

DTIC FILE COPY

1



AD-A203 291

NUMERICAL SIMULATION OF FLOW OVER
 ICED AIRFOILS
 THESIS
 Larry A. Coleman
 Captain, USAF
 AFIT/GAE/AA/88D-4

DTIC
 ELECTE
 JAN 18 1989
 S D
 UD

DISTRIBUTION STATEMENT A
 Approved for public release
 Distribution Unlimited

DEPARTMENT OF THE AIR FORCE
 AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

89 1 17 139

①

AFIT/GAE/AA/88D-4

DTIC
SELECTED
S JAN 18 1989 D
DCS

NUMERICAL SIMULATION OF FLOW OVER
ICED AIRFOILS
THESIS

Larry A. Coleman
Captain, USAF

AFIT/GAE/AA/88D-4

Approved for public release; distribution unlimited

AFIT/GAE/AA/88D-4

NUMERICAL SIMULATION OF FLOW OVER ICED AIRFOILS

THESIS

Presented to the Faculty of the School of Engineering
of the Air Force Institute of Technology
Air University

In Partial Fulfillment of the
Requirements for the Degree of
Master of Science in Aeronautical Engineering

Larry A. Coleman, B.S.
Captain, USAF

December 1988



Accession For	
NTIS - ORBIT	<input checked="" type="checkbox"/>
DTIC - TAB	<input type="checkbox"/>
Unannounced	<input type="checkbox"/>
Justified	
By	
Distribution	
Availability Codes	
Dist	Avail and/or Special
A-1	

Approved for public release; distribution unlimited

Preface

The objective of the current study was to numerically solve the full Navier-Stokes equations for flow over an iced NACA 0012 airfoil. The Beam-Warming algorithm was used to obtain steady-state solutions for angles of attack ranging from 2-8 degrees. Comparison of these results was made with other numerical results obtained from algorithms based on approximate forms of the Navier-Stokes equations. The present results were compared to experimental data as well.

These comparisons revealed that the Beam-Warming scheme accurately predicts the iced airfoil lift and drag for angles of attack below stall. The post-stall lift is over predicted, however, and a study of the unsteady nature of the flowfield that results after stall will be needed to resolve this problem. Further research is also needed to refine the computed local results, such as the reattachment location of the separation bubbles aft of the ice shape. Continuation of this effort in future thesis work could lead to resolution of these issues.

Throughout the course of my work on this thesis, I have received considerable assistance from others. I would especially like to thank my advisor, Dr. A.A.M. Halim, for the guidance and direction he provided throughout this research effort. From start to finish, Dr. Halim was always willing to answer my questions, and help me to understand

and resolve the many problems that I faced along the way. I am also deeply indebted to my sponsor, Dr. Joe Shang of the Flight Dynamics Laboratory, for providing the funding for the computer resources used in this project. He provided much more than financial support, though. Dr. Shang also took time from his busy schedule to provide very valuable technical advice, and for his help in this respect I am quite grateful. I also wish to thank Dr. Miguel Visbal of the Flight Dynamics Laboratory for his assistance with the Beam-Warming code.

I would like to thank the other members of my thesis committee, Dr. Milton Franke and Capt Phil Beran, for their advice and comments pertaining to the written report. Two fellow AFIT students deserve acknowledgement as well. I want to thank Capt Faran Hafeez for working with me to modify Dr. Halim's code. I also wish to thank Capt Paul D. Boyles for providing a copy of his data reduction program for use in this work.

Finally, I would like to pay a special tribute to my family for their constant support and understanding during the many hours I spent away from them working on this project. To my wife, [REDACTED] and my children, [REDACTED] [REDACTED] a very special thank you for giving me the moral support and encouragement which helped me complete this thesis successfully.

Larry A. Coleman

Table of Contents

	Page
Preface	ii
List of Figures	vi
List of Symbols	viii
Abstract	xiv
I. Introduction	1
II. Analysis	8
Governing Equations	8
Non-dimensional Form of the Equations	13
Equations in Generalized Coordinates	16
Grid Generation	21
Boundary Conditions	25
III. Numerical Procedures	27
Beam-Warming Scheme	27
Linearization of the Flux Vectors	30
Approximate Factorization	33
Added Dissipation Terms	36
Halim's Formulation of the Thin-Layer Equations	37
Linearization of the Vorticity Equation	39
Boundary Conditions in Terms of Stream Function and Vorticity	40
Modification of the Original Code	43
IV. Results and Discussion	45
Results for an Iced NACA 0012 Using the Beam-Warming Code	46
Convergence Criteria	47
Global Results	51
Local Results	60
Results for a Clean NACA 0012 Using Halim's Formulation	81
V. Conclusions and Recommendations	86
Appendix A: Nondimensionalization of the Governing Equations	90

Appendix B: Transformation to Generalized Coordinates	97
Appendix C: Jacobian Matrices for the Beam-Warming Algorithm	109
Bibliography	123
Vita	126

List of Figures

Figure	Page
1. Typical Rime and Glaze Ice Formations on an Airfoil Leading Edge	3
2. Coordinate Transformation from the Physical Domain to the Computational Domain	18
3. Typical C-Grid for an Iced NACA 0012 Airfoil	24
4. Behavior of the Root Mean Square of the Residual with Number of Iterations for 4.0 Degrees Angle of Attack	48
5. Behavior of the Lift Coefficient with Number of Iterations for 4.0 Degrees Angle of Attack	49
6. Behavior of the Drag Coefficient with Number of Iterations for 4.0 Degrees Angle of Attack	50
7. Lift Coefficient Curve for an Iced NACA 0012 Airfoil	52
8. Drag Coefficient Curve for an Iced NACA 0012 Airfoil	55
9. Comparison of Computed and Experimental Pressure Coefficient for 2.0 Degrees Angle of Attack	56
10. Comparison of Computed and Experimental Pressure Coefficient for 4.0 Degrees Angle of Attack	57
11. Comparison of Computed and Experimental Pressure Coefficient for 6.0 Degrees Angle of Attack	58
12. u-Velocity Contours for 2.0 Degrees Angle of Attack-Unlabeled Contours	61
13. u-Velocity Contours for 2.0 Degrees Angle of Attack-Labeled Contours	62
14. u-Velocity Contours for 4.0 Degrees Angle of Attack-Unlabeled Contours	63
15. u-Velocity Contours for 4.0 Degrees Angle of Attack-Labeled Contours	64

16.	u-Velocity Contours for 6.0 Degrees Angle of Attack-Unlabeled Contours	65
17.	u-Velocity Contours for 6.0 Degrees Angle of Attack-Labeled Contours	66
18.	u-Velocity Contours for 8.0 Degrees Angle of Attack-Unlabeled Contours	67
19.	u-Velocity Contours for 8.0 Degrees Angle of Attack-Labeled Contours	68
20.	u-Velocity Contours over Entire Airfoil for 6.0 Degrees Angle of Attack	70
21.	u-Velocity Contours over Entire Airfoil for 8.0 Degrees Angle of Attack	71
22.	Velocity Vector Plot for 2.0 Degrees Angle of Attack	72
23.	Velocity Vector Plot for 4.0 Degrees Angle of Attack	73
24.	Velocity Vector Plot for 6.0 Degrees Angle of Attack	74
25.	Velocity Vector Plot for 8.0 Degrees Angle of Attack	75
26.	Velocity Vector Plot over Entire Airfoil for 6.0 Degrees Angle of Attack	76
27.	Velocity Vector Plot over Entire Airfoil for 8.0 Degrees Angle of Attack	77
28.	Comparison of Computed Reattachment Locations with Experiment	80
29.	Velocity Profile Comparison at $x/c=0.0$ on the Upper Surface for 4.0 Degrees Angle of Attack	82
30.	Streamline Contours over a Clean NACA 0012 Airfoil at 0.0 Degrees Angle of Attack	84
31.	Skin Friction Distribution over a Clean NACA 0012 Airfoil at 0.0 Degrees Angle of Attack	85

List of Symbols

Symbol	Definition
$a_{11}-a_{44}$	components of the Jacobian matrix, [A]
b_1-b_4	parameters used in defining the viscous flux vector, V_1
$b_{11}-b_{44}$	components of the Jacobian matrix, [B]
c	airfoil chord
c_p	specific heat at constant pressure
c_v	specific heat at constant volume
c_1-c_4	parameters used in defining the viscous flux vectors, V_2 and W_1
d_1-d_4	parameters used in defining the viscous flux vector, W_2
e	internal energy per unit mass
f	general function used in defining the finite-difference operators
k	thermal conductivity
k_T	turbulent thermal conductivity
p	static pressure
q_x	heat flux component in the x-direction
q_y	heat flux component in the y-direction
$r_{11}-r_{44}$	components of the Jacobian matrix, [R]
$s_{11}-s_{44}$	components of the Jacobian matrix, [S]
t	time

u	velocity component along the x-direction in the physical domain
v	velocity component along the y-direction in the physical domain
x	coordinate direction along the airfoil in the physical domain
y	coordinate direction normal to the airfoil in the physical domain
[A]	Jacobian matrix defined as $\partial E_1 / \partial \hat{U}$
[B]	Jacobian matrix defined as $\partial E_2 / \partial \hat{U}$
[C]	cofactor matrix
E	flux vector differentiated with respect to x in the Navier-Stokes equations
\hat{E}	flux vector differentiated with respect to ξ in the Navier-Stokes equations
E_1	vector of inviscid flux and pressure terms from \hat{E}
E_2	vector of inviscid flux and pressure terms from \hat{F}
E_1-E_4	components of the vectors E_1 or E_2
E_t	total energy per unit mass
F	flux vector differentiated with respect to y in the Navier-Stokes equations
\hat{F}	flux vector differentiated with respect to η in the Navier-Stokes equations
G	flux vector differentiated with respect to z in the Navier-Stokes equations
[I]	identity matrix
J	Jacobian of the transformation from the physical domain to the computational domain

LHS	terms on the left hand side of an equality
M	Mach number
O()	denotes order of ()
Pr	Prandtl number
Pr _T	turbulent Prandtl number
R	gas constant for air
Re _c	Reynolds number based on chord
RHS	terms on the right hand side of an equality
[R]	Jacobian matrix defined as $\partial v_1 / \partial \hat{U}_\xi$
S ₁	constant in Sutherland's equation
[S]	Jacobian matrix defined as $\partial w_2 / \partial \hat{U}_\eta$
T	static temperature
T ₀	reference temperature in Sutherland's equation
U	vector of conserved variables
\hat{U}	vector of conserved variables divided by the Jacobian, J
U	velocity component parallel to the airfoil in the computational domain
U ₁ -U ₄	components of the vectors \hat{U} or \hat{U}_ξ
V ₁	vector of viscous and heat flux terms from \hat{E} which are functions of \hat{U} and \hat{U}_ξ
V ₂	vector of viscous and heat flux terms from \hat{E} which are functions of \hat{U} and \hat{U}_η
V	velocity component normal to the airfoil in the computational domain
V _∞	magnitude of the freestream velocity

$V_1 - V_4$	components of the vector V_1
W_1	vector of viscous and heat flux terms from \hat{F} which are functions of \hat{U} and \hat{U}_ξ
W_2	vector of viscous and heat flux terms from \hat{F} which are functions of \hat{U} and \hat{U}_η
$W_1 - W_4$	components of the vector W_2
$\left[\quad \right]$	denotes a matrix
$\left \quad \right $	denotes a determinant
Greek Symbols	
α	angle of attack; metric in Halim's formulation
β	metric in Halim's formulation
γ	ratio of specific heats (c_p/c_v) ; metric in Halim's formulation
δ_ξ, δ_η	finite difference operators
ϵ	turbulent eddy viscosity
η	coordinate direction normal to the airfoil surface in the computational domain
θ_1, θ_2	parameters in the general time differencing expression
λ_T	parameter relating μ and ϵ in Stokes' hypothesis
μ	molecular viscosity
μ_0	reference viscosity in Sutherland's equation
μ_ξ, μ_η	finite difference operators
ξ	coordinate direction parallel to the airfoil surface in the computational domain

ρ	density
τ_{xx}	shear stress component in the x-direction, normal to an x-face
τ_{xy}	shear stress component in the y-direction, normal to an x-face
τ_{yy}	shear stress component in the y-direction, normal to a y-face
ϕ	parameter used in defining the flux vectors
ψ	stream function
ω	vorticity
ω_E	explicit damping parameter in the Beam-Warming scheme
ω_I	implicit damping parameter in the Beam-Warming scheme
$\Delta()$	denotes an incremental value of ()

Superscripts

$n-1$	previous iteration or time level
n	current iteration or time level
$n+1$	next iteration or time level
$*$	denotes a dimensionless quantity

Subscripts

i	grid point location in x- or ξ -direction
j	grid point location in y- or η -direction
max	maximum value
o	reference condition
t	total, as in total energy; differentiation with respect to time

- x differentiation with respect to x (except shear stress and heat flux terms, where it denotes direction)
- y differentiation with respect to y (except shear stress and heat flux terms, where it denotes direction)
- T turbulent quantity
- ξ differentiation with respect to ξ
- η differentiation with respect to η
- ∞ free stream conditions

Abstract

The accumulation of ice on an aircraft reduces the lift and increases the drag of the aircraft, thus posing a serious hazard to flight safety. The primary goal of research into the aircraft icing problem is to develop methods for quantifying the detrimental effects on aircraft performance. This task can be handled experimentally or with a numerical approach, or with a combination of both techniques. Current research interest focuses on developing computer codes which can be used to predict iced airfoil performance. The experimental data gathered over the years provides a valuable database for verifying the codes in development.

The objective of the present study is to evaluate the performance of an iced NACA 0012 airfoil numerically. The full Navier-Stokes equations are solved using the Beam-Warming implicit factored scheme. Steady-state solutions are obtained for flow at a free stream Mach number of 0.12 and a Reynolds number based on chord of 1.41×10^6 , for angles of attack of 2, 4, 6, and 8 degrees.

The lift and drag curves obtained from the numerical solutions were compared to experimental data and other numerical results. It was found that the Beam-Warming algorithm provides a good estimate of the lift and drag at angles of attack below stall. Computed lift coefficients

were within 11.5% of the experimental data, while the drag coefficients differed by as much as 24%. These results were in excellent agreement with the other numerical solutions. At an angle of attack above stall, however, the code did not predict the expected decrease in lift, and the calculated drag coefficient was much lower than the experimental data.

The comparison of two local flowfield characteristics with the experimental data was less encouraging. A computed velocity profile was compared to the experimental profile for a station in the separated region on the upper surface. This comparison showed that the computed separation bubble is approximately one half the thickness of the bubble measured experimentally. Flow reattachment location is another measure of the predictive accuracy of the numerical scheme. The reattachment locations from the numerical solutions were 30-40% less than the experimental values.

The conclusion to be drawn from these comparisons is that accurate results can be obtained for the global performance parameters using the code in its present state, but more work is certainly needed to refine the local results. To achieve this goal, further study is needed in three areas: 1) effect of the grid on solution accuracy, 2) effect of the transition location on the results, and 3) effect of the turbulence model on the solution.

NUMERICAL SIMULATION OF FLOW OVER ICED AIRFOILS

1. Introduction

Ice accumulation on aircraft surfaces results in substantial performance penalties. Formation of ice on the leading edge of an airfoil can lead to separated flow regions on the upper and lower surfaces. Flow separation results in severe loss of lift and a dramatic increase in the drag of the airfoil. For example, in experimental work at Ohio State University, Bragg has tested a NACA 0012 airfoil with a simulated ice shape which resulted in a reduction of maximum lift of over 50% and a drag increase of 300% (1:1). The implications of such severe performance penalties on flight safety are obvious.

An aircraft can encounter icing when flying through clouds. Two types of ice can form on the aircraft surfaces depending upon the environmental conditions (2:2-3). When the air temperature is well below freezing and the liquid water content of the air is low, the water droplets freeze almost immediately upon impact forming spherical grains of ice. This type of ice formation is termed rime ice. If the air temperature is near freezing and the liquid water content of the air is high, the droplets impinging on the surface do not freeze immediately. Instead, droplets run

back on the airfoil surface before freezing which leads to the formation of horn-shaped structures on the upper and lower airfoil surfaces. This type of ice formation is called glaze ice. The glaze ice horns cause the flow to separate. Therefore, glaze ice is the most detrimental to aircraft performance. Typical rime and glaze ice formations are shown in Figure 1.

The problem of aircraft icing has been of interest for many years. In 1932, Jacobs tested a NACA 0012 airfoil with various leading edge protrusions (3). Although none of these experiments were conducted with actual ice accretions, the detrimental effect of distorting the leading edge shape was established in this work. Jacobs' results were reported in terms of changes in the airfoil's lift, drag, and moment.

In the 1950s Gray continued NACA's study with tests of the NACA 65-212 and NACA 65A004 airfoils. These experiments were conducted in NACA's Icing Research Tunnel and data were obtained for these airfoils with actual ice accretions. The adverse effect of the ice was again reported in terms of the global airfoil performance parameters--lift, drag, and moment coefficients (4,5). In 1964, Gray used the experimental data available at the time to correlate changes in aerodynamic performance with ice accumulation. He also attempted to generalize the correlation to other airfoils using leading edge radius as a parameter (6:1). While this correlation has proven useful in a qualitative sense, Cebeci

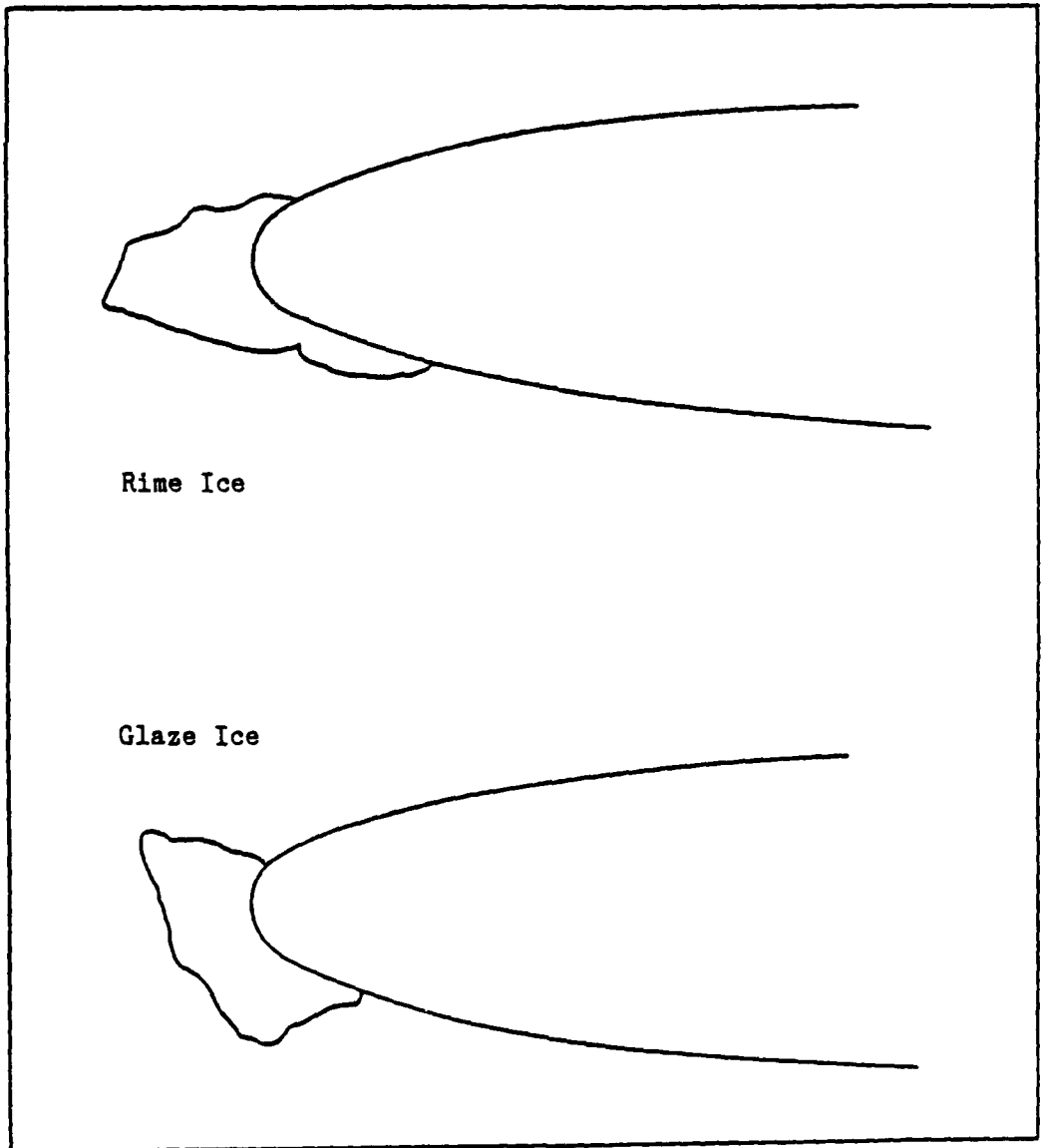


Figure 1. Typical Rime and Glaze Ice Formations on an Airfoil Leading Edge

points out that the drag rise predicted with Gray's correlation is generally too high (7:11).

The need to more accurately predict the effect of ice accumulation on airfoils has prompted further research in this area in the 1980s. The ultimate goal of the current NASA program, for instance, is to develop computer codes which can be used in the airfoil design process to predict the sensitivity of a proposed design to ice accumulation. Before any code can be used as a reliable design tool, however, it must be thoroughly tested. One method to test a code under development is to apply the code to an iced airfoil for which experimental data are available, and compare airfoil performance based on the computed solution to that obtained from the experimental data. Obviously, accurate, detailed experimental data for an existing iced airfoil are required to fully validate a given code.

In recent years, several investigators have worked to provide the detailed experimental data needed to verify the computer codes in development. Bragg and Coirier have performed a series of experiments using a NACA 0012 airfoil with a simulated glaze ice shape attached to the leading edge. In these experiments, the lift, drag, and moment of the airfoil was determined for the airfoil with and without ice. In addition, surface pressure taps were located in the regions behind the ice horns to allow measurement of the pressure distribution in the separated flow regions. Also, a split film probe was used to map out the velocity profiles

at several chordwise stations in the separation bubble behind the ice horns. These velocity profiles clearly indicate the regions of reversed flow in the bubble and the bubble reattachment point (1,8). In an extension of this work, Bragg and Spring have investigated the effect of adding surface roughness to the glaze ice shape (9,10). In addition to these quantitative studies, Khodadoust's flow visualization study of this same airfoil provides some qualitative information on the behavior of the flow in the leading edge separation bubble (11).

In addition to the experimental work on iced airfoils in recent years, there has been an effort to develop computational methods for determining the flow field over iced airfoils. Two researchers, Potapczuk at NASA's Lewis Research Center and Cebeci at the California State University, have published very promising results for the flow over the NACA 0012 airfoil tested experimentally by Bragg and Coirier. Both investigators have produced computational results which compare very favorably to the lift and drag data of Bragg and Coirier, even though different numerical schemes are used by each.

Cebeci has developed a code for iced airfoils based on an interactive boundary layer (IBL) method (12). The code allows for interaction of the viscous and inviscid flow regimes. With this feature, the boundary layer equations are well behaved even beyond the point of separation on the airfoil surface. Therefore, the code can be marched past

the point of separation and a solution can be obtained even in regions of reverse flow. Turbulent eddy viscosity is computed using Cebeci and Smith's algebraic turbulence model. Details of the IBL method are in Reference 12. The Cebeci-Smith turbulence model is described in Reference 13.

Potapczuk developed a computational method for the flow over iced airfoils using the thin layer Navier-Stokes equations (14,15). The code being used is ARC2D, a Navier-Stokes code developed by Steger and Pulliam at NASA Ames. The algebraic turbulence model of Baldwin and Lomax is used in this code to calculate the turbulent eddy viscosity. Details of the ARC2D code can be found in References 16 and 17. The Baldwin-Lomax turbulence model is described fully in Reference 18.

In the present work, two additional numerical methods are investigated for calculating the flow field over the NACA 0012 airfoil with glaze ice. The goal of the study is to compare the present numerical results to other numerical results and experimental data, and evaluate the advantages or disadvantages of the algorithms studied in this thesis. To match the experimental conditions of Bragg and Coirier, the numerical solutions are obtained for a Mach number of 0.12 and a Reynolds number of 1.41×10^6 . Numerical solutions are obtained for angles-of-attack ranging from 2-8 degrees.

One of the numerical methods investigated is a global space marching technique developed by Halim (19). This code

is based on an approximate form of the Navier-Stokes equations obtained by dropping the streamwise diffusion terms. The equations are written in terms of the stream function and vorticity rather than the primitive variables. The advantage of this formulation is that the number of unknowns is reduced by one.

A second numerical method applied to the iced airfoil is the implicit approximate factored scheme due to Beam and Warming (20). This code is a time marching implementation of the full Navier-Stokes equations. No simplifying assumptions have been made. This code is written in terms of the primitive variables.

There are two common features for these codes. First of all, each code is written in terms of generalized coordinates to allow computation over arbitrary geometries. This is accomplished by transforming an arbitrary physical space to a computational space with uniform spacing. Secondly, each code is written in delta formulation. This means that the solution obtained is in terms of the difference between a variable's value at two time levels. In the limit, this difference approaches zero as the solution approaches steady-state.

II. Analysis

In this chapter the governing equations for flow over an airfoil are described. Anderson and Tannehill (21) is the primary reference for the equations presented in this chapter. The governing equations are put into non-dimensional form and transformed from Cartesian coordinates to generalized coordinates. The method of generating the grid for solving the equations in the transformed domain is briefly discussed. Finally, the appropriate boundary conditions for flow over an airfoil are presented.

Governing Equations

The flow field over an iced airfoil is described by the Navier-Stokes equations. The vector form of the compressible Navier-Stokes equations in Cartesian coordinates is given by

$$\frac{\partial U}{\partial t} + \frac{\partial E}{\partial x} + \frac{\partial F}{\partial y} + \frac{\partial G}{\partial z} = 0 \quad (1)$$

These equations can be simplified for two-dimensional flow to

$$\frac{\partial U}{\partial t} + \frac{\partial E}{\partial x} + \frac{\partial F}{\partial y} = 0 \quad (2)$$

where the vector of conserved variables, U , is given by

$$U = \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ E_t \end{bmatrix} \quad (3)$$

and the flux vectors, E and F are

$$E = \begin{bmatrix} \rho u \\ \rho u^2 + p - \tau_{xx} \\ \rho uv - \tau_{xy} \\ (E_t + p)u - u\tau_{xx} - v\tau_{xy} + q_x \end{bmatrix} \quad (4)$$

$$F = \begin{bmatrix} \rho v \\ \rho uv - \tau_{xy} \\ \rho v^2 + p - \tau_{yy} \\ (E_t + p)v - u\tau_{xy} - v\tau_{yy} + q_y \end{bmatrix} \quad (5)$$

For turbulent flow the shear stress components are expressed as

$$\tau_{xx} = 2(\mu + \epsilon)u_x + \lambda_T(u_x + v_y) \quad (6)$$

$$\tau_{xy} = (\mu + \epsilon)(u_y + v_x) \quad (7)$$

$$\tau_{yy} = 2 (\mu + \epsilon) v_y + \lambda_T (u_x + v_y) \quad (8)$$

where

μ = molecular dynamic viscosity
 ϵ = turbulent eddy viscosity

Also, assuming Stokes' hypothesis to be valid for this flow,

λ_T is defined as

$$\lambda_T = - 2/3 (\mu + \epsilon) \quad (9)$$

The heat flux components for turbulent flow are

$$q_x = - \left(k + k_T \right) \frac{\partial T}{\partial x} \quad (10)$$

$$q_y = - \left(k + k_T \right) \frac{\partial T}{\partial y} \quad (11)$$

where

k = thermal conductivity of air
 k_T = turbulent thermal conductivity

Introducing the Prandtl number, Pr , and turbulent Prandtl number, Pr_T , defined as

$$Pr = \frac{\mu c_p}{k} \quad Pr_T = \frac{\epsilon c_p}{k_T} \quad (12)$$

the heat flux terms can be rewritten as

$$q_x = - c_p \left(\frac{\mu}{Pr} + \frac{\epsilon}{Pr_T} \right) \frac{\partial T}{\partial x} \quad (13)$$

$$q_y = -c_p \left(\frac{\mu}{Pr} + \frac{\epsilon}{Pr_T} \right) \frac{\partial T}{\partial y} \quad (14)$$

The energy term, E_t , in the E and F vectors above denotes the total energy per unit volume, given by

$$E_t = \rho \left(e + \frac{u^2 + v^2}{2} \right) \quad (15)$$

where

u, v = velocity components
 e = internal energy per unit mass

The equation of state for a perfect gas is used to close the system of equations. This equation is

$$P = \rho RT \quad (16)$$

where R is the gas constant for air. Using the relation for a calorically perfect gas

$$e = c_v T \quad (17)$$

and the relationships

$$R = c_p - c_v \quad (18)$$

$$\gamma = c_p / c_v \quad (19)$$

where

c_p = specific heat at constant pressure
 c_v = specific heat at constant volume

the internal energy can be expressed as

$$e = \frac{RT}{\gamma - 1} \quad (20)$$

Substituting for RT in the equation of state yields

$$P = \rho e (\gamma - 1) \quad (21)$$

Using Eq (21) the pressure can be expressed in terms of the total energy as

$$P = (\gamma - 1) \left[E_t - 1/2 \rho (u^2 + v^2) \right] \quad (22)$$

In addition to the equations developed above, expressions are needed to evaluate the molecular and eddy viscosities. The molecular viscosity is determined using Sutherland's formula (22:328), given by

$$\frac{\mu}{\mu_0} = \left(\frac{T}{T_0} \right)^{3/2} \left(\frac{T_0 + S_1}{T + S_1} \right) \quad (23)$$

where

T_0 = absolute reference temperature
 μ_0 = molecular viscosity at reference temperature
 S_1 = Sutherland's constant (110 K)

The turbulent eddy viscosity is evaluated using the algebraic eddy viscosity model of Baldwin and Lomax (18), as modified by Visbal (23).

Non-dimensional Form of the Equations

It is customary to cast the governing equations into non-dimensional form so they are expressed in terms of parameters such as Reynolds number, Mach number and Prandtl number. One method for non-dimensionalizing the equations, as suggested by Anderson and Tannehill (21), is to define dimensionless variables of the form

$$\begin{aligned}
 x^* &= x/c & y^* &= y/c & t^* &= \frac{t}{c/V_\infty} \\
 u^* &= u/V_\infty & v^* &= v/V_\infty & \rho^* &= \rho/\rho_\infty \\
 \mu^* &= \mu/\mu_\infty & \epsilon^* &= \epsilon/\mu_\infty & T^* &= T/T_\infty & (24) \\
 p^* &= \frac{p}{\rho V_\infty^2} & e^* &= \frac{e}{V_\infty^2}
 \end{aligned}$$

where

$$\begin{aligned}
 c &= \text{airfoil chord length} \\
 V_\infty &= \text{free stream velocity}
 \end{aligned}$$

The asterisk denotes the dimensionless variables and the subscript ∞ indicates free stream conditions. Applying these expressions to the Navier-Stokes equations yields

$$\frac{\partial U^*}{\partial t^*} + \frac{\partial E^*}{\partial x^*} + \frac{\partial F^*}{\partial y^*} = 0 \quad (25)$$

where the dimensionless vectors are expressed as

$$U^* = \begin{bmatrix} \rho^* \\ \rho^* u^* \\ \rho^* v^* \\ E_t^* \end{bmatrix} \quad (26)$$

$$E^* = \begin{bmatrix} \rho^* u^{*2} \\ \rho^* u^{*2} + p^* - \tau_{xx}^* \\ \rho^* u^* v^* - \tau_{xy}^* \\ (E_t^* + p^*) u^* - u^* \tau_{xx}^* - v^* \tau_{xy}^* + q_x^* \end{bmatrix} \quad (27)$$

$$F^* = \begin{bmatrix} \rho^* v^{*2} \\ \rho^* u^* v^* - \tau_{xy}^* \\ \rho^* v^{*2} + p^* - \tau_{yy}^* \\ (E_t^* + p^*) v^* - u^* \tau_{xy}^* - v^* \tau_{yy}^* + q_y^* \end{bmatrix} \quad (28)$$

In these vectors the dimensionless shear stress terms are given by

$$\tau_{xx}^* = \frac{1}{Re_c} \left[2(\mu^* + \epsilon^*) \frac{\partial u^*}{\partial x^*} - \frac{2}{3}(\mu^* + \epsilon^*) \left(\frac{\partial u^*}{\partial x^*} + \frac{\partial v^*}{\partial y^*} \right) \right] \quad (29)$$

$$\tau_{xy}^* = \frac{1}{Re_c} (\mu^* + \epsilon^*) \left(\frac{\partial u^*}{\partial x^*} + \frac{\partial v^*}{\partial y^*} \right) \quad (30)$$

$$\tau_{yy}^* = \frac{1}{Re_c} \left[2(\mu^* + \epsilon^*) \frac{\partial v^*}{\partial y^*} - \frac{2}{3}(\mu^* + \epsilon^*) \left(\frac{\partial u^*}{\partial x^*} + \frac{\partial v^*}{\partial y^*} \right) \right] \quad (31)$$

The corresponding heat flux expressions are

$$q_x^* = \frac{-1}{(\gamma-1)M_\infty^2 Re_c} \left(\frac{\mu^*}{Pr} + \frac{\epsilon^*}{Pr_T} \right) \frac{\partial T^*}{\partial x^*} \quad (32)$$

$$q_y^* = \frac{-1}{(\gamma-1)M_\infty^2 Re_c} \left(\frac{\mu^*}{Pr} + \frac{\epsilon^*}{Pr_T} \right) \frac{\partial T^*}{\partial y^*} \quad (33)$$

The dimensionless form of the total energy is expressed as

$$E_t^* = \rho^* \left(e^* + \frac{u^{*2} + v^{*2}}{2} \right) \quad (34)$$

and the corresponding dimensionless equations of state are

$$p^* = (\gamma - 1) \rho^* e^* \quad (35)$$

$$T^* = \frac{\gamma M_\infty^2 p^*}{\rho^*} \quad (36)$$

Note that the dimensionless vectors have exactly the same form as the dimensional ones. For this reason, the * notation is generally dropped once the dimensionless form of the equations has been derived. This convention will be followed in this thesis. In all subsequent references to the governing equations the dimensionless form is implied. Details of the non-dimensionalization for representative terms are shown in Appendix A.

Equations in Generalized Coordinates

The equations as developed so far are suitable for application to problems in the Cartesian coordinate system. For problems with flow over irregular geometries, however, the equations can be more conveniently solved if expressed in terms of generalized coordinates. The general transformation given by

$$\xi = \xi(x, y) \quad (37)$$

$$\eta = \eta(x, y) \quad (38)$$

can be applied to transform the governing equations from the physical domain (x, y) to a computational domain (ξ, η) . The

transformation is shown in Figure 2. As can be seen, the advantage of the transformation is that a uniformly spaced computational grid results.

As outlined in Anderson and Tannehill (21), the transformation from physical coordinates to computational coordinates is performed by applying the chain rule of partial differentiation as follows:

$$\frac{\partial(\cdot)}{\partial x} = \xi_x \frac{\partial(\cdot)}{\partial \xi} + \eta_x \frac{\partial(\cdot)}{\partial \eta} \quad (39)$$

$$\frac{\partial(\cdot)}{\partial y} = \xi_y \frac{\partial(\cdot)}{\partial \xi} + \eta_y \frac{\partial(\cdot)}{\partial \eta} \quad (40)$$

Applying the chain rule to Eq (25) yields

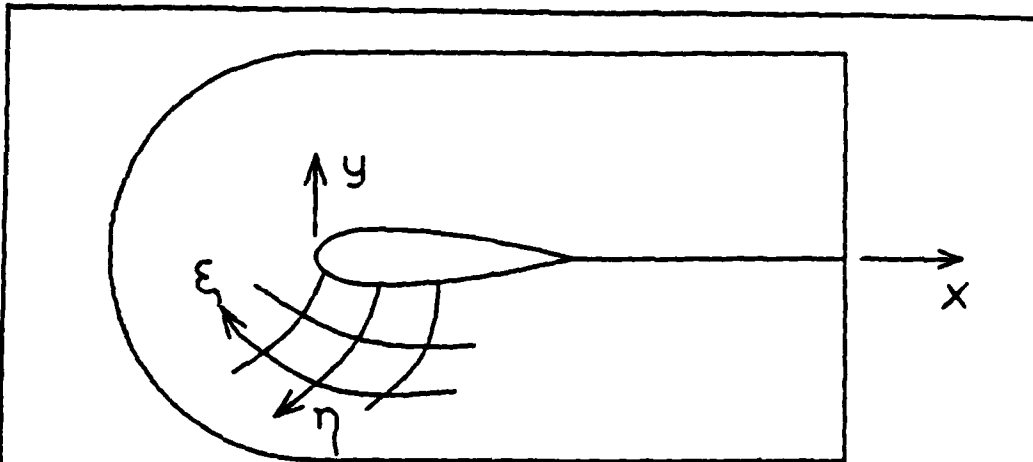
$$\begin{aligned} \frac{\partial}{\partial t} \left(\frac{U}{J} \right) + \frac{\partial}{\partial \xi} \left[\frac{1}{J} \left(\xi_x E + \xi_y F \right) \right] \\ + \frac{\partial}{\partial \eta} \left[\frac{1}{J} \left(\eta_x E + \eta_y F \right) \right] \end{aligned} \quad (41)$$

where J is the Jacobian of the transformation defined as

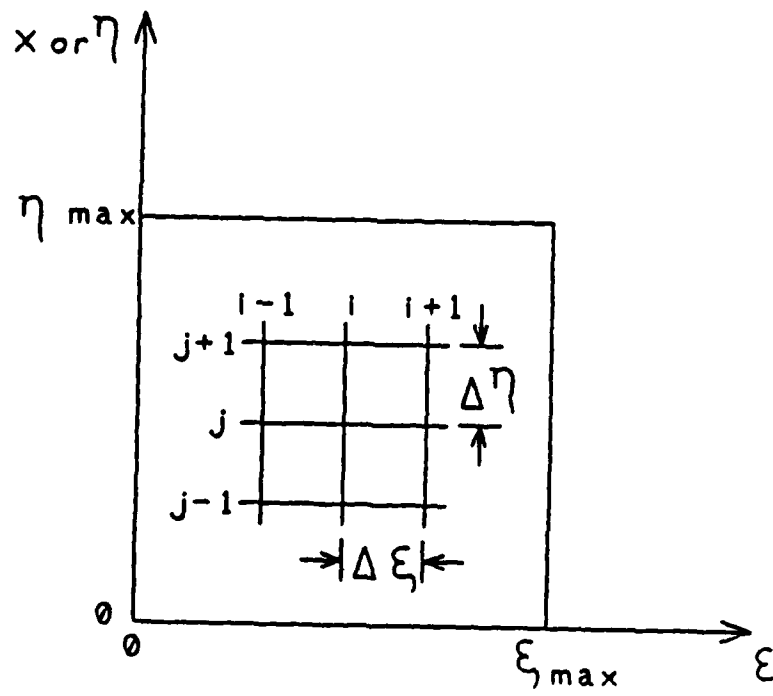
$$J = \frac{\partial(\xi, \eta)}{\partial(x, y)} = \begin{vmatrix} \xi_x & \xi_y \\ \eta_x & \eta_y \end{vmatrix} \quad (42)$$

or

$$J = \xi_x \eta_y - \eta_x \xi_y \quad (43)$$



(a) Physical Plane



(b) Transformed Plane

Figure 2. Coordinate Transformation from the Physical Domain to the Computational Domain

Further details of the coordinate transformation are in Appendix B.

Now Eq (41) can be rewritten as

$$\frac{\partial \hat{U}}{\partial t} + \frac{\partial \hat{E}}{\partial \xi} + \frac{\partial \hat{F}}{\partial \eta} = 0 \quad (44)$$

where

$$\hat{U} = U/J \quad (45)$$

$$\hat{E} = 1/J \left(\epsilon_x E + \epsilon_y F \right) \quad (46)$$

$$\hat{F} = 1/J \left(\eta_x E + \eta_y F \right) \quad (47)$$

One further rearrangement of the governing equations is needed to aid in the implementation of the implicit algorithm. It is desirable to separate the inviscid and viscous terms in the \hat{E} and \hat{F} vectors, and it is also necessary to separate the cross-derivative terms. This is accomplished by rearranging terms into six new vectors.

The inviscid and pressure terms in \hat{E} in Eq (44) form a new vector E_1 . Similarly, the inviscid and pressure terms in \hat{F} form a new vector E_2 . These vectors can be expressed as

$$E_1 = \frac{1}{J} \begin{bmatrix} \rho U \\ \rho u U + \xi_x p \\ \rho v U + \xi_y p \\ (E_t + p) U \end{bmatrix} \quad (48)$$

$$E_2 = \frac{1}{J} \begin{bmatrix} \rho V \\ \rho u V + \eta_x p \\ \rho v V + \eta_y p \\ (E_t + p) V \end{bmatrix} \quad (49)$$

where the contravariant velocities, U and V , are defined as

$$U = \xi_x u + \xi_y v \quad (50)$$

$$V = \eta_x u + \eta_y v \quad (51)$$

The viscous and heat flux terms which remain in Eq (44) are separated into four vectors to isolate the cross-derivatives. These terms in \hat{E} can be expressed in general as

$$v = v_1(\hat{U}, \hat{U}_\xi) + v_2(\hat{U}, \hat{U}_\eta) \quad (52)$$

and the terms in \hat{F} can be written in general as

$$w = w_1(\hat{U}, \hat{U}_\xi) + w_2(\hat{U}, \hat{U}_\eta) \quad (53)$$

With these definitions, Eq (44) can be rewritten as

$$\begin{aligned}
\frac{\partial}{\partial t} \left(\hat{U} \right) + \frac{\partial}{\partial \xi} \left(E_1 \right) + \frac{\partial}{\partial \eta} \left(E_2 \right) = \\
\frac{\partial}{\partial \xi} \left[v_1 \left(\hat{U}, \hat{U}_\xi \right) \right] + \frac{\partial}{\partial \xi} \left[v_2 \left(\hat{U}, \hat{U}_\eta \right) \right] \\
+ \frac{\partial}{\partial \eta} \left[w_1 \left(\hat{U}, \hat{U}_\xi \right) \right] + \frac{\partial}{\partial \eta} \left[w_2 \left(\hat{U}, \hat{U}_\eta \right) \right] \quad (54)
\end{aligned}$$

The specific components of V_1 , V_2 , W_1 , and W_2 involve the expressions for the shear stress and heat flux terms in generalized coordinates. These expressions are included in Appendix B.

The Navier-Stokes equations are now in a form suitable for application to the Beam-Warming implicit algorithm. Details of this numerical scheme are discussed in the next chapter.

Grid Generation

In the numerical study of any fluid dynamics problem, the continuous solution is approximated by solving the governing equations at discrete points in the flow field. The distribution of these points in space forms a grid or mesh system. Accuracy of the solution for a given problem is highly dependent on the number and placement of the grid points.

For simple geometries, the grid can be defined by simply specifying the spacing and number of points in the directions parallel and normal to the body. The governing equations are then solved directly in the physical domain. For more complicated shapes, such as two-dimensional airfoils, fitting the grid to the body results in rather arbitrary spacing in the physical domain. Specifying the boundary conditions on the surface in physical coordinates can also be difficult for complex shapes. Therefore, the physical domain is generally transformed into a computational domain. The transformation is such that points in the physical domain are mapped one-for-one into a unit square computational space, as shown in Figure 2. Note that in the computational domain the spacing is uniform throughout. Also, the airfoil surface maps to the $\eta=0$ line in the computational domain which greatly simplifies specification of the surface boundary conditions.

Several techniques are available for performing this transformation, including algebraic methods and methods requiring the solution of elliptic or hyperbolic partial differential equations. Each has advantages and limitations. In this study, airfoil grids were generated using both partial differential equation methods.

Initial work was done using an elliptic grid generator developed by David Amdahl (24) of the Flight Dynamics Laboratory at Wright-Patterson AFB, Ohio. The code worked very well, but the number of iterations required to solve

the elliptic equations translated into long computer running times for each grid. Also, the version of the code used in this study requires the user to specify the outer boundary points in addition to the inner boundary or airfoil surface points. Given these limitations, an alternative grid generation method was chosen.

Subsequent grids were generated using a hyperbolic grid generator developed by Steger and Chaussee, and coded by Kinsey and Barth (25). The solution of the hyperbolic system of equations used in this code was obtained much faster than with the elliptic grid generator. Also, since the hyperbolic equations are marched in space, only an inner boundary must be specified by the user. The program allows for selection and modification of the desired outer boundary shape. For example, C-shaped or O-shaped grids can be generated about an airfoil simply by adjusting program default parameters. In the present study, C-grids were generated. A grid with 299 points in the ξ -direction and 65 points in the η -direction is shown in Figure 3.

The hyperbolic grid generator offers complete control over grid point clustering parallel and normal to the surface. For the iced airfoil, it is necessary to cluster more points around the leading edge region to capture the rapid changes in the flow over the ice shape. The grid generator allows the user to specify the minimum spacing along the surface at any point, either on the body or in the wake. Once these points are all defined, the user can

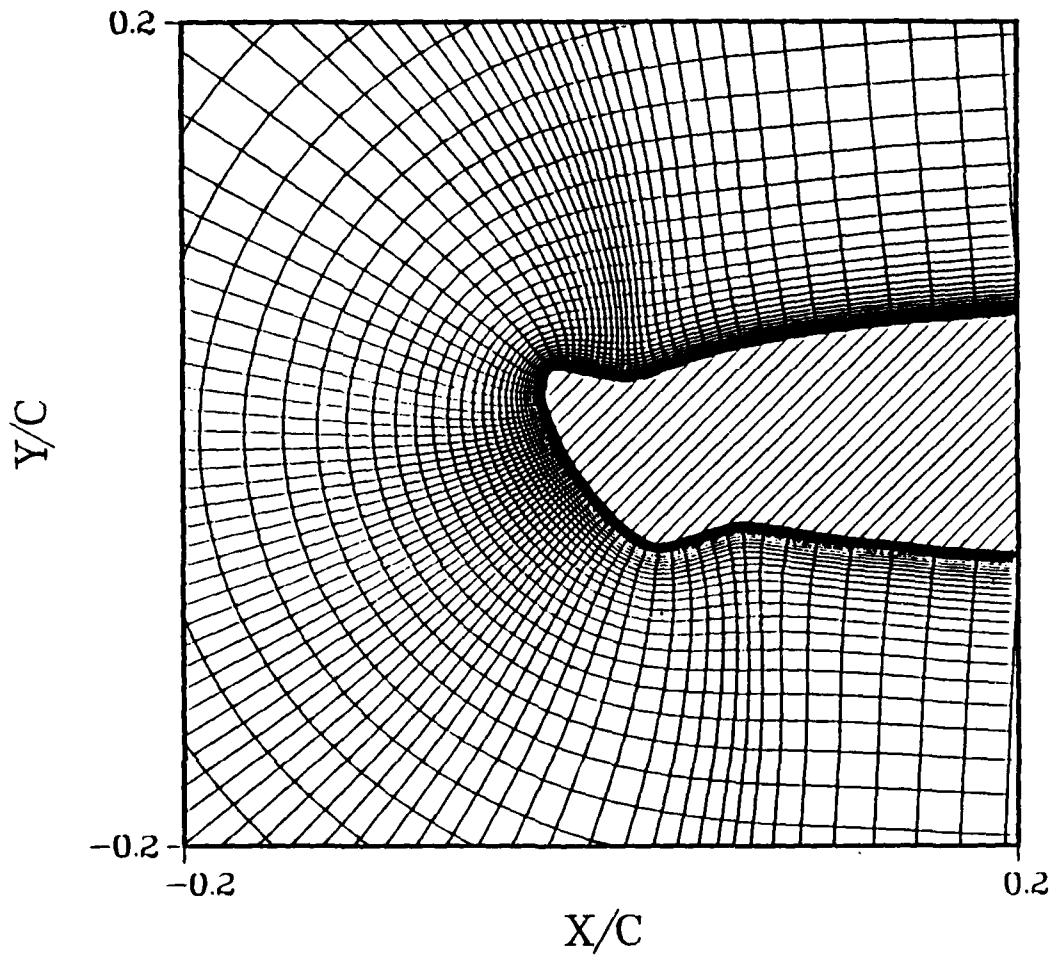


Figure 3. Typical C-Grid for an Iced NACA 0012 Airfoil

specify the number of points needed between any two minimum spacing locations. Exponential stretching functions are used to space these grid points on the surface or in the wake about the minimum spacing points previously defined.

Similarly, the code allows the user to specify the minimum spacing required normal to the airfoil surface. This provides the control needed to cluster more points in the boundary layer where rapid flow changes are occurring. Again, exponential stretching is applied to space the remaining points normal to the surface.

Boundary Conditions

For solution of the compressible Navier-Stokes equations, boundary conditions must be specified at the grid outer boundary and on the airfoil surface. It is assumed that the outer boundary is far enough away from the airfoil that free stream conditions can be applied. Therefore, in the computational domain at $\eta = \eta_{\max}$, the boundary conditions are

$$\begin{aligned} p &= p_{\infty} \\ \rho &= \rho_{\infty} \\ u &= V_{\infty} \cos(\alpha) \\ v &= V_{\infty} \sin(\alpha) \end{aligned} \quad (55)$$

where

V_{∞} = magnitude of the free stream velocity
 α = airfoil angle of attack

At the downstream boundaries, $\xi=0$ and $\xi=\xi_{\max}$, the specified boundary conditions are

$$\begin{aligned} \frac{\partial \rho}{\partial \xi} &= 0 \\ \frac{\partial u}{\partial \xi} &= 0 \\ \frac{\partial v}{\partial \xi} &= 0 \\ T &= T_{\infty} \end{aligned} \tag{56}$$

For the viscous flow on the airfoil surface, the no-slip boundary condition is enforced. An adiabatic wall is assumed, and the pressure at the surface is assumed to be equal to the pressure one grid point above the surface. Thus, for $\eta=0$, the boundary conditions are

$$\begin{aligned} u &= v = 0 \\ \frac{\partial T}{\partial \eta} &= 0 \\ \frac{\partial p}{\partial \eta} &= 0 \end{aligned} \tag{57}$$

In the wake, continuity must be maintained across the wake cut. This is accomplished by averaging values of ρ , u , v , and p across the wake cut and using these average values as the boundary condition on the wake line.

III. Numerical Procedures

The numerical methods used to solve the governing equations are described in this chapter. Two numerical schemes have been used to study the flow over a NACA 0012 airfoil--the Beam-Warming finite-difference scheme (20,26) and a space marching algorithm developed by Halim (19). The Beam-Warming scheme is an implementation of the full compressible Navier-Stokes equations. The actual code used was written by Visbal (23). The Halim formulation is an implementation of the incompressible Navier-Stokes equations utilizing the thin shear layer approximation. It was originally written and applied to symmetric airfoils at zero angle of attack. By taking advantage of the symmetry condition, the solution only needed to be computed over half the domain. The author and a fellow AFIT student, Capt Faran Hafeez (27), worked to modify the code to handle computations in the full domain.

Beam-Warming Scheme

The Beam-Warming scheme is a time marching implicit algorithm for the solution of the compressible Navier-Stokes equations. The general difference formula for the scheme is given by

$$\Delta U^n = \frac{\theta_1 \Delta t}{1+\theta_2} \frac{\partial}{\partial t} (\Delta U^n) + \frac{\Delta t}{1+\theta_2} \frac{\partial}{\partial t} (U^n) + \frac{\theta_2}{1+\theta_2} \Delta U^{n-1} + O \left[(\theta_1 - 1/2 - \theta_2) (\Delta t)^2 + (\Delta t)^3 \right] \quad (58)$$

where

$$\Delta U^n = U^{n+1} - U^n \quad (59)$$

The temporal accuracy of the scheme is determined by the choice of the parameters θ_1 , and θ_2 . The Euler implicit scheme is obtained for $\theta_1 = 1$ and $\theta_2 = 0$. This scheme is first order accurate in time. A variety of second order accurate schemes can be obtained by specifying $\theta_1 = \theta_2 + 1/2$. The specific code used in this thesis uses first order accurate Euler implicit time differencing. This accuracy is sufficient in the present work since only the steady-state solutions are of interest.

Recall that the Navier-Stokes equations are given by

$$\begin{aligned} \frac{\partial}{\partial t} (\hat{U}) + \frac{\partial}{\partial \xi} (E_1) + \frac{\partial}{\partial \eta} (E_2) = \\ \frac{\partial}{\partial \xi} \left[v_1(\hat{U}, \hat{U}_\xi) \right] + \frac{\partial}{\partial \xi} \left[v_2(\hat{U}, \hat{U}_\eta) \right] \\ + \frac{\partial}{\partial \eta} \left[w_1(\hat{U}, \hat{U}_\xi) \right] + \frac{\partial}{\partial \eta} \left[w_2(\hat{U}, \hat{U}_\eta) \right] \quad (54) \end{aligned}$$

If this equation is written at time levels $n+1$ and n , and the equation at level n is subtracted from the equation at level $n+1$, the result is

$$\begin{aligned} \frac{\partial}{\partial t} \left(\Delta \hat{U} \right) + \frac{\partial}{\partial \xi} \left(\Delta E_1 \right) + \frac{\partial}{\partial \eta} \left(\Delta E_2 \right) = \\ \frac{\partial}{\partial \xi} \left[\Delta V_1 \left(\hat{U}, \hat{U}_\xi \right) \right] + \frac{\partial}{\partial \xi} \left[\Delta V_2 \left(\hat{U}, \hat{U}_\eta \right) \right] \\ + \frac{\partial}{\partial \eta} \left[\Delta W_1 \left(\hat{U}, \hat{U}_\xi \right) \right] + \frac{\partial}{\partial \eta} \left[\Delta W_2 \left(\hat{U}, \hat{U}_\eta \right) \right] \quad (60) \end{aligned}$$

This is the so-called delta form of the equations, where

$$\begin{aligned} \Delta U^n &= U^{n+1} - U^n \\ \Delta E_1^n &= E_1^{n+1} - E_1^n \\ \Delta E_2^n &= E_2^{n+1} - E_2^n \\ \Delta V_1^n &= V_1^{n+1} - V_1^n \\ \Delta V_2^n &= V_2^{n+1} - V_2^n \\ \Delta W_1^n &= W_1^{n+1} - W_1^n \\ \Delta W_2^n &= W_2^{n+1} - W_2^n \end{aligned} \quad (61)$$

Now Eq (60) can be solved for $\Delta \hat{U}_t^n$ and the result substituted into Eq (58) to yield

$$\begin{aligned}
\Delta \hat{U}^n = & \frac{\theta_1 \Delta t}{1+\theta_2} \left\{ \frac{\partial}{\partial \xi} \left(-\Delta E_1^n + \Delta V_1^n + \Delta V_2^n \right) + \frac{\partial}{\partial \eta} \left(-\Delta E_1^n + \Delta V_1^n + \Delta V_2^n \right) \right\} \\
& + \frac{\Delta t}{1+\theta_2} \left\{ \frac{\partial}{\partial \xi} \left(-E_1^n + v_1^n + v_2^n \right) + \frac{\partial}{\partial \eta} \left(-E_2^n + w_1^n + w_2^n \right) \right\} \\
& + \frac{\theta_2}{1+\theta_2} \Delta \hat{U}^{n-1} + \left\{ \left(\theta_1 - 1/2 - \theta_2 \right) (\Delta t)^2 + (\Delta t)^3 \right\} \quad (62)
\end{aligned}$$

For the Beam-Warming algorithm, the spatial derivatives in Eq (62) cannot be approximated by finite differences yet because the equation is nonlinear. This arises from the fact that the flux vector increments, ΔE_1 , ΔE_2 , ΔV_1 , ΔV_2 , ΔW_1 , and ΔW_2 , are nonlinear functions of \hat{U} . Each of the flux vectors must be linearized before differencing the equation.

Linearization of the Flux Vectors. Each of the flux vectors can be linearized using a truncated Taylor series expansion in \hat{U} . For example, ΔE_1^n can be linearized from

$$E_1^{n+1} = E_1^n + \left(\frac{\partial E_1}{\partial U} \right) \left(\hat{U}^{n+1} - \hat{U}^n \right) + O(\Delta t)^2 \quad (63)$$

which can be expressed as

$$\Delta E_1 = \left[A \right]^n \Delta \hat{U}^n + O(\Delta t)^2 \quad (64)$$

where $\left[A \right]^n$ is the Jacobian matrix given by

$$\left[A \right]^n = \left(\frac{\partial E_1}{\partial U} \right)^n \quad (65)$$

Similarly, ΔE_2 is linearized as

$$\Delta E_2 = \left[B \right]^n \Delta \hat{U}^n + O(\Delta t)^2 \quad (66)$$

where $\left[B \right]^n$ is the Jacobian matrix given by

$$\left[B \right]^n = \left(\frac{\partial E_2}{\partial U} \right)^n \quad (67)$$

The viscous delta terms are linearized using the method suggested by Steger (28). In this method, the coefficients of viscosity, μ and ϵ , and the thermal conductivities, k and k_T , are assumed to be locally independent of \hat{U} . In addition, the cross derivative terms, $\Delta V_{2\xi}^n(\hat{U}, \hat{U}_\eta)$ and $\Delta W_{1\eta}^n(\hat{U}, \hat{U}_\xi)$, are neglected. With these approximations, the viscous terms are linearized as

$$\Delta V_1^n = \left(\left[R \right]^n \Delta \hat{U}^n \right)_\xi + O(\Delta t)^2 \quad (68)$$

$$\Delta W_2^n = \left(\left[S \right]^n \Delta \hat{U}^n \right)_\eta + O(\Delta t)^2 \quad (69)$$

where $\left[R \right]^n$ and $\left[S \right]^n$ are the Jacobian matrices

$$[R]^n = \left(\frac{\partial v_1}{\partial u_\xi} \right)^n \quad (70)$$

$$[S]^n = \left(\frac{\partial w_2}{\partial u_\eta} \right)^n \quad (71)$$

The four Jacobian matrices resulting from the linearization of the governing equations are presented in Appendix C.

Substituting the linearized flux vectors into Eq (62) yields

$$\begin{aligned} & \left\{ [I] + \frac{\theta_1 \Delta t}{1+\theta_2} \left(\frac{\partial [A]}{\partial \xi} - \frac{\partial^2 [R]}{\partial \xi^2} + \frac{\partial [B]}{\partial \eta} - \frac{\partial^2 [S]}{\partial \eta^2} \right) \right\} \Delta \hat{U}^n \\ & = \frac{\Delta t}{1+\theta_2} \left\{ \frac{\partial}{\partial \xi} (-\epsilon_1^n + v_1^n + v_2^n) + \frac{\partial}{\partial \eta} (-\epsilon_2^n + w_1^n + w_2^n) \right\} \\ & + \frac{\theta_2}{1+\theta_2} \Delta \hat{U}^{n-1} + \left\{ (\theta_1 - 1/2 - \theta_2) (\Delta t)^2 + (\Delta t)^3 \right\} \quad (72) \end{aligned}$$

where

$$[I] = \text{identity matrix}$$

and notation such as

$$\left(\frac{\partial [A]}{\partial \xi} \right) \Delta \hat{U}^n$$

implies

$$\frac{\partial}{\partial \xi} \left([A] \Delta \hat{U}^n \right)$$

Approximate Factorization. Beam and Warming (20) have shown that spatial factorization of Eq (72) results in a set of equations that can be solved using an alternating direction numerical scheme. Specifically, the left hand side (LHS) OF Eq (72) can be factored into

$$\left\{ [I] + \frac{\theta_1 \Delta t}{1+\theta_2} \left(\frac{\partial [A]}{\partial \xi} - \frac{\partial^2 [R]}{\partial \xi^2} \right)^n \right\} \times \left\{ [I] + \frac{\theta_1 \Delta t}{1+\theta_2} \left(\frac{\partial [B]}{\partial \eta} - \frac{\partial^2 [S]}{\partial \eta^2} \right)^n \right\} \Delta \hat{U}^n + O(\Delta t)^3 \quad (73)$$

Since the term remaining after factorization is $O(\Delta t)^3$, and the leading truncation error term in Eq (72) is at best $O(\Delta t)^2$, the factorization does not affect the temporal accuracy of the scheme.

In practice, Eq (72) is implemented in three steps as follows:

Step 1.

$$\left\{ [I] + \frac{\theta_1 \Delta t}{1+\theta_2} \left(\frac{\partial [A]}{\partial \xi} - \frac{\partial^2 [R]}{\partial \xi^2} \right)^n \right\} \Delta \hat{U}^{*n} = \text{RHS} \left[\text{Eq (72)} \right] \quad (74)$$

Step 2.

$$\left\{ \left[\mathbf{I} \right] + \frac{\theta_1 \Delta t}{1 + \theta_2} \left(\frac{\partial \left[\mathbf{B} \right]^n}{\partial \eta} - \frac{\partial^2 \left[\mathbf{S} \right]^n}{\partial \eta^2} \right) \right\} \Delta \hat{\mathbf{U}}^n = \Delta \hat{\mathbf{U}}^{*n} \quad (75)$$

Step 3.

$$\hat{\mathbf{U}}^{n+1} = \hat{\mathbf{U}}^n + \Delta \hat{\mathbf{U}}^n \quad (76)$$

where

$\Delta \hat{\mathbf{U}}^{*n}$ = intermediate solution vector

RHS [Eq (72)] = terms on right hand side of Eq (72)

The equations above are written in terms of a general time differencing algorithm. The code used in this thesis incorporated first-order Euler implicit time differencing, obtained by setting $\theta_1 = 1$ and $\theta_2 = 0$. With these substitutions for θ_1 and θ_2 , Eqs (74) and (75) simplify to

$$\begin{aligned} & \left\{ \left[\mathbf{I} \right] + \Delta t \left(\frac{\partial \left[\mathbf{A} \right]^n}{\partial \xi} - \frac{\partial^2 \left[\mathbf{R} \right]^n}{\partial \xi^2} \right) \right\} \Delta \hat{\mathbf{U}}^{*n} \\ & = \Delta t \left\{ \frac{\partial}{\partial \xi} \left(-E_1^n + v_1^n + v_2^n \right) + \frac{\partial}{\partial \eta} \left(-E_2^n + w_1^n + w_2^n \right) \right\} \end{aligned} \quad (77)$$

$$\left\{ [I] + \Delta t \left(\frac{\partial [B]^n}{\partial \eta} - \frac{\partial^2 [S]^n}{\partial \eta^2} \right) \right\} \Delta \hat{U}^n = \Delta \hat{U}^{*n} \quad (78)$$

The spatial derivatives in Eqs (77) and (78) are written as second-order central differences. Visbal (23) shows that this yields

$$\begin{aligned} & \left\{ [I] + \Delta t \left(\mu_{\xi} [A]_{i,j} - \delta_{\xi}^2 [R]_{i,j} \right) \right\} \Delta \hat{U}_{i,j}^{*n} \\ &= -\Delta t \left\{ \mu_{\xi} (E_1 - V_2)_{i,j} + \mu_{\eta} (E_2 - W_1)_{i,j} - \delta_{\xi} V_{1,i,j} - \delta_{\eta} W_{2,i,j} \right\} \end{aligned} \quad (79)$$

and

$$\left\{ [I] + \Delta t \left(\mu_{\eta} [B]_{i,j} - \delta_{\eta}^2 [S]_{i,j} \right) \right\} \Delta \hat{U}_{i,j}^n = \Delta \hat{U}_{i,j}^{*n} \quad (80)$$

where

$$\begin{aligned} \delta_{\xi} f_{i,j} &= \left(f_{i+1/2,j} - f_{i-1/2,j} \right) / \Delta \xi \\ \delta_{\eta} f_{i,j} &= \left(f_{i,j+1/2} - f_{i,j-1/2} \right) / \Delta \eta \\ \mu_{\xi} f_{i,j} &= \left(f_{i+1,j} - f_{i-1,j} \right) / 2\Delta \xi \\ \mu_{\eta} f_{i,j} &= \left(f_{i,j+1} - f_{i,j-1} \right) / 2\Delta \eta \end{aligned} \quad (81)$$

are finite difference operators.

This discretization of the equations results in a 4X4 block tridiagonal system of linear equations to be solved. The solution is obtained by first sweeping the domain in the ξ -direction, solving Eq (79) at each η -line. This sweep is followed by sweeping the domain in the η -direction, solving Eq (80) along each ξ -line. The result of both sweeps is a new estimate for the unknowns, $\hat{\Delta U}^n$.

Added Dissipation Terms. In the application of their algorithm, Beam and Warming have found it necessary to add artificial dissipation terms to damp short wave length oscillations (26). In Visbal's implementation of the Beam-Warming scheme, both explicit fourth-order damping terms and implicit second-order damping terms have been added for numerical stability. Explicit damping is included by adding the term

$$\omega_E \Delta t J_{i,j}^{-1} \left(\delta_\xi^4 + \delta_\eta^4 \right) \hat{U}_{i,j}^n \quad (82)$$

to the RHS of Eq (79), while implicit damping is incorporated by including the terms

$$- \omega_I \Delta t J_{i,j}^{-1} \delta_\xi^2 J_{i,j} \left[I \right] \quad (83)$$

and

$$- \omega_I \Delta t J_{i,j}^{-1} \delta_\eta^2 J_{i,j} \left[I \right] \quad (84)$$

on the LHS of Eqs (79) and (80). In these expressions, ω_E and ω_I are the explicit and implicit damping factors, respectively. Anderson and Tannehill (21) suggest ω_E be less than 0.0625 for stability, and Visbal (23) and Pulliam (17) recommend that

$$\omega_I \geq 2\omega_E \quad (85)$$

Note that the addition of the second-order damping terms does not affect the steady-state solution, since $\Delta \hat{U}^n$ approaches zero for this condition.

Halim's Formulation of the Thin-Layer Equations

Another numerical method considered in the present study was that developed by Halim (19) for steady, incompressible, laminar flow over arbitrary airfoils. This code employs the thin-layer form of the Navier-Stokes. The thin-layer approximation assumes the viscous shear stresses parallel to the body are negligible compared to the shear stress normal to the body. Therefore, the streamwise diffusion terms are dropped from the Navier-Stokes equations.

Halim's formulation of the thin-layer equations is written in terms of the stream function and vorticity rather than the primitive variables. The advantage of this formulation for 2-D flow is that the number of unknowns is reduced by one. In terms of stream function and vorticity,

the governing equations can be written in generalized coordinates as

$$\frac{\partial}{\partial \xi}(\omega \psi_{\eta}) - \frac{\partial}{\partial \eta}(\omega \psi_{\xi}) = \frac{1}{Re_c} \frac{\partial}{\partial \eta} \left(\frac{\gamma}{J} \frac{\partial \omega}{\partial \eta} \right) \quad (86)$$

and

$$J\omega + \frac{\partial}{\partial \xi} \left(\frac{\alpha}{J} \frac{\partial \psi}{\partial \xi} - \frac{\beta}{J} \frac{\partial \psi}{\partial \eta} \right) + \frac{\partial}{\partial \eta} \left(\frac{\gamma}{J} \frac{\partial \psi}{\partial \eta} - \frac{\beta}{J} \frac{\partial \psi}{\partial \xi} \right) = 0 \quad (87)$$

where the metrics and Jacobian are given by

$$\begin{aligned} \alpha &= x_{\eta}^2 + y_{\eta}^2 \\ \beta &= x_{\xi} x_{\eta} + y_{\xi} y_{\eta} \\ \gamma &= x_{\xi}^2 + y_{\xi}^2 \\ J &= x_{\xi} y_{\eta} - x_{\eta} y_{\xi} \end{aligned} \quad (88)$$

The vorticity, ω , is defined as

$$\omega = \frac{\partial v}{\partial x} - \frac{\partial u}{\partial y} \quad (89)$$

and the stream function is defined by the equations

$$\frac{\partial \psi}{\partial y} = u \quad (90)$$

$$\frac{\partial \psi}{\partial x} = -v \quad (91)$$

Linearization of the Vorticity Equation. Equation (86)

is nonlinear due to the products $\omega\psi_\xi$ and $\omega\psi_\eta$. This equation can be linearized by writing

$$\omega^{n+1} = \omega^n + \Delta\omega^n \quad (92)$$

$$\psi^{n+1} = \psi^n + \Delta\psi^n \quad (93)$$

and substituting these expressions into Eq (86). Expanding after the substitution, and neglecting higher order terms, results in

$$\begin{aligned} \psi_\eta^n \Delta\omega_\xi^n + \omega_\xi^n \Delta\psi_\eta^n - \psi_\xi^n \Delta\omega_\eta^n - \omega_\eta^n \Delta\psi_\xi^n - \frac{1}{Re_c} \frac{\partial}{\partial\eta} \left(\frac{\gamma}{J} \Delta\omega_\eta^n \right) \\ = -\psi_\eta^n \omega_\xi^n + \psi_\xi^n \omega_\eta^n + \frac{1}{Re_c} \frac{\partial}{\partial\eta} \left(\frac{\gamma}{J} \omega_\eta^n \right) \end{aligned} \quad (94)$$

The equation is now linear in the unknowns $\Delta\omega$ and $\Delta\psi$.

Equation (87) is already linear, but it can be put in delta form by applying Eqs (92) and (93), which gives

$$\begin{aligned} J\Delta\omega^n + \frac{\partial}{\partial\xi} \left(\frac{\alpha}{J} \frac{\partial\Delta\psi^n}{\partial\xi} - \frac{\beta}{J} \frac{\partial\Delta\psi^n}{\partial\eta} \right) \\ + \frac{\partial}{\partial\eta} \left(\frac{\gamma}{J} \frac{\partial\Delta\psi^n}{\partial\eta} - \frac{\beta}{J} \frac{\partial\Delta\psi^n}{\partial\xi} \right) = -J\omega^n \\ - \frac{\partial}{\partial\xi} \left(\frac{\alpha}{J} \frac{\partial\psi^n}{\partial\xi} - \frac{\beta}{J} \frac{\partial\psi^n}{\partial\eta} \right) - \frac{\partial}{\partial\eta} \left(\frac{\gamma}{J} \frac{\partial\psi^n}{\partial\eta} - \frac{\beta}{J} \frac{\partial\psi^n}{\partial\xi} \right) \end{aligned} \quad (95)$$

Equations (94) and (95) are discretized using second-order central differences for all terms except $\psi_\eta \Delta\omega_\xi$, which is

treated as an upwind difference for stability. A 2X2 block tridiagonal system of equations results, which is solved using the Thomas algorithm (21:99)

Boundary Conditions in Terms of Stream Function and Vorticity. The boundary conditions used for this algorithm are similar to those applied for the Beam-Warming scheme. However, the appropriate boundary conditions must now be specified implicitly in terms of ψ and ω .

On the airfoil surface, the velocity components are zero. Specifying this condition in terms of the stream function requires applying the chain rule to the stream function definition, which yields

$$u = \psi_y = \psi_\xi \xi_y + \psi_\eta \eta_y \quad (96)$$

and

$$v = -\psi_x = -(\psi_\xi \xi_x + \psi_\eta \eta_x) \quad (97)$$

For u and v equal to zero on the surface, Eqs (96) and (97) can be combined to yield

$$\psi_\xi (\xi_x + \xi_y) + \psi_\eta (\eta_y + \eta_x) = 0 \quad (98)$$

Now, since $\eta = 0$ corresponds to the airfoil surface, this η -line is a streamline along which $\psi = 0$ everywhere. Consequently, the rate of change of ψ with respect to ξ , ψ_ξ , must be zero along this line. Then, noting that the metrics

are not zero in general, the requirement for zero velocity on the airfoil surface can only be met by prescribing

$$\psi_{\eta} = 0 \quad \text{at} \quad \eta = 0 \quad (99)$$

which is the implicit form of the surface boundary condition for ψ . The vorticity on the airfoil surface can be determined from the definition of vorticity,

$$\omega = -(\psi_{xx} + \psi_{yy}) \quad (100)$$

Since $v = 0$ everywhere on the surface, however, the rate of change of v with respect to x is also zero. So, the vorticity on the surface simplifies to

$$\omega = -\psi_{yy} \quad (101)$$

This boundary condition can be written in terms of the computational domain variables by applying the chain rule, which gives

$$-\omega = \xi_y \frac{\partial}{\partial \xi} (\psi_{\xi} \xi_y + \psi_{\eta} \eta_y) + \eta_y \frac{\partial}{\partial \eta} (\psi_{\xi} \xi_y + \psi_{\eta} \eta_y) \quad (102)$$

This equation can be simplified, however, by noting that both ψ_{ξ} and the rate of change of ψ_{η} with respect to ξ are zero at $\eta = 0$, so that

$$-\omega = \eta_y \frac{\partial}{\partial \eta} (\psi_\eta \eta_y) \quad \text{at } \eta = 0 \quad (103)$$

which is the implicit form of the surface boundary condition for ω .

The boundary conditions in the far field are derived by assuming the outer boundary is far enough away from the airfoil that uniform flow at free stream conditions prevails. Thus, for $\xi = 0$ and $\xi = \xi_{\max}$,

$$\begin{aligned} u &= V_\infty \\ v &= 0 \\ \omega &= 0 \end{aligned} \quad (104)$$

or alternatively,

$$\begin{aligned} \psi_\eta &= V_\infty / \eta_y \\ \psi_\xi &= 0 \\ \omega &= 0 \end{aligned} \quad (105)$$

The boundary conditions along the wake cut depend on the airfoil and angle of attack. For a symmetric airfoil at zero angle of attack, the wake cut can be treated as a streamline and the appropriate boundary conditions are

$$\begin{aligned} \psi &= 0 \\ \omega &= 0 \end{aligned} \quad (106)$$

These boundary conditions assume uniform flow in the streamwise direction and no component of velocity in the y

or η direction. Therefore, the wake boundary condition must be modified for non-symmetric airfoils or airfoils at angle of attack. Hafeez (27) is currently working on this modification.

Modification of the Original Code. The original computer program solved for flow over a symmetric airfoil at zero angle of attack. The code was written to take advantage of the symmetry by computing the solution for only half the domain. As a first step in generalizing the program to non-symmetric airfoils, the author and fellow student Capt Faran Hafeez, modified the original code to extend the computations over the entire domain. The modified code was tested by applying it to a NACA 0012 airfoil at zero angle of attack, for which the solution had been previously computed with the original code. Due to problems in applying the implicit boundary conditions correctly, all the time available to devote to this algorithm was spent extending the scheme to the full domain and testing the modifications. Therefore, two further modifications of this code are needed before it can be applied to an iced airfoil.

First of all, a suitable wake boundary condition must be derived for flows at angle of attack or flow over non-symmetric airfoils. One possibility is to average the velocities across the wake cut, as is done in the Beam-Warming code. The task is to obtain expressions for the average velocities in terms of ω and ψ , however. As

mentioned previously, Hafeez (27) is working to incorporate a suitable wake boundary condition. The second modification needed is the addition of a turbulence model. The separated flow behind the glaze ice shape on an iced airfoil is definitely turbulent and accurate results cannot be expected unless this phenomena is modeled. Hafeez (27) is in the process of modifying the code to handle turbulent flow.

IV. Results and Discussion

The Beam-Warming algorithm was applied to an iced NACA 0012 airfoil at angles of attack ranging from 2-8 degrees. The criteria used to define a fully converged or steady-state solution are presented in this section. Also, the global results of this study are presented in the form of lift and drag curves, and pressure coefficient plots for the flowfield. Some local flowfield results are presented as well. These include u-velocity component contours over the leading edge region for 2.0, 4.0, 6.0, and 8.0 degrees angle of attack, and velocity vector plots covering the separated flow regions. The u-velocity contours illustrate the separation and reattachment of the flow over the ice horns, and the velocity vector plots detail the reverse flow in the separation bubbles. The velocity profile on the upper surface at $x/c=0.0$ is also plotted and compared to Spring's experimental data. Computed and experimental flow reattachment locations are compared as well.

In a separate section, the results of the work done to extend Halim's formulation to the full domain are presented. Streamline contours for a clean NACA 0012 airfoil at zero angle of attack are presented, along with a plot of the skin friction coefficient over the airfoil.

Results for an Iced NACA 0012 Using the Beam-Warming Code

The Beam-Warming code developed by Visbal was used to compute the flowfield over an iced NACA 0012 airfoil with the same geometry as that tested experimentally by Bragg and Coirier. Computations are for flow at a free stream Mach number of 0.12 with a Reynolds number of 1.41×10^6 . Transition from laminar to turbulent flow is fixed at $x/c = -0.018$ on the upper surface, and at $x/c = .0327$ on the lower surface. These locations correspond to grid points one step upstream of flow separation. Steady-state numerical solutions were obtained for angles of attack from 2-8 degrees. Comparison of these solutions is made with experimental data and other numerical results.

The grid used in these computations is comprised of 299 points in the direction around the airfoil, and 65 points normal to the airfoil. There are 201 points on the airfoil surface to define the airfoil and ice shape, leaving 49 points on each side of the wake cut extending to the outer boundary. The outer boundary is located 20 chord lengths from the airfoil. The minimum spacing normal to the airfoil surface is 2.0×10^{-3} chord lengths. The minimum spacing along the airfoil surface is .001 chord lengths, specified at the airfoil trailing edge. Deviation from the normal NACA 0012 shape to describe the ice geometry begins on the upper and lower surfaces at 14% chord. At these positions, a minimum spacing along the surface of .02 chord lengths was

specified. The ice shape geometric data used are those reported by Khodadoust (11). A close-up of the leading edge portion of this grid is shown in Figure 3.

Convergence Criteria. Before discussing the steady-state results obtained with the Beam-Warming code, it is necessary to specify how the steady-state condition was defined. Visbal (23) has shown that convergence can be determined by monitoring the lift and drag coefficients as the solution progresses. This procedure was used to define convergence in the current study. A solution was considered fully converged when the lift coefficient changed by less than .1%, and the change in drag coefficient was less than .0002 over 1000 iterations. When these conditions were met, the root mean square of the residual was less than 1.0×10^{-8} for every solution obtained.

To illustrate the convergence behavior for a typical case, the root mean square of the residual, and the lift and drag coefficients for $\alpha=4.0$ degrees are plotted versus the number of iterations in Figures 4-6, respectively. This solution was obtained using the solution for $\alpha=2.0$ degrees as an initial condition. With a previous solution as an initial condition, note that the oscillations in the lift and drag coefficients damp out very quickly. Similar results were obtained for $\alpha=6.0$ and $\alpha=8.0$ degrees.

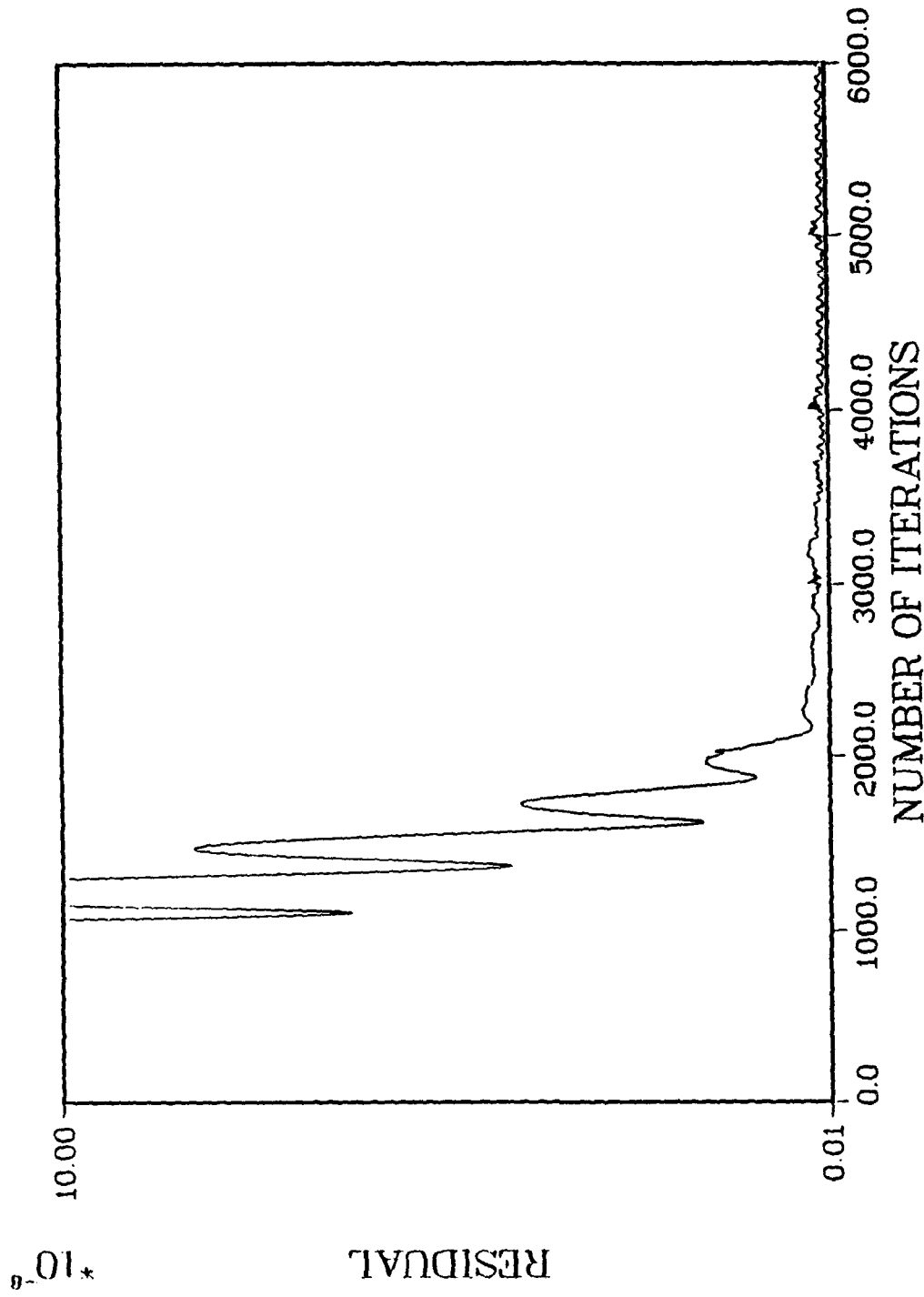


Figure 4. Behavior of the Root Mean Square of the Residual with Number of Iterations for 4.8 Degrees Angle of Attack

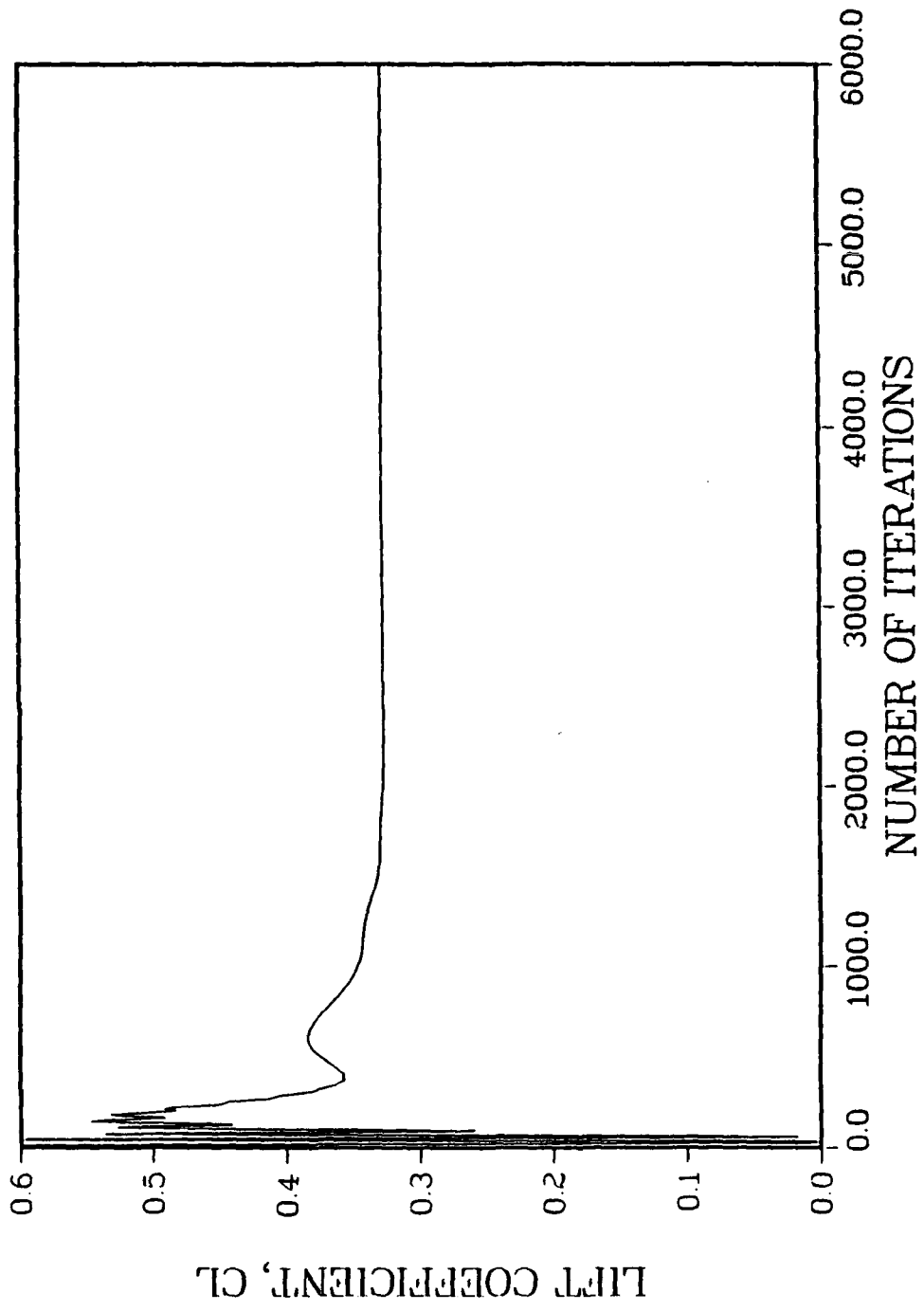


Figure 5. Behavior of the Lift Coefficient with Number of Iterations for 4.0 Degrees Angle of Attack

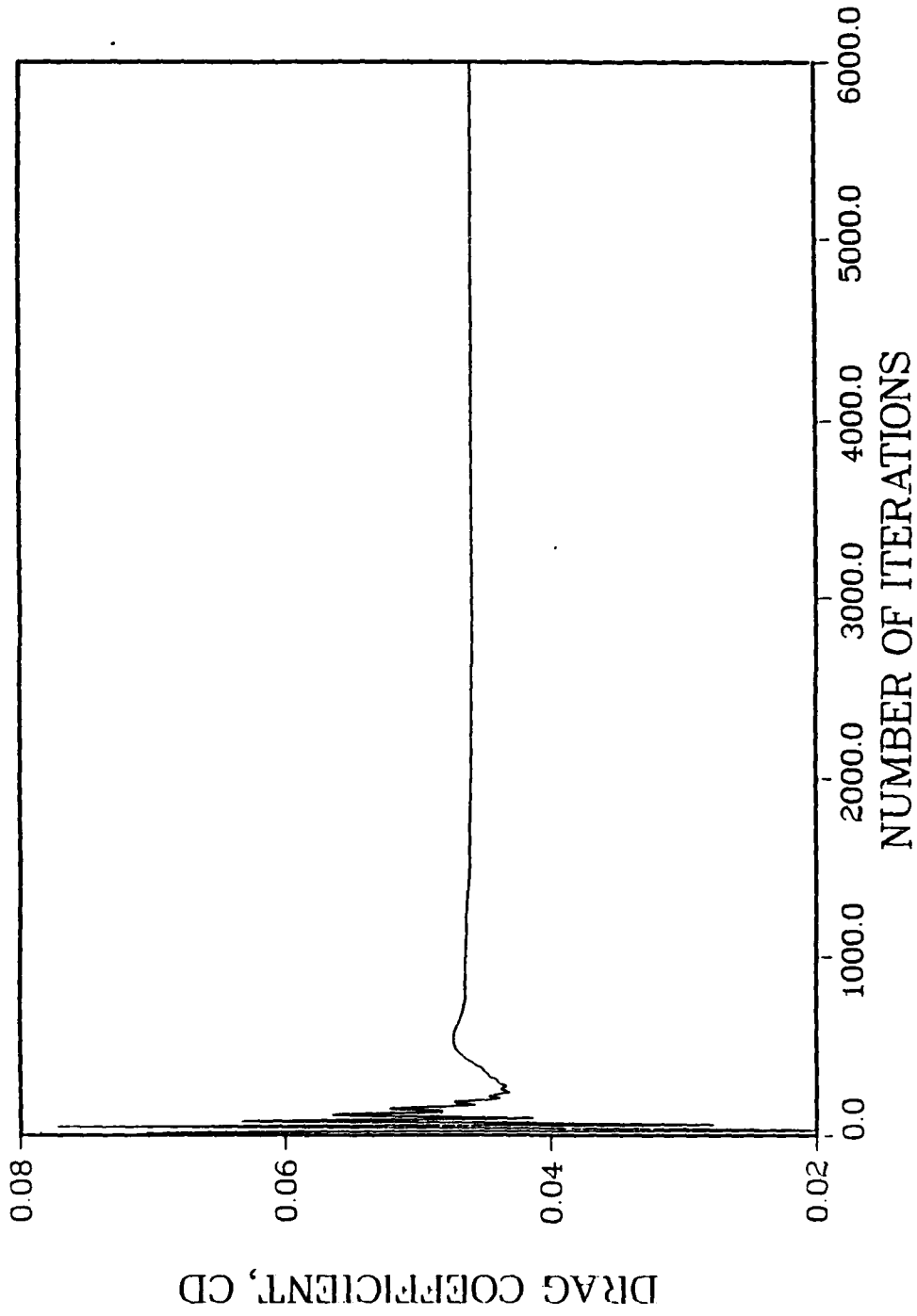


Figure 6. Behavior of the Drag Coefficient with Number of Iterations for 4.0 Degrees Angle of Attack

Global Results. The computed values of airfoil lift coefficient versus angle of attack are presented in Figure 7. Also plotted on the figure are the experimental data of Bragg and Coirier, and the numerical results of Potapczuk and Cebeci. The lift curve for a clean airfoil is included as well to illustrate the overall effect of ice on the lift.

The computed results from the Beam-Warming code compare reasonably well with the experimental data for angles of attack from 2-6 degrees, with a maximum difference of 11.5% occurring at $\alpha=6$ degrees. There is an obvious discrepancy between the experimental data and the computed solution for $\alpha=8.0$ degrees, however.

The experimental data indicate that the airfoil stalls at an angle of attack of 7.0 degrees. Increasing the angle of attack to 8.0 degrees results in a 4.6% decrease in the lift measured experimentally. This decrease is not seen in the present numerical results, as the computed lift coefficient for $\alpha=8.0$ degrees is 7.6% higher than the experimental value. The reason for this discrepancy is that the flowfield characteristic is completely different after the airfoil stalls.

For angles of attack below stall, the flow separates over the ice shape and then reattaches at some downstream location on the upper and lower surfaces. Thus, regions of separated flow, or separation bubbles, form on the airfoil. At angles of attack above stall, the flow never reattaches

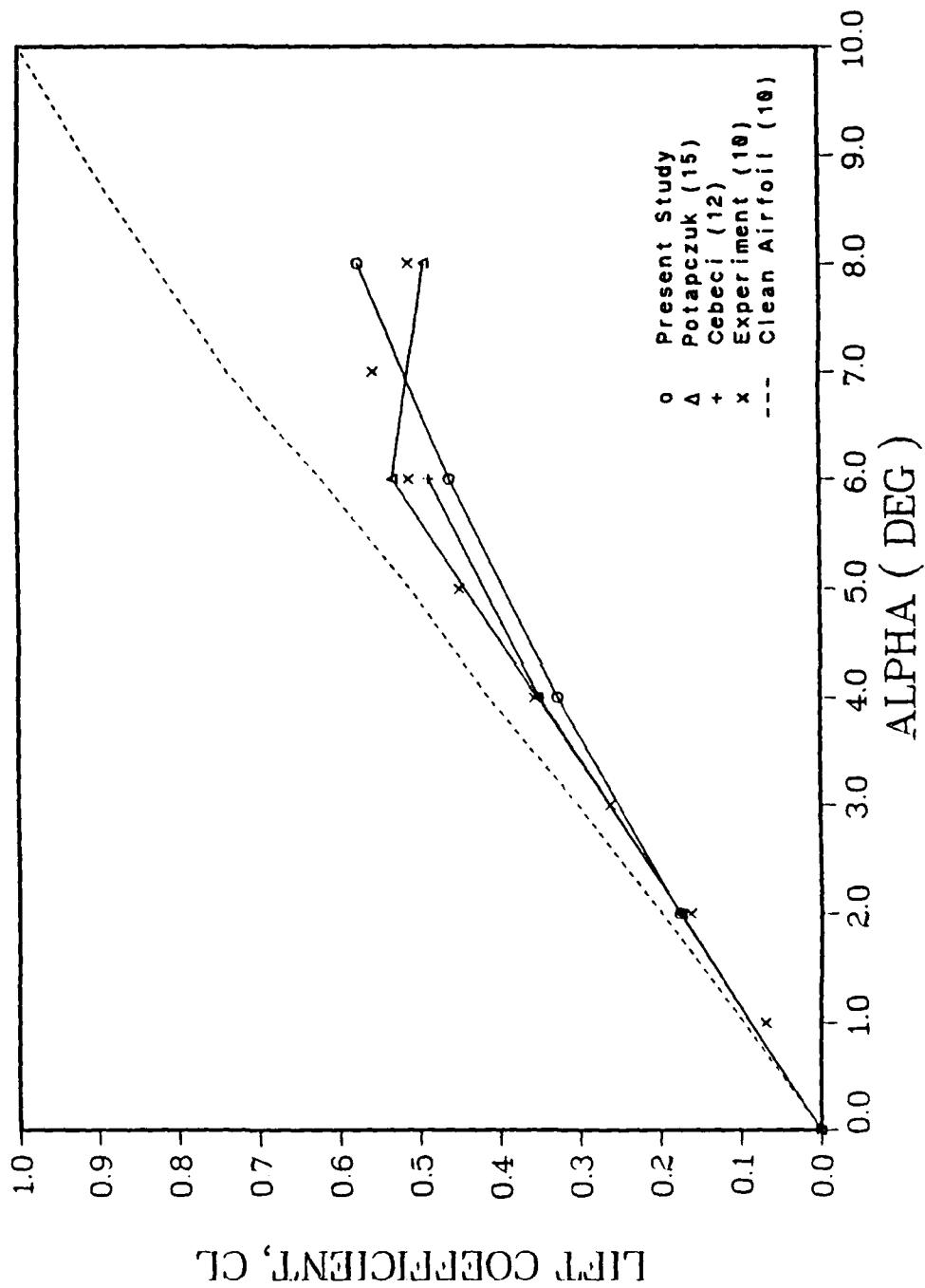


Figure 7. Lift Coefficient Curve for an Iced NACA 0012 Airfoil (Experimental Values from 10:44)

to the upper surface, however. This phenomena is pointed out in Khodadoust's flow visualization study, and Potapczuk has noted this trend in his numerical studies as well. This same flow behavior is seen in the present study also, as will be shown later

Therefore, the flow over the iced airfoil behaves quite differently before stall than it does after stall. The results in Figure 7 indicate that the Beam-Warming code does a good job modeling the separated/reattached flow which occurs before stall, but tends to over predict the lift in the completely separated flowfield which results after stall.

Further examination of Figure 7 shows that Potapczuk's result for $\alpha=8.0$ degrees correctly predicts the decreased lift expected after stall. To compute the correct post-stall lift, Potapczuk investigated the unsteady behavior of the flow for $\alpha=8.0$ degrees, which he characterizes as periodic vortex shedding. He obtained the result plotted in Figure 7 for this angle of attack by averaging the pressure coefficient values at each station over several vortex shedding periods (15). Therefore, even though a steady-state solution can be obtained for $\alpha=8.0$ degrees, it appears to be an inaccurate representation of the actual flowfield, and results in a computed lift coefficient that is too high.

The drag curve for the iced airfoil is shown in Figure 8. Drag values computed in the present study compare favorably with the experimental data for angles of attack from 2-6 degrees. The computed C_D differs from the experimental value by 24% for $\alpha=2.0$ degrees, improving to less than 10% difference at $\alpha=6.0$ degrees. Similar results can be seen for the other numerical solutions. All three schemes over estimate the drag slightly for small angles of attack, while under estimating it for larger angles of attack. For $\alpha=8.0$ degrees, once again the large discrepancy between the numerical result from the Beam-Warming code and the experimental data is attributable to the unsteady nature of the flowfield at this angle of attack. The steady-state solution computed in this study is just not accurately describing the post-stall flow characteristics.

The computed pressure coefficient over the iced airfoil is shown in Figures 9-11 for angles of attack of 2.0, 4.0, and 6.0 degrees, respectively. Spring's experimental data are plotted as well for comparison. In addition to the experimental data, comparison is also made with other numerical results. For $\alpha=2.0$ degrees, Potapczuk's numerical results are shown, while the numerical results of Cebeci are included in the plots for $\alpha=4.0$ and $\alpha=6.0$ degrees. It should be noted that the experimental data and previous numerical results were obtained by scaling plots

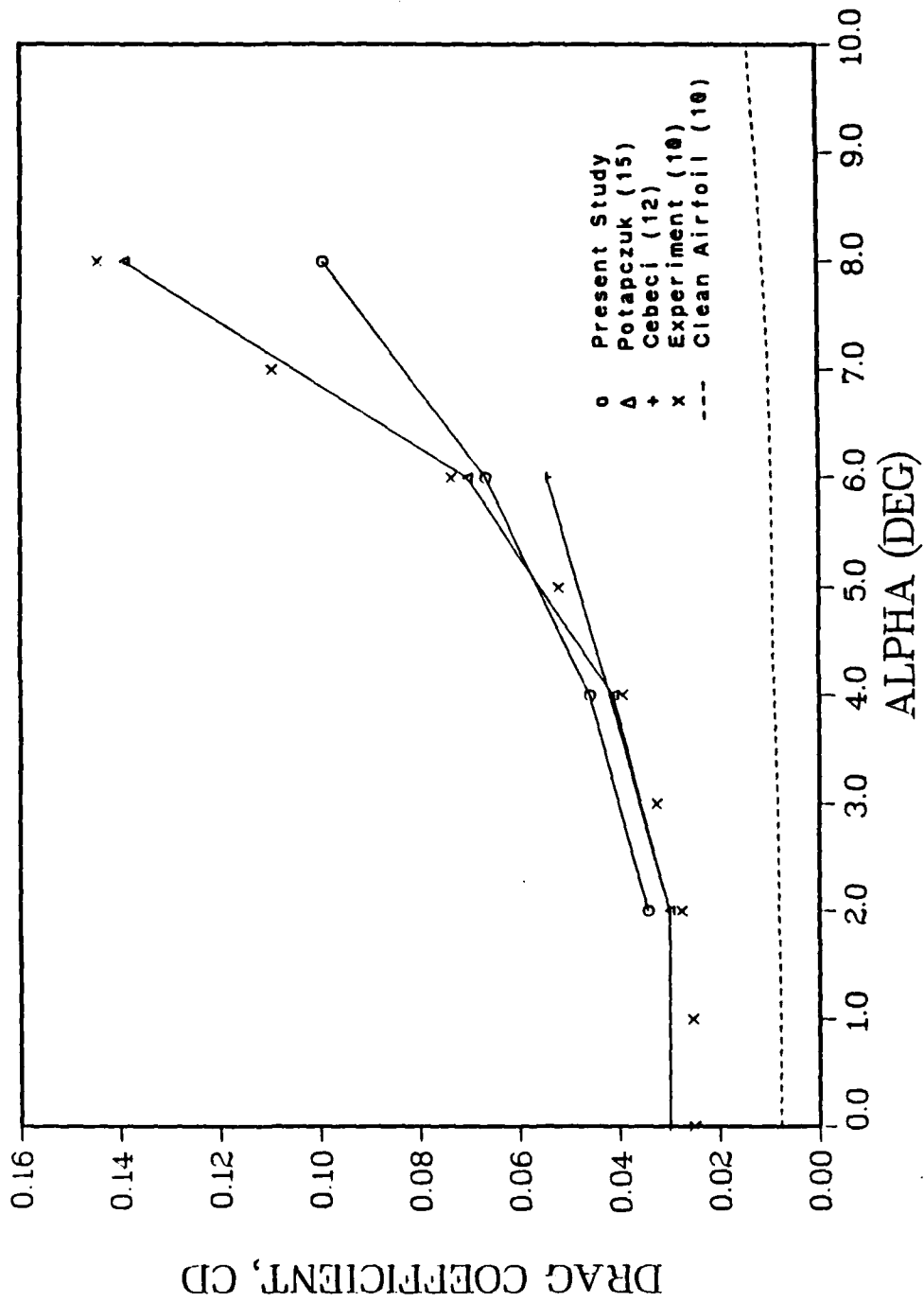


Figure 8. Drag Coefficient Curve for an Iced NACA 0012 Airfoil (Experimental Values from 10:45)

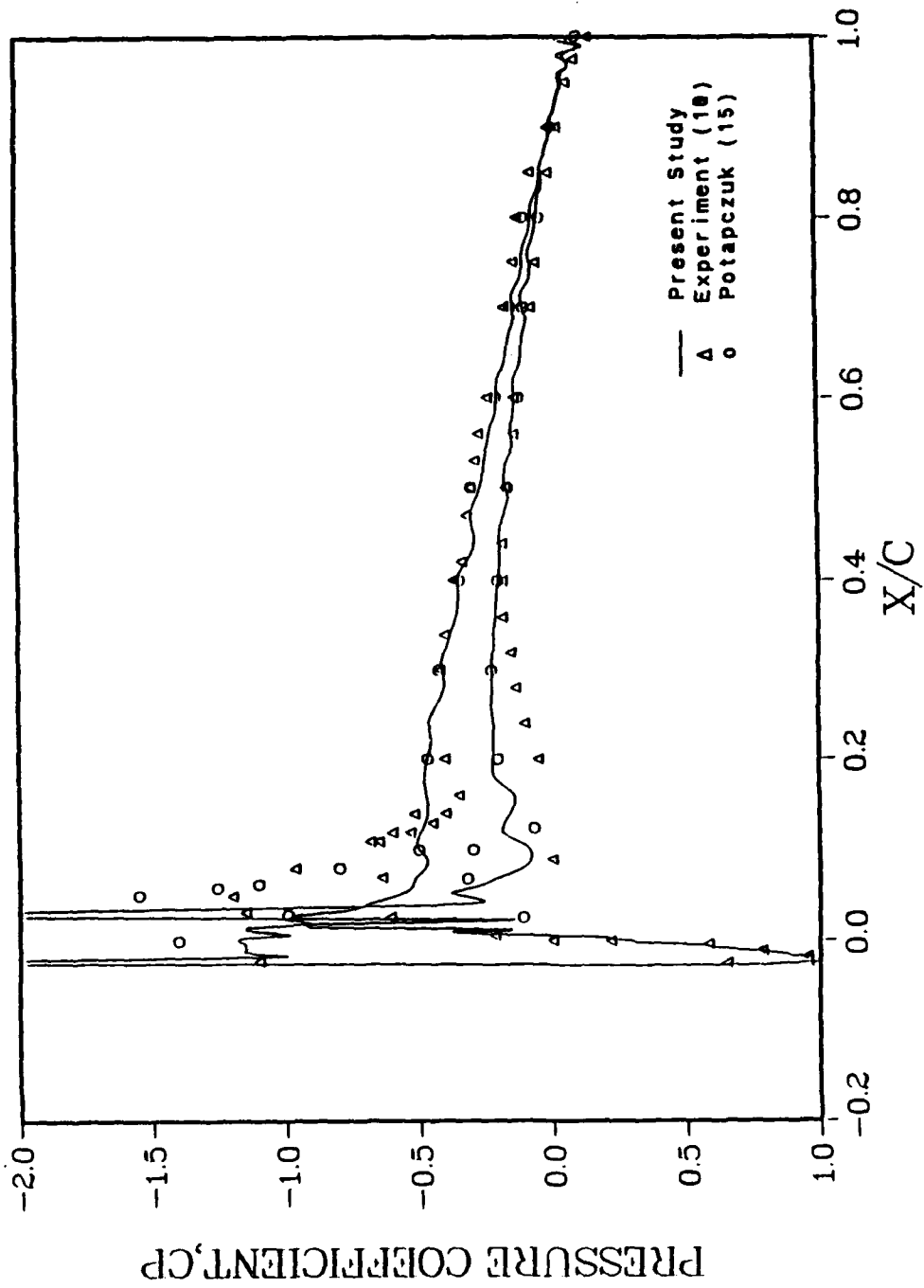


Figure 9. Comparison of Computed and Experimental Pressure Coefficient for 2.0 Degrees Angle of Attack (Experimental Values from 10:41)

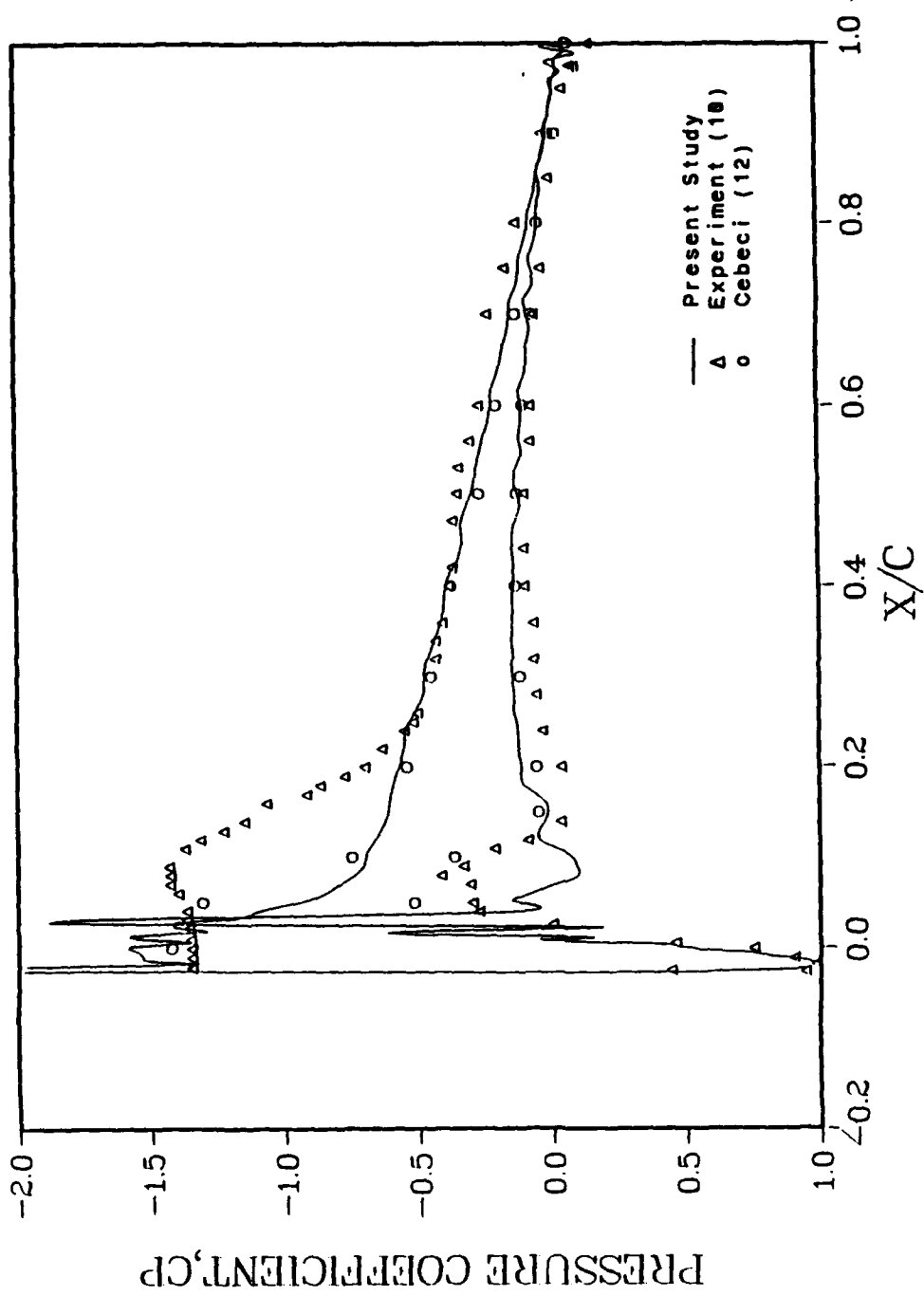


Figure 10. Comparison of Computed and Experimental Pressure Coefficient for 4.0 Degrees Angle of Attack (Experimental Values from 10:42)

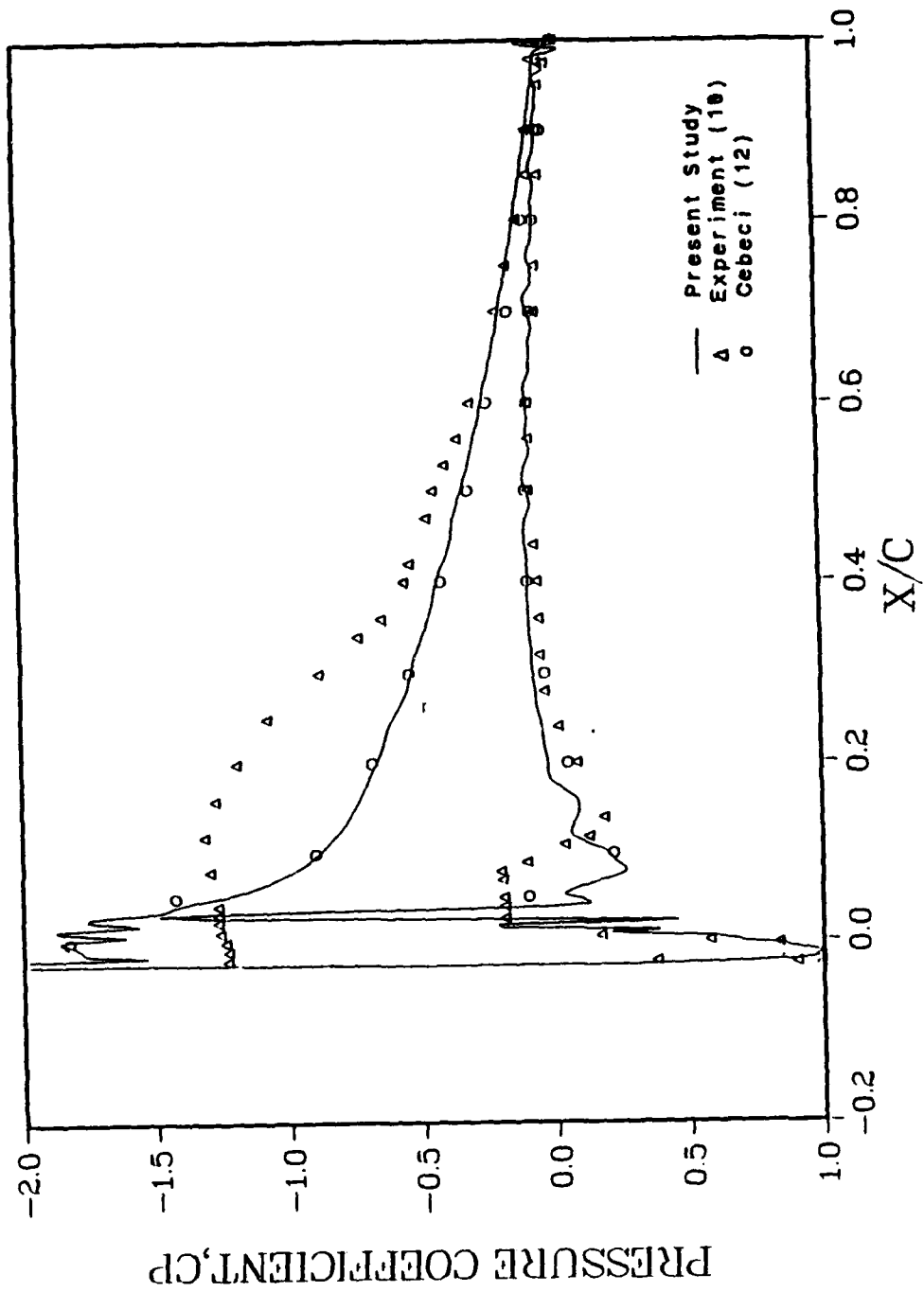


Figure 11. Comparison of Computed and Experimental Pressure Coefficient for 6.0 Degrees Angle of Attack (Experimental Values from 10:43)

contained in the references previously cited for these authors. Care was taken to obtain as much accuracy as possible, but there is certainly some error involved in obtaining data by scaling. Therefore, the intent of the comparisons shown in these plots is simply to reveal general trends between previous work and the current study.

Note that the computed results of the present study show sharp peaks in the pressure coefficient as the flow goes over the ice shape. These peaks are due to the very large gradients in the velocity and pressure in this region. The oscillations in the C_p profiles near the ice shape result as the code attempts to resolve the gradients. The original data of Cebeci and Potapczuk also showed sharp peaks in the pressure coefficient near the ice shape, but these are not seen in Figures 9-11 because too few points are shown. The experimental data do not exhibit the sharp peaks, however. This is because the pressure taps do not sense the instantaneous pressure, but rather some average value in the region.

Examining Figures 9-11 shows that the computed pressure coefficient on the lower surface agrees well with the experimental data for all angles of attack. Similar agreement can be seen for the other numerical results as well. On the upper surface, however, agreement with the experimental data is a function of angle of attack. For $\alpha=2.0$ degrees, the computed pressures agree well except in

the region $x/c=.18-.20$. At this location, there is a change in the slope of the experimental data that the code does not pick up. The numerical results are relatively flat in this region. Potapczuk's results agree more closely with the experimental data in the region just behind the ice horn on the upper surface.

The discrepancy behind the ice shape on the upper surface is worse as the angle of attack increases, as can be seen in Figures 10 and 11. For $\alpha=4.0$ degrees, the experimental and computed results differ substantially over about the first 20% chord. In this region, the computed pressure coefficients are much higher than the experimental results. The high velocity, low pressure peak in the experimental data at approximately $x/c=.15$ is not predicted in the numerical results at all. Note that Cebeci's data does not compare well with the experimental data in this region, either. For $\alpha=6.0$ degrees, the same trend can be seen. Again, Cebeci's data and the results of the present study are essentially the same, with both codes predicting pressure coefficients much higher than the experimental data. However, at this angle of attack the discrepancy in the results extends to about 40% chord.

Local Results. Local flowfield results are included to show the size and location of the separation bubbles. Separation over the ice horns is illustrated in the u-velocity contours plotted in Figures 12-19, corresponding

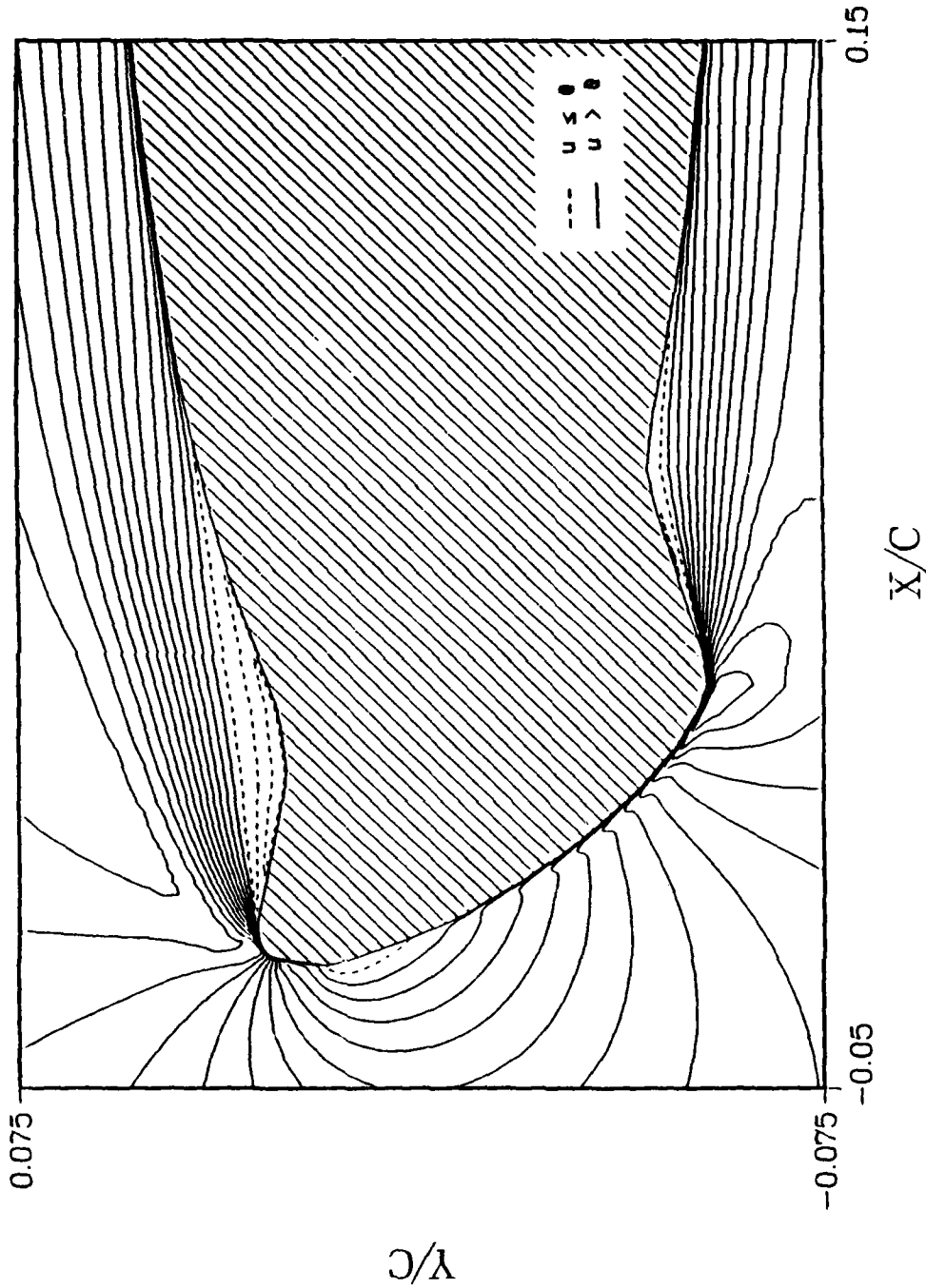


Figure 12. u-Velocity Contours for 2.0 Degrees Angle of Attack- Unlabeled Contours

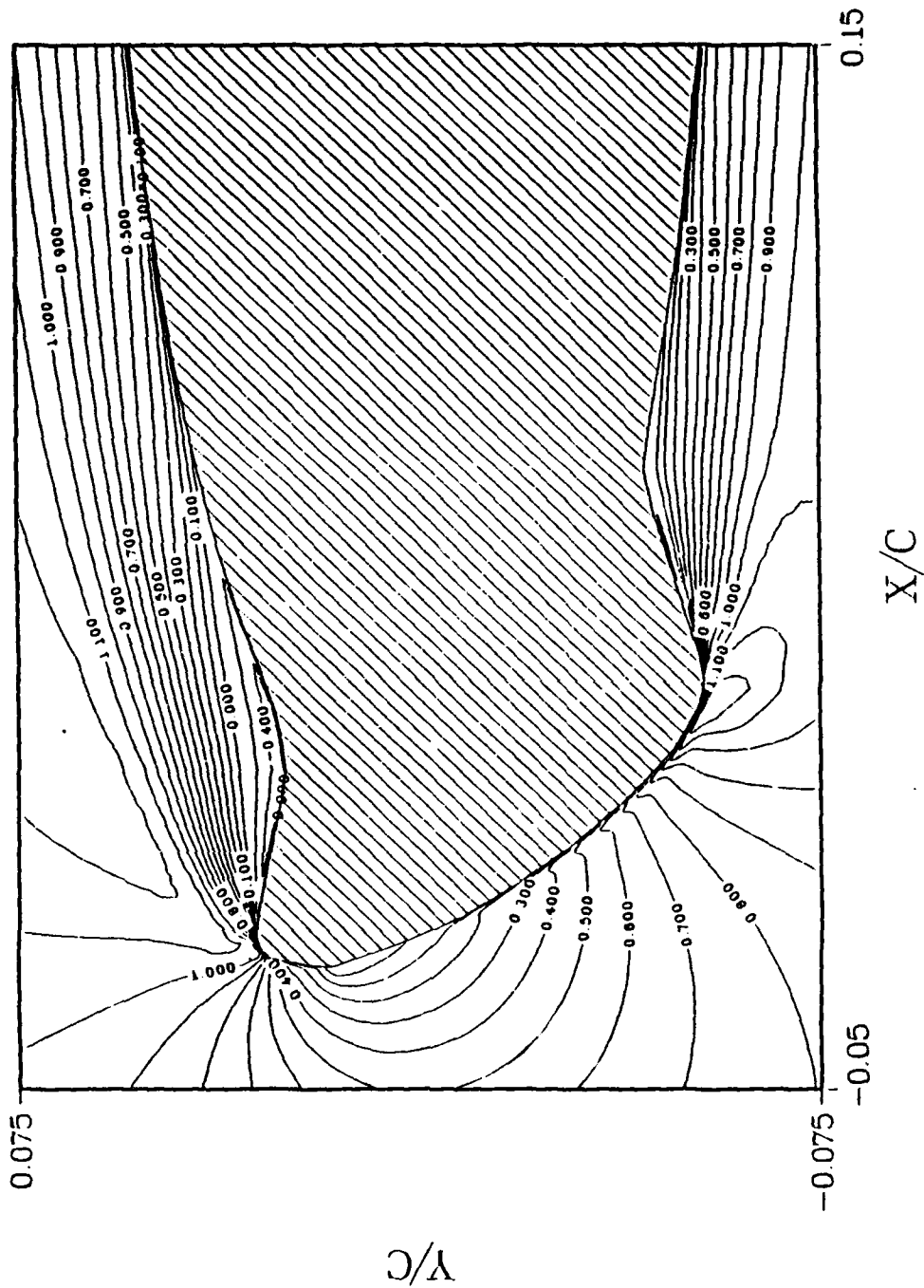


Figure 13. u-Velocity Contours for 2.0 Degrees Angle of Attack-Labeled Contours

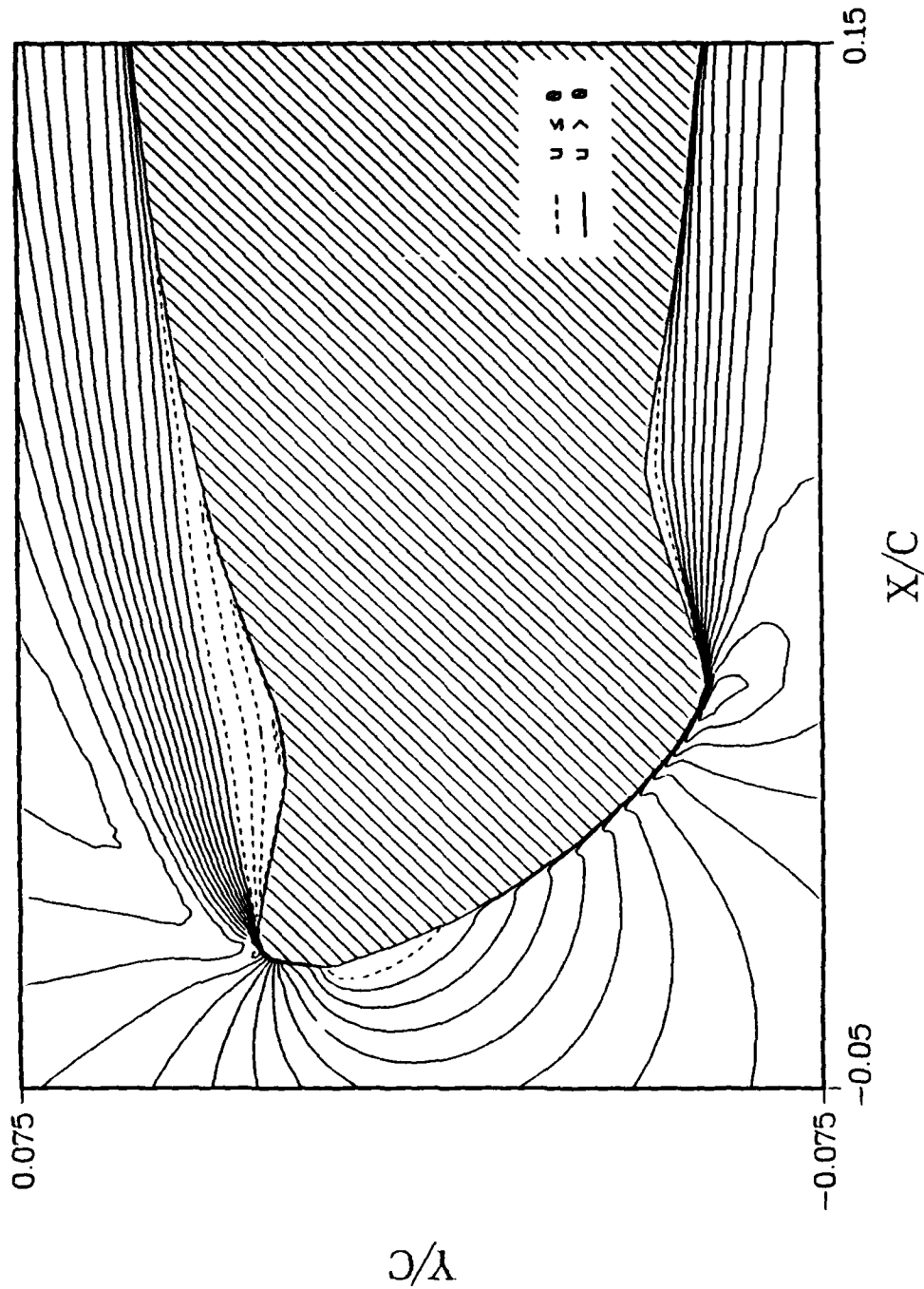


Figure 14. u-Velocity Contours for 4.0 Degrees Angle of Attack-Unlabeled Contours

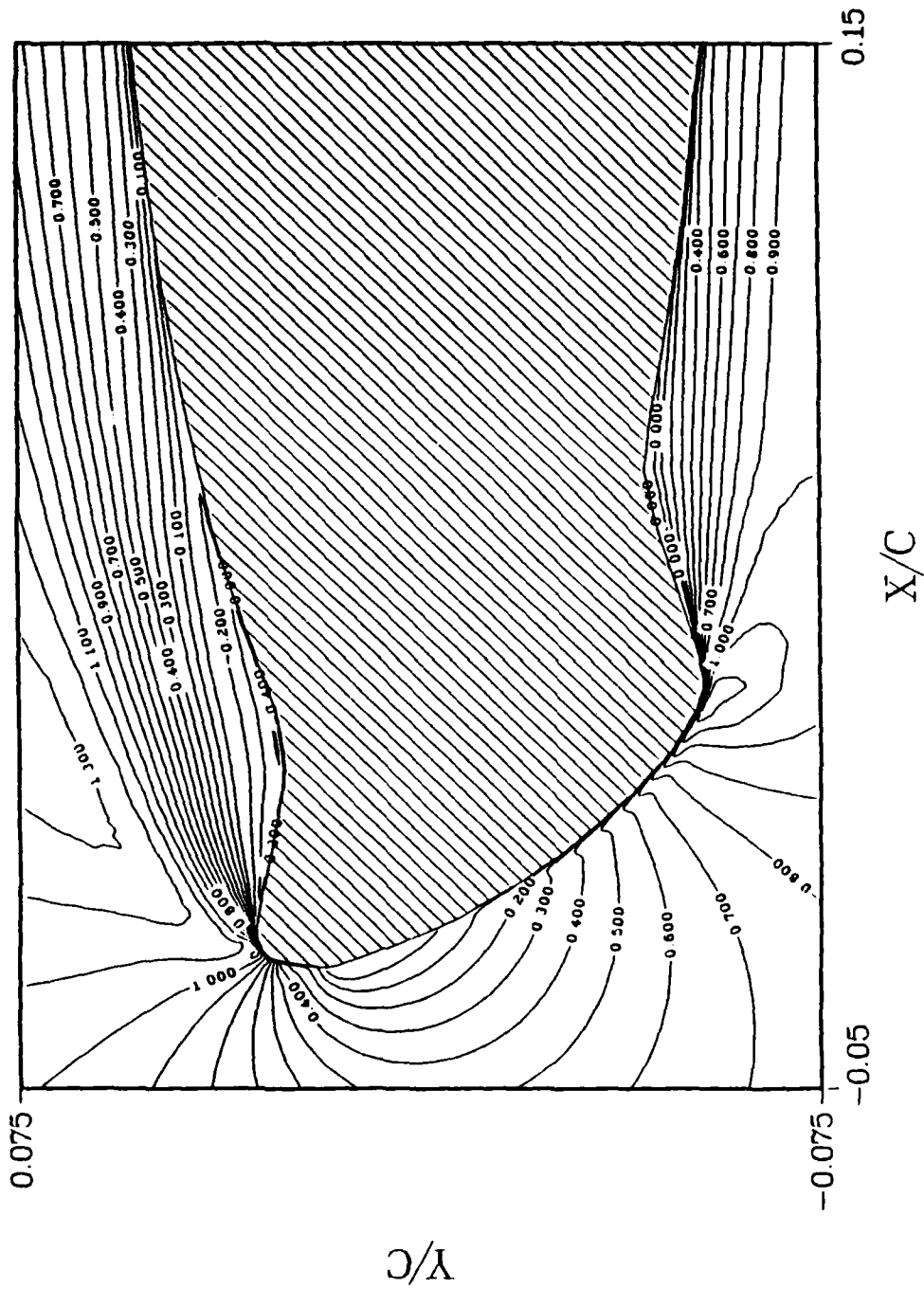


Figure 15. u-Velocity Contours for 4.0 Degrees Angle of Attack-Labeled Contours

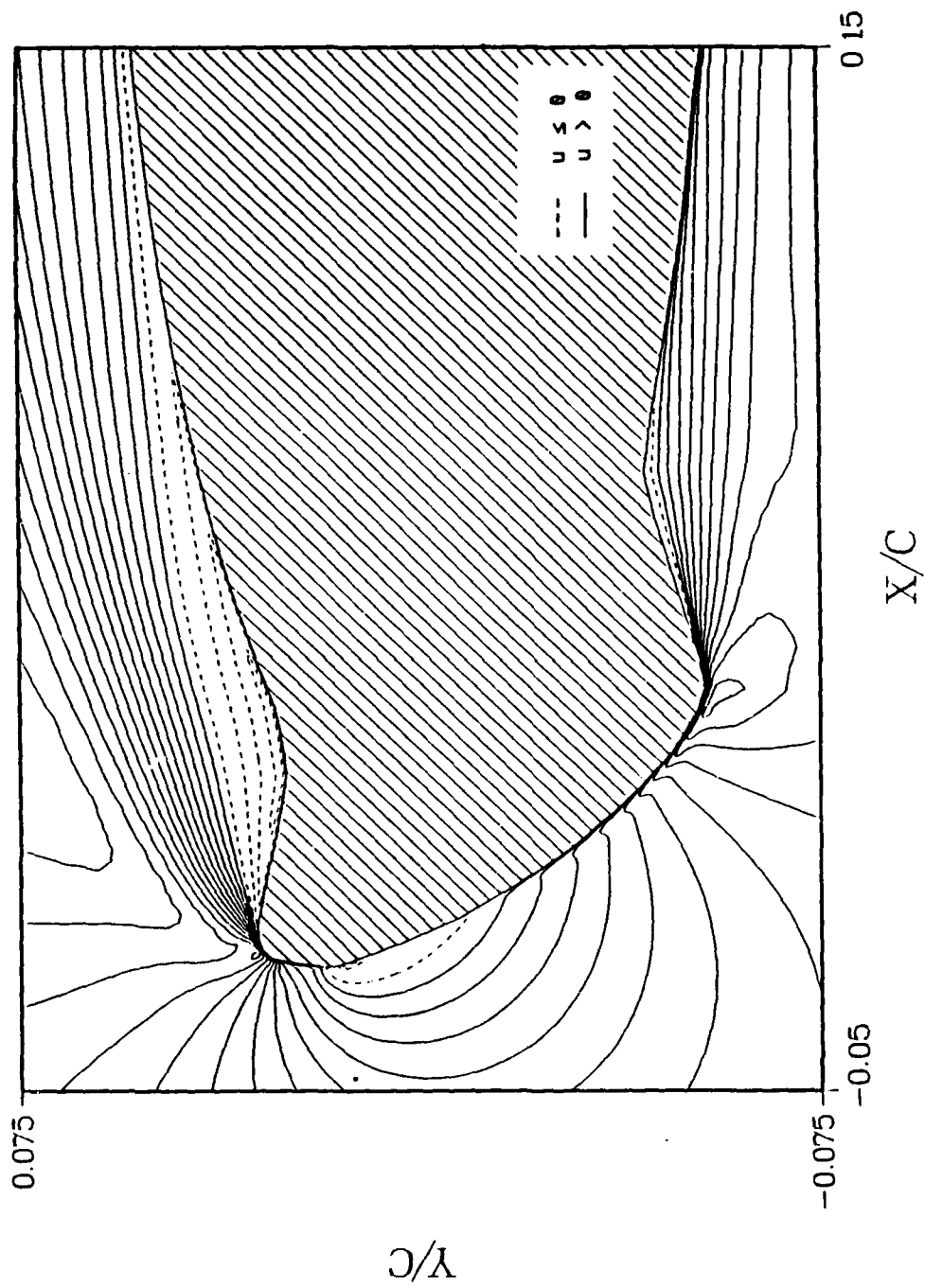


Figure 16. u-Velocity Contours for 6.0 Degrees Angle of Attack-Unlabeled Contours

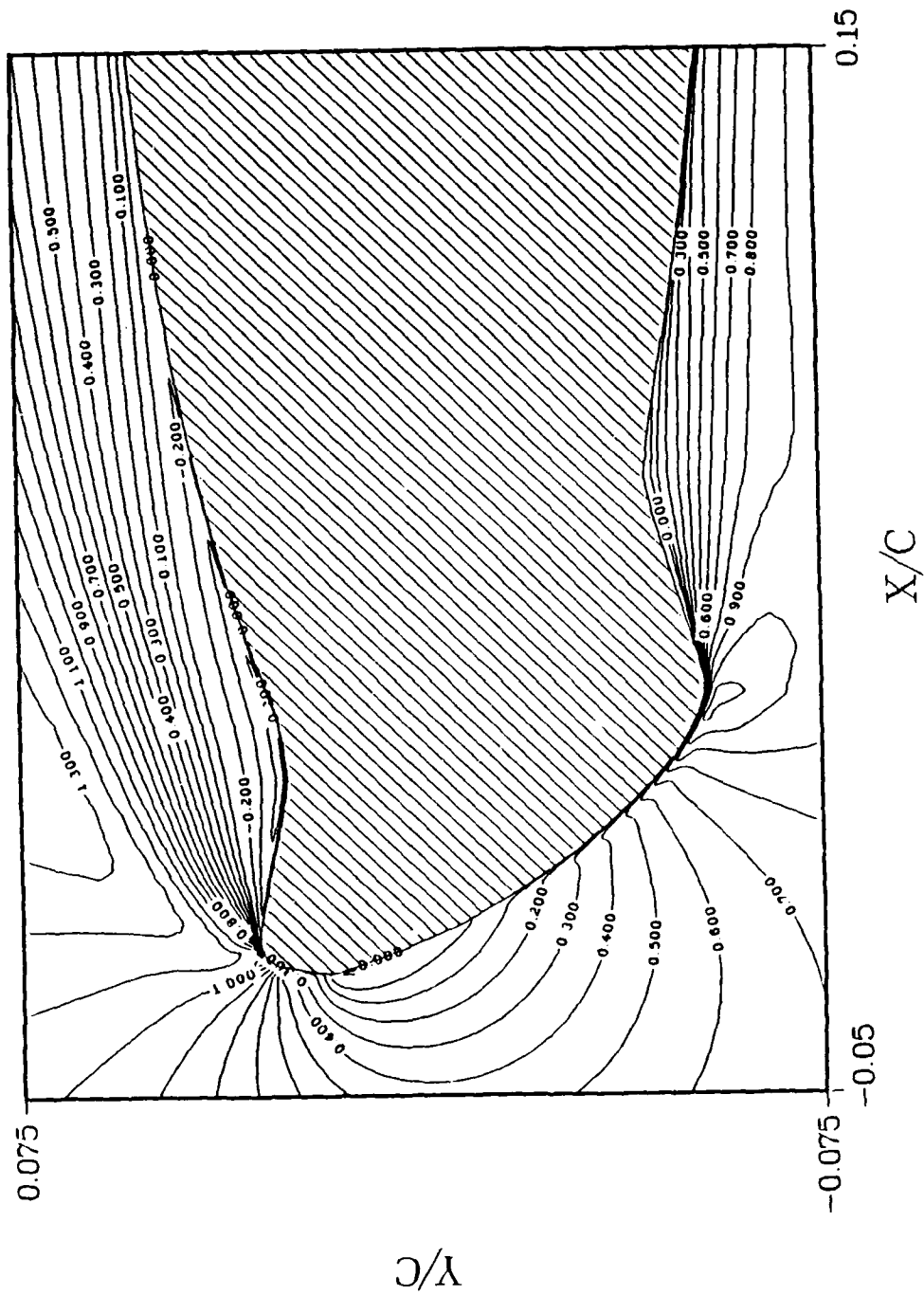


Figure 17. u-Velocity Contours for 6.0 Degrees Angle of Attack-Labeled Contours

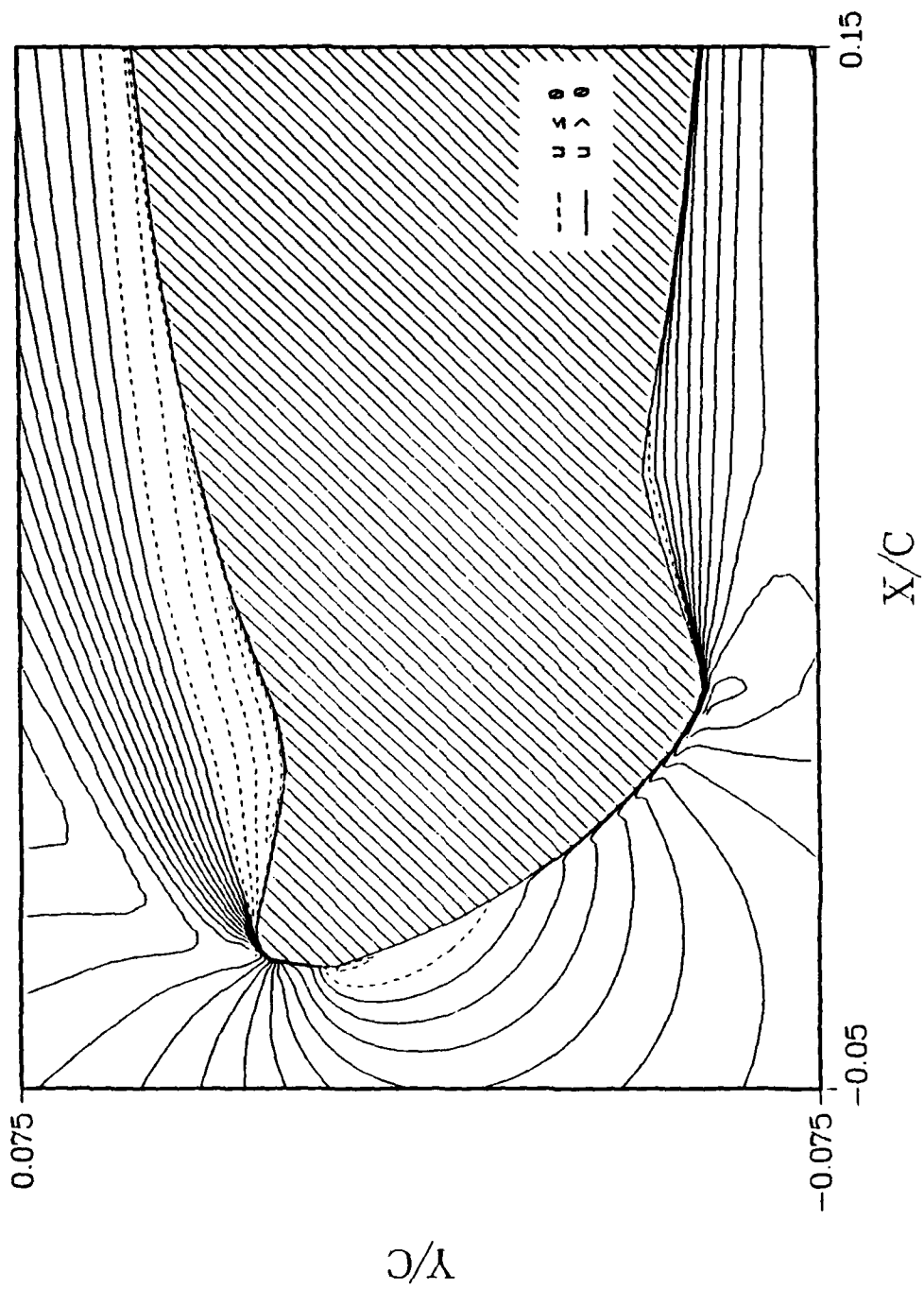


Figure 18. u-Velocity Contours for 8.0 Degrees Angle of Attack-Unlabeled Contours

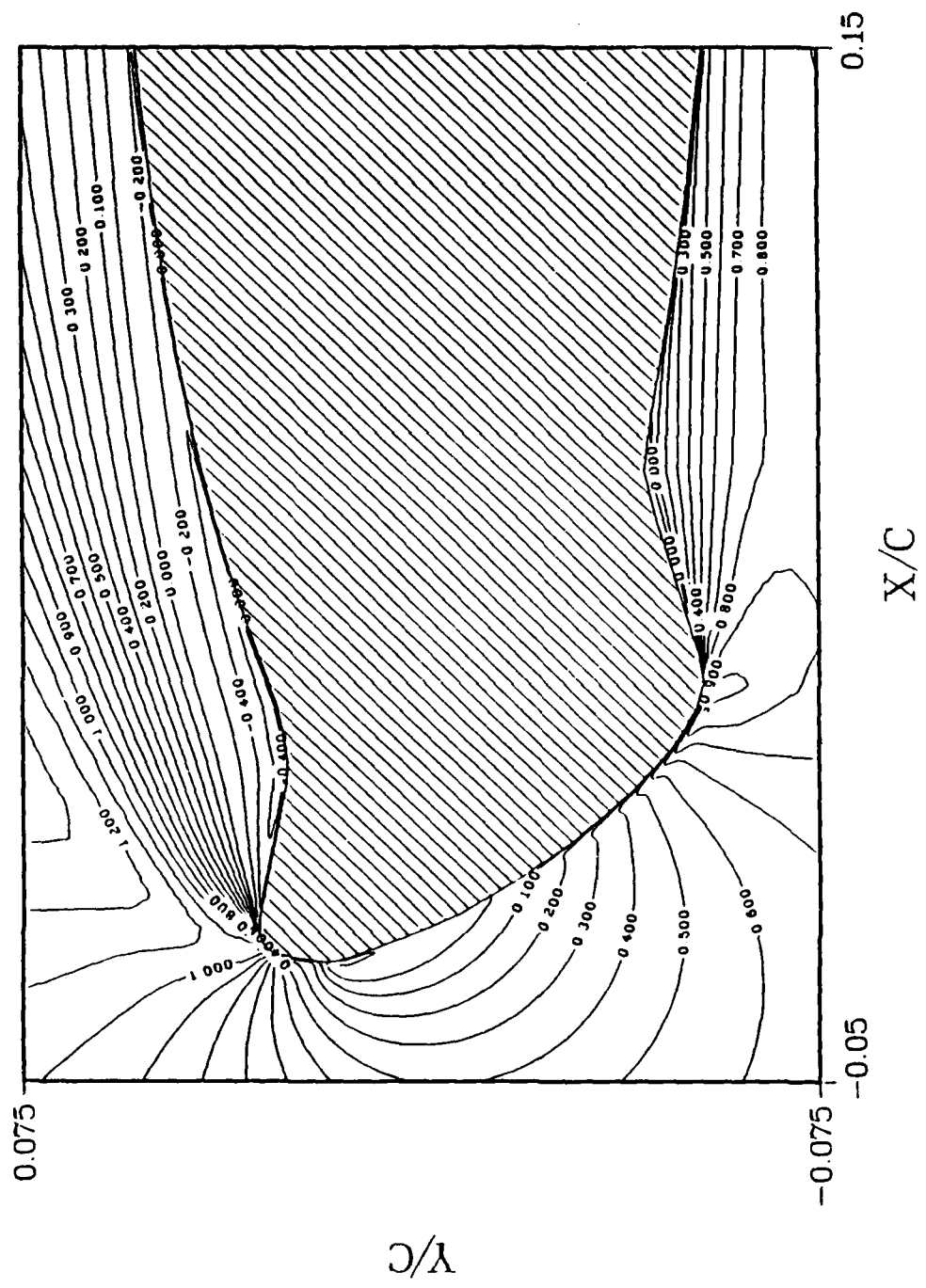


Figure 19. u-Velocity Contours for 8.0 Degrees Angle of Attack-Labeled Contours

to angles of attack of 2.0, 4.0, 6.0, and 8.0 degrees. For each angle of attack, one labeled and one unlabeled contour plot is shown. The growth of the upper surface separation bubble with increasing angle of attack can be seen clearly in these figures. For $\alpha=6.0$ and $\alpha=8.0$ degrees, it can be seen from Figures 16-19 that the flow does not reattach like it did for lesser angles of attack. The u-velocity contours over the entire airfoil for $\alpha=6.0$ degrees are shown in Figure 20. This figure indicates a region of reverse flow near the surface along the entire airfoil upper surface. The data for this angle of attack indicate the flow does reattach at $x/c=.9899$. Figure 21 shows than an even thicker separated flow region extends over the airfoil at an angle of attack of 8.0 degrees. At this angle of attack, the flow never reattaches to the airfoil upper surface.

Another representation of the separation bubbles is given in the vector velocity plots of Figures 22-27. These figures clearly indicate the separated flow regions caused by the ice shape, as well as the reverse flow within the bubbles. Again, for $\alpha=6.0$ and $\alpha=8.0$ degrees, plots of the entire airfoil are included to show the separated flow over the upper surface.

The separation and reattachment points for the numerical results are shown in Table 1 for each angle of attack. In all cases, the flow separates at the same

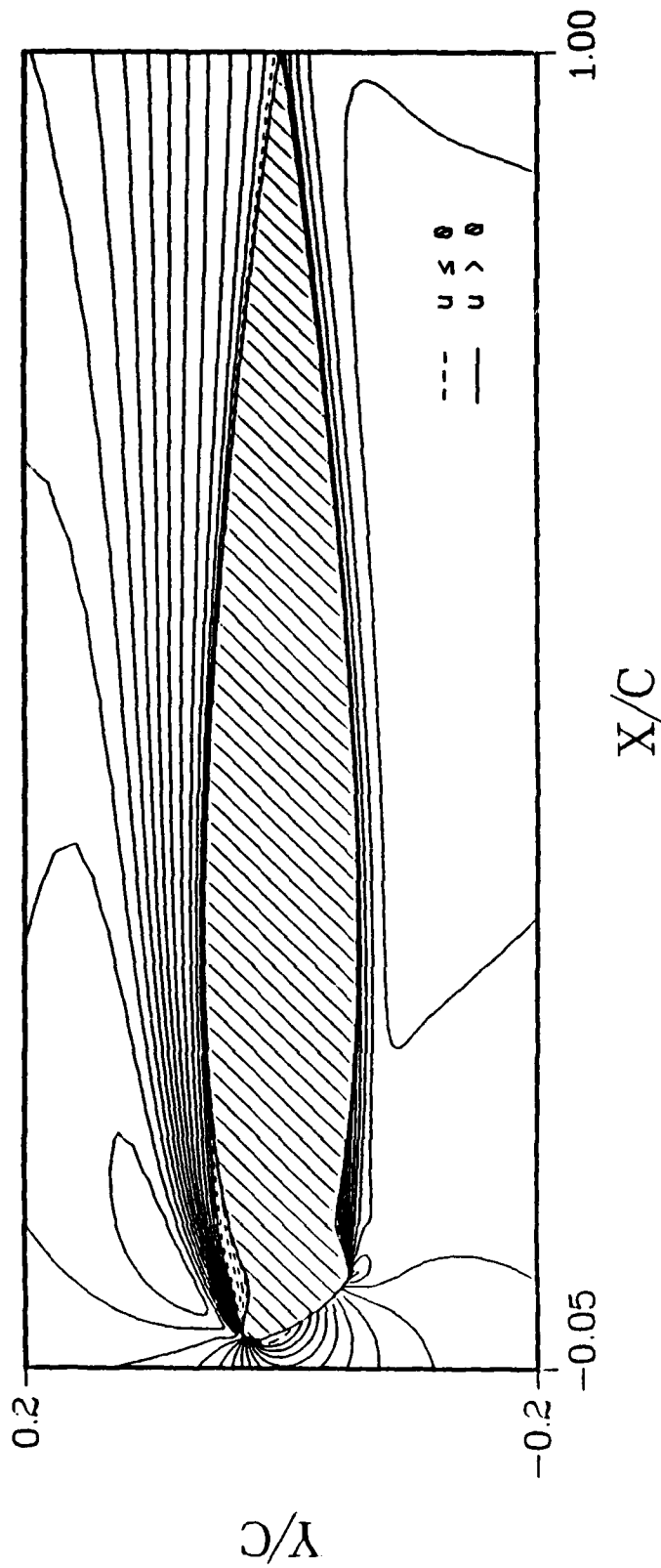


Figure 20. u-Velocity Contours over Entire Airfoil for 6.0 Degrees Angle of Attack

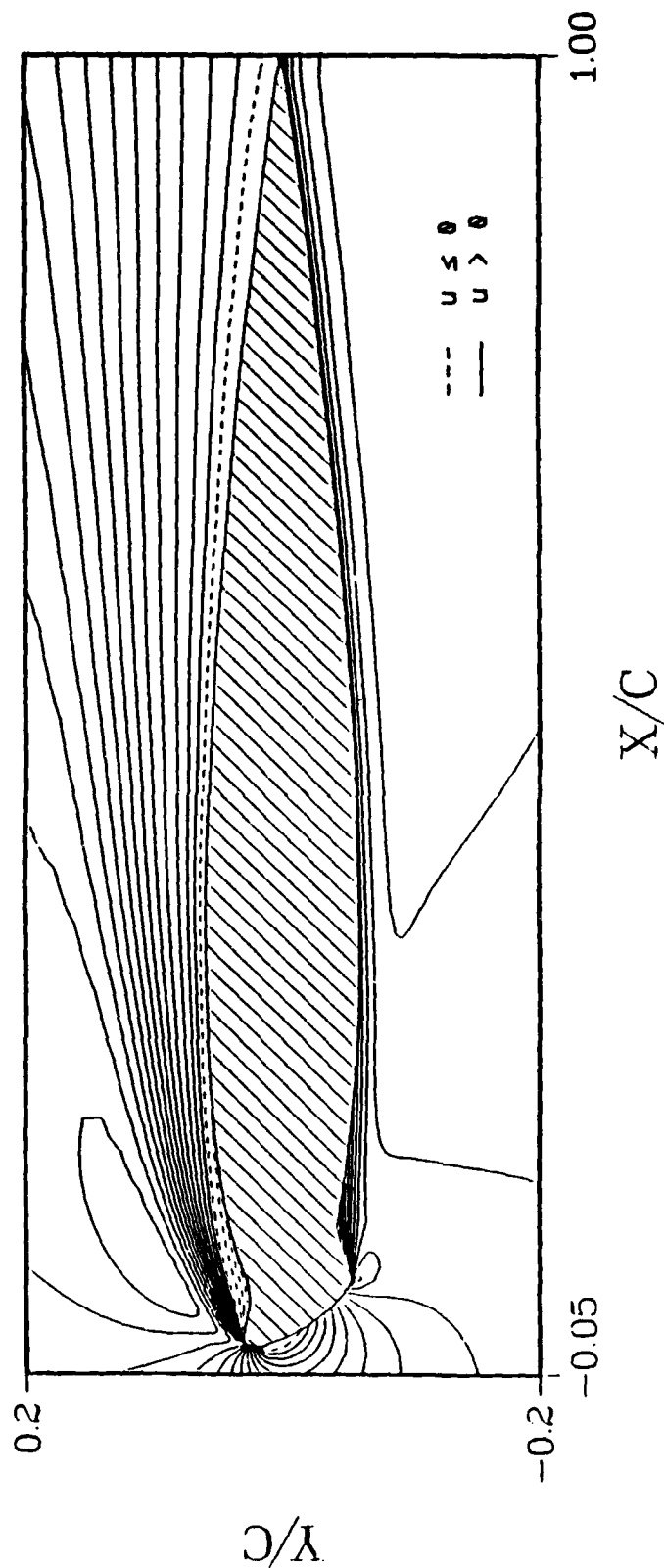


Figure 21. u -Velocity Contours over Entire Airfoil for 8.0 Degrees Angle of Attack

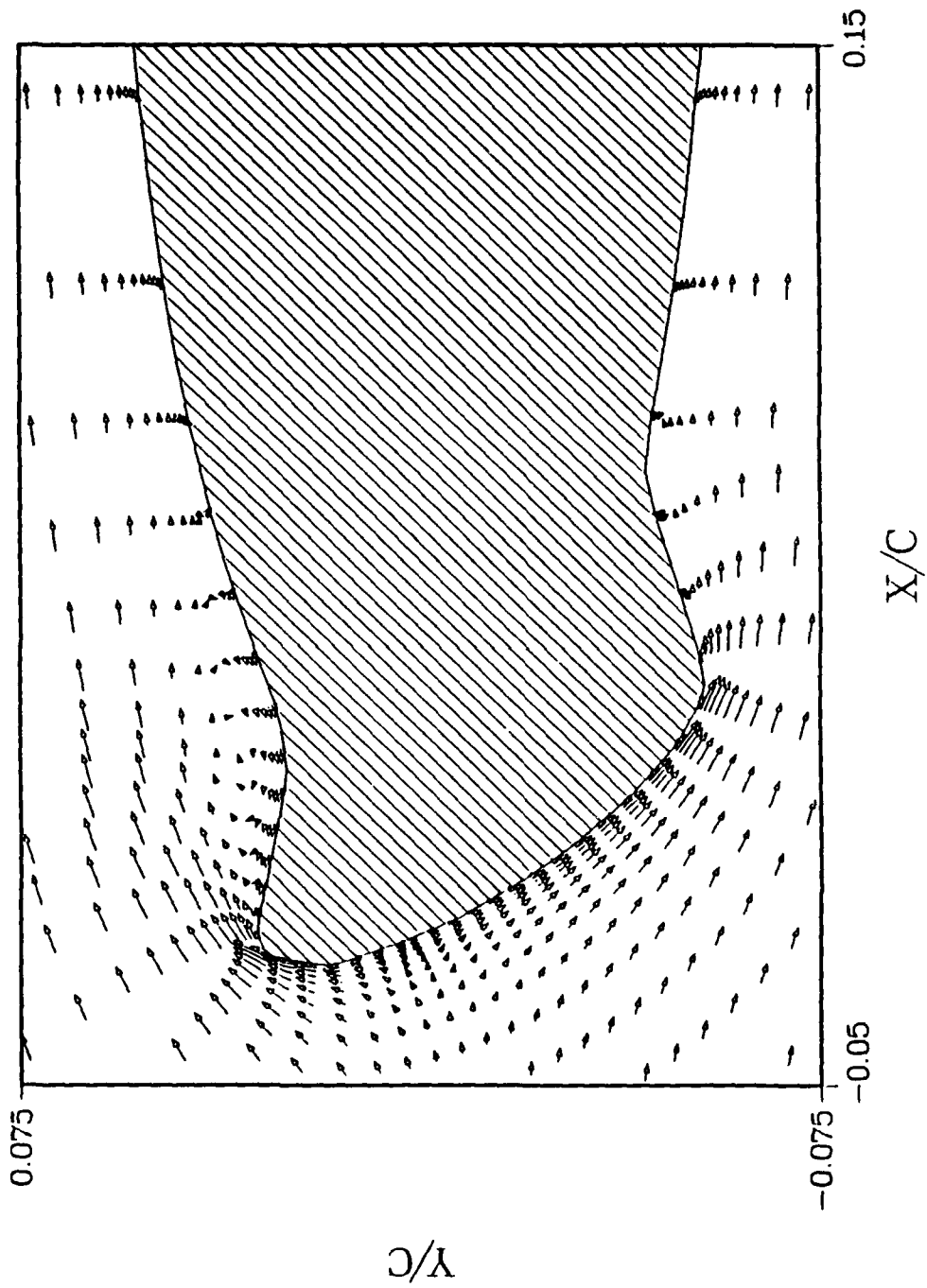


Figure 22. Velocity Vector Plot for 2.0 Degrees Angle of Attack

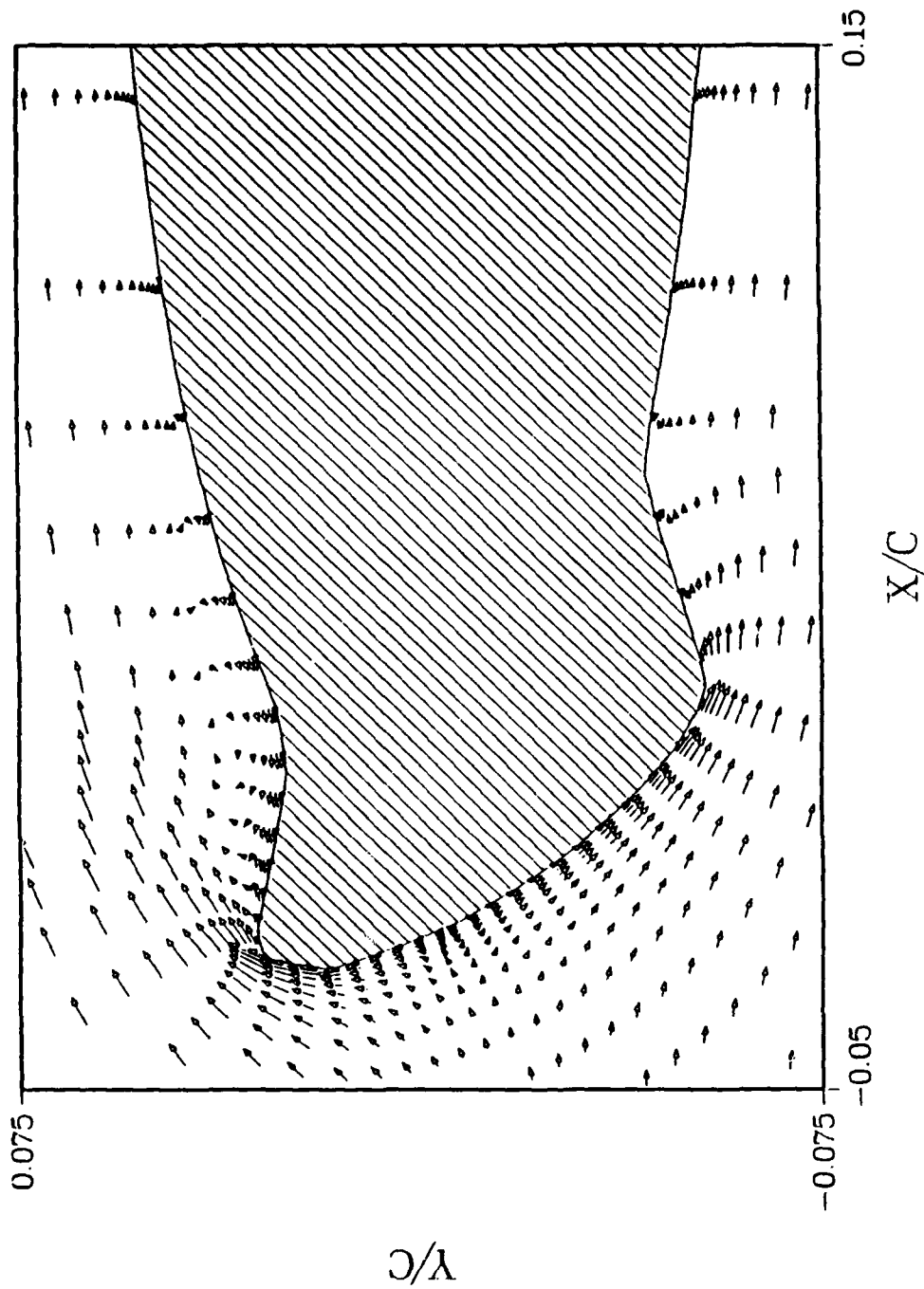


Figure 23. Velocity Vector Plot for 4.0 Degrees Angle of Attack

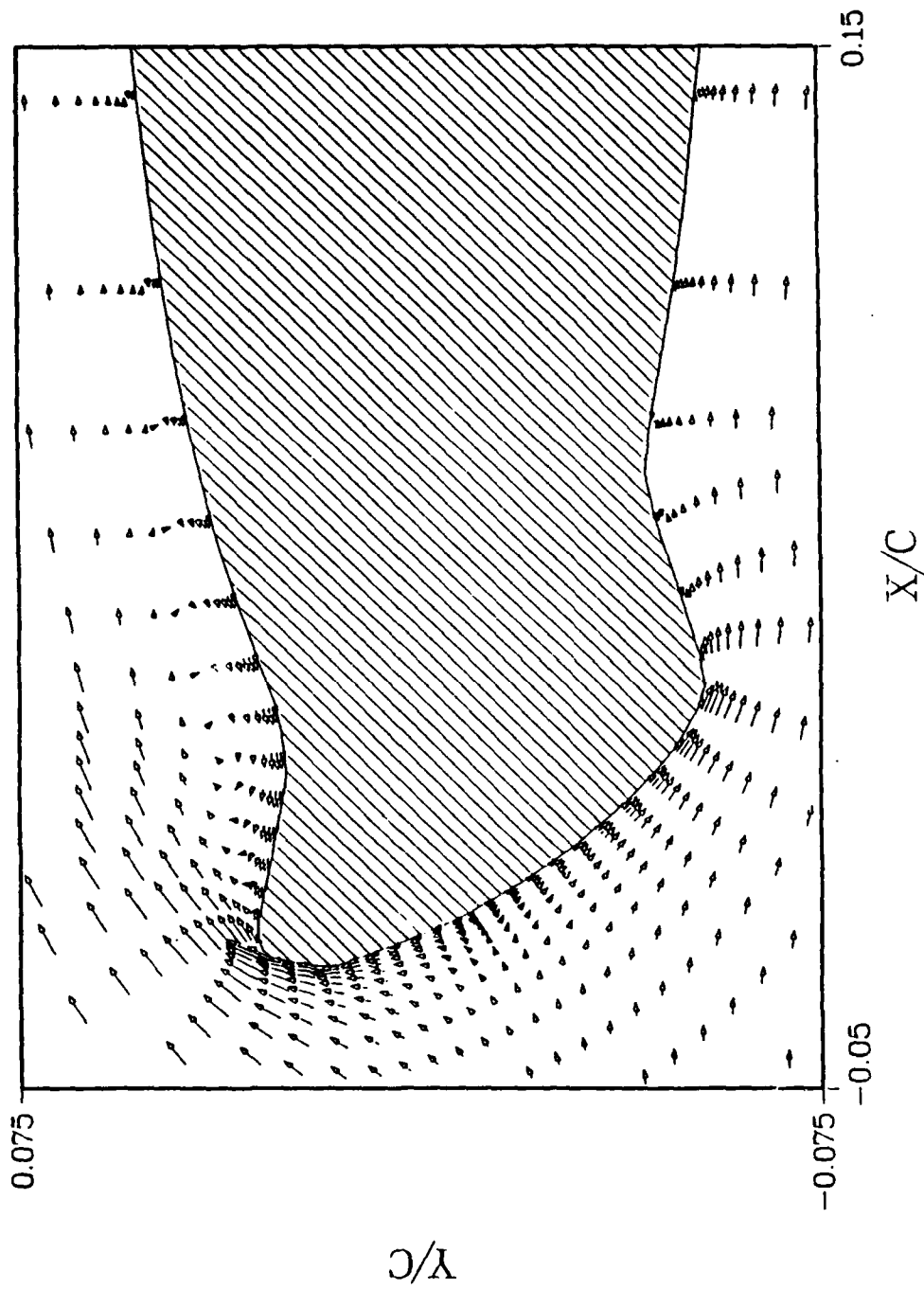


Figure 24. Velocity Vector Plot for 6.0 Degrees Angle of Attack

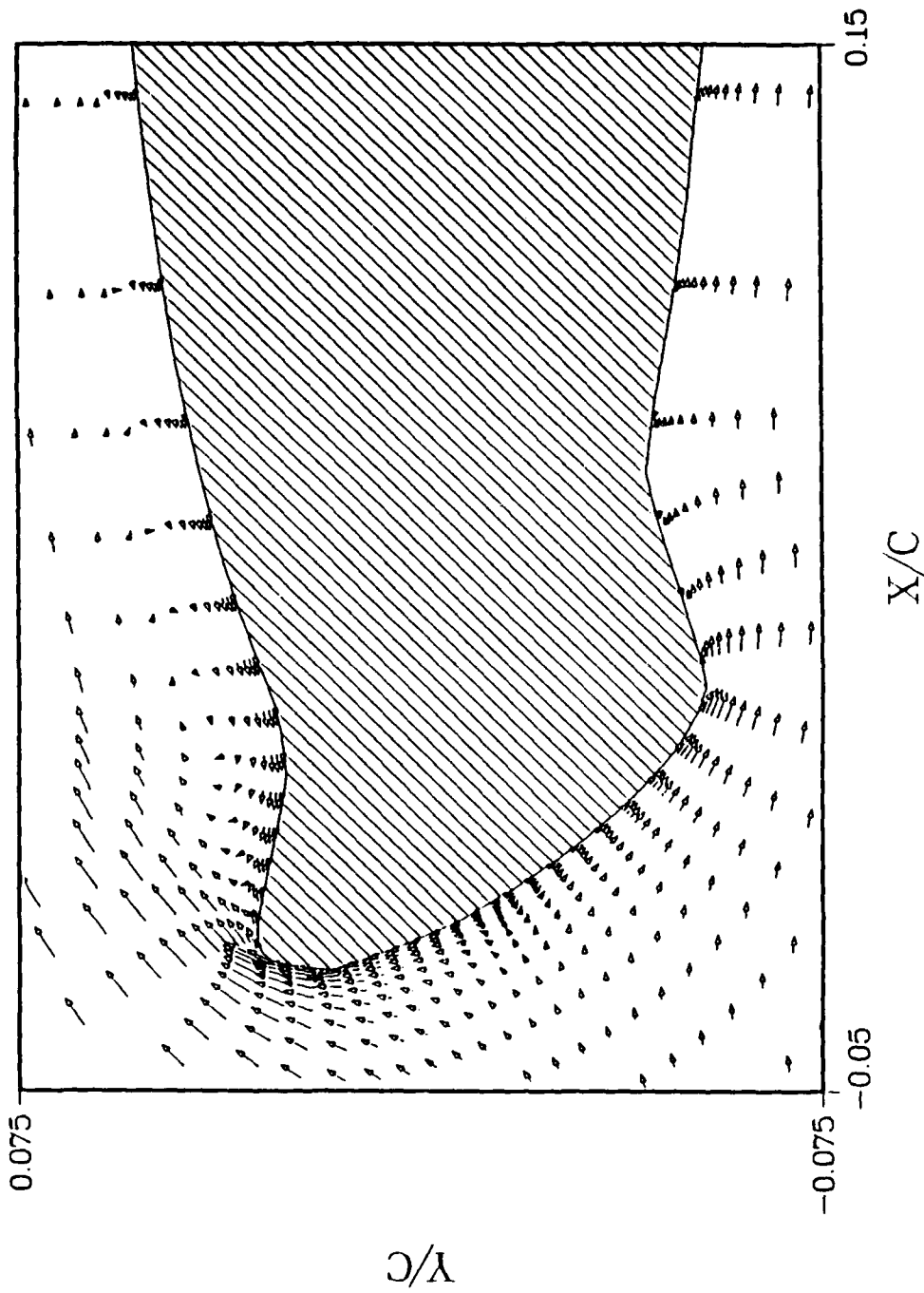


Figure 25. Velocity Vector Plot for 8.0 Degrees Angle of Attack

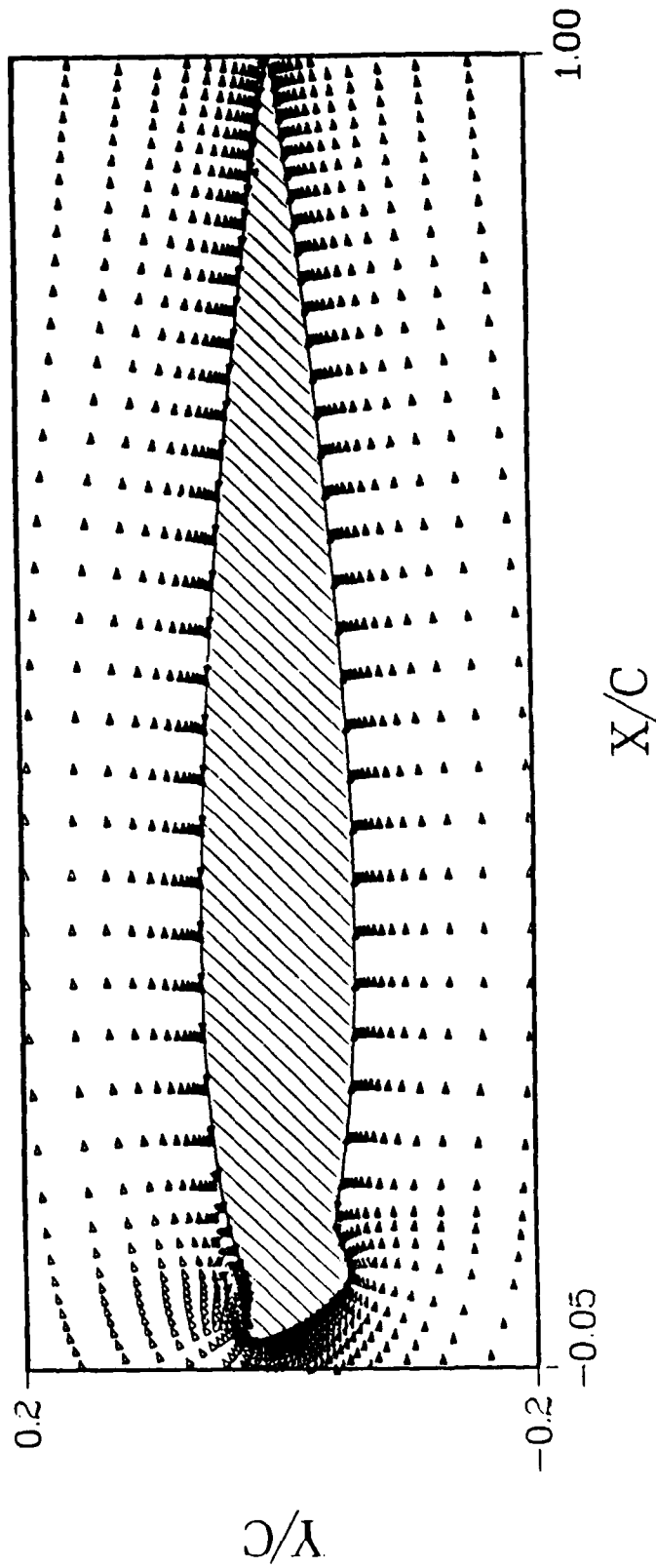


Figure 26. Velocity Vector Plot over Entire Airfoil for 8.0 Degrees Angle of Attack

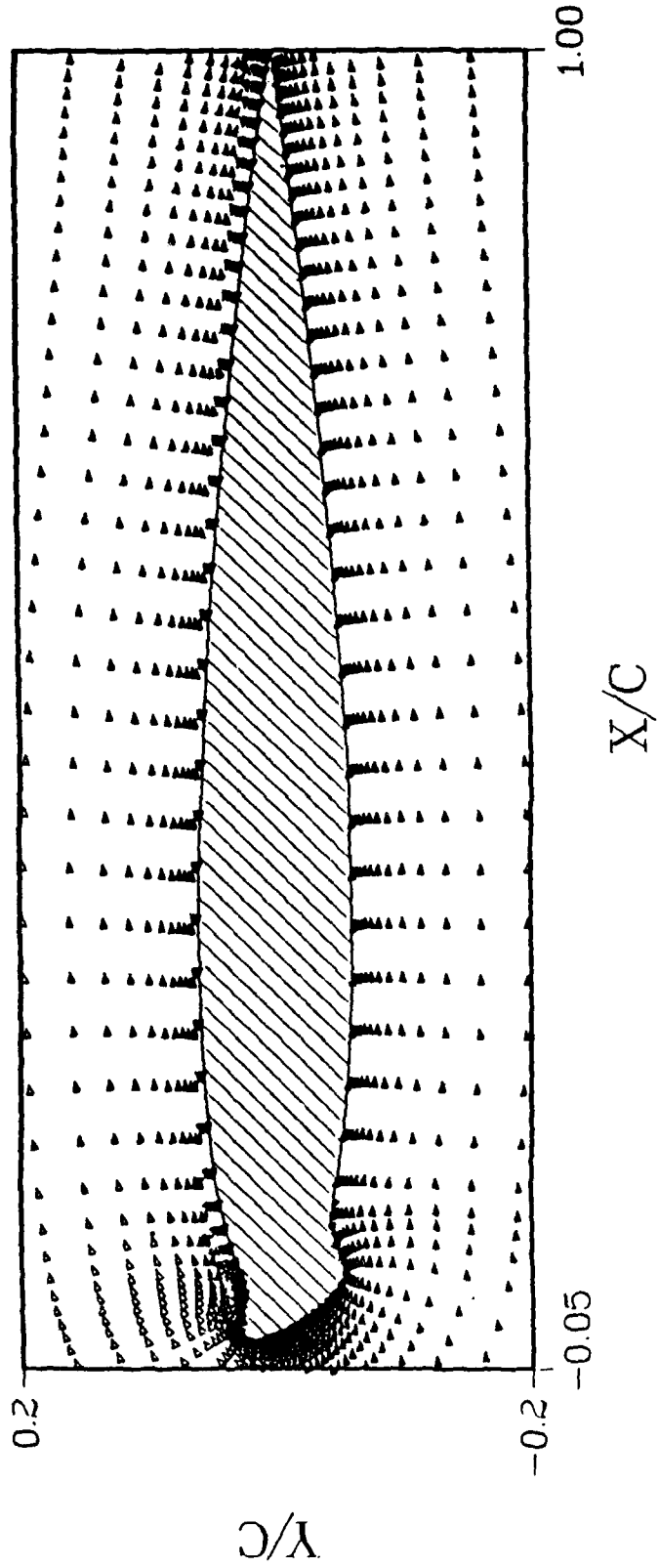


Figure 27. Velocity Vector Plot over Entire Airfoil for 8.0 Degrees Angle of Attack

Table 1. Comparison of Computed and Experimental Reattachment Locations

Angle of Attack (Deg)	Lower Surface		Upper Surface	
	Separation Computed (x/c)	Reattachment Computed/Experimental (x/c)	Separation Computed (x/c)	Reattachment Computed/Experimental (x/c)
2	.0385	.0902	-.01898	.0684
4	.0385	.130	-.01898	.1040
6	.0385	.0785	-.01898	.9899
8	.0385	None	-.01898	None

location on the ice horns, while reattachment location is a function of angle of attack. As indicated previously, the flow over the upper surface does not reattach at all for $\alpha=8.0$ degrees. Table 1 shows that the flow is essentially separated over the entire airfoil upper surface for 6.0 degrees angle of attack as well.

Comparison of reattachment locations with the experimental data for $\alpha=2.0$ and $\alpha=4.0$ degrees is shown in Figure 28. It can be seen from this figure that the computed results exhibit the correct trend for flow reattachment. On the upper surface, the reattachment point shifts rearward as the angle of attack is increased, while on the lower surface reattachment shifts forward as angle of attack increases. Beyond this general trend, however, comparison with the experimental data is poor.

The code predicts substantially smaller separation bubbles on both the upper and lower surfaces. On the upper surface, the computed reattachment location is 24% less than the experimental value for $\alpha=2.0$ degrees, and 31% less at $\alpha=4.0$ degrees. On the lower surface, the computed result is approximately 40% less than the experimental value for both angles of attack. If the experimental uncertainty is considered, the comparison is a little better but the differences are still substantial.

The computed separation bubbles are not only shorter than the experimental ones, but they are much thinner also.

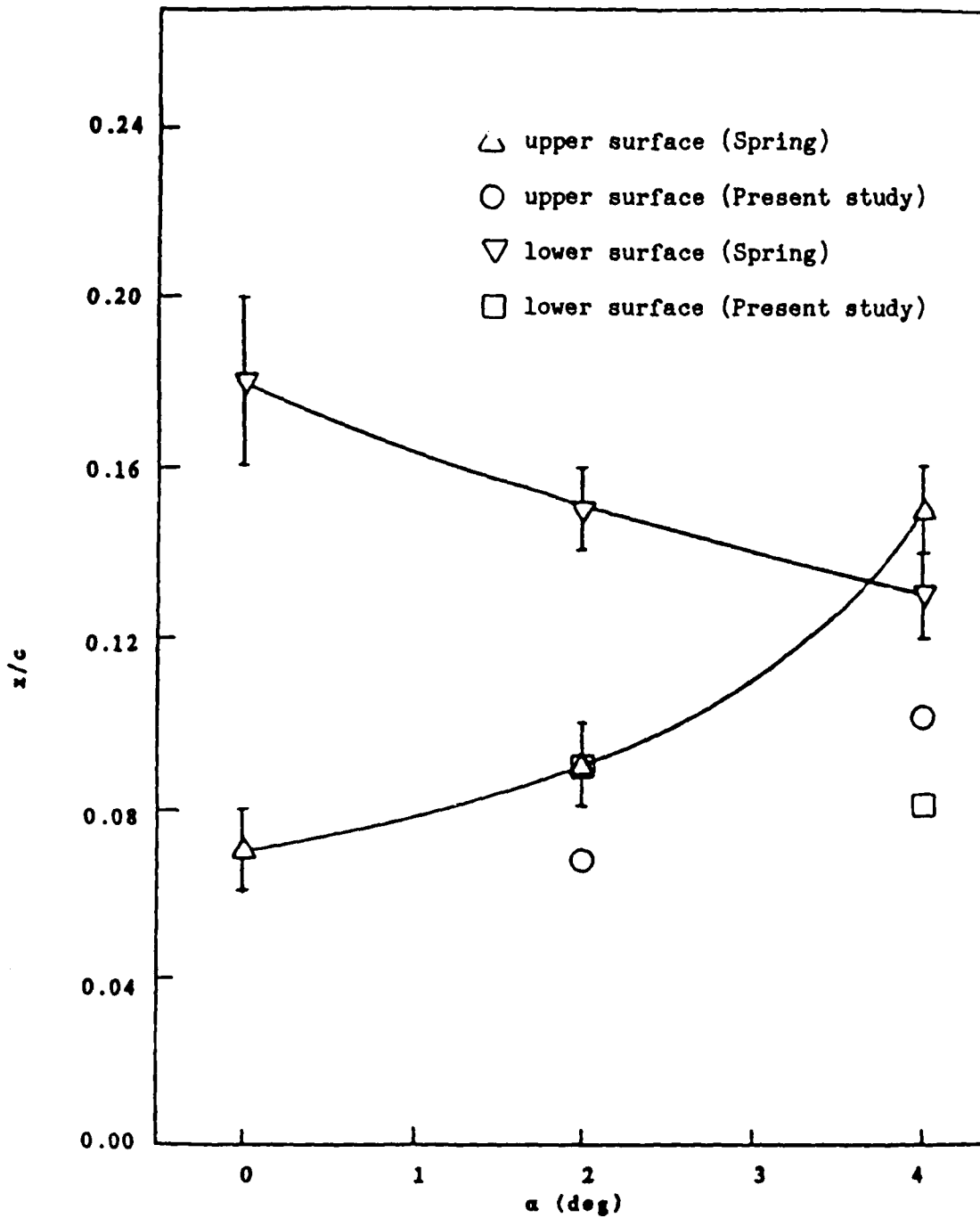


Figure 28. Comparison of Computed Reattachment Locations with Experiment (Experimental Values from 10:52)

This can be seen from the velocity profiles plotted in Figure 29. This plot compares the computed velocity profile on the upper surface at $x/c=0.0$ to Spring's experimental data for an angle of attack of 4.0 degrees. The two velocity profiles are considerably different. The experimental data show reverse flow in the bubble from the surface up to approximately $y=.5$ inches. Computed results indicate reverse flow only up to approximately $y=.15$ inches. In general, the computed bubble is roughly half the thickness of that measured experimentally.

This mismatch between computed and experimental velocity profiles at a location near the ice shape is not unique to the present study. Cebeci has also noted discrepancies between the experimental velocity profiles and those computed with his interactive boundary layer technique for locations within the separation bubble (29).

Results for a Clean NACA 0012 Using Halim's Formulation

The code developed by Halim was extended to the full domain during the course of this study, and the original intention was to perform the other modifications required to apply this code to an iced airfoil. The modifications needed could not be completed within the time allowed, however, so the code was never applied to an iced airfoil. Nonetheless, a test case for a clean NACA 0012 was run with this code to check out the modifications which were made to

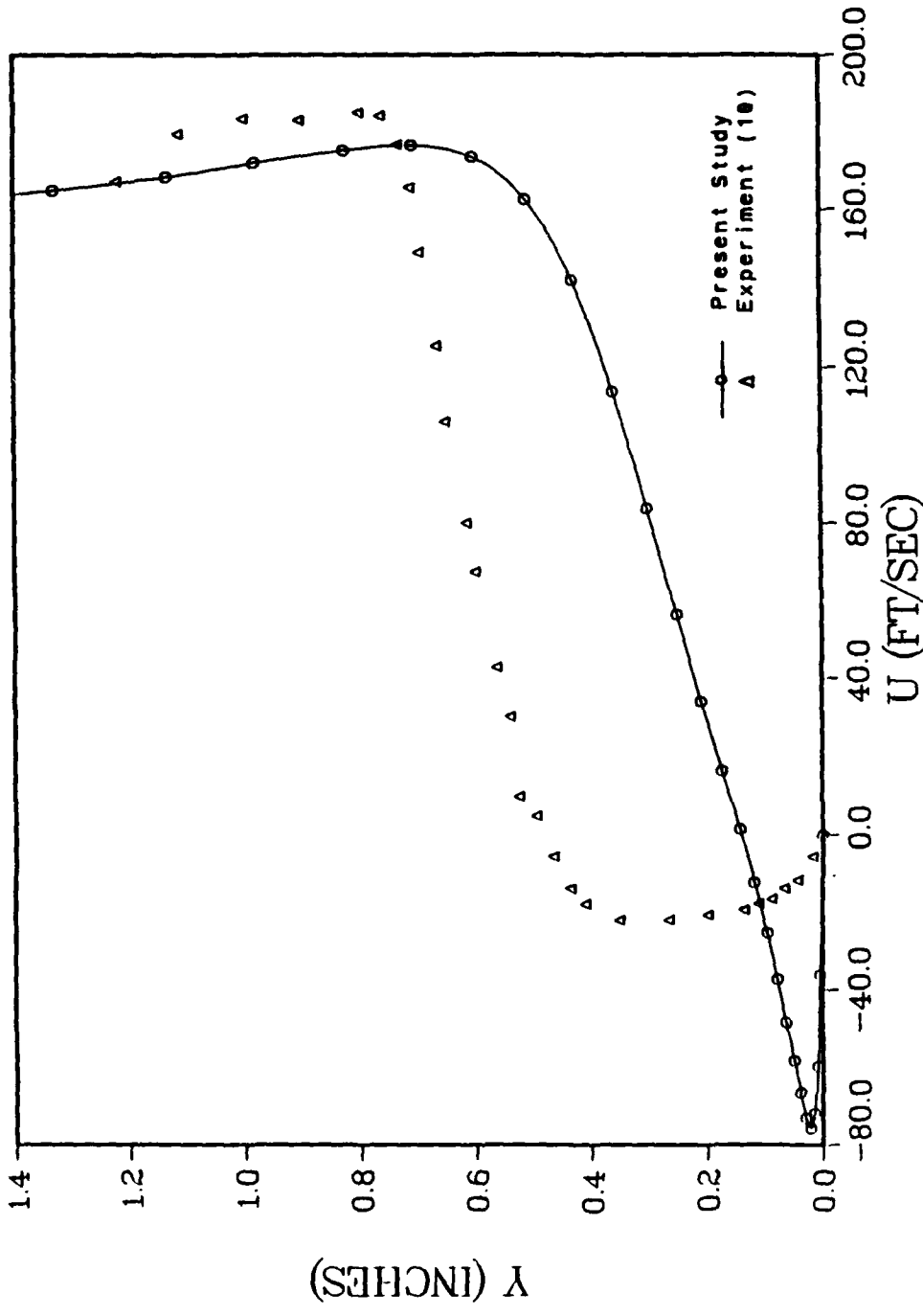


Figure 29. Velocity Profile Comparison at $x/c=0.0$ on the Upper Surface for 4.0 Degrees Angle of Attack (Experimental Values from 10:39)

extend the code to the full domain. For completeness, the results of this test case are presented here.

The code was run for a NACA 0012 airfoil at zero angle of attack, with a Reynolds number of 12,500. The salient feature of this flow is the separation that occurs near the trailing edge. In Halim's previous study with this airfoil, separation was predicted at $x/c=.8178$. For the grid used in the present study, the predicted separation point was $x/c=.9017$. Streamline contours over the airfoil showing the separated flow regions can be seen in Figure 30. In Figure 31, the skin friction is plotted versus x/c . Halim's (19) results for this airfoil are shown for comparison. From this figure it can be seen that the skin friction goes to zero near $x/c=0.9$.

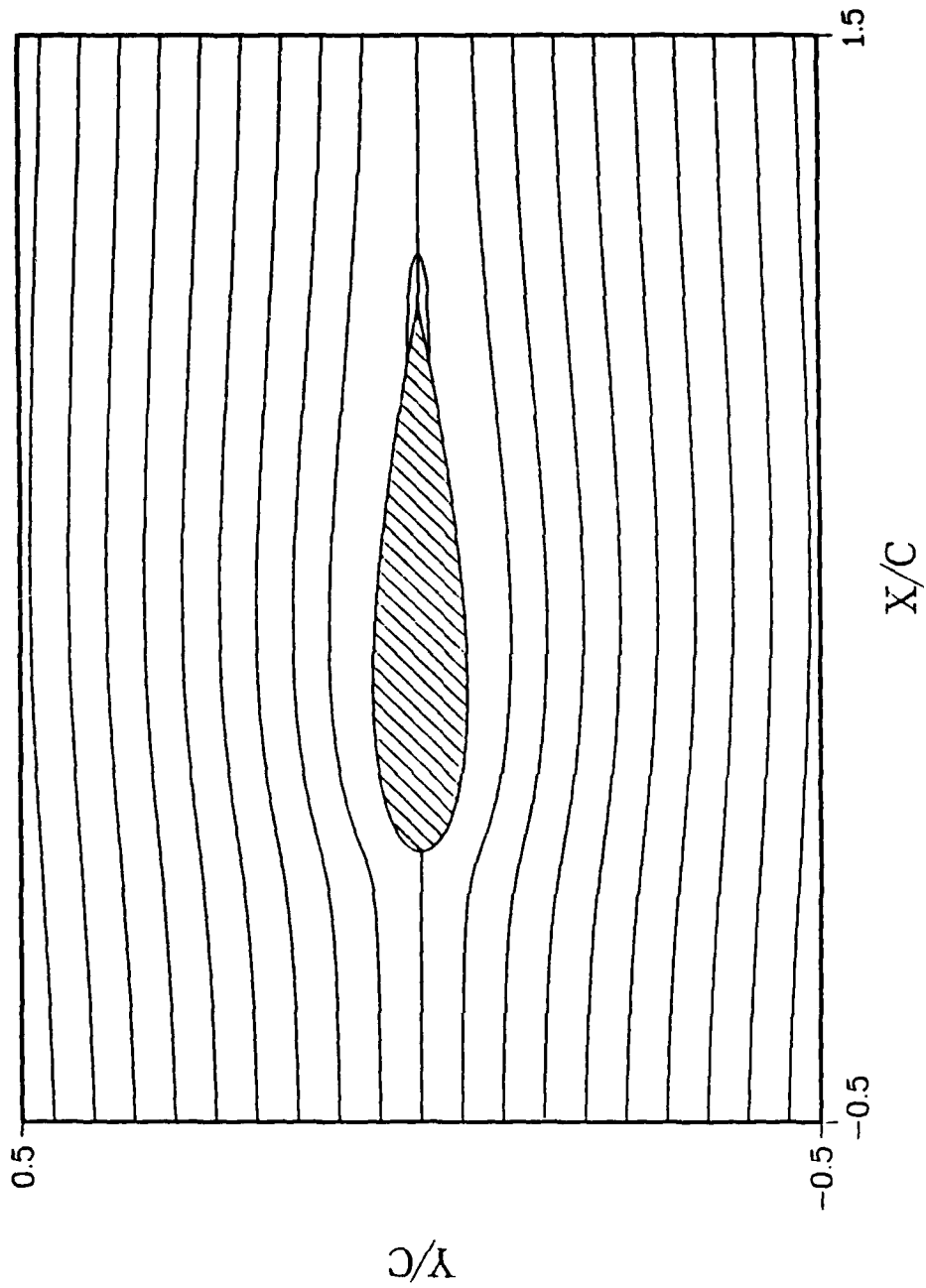


Figure 30. Streamline Contours over a Clean NACA 0012 Airfoil at 0.0 Degrees Angle of Attack

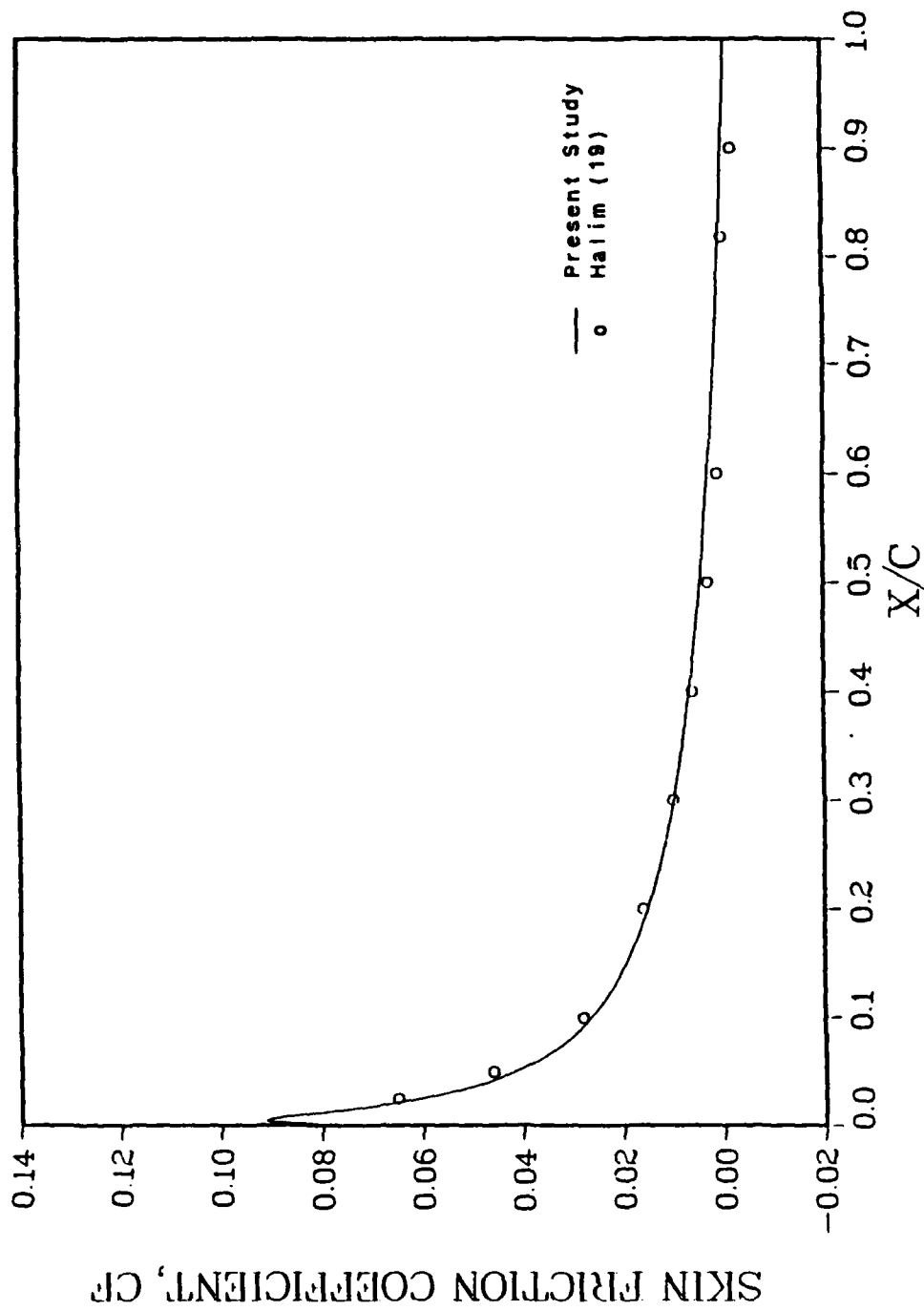


Figure 31. Skin Friction Distribution over a Clean NACA 0012 Airfoil at 0.0 Degrees Angle of Attack

V. Conclusions and Recommendations

In the introduction to this thesis, it was pointed out that one of the primary reasons for applying computational fluid dynamics codes to the iced airfoil problem is to develop and validate a code that can be used as a design tool. Such a code could then be used to predict the sensitivity of proposed airfoil designs to surface roughness and icing. It is evident from the results presented here, however, that much more research will be needed before an accurate computational design tool can be fully developed. Clearly, this goal has not been achieved in the present study.

What has been shown in this work is that the Beam-Warming code, as implemented, produces very good estimates of the airfoil lift and drag for angles of attack below stall. For flow at a free stream Mach number of 0.12 and a Reynolds number of 1.41×10^6 , the computed results compared favorably to experimental data and previous numerical results. The situation is quite different for angles of attack above stall, however. Although a converged steady-state solution was obtained for $\alpha = 8.0$ degrees, the lift and drag coefficients obtained from this solution did not compare well with experimental data. The predicted lift for this condition was too high, while the drag coefficient

was too low compared to experimental values. Potapczuk has shown that the flow is really unsteady at this angle of attack, and pressures must be averaged over several vortex shedding periods to obtain an accurate estimate of the lift and drag.

While some success was achieved in the computation of the global airfoil performance parameters, the computed local flowfield results generally compared poorly with the experimental data. In particular, the separated flow behind the ice horns reattached sooner than expected for $\alpha=2.0$ and $\alpha=4.0$ degrees, resulting in computed separation bubbles much smaller than those measured experimentally. A comparison of the experimental and computed velocity profiles in the upper surface separation bubble showed that the computed separation region is also much thinner than that measured experimentally. Clearly, the local results obtained in this study are of questionable value.

There is no doubt that the Beam-Warming implementation of the full Navier-Stokes equations can be used to obtain accurate flowfield information for the iced airfoil problem, however. Given that reasonable results have been obtained by Cebeci and Potapczuk using approximate forms of the governing equations, one would certainly expect to achieve results as good as or better than these from the full

Navier-Stokes equations. To do so, however, will require further investigation in at least three areas.

One aspect of the problem requiring further study is the effect of the grid on the accuracy of the results. In this study, only two grids were applied to the problem. The first of these grids was used to generate the solutions presented here, and has been described earlier in this thesis. A second grid with about four times as many grid points concentrated around the ice shape was also applied to the problem. The minimum spacing normal to the wall was less by a factor of 10, as well. Unfortunately, a converged solution could not be obtained with this grid. This may have been due to the fact that clustering more points around the leading edge left too few points remaining over the aft portion of the airfoil. Nonetheless, this does illustrate the point that the particular grid used has a major impact on the resulting solution. Given adequate time to properly refine the grid for the iced airfoil, there is every reason to expect that better results could be obtained.

The transition location specified is another aspect of the problem requiring further study. The location of transition has a definite effect on the lift and drag results. Computations were made for $\alpha=2.0$ degrees with the transition location fixed at $x/c=.05$ on both the upper and lower surfaces. Shifting the transition to these

locations, from $x/c = -.018$ on the upper surface and $x/c = .0327$ on the lower surface, resulted in a lift increase of 13.0%, and a decrease in the drag of 13.0%. Based on these results for just two different transition locations, it certainly seems worthwhile to systematically investigate the effects of transition on both the global and local results for a range of transition points.

The final item recommended for further study is the turbulence model. Potapczuk has pointed out that the Baldwin-Lomax turbulence model does not always compute the correct eddy viscosity in a massively separated turbulent boundary layer. This problem arises from the way the length scale in the outer region of the boundary layer is calculated. Based on the length scale, it is possible for the resulting eddy viscosity to be too high or too low. In the present study, no systematic investigation of the turbulence model was accomplished. Frankly, such an undertaking would be a thesis topic in itself. The recommendation for future study of the model is based solely on Potapczuk's reported experiences with the model. Insofar as he has identified the model as a potential problem in computing the flow over an iced airfoil, investigation of this model would definitely be an important part of any follow-on effort designed to improve upon the results of the present study.

Appendix A: Nondimensionalization of the Governing Equations

The procedure used here to normalize the governing equations is based on the method outlined in Anderson and Tannehill (21). As shown in that reference, all terms in the Navier-Stokes equations can be put in dimensionless form by substituting the following definitions into the dimensional form of the equations:

$$\begin{aligned}x^* &= x/c & y^* &= y/c & t^* &= \frac{t}{c/V_\infty} \\u^* &= u/V_\infty & v^* &= v/V_\infty & \rho^* &= \rho/\rho_\infty \\ \mu^* &= \mu/\mu_\infty & \epsilon^* &= \epsilon/\mu_\infty & T^* &= T/T_\infty \\ \rho^* &= \frac{p}{\rho V_\infty^2} & e^* &= \frac{e}{V_\infty^2}\end{aligned} \quad (A1)$$

The procedure is illustrated for representative terms. One shear stress term is non-dimensionalized, along with one heat flux term, the total energy term and the equation of state.

Shear Stress Term

In dimensional form the shear stress term, τ_{xx} , is given by

$$\tau_{xx} = 2(\mu + \epsilon)u_x - 2/3 \left(u_x + v_y \right) \quad (A2)$$

Substituting from Eq (A1) for each term in Eq (A2) yields

$$\begin{aligned} \tau_{xx} = & 2 \left(\mu_{\infty} \mu^* + \mu_{\infty} \epsilon^* \right) \frac{\partial (u^* v_{\infty})}{\partial (x^* c)} \\ & - 2/3 \left(\mu_{\infty} \mu^* + \mu_{\infty} \epsilon^* \right) \left(\frac{\partial (u^* v_{\infty})}{\partial (x^* c)} + \frac{\partial (v^* v_{\infty})}{\partial (y^* c)} \right) \end{aligned} \quad (A3)$$

Factoring out the constants μ_{∞} , v_{∞} , and c gives

$$\begin{aligned} \tau_{xx} = & \frac{\mu_{\infty} v_{\infty}}{c} \left\{ 2 \left(\mu^* + \epsilon^* \right) \frac{\partial u^*}{\partial x^*} \right. \\ & \left. - 2/3 \left(\mu^* + \epsilon^* \right) \left(\frac{\partial u^*}{\partial x^*} + \frac{\partial v^*}{\partial y^*} \right) \right\} \end{aligned} \quad (A4)$$

The Reynolds number based on the chord, c , is defined as

$$Re_c = \frac{\rho_{\infty} v_{\infty} c}{\mu_{\infty}} \quad (A5)$$

which can be rewritten as

$$\frac{\mu_{\infty}}{c} = \frac{\rho_{\infty} v_{\infty}}{Re_c} \quad (A6)$$

so that

$$\frac{\mu_{\infty} V_{\infty}}{c} = \frac{\rho_{\infty} V_{\infty}^2}{Re_c} \quad (A7)$$

Comparing Eq (A7) to Eq (A4), it can be seen that τ_{xx} can be made dimensionless by dividing by ρV_{∞}^2 . Therefore,

$$\tau_{xx}^* = \frac{\tau_{xx}}{\rho V_{\infty}^2} \quad (A8)$$

or

$$\tau_{xx}^* = \frac{1}{Re_c} \left[2(\mu^* + \epsilon^*) \frac{\partial u^*}{\partial x^*} - \frac{2}{3}(\mu^* + \epsilon^*) \left(\frac{\partial u^*}{\partial x^*} + \frac{\partial v^*}{\partial y^*} \right) \right] \quad (A9)$$

The dimensionless expressions for τ_{xy} and τ_{yy} can be derived in the same manner.

Heat Flux Term

The dimensional heat flux term in the x-direction is

$$q_x = -c_p \left(\frac{\mu}{Pr} + \frac{\epsilon}{Pr_T} \right) \frac{\partial T}{\partial x} \quad (A10)$$

Substituting for the dimensional variables gives

$$q_x = -c_p \left(\frac{\mu_\infty \mu^*}{Pr} + \frac{\mu_\infty \epsilon^*}{Pr_T} \right) \frac{\partial (T_\infty T^*)}{\partial (x^* c)} \quad (A11)$$

which can be rearranged as

$$q_x = \frac{-c_p T_\infty \mu_\infty}{c} \left(\frac{\mu^*}{Pr} + \frac{\epsilon^*}{Pr_T} \right) \frac{\partial T^*}{\partial x^*} \quad (A12)$$

Now, c_p can be written as

$$c_p = \frac{\gamma R}{\gamma - 1} \quad (A13)$$

and T_∞ can be expressed in terms of the Mach number as

$$T_\infty = \frac{v_\infty^2}{\gamma R M_\infty^2} \quad (A14)$$

Substituting Eqs (A5), (A6), (A13), and (A14) into Eq (A12) yields

$$q_x = \rho_\infty v_\infty^3 \left\{ \frac{-1}{(\gamma - 1) M_\infty^2 Re_c} \left(\frac{\mu^*}{Pr} + \frac{\epsilon^*}{Pr_T} \right) \frac{\partial T^*}{\partial x^*} \right\} \quad (A15)$$

from which

$$q_x^* = \frac{q_x}{\rho_\infty V_\infty^3} \quad (\text{A16})$$

Similarly,

$$q_y^* = \frac{q_y}{\rho_\infty V_\infty^3} \quad (\text{A17})$$

Total Energy Term

The total energy is given by

$$E_t = \rho \left(e + \frac{u^2 + v^2}{2} \right) \quad (\text{A18})$$

Substituting for the dimensional variables results in

$$E_t = \rho_\infty V_\infty^2 \left(\rho^* e^* + \rho^* \frac{u^{*2} + v^{*2}}{2} \right) \quad (\text{A19})$$

from which

$$E_t^* = \frac{E_t}{\rho V_\infty^2} \quad (\text{A20})$$

or

$$E_t^* = \rho^* \left(e^* + \frac{u^{*2} + v^{*2}}{2} \right) \quad (\text{A21})$$

Equation of State

Dimensionless forms of the pressure and temperature can be derived from the equation of state given by

$$p = \rho RT \quad (A22)$$

The gas constant, R , is related to the specific heat at constant volume, c_v , by

$$R = c_v(\gamma-1) \quad (A23)$$

and c_v is in turn related to the temperature by

$$e = c_v T \quad (A24)$$

Combining Eqs (A23) and (A24) with Eq (A22) gives

$$p = \rho e(\gamma-1) \quad (A25)$$

Substituting for ρ and e from Eq (A1) yields

$$p = \rho^* \rho_\infty V_\infty^2 e^*(\gamma-1) \quad (A26)$$

which shows

$$p^* = \frac{p}{\rho_\infty V_\infty^2} \quad (A27)$$

or

$$p^* = \rho^* e^* (\gamma - 1) \quad (\text{A28})$$

The temperature can be non-dimensionalized from

$$T = \frac{p}{\rho R} \quad (\text{A29})$$

by substituting for the dimensional variables as

$$T_\infty T^* = \frac{p^* \rho_\infty V_\infty^2}{\rho^* \rho_\infty R} \quad (\text{A30})$$

But V_∞^2 is related to Mach number by

$$V_\infty^2 = M_\infty^2 \gamma R T_\infty \quad (\text{A31})$$

Making this substitution into Eq (A30) gives the dimensionless temperature

$$T^* = \frac{\gamma M_\infty^2 p^*}{\rho^*} \quad (\text{A32})$$

With these definitions, all variables have been put in dimensionless form.

Appendix B: Transformation to Generalized Coordinates

The transformation of the governing equations to generalized coordinates presented in this appendix is based on the method outlined in Anderson and Tannehill (21). From that reference, the general transformation from the physical domain (x,y) to the computational domain (ξ,η) is given by

$$\xi = \xi(x,y) \quad (B1)$$

$$\eta = \eta(x,y) \quad (B2)$$

This transformation is accomplished by applying the following chain rule to the Navier Stokes equations:

$$\frac{\partial(\cdot)}{\partial x} = \xi_x \frac{\partial(\cdot)}{\partial \xi} + \eta_x \frac{\partial(\cdot)}{\partial \eta} \quad (B3)$$

$$\frac{\partial(\cdot)}{\partial y} = \xi_y \frac{\partial(\cdot)}{\partial \xi} + \eta_y \frac{\partial(\cdot)}{\partial \eta} \quad (B4)$$

where ξ_x , η_x , ξ_y , and η_y are the metrics of the transformation. Differential changes in the x and y directions in the physical domain are related to corresponding changes in the computational domain by

$$d\xi = \xi_x dx + \xi_y dy \quad (B5)$$

$$d\eta = \eta_x dx + \eta_y dy \quad (B6)$$

Rewritten in matrix form, these equations are

$$\begin{bmatrix} d\xi \\ d\eta \end{bmatrix} = \begin{bmatrix} \xi_x & \xi_y \\ \eta_x & \eta_y \end{bmatrix} \begin{bmatrix} dx \\ dy \end{bmatrix} \quad (\text{B7})$$

A similar expression holds for the inverse transformation from (ξ, η) to (x, y) , which can be written as

$$dx = x_\xi d\xi + x_\eta d\eta \quad (\text{B8})$$

$$dy = y_\xi d\xi + y_\eta d\eta \quad (\text{B9})$$

or in matrix form

$$\begin{bmatrix} dx \\ dy \end{bmatrix} = \begin{bmatrix} x_\xi & x_\eta \\ y_\xi & y_\eta \end{bmatrix} \begin{bmatrix} d\xi \\ d\eta \end{bmatrix} \quad (\text{B10})$$

Substituting for $[dx, dy]$ in Eq (B7) from Eq (B10) gives

$$\begin{bmatrix} d\xi \\ d\eta \end{bmatrix} = \begin{bmatrix} \xi_x & \xi_y \\ \eta_x & \eta_y \end{bmatrix} \begin{bmatrix} x_\xi & x_\eta \\ y_\xi & y_\eta \end{bmatrix} \begin{bmatrix} d\xi \\ d\eta \end{bmatrix} \quad (\text{B11})$$

which means the transformation metrics are related by

$$\begin{bmatrix} \xi_x & \xi_y \\ \eta_x & \eta_y \end{bmatrix} = \begin{bmatrix} x_\xi & x_\eta \\ y_\xi & y_\eta \end{bmatrix}^{-1} \quad (\text{B12})$$

Recall that for a square matrix [A]

$$[A^{-1}] = \frac{[C]}{\det [A]} \quad (\text{B13})$$

where

[C] = cofactor matrix of [A]
 C_{ij} = elements of [C] which occupy same position as a_{ji}

Now let [A] be given by

$$[A] = \begin{bmatrix} \xi_x & \xi_y \\ \eta_x & \eta_y \end{bmatrix} \quad (\text{B14})$$

so that

$$[A^{-1}] = \frac{\begin{bmatrix} \eta_y & -\xi_y \\ -\eta_x & \xi_x \end{bmatrix}}{\begin{vmatrix} \xi_x & \xi_y \\ \eta_x & \eta_y \end{vmatrix}} \quad (\text{B15})$$

But from Eq (B12)

$$[A^{-1}] = \begin{bmatrix} x_\xi & x_\eta \\ y_\xi & y_\eta \end{bmatrix} \quad (\text{B16})$$

so it follows that

$$\begin{bmatrix} \eta_y & -\epsilon_y \\ -\eta_x & \epsilon_x \end{bmatrix} = \begin{vmatrix} \epsilon_x & \epsilon_y \\ \eta_x & \eta_y \end{vmatrix} \begin{bmatrix} x_\xi & x_\eta \\ y_\xi & y_\eta \end{bmatrix} \quad (\text{B17})$$

Defining the determinant above as the Jacobian of the transformation, J , the metrics are related by

$$\begin{aligned} \eta_x &= -J y_\xi \\ \eta_y &= J x_\xi \\ \epsilon_x &= J y_\eta \\ \epsilon_y &= -J x_\eta \end{aligned} \quad (\text{B18})$$

where

$$J = \begin{vmatrix} \epsilon_x & \epsilon_y \\ \eta_x & \eta_y \end{vmatrix} = \epsilon_x \eta_y - \eta_x \epsilon_y \quad (\text{B19})$$

With the relationship between the transformation metrics defined, the chain rule can be applied to the Navier-Stokes equations given by

$$\frac{\partial U}{\partial t} + \frac{\partial E}{\partial x} + \frac{\partial F}{\partial y} = 0 \quad (\text{B20})$$

Applying the chain rule yields

$$U_t + E_\xi \epsilon_x + E_\eta \eta_x + F_\xi \epsilon_y + F_\eta \eta_y = 0 \quad (\text{B21})$$

It is desirable to have this equation in strong conservation form. Anderson and Tannehill (21:254) suggest this can be done by dividing through by the Jacobian and rearranging into conservational law form by adding and subtracting like terms. For example,

$$\frac{\partial}{\partial t} \left(\frac{U}{J} \right) = \frac{1}{J} \frac{\partial U}{\partial t} + U \frac{\partial}{\partial t} \left(\frac{1}{J} \right) \quad (\text{B22})$$

Since J is not a function of t , this reduces to

$$U_t = J \left(\frac{U}{J} \right)_t \quad (\text{B23})$$

For the next term, $\epsilon_x \epsilon_\xi$, dividing by J and expanding gives

$$\left(\frac{\epsilon_x \epsilon}{J} \right)_\xi = \left(\frac{\epsilon_x \epsilon_\xi}{J} \right) + \epsilon \left(\frac{\epsilon_x}{J} \right)_\xi \quad (\text{B24})$$

from which

$$\epsilon_x \epsilon_\xi = J \left\{ \left(\frac{\epsilon_x \epsilon}{J} \right)_\xi - \epsilon \left(\frac{\epsilon_x}{J} \right)_\xi \right\} \quad (\text{B25})$$

Using this procedure for the remaining terms in Eq (B21) gives the following:

$$\eta_x E_\eta = J \left\{ \left(\frac{\eta_x E}{J} \right)_\eta - E \left(\frac{\eta_x}{J} \right)_\eta \right\} \quad (\text{B26})$$

$$\xi_y F_\xi = J \left\{ \left(\frac{\xi_y F}{J} \right)_\xi - F \left(\frac{\xi_y}{J} \right)_\xi \right\} \quad (\text{B27})$$

$$\eta_y F_\eta = J \left\{ \left(\frac{\eta_y F}{J} \right)_\eta - F \left(\frac{\eta_y}{J} \right)_\eta \right\} \quad (\text{B28})$$

Combining terms with like derivatives yields

$$\begin{aligned} J \left\{ \left(\frac{U}{J} \right)_t + \left(\frac{E\xi_x + F\xi_y}{J} \right)_\xi + \left(\frac{E\eta_x + F\eta_y}{J} \right)_\eta \right\} \\ - J \left\{ E \left[\left(\frac{\xi_x}{J} \right)_\xi + \left(\frac{\eta_x}{J} \right)_\eta \right] \right. \\ \left. + F \left[\left(\frac{\xi_y}{J} \right)_\xi + \left(\frac{\eta_y}{J} \right)_\eta \right] \right\} = 0 \quad (\text{B29}) \end{aligned}$$

The terms in the last set of braces are equal to zero, however, as can be seen by substituting the previously derived expressions for the metrics. For example, using

$$\xi_x = Jy_\eta \quad (\text{B30})$$

$$\eta_x = -Jy_\xi \quad (\text{B31})$$

the first term in the last brace becomes

$$- JE \left\{ \left(\frac{Jy_\eta}{J} \right)_\xi + \left(\frac{-Jy_\xi}{J} \right)_\eta \right\} \quad (B32)$$

or

$$- JE \left(y_{\eta\xi} - y_{\xi\eta} \right) \quad (B33)$$

which equals zero. Similarly, the second term in the last brace of Eq (B29) reduces to zero. Therefore, the strong conservative form of the Navier-Stokes equations in generalized coordinates is

$$\left(\frac{U}{J} \right)_t + \left(\frac{E\xi_x + F\xi_y}{J} \right)_\xi + \left(\frac{E\eta_x + F\eta_y}{J} \right)_\eta = 0 \quad (B34)$$

or

$$\hat{U}_t + \hat{E}_\xi + \hat{F}_\eta = 0 \quad (B35)$$

where

$$\begin{aligned} \hat{U} &= U/J \\ \hat{E} &= 1/J \left(\xi_x E + \xi_y F \right) \\ \hat{F} &= 1/J \left(\eta_x E + \eta_y F \right) \end{aligned} \quad (B36)$$

This equation can be further manipulated to separate the viscous and inviscid terms, as well as the cross derivative terms. Define a vector E_1 to contain the inviscid flux and

pressure terms in \hat{E} , and a vector E_2 to contain the corresponding terms in \hat{F} . The components of E_1 are

$$E_1 = \frac{1}{J} \begin{bmatrix} \epsilon_x \rho u + \epsilon_y \rho v \\ \epsilon_x (\rho u^2 + p) + \epsilon_y \rho u v \\ \epsilon_x \rho u v + \epsilon_y (\rho v + p^2) \\ \epsilon_x (E_t + p) u + \epsilon_y (E_t + p) v \end{bmatrix} \quad (B37)$$

which can be rewritten as

$$E_1 = \frac{1}{J} \begin{bmatrix} \rho (\epsilon_x u + \epsilon_y v) \\ \rho u (\epsilon_x u + \epsilon_y v) + \epsilon_x p \\ \rho v (\epsilon_x u + \epsilon_y v) + \epsilon_y p \\ (E_t + p) (\epsilon_x u + \epsilon_y v) \end{bmatrix} \quad (B38)$$

The velocity components in the computational domain, termed contravariant velocities, are defined as

$$U = \epsilon_x u + \epsilon_y v \quad (B39)$$

$$V = \eta_x u + \eta_y v \quad (B40)$$

Using these definitions, E_1 can be written as

$$E_1 = \frac{1}{J} \begin{bmatrix} \rho U \\ \rho u U + \epsilon_x p \\ \rho v U + \epsilon_y p \\ (E_t + p) U \end{bmatrix} \quad (B41)$$

In a similar manner,

$$E_2 = \frac{1}{J} \begin{bmatrix} \rho V \\ \rho u V + \eta_x p \\ \rho v V + \eta_y p \\ (E_t + p) V \end{bmatrix} \quad (B42)$$

The remaining viscous flux and heat flux terms are separated into four vectors such that the cross-derivative terms are isolated. These vectors, in general, are expressed as

$$V = v_1 (\hat{U}, \hat{U}_\xi) + v_2 (\hat{U}, \hat{U}_\eta) \quad (B43)$$

$$W = w_1 (\hat{U}, \hat{U}_\xi) + w_2 (\hat{U}, \hat{U}_\eta) \quad (B44)$$

where v_1 and v_2 represent the terms in \hat{E} and w_1 and w_2 represent terms in \hat{F} . The final form of the transformed Navier-Stokes equations are

$$\hat{U}_t + E_{1\xi} + E_{2\eta} = v_{1\xi}(\hat{U}, \hat{U}_\xi) + v_{2\xi}(\hat{U}, \hat{U}_\eta) + w_{1\eta}(\hat{U}, \hat{U}_\xi) + w_{2\eta}(\hat{U}, \hat{U}_\eta) \quad (B45)$$

The components of vectors v_1 , v_2 , w_1 , and w_2 can be assembled after applying the chain rule to each viscous and heat flux term-- τ_{xx} , τ_{xy} , τ_{yy} , q_x , and q_y . For example, τ_{xx} is given by

$$\tau_{xx} = 2(\mu + \epsilon) u_x - 2/3(\mu + \epsilon) (u_x + v_y) \quad (B46)$$

After applying the chain rule, this equation becomes

$$\tau_{xx} = 4/3(\mu + \epsilon) (\xi_x u_\xi + \eta_x u_\eta) - 2/3(\mu + \epsilon) (\xi_y v_\xi + \eta_y v_\eta) \quad (B47)$$

The other viscous terms become

$$\tau_{xy} = (\mu + \epsilon) (u_\xi \xi_y + u_\eta \eta_y + v_\xi \xi_x + v_\eta \eta_x) \quad (B48)$$

$$\tau_{yy} = 4/3(\mu + \epsilon) (\xi_y v_\xi + \eta_y v_\eta) - 2/3(\mu + \epsilon) (\xi_x u_\xi + \eta_x u_\eta) \quad (B49)$$

Applying the chain rule to the heat flux terms yields

$$q_x = -c_p \left(\frac{\mu}{Pr} + \frac{\epsilon}{Pr_T} \right) \left(T_\xi \epsilon_x + T_\eta \eta_x \right) \quad (B50)$$

$$q_y = -c_p \left(\frac{\mu}{Pr} + \frac{\epsilon}{Pr_T} \right) \left(T_\xi \epsilon_y + T_\eta \eta_y \right) \quad (B51)$$

Visbal (23) has shown that these terms can be assembled such that V_1 and W_1 are functions of derivatives of ξ only, while V_2 and W_2 are functions of derivatives of η only. The result of rearranging the terms in this manner follows:

$$V_1 = \frac{1}{J} \begin{bmatrix} 0 \\ b_1 u_\xi + b_2 v_\xi \\ b_2 u_\xi + b_3 v_\xi \\ b_1 u u_\xi + b_2 \left(u v_\xi + v u_\xi \right) + b_3 v v_\xi + b_4 T_\xi \end{bmatrix} \quad (B52)$$

$$V_2 = \frac{1}{J} \begin{bmatrix} 0 \\ c_1 u_\eta + c_2 v_\eta \\ c_3 u_\eta + c_4 v_\eta \\ c_1 u u_\eta + c_2 u v_\eta + c_3 v u_\eta + c_4 v v_\eta + c_5 T_\eta \end{bmatrix} \quad (B53)$$

$$W_1 = \frac{1}{J} \begin{bmatrix} 0 \\ c_1 u_\xi + c_3 v_\xi \\ c_2 u_\xi + c_4 v_\xi \\ c_1 u u_\xi + c_2 v u_\xi + c_3 u v_\xi + c_4 v v_\xi + c_5 T_\xi \end{bmatrix} \quad (B54)$$

$$W_2 = \frac{1}{J} \begin{bmatrix} 0 \\ d_1 u_\eta + d_2 v_\eta \\ d_2 u_\eta + d_3 v_\eta \\ d_1 u u_\eta + d_2 (u v_\eta + v u_\eta) + d_3 v v_\eta + d_4 T_\eta \end{bmatrix} \quad (B55)$$

where

$$\begin{aligned} b_1 &= (\mu + \epsilon) \left(\frac{4}{3} \epsilon_x^2 + \epsilon_y^2 \right) \\ b_2 &= (\mu + \epsilon) \left(\frac{1}{3} \epsilon_x \epsilon_y \right) \\ b_3 &= (\mu + \epsilon) \left(\epsilon_x^2 + \frac{4}{3} \epsilon_y^2 \right) \\ b_4 &= c_p \left(\frac{\mu}{\rho r} + \frac{\mu}{\rho r_T} \right) \left(\epsilon_x^2 + \epsilon_y^2 \right) \end{aligned} \quad (B56)$$

$$\begin{aligned} c_1 &= - (\mu + \epsilon) \left(\frac{4}{3} \epsilon_x \eta_x + \epsilon_y \eta_y \right) \\ c_2 &= - (\mu + \epsilon) \left(\frac{2}{3} \epsilon_x \eta_y - \epsilon_y \eta_x \right) \\ c_3 &= (\mu + \epsilon) \left(\epsilon_x \eta_y - \frac{2}{3} \epsilon_y \eta_x \right) \\ c_4 &= - (\mu + \epsilon) \left(\epsilon_x \eta_x + \frac{4}{3} \epsilon_y \eta_y \right) \\ c_5 &= - \left(\frac{\mu}{\rho r} + \frac{\mu}{\rho r_T} \right) \left(\epsilon_x \eta_x + \epsilon_y \eta_y \right) \end{aligned} \quad (B57)$$

$$\begin{aligned} d_1 &= (\mu + \epsilon) \left(\frac{4}{3} \eta_x^2 + \eta_y^2 \right) \\ d_2 &= (\mu + \epsilon) \left(\frac{1}{3} \eta_x \eta_y \right) \\ d_3 &= (\mu + \epsilon) \left(\eta_x^2 + \frac{4}{3} \eta_y^2 \right) \\ d_4 &= c_p \left(\frac{\mu}{\rho r} + \frac{\mu}{\rho r_T} \right) \left(\eta_x^2 + \eta_y^2 \right) \end{aligned} \quad (B58)$$

Appendix C: Jacobian Matrices for the Beam-Warming Scheme

The Jacobian matrices which arise during linearization of the governing equations are derived in this section. Two inviscid flux vectors and two viscous flux vectors need to be linearized. The Jacobian for one of each will be derived to illustrate the method, and the other results simply stated. Derivation of the linearized equations follows the method outlined in Anderson and Tannehill (21), and uses Visbal's (23) notation for the Jacobian matrix elements.

Jacobians for the Inviscid Terms

The inviscid flux vectors are linearized as

$$\Delta E_1^n = [A] \Delta \hat{U}^n \quad (C1)$$

$$\Delta E_2^n = [B] \Delta \hat{U}^n \quad (C2)$$

where the Jacobians are

$$[A] = \frac{\partial E_1(\hat{U})^n}{\partial \hat{U}} \quad (C3)$$

$$[B] = \frac{\partial E_2(\hat{U})^n}{\partial \hat{U}} \quad (C4)$$

The Jacobian for E_1 is derived by expressing E_1 in terms of the components of \hat{U} , where \hat{U} is given by

$$\hat{U} = \frac{1}{J} \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ E_t \end{bmatrix} = \frac{1}{J} \begin{bmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \end{bmatrix} \quad (C5)$$

E_1 is expressed as

$$E_1 = \frac{1}{J} \begin{bmatrix} \epsilon_x \rho u + \epsilon_y \rho v \\ \epsilon_x (\rho u^2 + p) + \epsilon_y \rho u v \\ \epsilon_x \rho u v + \epsilon_y (\rho v^2 + p) \\ \epsilon_x (E_t + p) u + \epsilon_y (E_t + p) v \end{bmatrix} = \frac{1}{J} \begin{bmatrix} E_1 \\ E_2 \\ E_3 \\ E_4 \end{bmatrix} \quad (C6)$$

In terms of the components of \hat{U} , the E_1 components can be written as

$$E_1 = 1/J (U_2 \epsilon_x + U_3 \epsilon_y) \quad (C7)$$

$$E_2 = 1/J \left(\frac{U_2^2}{U_1} \epsilon_x + \frac{U_2 U_3}{U_1} \epsilon_y + \epsilon_x p \right) \quad (C8)$$

$$E_3 = 1/J \left(\frac{U_2 U_3}{U_1} \epsilon_x + \frac{U_3^2}{U_1} \epsilon_y + \epsilon_y p \right) \quad (C9)$$

$$E_4 = 1/J \left[(E_t + p) \left(\frac{U_2}{U_1} \epsilon_x + \frac{U_3}{U_1} \epsilon_y \right) \right] \quad (C10)$$

The pressure term appearing in Eqs (C8), (C9), and (C10) must be expressed in terms of \hat{U} components before E_1 can be differentiated. The pressure is given by

$$p = (\gamma - 1) \left[E_t - 1/2 \rho \left(u^2 + v^2 \right) \right] \quad (C11)$$

which can be rewritten as

$$p = (\gamma - 1) \left[U_4 - \frac{1}{2U_1} \left(U_2^2 + U_3^2 \right) \right] \quad (C12)$$

Now that all components have been expressed in terms of the \hat{U} variables, the derivative of E_1 can be taken with respect to \hat{U} . The resulting Jacobian, $[A]$, can be expressed as

$$[A] = \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix} \quad (C13)$$

where

$$a_{ij} = \frac{\partial E_i}{\partial U_j} \quad \text{for } i = 1, 4; j = 1, 4 \quad (C14)$$

For example,

$$a_{11} = \frac{\partial E_1}{\partial U_1} = \frac{\partial}{\partial U_1} (U_2 \xi_x + U_3 \xi_y) = 0 \quad (C15)$$

$$a_{12} = \frac{\partial E_1}{\partial U_2} = \frac{\partial}{\partial U_2} (U_2 \xi_x + U_3 \xi_y) = \xi_x \quad (C16)$$

$$a_{13} = \frac{\partial E_1}{\partial U_3} = \frac{\partial}{\partial U_3} (U_2 \xi_x + U_3 \xi_y) = \xi_y \quad (C17)$$

$$a_{14} = \frac{\partial E_1}{\partial U_4} = \frac{\partial}{\partial U_4} (U_2 \xi_x + U_3 \xi_y) = 0 \quad (C18)$$

The remaining elements are determined in the same manner.

The result is

$$\begin{aligned} a_{21} &= -uU + \phi \xi_x \\ a_{22} &= U - (\gamma-2)u\xi_x \\ a_{23} &= u\xi_y - (\gamma-1)v\xi_x \\ a_{24} &= (\gamma-1)\xi_x \end{aligned} \quad (C19)$$

$$\begin{aligned} a_{31} &= -vU + \phi \xi_y \\ a_{32} &= v\xi_x - (\gamma-1)u\xi_y \\ a_{33} &= U - (\gamma-2)v\xi_y \\ a_{34} &= (\gamma-1)\xi_y \end{aligned} \quad (C20)$$

$$\begin{aligned}
a_{41} &= \left(2\phi - \frac{\gamma E_t}{\rho} \right) U \\
a_{42} &= \left(\frac{\gamma E_t}{\rho} - \phi \right) \xi_x - (\gamma-1)uU \\
a_{43} &= \left(\frac{\gamma E_t}{\rho} - \phi \right) \xi_y - (\gamma-1)vU \\
a_{44} &= \gamma U
\end{aligned} \tag{C21}$$

where

$$U = \xi_x u + \xi_y v \tag{C22}$$

$$\phi = 1/2 (\gamma-1) \left(u^2 + v^2 \right) \tag{C23}$$

The Jacobian for the E_2 vector is derived in the same way. E_2 is given by

$$E_2 = \frac{1}{J} \begin{bmatrix} \eta_x \rho u + \eta_y \rho v \\ \eta_x (\rho u^2 + p) + \eta_y \rho u v \\ \eta_x \rho u v + \eta_y (\rho v + p^2) \\ \eta_x (E_t + p) u + \eta_y (E_t + p) v \end{bmatrix} = \frac{1}{J} \begin{bmatrix} E_1 \\ E_2 \\ E_3 \\ E_4 \end{bmatrix} \tag{C24}$$

The components of E_2 can be expressed in terms of the \hat{U} components as

$$E_1 = 1/J (U_2 \eta_x + U_3 \eta_y) \tag{C25}$$

$$E_2 = 1/J \left(\frac{U_2^2}{U_1} \eta_x + \frac{U_2 U_3}{U_1} \eta_y + \eta_x p \right) \quad (C26)$$

$$E_3 = 1/J \left(\frac{U_2 U_3}{U_1} \eta_x + \frac{U_3^2}{U_1} \eta_y + \eta_y p \right) \quad (C27)$$

$$E_4 = 1/J \left[\left(E_t + p \right) \left(\frac{U_2}{U_1} \eta_x + \frac{U_3}{U_1} \eta_y \right) \right] \quad (C28)$$

where the expression for the pressure is the same as derived previously in Eq (C12).

Defining the Jacobian for E_2 as

$$[B] = \begin{bmatrix} b_{11} & b_{12} & b_{13} & b_{14} \\ b_{21} & b_{22} & b_{23} & b_{24} \\ b_{31} & b_{32} & b_{33} & b_{34} \\ b_{41} & b_{42} & b_{43} & b_{44} \end{bmatrix} \quad (C29)$$

and differentiating E_2 with respect to \hat{U} as before, results in the following components:

$$\begin{aligned} b_{11} &= 0 \\ b_{12} &= \eta_x \\ b_{13} &= \eta_y \\ b_{14} &= 0 \end{aligned} \quad (C30)$$

$$\begin{aligned}
b_{21} &= -uV + \phi\eta_x \\
b_{22} &= V - (\gamma-2)u\eta_x \\
b_{23} &= u\eta_y - (\gamma-1)v\eta_x \\
b_{24} &= (\gamma-1)\eta_x
\end{aligned}
\tag{C31}$$

$$\begin{aligned}
b_{31} &= -vV + \phi\eta_y \\
b_{32} &= v\eta_x - (\gamma-1)u\eta_y \\
b_{33} &= V - (\gamma-2)v\eta_y \\
b_{34} &= (\gamma-1)\eta_y
\end{aligned}
\tag{C32}$$

$$\begin{aligned}
b_{41} &= \left(2\phi - \frac{\gamma E_t}{\rho} \right) v \\
b_{42} &= \left(\frac{\gamma E_t}{\rho} - \phi \right) \eta_x - (\gamma-1)uV \\
b_{43} &= \left(\frac{\gamma E_t}{\rho} - \phi \right) \eta_y - (\gamma-1)vV \\
b_{44} &= \gamma V
\end{aligned}
\tag{C33}$$

where

$$V = \eta_x u + \eta_y v \tag{C34}$$

Jacobians for the Viscous Terms

The Jacobians for the viscous terms arise from the linearization of V_1 and W_2 , given by

$$\Delta V_1^n(\hat{U}, \hat{U}_\xi) = [R]^n \Delta \hat{U}_\xi^n \tag{C35}$$

$$\Delta W_2^n(\hat{U}, \hat{U}_\eta) = [S]^n \Delta \hat{U}_\eta^n \tag{C36}$$

where the Jacobians are

$$\left[R \right] = \frac{\partial v_1^n(\hat{U}, \hat{U}_\xi)}{\partial \hat{U}_\xi} \quad (C37)$$

$$\left[S \right] = \frac{\partial w_2^n(\hat{U}, \hat{U}_\eta)}{\partial \hat{U}_\eta} \quad (C38)$$

As before, the Jacobian matrices can be written as

$$\left[R \right] = \begin{bmatrix} r_{11} & r_{12} & r_{13} & r_{14} \\ r_{21} & r_{22} & r_{23} & r_{24} \\ r_{31} & r_{32} & r_{33} & r_{34} \\ r_{41} & r_{42} & r_{43} & r_{44} \end{bmatrix} \quad (C38)$$

and

$$\left[S \right] = \begin{bmatrix} s_{11} & s_{12} & s_{13} & s_{14} \\ s_{21} & s_{22} & s_{23} & s_{24} \\ s_{31} & s_{32} & s_{33} & s_{34} \\ s_{41} & s_{42} & s_{43} & s_{44} \end{bmatrix} \quad (C40)$$

The Jacobian $\left[R \right]$ can be obtained by getting an expression for \hat{U}_ξ , rewriting V_1 in terms of the \hat{U}_ξ components, and differentiating the resulting vector with

respect to \hat{U}_ξ . \hat{U}_ξ is obtained by differentiating \hat{U} , which results in

$$\hat{U}_\xi = \frac{1}{J} \begin{bmatrix} \rho_\xi \\ \rho u_\xi + u \rho_\xi \\ \rho v_\xi + v \rho_\xi \\ E_{t\xi} \end{bmatrix} = \frac{1}{J} \begin{bmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \end{bmatrix} \quad (C41)$$

Also, V_1 is given by

$$V_1 = \frac{1}{J} \begin{bmatrix} 0 \\ b_1 u_\xi + b_2 v_\xi \\ b_2 u_\xi + b_3 v_\xi \\ b_1 u u_\xi + b_2 (u v_\xi + v u_\xi) + b_3 v v_\xi + b_4 T_\xi \end{bmatrix} \quad (C42)$$

$$= \frac{1}{J} \begin{bmatrix} V_1 \\ V_2 \\ V_3 \\ V_4 \end{bmatrix} \quad (C43)$$

where b_1 , b_2 , b_3 , and b_4 are as defined in Appendix B.

Rewritten in terms of the \hat{U}_ξ components, the components of V_1 can be expressed as

$$v_1 = 0 \quad (C44)$$

$$v_2 = 1/\rho \left[b_1 (U_2 - uU_1) + b_2 (U_3 - vU_1) \right] \quad (C45)$$

$$v_3 = 1/\rho \left[b_2 (U_2 - uU_1) + b_3 (U_3 - vU_1) \right] \quad (C46)$$

$$v_4 = 1/\rho \left[b_1 u (U_2 - uU_1) + b_2 \left[u (U_3 - vU_1) + v (U_2 - uU_1) \right] \right. \\ \left. + b_3 v (U_3 - vU_1) + b_4 T_\xi \right] \quad (C47)$$

In the v_4 component, T_ξ must be cast in terms of \hat{U}_ξ variables before v_1 can be differentiated. From the perfect gas relations,

$$e = c_v T \quad (C48)$$

Assuming c_v is locally constant, T_ξ is given by

$$T_\xi = e_\xi / c_v \quad (C49)$$

where e_ξ must now be expressed in terms of the \hat{U}_ξ component, $E_{t\xi}$. For E_t given as

$$E_t = \rho \left(e + \frac{u^2 + v^2}{2} \right) \quad (C50)$$

differentiation with respect to ξ yields

$$E_{t\xi} = \rho \left(e_{\xi} + uu_{\xi} + vv_{\xi} \right) + \rho_{\xi} \left(e + 1/2(u^2 + v^2) \right) \quad (C51)$$

or

$$E_{t\xi} = \rho \left(e_{\xi} + uu_{\xi} + vv_{\xi} \right) + \rho_{\xi} \left(E_t / \rho \right) \quad (C52)$$

Solving for e_{ξ} yields

$$e_{\xi} = \frac{E_{t\xi}}{\rho} - \frac{\rho_{\xi} E_t}{\rho^2} - uu_{\xi} - vv_{\xi} \quad (C53)$$

so that T_{ξ} can be expressed as

$$T_{\xi} = \frac{1}{c_v} \left(\frac{E_{t\xi}}{\rho} - \frac{\rho_{\xi} E_t}{\rho^2} - uu_{\xi} - vv_{\xi} \right) \quad (C54)$$

This expression can be rewritten in terms of \hat{U}_{ξ} variables as

$$T_{\xi} = \frac{1}{\rho c_v} \left(U_4 - U_1 \frac{E_t}{\rho} - u(U_2 - uU_1) - v(U_3 - vU_1) \right) \quad (C55)$$

Differentiation of V_1 with respect to \hat{U}_{ξ} can now be performed to find the components of $[R]$. For example,

$$r_{21} = \frac{\partial v_2}{\partial u_1} = \frac{\partial}{\partial u_1} \left\{ 1/\rho \left[b_1 (u_2 - u u_1) + b_2 (u_3 - v u_1) \right] \right\} \quad (C56)$$

$$= -1/\rho (b_1 u + b_2 v) \quad (C57)$$

$$r_{22} = \frac{\partial v_2}{\partial u_2} = b_1 / \rho \quad (C58)$$

$$r_{23} = \frac{\partial v_2}{\partial u_3} = b_2 / \rho \quad (C59)$$

$$r_{24} = \frac{\partial v_2}{\partial u_4} = 0 \quad (C60)$$

Applying the differentiation to the remaining components of V_1 yields the following:

$$\begin{aligned} r_{11} &= 0 \\ r_{12} &= 0 \\ r_{13} &= 0 \\ r_{14} &= 0 \end{aligned} \quad (C61)$$

$$\begin{aligned} r_{31} &= -1/\rho (b_2 u + b_3 v) \\ r_{32} &= b_2 / \rho \\ r_{33} &= b_3 / \rho \\ r_{34} &= 0 \end{aligned} \quad (C62)$$

$$\begin{aligned}
r_{41} &= 1/\rho \left[-b_1 u^2 - 2b_2 uv - b_3 v^2 + \frac{b_4}{c_v} \left(\frac{-E_t}{\rho} + u^2 + v^2 \right) \right] \\
r_{42} &= 1/\rho \left[\left(b_1 - \frac{b_4}{c_v} \right) u + b_2 v \right] \\
r_{43} &= 1/\rho \left[\left(b_3 - \frac{b_4}{c_v} \right) v + b_2 u \right] \\
r_{44} &= \frac{b_4}{\rho c_v}
\end{aligned}
\tag{C63}$$

Applying a similar procedure to W_2 , given by

$$W_2 = \frac{1}{J} \begin{bmatrix} 0 \\ d_1 u_\eta + d_2 v_\eta \\ d_2 u_\eta + d_3 v_\eta \\ d_1 uu_\eta + d_2 (uv_\eta + vu_\eta) + d_3 vv_\eta + d_4 T_\eta \end{bmatrix}
\tag{C64}$$

or

$$W_2 = \frac{1}{J} \begin{bmatrix} W_1 \\ W_2 \\ W_3 \\ W_4 \end{bmatrix}
\tag{C65}$$

where d_1 , d_2 , d_3 , and d_4 are as defined in Appendix B, the

components of the Jacobian $[S]$ can be expressed as

$$\begin{aligned}
 s_{11} &= 0 \\
 s_{12} &= 0 \\
 s_{13} &= 0 \\
 s_{14} &= 0
 \end{aligned}
 \tag{C66}$$

$$\begin{aligned}
 s_{21} &= -1/\rho \left(d_1 u + d_2 v \right) \\
 s_{22} &= d_1 / \rho \\
 s_{23} &= d_2 / \rho \\
 s_{24} &= 0
 \end{aligned}
 \tag{C67}$$

$$\begin{aligned}
 s_{31} &= -1/\rho \left(d_2 u + d_3 v \right) \\
 s_{22} &= d_2 / \rho \\
 s_{23} &= d_3 / \rho \\
 s_{24} &= 0
 \end{aligned}
 \tag{C68}$$

$$\begin{aligned}
 s_{41} &= 1/\rho \left[-d_1 u^2 - 2d_2 uv - d_3 v^2 + \frac{d_4}{c_v} \left(\frac{-E_t}{\rho} + u^2 + v^2 \right) \right] \\
 s_{42} &= 1/\rho \left[\left(d_1 - \frac{d_4}{c_v} \right) u + d_2 v \right] \\
 s_{43} &= 1/\rho \left[\left(d_3 - \frac{d_4}{c_v} \right) v + d_2 u \right] \\
 s_{44} &= \frac{d_4}{\rho c_v}
 \end{aligned}
 \tag{C69}$$

Bibliography

1. Bragg, M. B. and W. J. Coirier. "Aerodynamic Measurements of an Airfoil with Simulated Glaze Ice," AIAA-86-0484, AIAA 24th Aerospace Sciences Meeting, January 1986.
2. Pais, M. R. and S. N. Singh. "Determination of the Local Heat Transfer Characteristics on Glaze Ice Accretions on a Cylinder and a NACA 0012 Airfoil," AFWAL-TR-87-3001, Air Force Wright Aeronautical Laboratories, Wright-Patterson Air Force Base, Ohio, April 1987.
3. Jacobs, E. N. "Airfoil Section Characteristics as Affected by Protuberances," NACA Report 446, 1932.
4. Gray, V. H. and U. H. Von Glahn. "Effects of Ice and Frost Formations on Drag of NACA 65-212 Airfoil for Various Modes of Thermal Ice Protection," NACA-TN-2962, 1953.
5. ----- "Aerodynamic Effects Caused by Icing of an Unswept NACA 65A004 Airfoil," NACA-TN-4151, 1957.
6. Shaw, Robert J., Ray G. Sotos, and Frank R. Solano. "An Experimental Study of Airfoil Icing Characteristics," NASA-TM-82790, 1982.
7. Cebeci, Tuncer. "Effects of Environmentally Imposed Roughness on Airfoil Performance," NASA Contractor Report 179639, 1987.
8. Bragg, M. B. and W. J. Coirier. "Detailed Measurements of the Flowfield in the Vicinity of an Airfoil with Glaze Ice," AIAA-85-0409, AIAA 23rd Aerospace Sciences Meeting, January 1985.
9. Bragg, M. B. and S. A. Spring. "An Experimental Study of the Flow Field about an Airfoil with Glaze Ice," AIAA-87-0100, AIAA 25th Aerospace Sciences Meeting, January 1987.
10. Spring, Samuel A. "An Experimental Mapping of the Flow Field Behind a Glaze Ice Shape on a NACA 0012 Airfoil," NASA Contractor Report 180847, January 1988.

11. Khodadoust, Abdollah. "A Flow Visualization Study of the Leading Edge Separation Bubble on a NACA 0012 Airfoil with Simulated Glaze Ice," NASA Contractor Report 180846, January 1988.
12. Cebeci, T. "The Calculation of Flow over Iced Airfoils," AIAA-88-0112, AIAA 26th Aerospace Sciences Meeting, January 1988.
13. Cebeci, T. and A.M.O. Smith. Analysis of Turbulent Boundary Layers, Academic Press, Inc., Orlando, Florida, 1974.
14. Potapczuk, M. G. and P. M. Gerhart. "Progress in Development of a Navier-Stokes Solver for Evaluation of Iced Airfoil Performance," AIAA-85-0410, AIAA 23rd Aerospace Sciences Meeting, January 1985.
15. Potapczuk, Mark. "Numerical Analysis of a NACA 0012 Airfoil with Leading Edge Ice Accretions," AIAA-87-0101, AIAA 25th Aerospace Sciences Meeting, January 1987.
16. Steger, J. L. "Implicit Finite-Difference Simulation of Flow about Arbitrary Two-Dimensional Geometries," AIAA Journal, Vol. 16, pp. 679-686, July 1978.
17. Pulliam, T. H. "Euler and Thin-Layer Navier-Stokes Codes: ARC2D, ARC3D," Notes for Computational Fluid Dynamics User's Workshop, UTSI Publication No. E02-4005-023-84, March 12-16, 1984.
18. Baldwin, B. S. and H. Lomax. "Thin Layer Approximation and Algebraic Model for Separated Turbulent Flows," AIAA-78-257, AIAA 16th Aerospace Sciences Meeting, January 1978.
19. Halim, A. A. M. "A Global Marching Technique for the Prediction of Separated Flows over Arbitrary Airfoils," AIAA-87-0591, AIAA 25th Aerospace Sciences Meeting, January 1987.
20. Beam, R. M. and R. F. Warming. "An Implicit Factored Scheme for the Compressible Navier-Stokes Equations," AIAA Journal, Vol. 16, pp. 393-401, 1978.
21. Anderson, Dale A., John C. Tannehill, and Richard H. Pletcher. Computational Fluid Mechanics and Heat Transfer, Hemisphere Publishing Corporation, New York, N.Y., 1984.

22. Schlichting, Hermann. Boundary-Layer Theory, 7th Edition, McGraw-Hill, New York, N.Y., 1987.
23. Visbal, Miguel R. "Calculation of Viscous Transonic Flows About a Supercritical Airfoil," AFWAL-TR-86-3013, Air Force Wright Aeronautical Laboratories, Wright-Patterson Air Force Base, Ohio, July 1986.
24. Amdahl, David J., Air Force Wright Aeronautical Laboratories, Flight Dynamics Laboratory, Wright-Patterson Air Force Base, Ohio, Private Communication, June 1988.
25. Kinsey, Don W. and Timothy J. Barth. "Description of a Hyperbolic Grid Generating Procedure for Arbitrary Two Dimensional Bodies," AFWAL-TM-84-191, Air Force Wright Aeronautical Laboratories, Wright-Patterson Air Force Base, Ohio, 1984.
26. Warming, R. F. and Richard M. Beam. "On the Construction and Application of Implicit Factored Schemes for Conservation Laws," Symposium on Computational Fluid Dynamics, SIAM-AMS Proceedings, Vol. 11, pp. 85-129, 1978.
27. Hafeez, Faran, Air Force Institute of Technology, Wright-Patterson Air Force Base, Ohio, Private Communication, July 1988.
28. Steger, J. L. "Implicit Finite-Difference Simulation of Flow About Arbitrary Geometries with Application to Airfoils," AIAA-77-665, 1977.
29. Cebeci, Tuncer, California State University, Long Beach, California, Private Communication, November 1988.

Vita

Captain Larry A. Coleman was born [REDACTED]

[REDACTED] He graduated from high school in LaGrande, Oregon in 1971, and enlisted in the USAF in October 1972. He was accepted for the Airman Education and Commissioning Program in 1980. While in this program he attended the University of Arizona, from which he received the degree of Bachelor of Science in Aeronautical Engineering in May 1983. Upon graduation, he attended Officer Training School (OTS) and received a commission in August 1983. Following OTS, he was assigned to the Propulsion Laboratory at Wright-Patterson AFB, Ohio, where he served for four years as a test engineer in the Compressor Research Facility. He left the laboratory to enter the School of Engineering at the Air Force Institute of Technology in June 1987.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1a. REPORT SECURITY CLASSIFICATION UNCLASSIFIED		1b. RESTRICTIVE MARKINGS	
2a. SECURITY CLASSIFICATION AUTHORITY		3. DISTRIBUTION / AVAILABILITY OF REPORT Approved for public release; distribution unlimited	
2b. DECLASSIFICATION / DOWNGRADING SCHEDULE			
4. PERFORMING ORGANIZATION REPORT NUMBER(S) AFIT/GAE/AA/88D-4		5. MONITORING ORGANIZATION REPORT NUMBER(S)	
6a. NAME OF PERFORMING ORGANIZATION School of Engineering	6b. OFFICE SYMBOL (if applicable) AFIT/ENY	7a. NAME OF MONITORING ORGANIZATION	
6c. ADDRESS (City, State, and ZIP Code) Air Force Institute of Technology (AU) Wright-Patterson AFB, Ohio 45433-6853		7b. ADDRESS (City, State, and ZIP Code)	
8a. NAME OF FUNDING / SPONSORING ORGANIZATION	8b. OFFICE SYMBOL (if applicable)	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER	
6c. ADDRESS (City, State, and ZIP Code)		10. SOURCE OF FUNDING NUMBERS	
		PROGRAM ELEMENT NO.	PROJECT NO.
		TASK NO.	WORK UNIT ACCESSION NO.
11. TITLE (Include Security Classification) NUMERICAL SIMULATION OF FLOW OVER ICED AIRFOILS			
12. PERSONAL AUTHOR(S) Larry A. Coleman, B.S., Capt, USAF			
13a. TYPE OF REPORT MS Thesis	13b. TIME COVERED FROM _____ TO _____	14. DATE OF REPORT (Year, Month, Day) 1988 December	15. PAGE COUNT 144
16. SUPPLEMENTARY NOTATION			
17. COSATI CODES		18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number) Computational Fluid Dynamics, Navier-Stokes Equations, Subsonic Flows, Airfoils, Applied Aerodynamics, Aircraft Performance	
FIELD	GROUP		
20	4		
19. ABSTRACT (Continue on reverse if necessary and identify by block number) Thesis Advisor: Ahmad A. M. Halim Associate Professor Department of Aeronautics and Astronautics			
20. DISTRIBUTION / AVAILABILITY OF ABSTRACT <input checked="" type="checkbox"/> UNCLASSIFIED/UNLIMITED <input type="checkbox"/> SAME AS RPT. <input type="checkbox"/> DTIC USERS		21. ABSTRACT SECURITY CLASSIFICATION UNCLASSIFIED	
22a. NAME OF RESPONSIBLE INDIVIDUAL Ahmad A. M. Halim, Associate Professor		22b. TELEPHONE (Include Area Code) (513)255-2040	22c. OFFICE SYMBOL AFIT/ENY

Approved
10 Jan 89

↳ This thesis evaluates the

UNCLASSIFIED

The objective of this study is to evaluate the performance of an iced NACA 0012 airfoil numerically. The full Navier-Stokes equations are solved using the Beam-Warming algorithm. Steady-state solutions are obtained at a Mach number of 0.12 and a Reynolds number based on chord of 1.41×10^6 , for angles of attack from 2-8 degrees.

1.41 million

Lift and drag curves obtained from the numerical solutions were compared to experimental data and other numerical results. This comparison showed that the Beam-Warming algorithm provides a good estimate of the lift and drag at angles of attack below stall. Computed lift coefficients were within 11.5% of the experimental data. The drag coefficients differed by as much as 24%. These results were in excellent agreement with other numerical solutions. After stall, however, the code did not predict the expected decrease in lift, and the calculated drag coefficient was much lower than the experimental data.

The comparison of two local flowfield characteristics with the experimental data was less encouraging. A computed velocity profile was compared to the experimental profile for a station in the separated region on the upper surface. This comparison showed that the computed separation bubble is approximately one half the thickness of the bubble measured experimentally. Flow reattachment location is another measure of the predictive accuracy of the numerical scheme. The reattachment locations from the numerical solutions were 30-40% less than the experimental values.

is concluded
The conclusion to be drawn from these comparisons is that accurate results can be obtained for the global performance parameters using this code, but *more* research is needed to refine the local results. To accomplish this, further study is needed in three areas: 1) effect of the grid on solution accuracy, 2) effect of the transition location on the results, and 3) effect of the turbulence model on the solution.

(edc)
A

UNCLASSIFIED