AN APPROXIMATE FACTORIZATION SOLUTION OF THE NAVIER-STOKES EQUATIONS FOR TRANSONIC FLOW USING BODY-FITTED COORDINATES WITH APPLICATION TO NACA 64A010 AIRFOILS

BY

G. KYLE COOPER

A Dissertation
Submitted to the Faculty of
Mississippi State University
in Partial Fulfillment of the Requirements
for the Degree of Doctor of Philosophy
in the College of Engineering

Mississippi State, Mississippi

August 1980

AN APPROXIMATE FACTORIZATION SOLUTION OF THE NAVIER-STOKES EQUATIONS FOR TRANSONIC FLOW USING BODY-FITTED COORDINATES WITH APPLICATION TO NACA 64A010 AIRFOILS

BY

G. KYLE COOPER

A Dissertation
Submitted to the Faculty of
Mississippi State University
in Partial Fulfillment of the Requirements
for the Degree of Doctor of Philosophy
in the College of Engineering

Mississippi State, Mississippi

August 198[illegible]

AN APPROXIMATE FACTORIZATION SOLUTION
OF THE NAVIER-STOKES EQUATIONS FOR TRANSONIC FLOW
USING BODY-FITTED COORDINATES WITH APPLICATION TO
NACA 64A010 AIRFOILS

By

G. Kyle Cooper

Professor of Aerospace
Engineering (Chairman of
Committee and Dissertation
Director)

Professor of Aerospace
Engineering (Member of
Committee)

Associate Professor of
Aerospace Engineering
(Member of Committee)

Associate Professor of
Mathematics (Member of
Committee)

Associate Professor of
Mechanical Engineering
(Member of Committee)

Professor and Head of the
Department of Aerospace
Engineering

Dean of the College of
Engineering

Vice President for
Graduate Studies and Research

August 1980

ACKNOWLEDGEMENTS

Every degree recipient, just as every Oscar winner, has a whole cast of individuals without whom he would not have attained the distinction awarded him. I can not begin to list the people who in some way made this dissertation possible. Thus, without attempting to be all inclusive, I wish to recognize the following individuals who have made extraordinary contributions toward the successful conclusion of this research:

Marla P. Cooper, loving wife,
Dr. Joe F. Thompson, advisor and exemplar,
Dr. Suhrit K. Dey, advisor and instigator,
Denice Heath, ace typist and editor, and
Kitty Haigler, computer wizard and my NASA connection.

This research was supported by NASA Grant 25-001-055, Langley Research Center.

Thank you.

G.K.C.

Mississippi State University
August 1980

ABSTRACT

G. Kyle Cooper, Doctor of Philosophy, 1980

Major: Engineering, Department of Aerospace Engineering

Title of Dissertation: An Approximate Factorization Solution of the Navier-Stokes Equations for Transonic Flow Using Body-Fitted Coordinates with Application to NACA 64A010 Airfoils

Directed by: Dr. Joe F. Thompson

Pages in Dissertation: 110 Words in Abstract: 525

Abstract

Although aircraft have been routinely flown at transonic speeds for the last two decades, the designers of these machines have had to formulate their designs almost exclusively on the basis of experience, in contrast to subsonic or supersonic aircraft design in which a welter of analytical, experimental, and numerical techniques exist. Since this type of fluid flow is characterized by complex viscid-inviscid interactions, the development of fast numerical models of the full Navier-Stokes equations has promised to alleviate this situation. One such model is the approximate factorization algorithm introduced by beam and Warming and implemented by Steger, et al. This research, then, is principally concerned with an independent implementation of this numerical algorithm and initial studies of its ability to efficiently and accurately describe transonic flow about an NACA 64A010 airfoil section.

The approximate factorization algorithm is developed from the non-dimensional, conservative, vectorized Navier-Stokes equations expressed

in curvilinear coordinates. Equations of state and transport coefficient relations appropriate to atmospheric air are appended to close the system of partial differential equations. An algebraic turbulence model due to Baldwin and Lomax is also incorporated into the equation set. The coordinate generation method developed by Thompson, et al is used to produce the desired coordinate transformations. Boundary conditions on the airfoil surface are formulated so as to allow suction and/or blowing from the surface and to emulate either an isothermal or adiabatic wall. Outer boundaries are placed ten chord lengths from the airfoil and their boundary conditions formulated so that the inflow properties can be varied and the outflow properties determined by extrapolation. Fourth-order artificial viscosity proves to be necessary for high Reynolds number flows.

This algorithm was verified by investigating the flow about an NACA 64A010 airfoil at 0°, 2°, and 3.5° angle of attack for free-stream conditions of 2 x 10^6 Reynolds number and 0.8 Mach number. The flow was initiated by either gradually decreasing the degree of fluid penetration of the airfoil from total to none, or by using a body force to gradually accelerate the airfoil and its attached coordinate system from zero velocity to free-stream values. One zero degree case was run as laminar flow while all the other cases used the algebraic turbulence model. Also, cases were run to verify the model's ability to maintain a free-stream solution. As an aid in evaluating the results, a set of Mach number contour plots and coefficient of pressure graphs were prepared.

Overall results were in good qualitative agreement with the wind tunnel data sets of Johnson and Bachalo. Unfortunately, while nondimensional times of six were attained, numerical difficulties prevented any case from reaching a true steady state. In the last test case attempted, that at an angle of attack of 3.5°, there was no doubt that a shock was forming on the airfoil and that separation had occurred. Computer times for the 113 x 51 grid used were encouraging, averaging 35 seconds/time-step on a Univac 1100/80 computer and 5 seconds/time-step in scalar mode on the Cyber 203 (Star). It is concluded that approximate factorization techniques, while they still need some work, can definitely be used to advantage in at least two-dimensional transonic flow problems.

TABLE OF CONTENTS

TABLE OF CONTENTS (continued)

LIST OF SYMBOLS

Symbol	
a_i	Viscous Coefficients, Equations (3.23) and (3.30)
A	u- or U-Convective Flux Vector, Equation (3.2) or (3.17)
b_i	Viscous Coefficients, Equations (3.24), (3.25) and (3.30)
B	v- or V-Convective Flux Vector, Equation (3.2) or (3.18)
c_i	Viscous Coefficients, Equations (3.26) and (3.30)
D	x- or ξ-Viscous Flux Vector, Equation (3.3) or (3.19)
D_1	ξ-Derivative Part of D^*, Compare Equations (3.20) and (3.21)
D_2	η-Derivative Part of D^*, Compare Equations (3.20) and (3.21)
e	Nondimensional Total Energy
E	y- or η-Viscous Flux Vector, Equation (3.4) or (3.20)
E_1	ξ-Derivative Part of E^*, Compare Equations (3.20) and (3.21)
E_2	η-Derivative Part of E^*, Compare Equations (3.20) and (3.21)
I	Four-by-Four Identity Matrix
J	Determinant of Coordinate Transformation Jacobian Matrix, Equation (3.27)
k	Nondimensional Thermal Conductivity, Equation (3.7)
M	Mach Number, Appendix A
P	Jacobian Matrix of "A" with Respect to "q", Equation (4.5), or Nondimensional Pressure, Equation (3.6)
Pr	Prandtl Number, Appendix A
Pr_T	Turbulent Prandtl Number
q	Conservative Dependent Variable Vector, Equation (3.2) or (3.16)
$\dot{q}''$	Nondimensional Surface Heat Flux, Equation (3.31)
Q	Jacobian Matrix of "B" with Respect to "q", Equation (4.5)

Symbol	
Re	Reynolds Number, Appendix A
S_1	Nondimensional Sutherland Viscosity Law Reference Temperature, Appendix A
t	Nondimensional Time, Independent Variable
T	Nondimensional Temperature, Equation (3.5)
u	Nondimensional Velocity Component in x-Direction
U	Contravariant Velocity Tangent to ξ-Lines, Equation (3.28)
v	Nondimensional Velocity Component in y-Direction
V	Contravariant Velocity Tangent to η-Lines, Equation (3.29)
W	Jacobian Matrix of "D", with Respect to "q_ξ", Equation (4.5)
x	Nondimensional Length Parallel to Chord Line, Independent Variable
X	Jacobian Matrix of "D_2" with Respect to "q_η", Equation (4.5)
y	Nondimensional Length Perpendicular to Chord Line, Independent Variable
Y	Jacobian Matrix of "E_1" with Respect to "q_ξ", Equation (4.5)
Z	Jacobian Matrix of "E_2" with Respect to "q_η", Equation (4.5)
α	Normal Derivative Coefficient, Equation (3.32b), or Angle of Attack (Figures)
β	Normal Derivative Coefficient, Equation (3.32b)
γ	Ratio of Specific Heats, Appendix A
Δ	Forward Difference Operator in Time ($\Delta q^n = q^{n+1} - q^n$)
Δt	Nondimensional Time-Step
ε	Explicit Artificial Viscosity Coefficient, Equation (4.16)
η	y-Like Curvilinear Coordinate, Equation (3.14)

Symbol

θ	Time-Difference Parameter, Equation (4.1)
λ	Nondimensional Second Coefficient of Viscosity, Equation (3.9)
μ	Nondimensional First Coefficient of Viscosity, Equation (3.8)
μ_T	Nondimensional Eddy (Turbulent) Viscosity
ξ	x-Like Curvilinear Coordinate, Equation (3.13)
ρ	Nondimensional Density
τ	Transformed Time Coordinate, Equation (3.12)
ϕ	Time-Difference Parameter, Equation (4.1)
∇_N	Derivative Normal to a Boundary Surface, Equation (3.31)

Superscripts

$(\ )^n$	Indicates Evaluation at Time-Step Number "n"
$(\ )^*$	Curvilinear Vector Analogous to Cartesian Vector, Equations (3.16) through (3.20)
$(\bar{\ })$	Intermediate Value Produced by ξ-Sweep, Equation (4.9)

Subscripts

$(\ )_t$	Partial Derivative with Respect to Time "t"
$(\ )_x$	Partial Derivative with Respect to "x"
$(\ )_y$	Partial Derivative with Respect to "y"
$(\ )_\eta$	Partial Derivative with Respect to "η"
$(\ )_\xi$	Partial Derivative with Respect to "ξ"
$(\ )_{i,j}$	Indicates Evaluation at the Intersection of the ith ξ-Line and jth η-Line

LIST OF FIGURES

LIST OF FIGURES (continued)

I. INTRODUCTION

Although the art of modeling fluid flows has progressed to the point where important parameters of certain complex flows can be routinely and confidently predicted, there are still some interesting flows for which current methods of analysis (experimental, analytical, or numerical) give approximate results at best. One of these difficult fluid flow problems is the description of transonic flow over geometries of practical interest, in particular single- and multi-element airfoils. This type of flow is of particular interest to the numerical analyst since it is of major concern to the aircraft industry and since it proves to be inherently too complicated a flow to be fully described by either pure analysis or experimental techniques. Until recently, the same could be said of the available numerical methods, however, with the appearance of improved ADI (alternating direction, implicit) techniques, this is no longer the case. The major thrust of the research reported herein was to establish just this premise and to verify the numerical algorithm developed for this purpose.

The numerical algorithm itself is based upon a finite difference representation of the full, Reynolds-averaged, Navier-Stokes equations as developed by Beam and Warming [c] and including some of the implementation ideas due to Steger [b]. The set of equations is closed by using the standard approximations for atmospheric air and including an algebraic turbulence model developed by Balwin and Lomax [a] from the work of Cebeci [p]. Chief among the attributes of this algorithm is that it is a non-iterative,second-order accurate, implicit formulation of the conservative form of the governing equations which remains stable and accurate for Courant numbers much larger than unity and which attains a steady-state solution independent of the time step. This method offers the practical advantage of being able to calculate accurate transient and steady-state solutions in time periods measured in minutes, rather than the hours of older methods, at the cost of requiring large amounts of data storage even for two-dimensional problems.

Almost all recent numerical models of fluid flows are based upon forms of the Navier-Stokes equations expressed in curvilinear coordinates. That this should be so is apparent from the fact that most fluid flows of practical interest have inherently disparate length scales arising from the geometry and/or the equations themselves. Thus, curvilinear coordinates offer an automatic adjustment to these varying length scales and the bonus of simple boundary condition specification. One of the most widely used techniques for automatically generating such coordinates, that due to Thompson, Thames, and Mastin [d], has been adapted to this research. It allows almost complete control over boundary location and provides for concentration of coordinate lines about the airfoil.

Of course any new numerical algorithm and in particular any computer coded version of any numerical algorithm, needs to be extensively verified against known solutions and/or trusted data sets. Although, as mentioned in the opening paragraph, there are scant theoretical solutions for the flows considered here, there have been some recent efforts at developing a systematic set of experimental data for use in testing numerical models. The paper of Johnson and Bachalo [m], detailing wind tunnel test results for the NACA 64A010 airfoil section, was particularly influential in choosing the flow conditions to be modeled. Their results include interferograms, calculated Mach number contours, and an extensive compilation of turbulence parameters for a freestream Mach number of 0.8, and Reynolds number of two million, for various angles of attack.

The dissertation contained in this volume basically follows the same logical development as the foregoing discussion. Following a survey of the literature which this author considers to be germane to the topics covered, the nondimensional, Reynolds-averaged, vectorized, fully conservative Navier-Stokes equations are put into a form appropriate to curvilinear coordinates. Also, the equations of state and the functional form of the transport coefficients are presented as part of the development. The succeeding chapter details the transformation of these partial differential equations into a set of difference equations which form an ADI scheme involving a non-iterative, block-tridiagonal inversion during each sweep. Determination of the transformed coordinates and appropriate finite difference boundary conditions are also included in this chapter. Next, the verification trials for the computer coded algorithm are set forth. Results for the NACA 64A010 airfoil at 0°, 2°, and 3.5° angle of attack under transonic conditions are presented and checked for internal consistency. The available experimental data is also compared with these results and appropriate comments made about their degree of and/or lack of correspondence. Lastly, in the concluding chapter, some general and some specific conclusions about the algorithm and data set are made and some recommendations for future research work given.

It is hoped that the casual reader, should there be such a person, will find the exposition clear and enlightening, and that the serious student will find the presentation to be useful, self-contained, and stimulating.

II. LITERATURE SURVEY

The purpose of this chapter is to acknowledge those authors whose published work has contributed to this research in some way; it is also intended to provide the interested reader with some related references outside of the narrow scope of this research. However, in no way is this chapter to be considered as a definitive, exhaustive or complete treatment of the significant publications dealing with the topics touched upon herein.

Although some current analytical work is being done on the problem of transonic flow, most of it is based upon the techniques developed earlier in this century before the advent of electronic computation. Landau and Lifshitz [q] in their survey of the state of fluid mechanics (1959) give a rigorous introduction to the analysis of transonic flows.

Some of the earliest numerical work in this area was based on various simplifications of the full potential flow equations; indeed, this approach appears to be a well-entrenched field of study for at least the foreseeable future. A good source book for some of the current (1978) potential methods is listed as reference [r].

Numerical solutions of the full Navier-Stokes equations became practical for transonic flows when MacCormack [s] introduced his "rapid solver" algorithm. However, being an explicit technique, it possesses certain relaxation limitations. Although implicit algorithms have a long history of development in this field, it was the appearance of approximate factorization type techniques which made them more than of just theoretical interest. These algorithms were first advanced by Beam and Warming [c,f] and by Briley and McDonald [t] independently.

The Beam and Warming formulation has been successfully applied to a variety of transonic flows by Steger [b], Pulliam and Steger [g], and Steger and Bailey [n].

The advantages inherent in the use of curvilinear coordinates ha've been made readily available by the pioneering work of Thompson, Thames, and Mastin [d]. Recent modifications of the original technique are detailed in reference [e,j,n].

Experimental data sets which are particularly adapted to verification of transonic algorithms have been developed in recent times. In addition to the NACA 64A010 airfoil data used in this research (Johnson and Bachalo, reference [m]), the data set gathered by Seegmiller, Marvin, and Levy [u,v] for an 18% thick circular arc airfoil are to be recommended.

III. VECTORIZED NAVIER-STOKES EQUATIONS IN CURVILINEAR COORDINATES

Although many simplified versions of the Navier-Stokes equations have been successfully applied to various transonic problems, in general this very complex type of fluid flow can only be adequately described by the full set of equations. However, due to certain unsolved problems in turbulence modeling, constraints imposed by the numerical technique used, and the desirability of keeping the analysis as simple as possible, a number of assumptions and restrictions are required. For example: due to the impossibility or impracticability of using enough grid lines to resolve the small scale eddies in turbulent flow, the Reynolds averaged Navier-Stokes equations are used along with an algebraic eddy viscosity turbulence model. This chapter, then, will present the particular form of the Navier-Stokes equations, the equations of state, the constitutive equations, and other subsidiary relations which were chosen to model the transonic flow problems discussed in this paper.

A. Problem Description in Cartesian Coordinates

The governing differential equations used to model the transonic flow of air in this work are the two-dimensional, Reynolds averaged Navier-Stokes equations for a Newtonian fluid which obeys a Fourier heat conduction law: This system of four partial differential equations (continuity, x-momentum, y-momentum, and energy equations) can be expressed in the following vectorized, nondimensional form:

$$\frac{\partial q}{\partial t} + \frac{\partial}{\partial x} A(q) + \frac{\partial}{\partial y} B(q) = \frac{1}{Re} [\frac{\partial}{\partial x} D(q,q_x,q_y) + \frac{\partial}{\partial y} E(q,q_x,q_y)] \quad (3.1)$$

where

$$q \equiv \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ e \end{bmatrix}, \quad A(q) \equiv \begin{bmatrix} \rho u \\ \rho u^2 + P \\ \rho uv \\ (e+P)u \end{bmatrix}, \quad B(q) \equiv \begin{bmatrix} \rho v \\ \rho uv \\ \rho v^2 + P \\ (e+P)v \end{bmatrix} \qquad (3.2)$$

$$D(q,q_x,q_y) \equiv \begin{bmatrix} 0 \\ (\lambda+2\mu)u_x + \lambda v_y \\ \mu v_x + \mu u_y \\ (\lambda+2\mu)uu_x + \mu vv_x + \mu vu_y + \lambda uv_y + \frac{k}{Pr} T_x \end{bmatrix} \qquad (3.3)$$

$$E(q,q_x,q_y) \equiv \begin{bmatrix} 0 \\ \mu v_x + \mu u_y \\ \lambda u_x + (\lambda+2\mu)v_y \\ \mu uv_x + \lambda vu_x + (\lambda+2\mu)vv_y + \mu uu_y + \frac{k}{Pr} T_y \end{bmatrix} \qquad (3.4)$$

In addition, since air can be assumed to be a (thermally and calorically) perfect gas for the flow studies, the following nondimensional equations of state were used:

$$T = \gamma[\frac{e}{\rho} - \frac{1}{2}(u^2 + v^2)] \qquad (3.5)$$

$$P = (\gamma-1)[e - \frac{1}{2}\rho(u^2 + v^2)] \qquad (3.6)$$

The transport coefficients were obtained by assuming that the Prandtl number is effectively constant, that a Sutherland viscosity law is valid, and that Stokes law can be used. These relations are listed below in nondimensional form:

$$k = \mu \tag{3.7}$$

$$\mu = \left[1 + (\gamma-1)M^2S_1\right](\gamma-1)^{\frac{1}{2}}M\,\frac{T^{3/2}}{T+S_1} \tag{3.8}$$

$$\lambda = -\frac{2}{3}\,\mu \tag{3.9}$$

The nondimensional quantities appearing in equations 3.2-3.9 are defined in the List of Symbols following the Table of Contents; details of the method of nondimensionalization are contained in Appendix A.

An algebraic turbulence model can be incorporated into the above set of equations by multiplying the viscosity coefficient, μ , by the factor:

$$1 + \mu_T/\mu \tag{3.10}$$

except where it replaces the thermal conductivity, k , (equation 3.7); then the factor

$$1 + \frac{Pr}{Pr_T}\,\frac{\mu_T}{\mu} \tag{3.11}$$

must be used. The eddy viscosity, μ_T , and the turbulent Prandtl number, Pr_T , depend upon the turbulence model used. Due to its relative simplicity and its prior use in transonic flow calculations, the Baldwin-Lomax [a] algebraic turbulence model was used when required.

Of course, no problem is fully specified until the initial and boundary conditions are stated. However, since this research involved distinct problem types (i.e. couette flow and transonic flow over multi-element airfoils) and since the particular boundary and/or initial conditions used are rather intimately related to the numerical solution procedure, further discussion of this topic will be deferred.

B. Problem Description in Curvilinear Coordinates

The computational grid on which this set of partial differential equations are solved usually does not form a Cartesian coordinate system. Thus it is advantageous to re-express the problem in terms of more general curvilinear coordinates while retaining the strong conservation form of the Navier-Stokes equations. In the notation commonly chosen (see Steger [b], for example) the following coordinate transformation is defined:

$$\tau = t \tag{3.12}$$

$$\xi = \xi(x,y,t) \tag{3.13}$$

$$\eta = \eta(x,y,t) \tag{3.14}$$

Using this transformation,equations 3.1-3.4 can be written as:

$$\frac{\partial}{\partial\tau} q^* + \frac{\partial}{\partial\xi} A^*(q^*) + \frac{\partial}{\partial\eta} B^*(q^*) = \frac{1}{Re}\left[\frac{\partial}{\partial\xi} D^*(q^*,q^*_\xi,q^*_\eta) + \frac{\partial}{\partial\eta} E^*(q^*,q^*_\xi,q^*_\eta)\right] \tag{3.15}$$

where

$$q^* = \frac{1}{J} q \tag{3.16}$$

$$A^*(q^*) = \frac{1}{J}[\xi_t q + \xi_x A + \xi_y B] \tag{3.17}$$

$$B^*(q^*) = \frac{1}{J}[\eta_t q + \eta_x A + \eta_y B] \tag{3.18}$$

$$D^*(q^*,q^*_\xi,q^*_\eta) = \frac{1}{J}[\xi_x D + \xi_y E] \tag{3.19}$$

$$E^*(q^*,q^*_\xi,q^*_\eta) = \frac{1}{J}[\eta_x D + \eta_y E] \tag{3.20}$$

The important details of this transformation are contained in Appendix B. In that, for the most part, the remainder of this work will deal with this generalized vector equation, the superscript "*" will be suppressed except where needed for clarity.

The equations of state (3.5 and 3.6) and the transport coefficients (3.7,3.8, and 3.9) are not affected by a coordinate transformation, except that the dependent variables must now be interpreted as functions of ξ,η, and τ (i.e. $P(\xi,\eta,\tau) \equiv P(x(\xi,\eta,\tau),y(\xi,\eta,\tau),t(\tau)))$.

Since the numerical algorithm which was selected to solve equation 3.15 must treat cross-derivatives specially, it is necessary to split the viscous vectors D and E into vectors which contain only ξ- or only η- derivatives. Thus equation 3.15 is written as:

$$\frac{\partial}{\partial\tau} q + \frac{\partial}{\partial\xi} A(q) + \frac{\partial}{\partial\eta} B(q) = \frac{1}{Re}\{\frac{\partial}{\partial\xi}[D_1(q,q_\xi) + D_2(q,q_\eta)] + \frac{\partial}{\partial\eta}[E_1(q,q_\xi) + E_2(q,q_\eta)]\} \quad (3.21)$$

where

$$q = \frac{1}{J}\begin{bmatrix}\rho \\ \rho u \\ \rho v \\ e\end{bmatrix}, \quad A = \frac{1}{J}\begin{bmatrix}\rho U \\ \rho uU + \xi_x P \\ \rho vU + \xi_y P \\ (e+P)U-\xi_t P\end{bmatrix}, \quad B = \frac{1}{J}\begin{bmatrix}\rho V \\ \rho uV + \eta_x P \\ \rho vV + \eta_y P \\ (e+P)V-\eta_t P\end{bmatrix} \quad (3.22)$$

$$D_1 = \frac{1}{J}\begin{bmatrix}0 \\ a_1 u_\xi + a_2 v_\xi \\ a_2 u_\xi + a_3 v_\xi \\ a_1 uu_\xi + a_2(vu_\xi + uv_\xi) + a_3 vv_\xi + a_4 T_\xi\end{bmatrix} \quad (3.23)$$

$$D_2 = \frac{1}{J}\begin{bmatrix}0 \\ b_1 u_\eta + b_2 v_\eta \\ b_3 u_\eta + b_4 v_\eta \\ b_1 uu_\eta + b_3 vu_\eta + b_2 uv_\eta + b_4 vv_\eta + b_5 T_\eta\end{bmatrix} \quad (3.24)$$

$$E_1 = \frac{1}{J}\begin{bmatrix} 0 \\ b_1u_\xi + b_3v_\xi \\ b_2u_\xi + b_4v_\xi \\ b_1uu_\xi + b_2vu_\xi + b_3uv_\xi + b_4vv_\xi + b_5T_\xi \end{bmatrix} \qquad (3.25)$$

$$E_2 = \frac{1}{J}\begin{bmatrix} 0 \\ c_1u_\eta + c_2v_\eta \\ c_2u_\eta + c_3v_\eta \\ c_1uu_\eta + c_2(vu_\eta + uv_\eta) + c_3vv_\eta + c_4T_\eta \end{bmatrix} \qquad (3.26)$$

In equations 3.15-3.26, the transformation Jacobian, J , is defined as:

$$J = \xi_x\eta_y - \xi_y\eta_x \qquad (3.27)$$

The contravariant velocities, U and V, are given by:

$$U = \xi_t + \xi_xu + \xi_yv \qquad (3.28)$$

$$V = \eta_t + \eta_xu + \eta_yv \qquad (3.29)$$

Also, the viscous coefficients are defined as:

$$\begin{aligned} a_1 &= (\lambda+2\mu)\xi_x^2 + \mu\xi_y^2 \\ a_2 &= (\lambda+\mu)\xi_x\xi_y \\ a_3 &= \mu\xi_x^2 + (\lambda+2\mu)\xi_y^2 \\ a_4 &= \frac{k}{Pr}(\xi_x^2 + \xi_y^2) \end{aligned} \qquad (3.30a)$$

$$b_1 = (\lambda+2\mu)\xi_x\eta_x + \mu\xi_y\eta_y$$

$$b_2 = \lambda\xi_x\eta_y + \mu\xi_y\eta_x$$

$$b_3 = \mu\xi_x\eta_y + \lambda\xi_y\eta_x$$

$$b_4 = \mu\xi_x\eta_x + (\lambda+2\mu)\xi_y\eta_y$$

$$b_5 = \frac{k}{Pr}(\xi_x\eta_x + \xi_y\eta_y) \tag{3.30b}$$

$$c_1 = (\lambda+2\mu)\eta_x^2 + \mu\eta_y^2$$

$$c_2 = (\lambda+\mu)\eta_x\eta_y$$

$$c_3 = \mu\eta_x^2 + (\lambda+2\mu)\eta_y^2$$

$$c_4 = \frac{k}{Pr}(\eta_x^2 + \eta_y^2) \tag{3.30c}$$

Note that except for the doubling of the number of viscous vectors, the curvilinear equations (3.21-3.26) are not much more complex than the Cartesian equations (3.1-3.4). This can be very useful in that some relations developed in Cartesian coordinates may be directly converted to curvilinear coordinates by maintaining the proper correspondence of terms.

Since none of the problems investigated in this research involved the use of a time-dependent coordinate system, henceforth, the derivatives ξ_t and η_t will be taken to be zero.

As was the case with the equations of state and the transport coefficients equations, any boundary or initial conditions which do not involve spatial derivatives will remain invariant in form. However, there are two common types of boundary conditions which are derivative relations. One of these is the "adiabatic wall" assumption, or more generally, a specification of the heat flux, $\dot{q}''$, through some portion of the boundary of the flow domain. In nondimensional form this condition can be expressed as

$$\dot{q}'' = - \frac{k}{RePr} \nabla_N T \tag{3.31}$$

which follows from the Fourier heat conduction law and the nondimensionalization (see Appendix A). The symbol "∇_N" indicates a derivative normal to the boundary surface. In curvilinear coordinates this relation becomes:

$$\dot{q}'' = \frac{k}{RePr} (\alpha T_\eta + \beta T_\xi)/\sqrt{\alpha} \tag{3.32a}$$

where

$$\alpha = \eta_x^2 + \eta_y^2 \quad , \quad \beta = \xi_x \eta_x + \xi_y \eta_y \tag{3.32b}$$

The other derivative boundary condition is one which arises from the fact that whereas the analytical formulation of a problem only allows three of the four dependent variables to be specified at a solid boundary, the numerical formulation requires all four to be specified. To maintain some degree of consistency with the analytical problem, this "extra" boundary condition should take the form of an extrapolation from the interior of the flow field. Thus, one of the most commonly used additional boundary conditions is to specify the normal pressure derivative in accordance with either boundary layer theory or the momentum equations. Using the second approach, the momentum equations and continuity equation may be combined to form:

$$\begin{aligned}\nabla_N P &+ [(\eta_x u_t + \eta_y v_t) + (\eta_x u_\xi + \eta_y v_\xi)U + (\eta_x u_\eta + \eta_y v_\eta)V](\frac{\gamma}{\gamma-1})(\frac{1}{\sqrt{\alpha}\,T})P = \\ \frac{J}{\sqrt{\alpha}\,Re} &\{[\eta_x(a_1' u_\xi + a_2' v_\xi)_\xi + \eta_y(a_2' u_\xi + a_3' v_\xi)_\xi] + \\ &[\eta_x(b_1' u_\eta + b_2' v_\eta)_\xi + \eta_y(b_3' u_\eta + b_4' v_\eta)_\xi] + \\ &[\eta_x(b_1' u_\xi + b_3' v_\xi)_\eta + \eta_y(b_2' u_\xi + b_4' v_\xi)_\eta] + \\ &[\eta_x(c_1' u_\eta + c_2' v_\eta)_\eta + \eta_y(c_2' u_\eta + c_3' v_\eta)_\eta]\}\end{aligned} \qquad (3.33)$$

where

$$\nabla_N P = (\alpha P_\eta + \beta P_\xi)/\sqrt{\alpha} \qquad (3.34)$$

and the primed coefficients are the viscous coefficients (3.30) divided by the Jacobian "J". Further details on the development of relation 3.33 are contained in Appendix C.

C. Curvilinear Coordinate System Generation

The use of a curvilinear coordinate system is desirable because, among other reasons, it allows easy application of boundary conditions since each boundary can be made to coincide with a coordinate line; it can concentrate coordinate lines in one region, place a minimum number in another region, and smoothly transition from one to the other; and it allows numerical algorithms to be relatively independent of the particular geometry of a problem. However, for curvilinear coordinates to be truly useful, one must utilize a method of coordinate generation which, in addition to providing the advantages of the previous sentence, must be relatively simple to implement, must generate "smooth" (i.e. second order and higher order derivatives small in some sense), single valued coordinates, and, at least for some of the applications considered in this paper, must be capable of handling multiple bodies easily. The method chosen for this work is the "boundary-fitted coordinate" algorithm due to Thompson, Thames, and Mastin [d,e], which achieves its purpose by solving a Poisson equation in the curvilinear coordinates ξ and η.

Briefly, this technique requires that the coordinates satisfy the elliptic system:

$$\xi_{xx} + \xi_{yy} = - \frac{\alpha_c}{J_c^2} P_c(\xi,\eta) \qquad (3.35)$$

$$\eta_{xx} + \eta_{yy} = - \frac{\gamma_c}{J_c^2} Q_c(\xi,\eta) \qquad (3.36)$$

where normally the boundary conditions to this system are specified so that the body(ies) lie along certain of the η coordinate lines with some desired distribution of ξ lines terminating on them. The functions P_c and Q_c are specified so that the coordinate spacing in the interior of the field is an approximation of the one desired. In order to obtain the calculation advantages of solving these equations on a curvilinear coordinate system, they are transformed so that it is the x and y distribution that is determined. That is, the following elliptic system is actually solved:

$$\alpha_c x_{\xi\xi} - 2\beta_c x_{\xi\eta} + \gamma_c x_{\eta\eta} = -(\alpha_c x_\xi P_c + \gamma_c x_\eta Q_c) \qquad (3.37)$$

$$\alpha_c y_{\xi\xi} - 2\beta_c y_{\xi\eta} + \gamma_c y_{\eta\eta} = -(\alpha_c y_\xi P_c + \gamma_c y_\eta Q_c) \qquad (3.38)$$

In practice, the above system is discretized and solved by standard methods of numerical analysis. Further details on the development and use of this method of coordinate generation and the definition of the parameters α_c, β_c, γ_c, and J_c are contained in Appendix D.

IV. NUMERICAL FORMULATION

It is an unfortunate fact that mathematical analysis of nonlinear partial differential equations, such as the Navier-Stokes equations, can not currently provide anything like a closed form solution to these equations for general boundary conditions. Thus, though itself more of an art than a science, methods of numerical analysis must of necessity be applied to these problems if more than a qualitative description of the flow is desired. One of the more common numerical methods used to solve the Navier-Stokes equations is to approximate this set of partial differential equations by an equivalent set of difference equations in such a way that they are consistent with these differential equations. In addition, their solution algorithm must produce a sequence of intermediate iterative values which converge to the actual solution to some estimatable degree of accuracy. From a practical standpoint, the iterative algorithm should possess the property of a rapid rate of convergence, both numerically (small number of iterations) and computationally (small number of operations). Thus, one of the most important aspects of this research was the selection and development of the numerical algorithm used to solve the Navier-Stokes equations presented in the previous chapter.

The difference scheme chosen for this work was the approximate factorization algorithm due to Beam and Warming [c,f]. In the form used in this research, this algorithm is an implicit, second-order accurate in time, non-iterative (in the sense that each time-step is calculated once only), unconditionally stable (in the linear approximation), three-time-level scheme. Its "delta" formulation insures that, although the cross-derivative terms are treated explicitly, the scheme remains second-order accurate and unconditionally stable; requires only two-time-levels of storage; and produces a steady-state solution which is independent of the time-step size. The most attractive feature of the algorithm is that, due to its spatial factorization, it forms an ADI type of scheme in which each sweep involves the inversion of a block-tridiagonal matrix. In principle this approximate factorization algorithm promises to solve complex, two-dimensional fluid flow problems on current high-speed scientific computers within an economically reasonable time (i.e., ten to thirty minutes).

The remainder of this chapter will attempt to put the assertions of the previous paragraph into a more concrete form. First the delta formulation is presented along with an indication of the method of spatial differencing. Then the techniques employed to impose the boundary conditions are detailed. Lastly a brief discussion of the problem of forming the geometric coefficients is included.

A. Beam and Warming Approximate Factorization Algorithm

The most general, consistent, three-time-level, linear expression relating the conservation variable, q , and its time-derivative, q_t , is [f]:

$$\Delta q^n = \frac{\Delta t}{1+\phi}\left[\theta \frac{\partial}{\partial t}(\Delta q^n) + \frac{\partial}{\partial t} q^n + \psi \frac{\partial}{\partial t}(\Delta q^{n-1})\right] + \frac{\phi}{1+\phi}\Delta q^{n-1} \qquad (4.1)$$

with truncation error of:

$$(\psi - \phi + \theta - \frac{1}{2})\Delta t^2 q_{tt}^n + O(\Delta t^3) \qquad (4.2)$$

where "Δ" is the usual forward difference operator, and θ,ϕ,ψ are arbitrary real constants. Note that since τ = t, they will be used interchangeably in this paper. Henceforth, the parameter ψ will always be taken to be zero and, to maintain second-order accuracy, the following relation is established between ϕ and θ:

$$\theta = \phi + \frac{1}{2} \tag{4.3}$$

Now the time-derivatives of the conservation vector are evaluated from equation (3.21) and substituted into equation (4.1) with the result:

$$\Delta q^n = \frac{\theta}{1+\phi}\Delta t\,[\frac{\partial}{\partial\xi}(-\Delta A + \frac{\Delta D_1+\Delta D_2}{Re})^n + \frac{\partial}{\partial\eta}(-\Delta B + \frac{\Delta E_1+\Delta E_2}{Re})^n]$$

$$+ \frac{\Delta t}{1+\phi}[\frac{\partial}{\partial\xi}(-A + \frac{D_1+D_2}{Re})^n + \frac{\partial}{\partial\eta}(-B + \frac{E_1+E_2}{Re})^n] + \frac{\phi}{1+\phi}\Delta q^{n-1} \tag{4.4}$$

This equation is not suitable for direct use since the time-differenced vectors on the right-hand-side involve nonlinear functions of q^{n+1}. However, due to special properties of the flux vectors, (A,B,D, and E), under a suitable assumption on the transport coefficients, these differenced vectors can be expressed in terms of their Jacobian matrices as follows:

$$\Delta A^n = \frac{\partial}{\partial q}A^n\,\Delta q^n = P^n\Delta q^n \tag{4.5a}$$

$$\Delta B^n = \frac{\partial}{\partial q}B^n\,\Delta q^n = Q^n\Delta q^n \tag{4.5b}$$

$$\Delta D_1^n = [\frac{\partial}{\partial q_\xi}D_1^n\ \ \Delta q^n]_\xi = (W^n\Delta q^n)_\xi \tag{4.5c}$$

$$\Delta D_2^n = [\frac{\partial}{\partial q_\eta}D_2^n\ \ \Delta q^n]_\eta = (X^n\Delta q^{n-1})_\eta \tag{4.5d}$$

$$\Delta E_1^n = [\frac{\partial}{\partial q_\xi} E_1^n \ \Delta q^n]_\xi = (Y^n \Delta q^{n-1})_\xi \tag{4.5e}$$

$$\Delta E_2^n = [\frac{\partial}{\partial q_\eta} E_2^n \ \Delta q^n]_\eta = (Z^n \Delta q^n)_\xi \tag{4.5f}$$

where the cross-derivative matrices (ΔD_2 and ΔE_1) have been lagged in time in order to facilitate the factorization described later. These expressions are exact to the order of Δt^2, and thus their use in equation (4.4) will not degrade the accuracy of the algorithm. Although not strictly necessary to the development, it is convenient to also use the following exact relationships between the flux vectors and their Jacobian matrices:

$$D_1 = Wq_\xi \ , \ D_2 = Xq_\eta$$

$$E_1 = Yq_\xi \ , \ E_2 = Zq_\eta \tag{4.6}$$

and

$$Wq = Xq = Yq = Zq = 0 \tag{4.7}$$

A detailed derivation of these equations (4.5, 4.6, and 4.7) is contained in Appendix E; the explicit form of the Jacobian matrices (P, Q, W, X, Y, and Z) are contained in Appendix F.

Using relations (4.5) and (4.6), equation (4.4) may be re-expressed as:

$$\{I + \frac{\theta\Delta t}{1+\phi}[(\frac{\partial}{\partial\xi}P - \frac{1}{Re}\frac{\partial^2}{\partial\xi^2}W) + (\frac{\partial}{\partial\eta}Q - \frac{1}{Re}\frac{\partial^2}{\partial\eta^2}Z)]\}^n \Delta q^n =$$

$$\frac{\Delta t}{1+\phi}\{\frac{\partial}{\partial\xi}[- A + \frac{1}{Re}(Wq_\xi + Xq_\eta)] + \frac{\partial}{\partial\eta}[- B + \frac{1}{Re}(Yq_\xi + Zq_\eta)]\}^n$$

$$+ \frac{\theta\Delta t}{1+\phi}\frac{1}{Re}\frac{\partial^2}{\partial\xi\partial\eta}[(X+Y)^n \ \Delta q^{n-1}] + \frac{\phi}{1+\phi}\Delta q^{n-1} \tag{4.8}$$

where "I" is the four-by-four identity matrix and the bracketed term on the left of Δq^n is understood to be a derivative operator. Since equation (4.8) is of order Δt^3 and since Δq^n is itself of order Δt, this equation can be "approximately factored" into the following set of equations with no loss of accuracy:

$$\{I + \frac{\theta\Delta t}{1+\phi}[\frac{\partial}{\partial\xi} P - \frac{1}{Re}\frac{\partial^2}{\partial\xi^2} W]\}^n \ \overline{\Delta q^n} = \text{RHS}(4.8) \qquad (4.9a)$$

$$\{I + \frac{\theta\Delta t}{1+\phi}[\frac{\partial}{\partial\eta} Q - \frac{1}{Re}\frac{\partial^2}{\partial\eta^2} Z]\}^n \ \Delta q^n = \overline{\Delta q^n} \qquad (4.9b)$$

$$q^{n+1} = q^n + \Delta q^n \qquad (4.9c)$$

where $\overline{\Delta q^n}$ is an intermediate value and RHS (4.8) represents the right-hand-side of equation (4.8). Although equation set (4.9) represents the basic approximate factorization algorithm, the spatial derivatives must also be discretized for it to be of practical utility. Using the central difference approximations developed in Appendix G, equation (4.9) can be put in the finite difference form:

$$\begin{aligned}
\text{RHS}^n_{i,j} = {} & \frac{1}{4\theta}[X^n_{i+1,j}(q^n_{i+1,j+1} - q^n_{i+1,j-1}) - X^n_{i-1,j}(q^n_{i-1,j+1} - q^n_{i-1,j-1}) \\
& + \quad Y^n_{i,j+1}(q^n_{i+1,j+1} - q^n_{i-1,j+1}) - Y^n_{i,j-1}(q^n_{i+1,j-1} - q^n_{i-1,j-1})] \\
& + \frac{1}{2}[(R^n\Delta q^{n-1})_{i+1,j+1} - (R^n\Delta q^{n-1})_{i+1,j-1} \\
& \qquad - (R^n\Delta q^{n-1})_{i-1,j+1} + (R^n\Delta q^{n-1})_{i-1,j-1}] \\
& + \frac{1}{2\theta}[W^n_{i,j}(q^n_{i+1,j} + q^n_{i-1,j}) - (W^n_{i+1,j} + W^n_{i-1,j})q^n_{i,j}] \\
& + \frac{1}{2\theta}[Z^n_{i,j}(q^n_{i,j+1} + q^n_{i,j-1}) - (Z^n_{i,j+1} + Z^n_{i,j-1})q^n_{i,j}] \\
& - \frac{1}{\theta}[(A^n_{i+1,j} - A^n_{i-1,j}) + (B^n_{i,j+1} - B^n_{i,j-1})] + \frac{\phi}{1+\phi}\Delta q^{n-1}_{i,j} \qquad (4.10a)
\end{aligned}$$

$$-(P+W)^n_{i-1,j}\,\overline{\Delta q^n_{i-1,j}} + (I+2W^n_{i,j})\overline{\Delta q^n_{i,j}} + (P-W)^n_{i+1,j}\,\overline{\Delta q^n_{i+1,j}} = RHS^n_{i,j} \tag{4.10b}$$

$$-(Q+Z)^n_{i,j-1}\,\Delta q^n_{i,j-1} + (I+2Z^n_{i,j})\Delta q^n_{i,j} + (Q-Z)^n_{i,j+1}\,\Delta q^n_{i,j+1} = \overline{\Delta q^n_{i,j}} \tag{4.10c}$$

$$q^{n+1}_{i,j} = q^n_{i,j} + \Delta q^n_{i,j} \tag{4.10d}$$

where $R = \frac{1}{2}(X+Y)$ and the coefficients $\frac{\theta\Delta t}{1+\phi}\,\frac{1}{Re}$ and $\frac{1}{2}\,\frac{\theta\Delta t}{1+\phi}$ have been absorbed by the matrices W, X, Y, Z and A, B, P, Q, respectively.

This final set of difference equations (4.10) also defines the basic solution algorithm. First the explicit vector RHS is calculated over the interior of the field. Then each η-line is swept and a block-tridiagonal matrix inverted to determine the intermediate values $\overline{\Delta q^n_{i,j}}$. Similarly, each ξ-line is swept and the resulting block-tridiagonal matrix inverted, thus determining the difference conservation vector $\Delta q^n_{i,j}$. Lastly the conservation vector $q^n_{i,j}$ is determined from equation (4.10d). In essence, then, this method is an ADI type of implicit algorithm in which just one iteration per time-step is performed due to the direct inversion of the block-tridiagonal matrices. For completeness, the block-tridiagonal inversion algorithm is included in Appendix H (see also Steger [b], his Appendix 1).

B. Initial Conditions

Even though it is the primary objective of most fluid flow analyses, including this one, to achieve a solution which is independent of the initial state of the fluid, thus making the boundary conditions all important, the eminent difficulties involved in getting a numerical simulation started at high Reynolds and Mach numbers results in the initial conditions taking on a computational importance which can not be neglected. Indeed, the starting problem often dictates, at least early on, the boundary conditions to be used and often modifies the basic algorithm itself. In the research reported in this volume, three different sets of initial conditions and their accompanying modifications to the boundary conditions and algorithm were tried.

The simplest initial condition, at least from the standpoint of initial formulation, is the so-called impulsive start. In this case the initial properties of the flow field are the same as the free-stream properties except for the velocities on any solid boundaries. This initial condition specification has two major disadvantages, both due to the usually large velocity jump near a body. Most deleterious is the fact that the algorithm must be altered so as to rapidly diffuse the strong velocity gradient at the body. One such alteration is to effectively greatly reduce the Reynolds number in the viscous Jacobian matrices W and Z of equations (4.10b and c)[g]; this has no effect on the steady-state solution, but can drastically affect the transient solutions. The other shortcoming of this type of start is that intense compression and rarefacation waves form, which, due to the inexact nature of the boundary conditions used in numerical simulations, can cause long term distortion of the flow field [h].

A second class of initial conditions is the same as the impulsive start except that the velocities at solid bodies are gradually varied from free-stream to the desired final values. This has been termed as a "penetration" initial condition, since, physically, this velocity boundary condition implies that fluid must be sucked into and/or ejected from the body. Initial conditions of this sort do not require a change in the character of the set of equations (4.10), however, they do require that they be flexible enough to handle suction and/or blowing from the boundaries. Similar to the impulsive start case, there are two major disadvantages. One is that for a closed body embedded in the flow field, such as an airfoil section, it is extremely difficult to satisfy global conservation laws; thus the early transient solutions, even after the desired boundary velocities are obtained, may be considerably different from expectations. Related to the first potential difficulty, the other problem with this class of initial conditions is that if suction and/or blowing is not a natural boundary condition for a problem, then, again, the transient solutions for the desired boundary conditions can not be obtained.

The last type of intial condition to be considered here is what will be called a "gradual" start. This case is, in a sense, the reverse of the impulsive start in that the initial properties of the flow field and the boundaries are uniformly the same as the desired boundary condition on the solid body boundary. In most cases this means that the velocity field is everywhere zero. The start-up is achieved by uniformly accelerating the flow field everywhere except the "fixed" body boundary. In practice, the easiest way to accomplish this is to add a body force term to the original equation set (3.1) and adjust the resulting difference equation set (4.10) accordingly. Details on how this modification is implemented are in Appendix I.

C. Boundary Conditions

Since the boundary conditions ultimately determine the form of the flow field, some care must be taken in formulating their numerical equivalent. This task is especially complicated by problems which have no counterpart in the analogous analytical formulation. It is not currently possible to directly impose an "infinity" boundary condition on the numerical problem; thus, certain boundaries must be designated as inflow and outflow boundaries and their boundary conditions determined so as to mimic the actual infinity boundary conditions. As mentioned previously (see discussion prior to equation (3.33)), an "extra" boundary condition is required on solid boundaries which must be determined from the solution and not determine the solution. Also, re-entrant boundary conditions must be specified so that these computational boundaries are effectively transparent to the flow field solution.

Inflow boundary conditions are the easiest to formulate since they are normally placed far enough upstream of any solid bodies so that the flow in their vicinity may be assumed to be unaffected by their pressence. Thus, the inflow boundary conditions are completely determined by the corresponding infinity boundary conditions.

Outflow boundary conditions, on the otherhand, are much harder to formulate, at least conceptually. Similar to the inflow boundary, this boundary is normally far downstream of any bodies, but, opposite to the inflow boundary, it must not affect the flow upstream of it. That is, this boundary must allow disturbances to pass "through" it without reflecting them. One common method of at least partially accomplishing this is to determine these boundary values by extrapolation from the flow field solution, or, equivalently, by specifying a Neumann boundary condition. The technique chosen for this research was to use second order extrapolation along the coordinate line which crosses the outflow boundary. For example, if the Ith ξ-line is an outflow boundary, then

$$\Delta q^{n}_{I,j} = 2\Delta q^{n}_{I-1,j} - \Delta q^{n}_{I-2,j} \tag{4.11a}$$

$$q^{n+1}_{I,j} = q^{n}_{I,j} + \Delta q^{n}_{I,j} \tag{4.11b}$$

Re-entrant boundaries can arise either due to the analytical formulation (i.e., Couette flow) or due to the choice of coordinate system (i.e., polar coordinates). In either case a choice must be made between treating these boundaries approximately, thus keeping the algorithm simple, or exactly. Typical of the approximate technique [b] is to lag the values on this boundary for the solution of equation set (4.10) (henceforth termed implicit boundary conditions), then to calculate these boundary values from an average of extrapolates of corresponding points on the re-entrant boundary after the interior field values are determined (henc forth termed explicit boundary conditions). In symbols this method could be implemented as:

$$\Delta q^{n}_{I,j} = \Delta q^{n-1}_{I,j} \qquad \text{(implicit)} \tag{4.12a}$$

$$\Delta q^{n}_{I,j} = -\frac{1}{2}[\Delta q^{n}_{I,j+2} - 2(\Delta q^{n}_{I,j+1} + \Delta q^{n}_{I,J+1}) + \Delta q^{n}_{I,J+2}] \qquad \text{(explicit)} \tag{4.12b}$$

where the Ith ξ-line is a re-entrant boundary and j and J are the indexes of the re-entrant η-line.

The approach taken by this author was the more exact, but much more complicated, technique of continuing the algorithm across the re-entrant boundary. For cases in which the coordinate lines that crossed the re-entrant boundary were periodic (i.e., polar coordinates), this only affected the inversion algorithm since the matrices were periodic block-triangular instead of just block-tridiagonal. In all other cases, this "just" means taking care that a block-tridiagonal matrix is formed in crossing the re-entrant boundary. In practice, the first, inexact, method is preferable since it keeps the basic algorithm simple and much more problem independent.

Body or wall boundary conditions are the most difficult type to formulate, primarily because of their importance, both analytically and numerically, in defining the flow. Unfortunately, this task is still mostly an art since a universally accepted method for treating the "missing" boundary condition on density does not exist. There is, of course, no problem in specifying the velocity boundary conditions in viscous flow; these are always assumed to be known quantities (usually identically zero). Also, the temperature boundary condition presents little trouble if it too is assumed to be specified (isothermal being the easiest of these).

It will be convenient to consider the implicit and explicit boundary conditions separately. For the implicit case, if the first η-line is a body boundary, for example, then the tridiagnoal element $-(Q^n_{i,1} + Z^n_{i,1})\Delta q^n_{i,1}$ must be re-expressed in terms of known quantities and/or other differenced conservation vectors on the same ξ-line. In general, this means determining the coefficient matrices H and K and the vector L in the linear combination:

$$\Delta q^n_{i,1} = H\Delta q^n_{i,2} + K\Delta q^n_{i,3} + L \tag{4.13}$$

which preserves a block-tridiagonal system of equations when substituted into equation (4.10c). These matrices were determined in this research by extrapolating the density along the ξ-line and, in those cases in which the temperature was not explicitly specified, by lagging the temperature in time. The details of this technique are outlined in Appendix J; for other successful methods of evaluating these coefficient matrices, see Beam and Warming [c] or Steger [b].

After the solution of equation set (4.10) the advanced boundary values on the body need to be determined. If a gradient temperature boundary condition is specified by equation (3.32), then a straightforward application of the appropriate difference approximations results in the following periodic tri-diagonal system of equation:

$$-\left(\frac{\beta}{\alpha}\right)_{i,1} T_{i-1,1} - 3T_{i,1} + \left(\frac{\beta}{\alpha}\right)_{i,1} T_{i+1,1} = T_{i,3} - 4T_{i,2} - 2RePr\left(\frac{1}{k\sqrt{\alpha}}\right)_{i,1} \dot{q}''_i \qquad (4.14)$$

To determine the surface density, the pressure gradient equation (3.33) was used. It also forms a slightly more complicated periodic tri-diagonal system:

$$-\left(\frac{\beta}{\alpha}\right)_{i,1} P_{i-1,1} + (\pi_1-3)P_{i,1} + \left(\frac{\beta}{\alpha}\right)_{i,1} P_{i+1,1} = P_{i,3} - 4P_{i,2} + \pi_2 \qquad (4.15)$$

where the expansion of the π_1 and π_2 terms is considered in Appendix C. For illustrative purposes, the body has been assumed to be coincident with the first η-line in both equations (4.14) and (4.15). Note that for bodies whose surfaces do not form continuous segments in the transformed coordinates, it was found to be advantageous to determine the value at the (re-entrant) end-points by the average of extrapolates method outlined previously and then solving the resultant pure tri-diagonal system of equations for each segment.

D. Artificial Viscosity

It is a well-known fact that, while the approximation of spatial derivatives by central differences, such as was done in developing equation (4.10), have many desirable properties, such as accuracy of derivative representation and simplicity of use, they make the algorithms containing them prone to nonlinear instabilities. Thus, in any fluid problem in which a region of very rapid change occurs relative to the corresponding coordinate density, some means of diffusing the resultant high frequency components of the solution must be appended to the basic solution scheme. The technique used in this research was to alter the basic set of equations (4.10) by adding to the explicit equation (4.10a) the difference expression:

$$\begin{aligned} -\frac{\varepsilon}{J_{i,j}}\big[&(q_{i+2,j} - 4q_{i+1,j} + 6q_{i,j} - 4q_{i-1,j} + q_{i-2,j}) \\ &+ (q_{i,j+2} - 4q_{i,j+1} + 6q_{i,j} - 4q_{i,j-1} + q_{i,j-2})\big]^n \end{aligned} \qquad (4.16a)$$

everywhere except for points near boundaries where

$$\frac{\varepsilon}{J_{i,j}}[q_{i+1,j} - 2q_{i,j} + q_{i-1,j}) + (q_{i,j+1} - 2q_{i,j} + q_{i,j-1})]^n \qquad (4.16b)$$

is used. Note that the symbol q is used in the sense of equation (3.2). It can be readily shown that, provided the coordinate variation is "smooth", equation (4.16a) adds a term proportional to the fourth power of the local grid spacing and equation (4.16b) adds a term proportional to the square of this spacing. Thus, if ϵ is proportional to Δt, for a dense enough mesh, these artificial diffusion terms should not seriously degrade the accuracy of the basic method except in those regions of high gradients (relative to grid spacing) where its accuracy is suspect anyway. A linear stability analysis places an upper bound on the value of ε of 1/12. Pulliam and Steger [g], from whom this dissipative technique was derived, also make use of an implicit artificial viscosity appended to the W and Q matrices of equations (4.10b and c); however, this author found its use to be required only for impulsive type starts.

E. Coordinate Derivatives

There is a serious potential problem with developing any numerical algorithm for solving the Navier-Stokes equations in conservative form. This can be descerned in equation (4.10a) by considering a region of flow in which the density, velocities, and energy can be taken to be constant. If proper consideration of the properties of the flux Jacobian matrices (equation (4.7)) is taken, then the first element of the vector $\mathrm{RHS}^n_{i,j}$ is proportional to:

$$\begin{aligned} &u\{[(y_\eta)_{i+1,j} - (y_\eta)_{i-1,j}]-[(y_\xi)_{i,j+1} - (y_\xi)_{i,j-1}]\} \\ &-v\{[(x_\eta)_{i+1,j} - (x_\eta)_{i-1,j}]-[(x_\xi)_{i,j+1} - (x_\xi)_{i,j-1}]\} \qquad (4.17) \end{aligned}$$

Thus, since similar relations develop for the other elements of the vector RHS of equation (4.10a), if this algorithm is not to have spurious sources and sinks, the quantities in braces in expression (4.17) must be identically zero. In two dimensions, as is considered here, this can be assured by evaluating the coordinate derivatives (x_ξ, x_η, y_ξ, y_η) by the same difference scheme used to difference the flux vectors A and B. As is pointed out in reference [b], this should be an over-riding concern, even if the resulting differences are poor representations of the true derivatives (i.e., analytically generated coordinates should have their derivatives evaluated by the appropriate difference and not by direct differentiation).

V. COMPUTATIONAL RESULTS

The types of flows modeled in this research can be divided into two classes on the basis of geometry. Couette flow, the first class of flows studied, was chosen primarily because it was the simpler of the two test cases that Beam and Warming [c] used to verify their original formulation of their approximate factorization algorithm. Thus, it was convenient to use this flow, both as a means of debugging the computer code form of the algorithm presented in the previous chapter, and as a means of verifying that the algorithm and computer code were consistent with the results of Beam and Warming. The other class of flow geometry, that of a NACA 64A010 airfoil section imbedded in an other wise uniform stream of air, comprises the major thrust of this work. The particular airfoil section studied was selected on the basis of the number of experimental and numerical papers currently being published on transonic flow about NACA 64A010 airfoil sections at various angles of attack [b,j,k,l,m]. Also, the parameters of the flow were chosen to be representative of the cases studies in the references; thus, the Mach number ($M=0.8$), Reynolds number ($Re=2,000,000$), Prandtl numbers ($Pr=0.72$ and $Pr_T=0.9$), ratio of specific heats ($\gamma=1.4$), and angles of attack ($\alpha=0°,2°,3.5°$) were all set so as to facilitate comparison of results. In all of the cases discussed here, the three-point-backward version of the basic scheme was used (i.e.,$\phi=\frac{1}{2}$).

A. Couette Flow

Since the main purpose of this flow calculation was to reproduce the results of reference [c], the coordinate system and flow parameters were copied directly. The computational domain was composed of a six by eleven (6 ξ-lines and 11 η-lines) rectangular grid with "movable" walls bounding the top and bottom surfaces (η=1 and η=11) and with the left and right boundaries being re-entrant boundaries (i.e., ξ=1 and ξ=5 are actually the same ξ-line). The flow parameters given by Beam and Warming are the Mach number (M=0.09), and the Reynolds number (Re=6.2) based on distance between the opposing walls; from their discussion, it was apparent that the working fluid used was atmospheric air, so the Prandtl number (Pr=0.72) and specific heat ratio (γ=1.4) were set at typical values.

In that the coordinate derivatives ($\xi_x,\xi_y,\eta_x,\eta_y$) are differenced by the algorithm used in this research irregardless of the state of the fluid or boundaries, it was always felt to be important that so-called "free-stream" trials be made in each case. For the Couette flow, this meant running a case with the fluid and walls initially in a uniformly quiescent state, and a case with both the fluid and the walls moving, in the plane of the walls, with an initial uniform velocity. In both cases the algorithm maintained the initial state for arbitrary lengths of time and for arbitrary $\Delta t < 1$.

For the case of developing Couette flow, in which one wall is held fast,the fluid is initially at rest, and one wall is impulsively brought up to the nondimensional velocity of one. The basic test case was that for a time step of Δt=0.0116, which roughly corresponds to a Courant number of one. The coded approximate factorization scheme reproduced Beam and Warming's results exactly (possibly better than exactly) for this case, reproducing their conclusion that this method is excellent for determining transient solutions if the Courant number is less than unity.

Similar cases were run for Courant numbers of ten and one-hundred ($\Delta t=0.08$ and 0.8 respectively) with excellent steady-state results and increasingly poorer transient results - again the same experience as reported in reference [c]. It is this author's opinion that these results establish at least the consistency of this formulation of the general approximate factorization scheme.

B. NACA 64A010 Airfoil Section

The coordinate system chosen for use with an airfoil with a relatively pointed trailing edge, such as this one, has proven to be of some importance in solving the numerical equivalent of the Navier-Stokes equations for the flow about the airfoil. There seems to be a consensus of opinion that the so-called "C" or "wake" coordinate system, shown schematically in Figure 1, is most appropriate to a problem of this type [b,j]. Refering to this figure, it is seen that in this type of coordinate transformation, the lines of constant η form C's about the airfoil, with the inner-most line collapsing onto the airfoil surface and onto itself to the right of the airfoil. While the lines of constant ξ extend from the outer η-line ($\eta=J$) to the inner η-line ($\eta=1$), either terminating on the body or on the "cut" (the collapsed inner η-line). If this coordinate system is to be useful, then care must be taken that the ξ-lines which terminate on the cut pair-up with ξ-lines on the other side of the cut so that they may be considered to be a single line running from the upper portion to the lower portion of the outer η-line. Looking now at the tranformed field, it is apparent that the left and right boundaries (labeled $\Gamma^*_{3,L}$ and $\Gamma^*_{3,U}$) will always be outflow boundaries, that the upper boundary (Γ^*_2) will be totally an inflow boundary for 0° angle of attack and partially an outflow boundary otherwise, and that the portions of the lower boundary labeled Γ_4^* and Γ_5^* are re-entrant to each other. Although not indicated in the figure, this coordinate system formed a grid with 113 ξ-lines and 51 η-lines; 20 of these η-lines being concentrated within .05% to 2% of a chord length from either the airfoil or the cut.

As discussed in the Couette flow case, free-stream trials were felt to be an important test of both the solution algorithm and the coordinate system. For this coordinate system, runs were made for flow field initial velocities of either zero or free-stream (including the body boundary, that is, the flow was forced to "penetrate" the airfoil). In both cases angles of attack were set at either 0° or 2°, and the time step at $\Delta t = 0.01$. Free-stream conditions were maintained for extended periods of time ($t > 1$ in all cases), within the truncation error limits, provided that the cautions of section IV.E were followed. That is, if the coordinate derivatives and the algorithm derivatives were not <u>all</u> of the same type (i.e., central differences), then free-stream, and indeed a stable solution, could not be maintained. In particular, both tests of fourth-order differencing of the trailing edge coordinate derivatives and of the convective terms in the explicit portion of the algorithm (see equation (4.10a)) proved to be destabilizing.

The flow results obtained for the NACA 64A010 airfoil can be divided into several classifications based upon the method of starting the solution process (impulsive, penetration, or gradual), the turbulence model (laminar or algebraic Baldwin and Lomax model), and the angle of attack (0°, 2°, or 3.5°). Initial cases were all run as laminar flows at zero degrees angle of attack, thus only being distinguished by the type of start utilized. Later cases were all started and/or changed gradually and employed the algebraic turbulence model after an initial laminar starting period; thus these cases are distinguished only by the angle of attack chosen.

One of the first starting techniques tried was the impulsive start method. Similar to the results of Steger and Bailey [n], numerous trials established that addition of only explicit artificial viscosity to the numerical algorithm was not sufficient to overcome the non-linear instabilities produced by the sharp gradients generated by this type of start. Although Steger and Bailey were able to successfully utilize this technique by also including implicit artificial viscosity, it was the experience of this author that no combination of explicit and implicit artificial viscosities would lead to suppression of the instabilities inherent in this method of starting the fluid flow. It is conjectured that this is due to the differences in the meshes in either case; although both were "C" type coordinates, the ones used in this work had much smaller cell sizes near the leading edge. Thus the courant number was appreciably larger in the leading edge region in this research as compared to that of Steger and Bailey. If this conclusion is correct, then use of a smaller initial time-step, Δt , should alleviate the problem. However, this was not attempted since a primary goal of this project was to develop techniques which allow the use of larger time-steps.

Figure 2 illustrates the transient behavior of the flow field for this type of start. Notice that even for this late a time (0.6 non-dimensional time units) that the flow, outside of a very thin region near the airfoil and the wake center-line, is essentially undisturbed. The exceptions being the wake expansion at 0.6 of a chord from the trailing edge and the compression wave emmanating from the leading edge. Also note the onset of fatal nonlinear instability indicated by the "wiggles" in the Mach 0.75 contour denoting the compression wave.

A more successful starting technique was the penetration start. As outlined earlier in this paper, this start was accomplished by using a fifth-order polynomial to reduce the suction and blowing at the airfoil surface from free-stream values to no-slip values in one nondimensional time unit (one-hundred time-steps at the standard time-step used throughout this work of $\Delta t=0.01$). This method was fully successful in getting the flow started, however, as noted previously, it had the disadvantages of requiring non-simple body boundary conditions and of causing early boundary layer separation over the aft portion of the airfoil. Ultimately, this approach was abandoned since it was felt that the last mentioned deficiency of this method could lead to a valid, but undesired, steady-state solution. That is, since the fluid flow itself can take on a variety of steady flow states for identical boundary conditions depending solely on the initial conditions [o], it was felt that the early, forced separation of the boundary layer due to this technique could lead to a steady-state not found under normal flight conditions. Also since there were more grid points on the forward section of the airfoil than there were on the aft section, the lack of global conservation of flow properties alluded to in a previous chapter produced by this starting technique lead to extreme cooling of the trailing edge region which took an undesirable number of time-steps to relax.

Figure 3 and Figure 4 present the Mach number contours of the transient flow developed midway through and at the end of the penetration start, respectively. Comparing these figures with the impulsive start Mach contours in Figure 2, it is immediately apparent that the penetration start allows a much larger region of the flow field to develop for the same time increment. Also note that the various expected flow regions are already well-developed by the end of the start-up period. However, it is also well to notice that the forward decelerated flow region and the surface accelerated regions are both distorted in the downstream direction; and that the wake contains an appreciable highly decelerated region in its core.

In that the single laminar flow case treated by this author was initiated by the penetration start method, the gradual start technique will be discussed following the presentation of the laminar flow results. These results are summarized in the Mach contour plots, Figures 4 through 9. Several interesting observations can be made by an inspection of this flow's time history. First note how the retarded flow region expands from its downstream skew due to the penetration start, to an overshoot upstream skew, and then contracts to the expected, slightly elliptical contours. Looking at the aft boundary layer and near wake, it should be noted that these regions first thin out from the thickness induced by the blowing during the start-up; then, between nondimensional times two and three, separation occurs resulting in a progressively thicker separated region and near wake region. Coincident with the occurance of separation, the accelerated regions start to split-up into two zones of maximum Mach numbers. Within each zone, the

Mach numbers decrease with time, although the portion of the flow within each zone remains relatively constant as the outer Mach contours "pinch-off". The forward zone progresses toward the leading edge as the aft zone lifts-off with the separated boundary layer.

The occurance of these later phenomena can be explained as logical consequences of the strongly separated flow. The flow field outside of the boundary layer and separated region "sees" the airfoil as if the separated region were a continuation of the airfoil surface; thus the outer flow must first accelerate around the leading edge region and then over the separated region. Similarly, the general drop in maximum Mach number, from a high of 1.15 before separation occured to 1.05 in Figure 9, and the contraction of the Mach contours delineating the accelerated region, can be attributed to the separated boundary layer giving the airfoil the appearance of a thin wedge to the outer flow. Unfortunately, this author was unable to obtain any comparative studies for this case (not to say that some might not exist).

The gradual starting method in all ways turned out to be an almost perfect flow initialization technique. It only required a modest amount of explicit artificial viscosity, smoothly accelerated the airfoil at 0.0 and 3.5 degrees angle of attack from rest to Mach 0.8 within a nondimensional time of one, and even provided the bonus capability of changing the angle of attack in mid-solution. Best of all, in this author's opinion, it allows the most realistic starting procedure of all those considered in this research (including some not actually utilized, such as potential flow solutions). In all applications the same fifth-order polynomial as that used in the penetration start to vary the airfoil surface velocities was used to vary the free-stream velocities from zero to the desired steady-state values (this was handy and also allows some comparison of the penetration and gradual starts to be made).

The first application of this method was to the NACA 64A010 airfoil at zero angle of attack, as illustrated by Figures 10 and 11. A comparison of the three starting techniques, impulsive, penetration, and gradual (Figures 2, 3, and 10) is instructive, but note that the free-stream Mach number of the flow plotted in Figure 10 is 0.73 rather than the Mach number of 0.8 for the other two. Contrasting the Mach contours of Figures 4 and 11, which indicate the flow fields at the end of the penetration and gradual starts, respectively, one sees that the flows are very similar except in the aft boundary layer and near wake regions. Here the gradual start produces a much thinner boundary layer and near wake, except for the Mach 0.75 contour which has the bubble shape characteristic of experimental results [m].

Immediately after the completion of the gradual start (i.e., at nondimensional time of one), the Baldwin and Lomax turbulence model was turned on and the solution continued out to a non-dimensional time of six, as with the laminar case. A comparison of the turbulent results (Figures 12 through 16) with the laminar results (Figures 5 through 9) proves to be enlightening. First of all, note that the forward decelerated region develops <u>almost</u> identically in both cases, which is as would be expected. Also, the location and intial slopes of the upstream accelerated flow Mach contours 0.85, 0.90, 0.95, and 1.00 remain in good agreement between the two flows until after the nondimensional time of four. Even after this point, the first three contours agree in upstream location and initial slope near the airfoil as far as both cases were taken. This particular behavior of both the decelerated and accelerated regions can be attributed to the fact that the turbulence model, as implemented by this author, does not become effective until slightly downstream of the first minimum surface pressure location (i.e., turbulent transition is modeled), thus, upstream of this point one would expect laminar and turbulent flows to behave similarly at least for early times and at the freestream Mach number chosen (0.8). A final obvious contrast is the presence and non-appearance of separated flow in the two cases, which reflects the elementary experimental and theoretical precept that turbulent flows can tolerate an adverse pressure gradient much better than laminar flows.

In comparing the turbulent Mach contour results it is first of all apparent that the near wake reaches a quasi-steady state very early, the only subsequent significant changes being the thickening and ultimate closure of the middle wake region. This last result is somewhat puzzling, especially since the entire wake shows practically no change at all between nondimensional times of four and five. Also note how the accelerated flow Mach contours progressively expand out from the airfoil and gradually shift toward the leading edge. Their initial downstream skew also tends to turn into an almost symmetrical appearance by the last time step plotted (Figure 16). The Mach one contour never becomes normal to either the chord line or the airfoil surface and the spacing of the downstream Mach contours never take on a crowded appearance. Thus it is hardly justified to claim that a shock develops on the airfoil; however, it appears from the interferogram of reference [m] that in the experimental case the shock is rather weak. None-the-less, it was decided at this point that the ability of the numerical algorithm to correctly develop shocks needed to be given a less ambiguous test.

Referring to either reference [k] or [m], it is clear that in experimental studies of this airfoil, a moderately strong shock appears on the upper surface for an angle of attack of two degrees. Thus, this angle of attack was chosen for the next test case which used the previous results at nondimensional time of four for initial conditions. The gradual start method was used to change the angle of attack from zero to two degrees, effectively by causing the airfoil to "fall" at a speed sufficient to produce this change (note that this is not the same as rotating the airfoil, and thus, will produce different transient states and could produce a different steady state). Then the solution was continued out to a nondimensional time of seven, at which point the solution process was terminated due to the financial suicide of the university's computer center.

Figures 17, 18, and 19 contain Mach contour plots similar in nature to those discussed previously. It is immediately noticeable that the flow in all regions is no longer symmetrical about the chord line. The progressive changes in the decelerated flow region and the wake region are as in the zero degree angle of attack case except for their respective skew in the upstream and downstream directions. More notably different is the evidence of much more strongly accelerated flow along the upper surface; however the form and density of the Mach contours still don't suggest that a shock is present. To clarify this lack of an expected flow phenomina, the Mach contours plot identified as Figure 20 was made with the range of Mach numbers being 0.96 to 1.04 and the increment being 0.01 rather than the range and increment used in the other Mach contour plots. As can be seen, there is a pronounced difference between the upper and lower contour groups, as expected. However, the angle and density of the upper groups downstream foot are not as would be expected for even a rather weak shock. Although, as mentioned previously, it was not possible to continue this solution so as to check this hypothesis, it was conjectured that much of the spread in these Mach contours was due to the relatively large amount of explicit artificial viscosity used. This was reasoned from the fact that the solution showed no signs of "wiggles", which is unusual for numerical solutions of flows containing shocks.

A valuable check of the internal consistency of the numerical solution, as well as an additional means of comparing the results with the available experimental data, can be obtained from a density contour plot such as Figure 21. First of all, comparison of these contours with those of reference [k] reveals at least good qualitative agreement with the "set" angle of attack of two degrees data set. Although reference [m] does not present results for this particular angle of attack, the density contours of Figure 21 do seem to represent a case between the angles of attack of 0 and 3.5° that are shown. These comments must be tempered with the fact that both references clearly show a shock on the upper surface of the airfoil whereas the Mach number and density contour plots of the algorithm used by this researcher clearly do not show such a shock.

Some interesting observations can be made in comparing the density contours of Figure 21 with the Mach contours of Figure 19 which have some bearing on the interpretation of experimentally determined density contours obtained by interferometer, such as those of reference [m]. If one makes the same assumption as used in the cited reference, that the flow is isentropic everywhere except in the boundary layer and across any shocks, then one can obtain the following relationship between non-dimensional density and Mach number:

$$\rho^{\gamma-1} = 1 + \frac{\gamma-1}{2}(M_o^2 - M^2) \qquad (5.1)$$

where for this equation only, M is the local Mach number and M_o is the reference Mach number (0.8 in this case). (For the convenience of the readeɪ, a rough rule of thumb summarizing this equation for the current application is that a change in density of 0.05 - one density contour of Figure 21 - results in a change of Mach number of about 0.06 to 0.08). Using this relation, it is easy to see that there is good agreement between the two sets of contours except for four flow regions. The most puzzling discrepancy is the lack of alignment of the Mach 0.75 and density 1.05 contours upstream of the leading edge, since this region would seem to be nearly isentropic. Possibly this is due to the fact that the flow has not quite reached steady state. Similarly, the lack of agreement between the innermost accelerated region contours of the two plots could be attributed to a combination of nearness to the non-isentropic boundary layer and the transient nature of the flow; however, these arguments are not convincing. More importantly, note how the downstream intersection of the accelerated flow region density contours with the boundary layer are shifted upstream and are more nearly normal to the surface than the corresponding Mach number contours. Also note the almost total lack of similarity of the Mach 0.75 and density 1.05 contours near (or in) the wake - clearly due to the non-isentropic nature of the wake region.

The last case considered was that of the same NACA 64A010 airfoil at an angle of attack of three and one-half degrees. This case was considered due to its moderately strong shock development and because of the excellent set of wind tunnel data gathered for this configuration by Johnson and Bachalo [m]. The results for this case are in many ways the most successful of those attempted in that a distinct shock is seen to form; however, there exist a number of unresolved problems connected with this solution. Most disappointing of which was the unexpected onset of numerical instability around a nondimension time of six.

Figures 22 through 27 illustrate the Mach number contours generated at various time increments by this solution. Again, the gradual start method was used, completing at nondimensional time of one. In this case, though, the airfoil is initially at an angle of attack. Comparing the progression of the Mach contours, one is struck by a similarity between all of the cases studies in this work; that is that all of the flow regions tend to expand from the start until sometime between nondimensional times four and five, at which point they start to contract, except for possibly the accelerated region on the shock side of the airfoil. The reason for this surely has something to do with the rapidity of the start; it is doubtful that the outer boundaries contribute to this phenomina since they are ten chord lengths away from the airfoil leading edge in all directions.

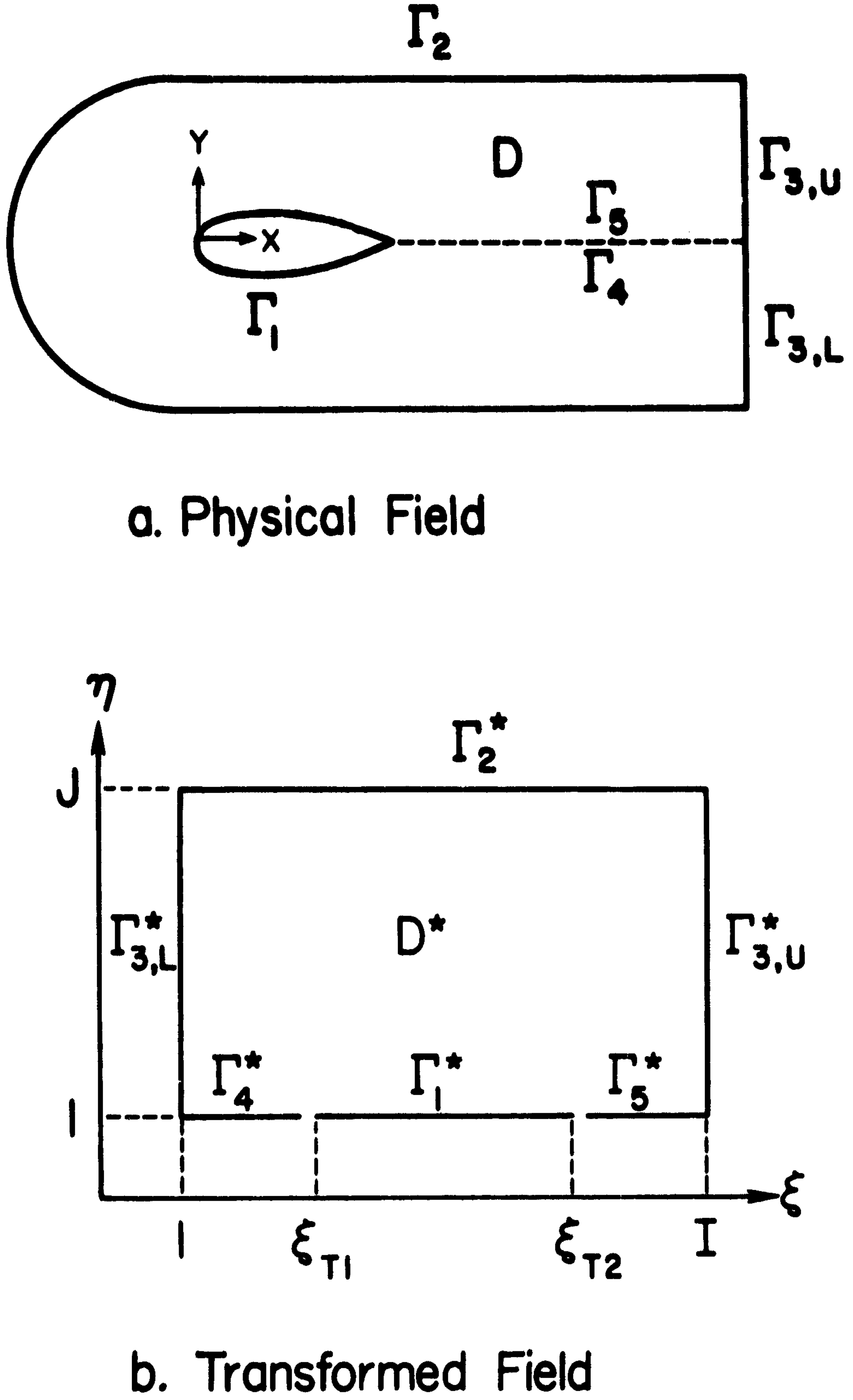
Since much of the description of the results for this case parallels that of the previously presented cases, this presentation will concentrate on the two distinctive results of this solution. The most heartening of these is the development of what can only be described as a shock on the upper surface of the airfoil. The Mach contours on the upper rear portion of the airfoil section are clearly crowding together into a thin bundle which is normal at least to the chord line. This shock formation can also be observed from the coefficient of pressure plots (Figures 28 through 33) in that the downstream portion of the upper surface curve progressively steepens. Lastly, the obvious signs of boundary layer separation on the upper airfoil surface near mid-chord, also gives support to the development of a shock-like pressure gradient. The other distinctive feature peculiar to this solution is the appearance of the wake "bubble" contour just off the trailing edge. While no definitive explanation of this effect can be put forward, it is speculated that this anomaly is either due to the onset of boundary layer separation, a "bug" in the coded algorithm, or the unexpected, but actual way a flow of this type would physically occur. Unfortunately, as a comparison of the Mach contours and coefficient of pressure plots with each other and with the experimental results of reference [m] indicate (see Figure 33), this flow is approaching, but has not yet reached, steady state. However, due to the aforementioned stability problem and the deplection of the time allotted to this research, this work must end on a promising, yet frustrating note.

As an afterword to those who may wish, for whatever reason, to reproduce or continue this work: in addition to the fixed flow parameters mentioned at the first of this chapter, all cases were attempted under the adiabatic wall assumption, with the non-dimensional time-step always fixed at $\Delta t=0.01$, with the explicit artificial viscosity coefficient set at $\varepsilon=8\Delta t$ except for the last two-hundred time-steps of the last study where $\varepsilon=4\Delta t$ was used, and with the coordinate system generated by Dr. Joe Thompson, Aerospace Department, Mississippi State University, Mississippi 39762.

VI. CONCLUSIONS AND RECOMMENDATIONS

On the whole, the primary objective of this research work was obtained, although some subsidiary goals must be left to future investigations. An approximate factorization algorithm modeling the full, Reynolds-averaged, Navier-Stokes equations was successfully developed and implemented in a general-purpose computer code. This computer program includes the desired features of an algebraic turbulence model, penetration and gradual start-up routines, variable inflow option, variable body surface velocity capability, body surface heat flux or temperature distribution specifications, implicit and/or explicit artificial viscosity options, and translational rigid body motion capability. In addition, the basic algorithm was formulated in conservation form on a body-fitted curvilinear coordinate system. The amount of computer time required per time step averaged 35 seconds on the Univac 1100/80, and 4.5 seconds on the CDC Cyber 203 (this works out to approximately 8 hours and 1 hour total computer time for a complete time-history of the flows considered here). However, although the cases studied are indicative of the full capability of the algorithm as conceived by this researcher, further work is clearly required in the area of model verification.

Of the multitude of observations that can be drawn from the body of this research, it is felt that the following are most worthy of attention. While the body-fitted coordinate technique was fundamental to the implementation of this algorithm, two areas in which improvements are urgently needed can be identified. These involve the development of rational and efficient means for determining coordinate line spacing and placement and for dynamically concentrating coordinate lines in the region of developing shock waves. As noted earlier in the text, and related to the last topic, the standing problem of how to express truncation errors in curvilinear coordinates in as concise and meaningful a manner as the "order of" method of Cartesian coordinates, should be resolved if differencing in transformed coordinates is ever to have a rational basis. Boundary condition problems tended to dominate the difficulties encountered in application of this algorithm. In particular, the much-neglected downstream boundary condition requires more thought than has been given it to date - in fact, the entire problem of replacing "infinity" boundary conditions by finite distance boundary conditions needs a firmer theoretical basis. In using the adiabatic boundary condition in this research, numerical evidence seems to indicate that, at least in the implicit portion of the algorithm, either the density or the temperature can be extrapolated into the field, but not both together if nonlinear instabilities are to be avoided. This points to the whole unsatisfactory situation in which there is no firmly established "fourth" body boundary condition - a topic which needs some more fundamental research efforts. Lastly, the results of this research indicate that construction of Mach number contours from experimentally determined density contours may not always accurately reflect the true Mach number contours. Thus, it is suggested that published experimental results include the density contour tracings in place of, or along with, the calculated Mach number contour plots.



a. Physical Field

b. Transformed Field

Figure 1. Wake Type Coordinate System

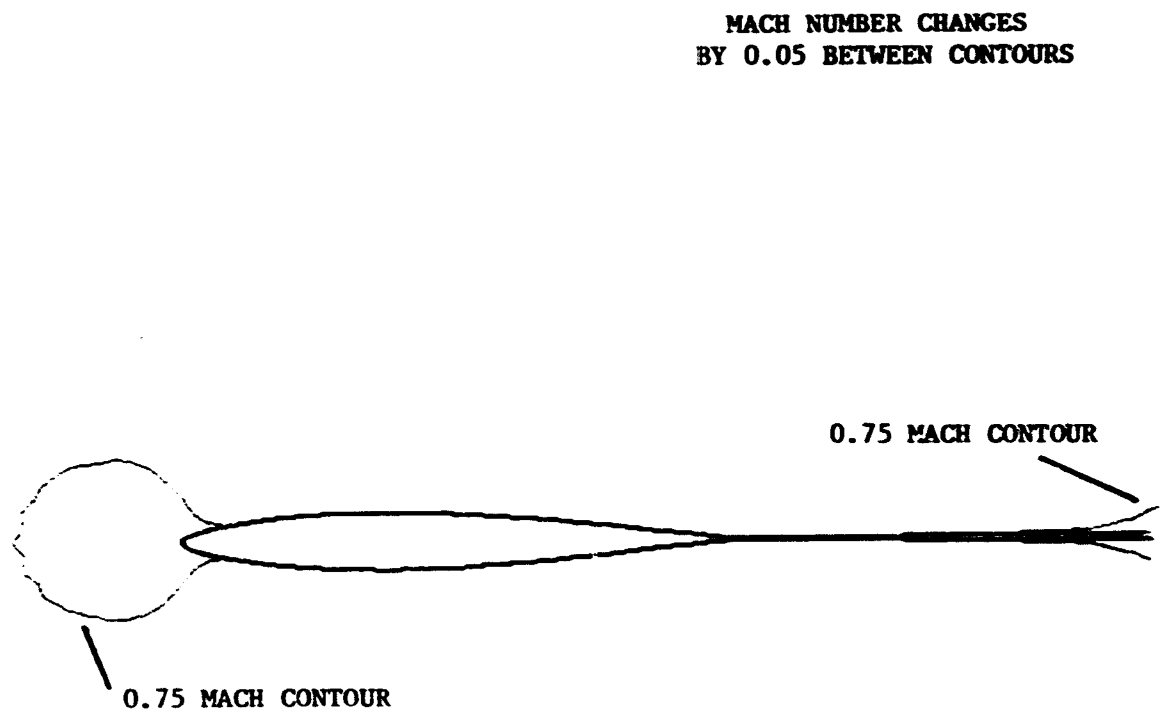


Figure 2. Mach Number Contours - Impulsive Start,
t = 0.6

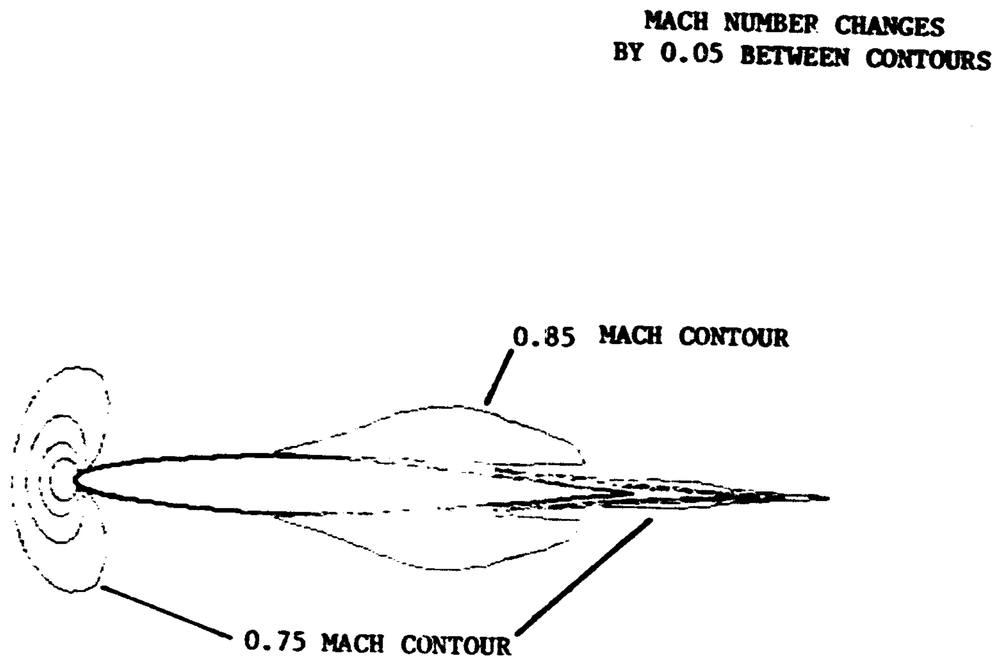


Figure 3. Mach Number Contours - Penetration Start, t = 0.5

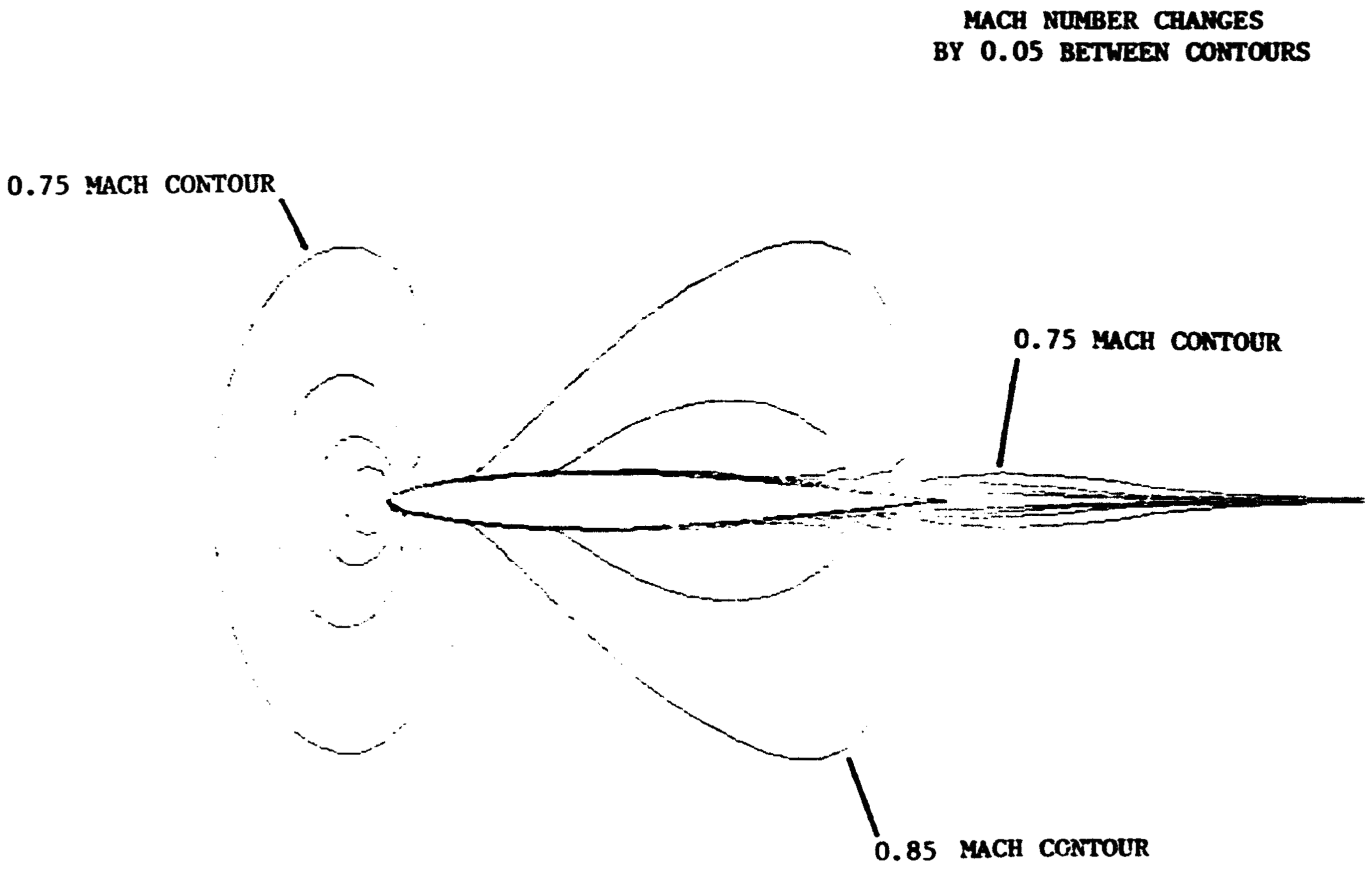


Figure 4. Mach Number Contours - Penetration Start,
t = 1.0

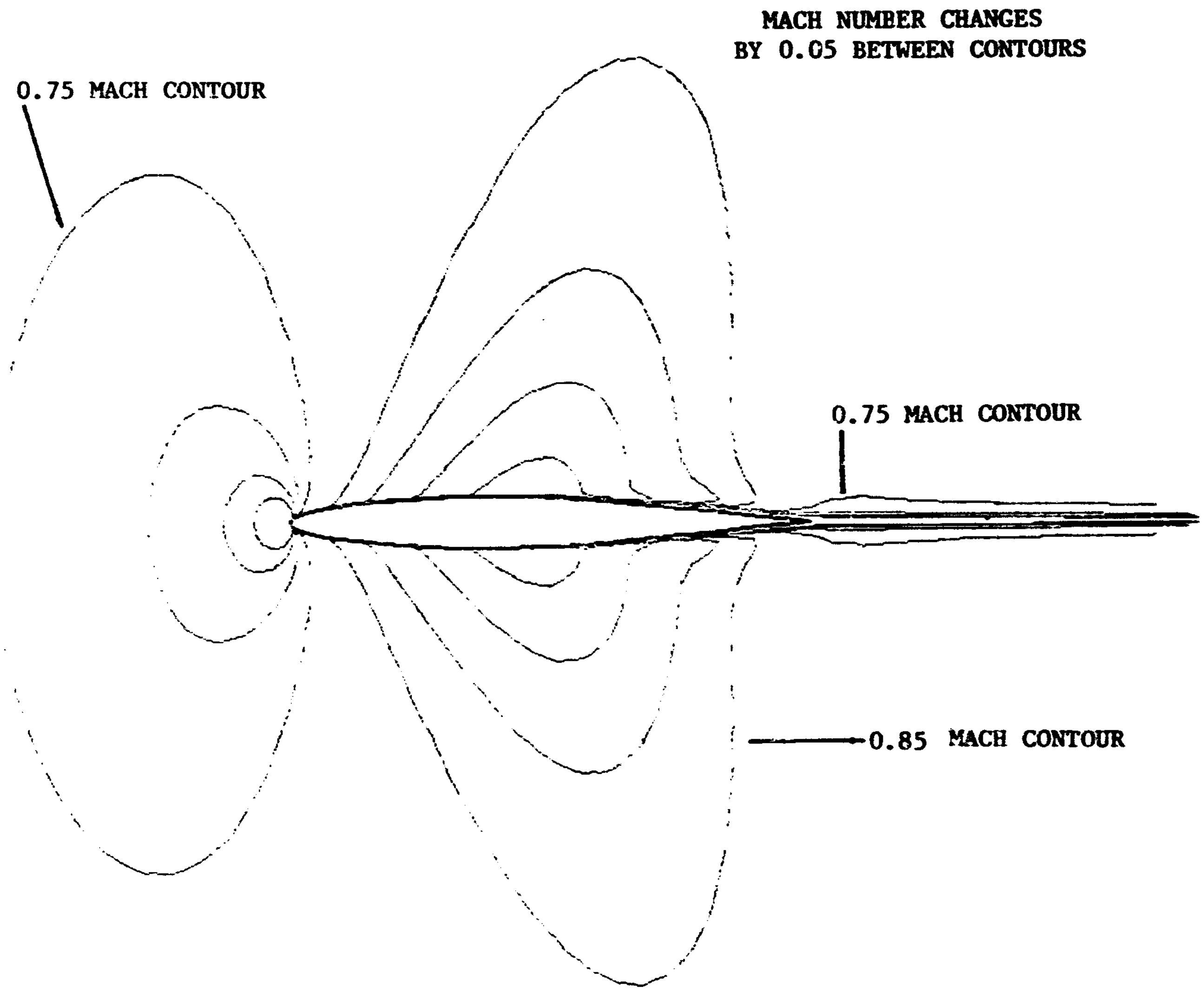


Figure 5. Mach Number Contours - Penetration Start, t = 2.0

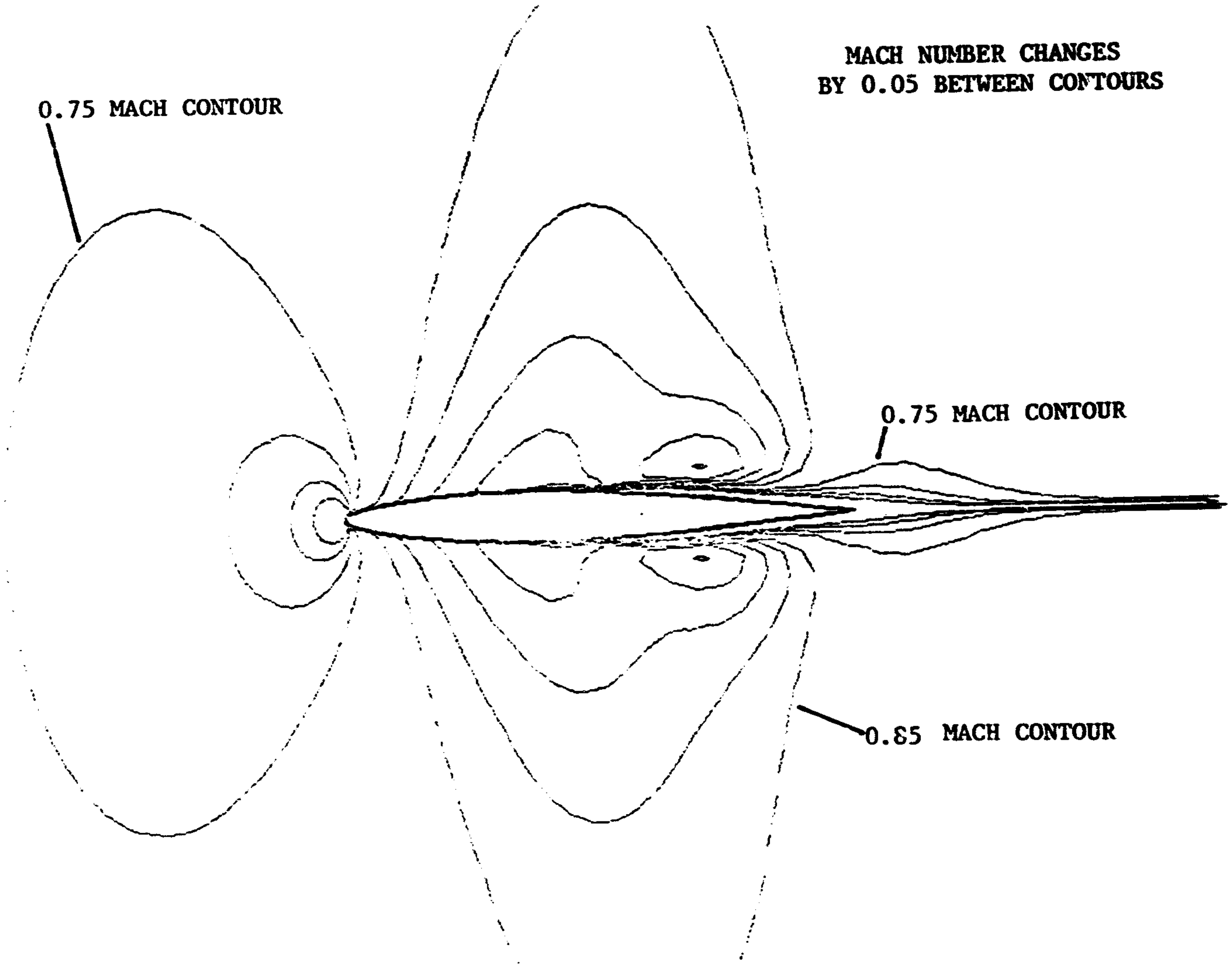


Figure 6. Mach Number Contours - Penetration Start, t = 3.0

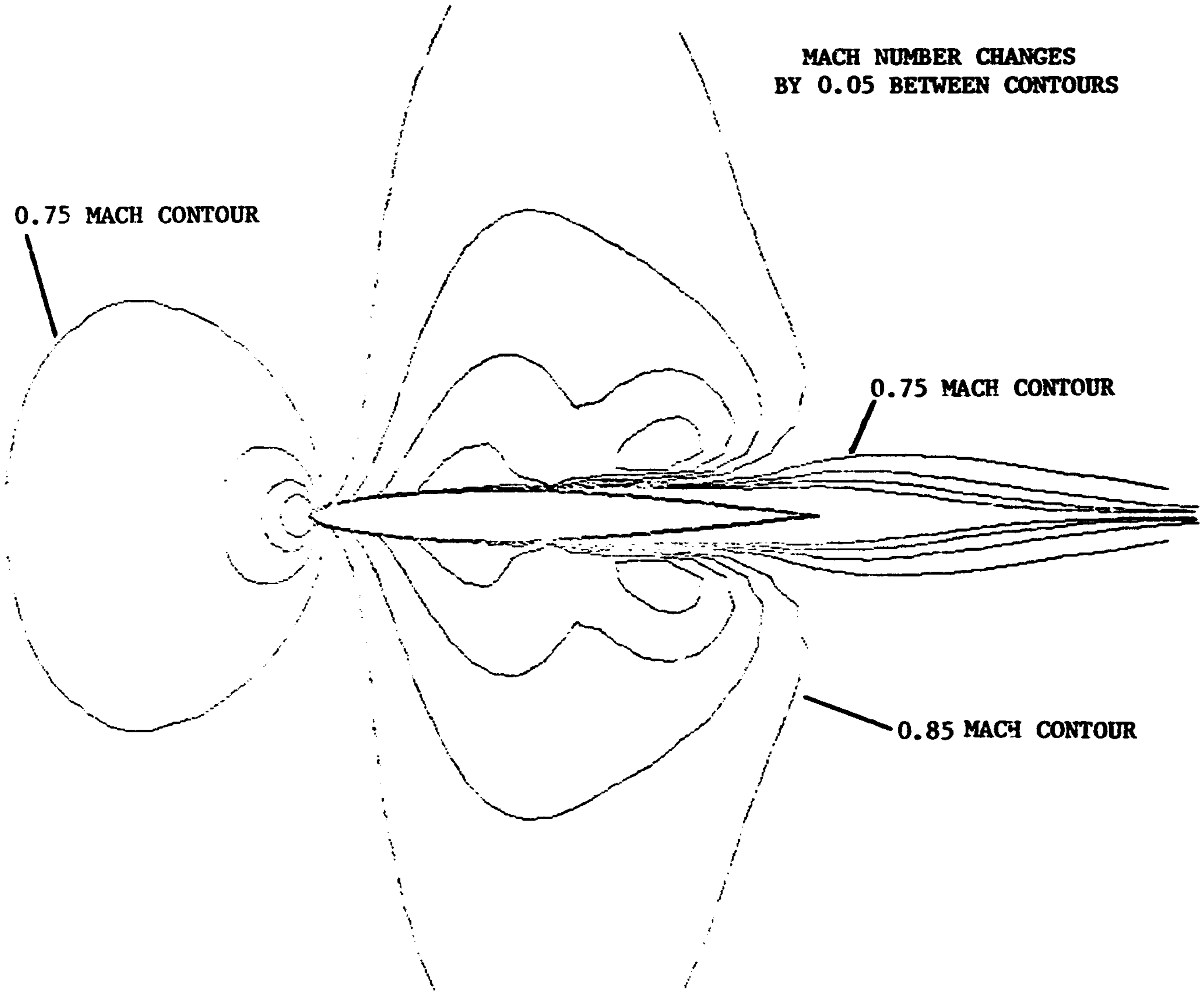


Figure 7. Mach Number Contours - Penetration Start, t = 4.0

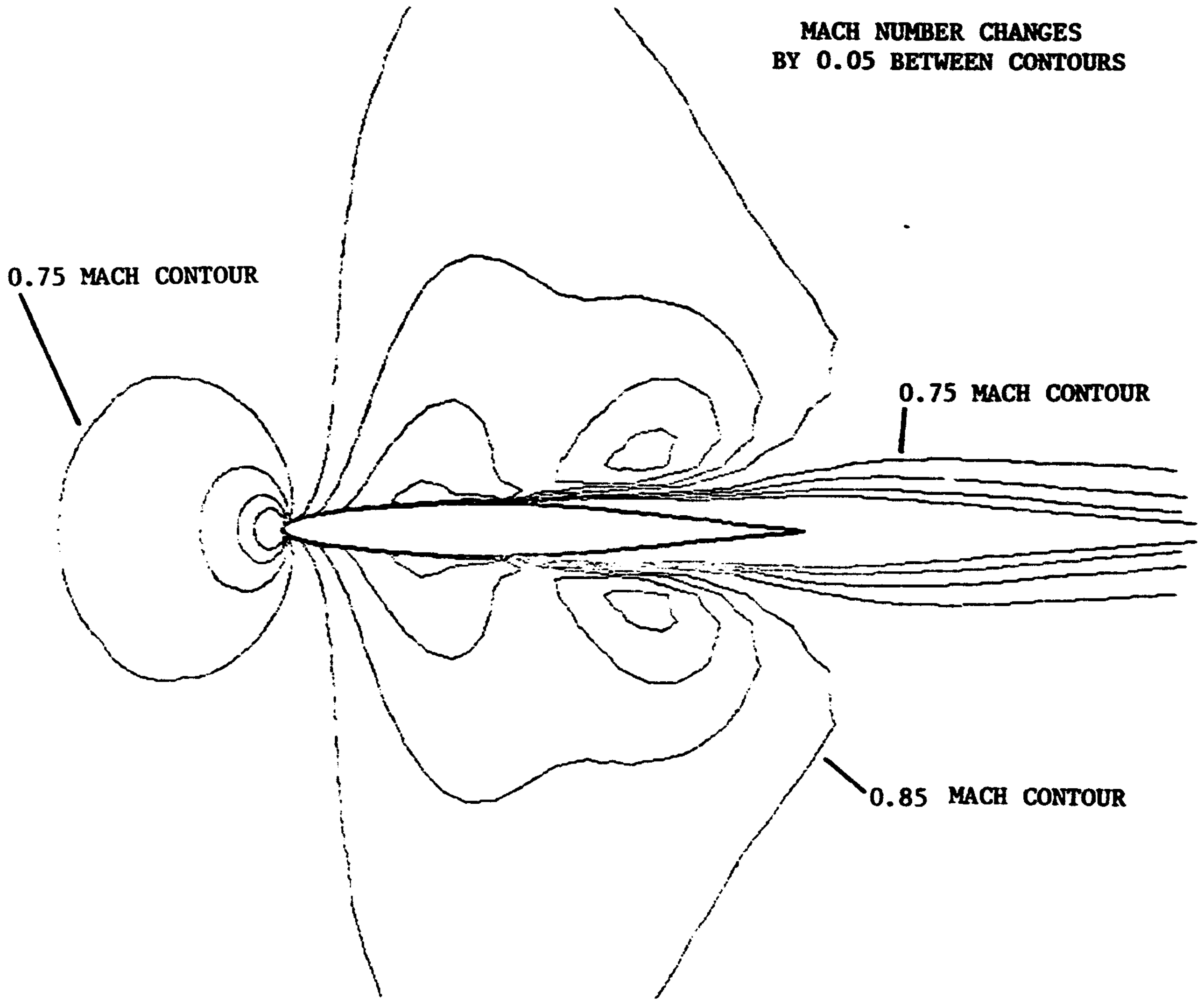


Figure 8. Mach Number Contours - Penetration Start,
t = 5.0

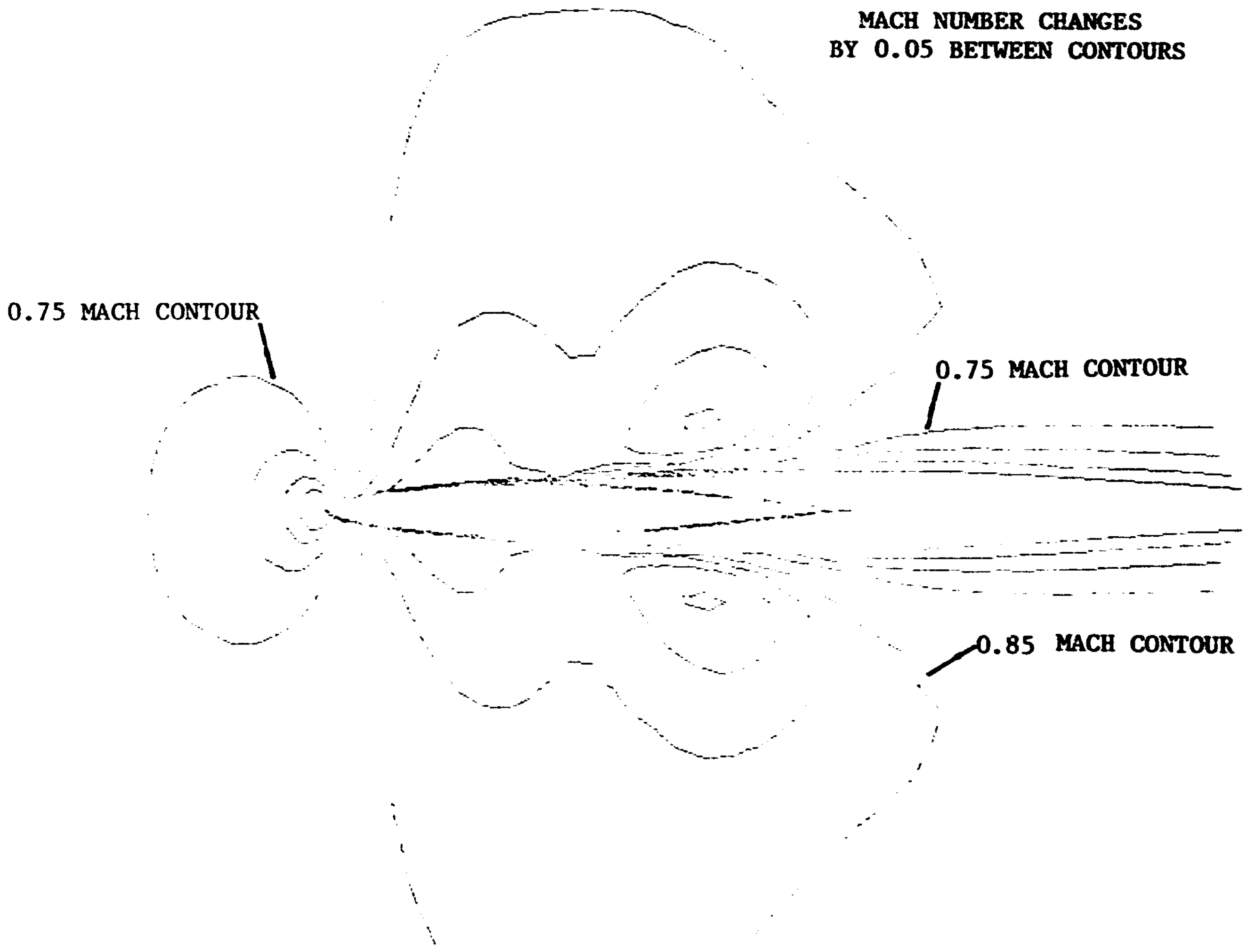


Figure 9. Mach Number Contours - Penetration Start, t = 6.0

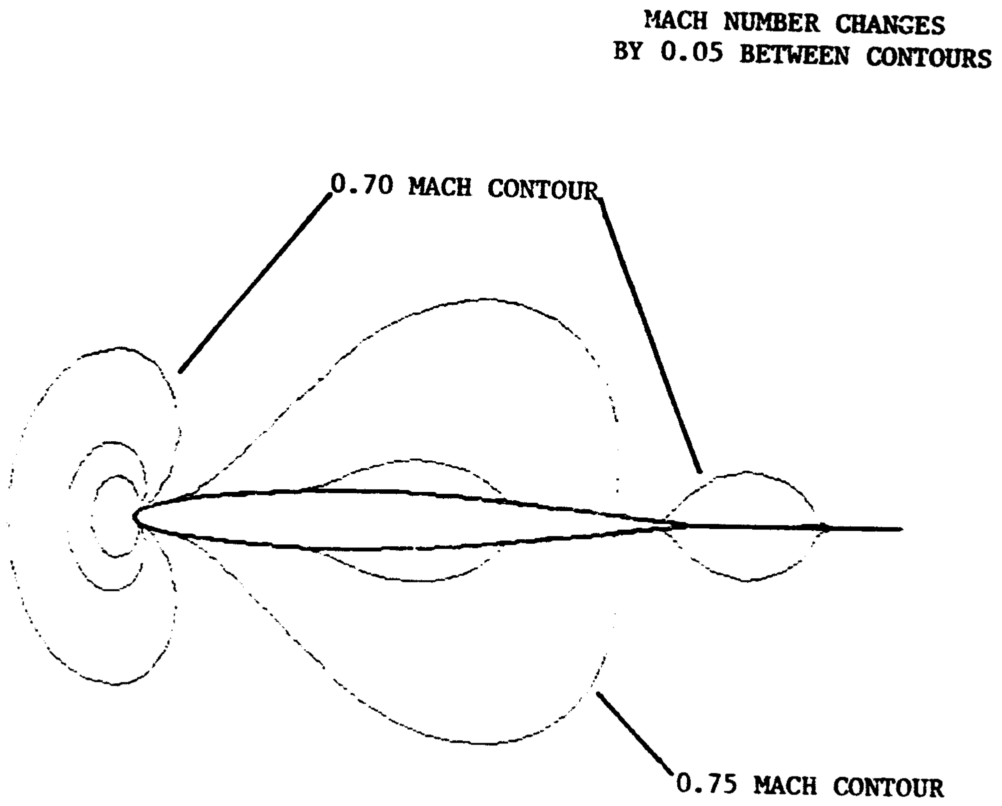


Figure 10. Mach Number Contours - Gradual Start, $\alpha = 0^\circ$, $t = 0.6$

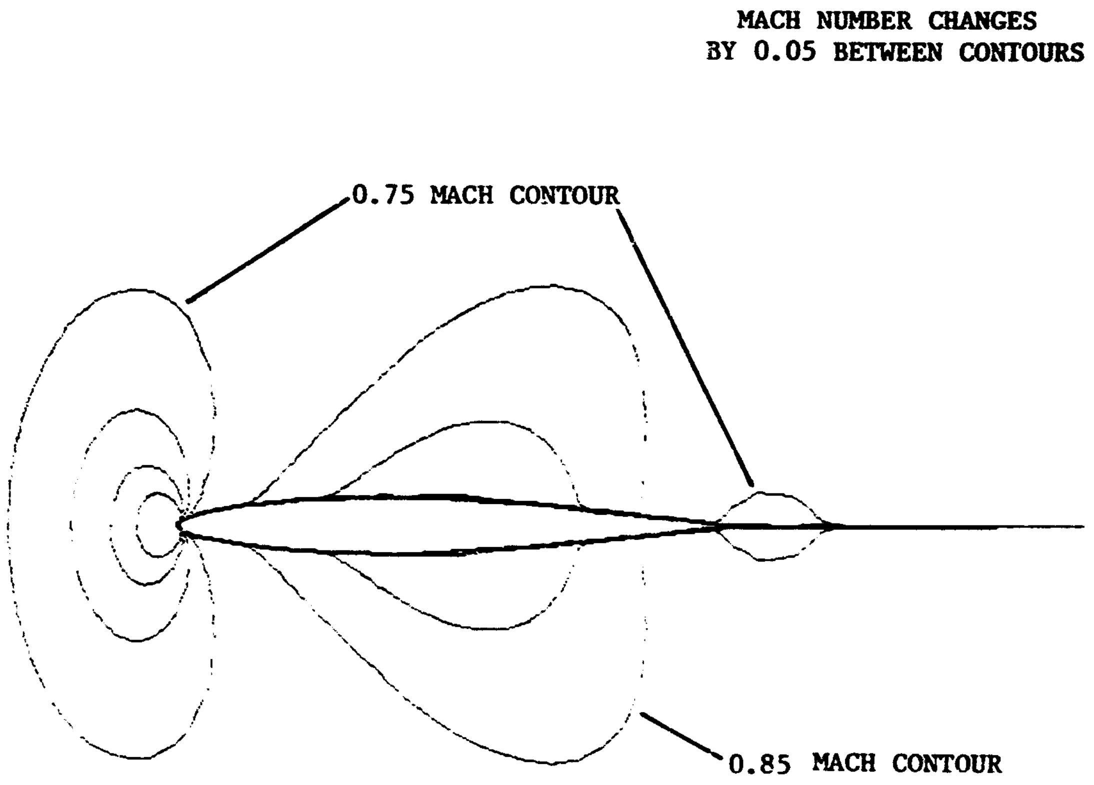


Figure 11. Mach Number Contours - Gradual Start,
$\alpha = 0^{\circ}$, $t = 1.0$

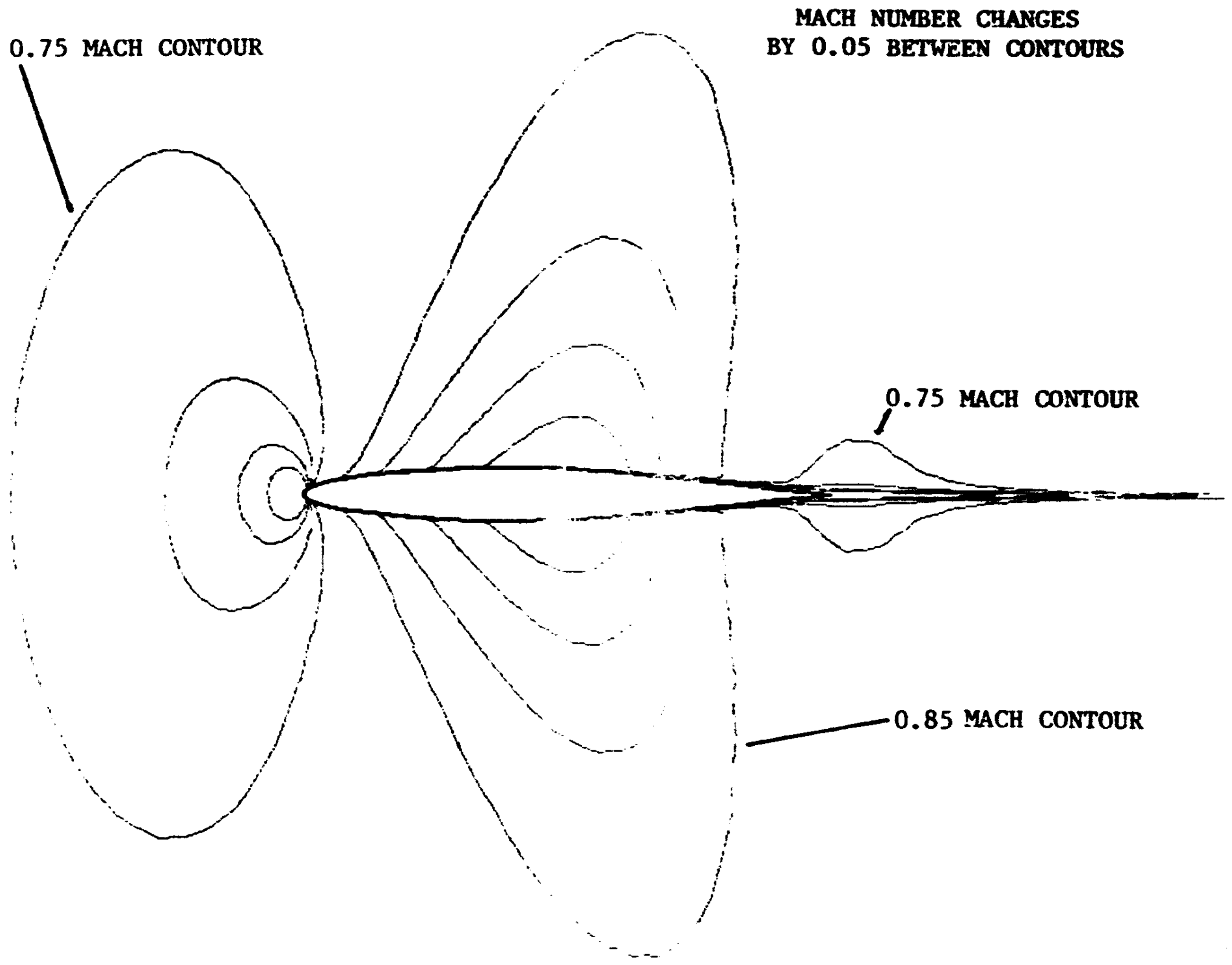


Figure 12. Mach Number Contours - Gradual Start,
$\alpha = 0^\circ$, $t = 2.0$

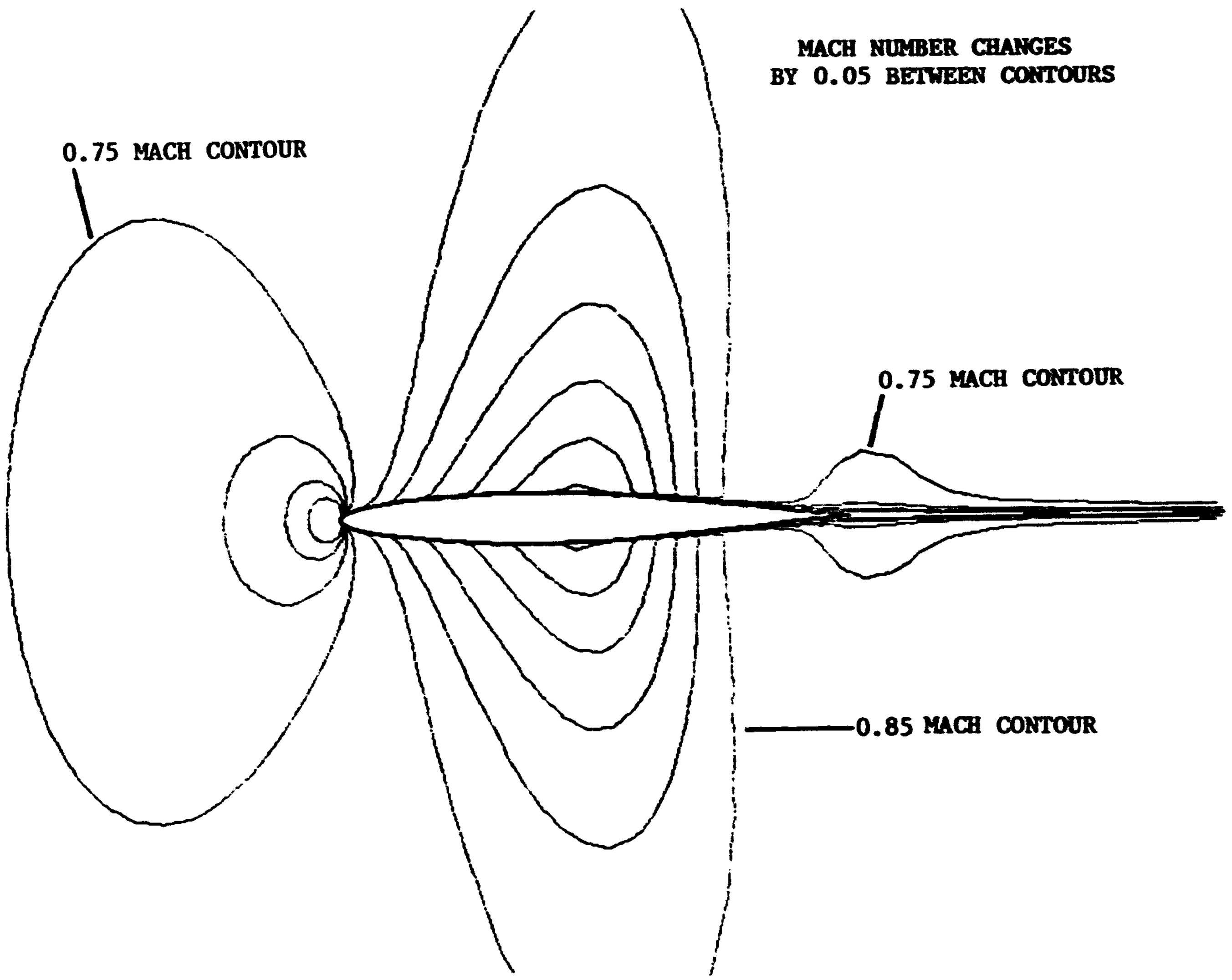


Figure 13. Mach Number Contours - Gradual Start, $\alpha = 0^{\circ}$, $t = 3.0$

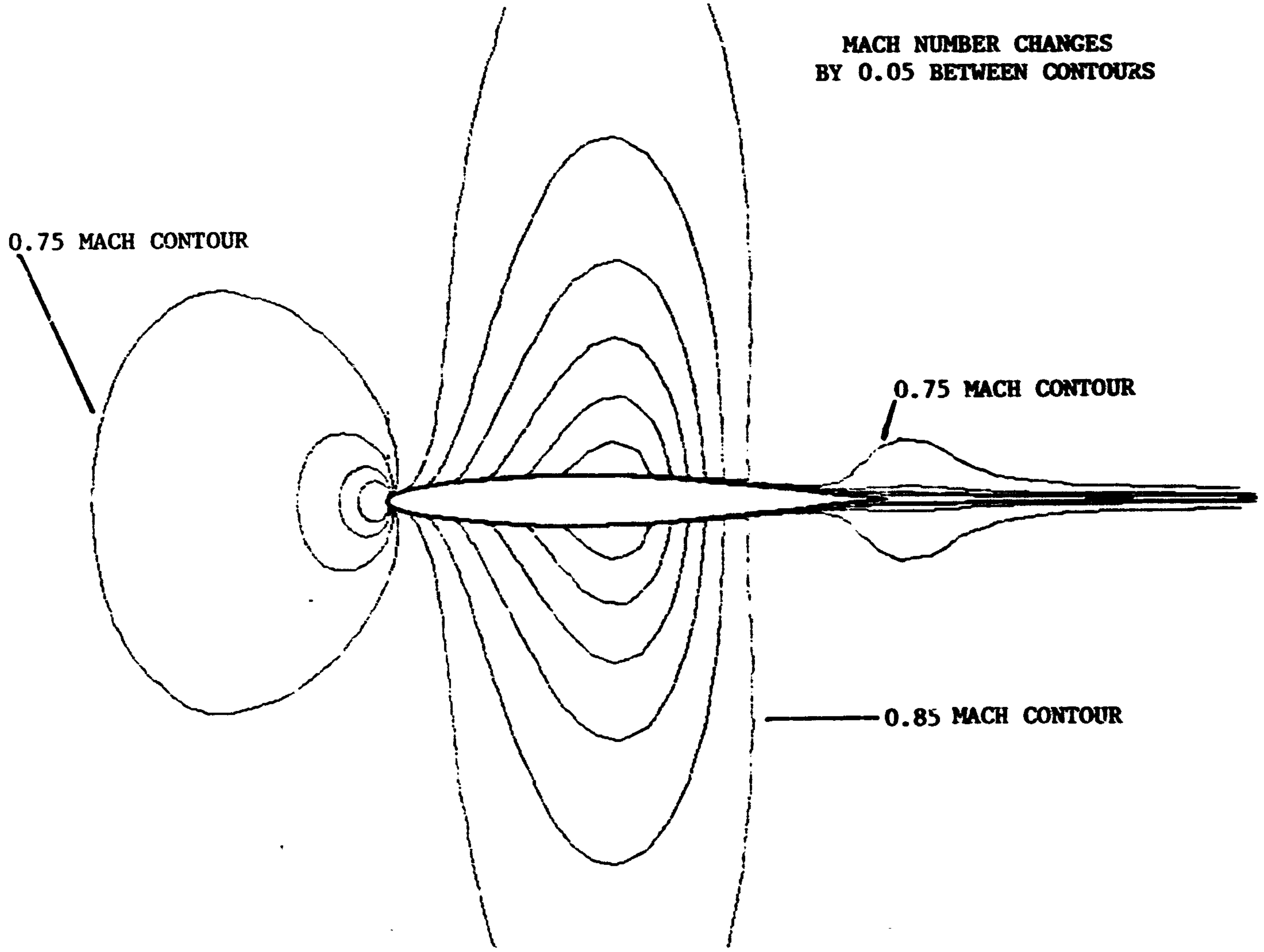


Figure 14. Mach Number Contours - Gradual Start, $\alpha = 0^{\circ}$, $t = 4.0$

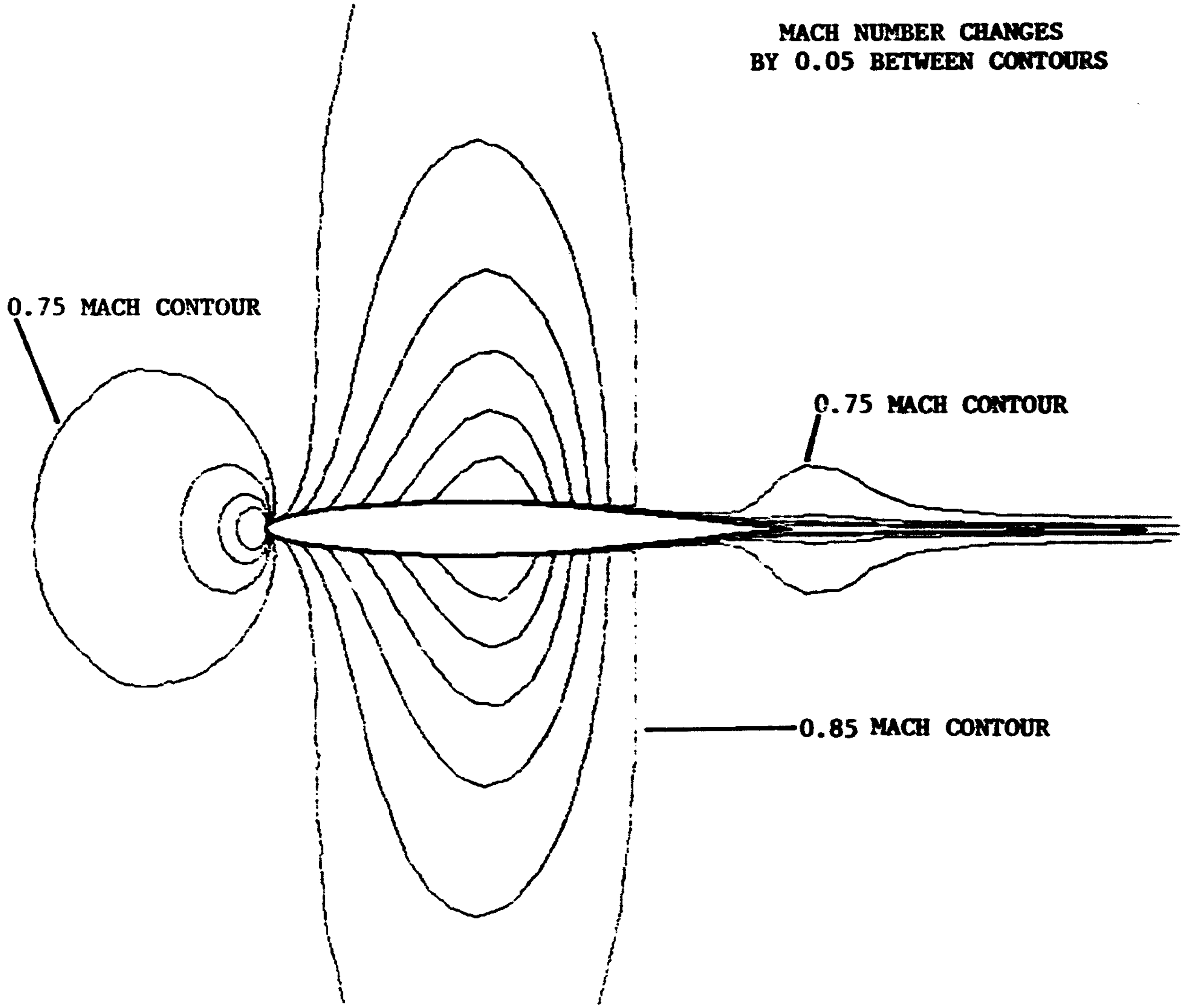


Figure 15. Mach Number Contours - Gradual Start, $\alpha = 0^\circ$, $t = 5.0$

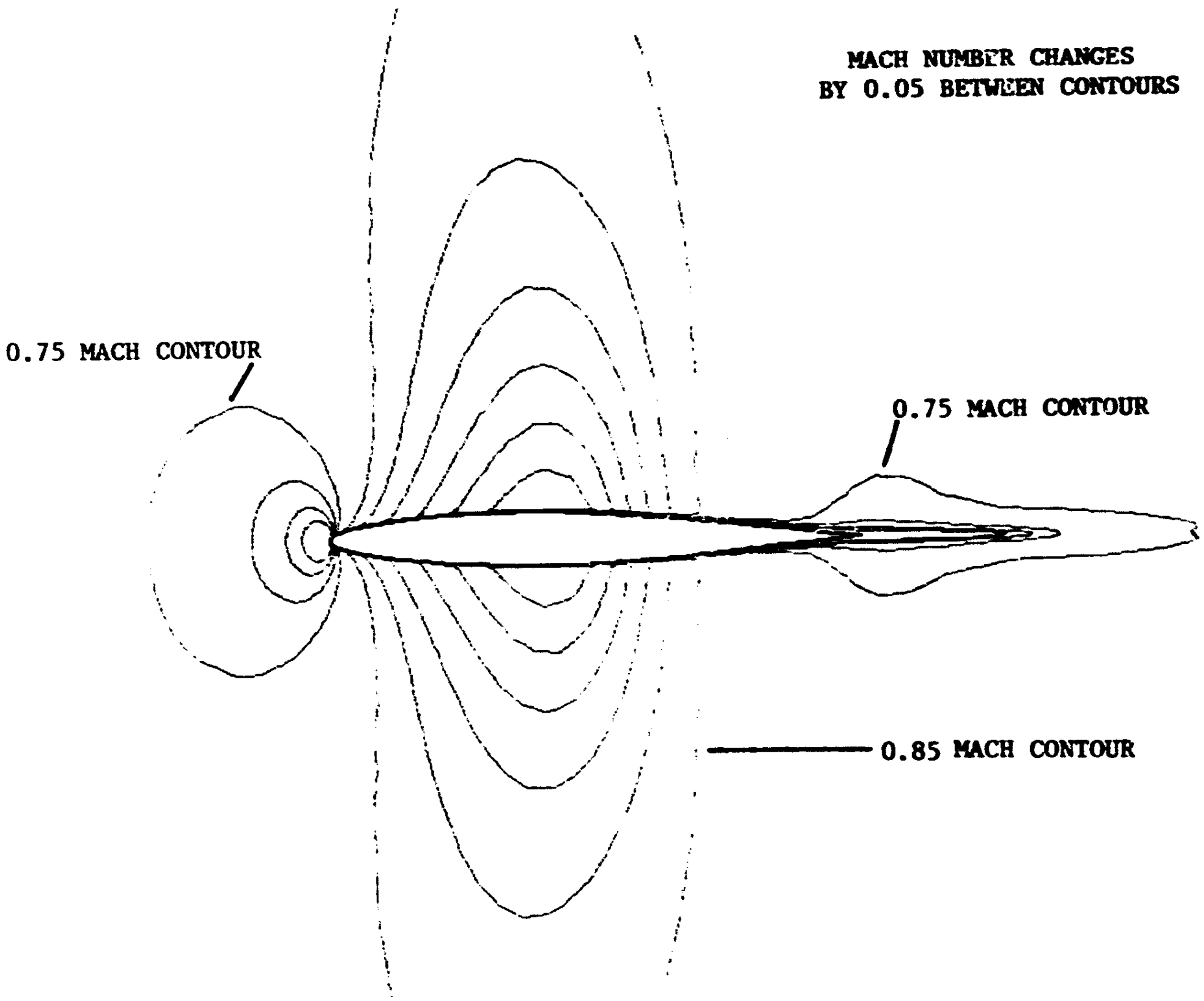


Figure 16. Mach Number Contours - Gradual Start,
$\alpha = 0^{\circ}$, $t = 6.0$

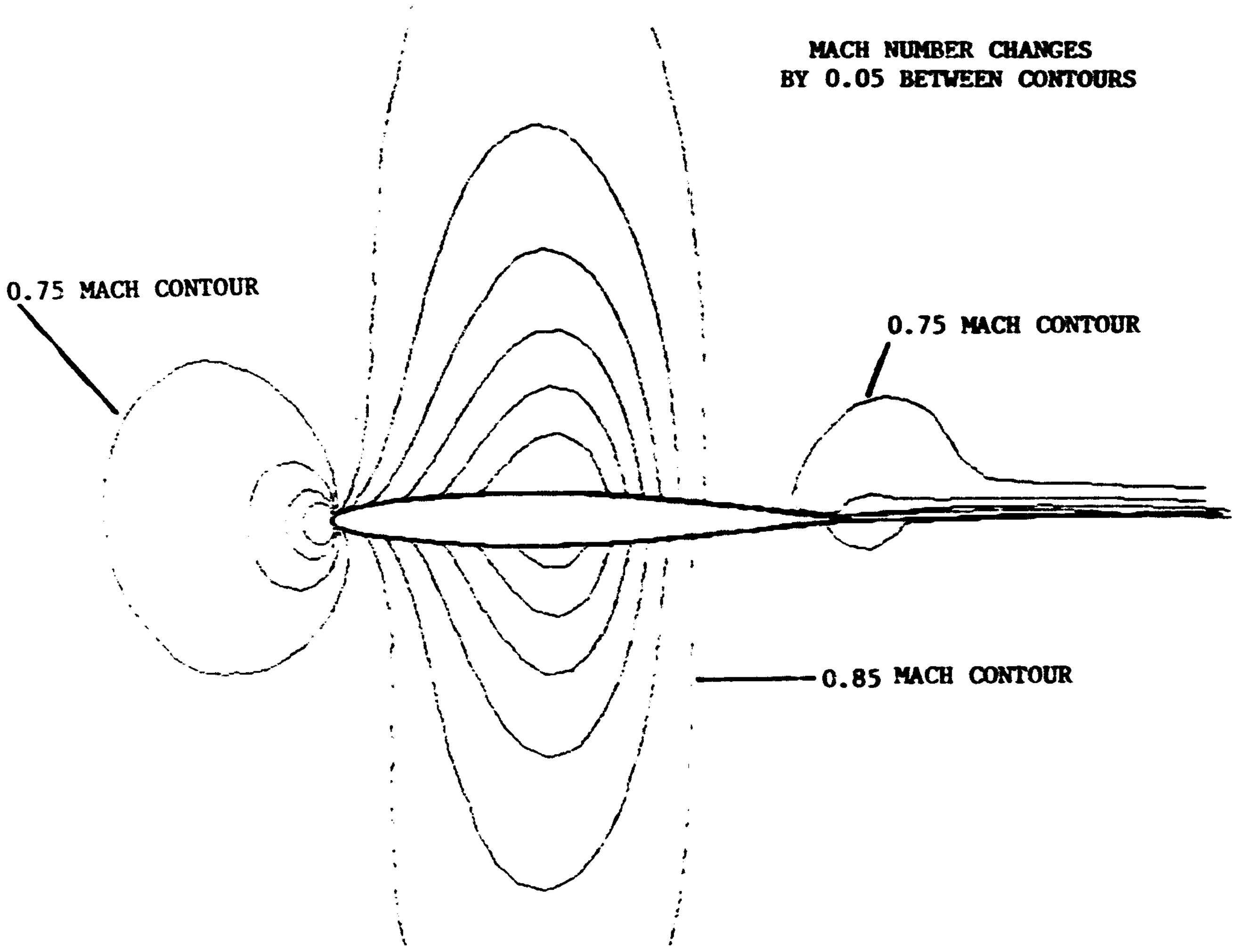


Figure 17. Mach Number Contours - Gradual Start,
$\alpha = 2^{\circ}$, $t = 5.0$

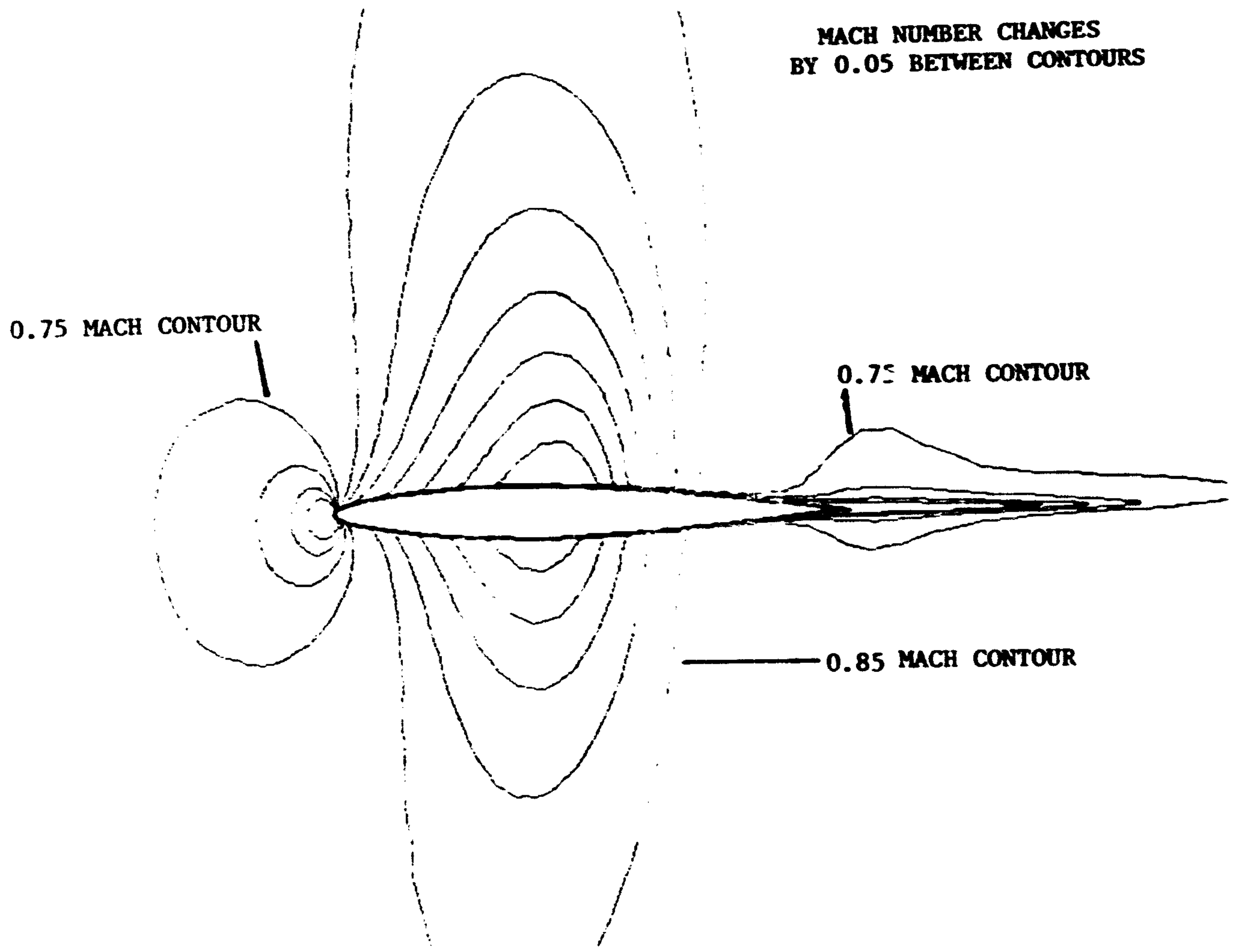


Figure 18. Mach Number Contours - Gradual Start
$\alpha = 2^{\circ}$, $t = 6.0$

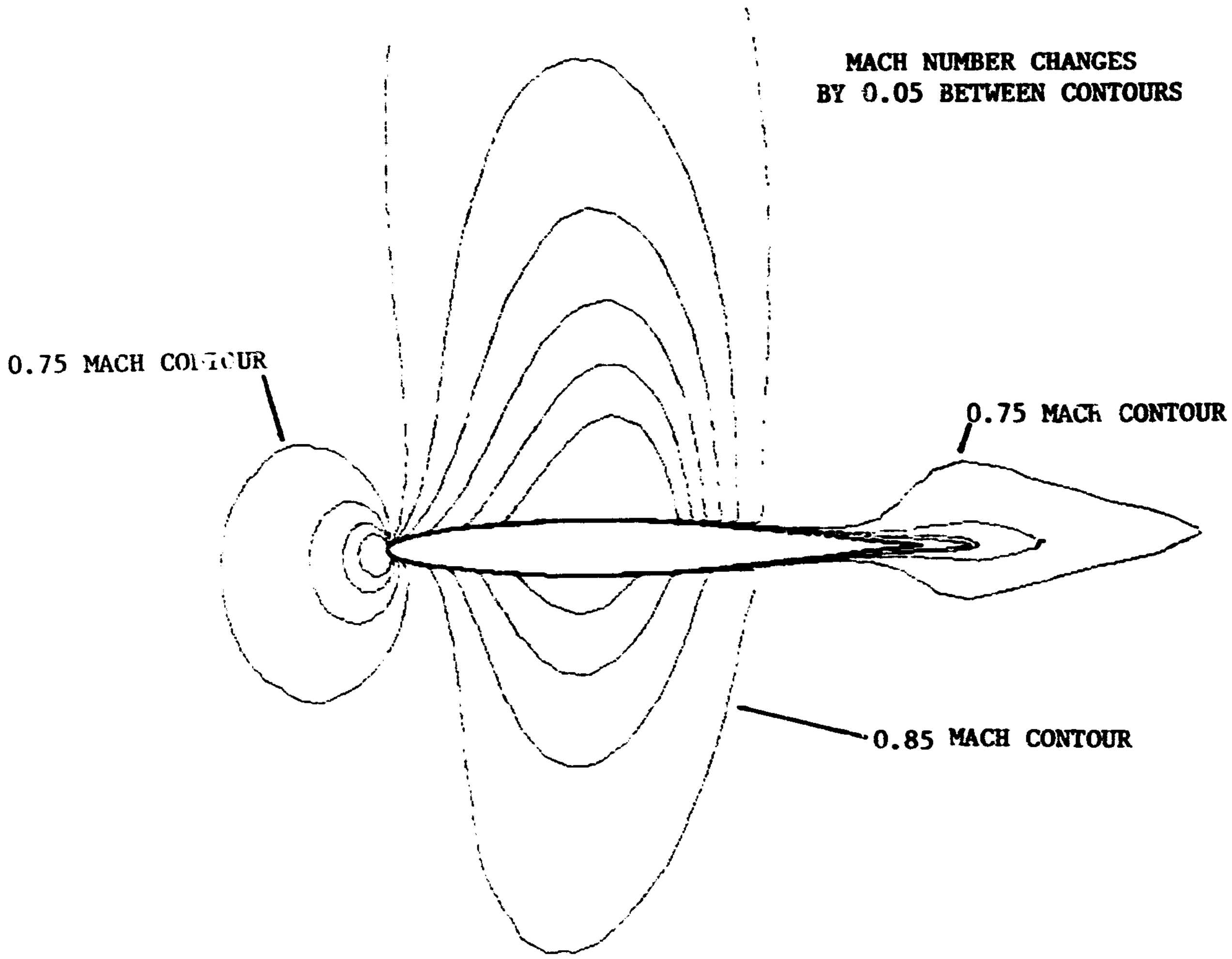


Figure 19. Mach Number Contours - Gradual Start, $\alpha = 2°$, $t = 7.0$

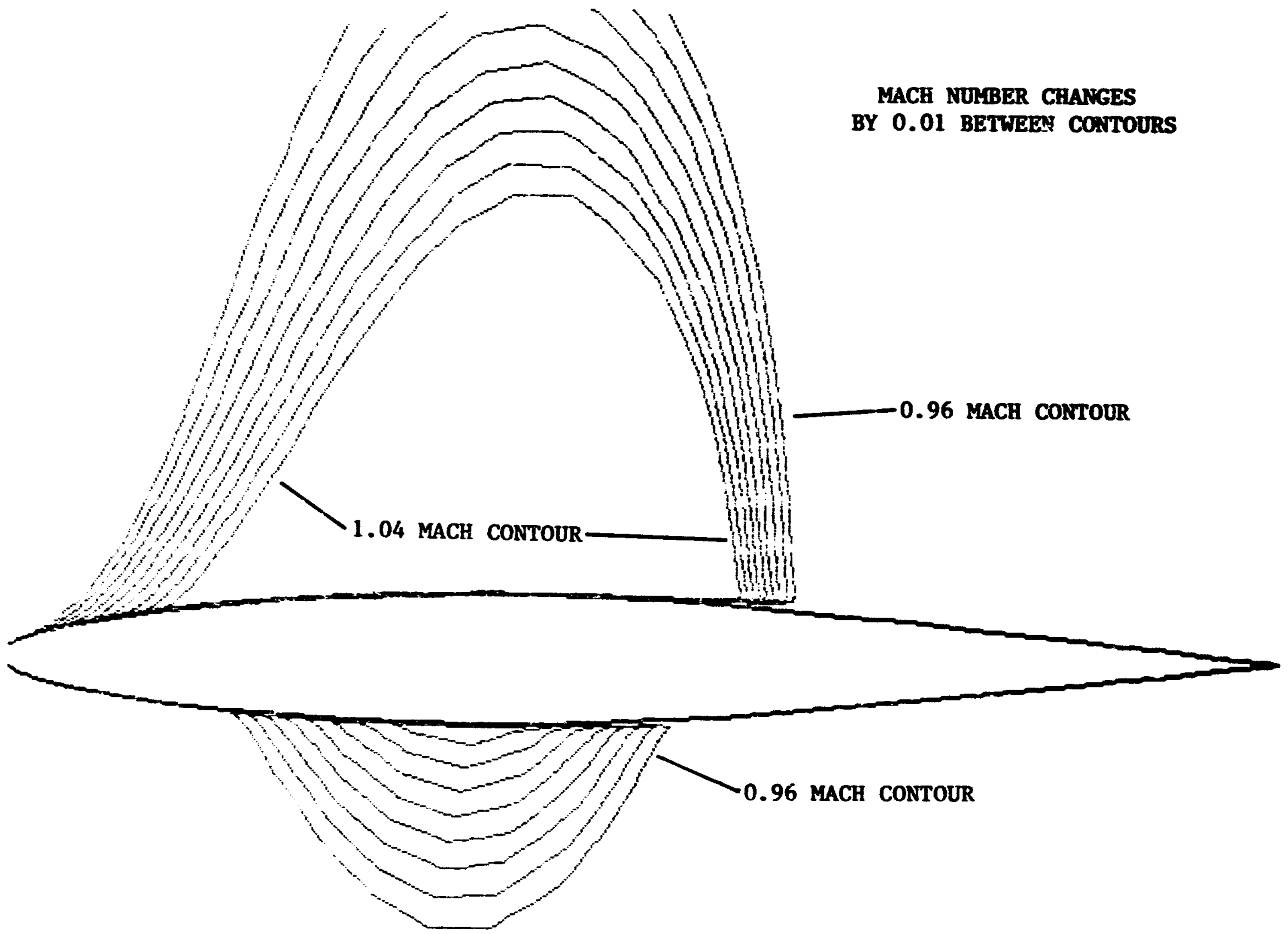


Figure 20. Mach Number Contours - Gradual Start, $\alpha = 2°$, $t = 7.0$ - Expanded Scale

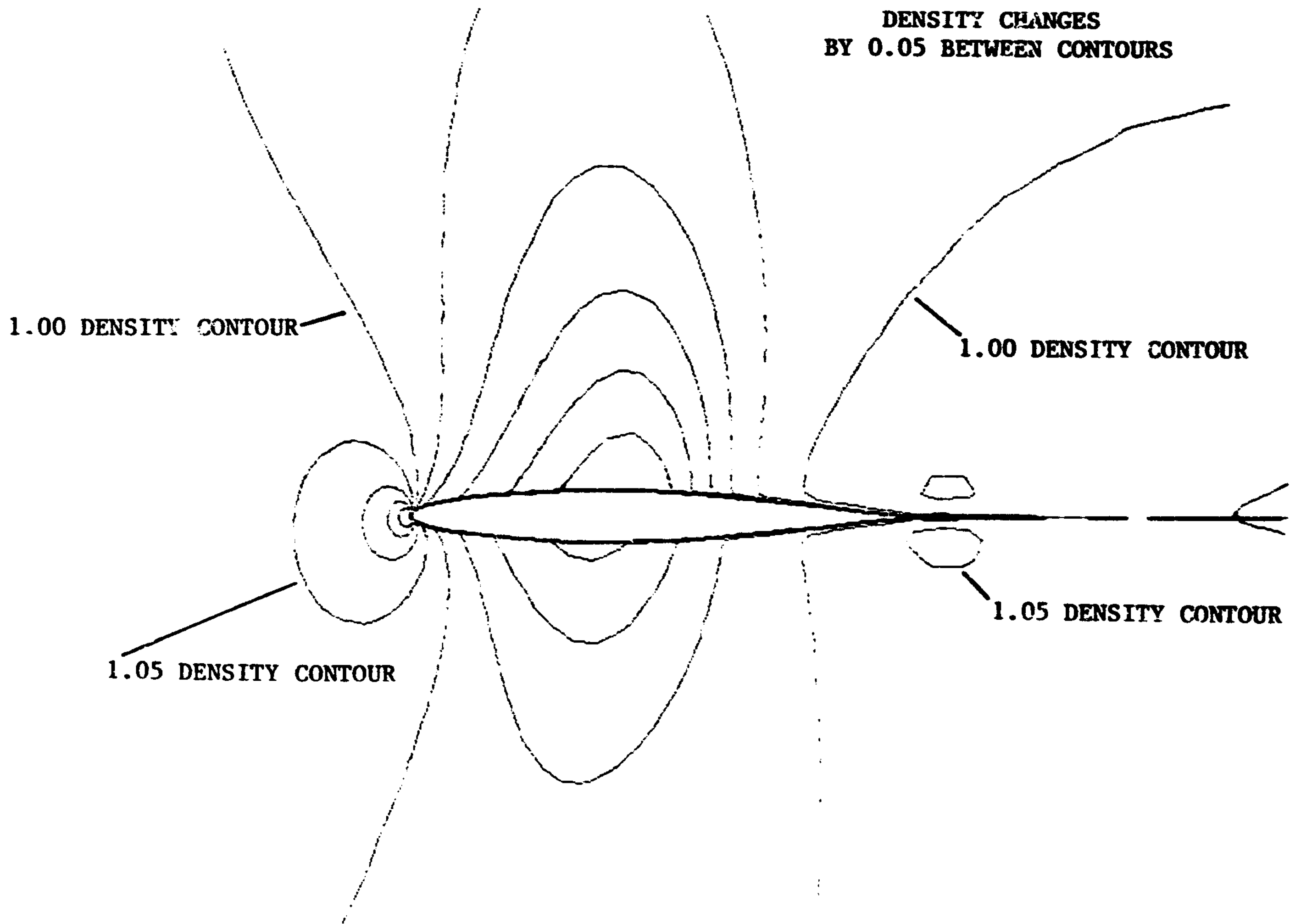


Figure 21. Density Contours - Gradual Start, $\alpha = 2°$, $t = 7.0$

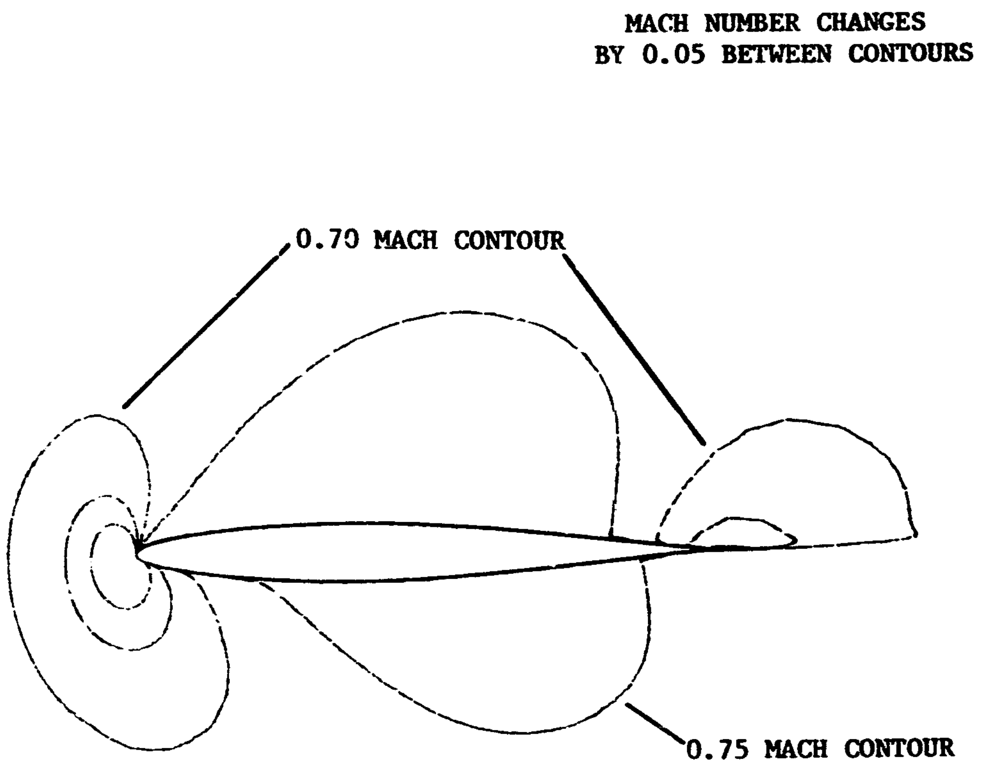


Figure 22. Mach Number Contours - Gradual Start,
$\alpha = 3.5°$, $t = 0.6$

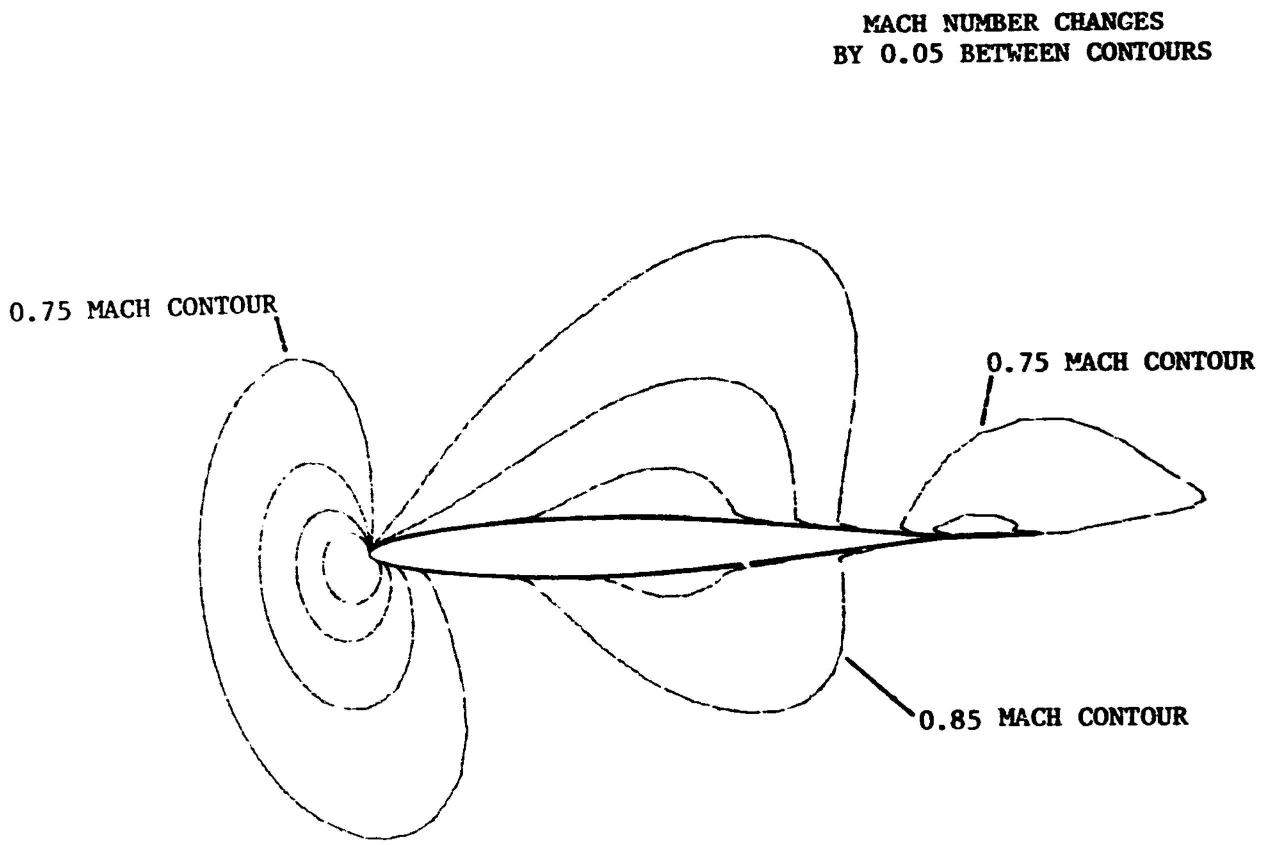


Figure 23. Mach Number Contours - Gradual Start, $\alpha = 3.5°$, $t = 1.0$

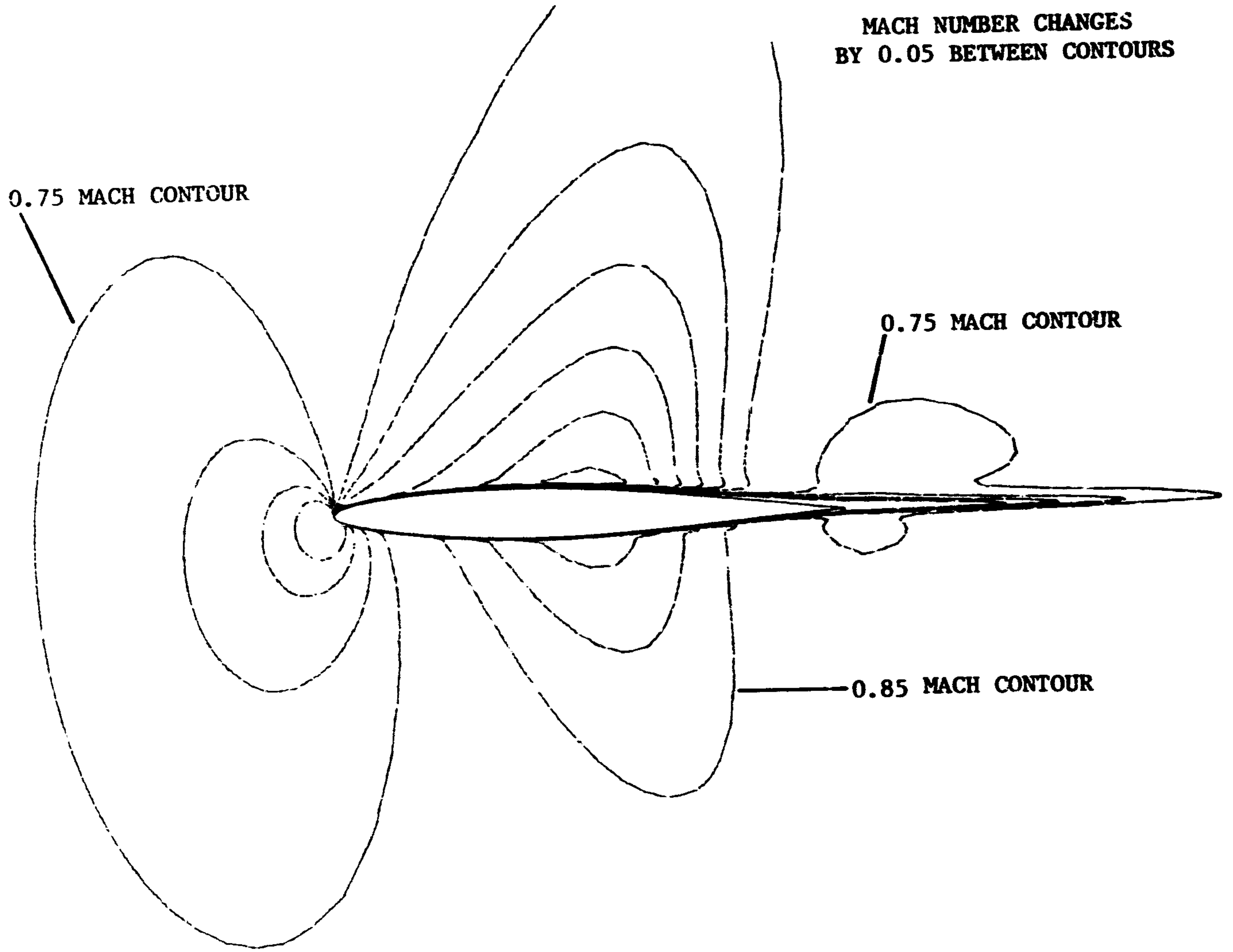


Figure 24. Mach Number Contours - Gradual Start, $\alpha = 3.5°$, $t = 2.0$

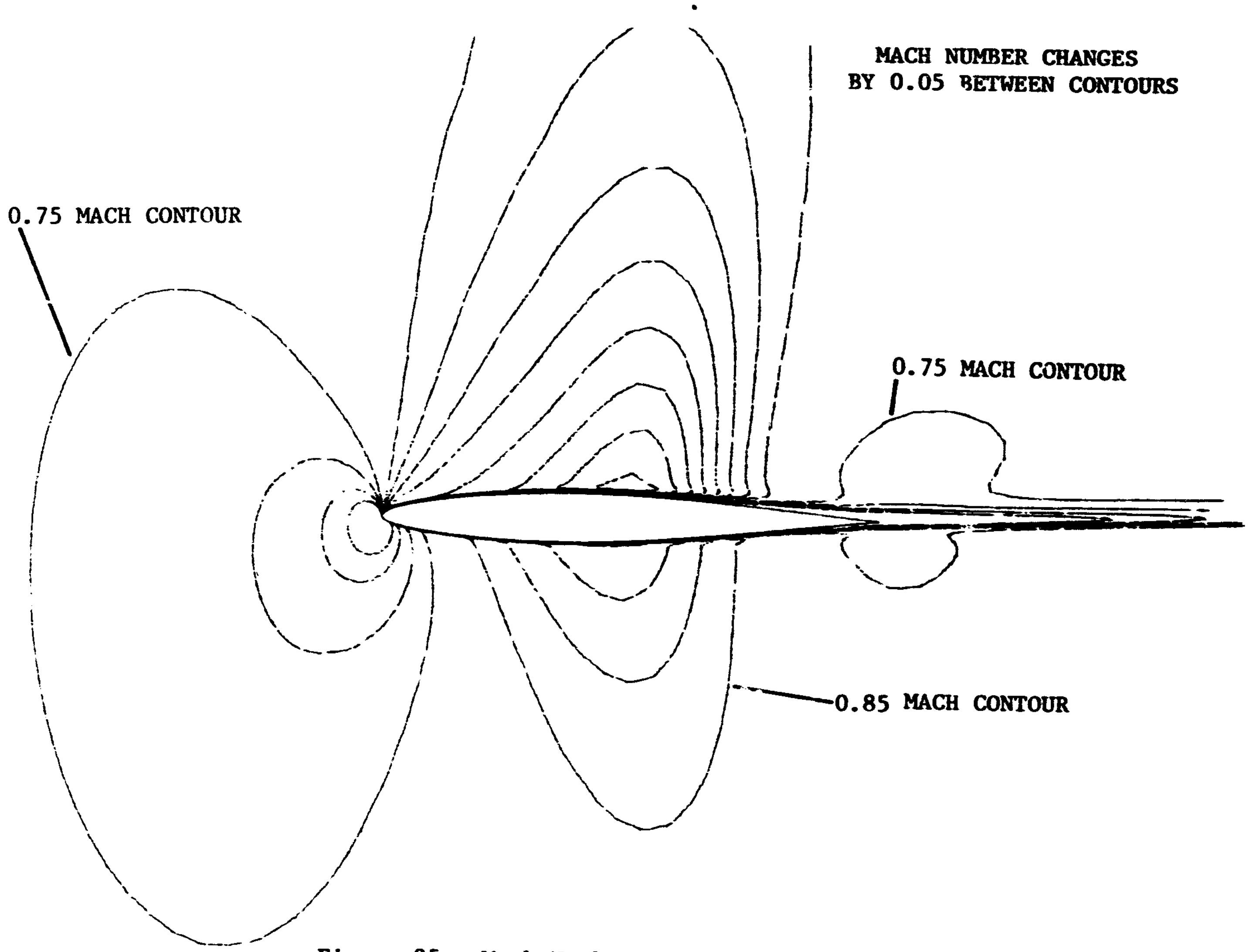


Figure 25. Mach Number Contours - Gradual Start,
$\alpha = 3.5°$, $t = 3.0$

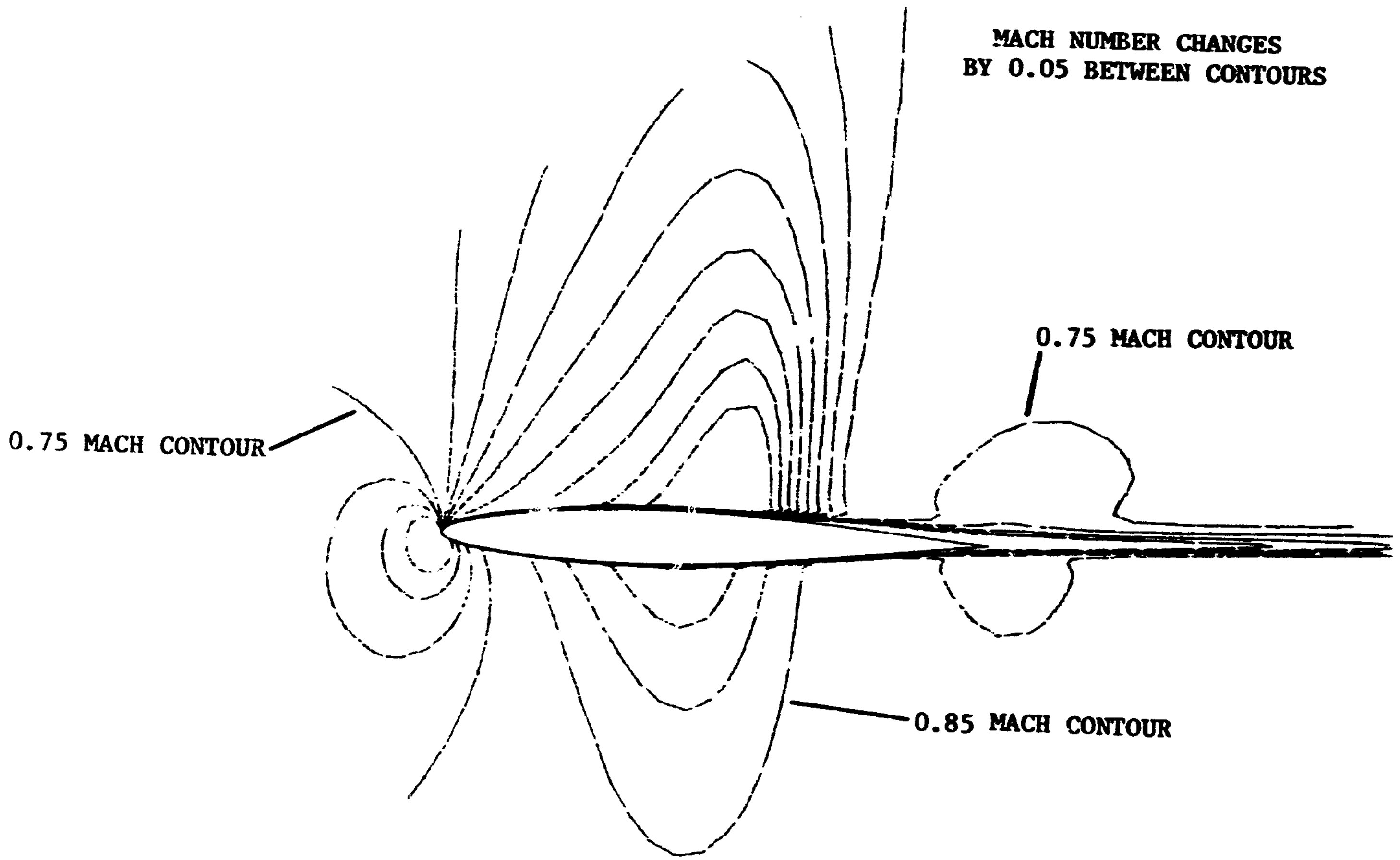


Figure 26. Mach Number Contours - Gradual Start,
$\alpha = 3.5°$, $t = 4.0$

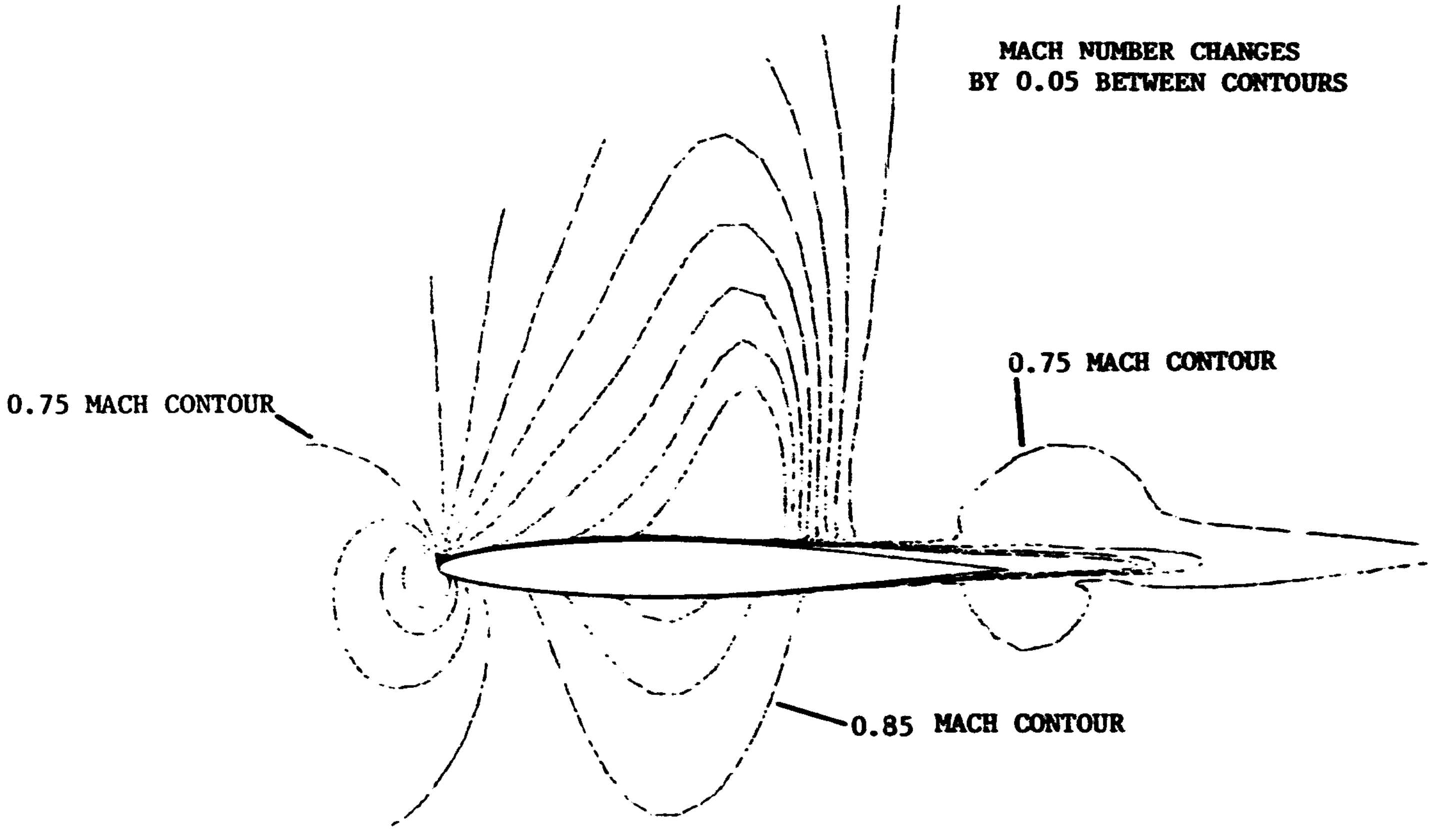


Figure 27. Mach Number Contours - Gradual Start,
$\alpha = 3.5°$, $t = 5.0$

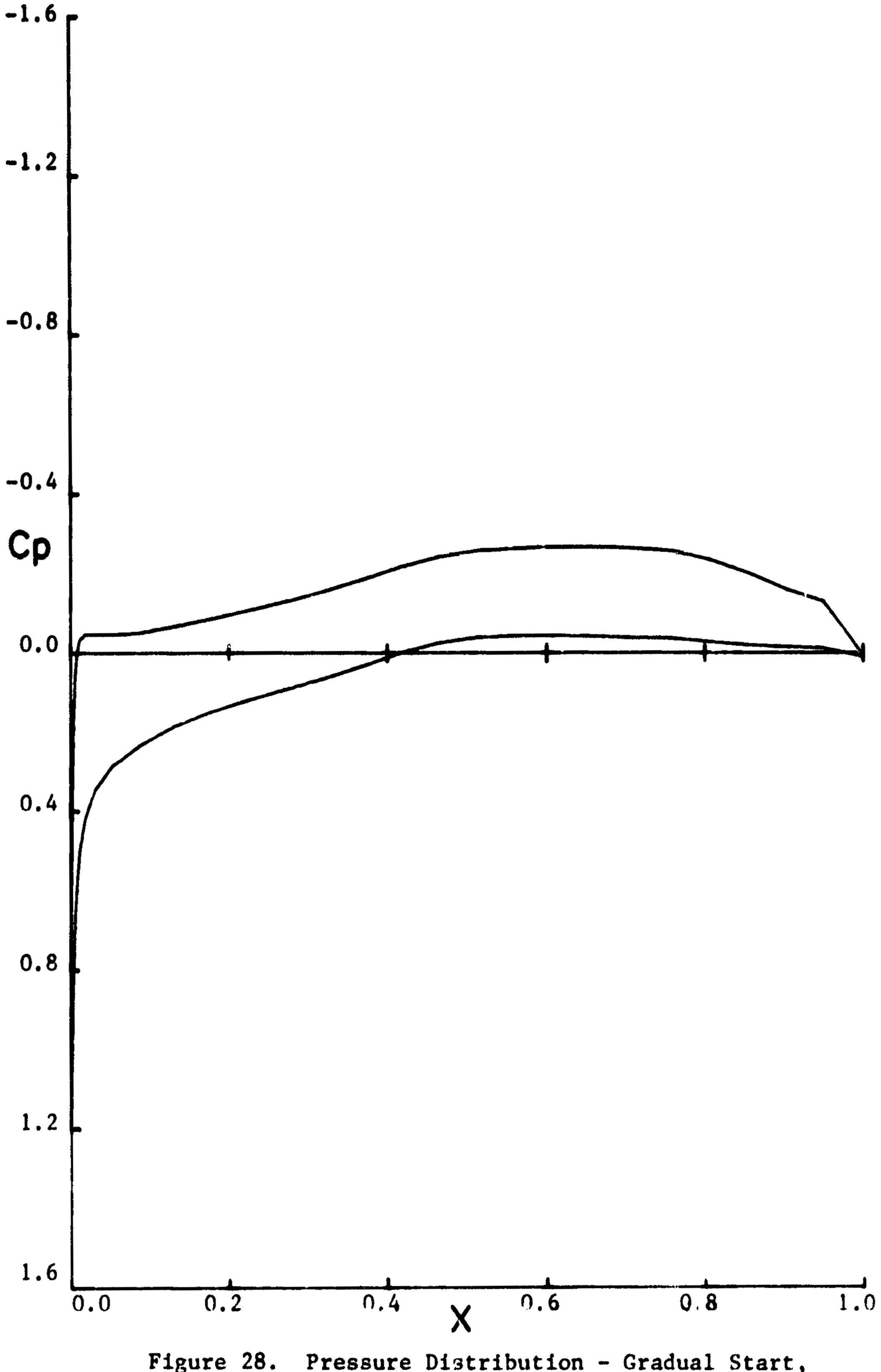


Figure 28. Pressure Distribution - Gradual Start, $\alpha = 3.5^\circ$, $\tau = 0.6$

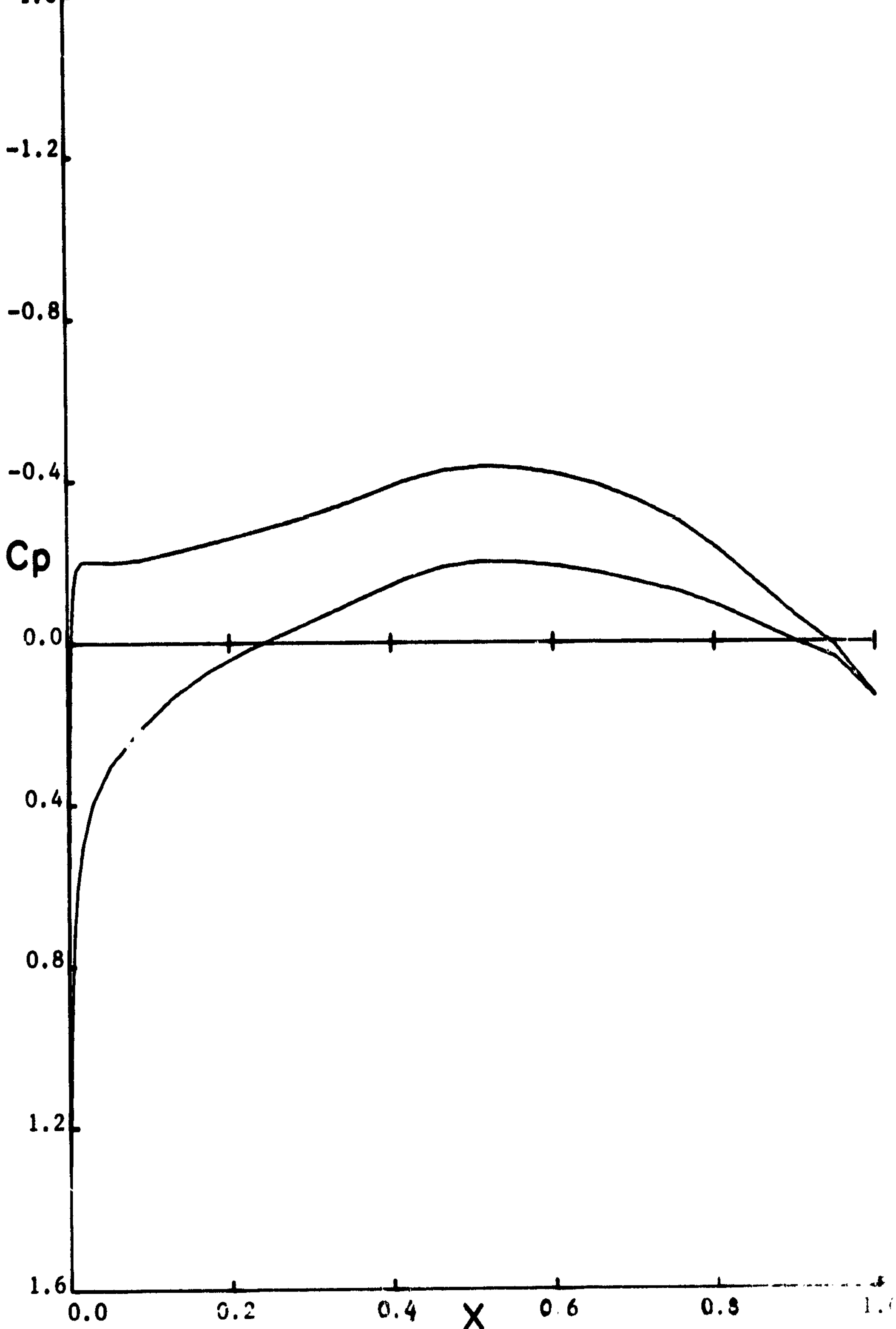


Figure 29. Pressure Distribution - Gradual Start, $\alpha = 3.5°$, $t = 1.0$

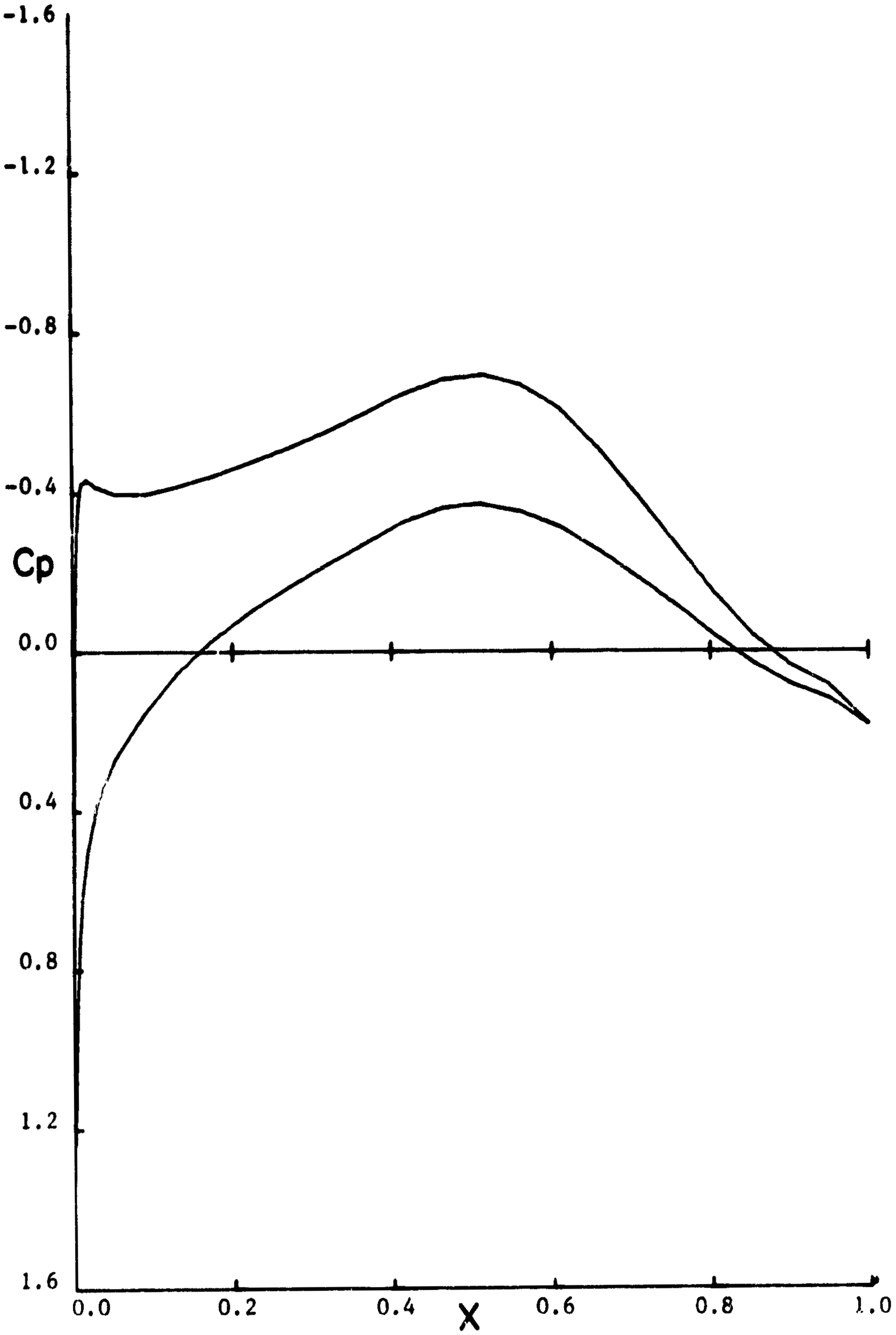


Figure 30. Pressure Distribution - Gradual Start, α = 3.5°, t = 2.0

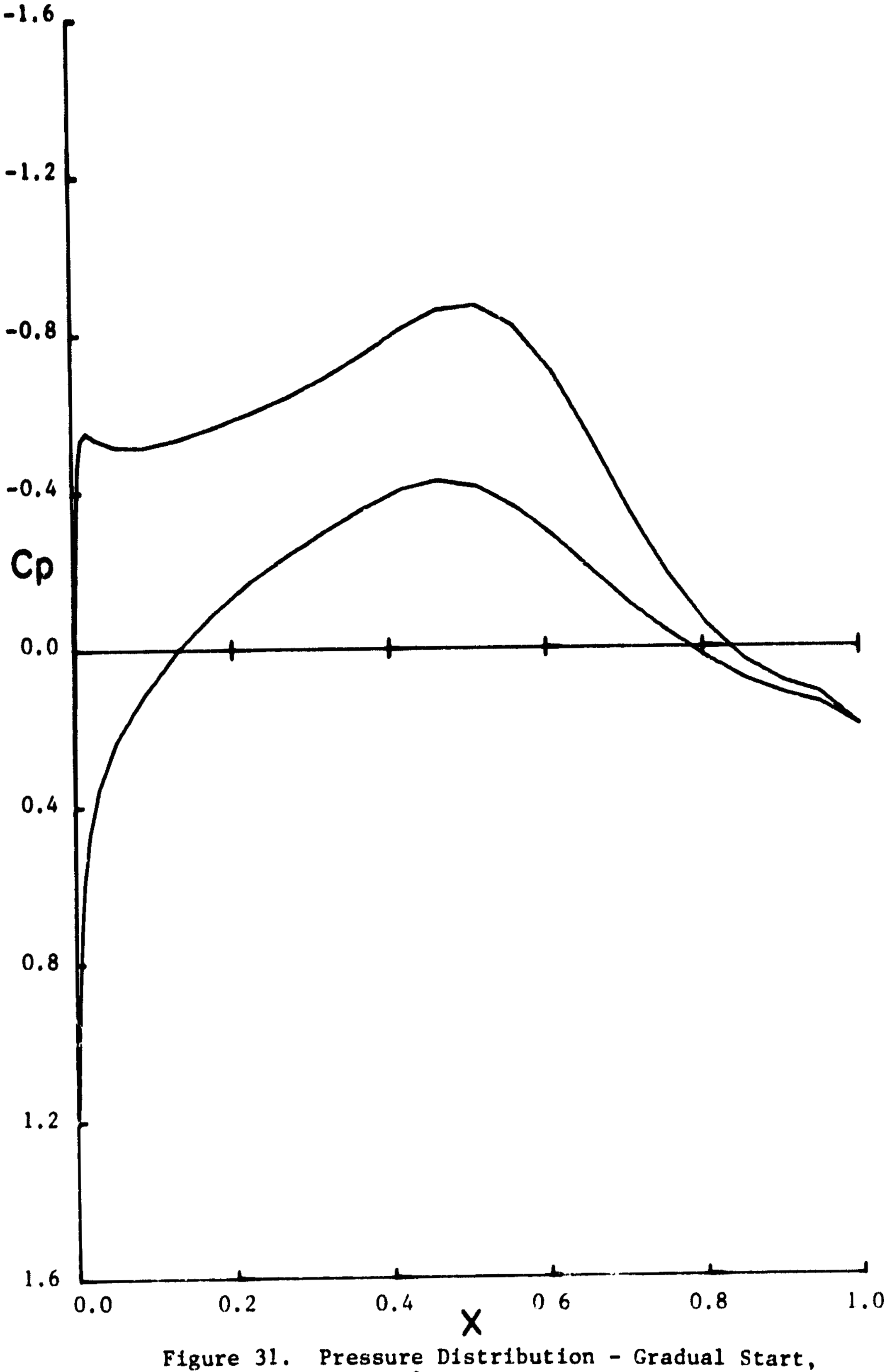


Figure 31. Pressure Distribution - Gradual Start,
α = 3.5°, t = 3.0

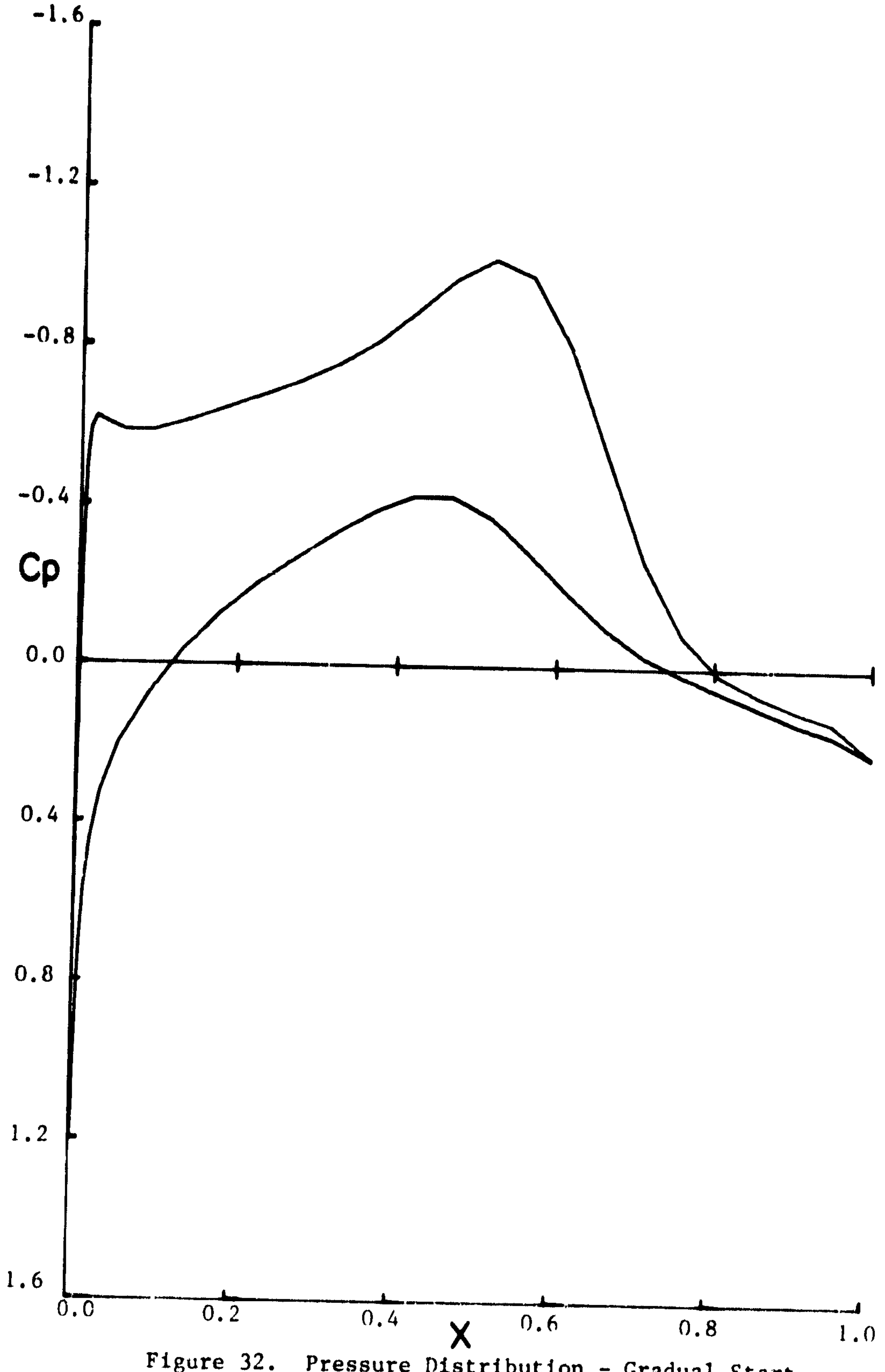


Figure 32. Pressure Distribution - Gradual Start, $\alpha = 3.5°$, $t = 4.0$

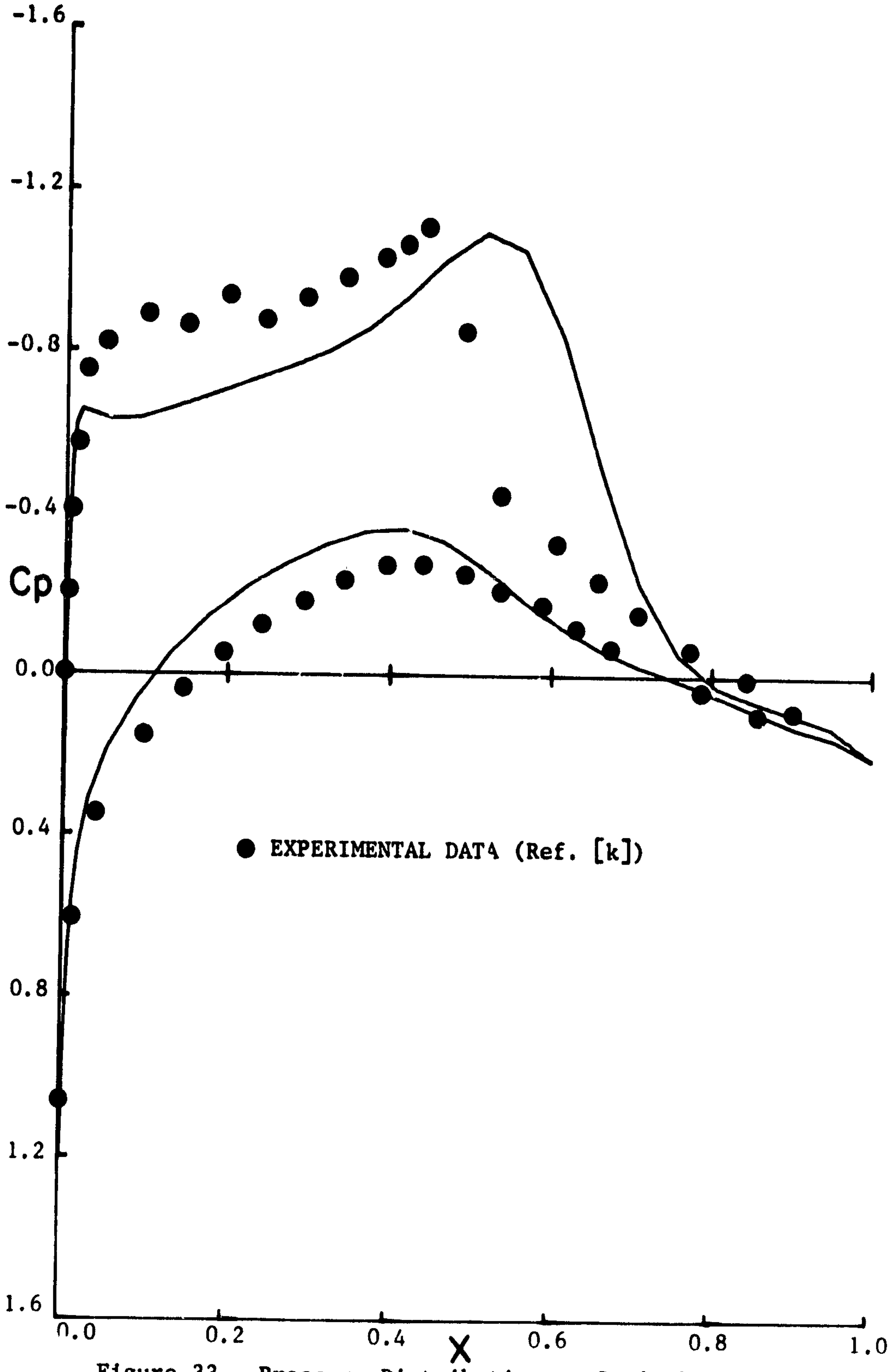


Figure 33. Pressure Distributions - Gradual Start and Experimental Data, α = 3.5, t · 5.0

APPENDIX A

Nondimensionalization

Although the use of nondimensional equations has many well-known advantages, it has the major draw-back that there is no unique dimensionless form for any given set of equations. Thus, two otherwise formally identical sets of equations may appear to be different solely due to the nondimensionalization used. Since, in this author's experience, the only consistent criteria for nondimensionalizing the Navier-Stokes equations is that the Reynolds number (Re) and Prandtl number (Pr) should appear in certain traditional locations, this has been taken as the primary objective of the nondimensionalization used in this work. The advantage of this criteria is that it keeps the set of partial differential equations from becoming cluttered; on the other hand, the suppressed dimensionless ratios now show up in the auxiliary algebraic relations and the boundary conditions.

All of the dimensionless variables and ratios ultimately are referenced relative to the following characteristic parameters:

Parameter	Name	Dimensions
ℓ	Length	L
u_o	Velocity	L/T
ρ_o	Density	M/L^3
μ_o	Shear Viscosity	M/L-T
k_o	Thermal Conductivity	$M\text{-}L/\theta\text{-}T^3$
C_p	Specific Heat at Constant Pressure	$L^2/\theta\text{-}T$
C_v	Specific Heat at Constant Volume	$L^2/\theta\text{-}T$

Parameter	Name	Dimensions
C_o	Speed of Sound	L/T

where the dimensions are length (L), time (T), mass (M), and temperature (θ). The speed of sound, C_o , is used only in situations where the Mach number would be more natural than a dimensionless temperature. These reference quantities are then used to form the following list of nondimensional variables or constants:

Nondimensional Variable	Reference Quantity
x,y	ℓ
u,v	u_o
ρ	ρ_o
μ,λ	μ_o
k	k_o
t	ℓ/u_o
P,e	$\rho_o u_o^2$
T	u_o^2/C_p
$\dot{q}''$	$\rho_o u_o^3$

with the Sutherland reference temperature, S_o , appended to the characteristic parameter list, the following nondimensional characteristic flow parameters can be formed:

$Re = \mu_o/\rho u_o \ell$	Reynolds number
$Pr = C_p \mu_o/k_o$	Prandtl number
$M = u_o/C_o$	Mach number
$\gamma = C_p/C_v$	Ratio of specific heats
$S_1 = CpS_o/u_o^2$	Sutherland reference temperature

APPENDIX B

Coordinate Transformation

It is widely accepted that any finite difference approximation to the Navier-Stokes equations should retain the global conservation properties of the integral form of these equations. This feature can be obtained, for usual differencing, only if the transformed equations retain the same general form as the differential Navier-Stokes equations in Cartesian coordinates. In this paper, this was accomplished in transforming equation (3.1) into equation (3.15) by use of the following relation (note the Einstein summation convention is used):

$$\frac{\partial A_i}{\partial x_i} = J\frac{\partial A_i^*}{\partial \xi_i} \tag{B.1}$$

where

$$A_i^* = \frac{1}{J}\,\frac{\partial \xi_i}{\partial x_j}\,A_j \tag{B.2}$$

and J is the Jacobian of the coordinate transformation. For the application under consideration, the vector components, A_i , and the coordinates, x_i and ξ_i , can be identified as:

$$\begin{array}{lll} A_1 = q & x_1 = t & \xi_1 = \tau \\ A_2 = A + D & x_2 = x & \xi_2 = \xi \\ A_3 = B + E & x_3 = y & \xi_3 = \eta \end{array} \tag{B.3}$$

and the Jacobian, J , is:

$$J = \begin{vmatrix} \tau_t & \tau_x & \tau_y \\ \xi_t & \xi_x & \xi_y \\ \eta_t & \eta_x & \eta_y \end{vmatrix} = \begin{vmatrix} \xi_x & \xi_y \\ \eta_x & \xi_y \end{vmatrix} = \xi_x\eta_y - \xi_y\eta_x \tag{B.4}$$

since $\tau \equiv t$.

Now relation (B.1) can be established by first developing the following chain of equalities:

$$\frac{\partial J}{\partial x_j} = C_{\ell k} \frac{\partial^2}{\partial x_j \partial x_k} \xi_\ell = C_{\ell k} \frac{\partial}{\partial x_k} \xi_i \frac{\partial^2}{\partial \xi_i \partial x_j} \xi_\ell$$

$$= J\delta_{i\ell} \frac{\partial^2}{\partial \xi_i \partial x_j} \xi_\ell = J \frac{\partial^2}{\partial \xi_i \partial x_j} \xi_i \qquad \text{(B.5)}$$

where $C_{\ell k}$ is the cofactor of the ℓ,k element of the Jacobian determinant (B.4) and the first equality is proved in most standard tensor treatments (e.g. Sokolnikoff [1], page 103). Then expansion of the right-hand-side of equation (B.1) gives:

$$J \frac{\partial}{\partial \xi_i} A_i^* = J \frac{\partial}{\partial \xi_i}\left(\frac{1}{J} A_j \frac{\partial}{\partial x_j} \xi_i\right) = -\frac{A_j}{J} \frac{\partial}{\partial x_j} J + A_j \frac{\partial^2}{\partial \xi_i \partial x_j} \xi_i + \frac{\partial}{\partial x_j} A_j \qquad \text{(B.6)}$$

which on use of the last equality of (B.5) reproduces equation (B.1).

APPENDIX C

Normal Pressure Derivative

The particular form of the normal pressure derivative equation used in this work (equation (3.33)) can most easily be developed from the Navier-Stokes equations in curvilinear coordinates, equations (3.21) through (3.26). First the x- and y- momentum equations are expanded to:

$$u[\rho^*_t+(\rho^*U)_\xi + (\rho^*V)_\eta] + \rho^*[u_t+Uu_\xi+Vu_\eta] + \frac{1}{J}[\xi_x P_\xi+\eta_x P_\eta] = R_1 \quad (C.1)$$

$$v[\rho^*_t+(\rho^*U)_\xi + (\rho^*V)_\eta] + \rho^*[v_t+Uv_\xi+Vv_\eta] + \frac{1}{J}[\xi_y P_\xi+\eta_y P_\eta] = R_2 \quad (C.2)$$

where $\rho^* = \rho/J$ and R_1, R_2 are the unexpanded viscous terms of the x- and y- momentum equations respectively. Now the first term in both equations (C.1) and (C.2)vanishes due to the continuity equation; thus, if equation (C.1) is multiplied through by η_x and equation (C.2) by η_y and then added, the result is:

$$\frac{1}{J}[\beta P_\xi+\alpha P_\eta]+[(\eta_x u_t+\eta_y v_t)+(\eta_x u_\xi+\eta_y v_\xi)U+(\eta_x u_\eta+\eta_y v_\eta)V]\rho^*$$

$$=\eta_x R_1 + \eta_y R_2 \quad (C.3)$$

Finally, if this equation is multiplied by $J/\sqrt{\alpha}$ and the equation of state, $P = (\gamma-1)\rho T/\gamma$, is used, then equation (3.33) results.

The finite difference form of equation (3.33), the normal pressure derivative equation, used in this research, equation (4.15), was put into the particular form chosen so as to facilitate the use of a common solution algorithm for pressure and temperature when the normal pressure derivative is specified (see equation (4.14)). The π terms, evaluated at i,1 are:

$$\pi_1 = (\frac{\gamma}{\gamma-1})(\frac{2}{\alpha T})[(\eta_x u_t + \eta_y v_t)+(\eta_x u_\xi + \eta_y v_\xi)U + (\eta_x u_\eta + \eta_y v_\eta)V] \quad (C.4)$$

$$\pi_2 = 2\,\frac{J}{\alpha}(\eta_x R_1+\eta_y R_2) \quad (C.5)$$

The derivatives are approximated by the appropriate finite difference expression from Appendix G and evaluated prior to the solution of equation (4.15).

APPENDIX D

Boundary Fitted Coordinates

Since all of the problems considered in this paper possess single body, two-dimensional geometries, and since this class of coordinate generation problem has been extensively examined in the literature (see [d,e,h,j], for example), this appendix will be devoted to clarification of equations (3.37) and (3.38) and a summary of the coordinate contraction technique discussed more completely in reference [j]. The symbols used in the aforementioned equations are defined to be:

$$\alpha_c = x_\eta^2 + y_\eta^2 \tag{D.1}$$

$$\beta_c = x_\xi x_\eta + y_\xi y_\eta \tag{D.2}$$

$$\gamma_c = x_\xi^2 + y_\xi^2 \tag{D.3}$$

$$J_c = J^{-1} = x_\xi y_\eta - x_\eta y_\xi \tag{D.4}$$

The boundary conditions on the coordinate generation equations (3.37) and (3.38) are determined by specifying the ξ and η distribution on the body and the inflow and outflow boundaries. As mentioned in the text, normally this is done so that each boundary is a line of constant ξ or η. The system of partial differential equations is then solved by approximating all derivatives by "second order" central differences (see Appendix G) and using a point successive over-relaxation iteration scheme to determine the solution of the resulting nonlinear simultaneous difference equations.

Of course, before the coordinate generation equations can be solved, the coordinate line attraction functions, P_c and Q_c, must be specified. The general effect that these functions can have on the resulting coordinate system, and cautions about the forms they can take are fully developed in reference [d]; it is enough to note here that these attraction functions can have a dramatic effect on the coordinate spacing near a boundary. Since the high Reynolds number flows considered in this work are very dependent on boundary layer interactions, it was desired that the coordinate spacing be very dense inside the boundary layers and, thus, near the body boundary. As noted in references [j,k], this can be readily accomplished by taking the appropriate attraction functions (Q_c if the body surface is a line of constant η) as:

$$Q_c(\xi,\eta) = -\frac{(2+\eta\ln K)}{(1+\eta\ln K)}\ln K \qquad (D.5)$$

where the spacing parameter K is determined from:

$$\frac{NK^{N-1}-1}{JK^{J-1}-1} = \frac{\ln(\frac{r_N}{r_1})}{\ln(\frac{r_J}{r_1})} \qquad (D.6)$$

The parameters in this last expression are defined so that if the body surface coincides with the first η-line, then r_1 is the radius of a circle which circumscribes it, and if the outer boundary lies on the Jth η-line, then r_J is the radius of a circle which is tangent to it. This set of equations has the effect that the Nth η-line will be approximately a distance $r_N - r_1$ from the body surface when they are used in the coordinate generation scheme. In this study, the 2nd η-line was specified to be one-hundredth of the boundary layer thickness from the body. So, by approximating the boundary layer thickness from the laminar Blasius flat-plate boundary layer solution, this is equivalent to:

$$r_N = 0.01 \frac{5}{\sqrt{Re}} + r_1 \qquad (D.7)$$

APPENDIX E

Flux Vectors and Their Jacobian Matrices

Since the dynamic flux vectors (A and B) and the viscous flux vectors (D_1,D_2,E_1 and E_2) are similar in form within each class, let the vectors A and D be representative of each class in the following discussion. If for the present, the transformed coordinates (ξ and η) are taken to be constant in time, then the chain-rule may be used to form the following identities:

$$A_t = Pq_t \tag{E.1}$$

$$D_t = \pi q_t + Wq_{\xi t} \tag{E.2}$$

where the Jacobian matrices P, π, and W are defined to be

$$P = \frac{\partial A}{\partial q}, \quad \pi = \frac{\partial D}{\partial q}, \quad W = \frac{\partial D}{\partial q_\xi} \tag{E.3}$$

Now the second identity (E.2) can be put in the form:

$$D_t = (\pi - W_\xi)q_t + (Wq_t)_\xi \tag{E.4}$$

If the viscous coefficient (i.e., μ times expressions (3.10) and (3.11)) are taken to be locally constant in time, which ultimately introcudes an error proportional to their first time derivative times Δt^2, then it can be shown by direct evaluation that $\pi = W_\xi$, and from equation (E.4):

$$D_t = (Wq_t)_\xi \tag{E.5}$$

Equations (4.5) then directly follow from

$$\Delta A \simeq A_t \Delta t = Pq_t \Delta t \simeq P\Delta q + O(\Delta t^2) \tag{E.6}$$

$$\Delta D \simeq D_t \Delta t \simeq (Wq_t)_\xi \simeq (W\Delta q)_\xi + O(\Delta t^2, \mu_t \Delta t) \tag{E.7}$$

Under the same a..umption as used in establishing equation (E.5), equation sets (4.6) and (4.7) can also be confirmed by direct evaluation. It is interesting to note that for any flow in which the viscosity is solely a function of temperature (such as laminar flow with a Sutherland viscosity law), then the flux vectors are homogeneous functions of various degrees in the conservation vector, q , and its first derivatives; thus equations (4.6) and other interesting relationships can be easily shown.

As a final note, if the transformed coordinates are known functions of time, or at least are advanced in time prior to solving for the flow variables, then it can be shown that if the derivatives $(\xi_t, \xi_x, \xi_y, \eta_t, \eta_x, \eta_y)$ are evaluated at time $(n+\theta)\Delta t$, where θ is as introduced in equation (4.1), then these derivatives can be treated as if they were local'y constant with time without degrading the accuracy of the approximate factorization scheme.

APPENDIX F

Explicit Form of the Jacobian Matrices

These matrices are found by a straight-forward, if rather tedious, process of re-expressing the vectors of equations (3.22) through (3.26) explicitly in terms of the conservation variables ($q_1 = \rho/J$, $q_2 = \rho u/J$, $q_3 = \rho v/J$, and $q_4 = e/J$, and then preforming the differentiations indicated by equations (4.5). They are:

$$P = \begin{bmatrix} 0 & \xi_x & \xi_y & 0 \\ \xi_x\Phi-uU & U-(\gamma-2)\xi_x u & \xi_y u-(\gamma-1)\xi_x v & (\gamma-1)\xi_x \\ \xi_y\Phi-vU & \xi_x v-(\gamma-1)\xi_y u & U-(\gamma-2)\xi_y v & (\gamma-1)\xi_y \\ (\Phi-T_s)U & \xi_x T_s-(\gamma-1)uU & \xi_y T_s-(\gamma-1)vU & \gamma U \end{bmatrix} \qquad (F.1)$$

$$Q = \begin{bmatrix} 0 & \eta_x & \eta_y & 0 \\ \eta_x\Phi-uV & V-(\gamma-2)\eta_x u & \eta_y u-(\gamma-1)\eta_x v & (\gamma-1)\eta_x \\ \eta_y\Phi-vV & \eta_x v-(\gamma-1)\eta_y u & V-(\gamma-2)\eta_y v & (\gamma-1)\eta_y \\ (\Phi-T_s)V & \eta_x T_s-(\gamma-1)uV & \eta_y T_s-(\gamma-1)vV & \gamma V \end{bmatrix} \qquad (F.2)$$

$$W = \frac{1}{\rho}\begin{bmatrix} 0 & 0 & 0 & 0 \\ -(a_1u+a_2v) & a_1 & a_2 & 0 \\ -(a_2u+a_3v) & a_2 & a_3 & 0 \\ -(W_{42}u+W_{43}v+a_4E) & -(W_{21}+a_4u) & -(W_{31}+a_4v) & a_4 \end{bmatrix} \qquad (F.3)$$

$$X = \frac{1}{\rho}\begin{bmatrix} 0 & 0 & 0 & 0 \\ -(b_1u+b_2v) & b_1 & b_2 & 0 \\ -(b_3u+b_4v) & b_3 & b_4 & 0 \\ -(X_{42}u+X_{43}v+b_5E) & -(Y_{21}+b_5u) & -(Y_{31}+b_5u) & b_5 \end{bmatrix} \qquad (F.4)$$

$$Y = \frac{1}{\rho}\begin{bmatrix} 0 & 0 & 0 & 0 \\ -(b_1u+b_3v) & b_1 & b_3 & 0 \\ -(b_2u+b_4v) & b_2 & b_4 & 0 \\ X_{41} & -(X_{21}+b_5u) & -(X_{31}+b_5v) & b_5 \end{bmatrix} \qquad (F.5)$$

$$Z = \frac{1}{\rho}\begin{bmatrix} 0 & 0 & 0 & 0 \\ -(c_1u+c_2v) & c_1 & c_2 & 0 \\ -(c_2u+c_3v) & c_2 & c_3 & 0 \\ -(Z_{42}u+Z_{43}v+c_4E) & -(Z_{21}+c_4u) & -(Z_{31}+c_4v) & c_4 \end{bmatrix} \qquad (F.6)$$

where the contravariant velocities, U and V, are given by equations (3.28) and (3.29) and the viscous coefficients are the same as those defined by equations (3.30). The subscripted Jacobian matrix symbols indicate the appropriate element in their explicit expansion (e.g., $W_{21} = -(a_1u+a_2v)$). The other symbols introduced here (Φ, T_s and E) are defined to be:

$$\begin{aligned} \Phi &= \frac{1}{2}(\gamma-1)(u^2+v^2) \\ T_s &= \gamma E-\Phi \\ E &= e/\rho \end{aligned} \qquad (F.7)$$

APPENDIX G

Difference Approximations

The finite difference expressions used in this work are presented below. In all cases the functions $\alpha(\xi,\eta)$ and $f(\xi,\eta)$ are dummy functions representing typical actual derivative expressions encountered in the development of the text. Equations which have an obvious analog when the variable of differentiation or the order of differentiation is changed have not been repeated.

Central Differences:

$$\begin{aligned}
f_{\xi} &= \tfrac{1}{2}(f_{i+1}-f_{i-1}) \\
f_{\xi\xi} &= f_{i+1}-2f_{i}+f_{i-1} \\
f_{\xi\eta} &= \tfrac{1}{4}(f_{i+1,j+1}-f_{i+1,j-1}-f_{i-1,j+1}+f_{i-1,j-1}) \qquad \text{(G.1)} \\
(\alpha f_{\xi})_{\xi} &= \tfrac{1}{2}[(\alpha_{i+1}+\alpha_{i})(f_{i+1}-f_{i})-(\alpha_{i}+\alpha_{i-1})(f_{i}-f_{i-1})] \\
(\alpha f_{\eta})_{\xi} &= \tfrac{1}{4}[\alpha_{i+1,j}(f_{i+1,j+1}-f_{i+1,j-1})-\alpha_{i-1,j}(f_{i-1,j+1}-f_{i-1,j-1})]
\end{aligned}$$

One-Sided Differences:

$$\begin{aligned}
f_{\xi} &= \tfrac{1}{2}(f_{i-2}-4f_{i-1}+3f_{i}) = -\tfrac{1}{2}(f_{i+2}-4f_{i+1}+3f_{i}) \\
f_{\xi\xi} &= -(f_{i-3}-4f_{i-2}+5f_{i-1}-2f_{i}) = -(f_{i+3}-4f_{i+2}+5f_{i+1}-2f_{i}) \\
(\alpha f_{\eta})_{\eta} &= \tfrac{1}{4}(\alpha_{j+2}-4\alpha_{j+1}+3\alpha_{j})(f_{j+2}-4f_{j+1}+3f_{j}) \\
&\quad -\alpha_{j}(f_{j+3}-4f_{j+2}+5f_{j+1}-2f_{j})
\end{aligned}$$

$$\begin{aligned}(\alpha f_\eta)_\xi = & -\frac{1}{2}\{(\alpha_{i+1,j}-\alpha_{i-1,j})(f_{i,j+2}-4f_{i,j+1}+3f_{i,j}) \\ & + \alpha_{i,j}[(f_{i+1,j+2}-4f_{i+1,j+1}+3f_{i+1,j}) \\ & -(f_{i-1,j+2}-4f_{i-1,j+1}+3f_{i-1,j})]\} \\ (\alpha f_\xi)_\eta = & -\frac{1}{2}\{(\alpha_{i,j+2}-4\alpha_{i,j+1}+3\alpha_{i,j})(f_{i+1,j}-f_{i-1,j}) \\ & + \alpha_{i,j}[(f_{i+1,j+2}-4f_{i+1,j+1}+3f_{i+1,j}) \\ & -(f_{i-1,j+2}-4f_{i-1,j+1}+3f_{i-1,j})]\}\end{aligned} \tag{G.2}$$

Extrapolations:

$$\begin{aligned} f_i &= 2f_{i+1}-f_{i+2} = 2f_{i-1}-f_{i-2} \\ f_i &= \frac{1}{2}(f_{i+1}+f_{i-1}) \\ f_i &= -\frac{1}{2}(f_{i+2}-2f_{i+1}-2f_{i-1}+f_{i-2}) \\ f_i &= -\frac{1}{6}(f_{i+2}-4f_{i+1}-4f_{i-1}+f_{i-2}) \end{aligned} \tag{G.3}$$

Although all of these expressions, except the last one, are in the form which is traditionally termed "second order accurate", since the coordinate transformation is always chosen so that $\Delta\xi=\Delta\eta=1$, the assignment of an order of accuracy to these expansions becomes ambiguous at best. For example, the central difference first derivative approximation has as its leading term in the truncation error the quantity: $-\frac{1}{6}f_{\xi\xi\xi}$; but, by itself, this does not give a meaningful approximation of the truncation error. There have been some attempts to develop estimates of the truncation error in curvilinear coordinates which are analogous to the well-known Cartesian error estimates [b,h], but more work is needed in this area.

APPENDIX H

Block-Tridiagonal Inversion Algorithm

This algorithm is basically identical to the one presented by Steger [b] in his Appendix 1. Since the method is widely available in the literature, it is sketched out below purely for the convenience of the reader.

Given the block tridiagonal system of equations: Bu = f, or explicitly:

$$\begin{bmatrix} B_1 & C_1 & & & & \\ A_2 & B_2 & C_2 & & & \\ & A_3 & B_3 & C_3 & & \\ & & \cdot & \cdot & \cdot & \\ & & & \cdot & \cdot & \cdot \\ & & & & \cdot & \cdot \quad \cdot \\ & & & A_{N-1} & B_{N-1} & C_{N-1} \\ & & & & A_N & B_N \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ \cdot \\ \cdot \\ \cdot \\ u_{N-1} \\ u_N \end{bmatrix} = \begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ \cdot \\ \cdot \\ \cdot \\ f_{N-1} \\ f_N \end{bmatrix} \qquad (H.1)$$

where each A_i, B_i, and C_i symbol represents a k x k matrix and the vectors u_i and f_i are k-dimensional, the solution may be obtained by a systematic application of Gaussian elimination without pivoting. The solution scheme can most clearly be developed by splitting the matrix B up into a product of a lower block triangular matrix L and an upper block triangular matrix U of the following form:

$$B = LU = \begin{bmatrix} G_1 & & & & & \\ A_2 & G_2 & & & & \\ & A_3 & G_3 & & & \\ & & \ddots & \ddots & & \\ & & & A_{N-1} & G_{N-1} & \\ & & & & A_N & G_N \end{bmatrix} \begin{bmatrix} I & U_1 & & & & \\ & I & U_2 & & & \\ & & I & U_3 & & \\ & & & \ddots & \ddots & \\ & & & & I & U_{N-1} \\ & & & & & I \end{bmatrix} \tag{H.2}$$

and then finding the solution to the set of matrix equations:

$$\begin{aligned} Lr &= f \\ Uu &= r \end{aligned} \tag{H.3}$$

This can be efficiently accomplished by use of the following recursive algorithm:

Forward Sweep

```
1)      $U_1 = B_1^{-1} C_1$
        $r_1 = B_1^{-1} f_1$
2) for  $i = 2$ to $i = N-1$
        $G_i = B_i - A_i U_{i-1}$
        $U_i = G_i^{-1} C_i$                                   (H.4)
        $r_i = G_i^{-1}(f_i - A_i r_{i-1})$
```

Back Sweep

```
1)      $G_N = B_N - A_N U_{N-1}$
        $u_N = G_N^{-1}(f_N - A_N r_{N-1})$
2) for $i = N-1$ to $i = 1$
        $u_i = r_i - U_i u_{i+1}$
```

As Steger points out in the previously cited reference, this algorithm can accommodate the addition of several stray matrices off of the tridiagonal strip of the matrix B, but only at the cost of more than doubling the operation count. Also, following the recommendation of Steger, the k x k (here 4 x 4) matrix inversion involved in the above solution scheme can be efficiently calculated by application of the same systemized Gaussian elimination scheme as used in this algorithm.

Suppose the matrix equation GS = M is to be solved for S, where G, S, and M are all 4 x 4 matrices. Let A and B be lower and upper triangular matrices, respectively, so that

$$G = AB = \begin{bmatrix} a_{11} & 0 & 0 & 0 \\ a_{21} & a_{22} & 0 & 0 \\ a_{31} & a_{32} & a_{33} & 0 \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix} \begin{bmatrix} 1 & b_{12} & b_{13} & b_{14} \\ 0 & 1 & b_{23} & b_{24} \\ 0 & 0 & 1 & b_{34} \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad \text{(H.5)}$$

Then, similar to equation (H.3), the set of equations

$$\begin{aligned} AR &= M \\ BS &= R \end{aligned} \qquad \text{(H.6)}$$

can be solved by the following algorithm:

$$a_{11} = g_{11}$$

$$a_{21} = g_{21}$$

$$a_{31} = g_{31}$$

$$a_{41} = g_{41}$$

$$b_{12} = g_{12}/a_{11}$$

$$a_{22} = g_{22} - b_{12}a_{21}$$

$$a_{32} = g_{32} - b_{12}a_{31}$$

$$a_{42} = g_{42} - b_{12}a_{41}$$

$$b_{13} = g_{13}/a_{11}$$

$$b_{23} = (g_{23} - b_{13}a_{21})/a_{22}$$

$$a_{33} = g_{33} - b_{13}a_{31} - b_{23}a_{32}$$

$$a_{43} = g_{43} - b_{13}a_{41} - b_{23}a_{42}$$

$$b_{14} = g_{14}/a_{11}$$

$$b_{24} = (g_{24} - b_{14}a_{21})/a_{22}$$

$$b_{34} = (g_{34} - b_{14}a_{31} - b_{24}a_{32})/a_{33}$$

$$a_{44} = g_{44} - b_{14}a_{41} - b_{24}a_{42} - b_{34}a_{43}$$

Repeat for each column of M matrix ($j = 1$ to $j = 4$ in this case):

$$r_{1j} = m_{1j}/a_{11}$$

$$r_{2j} = (m_{2j} - a_{21}r_{1j})/a_{22}$$

$$r_{3j} = (m_{3j} - a_{31}r_{1j} - a_{32}r_{2j})/a_{33}$$

$$r_{4j} = (m_{4j} - a_{41}r_{1j} - a_{42}r_{2j} - a_{43}r_{3j})/a_{44}$$

$$s_{4j} = r_{4j}$$

$$s_{3j} = r_{3j} - b_{34}s_{4j}$$

$$s_{2j} = r_{2j} - b_{24}s_{4j} - b_{23}s_{3j}$$

$$s_{1j} = r_{1j} - b_{14}s_{4j} - b_{13}s_{3j} - b_{12}s_{2j}$$

APPENDIX I

Body Force Modifications of Algorithm

Inclusion of a body force term is readily done by adding the vector

$$G = \begin{bmatrix} 0 \\ \rho g_1 \\ \rho g_2 \\ \rho u g_1 + \rho v g_2 \end{bmatrix} \tag{I.1}$$

to the right-hand-side of equation (3.1) and the corresponding vector $G^* = G/J$ to the right-hand-side of equation (3.15). Where g_1 and g_2 are nondimensional acceleration parameters whose effect is to accelerate the flow field in the positive x- and y-coordinate directions, or equivalently, to accelerate the body and its attached coordinates in the negative x- and y-coordinate directions. The generalized time-differenced Navier-Stokes equations (4.4) are then accordingly modified by adding the term

$$\frac{\theta \Delta t}{1+\phi} \Delta G^n + \frac{\Delta t}{1+\phi} G^n \tag{I.2}$$

where the superscript "*" has been dropped. The Jacobian matrix corresponding to the body force vector G is:

$$M = \frac{\partial G}{\partial q} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ g_1 & 0 & 0 & 0 \\ g_2 & 0 & 0 & 0 \\ 0 & g_1 & g_2 & 0 \end{bmatrix} \tag{I.3}$$

So if, as is discussed in Appendix E in regard to the coordinate derivatives, the acceleration parameters g_1 and g_2 are evaluated at time $(n+\theta)\Delta t$, then equation (4.8) is modified by adding $\frac{\Delta t}{1+\phi} G^n$ to the right hand side and $- \frac{\theta\Delta t}{1+\phi} M^{n+\theta}\Delta q^n$ to the left hand side. The final algorithm expressed in equations (4.9) can now be adjusted to take the following form:

$$\{I + \frac{\theta\Delta t}{1+\phi}[-M^{n+\theta} + \frac{\partial}{\partial\xi} P^n - \frac{1}{Re}\frac{\partial^2}{\partial\xi^2} W^n]\}\overline{\Delta q^n} = \frac{\Delta t}{1+\phi} G^n + RHS(4.8) \tag{I.4a}$$

$$\{I + \frac{\theta\Delta t}{1+\phi}[-M^{n+\theta} + \frac{\partial}{\partial\eta} Q^n - \frac{1}{Re}\frac{\partial^2}{\partial\eta^2} Z^n]\}\Delta q^n = (I-M^{n+\theta})\overline{\Delta q^n} \tag{I.4b}$$

$$q^{n+1} = q^n + \Delta q^n \tag{I.4c}$$

where, it should be emphasized that the acceleration factors g_1 and g_2 are to be evaluated at time $(n+\theta)\Delta t$ in the vector G^n.

The last set of equations may not seem to directly follow from the modified form of equation (4.8); that it does, and to the same order of accuracy, can be seen from the following discussion. Consider the symbolic form of equation (4.8) as modified by the body force term:

$$[I + \Delta t(M+P+Q)]\Delta q = R \tag{I.5}$$

where M, P, Q, and R represent the body force, ξ-derivative, η-derivative, and right-hand-side terms, respectively. Now, it was considered to be desirable to factor equation (I.5) in such a way that the body force matrix M would have an equivalent effect on the implicit matrices for both the ξ- and η-directions. This can be obtained if equation

(I.5) is multiplied by the matrix $S = (I + \Delta tM)^{-1}$, resulting in the equation:

$$[I + \Delta t(SP + SQ)]\Delta q = SR \qquad (I.6)$$

which can then be factored into the set of equations

$$(I + \Delta tSP)\overline{\Delta q} = SR \qquad (I.7a)$$

$$(I + \Delta tSQ)\Delta q = \overline{\Delta q} \qquad (I.7b)$$

to order of Δt^3. Finally, multiplying these equations by S^{-1} results in:

$$[I + \Delta t(M + P)]\overline{\Delta q} = R \qquad (I.8a)$$

$$[I + \Delta t(M + Q)]\Delta q = (I + \Delta tM)\overline{\Delta q} \qquad (I.8b)$$

But these last equations are seen to be the symbolic form of equation (I.4a and b).

As a final note, although the author was unable to prove this, it was felt that the magnitude of the acceleration parameters should be such that strong diagonal dominance of the implicit matrices is preserved. This results in the criteria

$$\underset{n}{\mathrm{Max}}(|g_1^n| \ , \ |g_2^n|) < \frac{1+\phi}{\theta} \frac{1}{\Delta t} \qquad (I.9)$$

which tended to be confirmed by numerical experimentation. Note that this criteria, if correctly conceived, means that large accelerations require small time steps and goes some way toward explaining the great difficulties encountered in impulsive starts.

APPENDIX J

Body Boundary Value Formulation

In determining the matrices of equation (4.13) it is convenient to consider the vector of dependent variables, $\Delta q^n_{i,1}$, a component at a time. The first component, $\Delta\rho^n_{i,1}$, was simply extrapolated along lines of ξ to obtain the "second order accurate" expression:

$$\Delta\rho^n_{i,1} = 2\Delta\rho^n_{i,2} - \Delta\rho^n_{i,3} \tag{J.1}$$

(Note that ρ and e in this Appendix should properly be written as ρ/J and e/J). This boundary condition on density was felt to be an accurate approximation since the η-line spacing was always very small near the solid bodies studied in this research. The x- and y-momentum differences can be exactly re-expressed as:

$$\Delta(\rho u)^n_{i,1} = u^{n+1}_{i,1}\ \Delta\rho^n_{i,1} + \rho^n_{i,1}\ \Delta u^n_{i,1} \tag{J.2}$$

$$\Delta(\rho v)^n_{i,1} = v^{n+1}_{i,1}\ \Delta\rho^n_{i,1} + \rho^n_{i,1}\ \Delta v^n_{i,1} \tag{J.3}$$

Similarly, the total energy can be exactly expressed as:

$$\Delta e^n_{i,1} = (\frac{e}{\rho})^{n+1}_{i,1}\ \Delta\rho^n_{i,1} + \rho^n_{i,1}\ \Delta(\frac{e}{\rho})^n_{i,1} \tag{J.4}$$

where for the case in which temperature on the body surface is specified, the quantities e/ρ can be evaluated from the equation of state (3.5). For all other thermal boundary conditions, equation (J.4) can be re-expressed as:

$$\Delta e^n_{i,1} = (\frac{e}{\rho})^n_{i,1}\ \Delta\rho^n_{i,1} + \rho^n_{i,1}\ \Delta(\frac{e}{\rho})^{n-1}_{i,1} + O(\Delta t^2) \tag{J.5}$$

Now on substituting relation (J.1) into the other equations, the following form for the H and K matrices and the L vector of equation (4.13) develops:

$$H = \begin{bmatrix} 2 & 0 & 0 & 0 \\ 2u & 0 & 0 & 0 \\ 2v & 0 & 0 & 0 \\ 2E & 0 & 0 & 0 \end{bmatrix}_{i,1}^{n+1} , \quad K = \begin{bmatrix} -1 & 0 & 0 & 0 \\ -u & 0 & 0 & 0 \\ -v & 0 & 0 & 0 \\ -E & 0 & 0 & 0 \end{bmatrix}_{i,1}^{n+1} , \quad L = \begin{bmatrix} 0 \\ \rho\Delta u \\ \rho\Delta v \\ \rho\Delta E \end{bmatrix}_{i,1}^{n} \qquad (J.6)$$

where

$$E = \frac{e}{\rho} = \frac{1}{\gamma}T + \frac{1}{2}(u^2 + v^2) \qquad (J.7)$$

and E and ΔE aıe lagged in time unless the surface temperature has been specified.

Although, in a sense, this treatment of the implicit boundary conditions is rather cavalier, it is this author's experience that the stability and, in particular, the accuracy of the algorithm is not very sensitive to these boundary conditions (that's not to say that the stability and accuracy can not be affected by inappropriate choice of the boundary condition). In fact, Steger [b,g] has had good success with merely lagging the entire difference vector, Δq , in time.

Bibliography

a. Baldwin, B.S. and Lomax, H., "Thin Layer Approximation and Algebraic Model for Separated Turbulent Flows," AIAA Paper 78-257, AIAA 16th Aerospace Sciences Meeting (1978).

b. Steger, J.L., "Implicit Finite Difference Simulation of Flow About Arbitrary Geometrics with Application to Airfoils," AIAA Paper 77-665, AIAA 10th Fluid and Plasmadynamics Conference (1977).

c. Beam, R.M. and Warming, R.F., "An Implicit Factored Scheme for the Compressible Navier-Stokes Equations," AIAA Paper 77-645, AIAA 3rd Computational Fluid Dynamics Conference (1977).

d. Thompson, J.F., Thames, F.C., and Mastin, C.M., "Automatic Numerical Generation of Body-Fitted Curvilinear Coordinate System for Field Containing Any Number of Arbitrary Two-Dimensional Bodies," J. of Comp. Physics, Vol. 15 (1974).

e. Thompson, J.F., "Numerical Solution of Flow Problems using Body-Fitted Coordinate Systems," Lecture Series in Computational Fluid Dynamics, von Karman Inst. for Fluid Dynamics, Belgium (1978).

f. Beam, R.M. and Warming, R.F., "An Implicit Factored Scheme for the Compressible Navier-Stokes Equations II: The Numerical ODE Connection," AIAA Paper 79-1446, AIAA Computational Fluid Dynamics Conference (1979).

g. Pulliam, T.H. and Steger, J.L., "On Implicit Finite-Difference Simulation of Three Dimensional Flow," AIAA Paper 78-10, AIAA 16th Aerospace Sciences Meeting (1978).

h. Turner, L., "Numerical Solution of the Navier-Stokes Equations for Laminar, Transonic Flows," Ph.D. Dissertation, Mississippi State University (1979).

i. Sokolnikoff, I.S., Tensor Analysis, John Wiley & Sons, Inc. (1967).

j. Thompson, D.S., "Numerical Solution of the Navier-Stokes Equations for High Reynolds Number Incompressible Turbulent Flow," M.S. Thesis, Mississippi State University (1980).

k. Rose, W.C., and Seginer, A., "Calculation of Transonic Flow over Supercritical Airfoil Sections," AIAA Paper 77-681, AIAA 10th Fluid and Plasmadynamics Conference (1977).

l. Magnus, R. and Yoshihara, H., "Unsteady Transonic Flows over an Airfoil," AIAA Journal, Vol. 13 (1975).

m. Johnson, D.A. and Bachalo, W.D., "Transonic Flow Past a Symmetrical Airfoil - Inviscid and Turbulent Flow Properties," AIAA Journal, Vol. 18 (1979).

n. Steger, J.L. and Bailey, H.E., "Calculation of Transonic Aileron Buzz," AIAA Paper 79-0134, AIAA 17th Aerospace Sciences Meeting (1979).

o. Roache, P.J., Computational Fluid Dynamics, Hermosa Publishers (1976).

p. Cebeci , T., "Calculation of Compressible Turbulent Boundary-Layers with Heat and Mass Transfer," AIAA Journal, Vol. 9 (1971).

q. Landau, L.D. and Lifshitz, E.M., Fluid Mechanics, Addison-Wesley Pub. Co. (1959).

r. Wirz, H.J. and Smolderen, J.J., Numerical Methods in Fluid Dynamics, Hemisphere Pub. Co. (1978).

s. MacCormack, R.W., "A Rapid Solver for Hyperbolic Systems of Equations," Lecture Notes in Physics, Vol. 59, Springer-Verlag (1976).

t. Briley, W.R. and McDonald, H., "An Implicit Numerical Method for the Multi-Dimensional Compressible Navier-Stokes Equations," Report M911363-6, United Aircraft Research Laboratories (1973).

u. Seegmiller, H.L., Marvin, J.G., and Levy Jr., L.L., "Steady and Unsteady Transonic Flow," AIAA Journal, Vol. 16 (1978).

v. Marvin, J.G., Levy Jr., L.L., and Seegmiller, H.L., "Turbulence Modeling for Unsteady Transonic Flows," AIAA Journal, Vol. 18 (1980).