,5)=Q(J,K,2)
      T(K+1,6)=Q(J,K,3)
      T(K+1,10)=Q(J,K,4)
   20 CONTINUE
C
      DO 30 K=2,KDIM1
      T(K,11)=T(K,1)*(T(K+1,5)-T(K,5))+T(K,2)*(T(K+1,6)-T(K,6))
   30 CONTINUE
C
      DO 32 K=2,KDIM1
      T(K,12)=T(K,3)*(T(K+1,6)-T(K,6))+T(K,2)*(T(K+1,5)-T(K,5))
   32 CONTINUE
C
      DO 34 K=2,KDIM1
      T(K,13)=.5*(T(K,1)*(T(K+1,5)*T(K+1,5)-T(K,5)*T(K,5))
     .   +T(K,3)*(T(K+1,6)*T(K+1,6)-T(K,6)*T(K,6)))
     .   +T(K,2)*(T(K+1,5)*T(K+1,6)-T(K,5)*T(K,6))
     .   +T00*(T(K+1,4)*T(K+1,10)-T(K,4)*T(K,10))
   34 CONTINUE
C
C LOWER BOUNDARY
C
      T(1,11)=T(1,1)*(-8.*T(1,5)+9.*T(2,5)-T(3,5))*G5
     .+T(1,2)*(-8.*T(1,6)+9.*T(2,6)-T(3,6))*G5
      T(1,12)=T(1,3)*(-8.*T(1,6)+9.*T(2,6)-T(3,6))*G5
     .+T(1,2)*(-8.*T(1,5)+9.*T(2,5)-T(3,5))*G5
      T(1,13)=.5*(T(1,1)*(9.*T(2,5)*T(2,5)-8.*T(1,5)*T(1,5)-
     .T(3,5)*T(3,5))*G5+T(1,3)*(9.*T(2,6)*T(2,6)-8.*T(1,6)*T(1,6)-
     .T(3,6)*T(3,6))*G5)+T(1,2)*(9.*T(2,6)*T(2,5)-8.*T(1,6)*T(1,5)-
     .T(3,6)*T(3,5))*G5+T00*(9.*T(2,4)*T(2,10)-8.*T(1,4)*T(1,10)-
     .T(3,4)*T(3,10))*G5
C
C TOP BOUNDARY
C
      T(KDIM,11)=T(KDIM,1)*(8.*T(KDIMP,5)-9.*T(KDIM,5)+T(KDIM1,5))*G5
     .+T(KDIM,2)*(8.*T(KDIMP,6)-9.*T(KDIM,6)+T(KDIM1,6))*G5
      T(KDIM,12)=T(KDIM,3)*(8.*T(KDIMP,6)-9.*T(KDIM,6)+T(KDIM1,6))*G5
     .+T(KDIM,2)*(8.*T(KDIMP,5)-9.*T(KDIM,5)+T(KDIM1,5))*G5
      T(KDIM,13)=.5*(T(KDIM,1)*
     .(-9.*T(KDIM,5)*T(KDIM,5)+8.*T(KDIMP,5)*T(KDIMP,5)
     .+T(KDIM1,5)*T(KDIM1,5))*G5+T(KDIM,3)*
     .(-9.*T(KDIM,6)*T(KDIM,6)+8.*T(KDIMP,6)*T(KDIMP,6)
     .+T(KDIM1,6)*T(KDIM1,6))*G5)+T(KDIM,2)*
```

```
     .(-9.*T(KDIM,6)*T(KDIM,5)+8.*T(KDIMP,6)*T(KDIMP,5)+
     .T(KDIM1,6)*T(KDIM1,5))*G5+T00*
     .(-9.*T(KDIM,4)*T(KDIM,10)+8.*T(KDIMP,4)*T(KDIMP,10)+
     .T(KDIM1,4)*T(KDIM1,10))*G5
C
      DO 80 K=1,KDIM
   80 T(K,9)=SK(J,K,3)*TT(J,K,1)
      DO 90 N=11,13
      DO 90 K=1,KDIM
   90 T(K,N)=T(K,N)*T(K,9)
C
      DO 100 N=2,4
      DO 100 K=1,KDIM1
      RES(J,K,N)=RES(J,K,N)-(T(K+1,N+9)-T(K,N+9))*REI
  100 CONTINUE
C
      IF (IJAC.NE.1) GO TO 200
C
C IMPLICIT TERMS
C
      DO 40 K=1,KDIM
      VM1(K,2,1)=(-T(K+1,4)*(T(K,1)*T(K+1,5)
     .                      +T(K,2)*T(K+1,6)))*T(K,9)
      VM1(K,2,2)=(T(K+1,4)*T(K,1))*T(K,9)
      VM1(K,2,3)=(T(K+1,4)*T(K,2))*T(K,9)
      VM1(K,3,1)=(-T(K+1,4)*(T(K,2)*T(K+1,5)
     .                      +T(K,3)*T(K+1,6)))*T(K,9)
      VM1(K,3,2)=(T(K+1,4)*T(K,2))*T(K,9)
      VM1(K,3,3)=(T(K+1,4)*T(K,3))*T(K,9)
      VM1(K,4,1)=(-T(K+1,4)*(T00*T(K+1,4)*T(K+1,10)
     .          +(T(K,1)-.5*T0)*T(K+1,5)*T(K+1,5)
     .          +2.*T(K,2)*T(K+1,5)*T(K+1,6)
     .          +(T(K,3)-.5*T0)*T(K+1,6)*T(K+1,6)))*T(K,9)
      VM1(K,4,2)=-VM1(K,2,1)-(T0*T(K+1,4)*T(K+1,5))*T(K,9)
      VM1(K,4,3)=-VM1(K,3,1)-(T0*T(K+1,4)*T(K+1,6))*T(K,9)
      VM1(K,4,4)=(T0*T(K+1,4))*T(K,9)
   40 CONTINUE
C
      DO 50 K=1,KDIM
      VM2(K,2,1)=(-T(K,4)*(T(K,1)*T(K,5)
     .                    +T(K,2)*T(K,6)))*T(K,9)
      VM2(K,2,2)=(T(K,4)*T(K,1))*T(K,9)
      VM2(K,2,3)=(T(K,4)*T(K,2))*T(K,9)
      VM2(K,3,1)=(-T(K,4)*(T(K,2)*T(K,5)
     .                    +T(K,3)*T(K,6)))*T(K,9)
      VM2(K,3,2)=(T(K,4)*T(K,2))*T(K,9)
      VM2(K,3,3)=(T(K,4)*T(K,3))*T(K,9)
      VM2(K,4,1)=(-T(K,4)*(T00*T(K,4)*T(K,10)
     .          +(T(K,1)-.5*T0)*T(K,5)*T(K,5)
     .          +2.*T(K,2)*T(K,5)*T(K,6)
     .          +(T(K,3)-.5*T0)*T(K,6)*T(K,6)))*T(K,9)
      VM2(K,4,2)=-VM2(K,2,1)-(T0*T(K,4)*T(K,5))*T(K,9)
      VM2(K,4,3)=-VM2(K,3,1)-(T0*T(K,4)*T(K,6))*T(K,9)
      VM2(K,4,4)=(T0*T(K,4))*T(K,9)
   50 CONTINUE
C
      DO 110 N=2,4
      DO 110 M=1,4
      DO 110 K=3,KDIM1
      AJ(J,K,N,M)=-VM2(K-1,N,M)*REI
      BJ(J,K,N,M)=(VM1(K-1,N,M)+VM2(K,N,M))*REI
      CJ(J,K,N,M)=-VM1(K,N,M)*REI
  110 CONTINUE
C
      DO 112 N=2,4
      DO 112 M=1,4
      AJ(J,2,N,M)=G1*VM2(1,N,M)*REI
      BJ(J,2,N,M)=(G2*VM1(1,N,M)+G3*VM2(2,N,M))*REI
      CJ(J,2,N,M)=G4*VM1(2,N,M)*REI
      AJ(J,KDIM,N,M)=G4*VM2(KDIM1,N,M)*REI
      BJ(J,KDIM,N,M)=(G3*VM1(KDIM1,N,M)+G2*VM2(KDIM,N,M))*REI
      CJ(J,KDIM,N,M)=G1*VM1(KDIM,N,M)*REI
```

```
 112  CONTINUE
C
 200  CONTINUE
C
      IF (ICNS.EQ.0) RETURN
C
C         COMPLETE NAVIER-STOKES VISCOUS TERMS
C
      DO 6 K=1,KDIM1
      DO 6 J=2,JDIM1
      TT(J,K,2)=SJ(J,K,3)*(VISC(J,K)+VISC(J-1,K))/(VOL(J,K)+VOL(J-1,K))
    6 CONTINUE
C
      DO 8 K=1,KDIM1
      TT(1,K,2)=SJ(1,K,3)*VISCJ0(K)/VOL(1,K)
    8 TT(JDIM,K,2)=SJ(JDIM,K,3)*VISC(JDIM,K)/VOL(JDIM1,K)
C
      DO 400 K=1,KDIM1
C
      DO 12 J=1,JDIM
      T(J,1)=1.+SJ(J,K,1)*SJ(J,K,1)*G5
      T(J,2)=SJ(J,K,1)*SJ(J,K,2)*G5
      T(J,3)=1.+SJ(J,K,2)*SJ(J,K,2)*G5
   12 CONTINUE
C
      T(1,4)=1.E0/QJ0(K,1)
      T(1,5)=QJ0(K,2)
      T(1,6)=QJ0(K,3)
      T(1,10)=QJ0(K,4)
C
      DO 22 J=1,JDIM
      T(J+1,4)=1./Q(J,K,1)
      T(J+1,5)=Q(J,K,2)
      T(J+1,6)=Q(J,K,3)
      T(J+1,10)=Q(J,K,4)
   22 CONTINUE
C
      DO 35 J=2,JDIM1
      T(J,11)=T(J,1)*(T(J+1,5)-T(J,5))+T(J,2)*(T(J+1,6)-T(J,6))
   35 CONTINUE
C
      DO 36 J=2,JDIM1
      T(J,12)=T(J,3)*(T(J+1,6)-T(J,6))+T(J,2)*(T(J+1,5)-T(J,5))
   36 CONTINUE
C
      DO 38 J=2,JDIM1
      T(J,13)=.5*(T(J,1)*(T(J+1,5)*T(J+1,5)-T(J,5)*T(J,5))
     .   +T(J,3)*(T(J+1,6)*T(J+1,6)-T(J,6)*T(J,6)))
     .   +T(J,2)*(T(J+1,5)*T(J+1,6)-T(J,5)*T(J,6))
     .   +T00*(T(J+1,4)*T(J+1,10)-T(J,4)*T(J,10))
   38 CONTINUE
C
C LEFT BOUNDARY
C
      T(1,11)=T(1,1)*(-8.*T(1,5)+9.*T(2,5)-T(3,5))*G5
     .+T(1,2)*(-8.*T(1,6)+9.*T(2,6)-T(3,6))*G5
      T(1,12)=T(1,3)*(-8.*T(1,6)+9.*T(2,6)-T(3,6))*G5
     .+T(1,2)*(-8.*T(1,5)+9.*T(2,5)-T(3,5))*G5
      T(1,13)=.5*(T(1,1)*(9.*T(2,5)*T(2,5)-8.*T(1,5)*T(1,5)-
     .T(3,5)*T(3,5))*G5+T(1,3)*(9.*T(2,6)*T(2,6)-8.*T(1,6)*T(1,6)-
     .T(3,6)*T(3,6))*G5)+T(1,2)*(9.*T(2,6)*T(2,5)-8.*T(1,6)*T(1,5)-
     .T(3,6)*T(3,5))*G5+T00*(9.*T(2,4)*T(2,10)-8.*T(1,4)*T(1,10)-
     .T(3,4)*T(3,10))*G5
C
C RIGHT BOUNDARY
C
      T(JDIM,11)=T(JDIM,1)*(8.*T(JDIMP,5)-9.*T(JDIM,5)+T(JDIM1,5))*G5
     .+T(JDIM,2)*(8.*T(JDIMP,6)-9.*T(JDIM,6)+T(JDIM1,6))*G5
      T(JDIM,12)=T(JDIM,3)*(8.*T(JDIMP,6)-9.*T(JDIM,6)+T(JDIM1,6))*G5
     .+T(JDIM,2)*(8.*T(JDIMP,5)-9.*T(JDIM,5)+T(JDIM1,5))*G5
      T(JDIM,13)=.5*(T(JDIM,1)*
     .(-9.*T(JDIM,5)*T(JDIM,5)+8.*T(JDIMP,5)*T(JDIMP,5)
```

```
     .+T(JDIM1,5)*T(JDIM1,5))*G5+T(JDIM,3)*
     .(-9.*T(JDIM,6)*T(JDIM,6)+8.*T(JDIMP,6)*T(JDIMP,6)
     .+T(JDIM1,6)*T(JDIM1,6))*G5)+T(JDIM,2)*
     .(-9.*T(JDIM,6)*T(JDIM,5)+8.*T(JDIMP,6)*T(JDIMP,5)+
     .T(JDIM1,6)*T(JDIM1,5))*G5+T00*
     .(-9.*T(JDIM,4)*T(JDIM,10)+8.*T(JDIMP,4)*T(JDIMP,10)+
     .T(JDIM1,4)*T(JDIM1,10))*G5
C
      DO 82 J=1,JDIM
   82 T(J,9)=SJ(J,K,3)*TT(J,K,2)
      DO 92 N=11,13
      DO 92 J=1,JDIM
   92 T(J,N)=T(J,N)*T(J,9)
C
      DO 300 N=2,4
      DO 300 J=1,JDIM1
      RES(J,K,N)=RES(J,K,N)-(T(J+1,N+9)-T(J,N+9))*REI
  300 CONTINUE
C
      IF (IJAC.NE.1) GO TO 400
C
C IMPLICIT TERMS
C
      DO 42 J=1,JDIM
      VM1(J,2,1)=(-T(J+1,4)*(T(J,1)*T(J+1,5)
     .                      +T(J,2)*T(J+1,6)))*T(J,9)
      VM1(J,2,2)=(T(J+1,4)*T(J,1))*T(J,9)
      VM1(J,2,3)=(T(J+1,4)*T(J,2))*T(J,9)
      VM1(J,3,1)=(-T(J+1,4)*(T(J,2)*T(J+1,5)
     .                      +T(J,3)*T(J+1,6)))*T(J,9)
      VM1(J,3,2)=(T(J+1,4)*T(J,2))*T(J,9)
      VM1(J,3,3)=(T(J+1,4)*T(J,3))*T(J,9)
      VM1(J,4,1)=(-T(J+1,4)*(T00*T(J+1,4)*T(J+1,10)
     .          +(T(J,1)-.5*T0)*T(J+1,5)*T(J+1,5)
     .          +2.*T(J,2)*T(J+1,5)*T(J+1,6)
     .          +(T(J,3)-.5*T0)*T(J+1,6)*T(J+1,6)))*T(J,9)
      VM1(J,4,2)=-VM1(J,2,1)-(T0*T(J+1,4)*T(J+1,5))*T(J,9)
      VM1(J,4,3)=-VM1(J,3,1)-(T0*T(J+1,4)*T(J+1,6))*T(J,9)
      VM1(J,4,4)=(T0*T(J+1,4))*T(J,9)
   42 CONTINUE
C
      DO 52 J=1,JDIM
      VM2(J,2,1)=(-T(J,4)*(T(J,1)*T(J,5)
     .                    +T(J,2)*T(J,6)))*T(J,9)
      VM2(J,2,2)=(T(J,4)*T(J,1))*T(J,9)
      VM2(J,2,3)=(T(J,4)*T(J,2))*T(J,9)
      VM2(J,3,1)=(-T(J,4)*(T(J,2)*T(J,5)
     .                    +T(J,3)*T(J,6)))*T(J,9)
      VM2(J,3,2)=(T(J,4)*T(J,2))*T(J,9)
      VM2(J,3,3)=(T(J,4)*T(J,3))*T(J,9)
      VM2(J,4,1)=(-T(J,4)*(T00*T(J,4)*T(J,10)
     .          +(T(J,1)-.5*T0)*T(J,5)*T(J,5)
     .          +2.*T(J,2)*T(J,5)*T(J,6)
     .          +(T(J,3)-.5*T0)*T(J,6)*T(J,6)))*T(J,9)
      VM2(J,4,2)=-VM2(J,2,1)-(T0*T(J,4)*T(J,5))*T(J,9)
      VM2(J,4,3)=-VM2(J,3,1)-(T0*T(J,4)*T(J,6))*T(J,9)
      VM2(J,4,4)=(T0*T(J,4))*T(J,9)
   52 CONTINUE
C
C
      DO 210 N=2,4
      DO 210 M=1,4
      DO 210 J=3,JDIM1
      AK(K,J,N,M)=-VM2(J-1,N,M)*REI
      BK(K,J,N,M)=(VM1(J-1,N,M)+VM2(J,N,M))*REI
      CK(K,J,N,M)=-VM1(J,N,M)*REI
  210 CONTINUE
C
      DO 212 N=2,4
      DO 212 M=1,4
      AK(K,2,N,M)=G1*VM2(1,N,M)*REI
      BK(K,2,N,M)=(G2*VM1(1,N,M)+G3*VM2(2,N,M))*REI
```

```
      CK(K,2,N,M)=G4*VM1(2,N,M)*REI
      AK(K,JDIM,N,M)=G4*VM2(JDIM1,N,M)*REI
      BK(K,JDIM,N,M)=(G3*VM1(JDIM1,N,M)+G2*VM2(JDIM,N,M))*REI
      CK(K,JDIM,N,M)=G1*VM1(JDIM,N,M)*REI
  212 CONTINUE
C
  400 CONTINUE
C
C CROSS DERIVATIVES, HORIZONTAL FACES, (SYMMETRICAL TREATMENT, RHS ONLY)
C
      DO 500 K=1,KDIM
      DO 500 J=1,JDIM1
      TT(J,K,3)=4.E0*SK(J,K,1)*CRK(J,K,1)*G5+SK(J,K,2)*CRK(J,K,2)
      TT(J,K,4)=SK(J,K,2)*CRK(J,K,1)-2.E0*SK(J,K,1)*CRK(J,K,2)*G5
      TT(J,K,5)=SK(J,K,1)*CRK(J,K,2)-2.E0*SK(J,K,2)*CRK(J,K,1)*G5
      TT(J,K,6)=4.E0*SK(J,K,2)*CRK(J,K,2)*G5+SK(J,K,1)*CRK(J,K,1)
      TT(J,K,7)=SK(J,K,2)*CRK(J,K,2)+SK(J,K,1)*CRK(J,K,1)
      TT(J,K,8)=(SK(J,K,2)*CRK(J,K,1)-SK(J,K,1)*CRK(J,K,2))*5.*G5
  500 CONTINUE
C
      DO 502 K=2,KDIM1
      DO 502 J=1,JDIM
      TT(J,K,9)=.5*(Q(J,K,2)+Q(J,K-1,2))
      TT(J,K,10)=.5*(Q(J,K,3)+Q(J,K-1,3))
      TT(J,K,11)=(Q(J,K,4)+Q(J,K-1,4))/(Q(J,K,1)+Q(J,K-1,1))
      TT(J,K,12)=.25*(Q(J,K,2)+Q(J,K-1,2))*(Q(J,K,3)+Q(J,K-1,3))
  502 CONTINUE
C
      DO 504 J=1,JDIM1
      TT(J,1,9)=QK0(J,2)
      TT(J,1,10)=QK0(J,3)
      TT(J,1,11)=QK0(J,4)/QK0(J,1)
      TT(J,1,12)=QK0(J,2)*QK0(J,3)
      TT(J,KDIM,9)=Q(J,KDIM,2)
      TT(J,KDIM,10)=Q(J,KDIM,3)
      TT(J,KDIM,11)=Q(J,KDIM,4)/Q(J,KDIM,1)
      TT(J,KDIM,12)=Q(J,KDIM,2)*Q(J,KDIM,3)
  504 CONTINUE
C
      TT(JDIM,1,9)=QK0(JDIM1,2)
      TT(JDIM,1,10)=QK0(JDIM1,3)
      TT(JDIM,1,11)=QK0(JDIM1,4)/QK0(JDIM1,1)
      TT(JDIM,1,12)=QK0(JDIM1,2)*QK0(JDIM1,3)
      TT(JDIM,KDIM,9)=Q(JDIM1,KDIM,2)
      TT(JDIM,KDIM,10)=Q(JDIM1,KDIM,3)
      TT(JDIM,KDIM,11)=Q(JDIM1,KDIM,4)/Q(JDIM1,KDIM,1)
      TT(JDIM,KDIM,12)=Q(JDIM1,KDIM,2)*Q(JDIM1,KDIM,3)
C
      DO 505 K=2,KDIM1
      T(K,1)=.5*(QJ0(K,2)+QJ0(K-1,2))
      T(K,2)=.5*(QJ0(K,3)+QJ0(K-1,3))
      T(K,3)=(QJ0(K,4)+QJ0(K-1,4))/(QJ0(K,1)+QJ0(K-1,1))
      T(K,4)=.25*(QJ0(K,2)+QJ0(K-1,2))*(QJ0(K,3)+QJ0(K-1,3))
  505 CONTINUE
C
      T(1,1)=QK0(1,2)
      T(1,2)=QK0(1,3)
      T(1,3)=QK0(1,4)/QK0(1,1)
      T(1,4)=QK0(1,2)*QK0(1,3)
      T(KDIM,1)=Q(1,KDIM,2)
      T(KDIM,2)=Q(1,KDIM,3)
      T(KDIM,3)=Q(1,KDIM,4)/Q(1,KDIM,1)
      T(KDIM,4)=Q(1,KDIM,2)*Q(1,KDIM,3)
C
      DO 506 L=1,4
      DO 506 K=1,KDIM
      DO 506 J=2,JDIM2
      TT(J,K,L+12)=.5*(TT(J+1,K,L+8)-TT(J-1,K,L+8))
  506 CONTINUE
C
      DO 507 L=1,2
      DO 507 K=1,KDIM
```

```
      DO 507 J=2,JDIM2
      TT(J,K,L+16)=.25*(TT(J+1,K,L+8)*TT(J+1,K,L+8)-TT(J-1,K,L+8)*
     .            TT(J-1,K,L+8))
  507 CONTINUE
C
      DO 508 L=1,4
      DO 508 K=1,KDIM
      TT(1,K,L+12)=(-4.*T(K,L)+3.*TT(1,K,8+L)+TT(2,K,8+L))*G5
      TT(JDIM1,K,L+12)=(4.*TT(JDIM,K,8+L)-3.*TT(JDIM1,K,8+L)
     .            -TT(JDIM2,K,8+L))*G5
  508 CONTINUE
C
      DO 509 L=1,2
      DO 509 K=1,KDIM
      TT(1,K,L+16)=(-4.*T(K,L)*T(K,L)+3.*TT(1,K,8+L)*TT(1,K,8+L)+
     .         TT(2,K,8+L)*TT(2,K,8+L))*.5*G5
      TT(JDIM1,K,L+16)=(4.*TT(JDIM,K,8+L)*TT(JDIM,K,8+L)-3.*
     .         TT(JDIM1,K,8+L)*TT(JDIM1,K,8+L)-TT(JDIM2,K,8+L)*
     .         TT(JDIM2,K,8+L))*.5*G5
  509 CONTINUE
C
      DO 510 K=1,KDIM
      DO 510 J=1,JDIM1
      TT(J,K,10)=(TT(J,K,3)*TT(J,K,13)+TT(J,K,4)*TT(J,K,14))*TT(J,K,1)
      TT(J,K,11)=(TT(J,K,5)*TT(J,K,13)+TT(J,K,6)*TT(J,K,14))*TT(J,K,1)
      TT(J,K,12)=(T00*TT(J,K,7)*TT(J,K,15)+TT(J,K,3)*TT(J,K,17)+
     . TT(J,K,6)*TT(J,K,18)+TT(J,K,5)*TT(J,K,16)+TT(J,K,8)*
     . TT(J,K,14)*TT(J,K,9))*TT(J,K,1)
  510 CONTINUE
C
      DO 512 L=2,4
      DO 512 K=1,KDIM1
      DO 512 J=1,JDIM1
      RES(J,K,L)=RES(J,K,L)-(TT(J,K+1,L+8)-TT(J,K,L+8))*REI
  512 CONTINUE
C
C CROSS DERIVATIVES, VERTICAL FACES, (SYMMETRICAL TREATMENT, RHS ONLY)
C
      DO 600 K=1,KDIM1
      DO 600 J=1,JDIM
      TT(J,K,3)=4.*SJ(J,K,1)*CRJ(J,K,1)*G5+SJ(J,K,2)*CRJ(J,K,2)
      TT(J,K,4)=SJ(J,K,2)*CRJ(J,K,1)-2.*SJ(J,K,1)*CRJ(J,K,2)*G5
      TT(J,K,5)=SJ(J,K,1)*CRJ(J,K,2)-2.*SJ(J,K,2)*CRJ(J,K,1)*G5
      TT(J,K,6)=4.*SJ(J,K,2)*CRJ(J,K,2)*G5+SJ(J,K,1)*CRJ(J,K,1)
      TT(J,K,7)=SJ(J,K,2)*CRJ(J,K,2)+SJ(J,K,1)*CRJ(J,K,1)
      TT(J,K,8)=(SJ(J,K,2)*CRJ(J,K,1)-SJ(J,K,1)*CRJ(J,K,2))*5.*G5
  600 CONTINUE
C
      DO 602 K=1,KDIM
      DO 602 J=2,JDIM1
      TT(J,K,9)=.5*(Q(J,K,2)+Q(J-1,K,2))
      TT(J,K,10)=.5*(Q(J,K,3)+Q(J-1,K,3))
      TT(J,K,11)=(Q(J,K,4)+Q(J-1,K,4))/(Q(J,K,1)+Q(J-1,K,1))
      TT(J,K,12)=.25*(Q(J,K,2)+Q(J-1,K,2))*(Q(J,K,3)+Q(J-1,K,3))
  602 CONTINUE
C
      DO 604 K=1,KDIM1
      TT(1,K,9)=QJ0(K,2)
      TT(1,K,10)=QJ0(K,3)
      TT(1,K,11)=QJ0(K,4)/QJ0(K,1)
      TT(1,K,12)=QJ0(K,2)*QJ0(K,3)
      TT(JDIM,K,9)=Q(JDIM,K,2)
      TT(JDIM,K,10)=Q(JDIM,K,3)
      TT(JDIM,K,11)=Q(JDIM,K,4)/Q(JDIM,K,1)
      TT(JDIM,K,12)=Q(JDIM,K,2)*Q(JDIM,K,3)
  604 CONTINUE
C
      TT(1,KDIM,9)=QJ0(KDIM1,2)
      TT(1,KDIM,10)=QJ0(KDIM1,3)
      TT(1,KDIM,11)=QJ0(KDIM1,4)/QJ0(KDIM1,1)
      TT(1,KDIM,12)=QJ0(KDIM1,2)*QJ0(KDIM1,3)
      TT(JDIM,KDIM,9)=Q(JDIM,KDIM1,2)
```

```
      TT(JDIM,KDIM,10)=Q(JDIM,KDIM1,3)
      TT(JDIM,KDIM,11)=Q(JDIM,KDIM1,4)/Q(JDIM,KDIM1,1)
      TT(JDIM,KDIM,12)=Q(JDIM,KDIM1,2)*Q(JDIM,KDIM1,3)
C
      DO 605 J=2,JDIM1
      T(J,1)=.5*(QK0(J,2)+QK0(J-1,2))
      T(J,2)=.5*(QK0(J,3)+QK0(J-1,3))
      T(J,3)=(QK0(J,4)+QK0(J-1,4))/(QK0(J,1)+QK0(J-1,1))
      T(J,4)=.25*(QK0(J,2)+QK0(J-1,2))*(QK0(J,3)+QK0(J-1,3))
605   CONTINUE
C
      T(1,1)=QJ0(1,2)
      T(1,2)=QJ0(1,3)
      T(1,3)=QJ0(1,4)/QJ0(1,1)
      T(1,4)=QJ0(1,2)*QJ0(1,3)
      T(JDIM,1)=Q(JDIM,1,2)
      T(JDIM,2)=Q(JDIM,1,3)
      T(JDIM,3)=Q(JDIM,1,4)/Q(JDIM,1,1)
      T(JDIM,4)=Q(JDIM,1,2)*Q(JDIM,1,3)
C
      DO 606 L=1,4
      DO 606 K=2,KDIM2
      DO 606 J=1,JDIM
      TT(J,K,L+12)=.5*(TT(J,K+1,L+8)-TT(J,K-1,L+8))
606   CONTINUE
C
      DO 607 L=1,2
      DO 607 K=2,KDIM2
      DO 607 J=1,JDIM
      TT(J,K,L+16)=.25*(TT(J,K+1,L+8)*TT(J,K+1,L+8)-TT(J,K-1,L+8)*
     .          TT(J,K-1,L+8))
607   CONTINUE
C
      DO 608 L=1,4
      DO 608 J=1,JDIM
      TT(J,1,L+12)=(-4.*T(J,L)+3.*TT(J,1,8+L)+TT(J,2,8+L))*G5
      TT(J,KDIM1,L+12)=(4.*TT(J,KDIM,8+L)-3.*TT(J,KDIM1,8+L)-
     .       TT(J,KDIM2,8+L))*G5
608   CONTINUE
C
      DO 609 L=1,2
      DO 609 J=1,JDIM
      TT(J,1,L+16)=(-4.*T(J,L)*T(J,L)+3.*TT(J,1,8+L)*TT(J,1,8+L)+
     .       TT(J,2,8+L)*TT(J,2,8+L))*.5*G5
      TT(J,KDIM1,L+16)=(4.*TT(J,KDIM,8+L)*TT(J,KDIM,8+L)-3.*
     .       TT(J,KDIM1,8+L)*TT(J,KDIM1,8+L)-TT(J,KDIM2,8+L)*
     .       TT(J,KDIM2,8+L))*.5*G5
609   CONTINUE
C
      DO 610 K=1,KDIM1
      DO 610 J=1,JDIM
      TT(J,K,10)=(TT(J,K,3)*TT(J,K,13)+TT(J,K,4)*TT(J,K,14))*TT(J,K,2)
      TT(J,K,11)=(TT(J,K,5)*TT(J,K,13)+TT(J,K,6)*TT(J,K,14))*TT(J,K,2)
      TT(J,K,12)=(T00*TT(J,K,7)*TT(J,K,15)+TT(J,K,3)*TT(J,K,17)+
     . TT(J,K,6)*TT(J,K,18)+TT(J,K,5)*TT(J,K,16)+TT(J,K,8)*
     . TT(J,K,14)*TT(J,K,9))*TT(J,K,2)
610   CONTINUE
C
      DO 612 L=2,4
      DO 612 K=1,KDIM1
      DO 612 J=1,JDIM1
      RES(J,K,L)=RES(J,K,L)-(TT(J+1,K,L+8)-TT(J,K,L+8))*REI
612   CONTINUE
C
      RETURN
      END
```

References

1. William, R. M., " National Aero-Space Plane ," *Aerospace America* , November 1986, pp. 18-22.

2. Rumsey, C. L., Thomas, J. L., Warren, G. P., and Liv, J. C., "Upwind Navier-Stokes Solutions for Separated Periodic Flows ," AIAA-86-0247, January 1986.

3. Thomas, J. L., and Walters, R. W., "Upwind Relaxation Algorithms for the Navier-Stokes Equations ," AIAA Journal, Vol. 25, April 1987, pp. 527-534.

4. MacCormack, R. W., "The Effect of Viscosity in Hypervelocity Impact Cratering ," AIAA-69-354, 1969.

5. Briley, W. R., and McDonald, H., "An Implicit Numerical Method for the Multidimensional Compressive Navier-Stokes Equations ," United Aircraft Research Lab, Report M911363-6, 1973.

6. Beam, R., and Warming, R., " An Implicit Factored Scheme for the Compressible Navier-Stokes Equations ," AIAA Journal, Vol. 16, 1978.

7. MacCormack, R. W., "A Numerical Method for solving the Equations of Compressible viscous flow ," AIAA-81-0110, 1981.

8. Ong, C., and Knight, D., " A Comparative Study of the Hybrid MacCormack and Implicit Beam-Warming Algorithms for a Two-Dimensional Supersonic Compression Corner ," AIAA-86-0204, 1986.

9. Anderson, W. K., Thomas, J. L., and Van Leer, B., "A Comparison of Finite Volume Flux Vector Splittings for the Euler Equations," AIAA 85-0122, January 1985.

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

11. Thomas, J. L., and Newsome, R. W., " Navier-Stokes Computations of Lee-Side Flows Over Delta Wings ," AIAA-86-1049, May 1980.

12. Newsome, R. W., Walters R. W., and Thomas, J. L., " An Efficient Iteration Strategy for Upwind/Relaxation Solutions to the Thin-Layer Navier-Stokes Equations ," AIAA-87-1113.

13. Steger, J. L., and Warming, R. F., " Flux Vector Splitting of the Inviscid Gasdynamics Equations with Application to Finite-Difference Methods, " AIAA -87-1113.

14. Van Leer, B., " Flux-Vector Splitting for the Euler Equations," ICASE Report No. 82-30, September 1982, also in *Lecture Notes in Physics,* Vol. 170, 1983, pp. 507-512.

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

16. Walters, R. W., and Thomas, J. L., " Advances in Upwind Relaxation Methods, " to be published.

17. Vigneron, Y. C., Tannehill, J. C., and Rakich, J. V., "Calculation of Supersonic Viscous Flow Over Delta Wings with Sharp Subsonic Leading Edges, " NASA TM-78500, June 1978.

18. Viviand, H.,"Comparison of Numerical Solutions to Internal Flow in a Double Throat Nozzle," GAMM Committee for Numerical Methods in Fluid Mechanics, France, December 1985.

19. Thomas, J. L., Walters, R. W., Van Leer, B., and Rumsey, C. L., "An Implicit Flux-Split Algorithm for the Navier-Stokes Equations" GAMM Committee for Numerical Methods in Fluid Mechanics, France, December 1985.

20. Mitchell, A. R., and Griffiths, D. F., *The Finite Difference Method in Partial Differential Equations*, John Wiley and Sons, 1980, pp. 125 - 129.

21. Skebe, S. A., Greber, I., and Hings, W. R., "Investigation of Two-Dimensional Shock-Wave/Boundary-Layer Interactions,"AIAA Journal, vol. 25, no.6, June 1987.

22. Skebe, S. A., "Investigation of Two Dimensional Shock Boundary Layer Interactions,"Ph.D. Dissertation, Case Western University, Cleveland, OH, August 25, 1983.

23. Richtmyer, R. D., and Morton, K. W., "Difference Methods for Initial-Value Problems,"Interscience Publishers, 1967, pp.206-208.

The vita has been removed from the scanned document