(2)

NAVAL POSTGRADUATE SCHOOL

Monterey, California

THESIS

COMPUTATIONAL INVESTIGATIONS
OF A NACA 0012 AIRFOIL
IN LOW REYNOLDS NUMBER FLOWS

by

Lisa M. Nowak

September 1992

Thesis Advisor: M. F. Platzer
Co-Advisor: M. S. Chandrasekhara

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

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)	5. MONITORING ORGANIZATION REPORT NUMBER(S)
6a. NAME OF PERFORMING ORGANIZATION Naval Postgraduate School 6b. OFFICE SYMBOL *(If applicable)* 55	7a. NAME OF MONITORING ORGANIZATION Naval Postgraduate School
6c. ADDRESS *(City, State, and ZIP Code)* Monterey, CA 93943-5000	7b. ADDRESS *(City, State, and ZIP Code)* Monterey, CA 93943-5000
8a. NAME OF FUNDING/SPONSORING ORGANIZATION 8b. OFFICE SYMBOL *(If applicable)*	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER
8c. ADDRESS *(City, State, and ZIP Code)*	10. SOURCE OF FUNDING NUMBERS Program Element No / Project No / Task No. / Work Unit Accession Number

11. TITLE *(Include Security Classification)*
COMPUTATIONAL INVESTIGATIONS OF A NACA 0012 AIRFOIL IN LOW REYNOLDS NUMBER FLOWS

12. PERSONAL AUTHOR(S) Nowak, Lisa M.

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT *(year, month, day)*	15. PAGE COUNT
Engineer's Thesis	From To	920924	147

16. SUPPLEMENTARY NOTATION
The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

17. COSATI CODES			18. SUBJECT TERMS *(continue on reverse if necessary and identify by block number)*
FIELD	GROUP	SUBGROUP	boundary layer, separation bubble

19. ABSTRACT *(continue on reverse if necessary and identify by block number)*

A steady flow analysis is conducted for a NACA 0012 airfoil in low Reynolds number flows ranging from 540,000 to 1,000,000. Emphasis is placed on prediction and location of the separation bubble. Computational methods include the direct boundary layer method, the viscous-inviscid method, and the time-averaged Navier-Stokes method. Characteristic trends in skin friction coefficient, displacement thickness, and boundary layer velocity profiles with increasing angle of attack are observed. Computational results are compared to each other and to experimental results visualizing the density flowfield using Point-Diffraction Interferometry. Both the viscous-inviscid method and the Navier-Stokes method failed to accurately represent leading edge separation bubbles. The direct boundary layer method, usually considered of very limited usefulness due to a singularity in the underlying equations at separation, is shown to exhibit unexpected recovery behavior for small amounts of separation. Furthermore, the results near the leading edge, where separation bubbles were computed, were validated by the experiment.

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT ☐ UNCLASSIFIED/UNLIMITED ☐ SAME AS REPORT ☐ DTIC USERS	21. ABSTRACT SECURITY CLASSIFICATION Unclassified	
22a. NAME OF RESPONSIBLE INDIVIDUAL Platzer, M. F.	22b TELEPHONE *(Include Area code)*	22c. OFFICE SYMBOL

DD FORM 1473, 84 MAR — 83 APR edition may be used until exhausted. All other editions are obsolete

SECURITY CLASSIFICATION OF THIS PAGE
Unclassified

Approved for public release; distribution is unlimited.

Computational Investigations of a NACA 0012 Airfoil in Low Reynolds Number Flows

by

Lisa M. Nowak
Lieutenant, United States Navy
B.S.,United States Naval Academy, 1985

Submitted in partial fulfillment
of the requirements for the degree of

AERONAUTICAL AND ASTRONAUTICAL ENGINEER

from the

NAVAL POSTGRADUATE SCHOOL
September 1992

Author: ______________________
Lisa M. Nowak

Approved by: ______________________
M. F. Platzer, Thesis Advisor

____
M. S. Chandrasekhara, Co-Advisor

____
D. J. Collins, Chairman
Department of Aeronautics and Astronautics

____
R. S. Elster, Dean of Instruction

ABSTRACT

A steady flow analysis is conducted for a NACA 0012 airfoil in low Reynolds number flows ranging from 540,000 to 1,000,000. Emphasis is placed on prediction and location of the separation bubble. Computational methods include the direct boundary layer method, the viscous-inviscid interaction method, and the time-averaged Navier-Stokes method. Characteristic trends in skin friction coefficient, displacement thickness, and boundary layer velocity profiles with increasing angle of attack are observed. Computational results are compared to each other and to experimental photographs visualizing the density flowfield using Point Diffraction Interferometry. Both the viscous-inviscid method and the Navier-Stokes method failed to accurately represent leading edge separation bubbles. The direct boundary layer method, usually considered of very limited usefulness due to a singularity in the underlying equations at separation, is shown to exhibit unexpected recovery behavior for small amounts of separation. Furthermore, the results near the leading edge, where separation bubbles were computed, were validated by the experiment.

TABLE OF CONTENTS

ACKNOWLEDGEMENTS

I would like to recognize several individuals for their unique contributions to my research. First, my new son Alexander, who patiently spent many of his first eight months with me at the computer. And his father, my husband Richard, whose total support and encouragement made both achievements possible.

My sincerest appreciation and respect go to my advisor, Professor Platzer. In addition to providing me with a solid background in theoretical and computational aerodynamics, he continually offered a broader perspective of technical issues through numerous discussions of applications, government, industry, and education.

Finally, I would like to thank Tony Cricelli for helping me over all the little stumbling blocks I encountered in learning a new computer system and programming language.

I. INTRODUCTION

While there are many reliable solution methods for high Reynolds number flows, the low Reynolds number regime currently has considerably fewer options. This is due in part to the fact that high Reynolds number flows account for most aeronautical applications of interest. There are, however, important applications involving low Reynolds numbers, such as turbomachinery blades. The more likely reason for the lack of reliable codes for low Reynolds numbers is the greater difficulty of accurately representing the flow. Most methods make use of approximations in the formulation of their underlying equations to obtain computational solutions in a reasonable amount of time. These approximations often become less and less accurate as the Reynolds number decreases. An obvious question which may arise concerns the value of bothering with such seemingly limited, simplistic codes when the state of the art is Navier-Stokes (NS) solvers. There are several considerations which make the effort worthwhile. First, not everyone has access to the supercomputers or mini-supercomputers that are necessary for extensive NS solutions. Even if these computers are available, user time may be limited. Preliminary research using a simpler method may cut down the amount of advanced calculations needed considerably, thus reducing overall cost. Another pertinent factor is the time savings. A design team for a new aircraft cannot afford months of detailed refinement using NS solvers when a proposal deadline looms near. Less than NS accuracy is certainly acceptable, especially when the significant gains in speed and cost reduction are considered. An efficient program only needs to calculate to the level of accuracy necessary to accomplish the desired goal. To this end, methods such as the direct boundary layer method and the viscous-inviscid interaction method offer opportunities to investigate low Reynolds number phenomena, such as laminar separation bubbles. A final point to consider is that all methods, including NS solvers, are really approximations in that they involve empirical models for transition and turbulence. The suitability of these models to low Reynolds number flows will influence the results directly.

This investigation explores the various computational methods, comparing them to each other and to experimental data. A panel code was first developed, which provided some of the input needed for a direct boundary layer code. The direct boundary layer code was studied extensively and several modifications were made to enable further analysis of boundary layer profiles and transition effects. Similar calculations were then performed for a viscous-inviscid interaction code. Experimental interferograms, obtained by Point Diffraction Interferometry, provided a reliable reference for comparison. Finally, a Navier-Stokes code was evaluated. Emphasis throughout the investigation was placed on detection and location of laminar separation bubbles, as well as a thorough consideration of transition and turbulence models.

II. PANEL CODES

A. THEORY

In potential flow theory, the flow field around an airfoil may be represented by the velocity potential. Considering contributions from the freestream flow and the source and vorticity distributions, the total potential may be constructed:

$$\Phi = \phi_\infty + \phi_s + \phi_v \tag{2.1}$$

where

$$\begin{aligned} \phi_\infty &= V_\infty(x \cos\alpha + y \sin\alpha) \\ \phi_s &= \int \frac{q(s)}{2\pi} \ln r \, ds \\ \phi_v &= -\int \frac{\gamma(s)}{2\pi} \theta \, ds \end{aligned} \tag{2.2}$$

The source distributions (q) vary from panel to panel, while the vorticity strength (γ) is assumed constant for all panels. The value of representing the flow past an airfoil by surface singularity distributions lies in the fact that these singularity distributions automatically satisfy Laplace's equation, the governing flow equation for inviscid, incompressible flow:

$$\phi_{xx} + \phi_{yy} = 0 \tag{2.3}$$

Since Laplace's equation is a linear homogeneous second order partial differential equation, the superposition principle used in Equation 2.1 is valid. The boundary conditions include flow tangency at control points (midpoints of panels) and the Kutta condition at the trailing edge, requiring equal tangential velocities for the first and last panels. By evaluating the integrals along the airfoil surface, the potential may be determined at any point in the flow field. Each point is defined at a radius (r) and angle (θ) from a chosen reference point on the airfoil. The reference point in this study is the leading edge.

B. PANEL METHOD GEOMETRY

For computational purposes, it is not feasible to evaluate every point in the flow field. The airfoil is represented by a number of defined points, called nodes. More points produce greater resolution and accuracy. One hundred to two hundred points are usually sufficient, with the larger numbers used for more complicated airfoil shapes or more involved calculations. The lines connecting these nodes are the panels. There are (n) panels and (n+1) nodes, with the first and last node overlapping. Figure Figure 2.1 depicts the panel geometry. Numbering starts at the

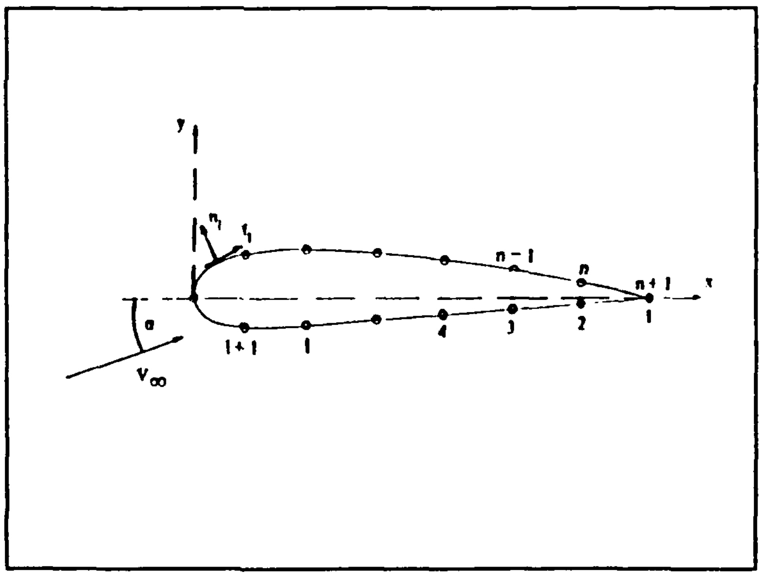

Figure 2.1 Panel Method Geometry

trailing edge, then progresses along the lower surface, leading edge, and upper surface, and ends at the trailing edge. The unit normal vectors ($\hat{n}_i$) are perpendicular to the panels and directed outward from the airfoil surface. The unit tangential vectors ($\hat{t}_i$) are parallel to the panels and the positive direction is defined with increasing numbering (n to n+1). The panels may vary in length, with the exception of the first and last panels, which must be equal in order to use the Kutta condition at the trailing edge.

C. COMPUTER CODE

1. Overview

In order to study the steady, incompressible, inviscid flow over arbitrary airfoils, a panel code called **panel** was developed. The required input consists of the number of nodes on the airfoil surface, the coordinates of the nodes referenced from the leading edge, and the angle of attack in degrees. The program produces normalized velocities and pressure coefficients at each control point as output. The program was later modified to produce an output file compatible with **bl2d**, a direct boundary layer program described in Chapter 2. Additional input consists of Reynolds number and transition information. This data is not used by the program **panel**, but is simply transferred to the output file which will be used as input for **bl2d**.

2. Influence Coefficients

The use of influence coefficients leads to a straightforward procedure for programming the equations. An influence coefficient is defined as the velocity induced at a field point by a unit strength singularity distribution on one panel. For the two dimensional steady flow problem, the following influence coefficients are needed:

- A^n_{ij}: normal velocity component induced at the i^{th} panel control point by unit source distribution on the j^{th} panel

$$A^n_{ij} = \frac{1}{2\pi}[\sin(\theta_i-\theta_j)\ln\frac{r_{i,j+1}}{r_{ij}} + \cos(\theta_i-\theta_j)\,\beta_{ij}], \quad i \neq j$$
$$= \frac{1}{2} \qquad , \quad i=j \tag{2.4}$$

- A^t_{ij}: tangential velocity component induced at the i^{th} panel control point by unit strength source distribution on the j^{th} panel

$$A^t_{ij} = \frac{1}{2\pi}[\sin(\theta_i-\theta_j)\,\beta_{ij} - \cos(\theta_i-\theta_j)\ln\frac{r_{i,j+1}}{r_{ij}}], \quad i \neq j$$
$$= 0 \qquad , \quad i=j \tag{2.5}$$

- B^n_{ij}: normal velocity component induced at the i^{th} panel control point by unit strength vorticity distribution on the j^{th} panel

$$B^n_{ij} = \frac{1}{2\pi}[\cos(\theta_i-\theta_j)\ln\frac{r_{i,j+1}}{r_{ij}} - \sin(\theta_i-\theta_j)\,\beta_{ij}], \quad i \neq j$$
$$= 0 \qquad , \quad i=j \tag{2.6}$$

- B^t_{ij}: tangential velocity component induced at the i^{th} panel control point by unit strength vorticity distribution on the j^{th} panel

$$B^t_{ij} = \frac{1}{2\pi}[\cos(\theta_i-\theta_j)\,\beta_{ij} + \sin(\theta_i-\theta_j)\ln\frac{r_{i,j+1}}{r_{ij}}], \quad i \neq j$$
$$= \frac{1}{2} \qquad , \quad i=j \tag{2.7}$$

where the geometrical quantities, depicted in Figure 2.2, are defined by:

$$r_{ij}=\sqrt{(xm_i-x_j)^2+(ym_i-y_j)^2}$$
$$xm_i=\frac{x_i+x_{i+1}}{2}$$
$$ym_i=\frac{y_i+y_{i+1}}{2} \tag{2.8}$$
$$\theta_i=\arctan(\frac{y_{i+1}-y_i}{x_{i+1}-x_i})$$
$$\beta_{ij}=\arctan(\frac{ym_i-y_{j+1}}{xm_i-x_{j+1}})-\arctan(\frac{ym_i-y_j}{xm_i-x_j})$$

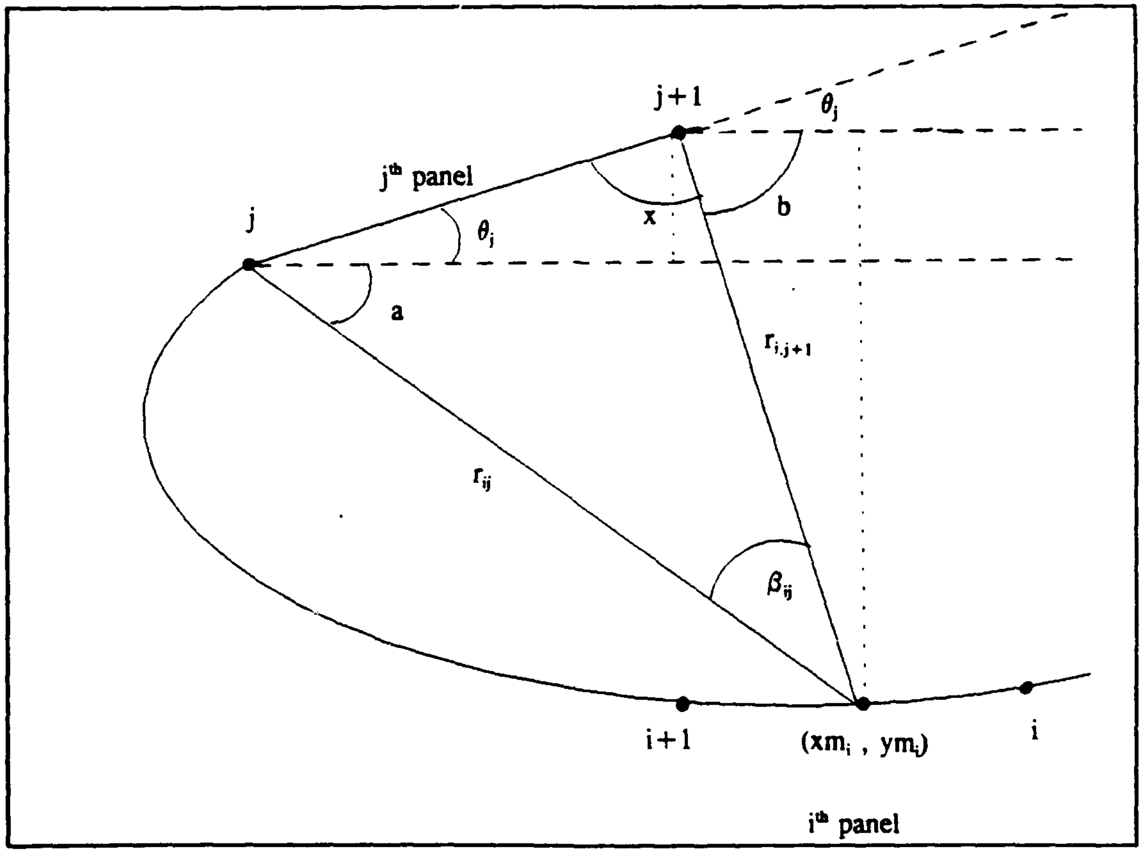

Figure 2.2 Relationships Between Geometrical Quantities

The formula for β_{ij} may be verified as follows:

$$a + \theta_j + x + \beta_{ij} = 180° \qquad \text{(triangle)}$$

$$x + b + \theta_j = 180° \qquad \text{(supplementary angles)}$$

Setting these equations equal to each other and eliminating common terms,

$$\beta_{ij} = b\text{-}a$$

Inspection of the diagram shows that angle b is in fact the arctangent of the quantity in parentheses in the first term of the formula for β_{ij}. Likewise, angle a matches the second term.

3. Program Description

a. Boundary conditions

The first boundary condition requires flow tangency at control points:

$$(V^n)_i = 0 \quad , \quad i = 1,2,...,n \tag{2.9}$$

In terms of influence coefficients (with $V_\infty = 1$),

$$\sum_{j=1}^{n} [A^n_{ij} q_j] + \gamma \sum_{j=1}^{n} B^n_{ij} + \sin(\alpha - \theta_i) = 0 \quad , \quad i = 1,2,...,n \tag{2.10}$$

The second boundary condition is the Kutta condition, which states that the pressures on the lower and upper panels at the trailing edge must be equal if the flow is to leave the trailing edge smoothly. Using a form of Bernoulli's equation,

$$C_p = \frac{p - p_\infty}{\frac{1}{2}\rho V_\infty^2} = 1 - \left(\frac{V_{total}}{V_\infty}\right)^2 \tag{2.11}$$

the pressure equilibrium also implies equal velocities for incompressible flow. Since the normal velocities are taken to be zero, the boundary condition may now be stated as:

$$(V^t)_1 = -(V^t)_n \tag{2.12}$$

where the negative sign is strictly due to the adopted convention of positive tangential velocities in the direction of increasing node numbering. Since the flow is positive to the right (as shown in Figure 2.1), the panels downstream of the front stagnation point will have negative values for computational purposes only. It is important to note that not all the lower surface panels have a reversed sign, *only those downstream from the stagnation point.* This is especially significant for non-symmetrical airfoils or any airfoil at an angle of attack.

In terms of influence coefficients, the normalized equation becomes:

$$\begin{aligned} -\sum_{j=1}^{n} [A^t_{1j} q_j] - \gamma \sum_{j=1}^{n} B^t_{1j} - \cos(\alpha - \theta_1) = \\ \sum_{j=1}^{n} [A^t_{nj} q_j] + \gamma \sum_{j=1}^{n} B^t_{nj} + \cos(\alpha - \theta_n) \end{aligned} \tag{2.13}$$

b. Solution procedure

Equations 2.10 and 2.13 represent a linear algebraic system of (n+1) equations and (n+1) unknowns. The unknowns are the source strengths which vary from panel to panel ($q_1 \ldots q_n$) and the vorticity strength γ.

Expanding and rearranging Equation 2.10 for an example airfoil of n=73 nodes and panels results in:

$$
\begin{aligned}
A^{n}_{11}q_1 + A^{n}_{12}q_2 + \ldots + \gamma(B^{n}_{11} + B^{n}_{12} + \ldots + B^{n}_{1,73}) &= -\sin(\alpha - \theta_1) \\
A^{n}_{21}q_1 + A^{n}_{22}q_2 + \ldots + \gamma(B^{n}_{21} + B^{n}_{22} + \ldots + B^{n}_{2,73}) &= -\sin(\alpha - \theta_2) \\
&\vdots \\
A^{n}_{73,1}q_1 + A^{n}_{73,2}q_2 + \ldots + \gamma(B^{n}_{73,1} + B^{n}_{73,2} + \ldots + B^{n}_{73,73}) &= -\sin(\alpha - \theta_{73})
\end{aligned}
\tag{2.14}
$$

The equations now readily lend themselves to solution in matrix form. Recasting with a simpler notation, the A^{n}_{ij} terms (coefficients of q_j) may be renamed a_{ij} and the sum of all B^{n}_{ij} terms in parentheses (coefficients of γ) renamed $a_{i,n+1}$, where $i=1,2,\ldots,n$ and $j=1,2,\ldots,n$. The terms on the right sides of the equations may be renamed b_i.

The $(n+1)^{st}$ equation, or in this example, the 74^{th} equation, comes from Equation 2.13 in a similar manner:

$$
\begin{gathered}
(A^{t}_{1,1} + A^{t}_{73,1})q_1 + (A^{t}_{1,2} + A^{t}_{73,2})q_2 + \ \ldots\ + (A^{t}_{1,73} + A^{t}_{73,73})q_{73} + \\
\gamma\,[(B^{t}_{1,1} + B^{t}_{73,1}) + (B^{t}_{1,2} + B^{t}_{73,2}) + \ \ldots\ + (B^{t}_{1,73} + B^{t}_{73,73})] = \\
\cos(\alpha - \theta_1) - \cos(\alpha - \theta_{73})
\end{gathered}
\tag{2.15}
$$

The coefficients of q_j may be renamed $a_{74,j}$. All of the B^{t} terms in the brackets together form the coefficient of γ, now renamed $a_{74,74}$. The entire right side of the equation constitutes the new term b_{74}.

Finally expressing this system in a concise matrix form for the general case,

$$\begin{bmatrix} a_{1,1} & a_{1,2} & a_{1,3} & \cdots\cdots & a_{1,n+1} \\ a_{2,1} & a_{2,2} & a_{2,3} & \cdots\cdots & a_{2,n+1} \\ a_{3,1} & a_{3,1} & a_{3,3} & \cdots\cdots & a_{3,n+1} \\ \vdots & \vdots & \vdots & & \vdots \\ a_{n,1} & a_{n,2} & a_{n,3} & \cdots\cdots & a_{n,n+1} \\ a_{n+1,1} & a_{n+1,2} & a_{n+1,3} & \cdots\cdots & a_{n+1,n+1} \end{bmatrix} \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ \vdots \\ q_n \\ \gamma \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ \vdots \\ b_n \\ b_{n+1} \end{bmatrix} \tag{2.16}$$

This system is solved in the program using a Gaussian Elimination subroutine.

With the values of the q_j and γ known, the velocity at each panel control point may be calculated:

$$V_i = \sum_{j=1}^{n} [A^t_{ij} q_j] + \cos(\alpha - \theta_i) \quad , \quad i = 1,2,\ldots,n \tag{2.17}$$

The total velocity is equal to the tangential velocity due to taking the normal velocity to be zero.

c. *Numerical techniques*

Although programming most of the described procedure is a relatively straightforward task, there are a number of potential pitfalls. Different programming languages each have their own special rules and format, but the following hints for FORTRAN, a commonly used engineering language, apply to many others as well.

All angles entered into the equations, such as α, β, and θ, must have values in radians. The angle of attack (α) is an input parameter that should also be converted relative to the zero lift line of the airfoil in order for the resulting velocities to match published values. The angle θ is used in calculating the influence coefficients. θ is the angle of a panel from the positive x axis, counter-clockwise positive. The formula given for θ, involving taking an arctangent, will produce the correct physical angle if interpreted correctly. Many programming languages use $-\pi/2$ to $\pi/2$ as the default range for the standard inverse tangent function, which uses only one value for an argument. The function cannot determine whether a negative sign was in the numerator or denominator. A problem arises when an angle is actually in the 2nd or 3rd quadrant because the function will assign values from the 1st or 4th quadrant. The values for θ must be in the range $-\pi < \theta < \pi$ to work properly. This may be accomplished by using the ATAN2 function in FORTRAN, which accepts both a numerator and denominator as arguments and assigns quadrants correctly.

The angle β may be calculated from two inverse tangents, as presented in the formula. However, a more efficient algorithm may be used for computer calculation. Recalling that β = b-a, it follows that:

$$\begin{aligned} \tan \beta &= \tan(b-a) \\ &= \frac{\sin(b-a)}{\cos(b-a)} \\ &= \frac{\sin b \cos a - \cos b \sin a}{\cos b \cos a + \sin b \sin a} \end{aligned} \qquad \textbf{(2.18)}$$

The sines and cosines for the angles a and b may be easily determined from the geometry of Figure 2.2. For example, sin b = $(ym_i - y_{j+1})$/hypotenuse b. Noting that all the hypotenuse values may be cancelled out in Equation 2.18, the arctangent of the angle may be found with simply x and y differences. The ATAN2 function in FORTRAN should also be used to calculate β.

D. RESULTS

1. Eppler E585 Airfoil

The first investigation was conducted for an Eppler E585 airfoil (shown in Figure 2.3), with n=71 nodes (or panels). It is one of a series of airfoils without flaps designed for the Reynolds number range of sailplanes, about 100,000 to 500,000. The angle relative to the zero lift line is 5.53°. If not compensated for, the results would all be shifted by this amount. Figure 2.4 shows the results of the program **panel** for α = 3, 7, 11°. The published data in Figure 2.5 [Ref. 1] compares favorably. The velocities match the given distribution well except for slight deviations at the trailing edge. This difference can be attributed to the higher order panel method used in Eppler's computations. The panels are defined by third degree polynomials whose coefficients are computed by a spline program. Additional points are splined in as needed. For the best precision from a panel method, the steps must be smaller near the leading and trailing edges. The simpler method of connecting

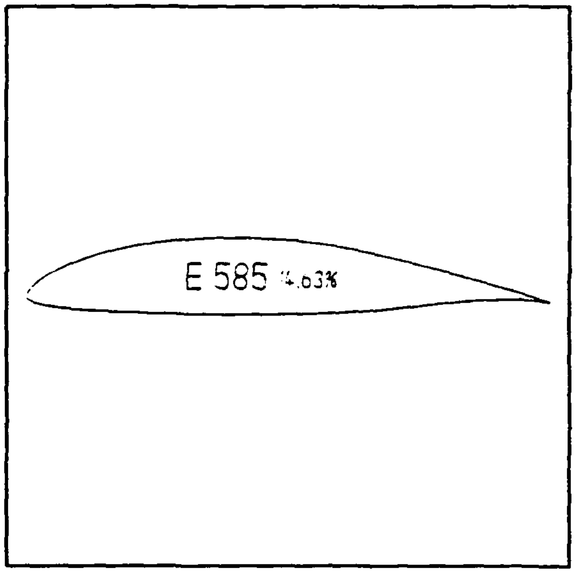

Figure 2.3 Eppler E585 Airfoil

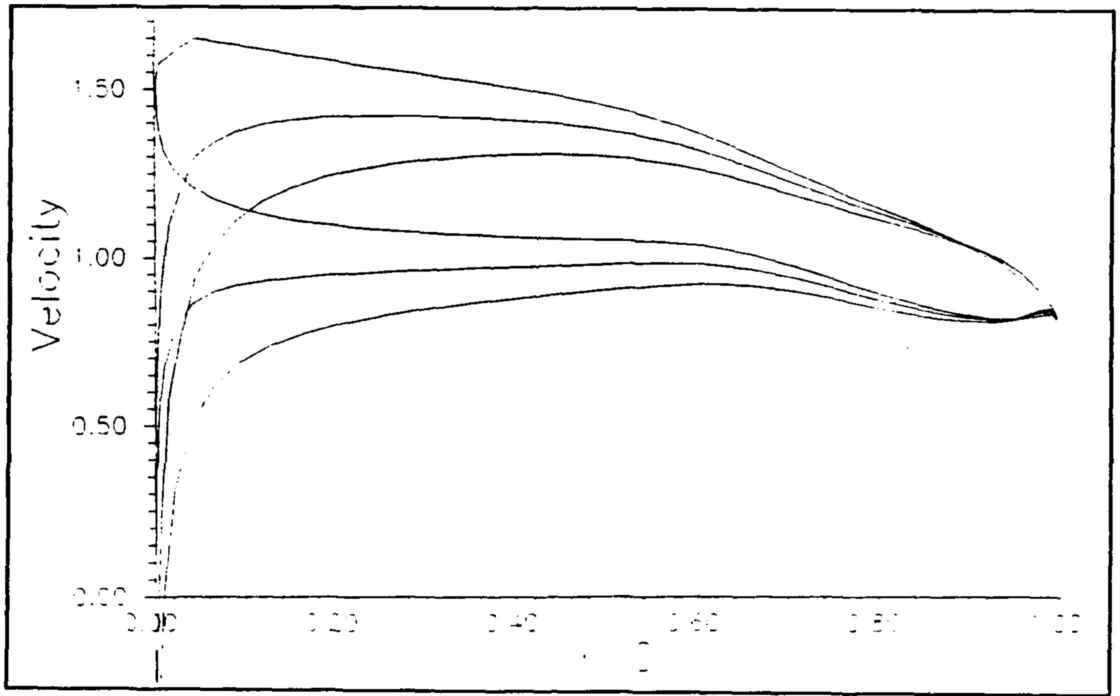

Figure 2.4 Velocity Distribution Computed by PANEL

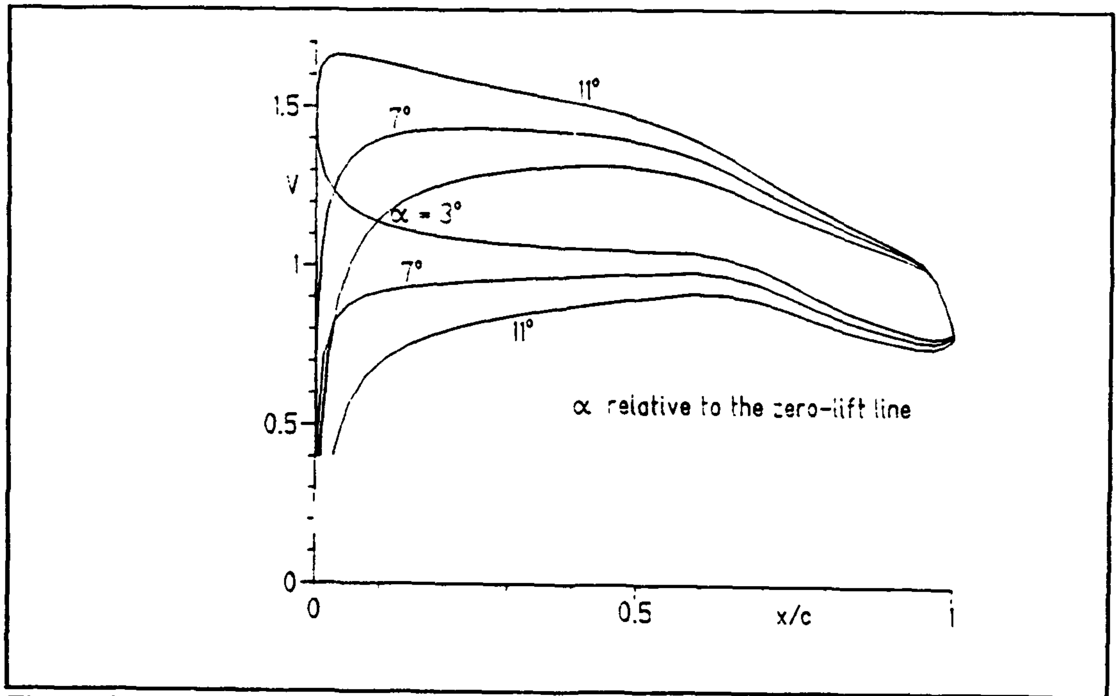

Figure 2.5 Velocity Distribution Computed by Eppler

just the given airfoil coordinates naturally leads to less accurate results in the critical trailing edge area. Experimentation with linearly interpolated extra points verified that a better resolution could be obtained in this manner. For most cases, the straight line segment panels using only the given coordinates produce quite sufficient resolution with a great advantage in computational speed.

2. NACA 0012 Airfoil

This is a well known airfoil (shown in Figure 2.6) used for many studies and comparisons, elementary to advanced, as well as practical applications. It is one of the original 4-digit series of 1932, where the first two digits indicate camber amount and location (00 is symmetrical) and the last two digits indicate maximum thickness in percent mean aerodynamic chord (12%). Figures 2.7 and 2.8 show a comparison of program results and those provided by Anderson [Ref. 2] for the NACA 0012 airfoil at 9° angle of attack. The pressure distributions both come to a suction peak of -5.2. It should be noted that Figure 2.8 differs slightly from the original reference plot in that the error in the decimal place of the ordinate values has been corrected.

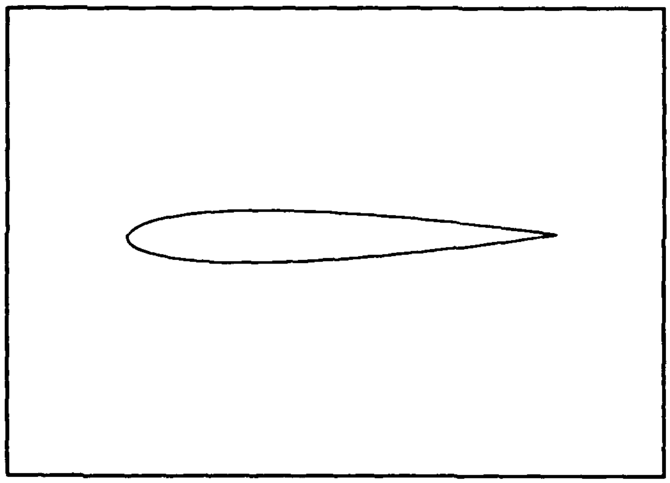

Figure 2.6 NACA 0012 Airfoil

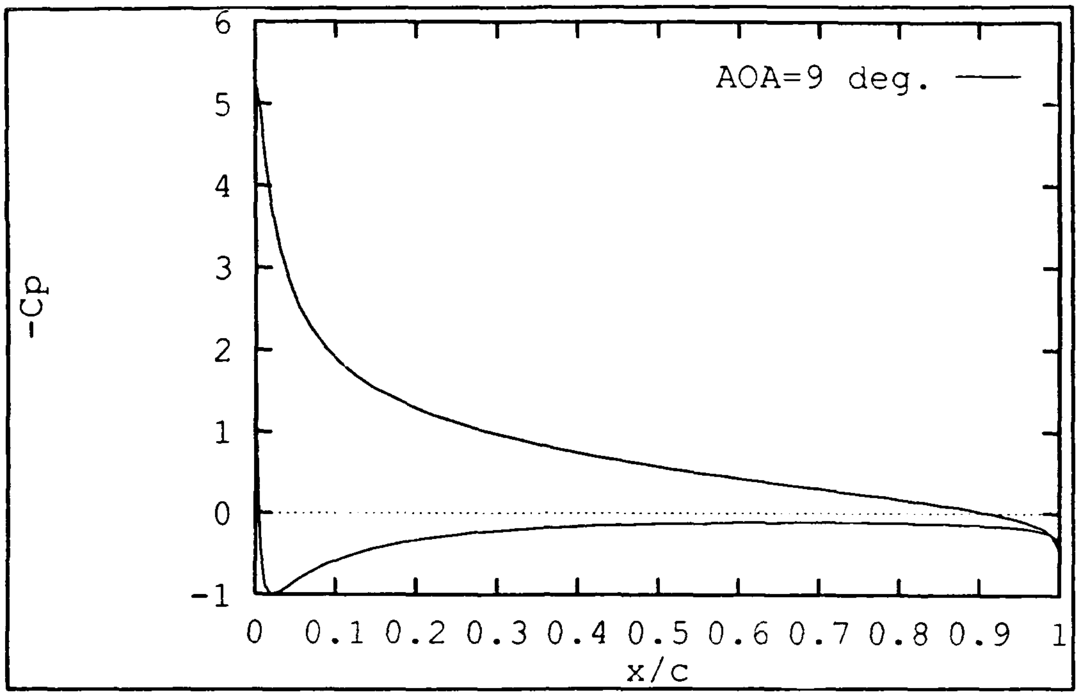

Figure 2.7 Pressure Coefficient Computed by PANEL

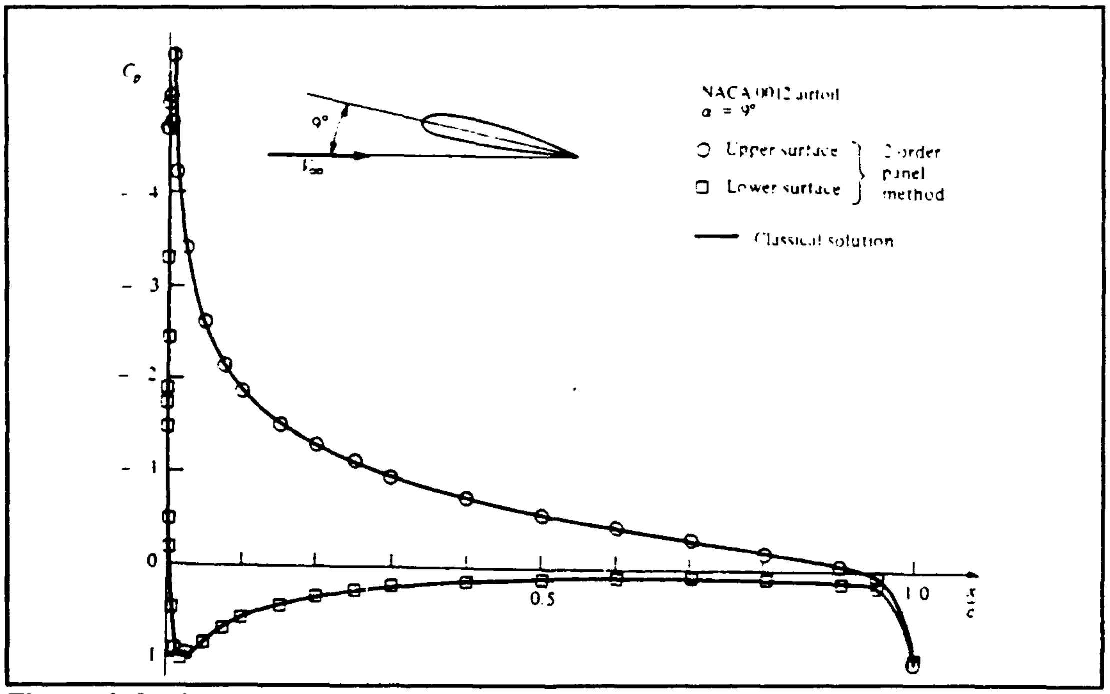

Figure 2.8 Pressure Coefficient Provided by Anderson

E. USER'S GUIDE

These detailed instructions are accurate at the time of writing, oriented toward the Advanced Computation Laboratory (rm. 136) of the Naval Postgraduate School. However, due to the dynamic nature of any computer environment, some instructions may change over a period of time. If difficulty is encountered, check with the manager for changes in:

1. the account location of the files
2. plotting package availability or procedures
3. computer informal addresses (i.e. madmax, suzqt, indigo1, etc.)

In all instruction lines, the prompt which appears on the screen is in normal type, while literal user input is in **boldface**. User input which is general and requires the actual word instead will be in *italics*. For example, *filename* could be replaced by **vel.dat**. Although not specifically stated on each line, a carriage return (ENTER) is assumed. As many commands are case sensitive, type each line in the format shown.

1. Stardent

a. Program operation

Using the Stardent terminal, login and change to the directory to be used for the panel code, making a new one if desired (i.e. **mkdir paneldir**, **cd paneldir**). Then type:

>**cp /alpha/*acctname*/panel panel**

to copy the compiled program from the account where it is stored. Also copy a sample input file for a NACA 0012 airfoil:

> **>cp /alpha/*acctname*/points.dat points.dat**

These two files are all that is needed to begin. To run the program, type:

> **>panel**

Answer the prompted questions, using 100 points and 0 for the transition code. When finished, the program will respond with:

> Calculations complete, output files are:
>
> vel.dat, cp.dat, bl2d.dat

The first two files are simply x-y type column data for plotting the velocity and pressure distributions. The last file is created for use with another program, **bl2d**, described in the next chapter.

b. Plotting procedures

To view the resulting velocity distribution, the file vel.dat can be used wi... any standard plotting package. On the Stardent, **gnuplot** can be used. First type:

> **>xterm -t &**

to open up a tektronics window for plotting that will operate in background. When it appears, move the cursor there and type:

> **>gnuplot**

Note that it is somewhat "messy" to work in this window because mistakes do not disappear from the screen with a backspace. The computer will take the overwritten characters as the input, however. Control-C may be pressed instead to simply type the line over or type **clear** to erase the whole screen when needed. At the gnuplot prompt, type

gnuplot>**set term tek40xx**

to set the proper terminal type for plotting. To plot the x-y format data file vel.dat, type:

gnuplot>**plot "vel.dat"**

For options in gnuplot such as scaling axes and plotting multiple data sets, see Appendix B.

2. IRIS

a. Using a Stardent window

To simply operate the program **panel** from an IRIS terminal, login and open up a Stardent window using the following procedures. First, click the right mouse button inside the original console window and select **Clone**. Move the cursor to the new window and change to the desired directory on the IRIS, making a new one (**mkdir**) if desired. Then type:

>**telnet suzqt**

Login to the Stardent and follow the instructions in the previous section to operate the program. Do not use the same plotting procedures when complete, however. Keeping both windows open, move the cursor back to the IRIS window and type:

>**rcp suzqt:/alpha/*loginname/directory*/vel.dat vel.dat**

This command remote copies from one system to the other. The *loginname* on the command line is the name of the account, usually the user's last name. The *directory* is the one created on the Stardent where the program was run. This command will only work if an account is held on both the Stardent and IRIS under the same *loginname*. If this is not the case, files can be transferred using the file transfer utility **ftp**, described in Appendix B.

b. Using the IRIS

It is also possible to do all calculations directly on the IRIS. This may be more useful when a user holds an account on the IRIS but not on the Stardent. After obtaining a copy of the source code **panel.f** from the Stardent using the **ftp** procedures (Appendix B), compile it for the IRIS:

>f77 -O3 -o panel panel.f

Program operation is as described for the Stardent.

c. Plotting procedures

The plotting package usually used on the IRIS is XYPLOT. At the prompt, from the directory with the plotting data (such as vel.dat) type:

>xyplot

Answer the questions that follow:

Name of 1st input file? **vel.dat**

Name of 2nd input file? *(press ENTER since only 1 file to plot)*

Default configuration file? *(ENTER, since none specified yet)*

A new window will pop up with the plot. The mouse can be used in this window to change the appearance of the plot in many ways. The plotting program is very user-friendly and can be operated with little prior instruction.

Gnuplot is also available on the NPS IRIS. It is not necessary to open a special window for the plot because one will be created automatically when the program is invoked.

III. DIRECT BOUNDARY LAYER CODE

A. THEORY

The two-dimensional flow around an airfoil may be represented by the simultaneous solution of the continuity equation,

$$\frac{\partial \rho}{\partial t}+\frac{\partial \rho u}{\partial x}+\frac{\partial \rho v}{\partial y}=0 \tag{3.1}$$

the components of the momentum equation,

$$\frac{\partial \rho u}{\partial t}+\frac{\partial \rho u^2}{\partial x}+\frac{\partial \rho uv}{\partial y}=-\frac{\partial p}{\partial x}+\frac{\partial \sigma_{xx}}{\partial x}+\frac{\partial \sigma_{xy}}{\partial y}+\rho f_x \tag{3.2}$$

$$\frac{\partial \rho v}{\partial t}+\frac{\partial \rho uv}{\partial x}+\frac{\partial \rho v^2}{\partial y}=-\frac{\partial p}{\partial x}+\frac{\partial \sigma_{xy}}{\partial x}+\frac{\partial \sigma_{yy}}{\partial y}+\rho f_y \tag{3.3}$$

and the energy equation, often collectively referred to as the Navier-Stokes equations, although technically this name applies only to the momentum equation applied to a Newtonian fluid. Equations 3.1-3.3 are expressed in general form for unsteady, compressible, viscous flows with body forces. In practice, however, such a complete solution is not usually feasible, or even possible, for many applications. The development of the Thin Shear Layer (TSL) equations, sometimes referred to as the boundary layer equations, enables a computationally practical scheme for solving the flow.

Considering the steady, incompressible, flow around an airfoil with no body forces, some initial simplifications may be made. The energy equation becomes decoupled from the other equations and is no longer needed in the solution. The continuity equation immediately reduces to

$$\frac{\partial u}{\partial x}+\frac{\partial v}{\partial y}=0 \tag{3.4}$$

Eliminating the time dependent term, expanding the partial derivatives, subtracting u times the continuity equation, and dividing through by ρ in the x momentum equation yields

$$u\frac{\partial u}{\partial x}+v\frac{\partial u}{\partial y}=-\frac{1}{\rho}\frac{\partial p}{\partial x}+\frac{1}{\rho}\frac{\partial \sigma_{xx}}{\partial x}+\frac{1}{\rho}\frac{\partial \sigma_{xy}}{\partial y} \tag{3.5}$$

with a similar expression for the y component. The equation is further developed by using the assumption of a Newtonian fluid, in which stress is proportional to rate of strain:

$$\sigma_{xx}=2\mu\left(\frac{\partial u}{\partial x}\right) \tag{3.6}$$

$$\sigma_{xy}=\mu\left(\frac{\partial u}{\partial y}+\frac{\partial v}{\partial x}\right) \tag{3.7}$$

where μ is the viscosity. In a constant-property flow, μ may be taken outside the derivative after substituting Equations 3.6 and 3.7 into Equation 3.5, and may be rewritten in terms of the kinematic viscosity, $\nu \equiv \mu/\rho$. Thus, the x-component momentum equation for a Newtonian fluid in constant-property, steady, two-dimensional, laminar flow is

$$u\frac{\partial u}{\partial x}+v\frac{\partial u}{\partial y}=-\frac{1}{\rho}\frac{\partial p}{\partial x}+\nu\left(\frac{\partial^2 u}{\partial x^2}+\frac{\partial^2 u}{\partial y^2}\right) \tag{3.8}$$

To incorporate the effects of a turbulent flow, all instantaneous flow quantities are replaced by a mean term plus a fluctuating part, i.e. $u=\overline{u}+u'$. Expanding, eliminating zero products, and rearranging the equation gives

$$u\frac{\partial u}{\partial x}+v\frac{\partial u}{\partial y}=-\frac{1}{\rho}\frac{\partial p}{\partial x}+\nu\left(\frac{\partial^2 u}{\partial x^2}+\frac{\partial^2 u}{\partial y^2}\right)-\frac{\partial \overline{u'^2}}{\partial x}-\frac{\partial \overline{u'v'}}{\partial y} \tag{3.9}$$

where the overbars on the mean velocity components u and v are omitted for simplicity. The extra turbulent stress terms in Equation 3.9, as compared to Equation 3.8, are often called the Reynolds stresses.

To obtain the TSL equations, an order of magnitude analysis is applied with the assumption of $\delta/l \ll 1$. In other words, the boundary layer (of thickness δ) is very small compared to the characteristic length of the body. Using the following order of magnitude approximations

$$u \sim u_e \qquad \frac{\partial u}{\partial y}\sim\frac{u_e}{\delta} \qquad \frac{\partial u}{\partial x}\sim\frac{u_e}{l} \tag{3.10}$$

where the subscript e refers to the edge of the boundary layer, the first term in the parentheses and the first Reynolds stress in Equation 3.9 may be neglected. When compared term by term to the x-component equation with the assumption $u \gg v$, the y-component of the momentum equation reduces to the approximation of constant pressure in the normal direction. Summarizing, the two-dimensional, incompressible, steady boundary-layer equations for both laminar and turbulent flows are:

$$\frac{\partial u}{\partial x}+\frac{\partial v}{\partial y}=0 \tag{3.11}$$

$$u\frac{\partial u}{\partial x}+v\frac{\partial u}{\partial y}=-\frac{1}{\rho}\frac{\partial p}{\partial x}+\nu\frac{\partial^2 u}{\partial y^2}-\frac{\partial \overline{u'v'}}{\partial y} \tag{3.12}$$

$$\frac{\partial p}{\partial y}=0 \tag{3.13}$$

Note that these equations are for a surface coincident with the x-axis. In order to use the equations for an airfoil, the airfoil surface must be "unwrapped" onto the x-axis. The usual x/c and y/c coordinates which define the airfoil must be transformed to a surface coordinate.

The applicable boundary conditions on the surface of a solid airfoil are

$$y=0 \qquad u=0 \qquad v=0$$

and at the outer edge of the boundary layer,

$$y=\delta \qquad u=u_e(x)$$

B. COMPUTER CODE: BL2D

1. Overview

The program **bl2d**, developed by T. Cebeci, provides a solution to the boundary layer equations. The same restrictions apply to the program as to the equations on which it is based: two-dimensional, steady, incompressible, viscous flow. The program accepts input of Reynolds number and prescribed transition locations, as well as panel coordinate and velocity information computed by a separate routine. Output is generated for many features of the resulting boundary layer, including skin friction coefficient and displacement thickness. Run time is less than one minute on a Stardent computer and less than two minutes on a personal computer (PC).

2. Models

a. Turbulence Model

In order to use equation 3.12, an expression must be found for the Reynolds shear stress term. Since it is not feasible to attempt calculating the actual value, empirical models are usually used. One such model is the eddy-viscosity concept:

$$-\rho\overline{u'v'} = \rho\epsilon_m \frac{\partial u}{\partial y} \tag{3.14}$$

where ϵ_m is an empirical term called the turbulent eddy viscosity. Another model is the mixing-length concept first proposed by Prandtl in 1925:

$$-\rho\overline{u'v'} = \rho l^2 \left|\frac{\partial u}{\partial y}\right| \frac{\partial u}{\partial y} \qquad (3.15)$$

where l is the empirically determined mixing length. Although both of these models were originally derived based on erroneous physical arguments, they have nevertheless produced remarkably successful results for many applications. A third model, which incorporates the eddy-viscosity concept, is the Cebeci-Smith (CS) model, in which the viscous region is divided into an inner layer and an outer layer, each with its own formula for ϵ_m:

$$\begin{aligned} \left(\frac{\epsilon_m}{\nu}\right)_i &= 0.16\sqrt{Re_x}[1-e^{-(\frac{y}{A})}]^2\eta^2 v\gamma_{tr} \\ \left(\frac{\epsilon_m}{\nu}\right)_o &= 0.0168\sqrt{Re_x}[\eta_e - f_e]\gamma_{tr} \end{aligned} \qquad (3.16)$$

where

$$\begin{aligned} Re_x &= \frac{u_e}{u_\infty}\xi R_L \\ \frac{y}{A} &= \frac{1}{26}\sqrt[4]{Re_x}\sqrt{v_w}\eta \\ \gamma_{tr} &= 1-\exp[-G(x-x_{tr})\int_{x_{tr}}^{x}\frac{1}{u_e}dx] \\ G &= \frac{1}{1200}\left(\frac{u_e}{u_\infty}\right)^3 R_L^2 Re_{x_{tr}}^{-1.34} \end{aligned} \qquad (3.17)$$

and the Falkner-Skan variables ξ, η, and f are used. The term γ_{tr} is a factor which models the length of the transition region, explained further in the next section. This turbulence model is used in the program. Using Equation 3.14 allows rewriting the momentum equation for a turbulent flow, Equation 3.12, in the same form as a laminar flow:

$$u\frac{\partial u}{\partial x}+v\frac{\partial u}{\partial y}=-\frac{1}{\rho}\frac{\partial p}{\partial x}+\frac{\partial}{\partial y}(b\frac{\partial u}{\partial y}) \tag{3.18}$$

where $b=\nu+\epsilon_m$. Thus, the only computational difference between laminar and turbulent boundary layers is the addition of the turbulent eddy viscosity.

b. Transition Model

The determination of the location of transition from laminar to turbulent flow is one of the most critical factors in the success of many computational efforts to predict or reproduce physical phenomena. Yet, even in today's age of supercomputers, this area of research remains widely neglected. The most advanced Navier-Stokes solvers often ignore the issue entirely, arbitrarily declaring the entire flow to be turbulent. Others make the effort of computing a transition point, at which the flow instantaneously changes from laminar to turbulent. These may be quite reasonable approximations for many applications, especially when the Reynolds number is high. However, there are still a number of important flows that require more accuracy. Until the transition mechanism and the many, varied factors which can affect it are more fully understood, programmers must rely on the traditional engineering approach of modeling.

The program incorporates a transition model determined by Chen and Thyson, utilizing a transition range rather than a point. This range is neither fully laminar nor fully turbulent. It is a region of *intermittency*, in which turbulent spots gradually appear with progression in the streamwise direction. This feature has been shown to be essential for low Reynolds number flows. The convergence of the entire boundary layer solution is very sensitive to transition related factors, such as the input value supplied for the start of transition. An experimental value may not have been measured accurately and an empirically calculated value may deviate from the actual onset of transition. If the code does not run and all other input has been verified to be correct, a solution may often be obtained by experimenting with slight deviations in the transition location for the upper surface specified as input. The lower surface value does not exhibit the same sensitivity.

3. Numerical Techniques

a. Transformation of Airfoil Coordinates

The x/c and y/c coordinates of the airfoil are supplied as part of the input, called xc and yc in the program. Starting from the stagnation point, the program redefines these coordinates into a single parameter corresponding to a surface distance:

$$x_i = x_{i-1} + \sqrt{(xc_i - xc_{i-1})^2 + (yc_i - yc_{i-1})^2} \tag{3.19}$$

Thus, the variable x used internally by the program in the boundary layer equations is this surface coordinate. The values are printed under the heading (S) in the output.

b. Transformation of Variables

The well known Falkner-Skan transformation is used to transform the variable y:

$$\eta = \sqrt{\frac{u_e}{\nu x}}\, y \tag{3.20}$$

where y is the normal coordinate along which the thickness of the boundary layer is measured. The dimensionless similarity variable η eliminates the growth of the boundary layer in laminar flow and reduces it in turbulent flow. This enables larger steps in the streamwise direction and improves computational efficiency. The x transformation is simply a scaling by the reference length, usually the chord for a airfoil, so that $\xi = x/c$. Since the surface distance x is used, these will not be the same as the input x/c coordinates.

The dimensionless stream function $f(x,\eta)$ is defined by

$$f(x,\eta) = \frac{\Psi}{\sqrt{u_e \nu x}} \tag{3.21}$$

Equations 3.11 and 3.18 and the boundary conditions may be rewritten in terms of the new variables:

$$\begin{gathered}(bf'')' + \frac{m+1}{2} f f'' + m[1-(f')^2 = \xi \left(f' \frac{\partial f'}{\partial \xi} - f'' \frac{\partial f}{\partial \xi}\right) \\ \eta = 0 \qquad f' = 0 \qquad f(\xi,0) \equiv f_w(\xi) = -\frac{\sqrt{R_L}}{\sqrt{\frac{u_e}{u_\infty}\xi}} \int_0^\xi \frac{v_w}{u_\infty} d\xi \\ \eta = \eta_e \qquad f' = 1\end{gathered} \tag{3.22}$$

where η_e is the transformed boundary layer thickness corresponding to δ and R_L is the Reynolds number based on reference velocity u_∞ and reference length, the chord for an airfoil. The prime denotes differentiation with respect to η. The dimensionless pressure gradient parameter m is defined by

$$m = \frac{\xi}{u_e/u_\infty}\frac{d(u_e/u_\infty)}{d\xi} \tag{3.23}$$

The velocity components v and v are related to the dimensionless stream function by

$$u = u_e f' \qquad v = -\sqrt{u_e \nu x}\left[\frac{f}{\sqrt{u_e x}}\frac{d}{dx}\sqrt{u_e x} + \frac{\partial f}{\partial x} + f'\frac{\partial \eta}{\partial x}\right] \tag{3.24}$$

c. *Keller Box Method*

Equation 3.22, a second-order partial differential equation, may be solved by various numerical methods such as the Crank-Nicholson or Keller Box methods. The latter method, depicted in Figure 3.1, has proven to be efficient for boundary layer calculations. The Keller Box method first requires reformulating higher

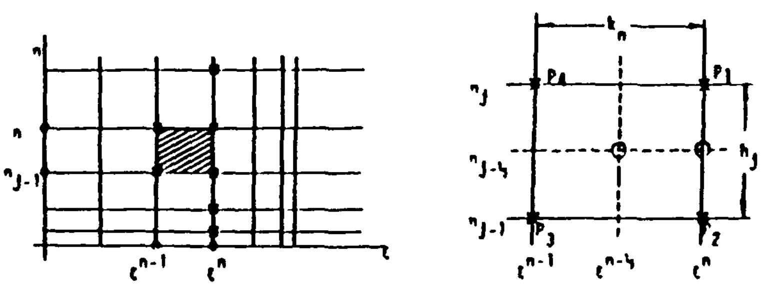

Figure 3.1 Grid Box for Centered-Difference Approximations

order equations into a set of first order equations. At each rectangular grid section, these equations are approximated using centered-difference derivatives, averaging values at the center of the "box". The truncation error is second order. The resulting implicit, nonlinear difference equations are linearized by Newton's method and solved by a block elimination method.

Using the following definitions,

$$f' = u \qquad u' = v \tag{3.25}$$

Equation 3.22 may be expressed as a first order system:

$$\begin{gathered}(bv)' + \frac{(m+1)}{2} fv + m(1-u^2) = \xi \left(u\frac{\partial u}{\partial \xi} - v\frac{\partial f}{\partial \xi}\right) \\ \eta = 0 \quad u = 0 \quad f = f_w(x) \quad , \quad \eta = \eta_e \quad u = 1\end{gathered} \tag{3.26}$$

Note that the u and v in Equations 3.25 and 3.26 are not the velocity components. They are two new, arbitrarily selected names of variables for the expression of the first order system. Since the program was coded using these particular variable names, the present numerical discussion will use them for consistency.

Referring to Figure 3.1, the grid points may be described as

$$\begin{array}{lll} \xi^n = \xi^{n-1} + k^n & n = 1, 2, \ldots, N & \xi^0 = 0 \\ \eta_j = \eta_{j-1} + h_n & j = 1, 2, \ldots, J & \eta_0 = 0 \\ & & \eta_J = \eta_e \end{array} \tag{3.27}$$

where the capital letters N and J are the maximum number of grid points used in the streamwise and normal directions, respectively. The superscript n is not an exponent, but a counter just like the subscript j. This upper and lower notation allows using both counters on the same variable when needed. Considering one box of the grid, the finite difference approximations of Equation 3.25 may be written for midpoint of the right side, segment P_1P_2, using centered difference derivatives:

$$\frac{f_j^n - f_{j-1}^n}{h_j} = \frac{u_j^n + u_{j-1}^n}{2} \equiv u_{j-1/2}^n$$
$$\frac{u_j^n - u_{j-1}^n}{h_j} = \frac{v_j^n + v_{j-1}^n}{2} \equiv v_{j-1/2}^n \qquad (3.28)$$

Equation 3.26 may be approximated in the same manner for the midpoint of the box by centering first in one direction then the other. The resulting finite difference equation is

$$\frac{1}{h_j}(b_j^n v_j^n - b_{j-1}^n v_{j-1}^n) + \alpha_1 (fv)_{j-1/2}^n - \alpha_2 (u^2)_{j-1/2}^n + \alpha^n (v_{j-1/2}^{n-1} f_{j-1/2}^n - f_{j-1/2}^{n-1} v_{j-1/2}^n) = R_{j-1/2}^{n-1} \qquad (3.29)$$

where

$$R_{j-1/2}^{n-1} = -L_{j-1/2}^{n-1} + \alpha^n [(fv)_{j-1/2}^{n-1} - (u^2)_{j-1/2}^{n-1}] - m^n$$
$$L_{j-1/2}^{n-1} = \{\frac{1}{h_j}(b_j v_j - b_{j-1} v_{j-1}) + \frac{m+1}{2}(fv)_{j-1/2} + m[1 - (u^2)_{j-1/2}]\}^{n-1}$$
$$\alpha^n = \frac{\xi^{n-1/2}}{k^n} \qquad \alpha_1 = \frac{m^n + 1}{2} + \alpha^n \qquad \alpha_2 = m^n + \alpha^n$$

The boundary conditions of Equation 3.26 are rewritten at $\xi = \xi^n$ as

$$f_0^n = f_w \qquad u_0^n = 0 \qquad u_j^n = 1 \qquad (3.30)$$

d. Newton's Method

Equations 3.28, 3.29, and 3.30 comprise a set of 3J+3 equations and 3J+3 unknowns (f_j^n, u_j^n, v_j^n, where j = 0, 1, 2, ..., J), with f_j^{n-1}, u_j^{n-1}, and v_j^{n-1} known. Newton's method is applied to linearize this system. The method assumes that an approximate solution is known, either from the preceding iteration cycle or from the previous streamwise station. Then small unknown quantities are added to the approximate solution. Using the arbitrary iteration variable i (the superscript n omitted for clarity),

$$f_j^{i+1} = f_j^i + \delta f_j^i \qquad u_j^{i+1} = u_j^i + \delta u_j^i \qquad v_j^{i+1} = v_j^i + \delta v_j^i \tag{3.31}$$

with i=0 corresponding to known values at the previous streamwise station (ξ^{n-1}), these expressions may be substituted into Equations 3.28 and 3.29 for the unknowns. After dropping higher order terms of δ, a linear system of equations results:

$$\begin{gathered}
\delta f_j^i - \delta f_{j-1}^i - \frac{h_j}{2}(\delta u_j^i + \delta u_{j-1}^i) = (r_1)_j \\
\delta u_j^i - \delta u_{j-1}^i - \frac{h_j}{2}(\delta v_j^i + \delta v_{j-1}^i) = (r_3)_j \\
(s_1)_j \delta v_j^i + (s_2)_j \delta v_{j-1}^i + (s_3)_j \delta f_j^i + (s_4)_j \delta f_{j-1}^i + (s_5)_j \delta u_j^i + (s_6)_j \delta u_{j-1}^i = (r_2)_j
\end{gathered} \tag{3.32}$$

where the right hand sides are

$$\begin{gathered}
(r_1)_j = f_{j-1}^i - f_j^i + h_j u_{j-1/2}^i \\
(r_3)_j = u_{j-1}^i - u_j^i + h_j v_{j-1/2}^i \\
(r_2)_j = R_{j-1/2}^{n-1} - [\frac{1}{h_j}(b_j^i v_j^i - b_{j-1}^i v_{j-1}^i) + \alpha_1 (fv)_{j-1/2}^i - \alpha_2 (u^2)_{j-1/2}^i + \alpha^n (v_{j-1/2}^{n-1} f_{j-1/2}^i - f_{j-1/2}^{n-1} v_{j-1/2}^i)]
\end{gathered}$$

and the coefficients are

$$(s_1)_j = \frac{1}{h_j} b_j^i + \frac{\alpha_1}{2} f_j^i - \frac{\alpha^n}{2} f_{j-1/2}^{n-1}$$

$$(s_2)_j = \frac{1}{h_j} b_{j-1}^i + \frac{\alpha_1}{2} f_{j-1}^i - \frac{\alpha^n}{2} f_{j-1/2}^{n-1}$$

$$(s_3)_j = \frac{\alpha_1}{2} v_j^i + \frac{\alpha^n}{2} v_{j-1/2}^{n-1}$$

$$(s_4)_j = \frac{\alpha_1}{2} f_{j-1}^i - \frac{\alpha^n}{2} v_{j-1/2}^{n-1}$$

$$(s_5)_j = -\alpha_2 u_j^i$$

$$(s_6)_j = -\alpha_2 u_{j-1}^i$$

The boundary conditions of Equation 30 become

$$\delta f_0^i = 0 \qquad \delta u_0^i = 0 \qquad \delta u_j^i = 0 \tag{3.33}$$

These equations may be easily identified in the subroutine COEF of **bl2d**. Since they may be arranged into a block tridiagonal structure in matrix-vector form, the subroutine SOLVE uses the efficient block elimination method to solve for the small δ quantities. The iteration of Newton's method continues until the small quantities are small enough to be neglected.

4. Program Modification for Boundary Layer Profiles

At each station along the airfoil surface, the program calculates the u velocities for each value of η in the grid. Eta is the coordinate in the normal direction representing the transformed boundary layer thickness. In order to retrieve the physical boundary layer thickness, an inverse transformation is required:

$$y=\frac{\eta}{\sqrt{\frac{u_e}{\nu x}}} \tag{3.34}$$

Since the kinematic viscosity shows up only indirectly in the non-dimensional form of the Reynolds number, the actual equation used is

$$y=\eta\sqrt{\frac{x}{R_L u_e}} \tag{3.35}$$

where $R_L \equiv \rho u_e x/\mu$, or equivalently, $u_e x/\nu$, and x and u_e are used in the non-dimensional forms of x/c and u_e/u_∞. The value of x used here is the surface distance.

Plotting the shape of the actual velocity profile at a given station requires the station number (NX), the corresponding x coordinate, the u velocities, and the corresponding y values. All of these values are provided by the original program or Equation 3.35. Velocity profiles may be plotted at this point, but only the shape will be revealed. To visualize the growth of the boundary layer, the height of the boundary layer at each station is needed. This may be determined by finding where the u velocity has reached freestream velocity, indicating the edge of the boundary layer. Computationally, this is accomplished by allowing u to reach 0.995 of u_∞, the freestream velocity. Even though the remaining u values in the grid above this height will still be calculated by the program, no more values are written to the plotting output file.

5. Program Modification for Estimating Transition Location

The original program uses input values to specify the *onset* of transition. The transitional flow region is then calculated using the Chen-Thyson model, shown as γ_{tr} in Equation 3.17. In order to provide an initial estimate for the transition location when no other method of determination is available, a modification using Michel's criterion was incorporated:

$$R_{\theta_{tr}} = 1.174\left(1 + \frac{22400}{Re_{x_{tr}}}\right) Re_{x_{tr}}^{\ 0.46} \tag{3.36}$$

where $R_{\theta tr} \equiv u_e\theta/\nu$ is the Reynolds number based on momentum thickness at transition and Re_{xtr} is the Reynolds number based on the transition location.

C. RESULTS

Studies were conducted to:

- validate the program by comparing to known data
- determine the effect of prescribed onset of transition
- investigate the possible occurrence of zero or negative skin friction to indicate the start of a laminar separation bubble before breakdown of the code
- analyze boundary layer velocity profiles on the airfoil upper surface
- evaluate the effects of changes in Reynolds number
- assess the validity of obtaining an unsteady boundary layer solution by extracting steady velocities from unsteady pressure distributions

All studies presented are for a NACA 0012 airfoil defined by 100 points. The Reynolds number is 540,000, except where noted in the validation study and the Reynolds number effect study.

1. Program Validation

In order to ascertain that the results of the program could be considered reliable, an initial test case was run to compare with published data [Ref. 3]. The test conditions were an AOA of 0° at a Reynolds number of 6,000,000. The first results obtained were in the expected range but the curves were not smooth. The problem was traced to a very small discrepancy in the original airfoil coordinate input file provided with the program which was not noticeable when the file was checked by plotting. After generating new airfoil coordinates and running them through **panel** to get new velocities, smooth boundary layer results were obtained. Figure 3.2 shows the computed skin friction coefficient and displacement thickness and Figure 3.3 shows the published results. The plots exhibit excellent agreement.

2. Transition Onset Location

In the next study, the convergence of **bl2d** at a lower Reynolds number of 540,000 was investigated. The input value for the location of the start of the transition range was found to be the most critical factor. Starting at an angle of attack of 0°, transition values obtained from **incompbl** (Chapter 4) were initially used. The program converged to a reasonable solution as determined by viewing plots of the various output files. At 2° however, the program would not converge using the estimated transition

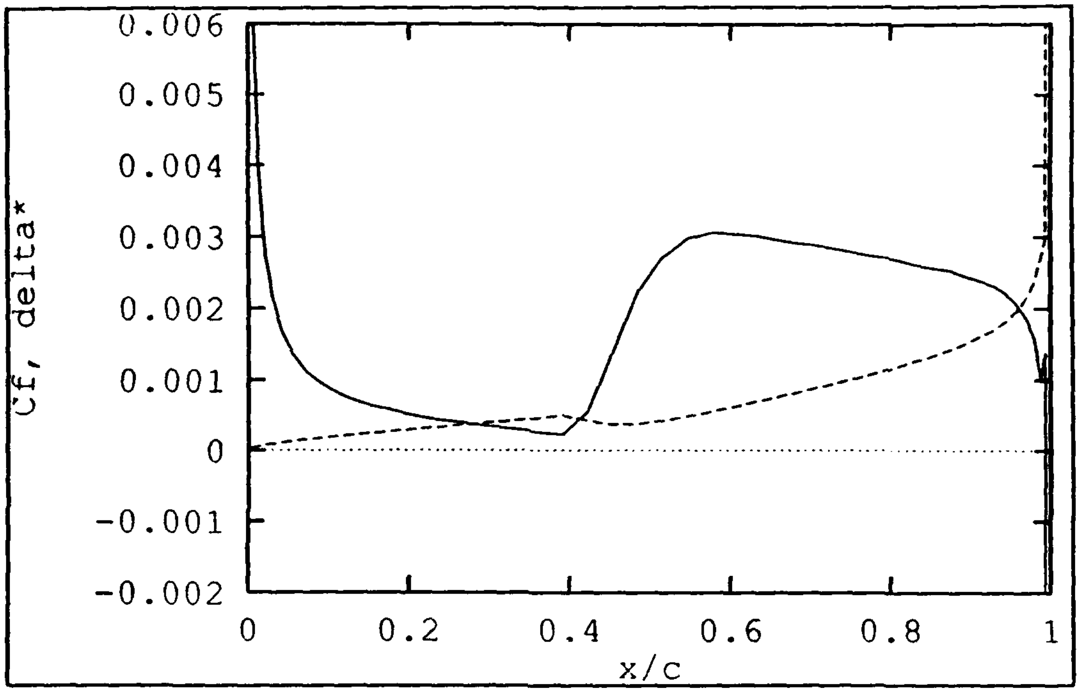

Figure 3.2 Computed C_f and δ^*, NACA 0012, AOA=0°, Re=6,000,000

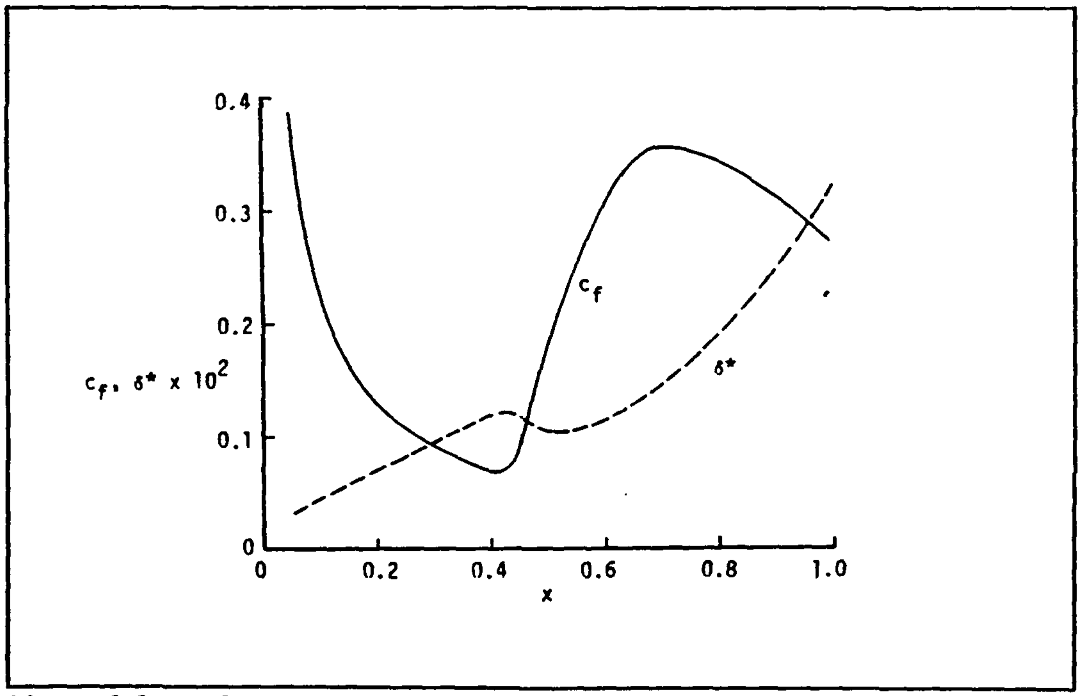

Figure 3.3 Reference C_f and δ^*, NACA 0012, AOA=0°, Re=6,000,000

value. Experimentation with this input parameter showed that moving it back usually made the convergence even worse, but in moving it forward, a point was reached where a solution could be obtained. Moreover, this solution was one that could be reasonably expected based on the previous solution at 0°. Similar experimentation was performed in AOA increments of 2° until excessive separation at high angle of attack caused the code to break down. The same study was also conducted later using the version of **bl2d** modified to make transition estimates. Table 3.1 shows a comparison between the values estimated by the two programs and the value actually needed for the program to converge.

Table 3.1 COMPARISON OF TRANSITION ONSET

AOA	INCOMPBL	BL2D	convergence	% diff
0°	.585	.597	either	0.0
2°	.453	.380	.380	0.0
4°	.334	.253	.306	17.3
6°	.0642	.0703	.055	27.8
8°	.0548	.0457	.045	1.6
10°	.0381	.0471	.042	12.1
12°	.0305	---	---	---

The first significant point to note is that the transition values produced by the modified version of **bl2d** are consistently close to the values produced by **incompbl**, showing that the criterion has, in fact, been programmed correctly. The differences can be attributed to several factors:

- slightly different values are input into the criterion equation for each program
- **incompbl** outputs the x/c value of the nodal point nearest the calculated transition onset, as opposed to the actual value
- if the onset of transition is located inside a separation bubble by the initial calculation, **incompbl** arbitrarily moves it to the start of the bubble

The third column shows how far forward the transition point was moved to obtain convergence, where the first AOA did not exhibit sensitivity. The last column shows the percent difference between the best estimate and the value required for convergence. Most were fairly close, with even the worst case less than 30% forward of the first estimate. This sets a reasonable bound for necessary experimentation with the transition location.

3. Laminar Separation

a. Skin Friction Coefficient and Displacement Thickness

Figures 3.4-3.10 show the progression of skin friction coefficient (C_f) and displacement thickness (δ^*) as the AOA ranges from 0 to 12 degrees. The transition onset may be observed as the point where C_f reaches as minimum then dramatically increases, indicating the change from laminar to turbulent flow. The transition point moves forward as the angle of attack increases. The minimum value of C_f decreases with increasing angle of attack. When the C_f reaches zero, separation is indicated.

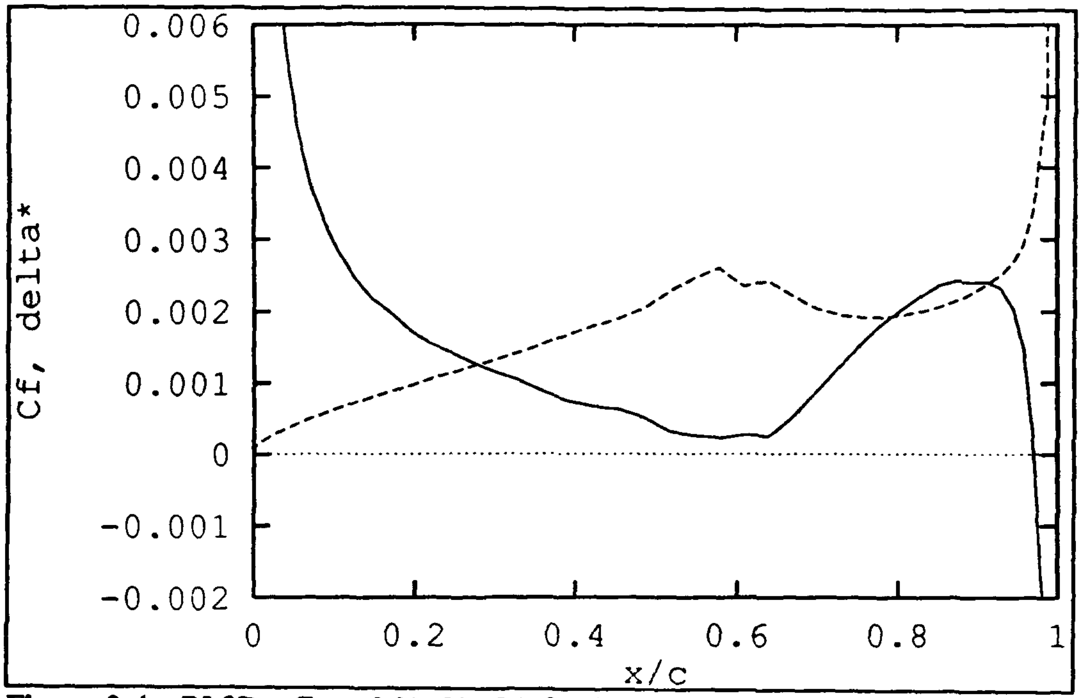

Figure 3.4 BL2D: C_f and δ^*, NACA 0012, AOA=0°, Re=540,000

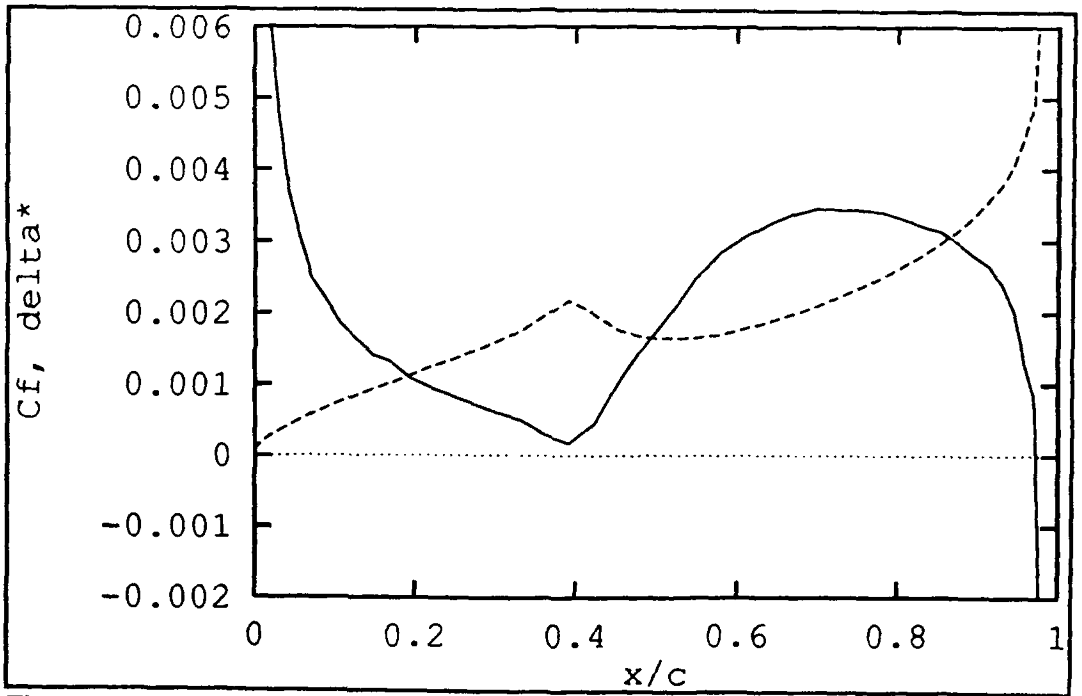

Figure 3.5 BL2D: C_f and δ^*, NACA 0012, AOA=2°, Re=540,000

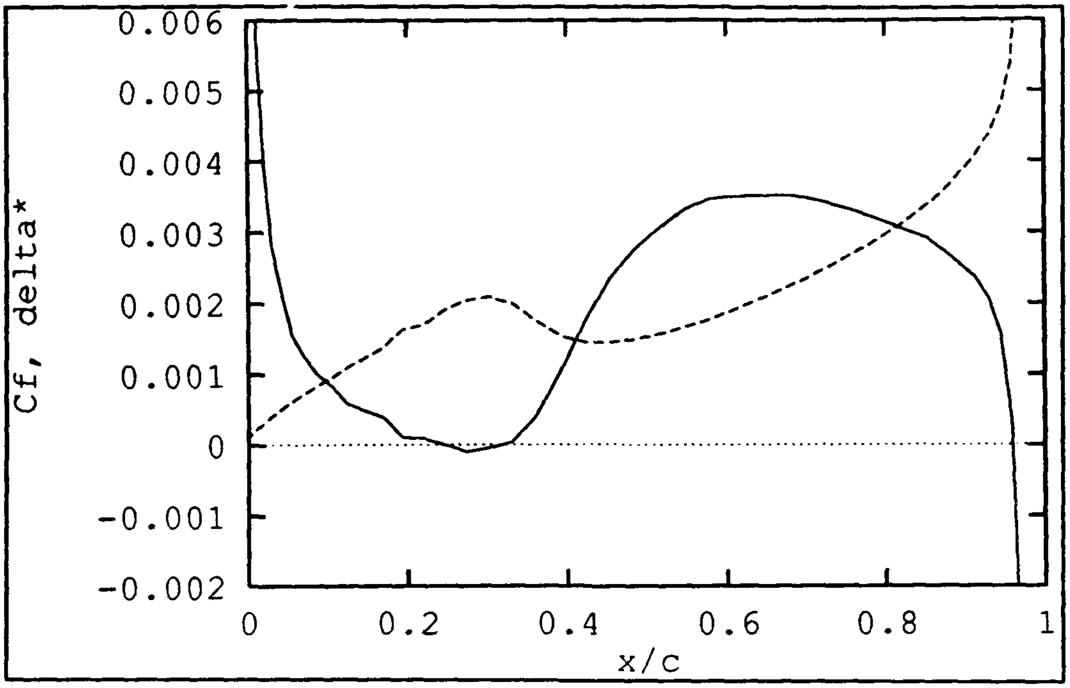

Figure 3.6 BL2D: C_f and δ^*, NACA 0012, AOA=4°, Re=540,000

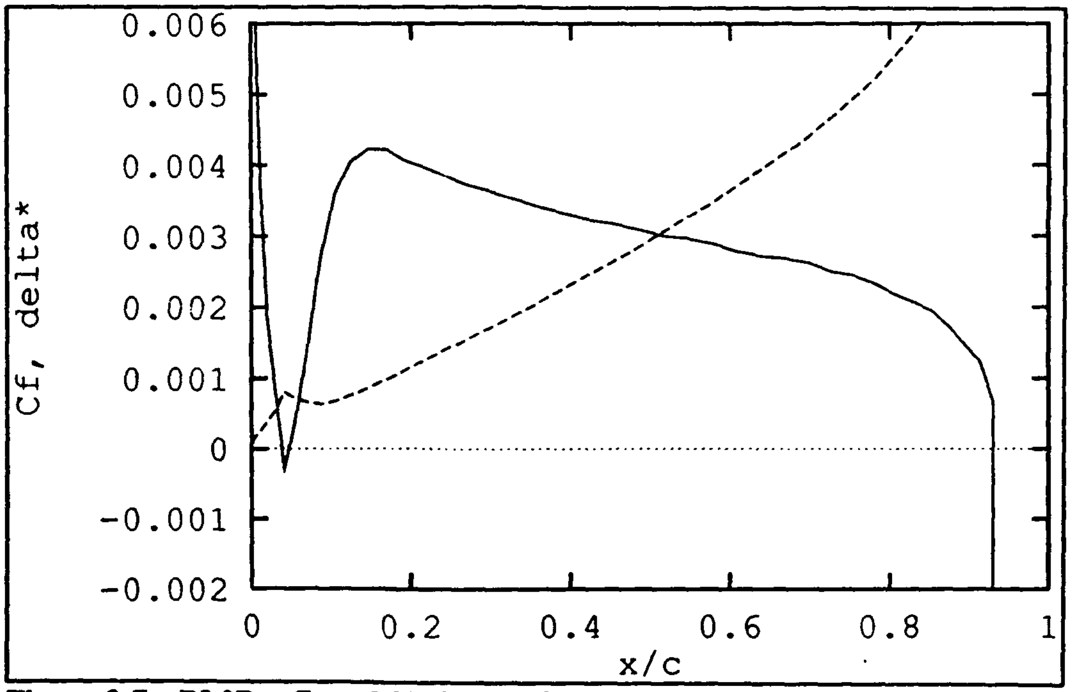

Figure 3.7 BL2D: C_f and δ^*, NACA 0012, AOA=6°, Re=540,000

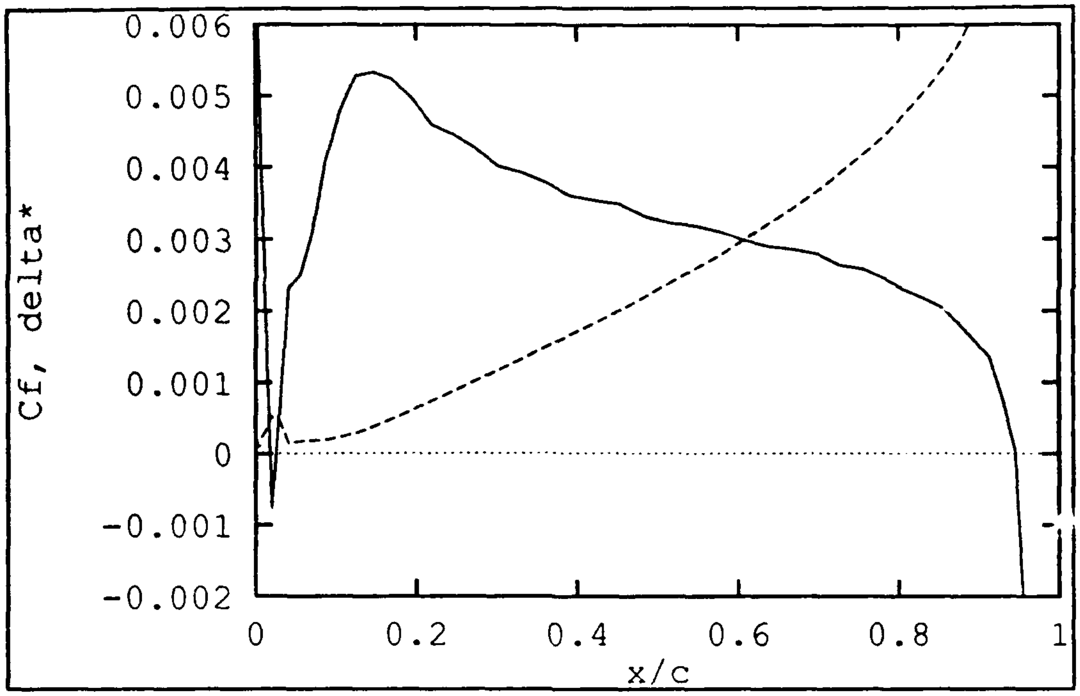

Figure 3.8 BL2D: C_f and δ^*, NACA 0012, AOA=8°, Re=540,000

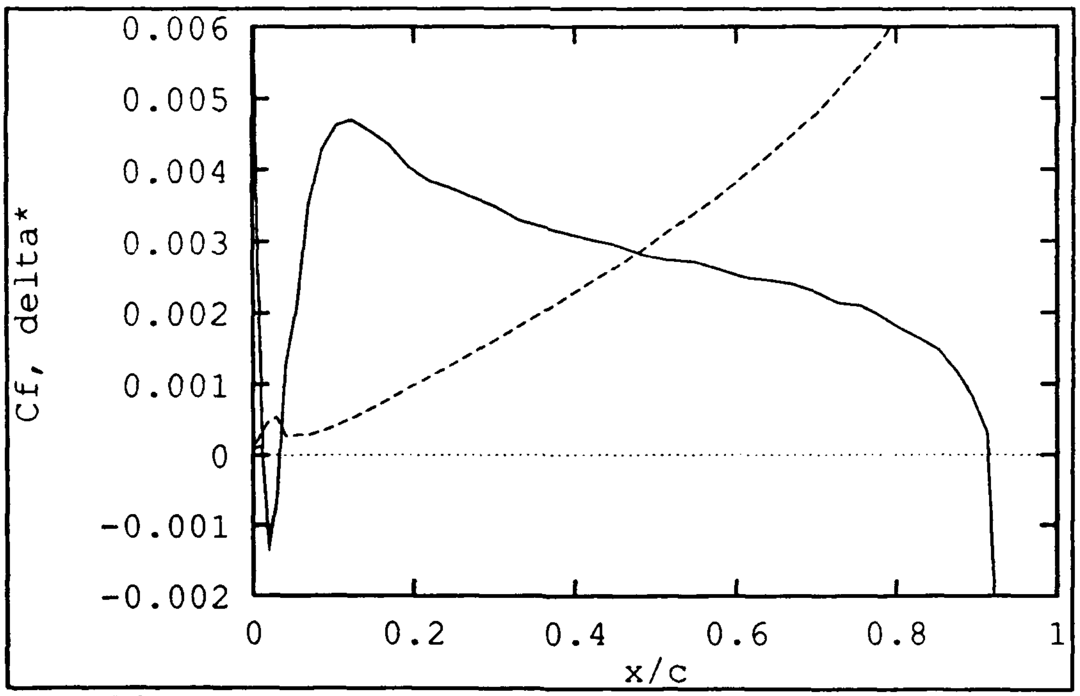

Figure 3.9 BL2D: C_f and δ^*, NACA 0012, AOA=10°, Re=540,000

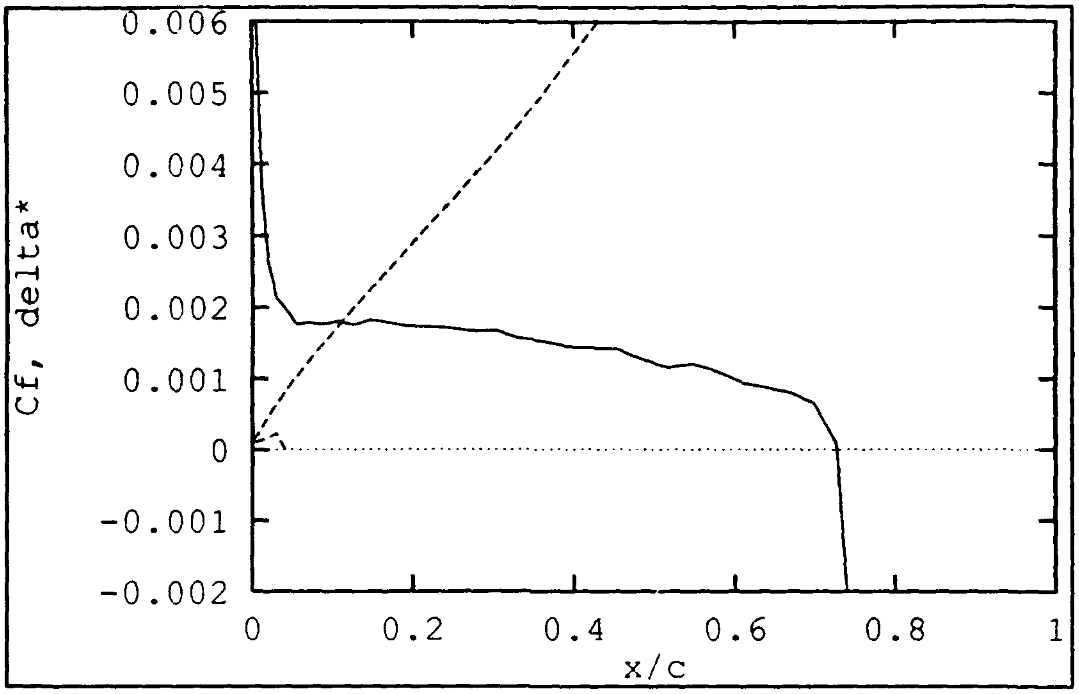

Figure 3.10 BL2D: C_f and δ*, NACA 0012, AOA=12°, Re=540,000

Knowing that the boundary layer equations break down when separation occurs, the anticipated information of this study was the x/c location of transition for as many angles of attack as possible before the C_f became negative, where it was assumed the program would not run. A beneficial discovery of this study is the ability of the program to recover from mild amounts of separation with meaningful results. At 4°, the first encounter with separation may be observed as the C_f just dips below zero. The pattern over the remaining airfoil surface suggests a separation "bubble" after which the flow reattaches, as opposed to near-stall separation. The program exhibited this recovery behavior all the way to 10°. The final plot at 12° shows that even though a solution was produced, convergence was not attained and the results were meaningless due to the greater amount of separation.

b. Boundary Layer Velocity Profiles

To complement the skin friction study and to further investigate the pattern of laminar separation, the program was modified to calculate and produce plotting output for velocity profiles at evenly spaced intervals along the top surface of the airfoil. Results are shown in Figures 3.11-3.16 for an AOA range of 0 to 10 degrees. Boundary layer growth is evident as the angle of attack increases. Furthermore, the region most prone to separation, as revealed by the point of inflection in the velocity profile, moves forward with increasing angle of attack, confirming the indications of the skin friction plots.

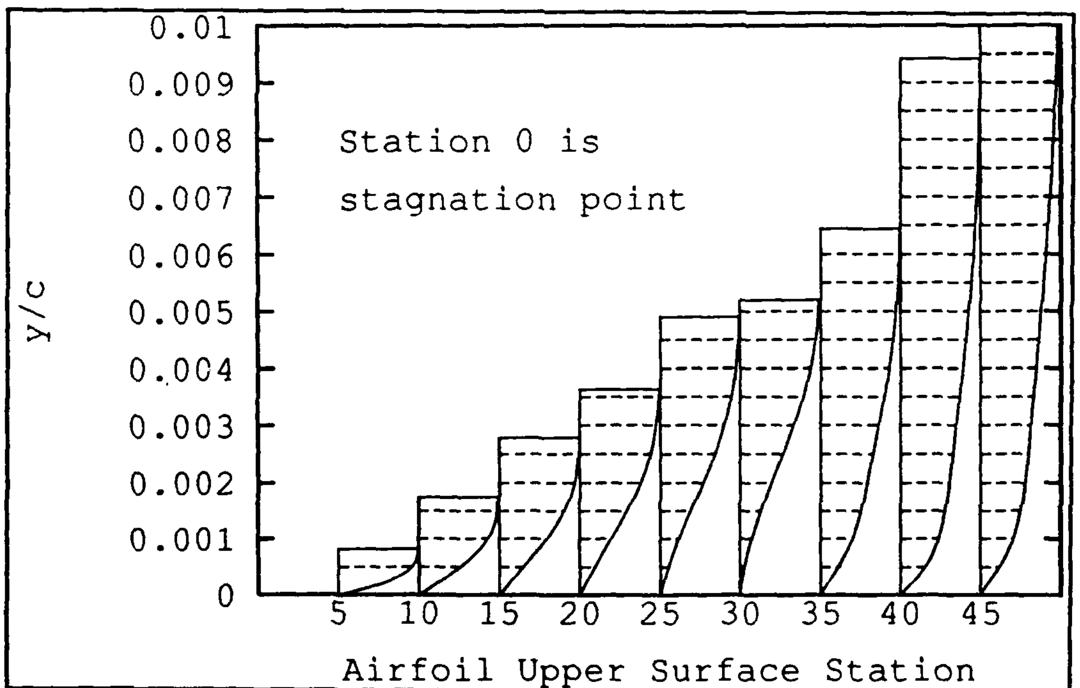

Figure 3.11 BL2D: Velocity Profiles, NACA 0012, AOA=0°, Re=540,000

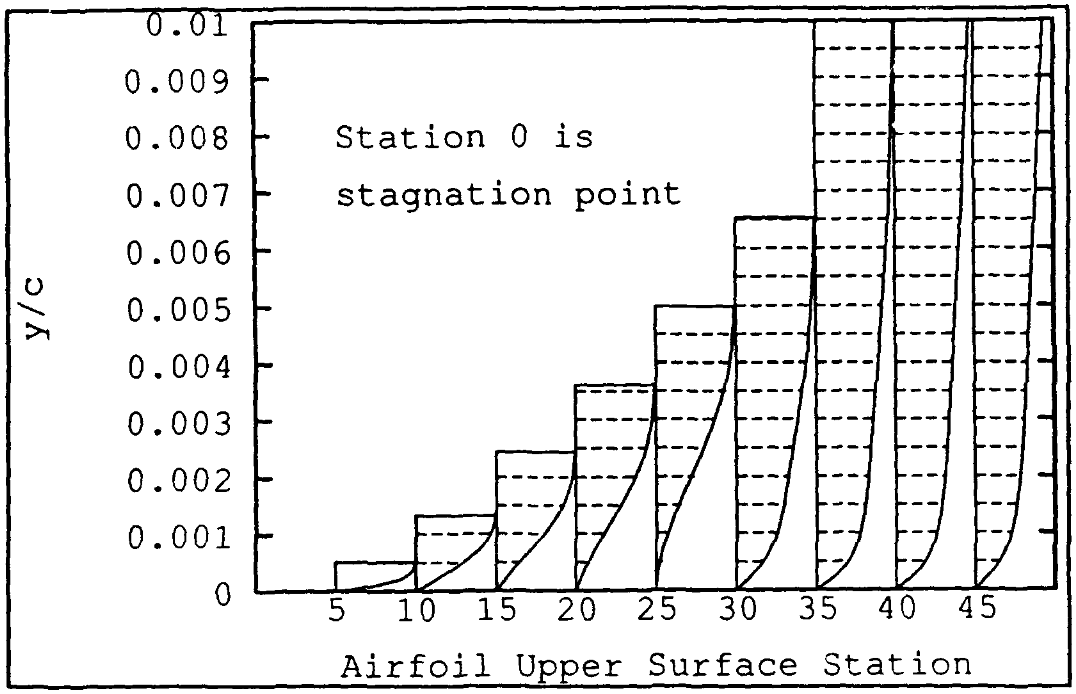

Figure 3.12 BL2D: Velocity Profiles, NACA 0012, AOA=2°, Re=540,000

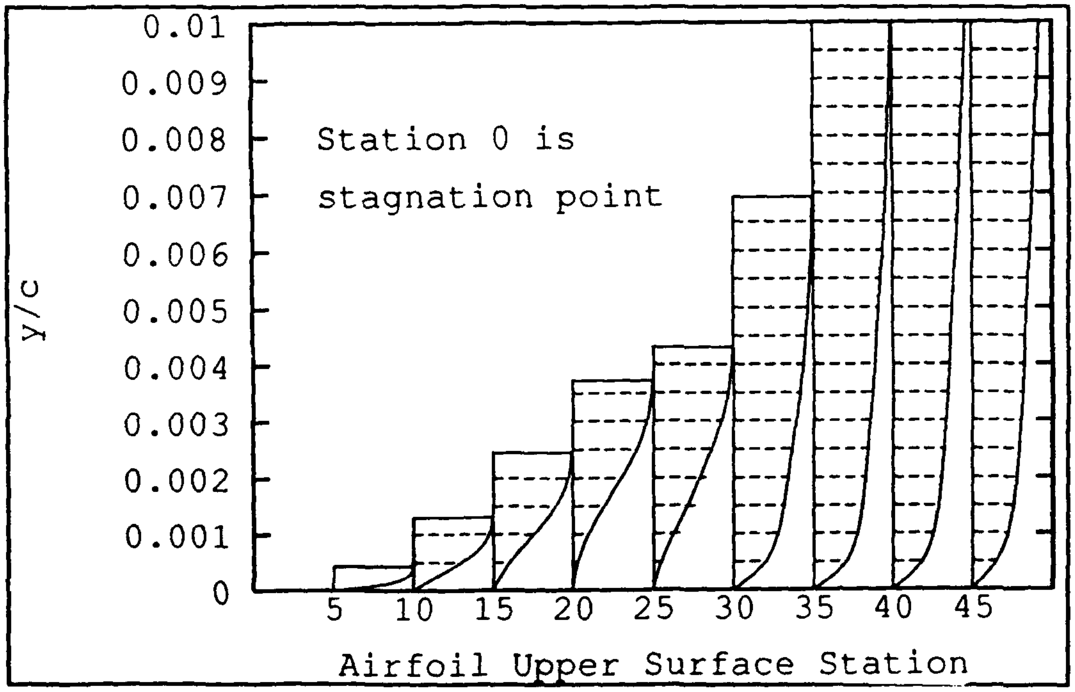

Figure 3.13 BL2D: Velocity Profiles, NACA 0012, AOA=4°, Re=540,000

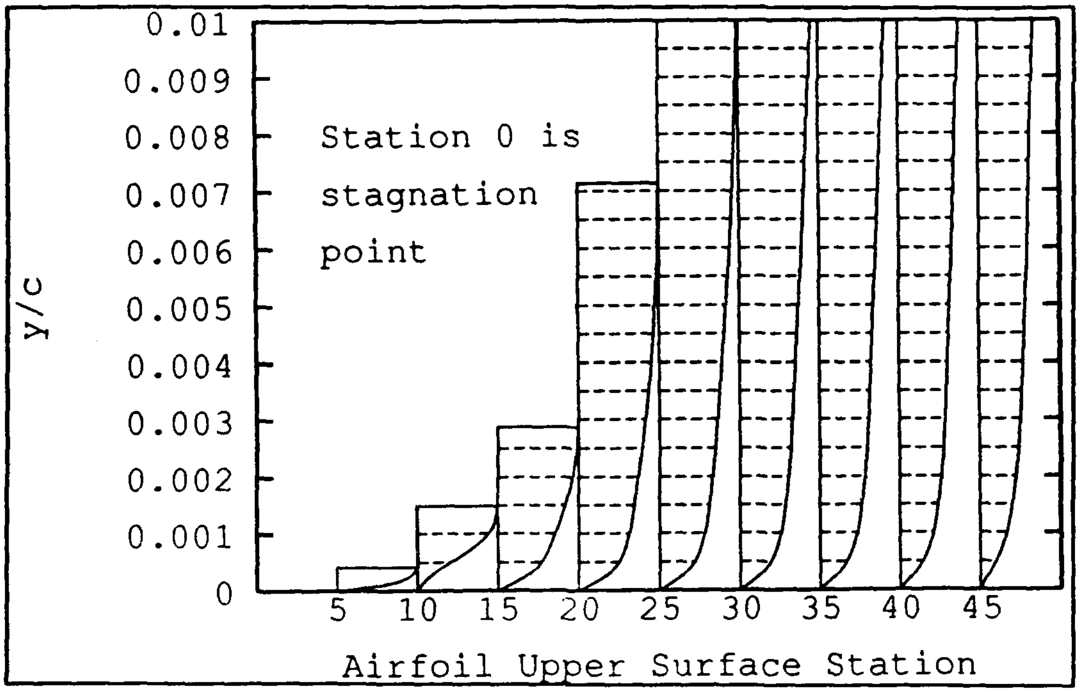

Figure 3.14 BL2D: Velocity Profiles, NACA 0012, AOA=6°, Re=540,000

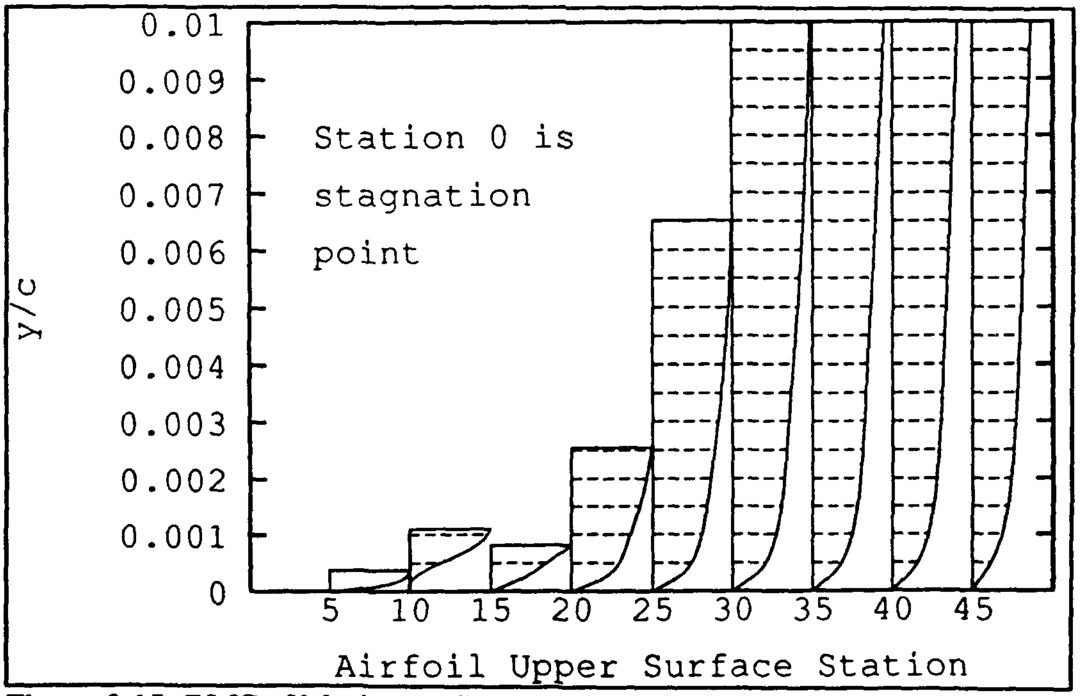

Figure 3.15 BL2D: Velocity Profiles, NACA 0012, AOA=8°, Re=540,000

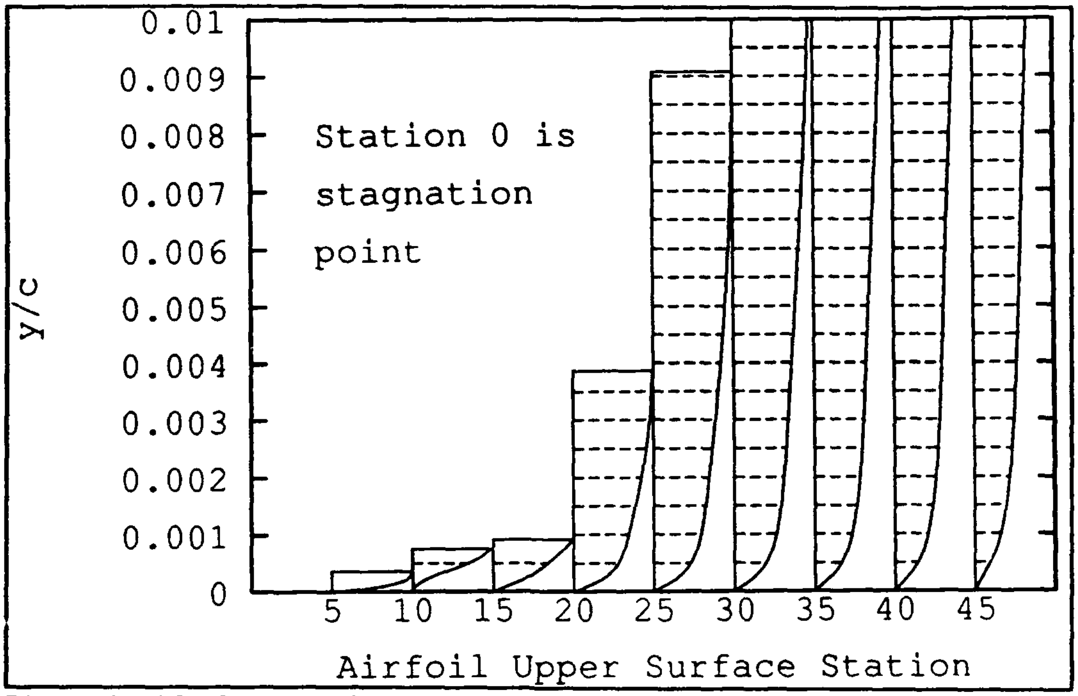

Figure 3.16 BL2D: Velocity Profiles, NACA 0012, AOA=10°, Re=540,000

4. Reynolds Number Changes

The direct boundary layer code was run for Reynolds numbers of 540,000, 750,000, and 1,000,000. A representative sample of the results at 2° is presented in Figure 3.17, showing a comparison of the skin friction coefficients. As the Reynolds number increases, two effects may be observed. First, the transition point moves forward, so there is more turbulent flow. This occurs on both surfaces, although only the upper surface is plotted. In addition, the minimum value of C_f increases. Both effects suggest that separation is less likely to occur as Reynolds number increases, all other conditions being constant.

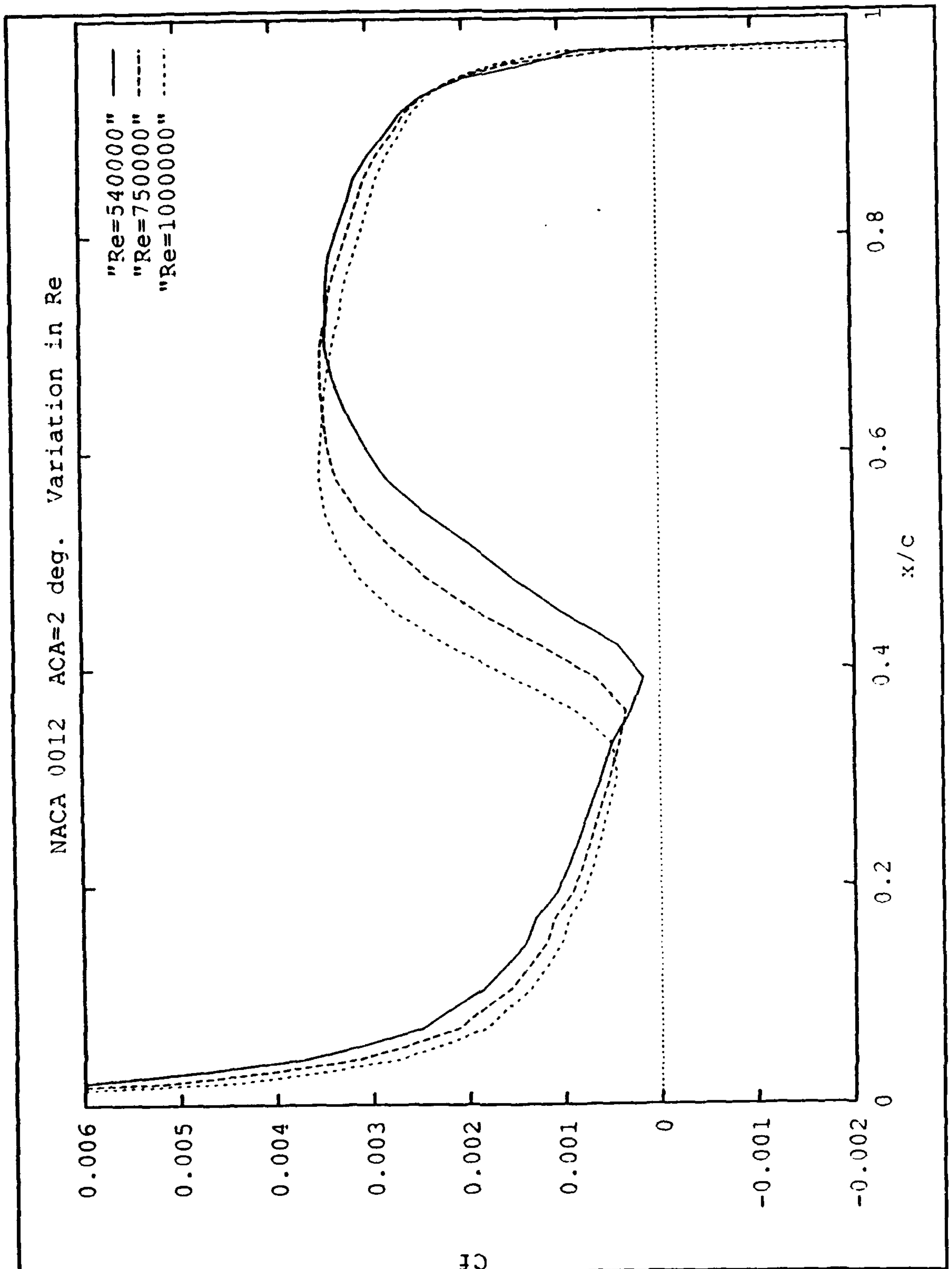

Figure 3.17 BL2D: Effects of Reynolds Number Changes

5. Unsteady Boundary Layers

By correlating an unsteady pressure distribution with a similar steady one, the question arises as to whether the steady direct boundary layer code can process the data in the form of a steady velocity distribution and produce results that correspond to the unsteady case. In the studies of Neace [Ref. 4], it was shown that an unsteady pressure distribution at a certain angle of attack has a closely matching steady pressure distribution at an angle of attack approximately 2° lower. For example, 10.14° unsteady corresponds to 8° steady. An unsteady panel method program called **U2DIIF** was used to calculate unsteady pressure distributions for ramp motion. With the hypothesis that the boundary layer characteristics, steady or unsteady, are driven by the pressure distribution, the pressures were converted to "steady" velocities for the unsteady angles of attack using Bernoulli's equation. Steady velocities for steady angles of attack were computed with the program **panel**. In Figure 3.18, the comparison between steady and unsteady velocity distributions reveals excellent agreement on the upper surface and fair agreement on the lower surface. Since velocity is the main input to **bl2d**, and the upper surface is much more critical (for positive angles of attack), the strong correlation suggests that the steady boundary layer code may indeed be able to produce results for the unsteady case. Many attempts were made to obtain such results using all of the methods explained in other sections to facilitate convergence. No solution was found for any of the cases investigated, however. Apparently, the differences on the lower surface had more effect than anticipated. Future investigations could include a modified velocity input, where the unsteady upper surface is spliced with the corresponding steady lower surface.

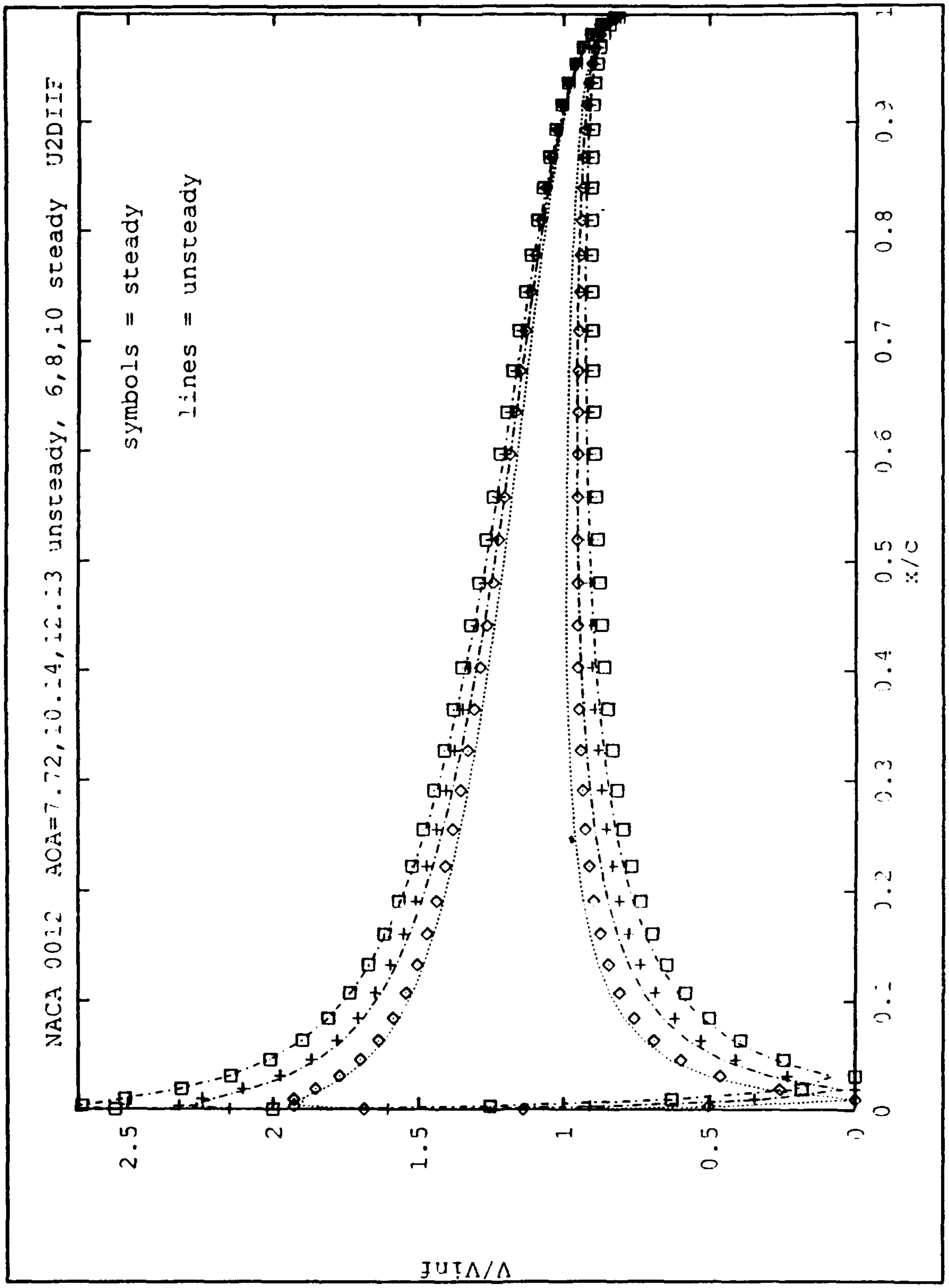

Figure 3.18 Comparison of Steady and Unsteady Velocity Distributions

Smoothness should be ensured in the connecting areas. A compressibility correction may also make a difference.

D. USER'S GUIDE TO BL2D

1. Output from the Program PANEL

Three files are output from the **panel** code described in the previous chapter. The file vel.dat is simply a printout of x/c and velocity relative freestream data, mainly for plotting purposes if desired. Even though, during computation in the program, panels on the lower surface downstream of the stagnation point were taken to have negative "directional" velocities, the output file correctly displays the positive "physical" velocities. The file cp.dat lists x/c and pressure distribution data. The other output file, called bl2d.dat, is generated to be compatible with the program **bl2d** as input. An example input file for the modified program is included in Appendix A.

2. Input Description

The first line of bl2d.dat consists of Reynolds number, transition location (x/c) on the upper surface, and transition location on the lower surface. The next line indicates the number of points and the i value of the location of the stagnation point. For the modified version, an additional input value is the transition flag. This indicates if the run is an initial estimate (0) or the transition values are fairly well determined and valid boundary layer calculations are desired. The remaining lines are identical to the velocity output file, except that y/c values of the airfoil coordinates are also included. Note that these velocities correspond to a particular angle of attack, the one that was specified when **panel** was run. The actual value of the angle of attack is not listed separately in the file.

The file bl2d.dat may be edited, replacing the originally specified values in the first few lines with new values for the desired flow. The Reynolds number is based on the appropriate reference length, usually the chord for an airfoil. The transition locations may be obtained from experimental data or from a calculation method. In the modified version, Michel's method may be used to initially estimate the location of transition onset. If no information is known before using the program, the transition values specified for the first run should be large enough to be downstream of the actual transition points, yet not so large that the program will not converge. The arbitrary values supplied by the program **panel** are 0.8 for the upper surface and 0.999 for the lower surface. If the transition flag at the end of the next line in the input file is set to 0, a run of the program will show estimates for transition location on the screen but these values will not be used for the boundary layer calculations. The input file should be edited, replacing the initial downstream transition values with the estimates and changing the transition flag to 1. If the program does not converge with these values, it will be necessary to experiment with slight deviations in the upper surface value. Since the transition value calculated is *only* an estimate, this experimentation procedure is the rule rather than the exception, especially at higher angles of attack.

For the stagnation location, it is important to not simply take the i value from the velocity output which corresponds to the velocity closest to zero. In order to work properly, **bl2d** usually requires the i value for the first point *after* the stagnation point, where points are numbered in a clockwise direction from the trailing edge. With only positive velocities to inspect, it is impossible to tell where this occurs. The program **panel**, however, uses the negative directional velocities in its calculations and automatically determines the proper i value to send to the output file bl2d.dat. If convergence cannot be obtained by varying the transition location, it may help in some cases to decrease the predetermined i value of the stagnation location by one.

3. Program Operation

When all necessary values have been changed, rename the input file appropriately for reference, such as bl5.dat for an AOA of 5°. Subsequent runs of the program **panel** for other AOA's will overwrite the output file bl2d.dat. When ready to run **bl2d**, copy the desired input file to bl2d.dat, the required input file name. To run the program, type:

>bl2d

Convergence may most easily be observed by plotting output files such as the skin friction coefficient. Modifications to various parameters as explained in the previous section may facilitate convergence. In some cases, however, such as an unusually shaped airfoil, a highly cambered airfoil, or a standard airfoil with a faulty input file, convergence may not be attainable. Additionally, all airfoils at a high enough angle of attack will cause the program to break down, as the direct boundary layer method cannot handle significant separation.

4. Output Description

The output to the screen indicates the progress of the program as it runs. In the modified program, the first estimates for transition locations will also print to the screen when the transition flag is set to 0. The remaining output in this case is not applicable and the program should be rerun with new transition values.

The standard output file is named bl2d.out. The first line repeats the values of Reynolds number and transition locations that were supplied as input. Next, a summary of boundary layer solutions is presented for ISF=1, the upper surface downstream of the forward stagnation point. Reading across, the data consists of station or point number (NX), x/c value (XC), distance over the airfoil surface starting at the forward stagnation point for that AOA (S), dimensionless wall shear parameter (VW), skin friction coefficient (CF), displacement thickness, δ^* (DLS), and θ, the momentum thickness (THT). The same data is listed for ISF=2, the lower surface at that AOA. It is important to realize that an otherwise converged solution may still have several highly divergent values for all parameters near the trailing edge. This does not invalidate the whole solution. Since the current investigation concerns primarily the leading edge area, the last few values may be ignored.

Several modifications were made to the program to allow additional informative output. The skin friction coefficient and displacement thickness are printed in the output files cf.dat and dls.dat, respectively, with x/c values for plotting. Individual boundary layer velocity profiles at every five stations along the airfoil upper surface are printed in various output files as follows:

NX=5	FOR031.DAT
NX=10	FOR032.DAT
NX=15	FOR033.DAT
NX=20	FOR034.DAT
NX=25	FOR035.DAT
NX=30	FOR036.DAT
NX=35	FOR037.DAT
NX=40	FOR038.DAT
NX=45	FOR039.DAT
etc.	

where station (NX) 1 is the first point after the stagnation point and numbering increases across the top of the airfoil. The example shown is for a 100 point airfoil with 50 points across the upper surface. For a different number of input points, the number of output files will adjust accordingly. The station nearest the trailing edge is not output because the results are often divergent. A comprehensive summary of these files is simultaneously stored in the output file FOR060.DAT for ease in plotting all of the profiles. An additional output file, FOR055.DAT, contains plotting data for drawing evenly spaced lines across the velocity profiles. This is a visual effect only and is not necessary if not desired.

While there are many ways to display this data, a command file compatible with **gnuplot** called **profile** shows the data to best advantage. A copy of this file is provided in Appendix A. It automatically plots all the profiles on one plot, showing boundary layer growth and the changing slope of the profiles, indicating when separation occurs. Using **gnuplot** in the directory where the data files and the file **profile** reside, type:

>load "profile"

Modifications may be made to the resulting plot using **gnuplot** commands, Appendix B, or editing the command file in a separate window.

5. PC Version

To increase its accessibility, **bl2d** was also converted to a version compatible with personal computers. Additional programs included on the **Boundary Layer Analysis** disk are a PC version of **panel**, an airfoil point generation program called **airfoil**, a PC version of **gnuplot**, and the command plotting file **profile**. Most of the instructions are the same for this version, but there are a few differences. The programs **panel** and **bl2d** are restricted to 100 or less airfoil points due to array limitations in the PC FORTRAN compiler. An automatic rerun feature was incorporated into **bl2d** for the case of initial transition estimates, where the first run internally restarts using the calculated estimates for boundary layer calculations. Finally, the velocity profiles are output only to a comprehensive file called profile1.dat instead of FOR060.DAT. There are no individual velocity profile files. The horizontal line file called FOR055.DAT in the UNIX version is called profile2.dat for the PC version.

IV. VISCOUS-INVISCID INTERACTION CODE

A. THEORY

The direct boundary layer code, described in the previous chapter, calculates a displacement thickness for a prescribed pressure distribution (or equivalently, a velocity distribution, for incompressible flow). As the name implies, it is a direct calculation involving one pass, thus very little time is required. Another method, known as the inverse boundary layer method, calculates a pressure (or velocity) distribution for a prescribed displacement thickness. The displacement thickness represents an "effective body" as far as the flow is concerned. Iteration is required and the method requires considerably more calculation time. The main advantage of the inverse method is the ability to calculate through regions of separation.

The simplest viscous-inviscid interaction method divides the flowfield into an inner viscous region where boundary layer calculations are performed and an outer inviscid region where potential flow analysis prevails. The solutions are then iterated until they match along the dividing line. This method has "weak" interaction because the only exchange of information is along the boundary.

In the strong interaction method, both the pressure and displacement thickness are treated as unknowns and are solved simultaneously with successive sweeps over the airfoil. The external boundary condition for the boundary layer equation at the outer edge of the viscous region is

$$u_e(x) = u_e^0(x) + \delta u_e(x) \tag{4.1}$$

where $u_e^0(x)$ is the inviscid velocity over the airfoil and $\delta u_e(x)$ is the perturbation due to viscous effects, expressed as

$$\delta u_e(x) = \frac{1}{\pi} \int_{x_a}^{x_b} \frac{d}{d\sigma}\left[u_e(\sigma)\delta^*(\sigma)\right] \frac{1}{x-\sigma}\, d\sigma \tag{4.2}$$

where $d(u_e\delta^*)/d\sigma$ is the blowing velocity. Equations 4.1 and 4.2 comprise the interaction, or coupling law. The interaction takes place between x_a and x_b. The integral term is known as the Hilbert integral, in which the displacement effect is modeled by source/sink distributions using potential flow theory.

B. COMPUTER CODE

1. Overview

The viscous-inviscid program incorporates a self-contained panel code based on the Hess-Smith method, a boundary layer calculation routine, and an interaction scheme. The inviscid panel method is similar to the program **panel** described in Chapter 2; however, the boundary condition of zero normal velocity on the surface of the body is replaced by a blowing velocity determined from the boundary layer calculations. This blowing velocity is used to represent the viscous effects on the inviscid flow. The boundary layer method is similar to the program **bl2d** with some modifications that account for the presence of the wake and for low Reynolds number flows. The viscous-inviscid code used for the investigation was developed by T. Cebeci. Run time on the Stardent computer is about five minutes.

2. Models

a. Turbulence Model

As in the direct boundary layer program, the eddy-viscosity formulation of Cebeci and Smith is used, with separate formulas for the inner and outer regions. This model in this program has the additional features of low Reynolds number effects and a wake flow model. The modified equations are expressed as:

$$
\begin{aligned}
(\epsilon_m)_i &= L^2\left|\frac{\partial u}{\partial y}\right|\gamma_{tr} && 0 \le y \le y_c \\
(\epsilon_m)_o &= \alpha u_e \delta^* \gamma_{tr}\gamma && y_c \le y \le \delta
\end{aligned}
\tag{4.3}
$$

where

$$
\begin{aligned}
& L = 0.4y[1-e^{-\left(\frac{y}{A}\right)}] \qquad A = 26\nu u_\tau^{-1} \qquad u_\tau = \sqrt{\nu\frac{\partial u}{\partial y}_{max}} \\
& \alpha = \frac{0.0168}{\left[1-\beta\left(\frac{\partial u/\partial x}{\partial u/\partial y}\right)(-\overline{u'v'})_{max}\right]^{2.5}} \qquad R_T = \frac{\tau_w}{(-\overline{u'v'})_{max}} \\
& \gamma = \frac{1}{1+5.5\left(\frac{y}{\delta}\right)^6} \qquad \beta = \begin{cases} \dfrac{6}{1+2R_T(2-R_T)} & R_T < 1.0 \\ \dfrac{1+R_T}{R_T} & R_T \ge 1.0 \end{cases}
\end{aligned}
\tag{4.4}
$$

and the transition length is represented by

$$\gamma_{tr} = 1 - \exp\left[-G(x-x_{tr})\int_{x_{tr}}^{x} \frac{1}{u_e}dx\right]$$
$$G = \left(\frac{3}{C^2}\right)\frac{u_e^3}{\nu^2}Re_{x_{tr}}^{-1.34} \tag{4.5}$$
$$C^2 = 213\,[\log Re_{x_{tr}} - 4.7323]$$

For the wake flow, the eddy-viscosity formulation is

$$\epsilon_m = (\epsilon_m)_w + [(\epsilon_m)_{te} - (\epsilon_m)_w]\exp\left[-\frac{x-x_{te}}{20\delta_{te}}\right] \tag{4.6}$$

where $(\epsilon_m)_w$ specifies the eddy viscosity of the far wake, taken as the maximum of the lower and upper wake eddy viscosities:

$$(\epsilon_m)_w^l = 0.064\int_{-\infty}^{y_{min}} (u_e - u)\,dy$$
$$(\epsilon_m)_w^u = 0.064\int_{y_{min}}^{\infty} (u_e - u)\,dy \tag{4.7}$$

with y_{min} the location where $u = u_{min}$.

b. Transition Model

The program uses an empirical formula called Michel's criterion to calculate a first approximation to the transition location on the upper and lower airfoil surfaces. It is expressed as a relationship between the Reynolds numbers based on momentum thickness and on the x (surface) location of transition:

$$R_{\theta_{tr}} = 1.174\left(1 + \frac{22400}{Re_{x_{tr}}}\right) Re_{x_{tr}}^{\ 0.46} \tag{4.8}$$

This is the same equation used in the modified version of **bl2d**. The program **incompbl**, however, sometimes adjusts the resulting value. If an area of separation with subsequent reattachment is calculated, the transition onset is moved to the beginning of the separation bubble. The transition location that is printed in the output file always corresponds to a nodal point. In many cases, the program will produce a fairly accurate result. Sometimes, however, a refinement may be needed. This may be done by experimenting with small shifts around the calculated value, as described in Chapter 3 for the direct boundary layer program. The experimentation may be implemented by using the transition specification option, which overrides calculation. Input options are described in the User's Guide section.

A more advanced technique, suggested by Cebeci, is the e^n method, which makes use of linear stability theory. A separate stability/transition code incorporating this method is run using the output of the viscous/inviscid code (Michel's criterion) as the first estimate. The new value is supplied as input to the first code, this time overriding Michel's criterion. This type of manual iteration continues until convergence, usually within three to four cycles. At the present time, determination of all required input values and analysis of the output require significant experience and judgment, prohibiting a programmed link between the two codes until further refinement is accomplished. The e^n method was not used in this study.

As in the program **bl2d**, the program **incompbl** uses the Chen-Thyson transition range model to calculate the length of the transition region. Equation 4.5 shows the modified version of this model. The program incorporates an improvement for $G_{\gamma tr}$, the transition length parameter. $G_{\gamma tr}$ may be identified by reducing the $(3/C^2)$ term to the form $(1/G_{\gamma tr})$, with $G_{\gamma tr}=C^2/3$. In the original model, the constant C has a recommended value of 60, resulting in $G_{\gamma tr}=1200$. Whereas a value of 1200 may work well for large Reynolds numbers, values from 20 to 80 have been shown to be most successful in low Reynolds number flows where separation bubbles exist [Ref. 5]. The program determines an appropriate value using an empirical correlation formula in the form of C^2, also shown in Equation 4.5.

3. Numerical Techniques

a. Hilbert Integral

Equation 4.1, containing the Hilbert integral, may be approximated in discretized form as

$$u_e(x) = u_e^{\kappa}(x) + \sum_{j=1}^{n} c_{ij}(u_e\delta^* - u_e\delta^{*\kappa})$$

where $u_e^{\kappa}(x)$ corresponds to the inviscid velocity distribution which contains the displacement thickness effect $(\delta^*)^{\kappa}$ and c_{ij} is a matrix of interaction coefficients which are functions of geometry only.

b. FLARE Approximation

In regions of recirculating flow, such as a separation bubble, numerical stability difficulties may be encountered. The FLARE approximation, due to Flügge-Lotz and Reyhner, neglects the longitudinal convective term $u(\partial u/\partial x)$ in the region of negative u velocity.

C. RESULTS

Studies were conducted to:

- Determine the effect of including the wake in the calculations
- Investigate the possible occurrence of negative skin friction and to determine its significance
- Compare the results with the direct boundary layer code

Since the viscous-inviscid method has the ability to calculate through regions of separation, additional information was anticipated beyond that provided by the direct boundary layer method.

1. Wake Calculations

Since one of the input options is for the inclusion of wake calculations, investigations were performed to determine its effect. Calculations may be limited to the airfoil surface only, or a grid extending into the wake region may also be used. A representative sample of the results is shown in Figure 4.1, depicting the skin friction coefficient for three angles of attack. At 2°, the main difference is a movement aft of the transition point, where the remainder of the curve maintains its original shape. At

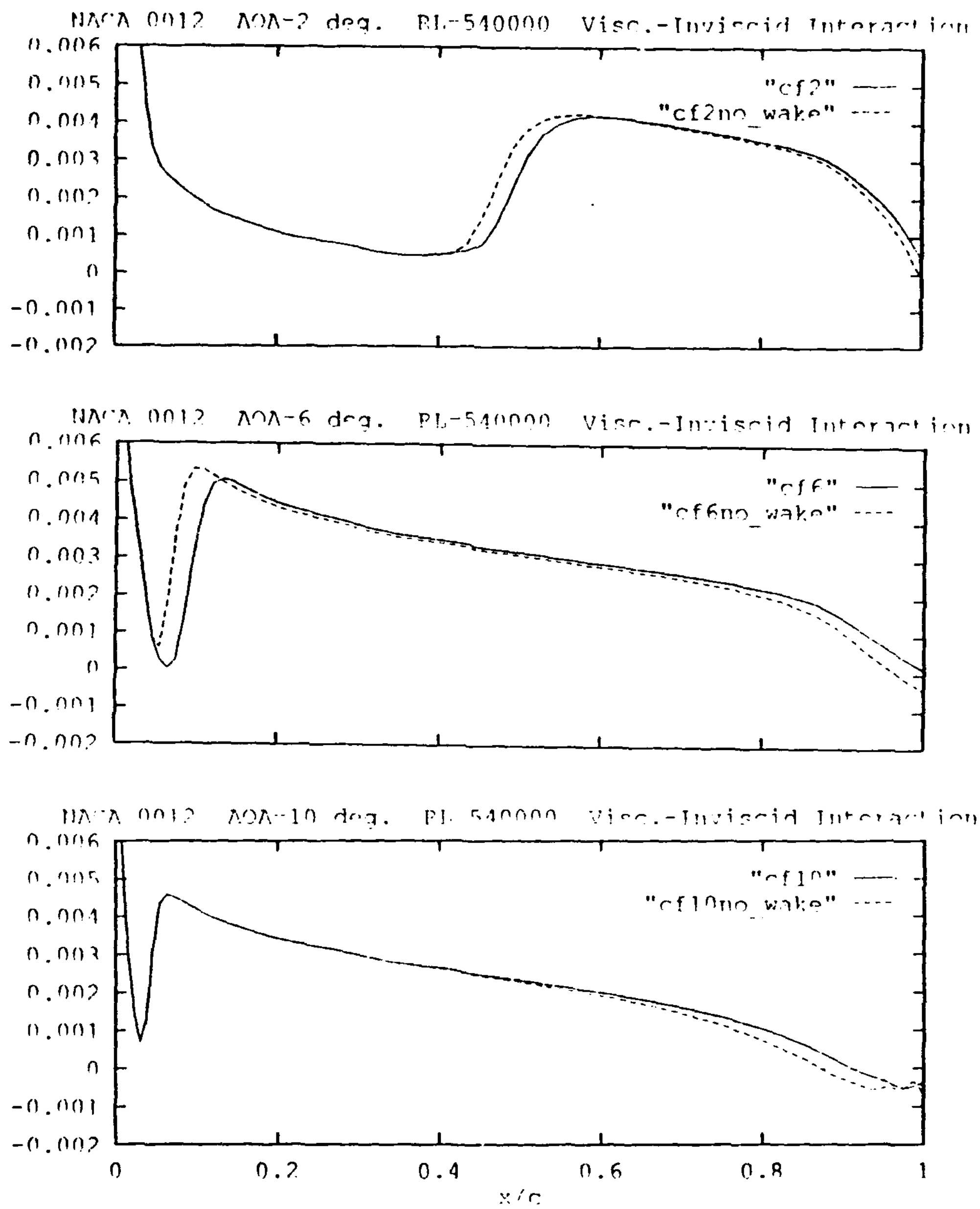

Figure 4.1 Effect of Wake Calculations

6°, the transition point also moves forward; however, the shape of the turbulent section is slightly altered. At 10°, the transition point is unchanged. More significant changes occur progressing across the top of the airfoil, leading to a delay in the point where the skin friction falls below zero. Thus, the primary effect of wake flow is to reduce flow separation on the airfoil, especially important at higher angles of attack. This will allow calculations to continue when convergence may not have been reached otherwise. These results are consistent with those found by Cebeci [Ref. 6]. Therefore, the wake was used in all further studies using this code.

2. Laminar Separation

a. Comparison with Direct Boundary Layer Method

The program **incompbl** was run for the same conditions as the previous study with **bl2d**, a NACA 0012 airfoil with a Reynolds number of 540,000. The angle of attack was increased in two degree increments. Figures 4.2 to 4.7 compare the skin friction results for both programs. The low angles of attack show excellent agreement in the laminar region. As the angle of attack increases, the most notable difference is the absence of $C_f < 0$ for the viscous-inviscid method. Since the direct boundary layer code failed to converge with separation greater than that produced at 10°, no comparison could be performed beyond this point.

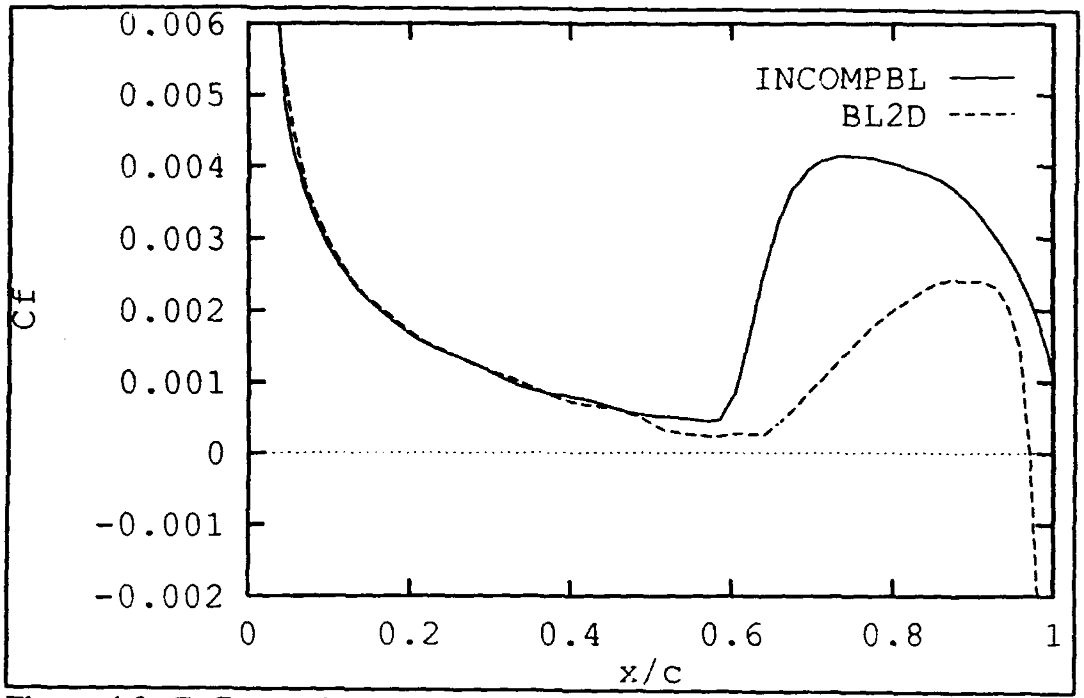

Figure 4.2 C_f **Comparison, NACA 0012, AOA=0°, Re=540,000**

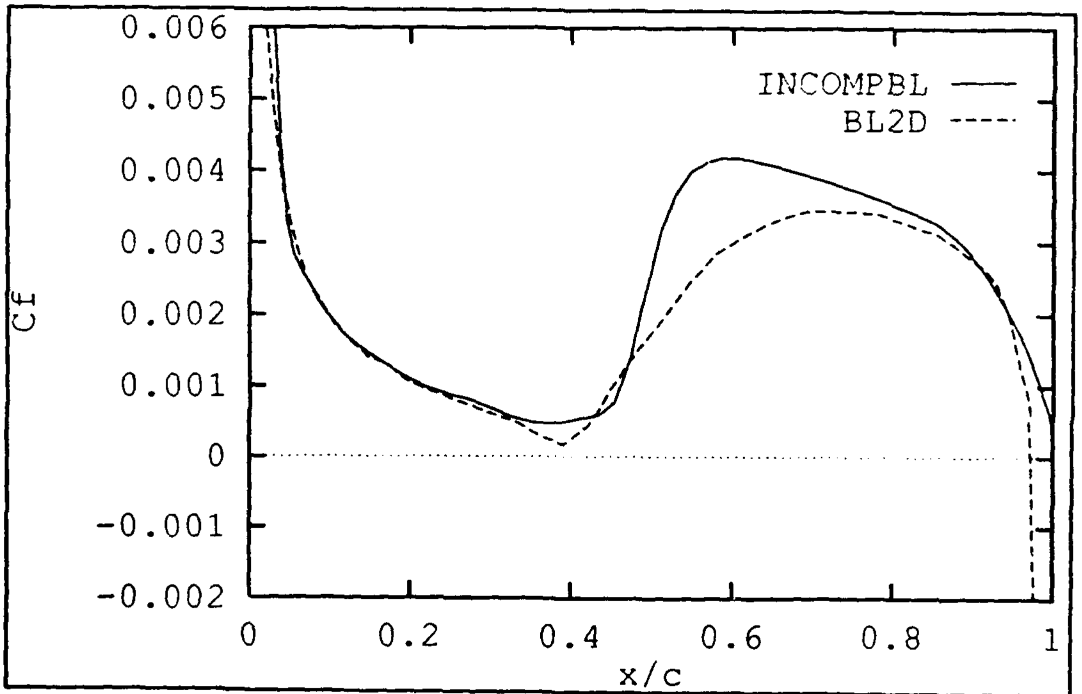

Figure 4.3 C_f **Comparison, NACA 0012, AOA=2°, Re=540,000**

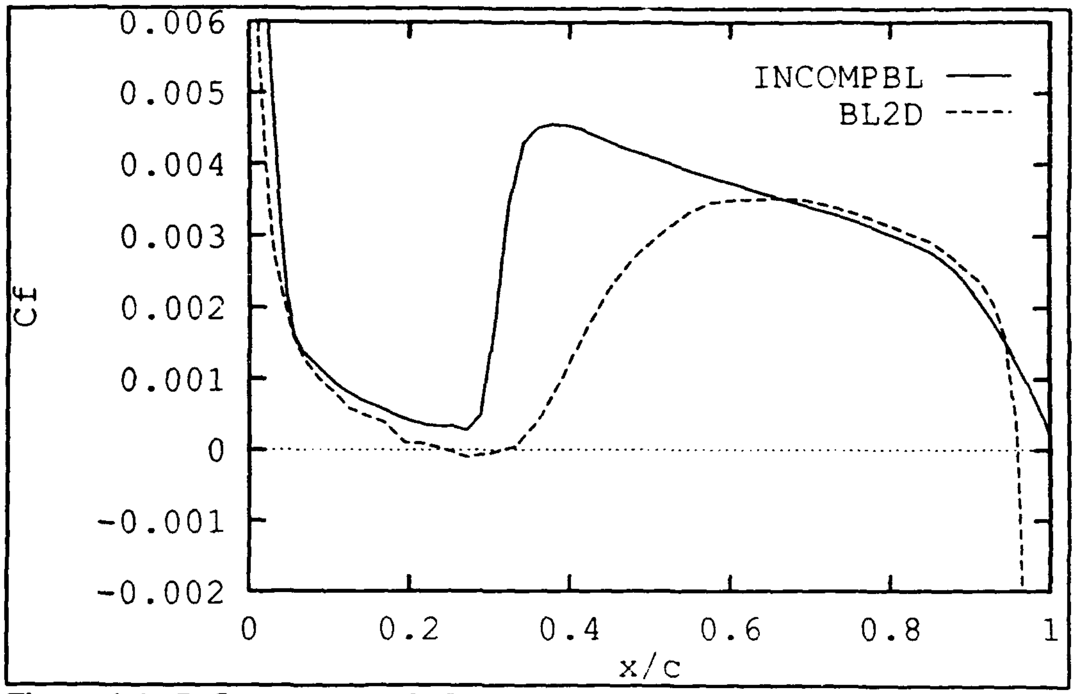

Figure 4.4 C_f Comparison, NACA 0012, AOA=4°, Re=540,000

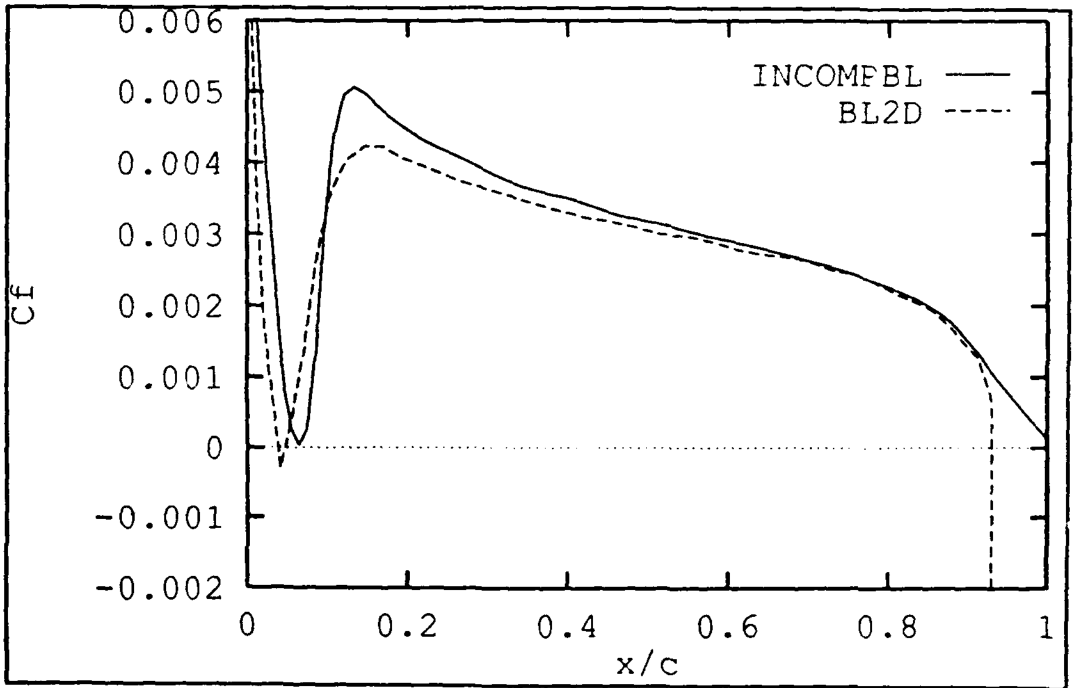

Figure 4.5 C_f Comparison, NACA 0012, AOA=6°, Re=540,000

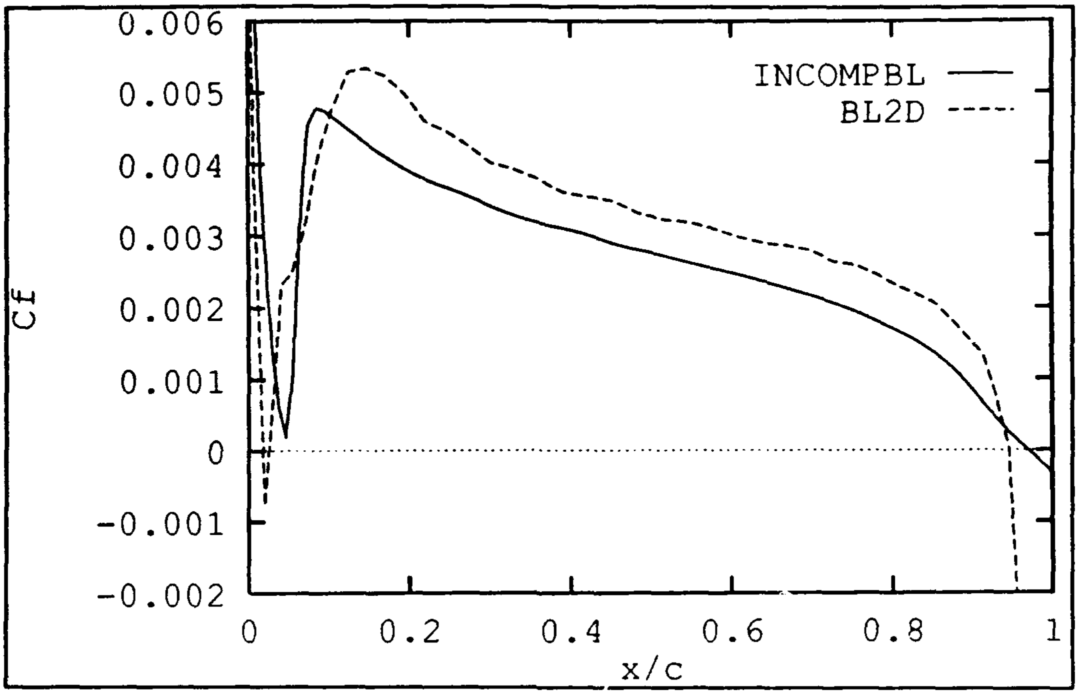

Figure 4.6 C_f Comparison, NACA 0012, AOA=8°, Re=540,000

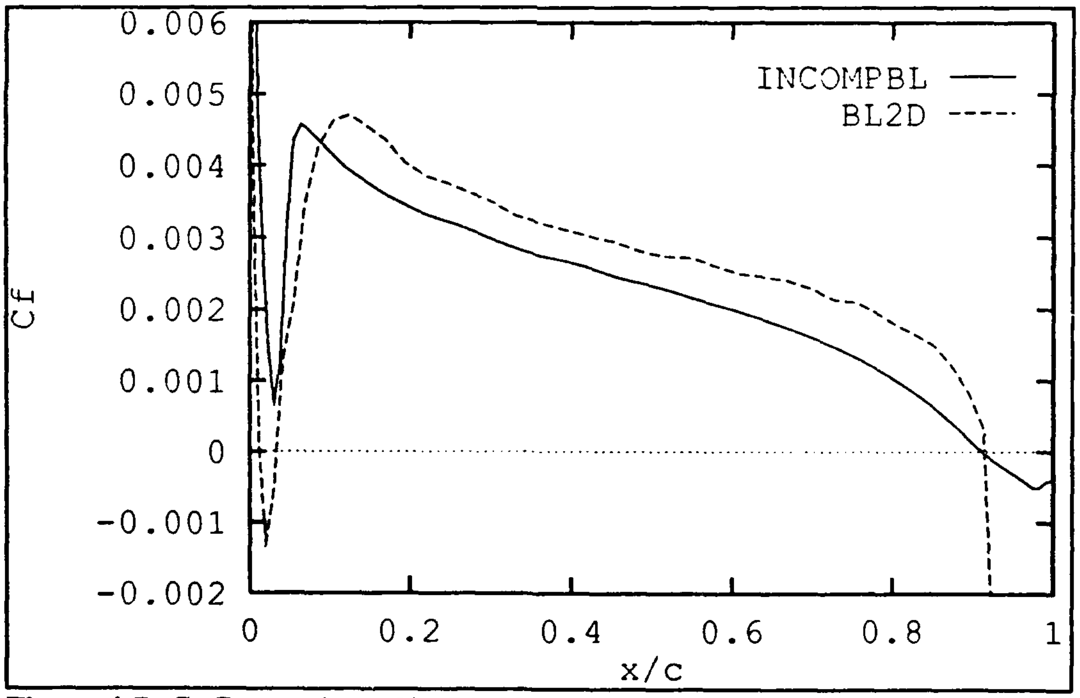

Figure 4.7 C_f Comparison, NACA 0012, AOA=10°, Re=540,000

b. Investigation of Higher Angles of Attack

In Figures 4.8 to 4.14, the angle of attack for the viscous-inviscid code was increased further in search of separation, indicated by $C_f < 0$. Finer increments were used when separation appeared to be imminent. Separation did not occur until 13.3° and was gone by 13.4°. A final plot at 14° shows that the remaining trend is an increase in the minimum value of C_f. The amount of separation at 13.3° was minuscule, as only one data point fell below zero with a value of -0.00065.

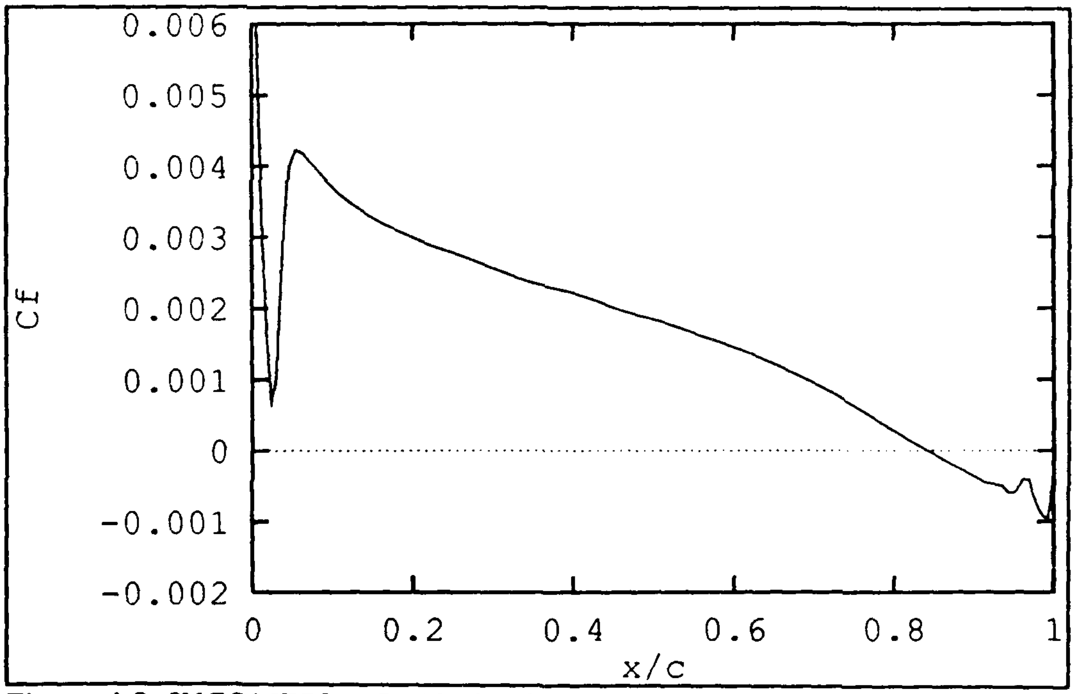

Figure 4.8 INCOMPBL: C_f, NACA 0012, AOA=12°, Re=540,000

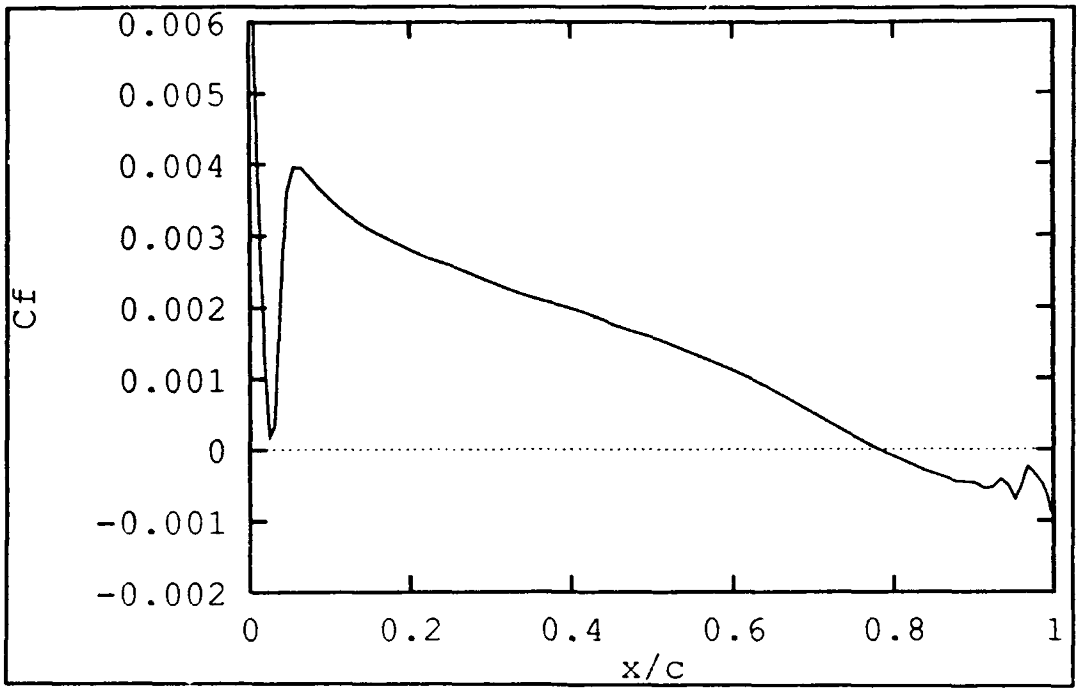

Figure 4.9 INCOMPBL: C_f, NACA 0012, AOA=13°, Re=540,000

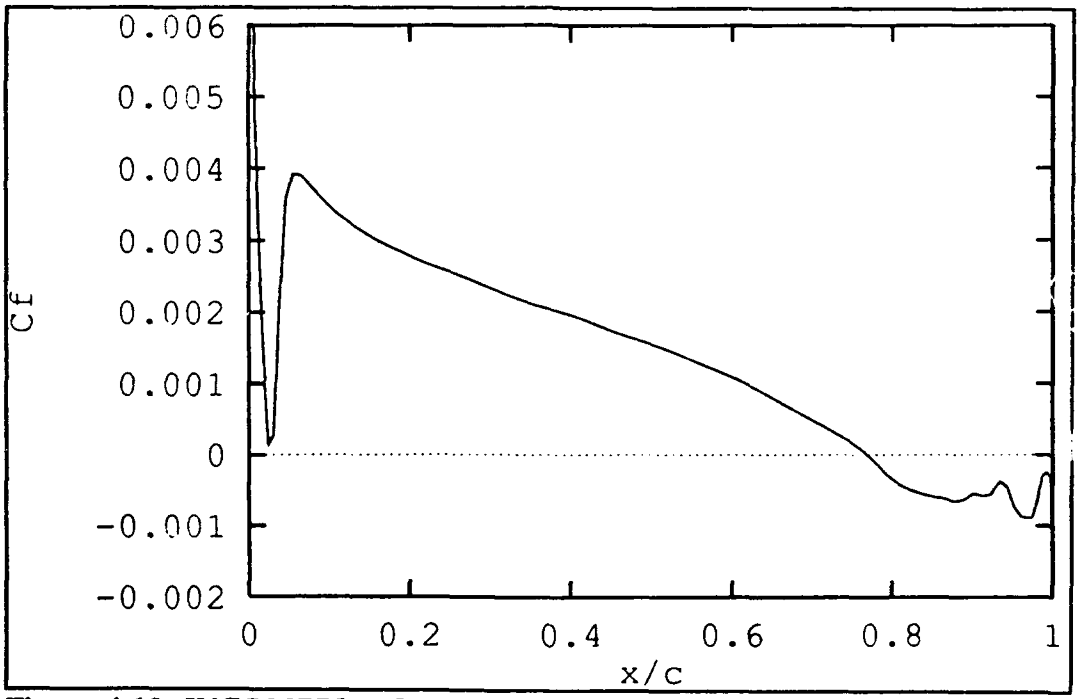

Figure 4.10 INCOMPBL: C_f, NACA 0012, AOA=13.1°, Re=540,000

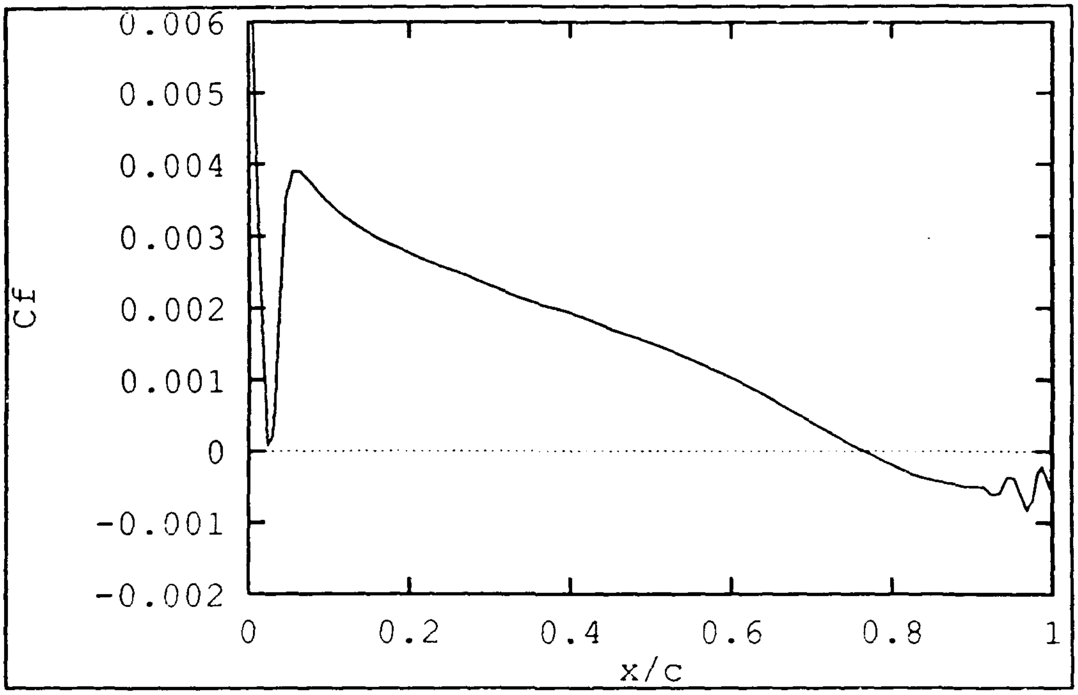

Figure 4.11 INCOMPBL: C_f, NACA 0012, AOA=13.2°, Re=540,000

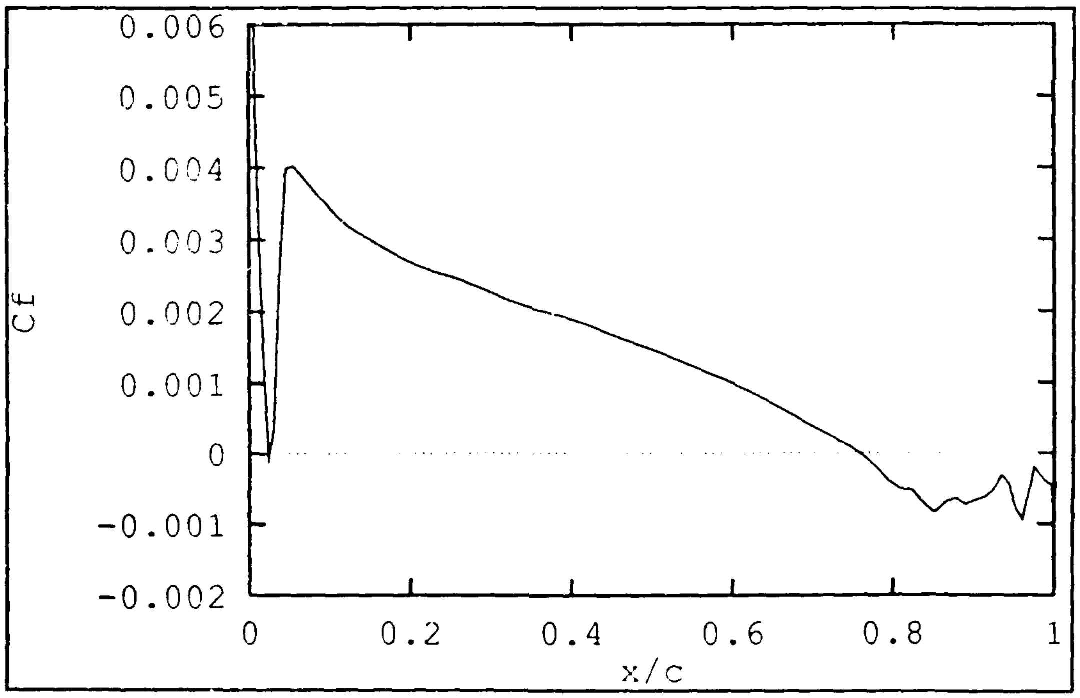

Figure 4.12 INCOMPBL: C_f, NACA 0012, AOA=13.3°, Re=540,000

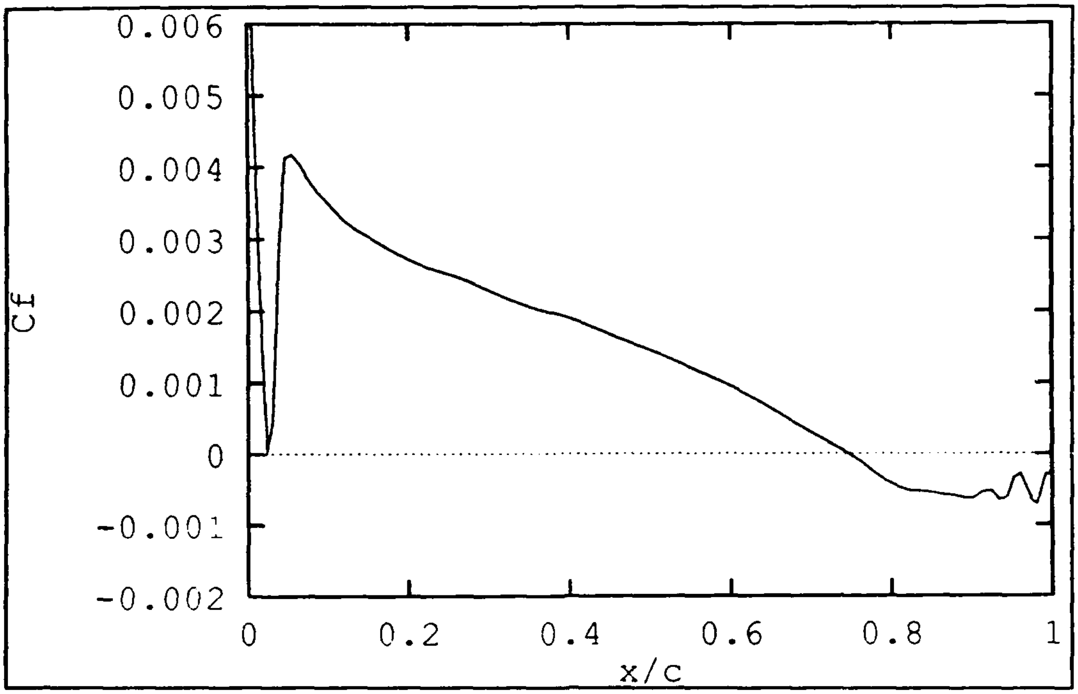

Figure 4.13 INCOMPBL: C_f, NACA 0012, AOA=13.4°, Re=540,000

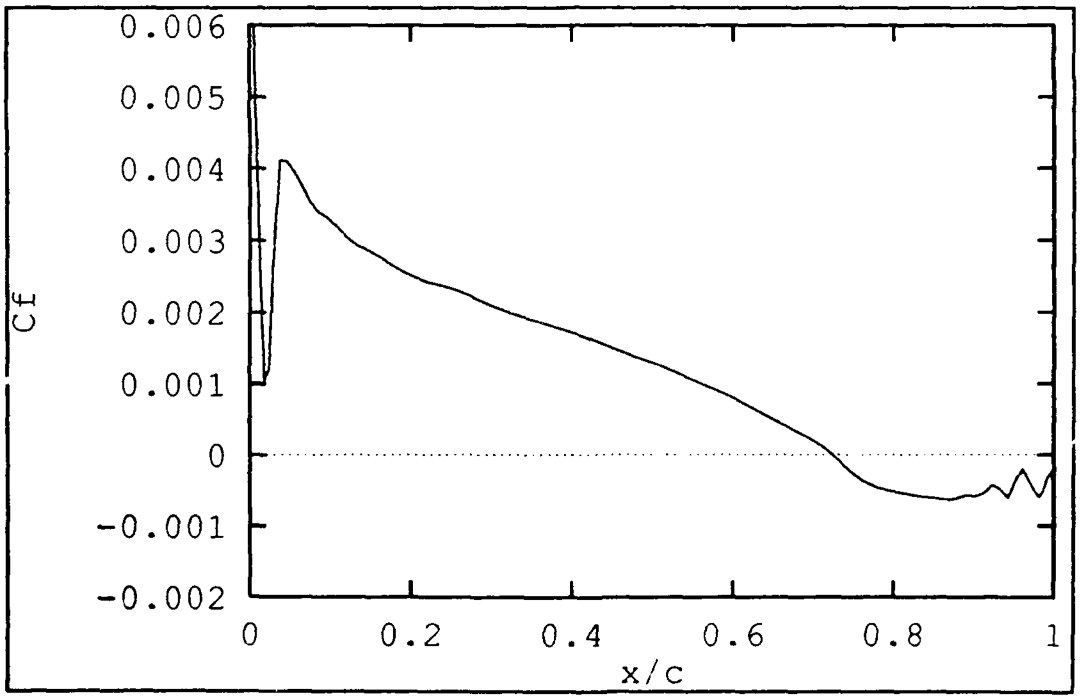

Figure 4.14 INCOMPBL: C_f, NACA 0012, AOA=14°, Re=540,000

D. USER'S GUIDE TO INCOMPBL

1. Required Files

The files required in a directory to run this program are:

- **incompbl***: a compiled, executable program compatible with the computer being used (arbitrary name if starting from source code)
- **FOR001.DAT** (Stardent) or **fort.1** (IRIS): first input file containing airfoil related data (required name)
- **incompbl.dat**: second input file containing flow and calculation information (arbitrary name)

2. UNIX FORTRAN

There are three ways to get data from external input files into a FORTRAN program on a UNIX based system. The first uses an OPEN statement which associates a specific input file name with a file number, just as on a personal computer. This method is not used in this program. The second method has READ statements using file numbers not declared by an OPEN statement. In this case, a default file name such as FOR001.DAT is assumed for a READ statement referencing a file number of one, for example. The program uses this method for the first input file (hence the required name) as well as many output files, which the user can modify or add to in the source code before compilation. The third method, which may be used only once in a program, is to specify an input file on the command line at execution time. The file can have any name chosen by the user as long as that file has data in the correct format expected by the program. READ statements using a file number of five assume interactive input from the keyboard during execution or an input file name in the command line.

3. Starting from the Source Code

If the compiled version is not available or it is not certain which source code file corresponds to the compiled file that exists, the source code is the best starting point. Otherwise, proceed with execution procedures in the next section.

The source code **incompbl.f** may be obtained from either the Stardent or IRIS computer and transferred from one to the other as needed using the **ftp** utility. Once compiled, however, the program will be computer specific, even if the executable versions have the same name.

Modifications may be made to the source code first if desired. For example, a WRITE (32,*) statement could be inserted to output specific data to a file with a default name of FOR032.DAT (Stardent) or fort.32 (IRIS). This method is often used to quickly obtain files of x/c vs. velocity, pressure, or other parameters for plotting.

To compile the FORTRAN source code for the Stardent, type:

>fc -O2 -o incompbl incompbl.f

The term **fc** is for FORTRAN Compile. The term **-O2** permits vector optimization. The program will run without it, but a warning notice will be issued after compilation. Note the use of the letter "O", not the number zero, "0". **DO NOT** use the option **-O3** instead. This option is for parallel processing, a feature not currently incorporated on the NPS Stardent. The program may appear to successfully compile and run, but there will usually be errors in the output. The term following the **-o** is the name of the output executable program. Any name may be used. The program name will be displayed with an asterisk (*) following it in the directory listing to indicate that it is an executable program. Finally, the source code or codes are listed in order, only one being used in this case. To compile on the IRIS, type

>f77 -O3 -o incompbl incompbl.f

The optimization levels have different meanings on the IRIS and **-O3** is the correct parameter.

During the compilation process, a file **incompbl.o**, called an object file, is produced appears in the directory listing. This file is not needed in this application and may be deleted.

4. Input File Editing

The first input file pertains to the panel method part of the program and is called **FOR001.DAT** (Stardent) or **fort.1** (IRIS). A sample file is included in Appendix A. The first line is simply a number telling how many of the following lines are for comments. The next few lines contain the comments, such as the type airfoil being analyzed, the date of the test, or any other information useful to the user. The next group of data consists of the angle of attack (ALPI), the x/c location of the pivot about which the airfoil rotates to a new angle of attack (PIVOT), and the number of panels defining the lower and upper surfaces (NLOWER and NUPPER, respectively). Finally, the x/c and y/c coordinates are listed in separate blocks, with the order starting at or near the trailing edge, proceeding across the lower surface, then the upper surface, and ending at or near the trailing edge. The number of points will be one higher than the number of panels, even if the first and last points coincide. The trailing edge point simply is listed twice in this case. The only part of this file that is likely to get changed on a routine basis is the angle of attack, as the other values are usually fixed for a given airfoil.

The other input file pertains to the boundary layer part of the code. This file is normally called **incompbl.dat**, although the user may give it any name and use this name on the command line at the time of execution. A sample input file is included in Appendix A. IWAKE is the viscous wake flow flag. A zero indicates that these effects will not be included, while a one indicates that they will be included. NXT dictates the number of chordwise stations on the body. NW sets the number of chordwise stations in the wake. ITREND refers to the number of calculation cycles, where 20 is a good starting number, and 30 or 40 may be needed. ITR(1) is a flag for the transition location specification method for the upper surface. A zero will activate a calculation using Michel's method, and a one indicates that the location will be provided as part of the input. ITR(2), for the lower surface, should be zero. ISWPMX is the number of sweeps in each cycle. A cycle corresponds to the calculation of inviscid and viscous flow equations. One sweep is usually sufficient but, in some cases, it may be necessary to use 2 or 3 sweeps in one cycle. RL is the Reynolds number based on chord length. XCTR(1) is the x/c value for the transition location on the upper surface. This value is only used if ITR(1)=1; otherwise, it will be ignored. IP is the print flag, which should normally be set to one to obtain output. This screen output can be redirected to a file for later use by including the proper option on the command line at the time of execution. If IP=0, the standard output will not be generated.

5. Program Execution

After ensuring the appropriate files are properly edited and are present in the directory, type:

>incompbl <incompbl.dat> incompbl.out

Several new files will appear in the directory after running the program. Any write statements incorporated into the program using the default numbering format will produce files such as **FOR032.DAT** (Stardent) for **fort.32** (IRIS). The output file **incompbl.out** will have a comprehensive summary of the input data, and output data such as C_p, C_D, C_L, C_M, and boundary layer properties, including skin friction coefficient and displacement thickness.

V. EXPERIMENT

A. BACKGROUND

The experimental work described in this chapter was performed as part of a continuing series of investigations by M. S. Chandrasekhara and L. W. Carr in the Compressible Dynamic Stall Facility (CDSF) of the Fluid Mechanics Laboratory (FML) at NASA Ames Research Center. The steady flow density field around a NACA 0012 airfoil at a Reynolds number of 540,000 was photographed using the technique of Point Diffraction Interferometry [Ref. 7].

A sample photograph, called an interferogram, is shown in Figure 5.1. The bright and dark areas emanating from the airfoil are called fringes, and each one represents a line of constant density. The stagnation point may be easily identified as the center of the smallest fringe on the lower surface (for a positive angle of attack). The flow accelerates around the leading edge of the airfoil. The example shown also reveals the presence of a laminar separation bubble just aft of the leading edge, distinguished by a characteristic fringe pattern.

For a given freestream Mach number, the fringes may also be correlated with particular Mach numbers and pressures in isentropic flow. For a standard interferometer and two dimensional flow, the path length difference ΔPL due to density changes can be related to the fringe number ϵ:

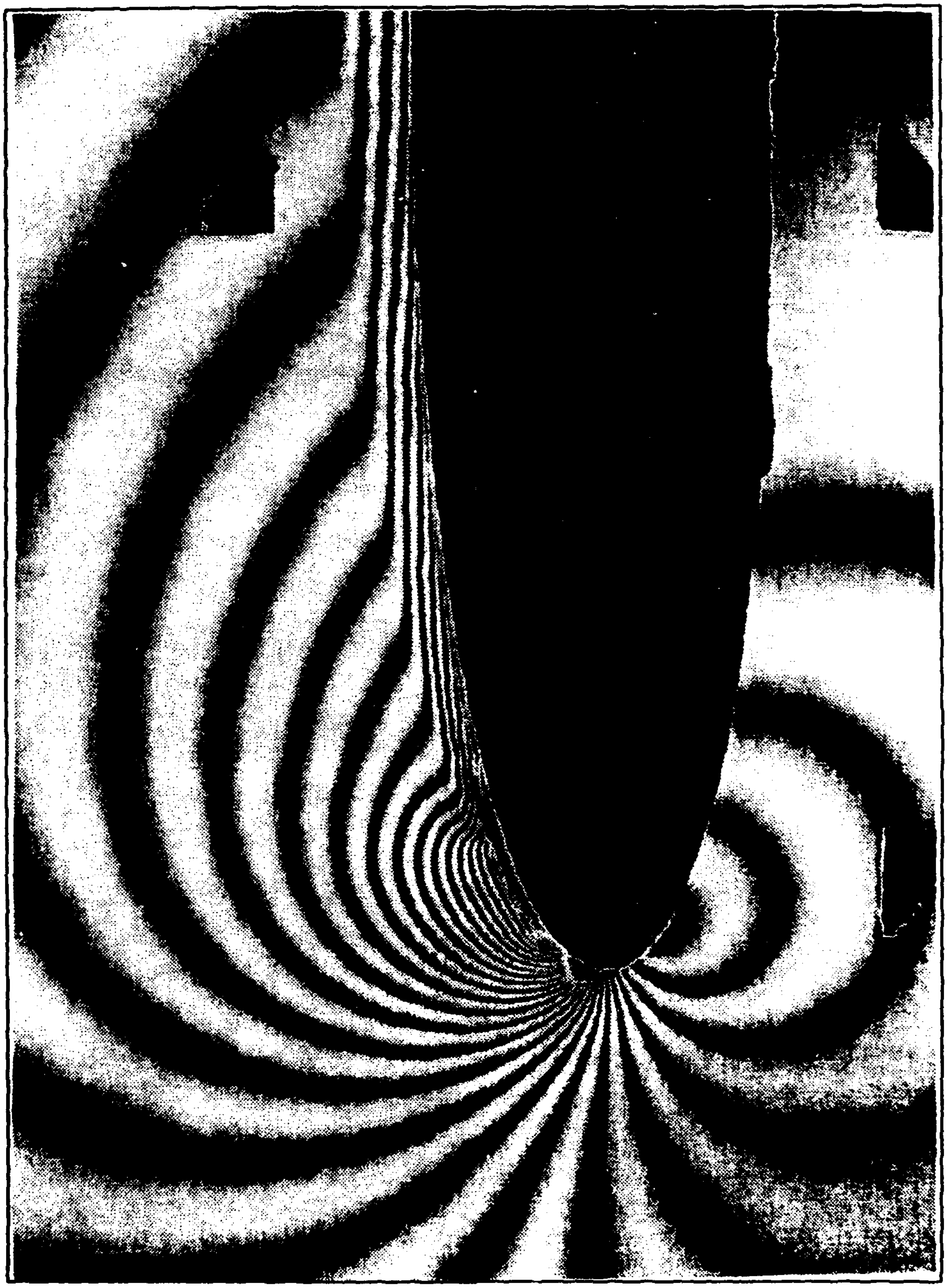

Figure 5.1 Interferogram of NACA 0012 Airfoil, AOA=10°

$$\epsilon = \frac{\Delta PL}{\lambda_0} = (n - n_{ref})\frac{L}{\lambda_0} \qquad (5.1)$$

where n is the refractive index of the signal beam, n_{ref} refers to the reference beam, λ_0 is the wave length of light used, and L is the test section span. An integer value for ϵ results in a bright fringe, while an integer plus one half corresponds to a dark fringe. Using the Gladstone-Dale equation [Ref. 8] and the perfect gas equation, Equation 5.1 reduces to

$$\rho - \rho_\infty = \left(\frac{\lambda_0}{n_0 - 1}\right)\left(\frac{\epsilon \rho_0}{L}\right) = A\epsilon \qquad (5.2)$$

where ρ is the density of the fringe, ρ_∞ refers to freestream conditions, and A is a constant determined from the experimental parameters. With λ_0=532 nm, L=25 cm, $(n_0-1)=2.733 \times 10^{-4}$, and total or stagnation density ρ_0=1.21 kg/m^3, A=0.009421 kg/m^3. Dividing by ρ_0,

$$\frac{\rho}{\rho_0} = \frac{\rho_\infty}{\rho_0} + \frac{A\epsilon}{\rho_0} \qquad (5.3)$$

Using the relationship

$$\frac{\rho}{\rho_0} = \left[1 + \frac{\gamma - 1}{2}M^2\right]^{\frac{1}{1-\gamma}} \qquad (5.4)$$

with $\rho = \rho_\infty$ and $M = M_\infty$, ρ_∞/ρ_0 is a function of freestream Mach number only for $\gamma = 1.4$. For the present case of M=0.3, the term ρ_∞/ρ_0 is constant at 0.956. Equation 5.3 may now be written as

$$\frac{\rho}{\rho_0} = 0.956 + 0.007786\epsilon \qquad (5.5)$$

Thus, quantitative density measurements are available knowing only the fringe number. The fringe numbers are determined by identifying the fringe number of the stagnation fringe as the highest fringe number, and counting down around the leading edge and upper surface of the airfoil. With $\rho/\rho_0 = 1$ in Equation 5.5, $\epsilon \approx 6$. Knowing the densities, pressures may be calculated in a straightforward manner:

$$\frac{p}{p_0} = \left(\frac{\rho}{\rho_0}\right)^{\gamma} \qquad (5.6)$$

Mach numbers for each fringe may be calculated using Equation 5.4.

The photographs for various angles of attack are identified using particular settings on the equipment. A correlation between settings and angles of attack is presented in Table 5.1.

Table 5.1 CORRELATION OF EXPERIMENTAL SETTINGS AND AOA

DEGREES	COUNT	DEGREES	COUNT	DEGREES	COUNT
0.5	6	5.5	63	10.5	119
1	11	6	68	11	125
1.5	17	6.5	74	11.5	131
2	23	7	80	12	137
2.5	28	7.5	85	12.5	142
3	34	8	91	13	148
3.5	40	8.5	97	13.5	154
4	46	9	102	14	159
4.5	51	9.5	108	14.5	165
5	57	10	114	15	171

B. IMAGE PROCESSING

1. Scanning

The processing task began with a set of experimental interferograms at angles of attack ranging from 0° to 10°. Each was scanned into an IRIS computer using the program **pixscan** at the Numerical Aerodynamic Simulation Facility at NASA Ames Research Center. Options for contrast enhancement (gamma correction) and grayscale were used. A photograph of the airfoil in no-flow conditions was also scanned. All of the photographs were carefully placed on the scanner in a position such that the top surfaces of the two bottom reference triangles made a horizontal line even with the edge of the scanning window. The resulting scanned images were transferred via **ftp** to the IRIS computer at the Naval Postgraduate School for further processing.

2. Editing

Using a program called **pixedit**, the original flowfield images were first overlaid with the airfoil image. This permitted a more defined surface for analysis when the diffraction caused by the interferometry technique distorted the airfoil surface. The images were then cropped closer to the airfoil. This procedure reduced the number of pixels in the image, which was necessary for use with the digitizing program. A small section near the leading edge was also selected for enlargement, thus giving better detail of the laminar separation bubble region.

In anticipation of the digitization process, reference marks at known coordinates were added to the two images. The original photographs provided the basis for coordinate system definition. The three dark triangles are located as shown in Figure 5.2. The distance between horizontal or vertical faces of the triangles is 0.2 of the chord. The vertical faces of the two triangles on the right are located at 0.25 chord. Thus, the coordinates of the point at the right angle of the bottom right triangle are (0.25, -0.1). A reference mark is needed in the top left corner for use with the digitization program, but none is provided. Therefore, a mark was constructed with several applications of the **snap new image** feature of **pixedit**. Very thin horizontal and vertical lines were initially saved as separate images. For each interferogram, these images were read in and placed to be even with existing reference marks of the airfoil leading edge (vertical line) and the bottom edge of the upper triangle (horizontal line). The small area in the top left corner where the lines crossed was saved as another image and the long lines were deleted. The coordinates of the cross mark are (0.0, 0.1). Finally, the complete image was saved for further processing. A similar procedure was used to make a cross mark at (0.05, 0.0) for the lower right reference on the enlarged leading edge images.

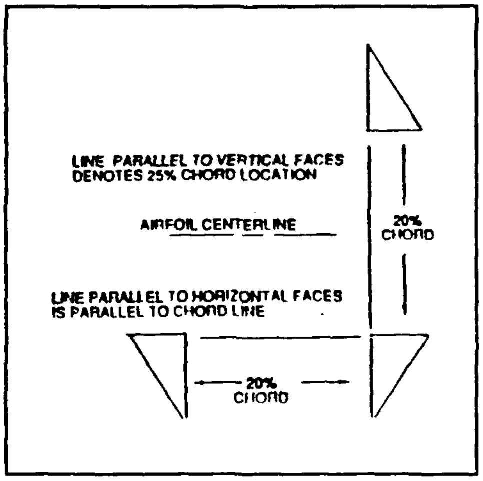

Figure 5.2 Reference Triangles

3. Fringe Tracing

Both of the new images, the density flowfield and the enlarged leading edge, were digitized using the program **DigiCurv**. The appropriate lower right and upper left coordinates were provided and the corresponding points on the image were selected with the mouse. With the coordinate system thus initialized, each curve was digitized separately. The centerline of the dark fringe was used for digitization. A representative number of points were chosen for each fringe using the left button of the mouse, always starting with the point where the fringe intersected the airfoil. Depressing the right button activated a menu with a curve fit option. The program calculated a best fit curve to the chosen points using a spline routine and output up to 30 (default value) new points to describe the curve. The fitted curve was also displayed on the screen for acceptance. In most cases, the computed curve had *outstanding agreement* with the centerline of the dark fringe, even when a sharp corner was involved near the end of a separation bubble. If the displayed curve needed refinement, the **backup** option removed points one by one, more closely spaced points were selected, and the curve was refit. After all dark fringes were digitized for an image, the data point output file was saved for plotting.

The digitized data can be viewed using any plotting routine compatible with sequential data separated by blank lines. The program **xyplot** does not recognize blank lines and is not a good choice. The program **gnuplot** will properly accept data in this format. The digitized plot corresponding to the interferogram in Figure 5.1 is shown in Figure 5.3.

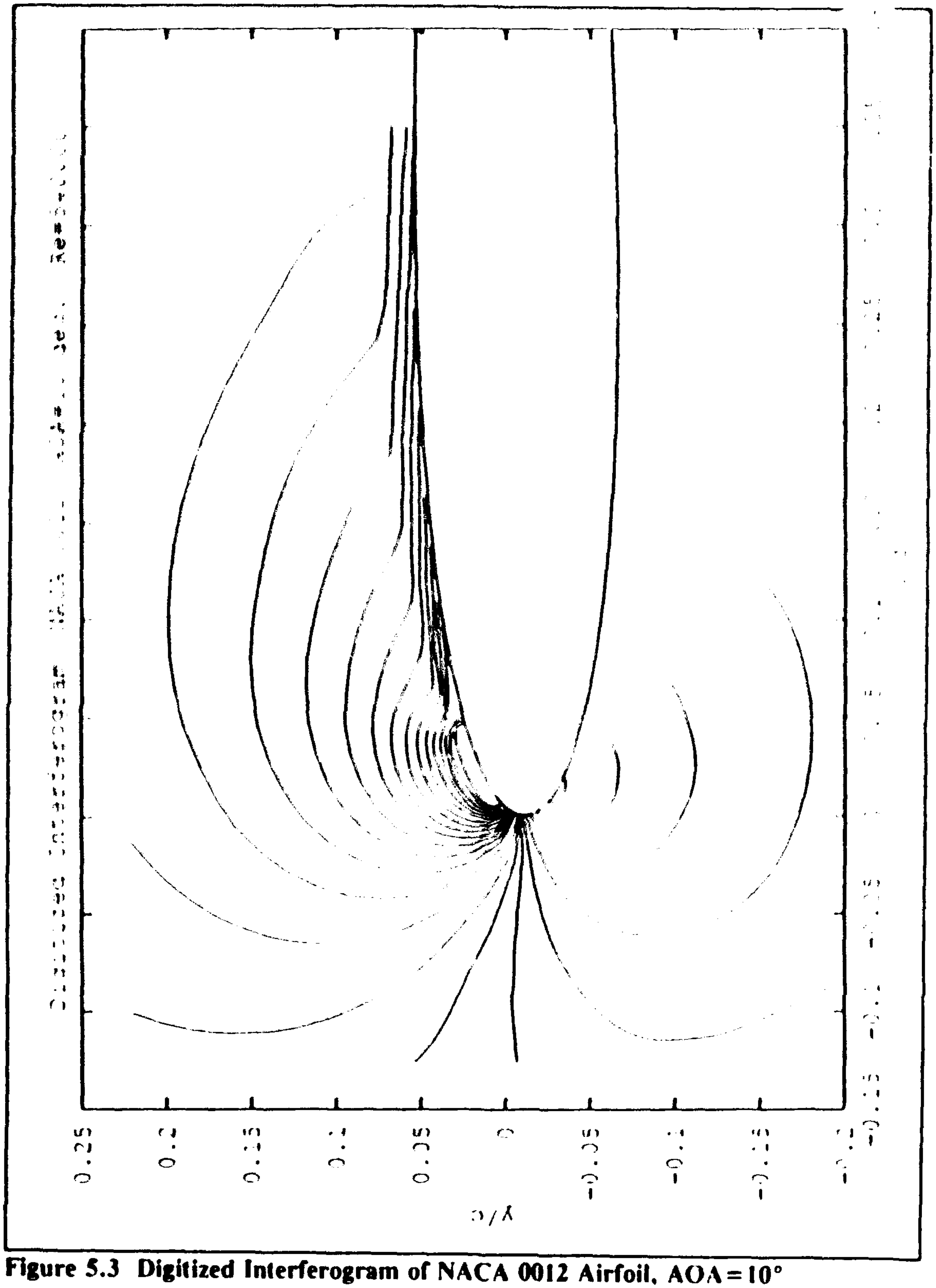

Figure 5.3 Digitized Interferogram of NACA 0012 Airfoil, AOA=10°

4. Printing

To obtain graphical output on a postscript laser printer, the files must be in postscript format. The digitized plots are in this format and can be easily printed using the following commands in the Advanced Computation Laboratory at the Naval Postgraduate School. From the Stardent, type

> **rglp** ***filename***

From the IRIS, type

> **lp -dlaser** ***filename***

The filenames may have the suffix **ps** to indicate that they are postscript files, but this is for reference only and is not required by the printer.

The images output from **pixedit** are not directly compatible with the postscript printer. Using the IRIS computer, they must first be converted to sgi format:

> **pix2sgi** ***infilename outfilename.sgi***

Again, the suffix is optional. The next step is a conversion to postscript format:

> **tops** ***infilename.sgi*** **-p 98.0 >** ***outfilename.ps***

All suffixes are optional. The **-p 98.0** option preserves the proportions of the original image; otherwise, it may be distorted when stretched to fill up the printed page. The > symbol redirects the screen output to the specified output file. There are a number of other options available, including size and color. For a full description of options, type

> **man tops**

to obtain the on-line help manual entry on this conversion program.

C. ANALYSIS

The primary information desired from the interferograms was the location of the start of the laminar separation bubble, when it existed. Figure 5.4 shows an enlarged image at 6°, where a bubble is first formed. Using the digitized data file of this image, precise x/c locations of the intersections of the fringes with the airfoil surface may be determined, as is the first point digitized for each fringe. Starting with the stagnation fringe, the first dark fringe always has a number of 5.5 (for the present case of $M_\infty = 0.3$). Subsequent dark fringes have numb… 4.5, 3.5, etc. Negative fringe numbers are possible. Converting pressures to pressure coefficients (C_p), a table of fringe numbers and corresponding C_p's may be produced and used for all interferograms with the same freestream Mach number. The program pres (Appendix A) was written to process the data as described and output a plotting file of x/c vs. C_p. Figure 5.5 shows the digitized image at 6° and the corresponding C_p plot. The most important feature is the pressure plateau, which reveals the existence and location of a separation bubble. The bubble starts at a chord location of 0.015. Interferograms for other angles of attack were analyzed in the same manner. Figure 5.6 shows a comprehensive plot for an angle of attack range from 6° to 10°. As the angle of attack increases, the length of the bubble increases, and the starting location moves slightly forward.

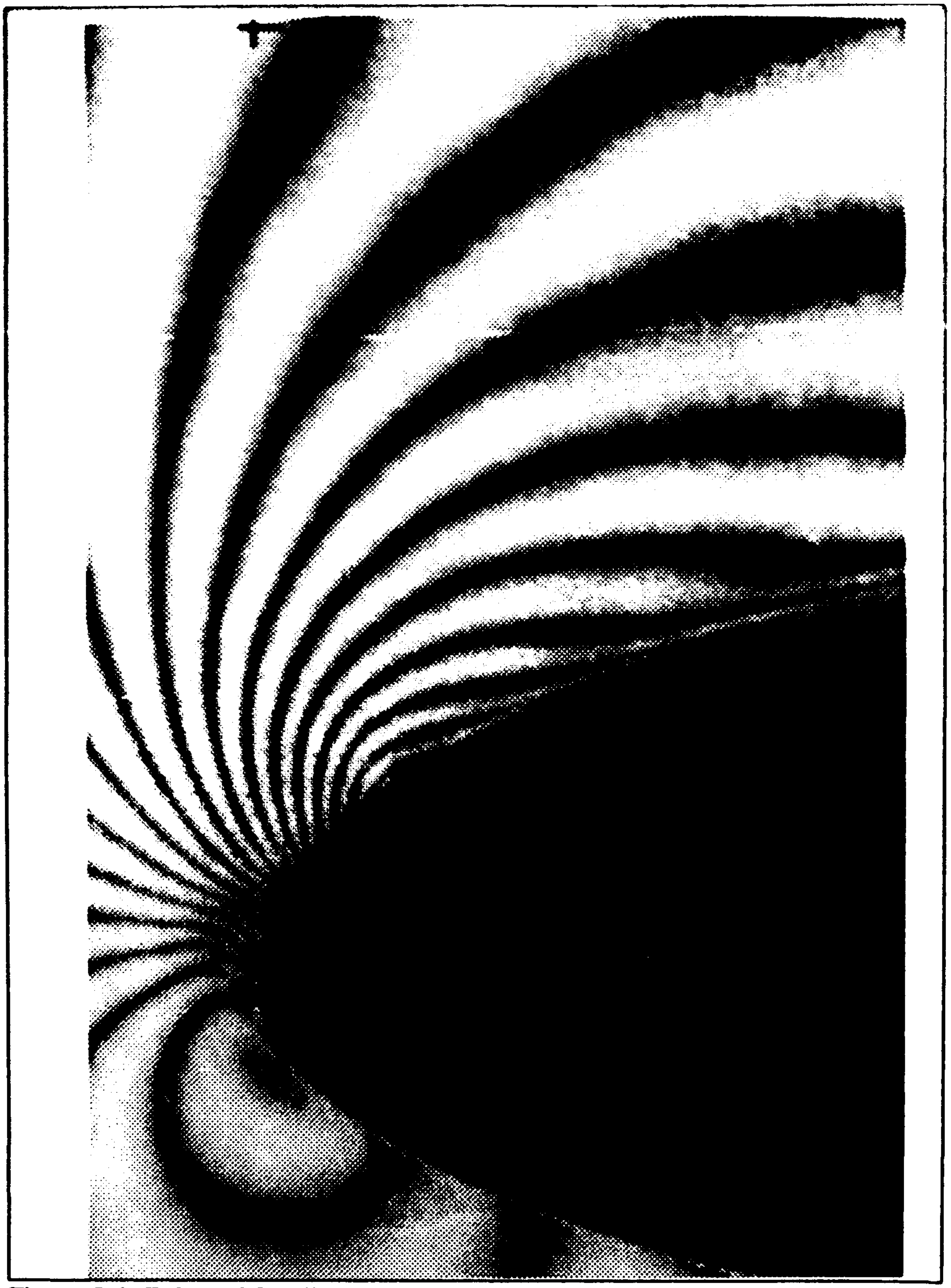

Figure 5.4 Enlarged Leading Edge Interferogram, AOA=6°

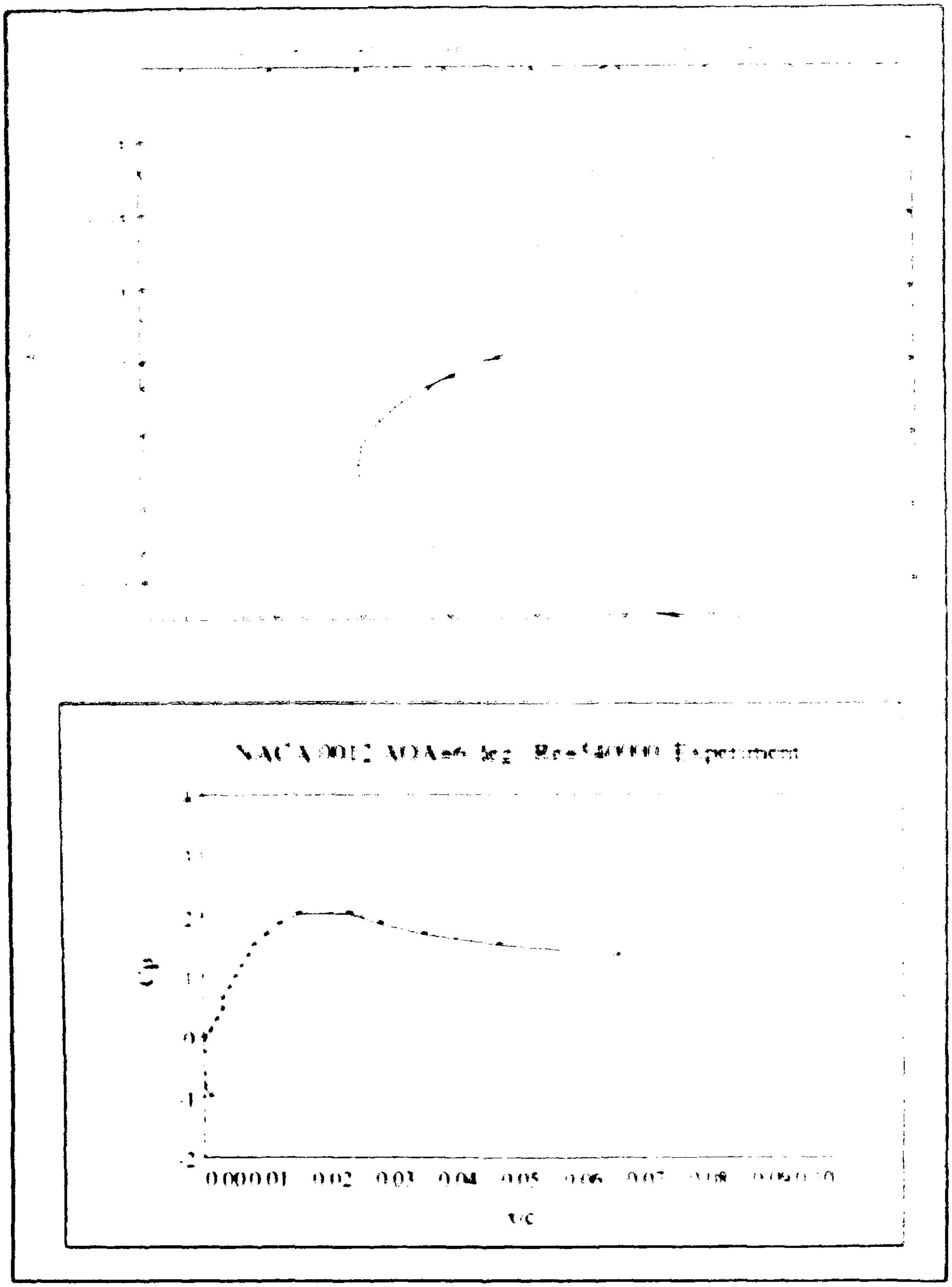

Figure 5.5 Digitized Interferogram and C_p Plot, AOA=6°

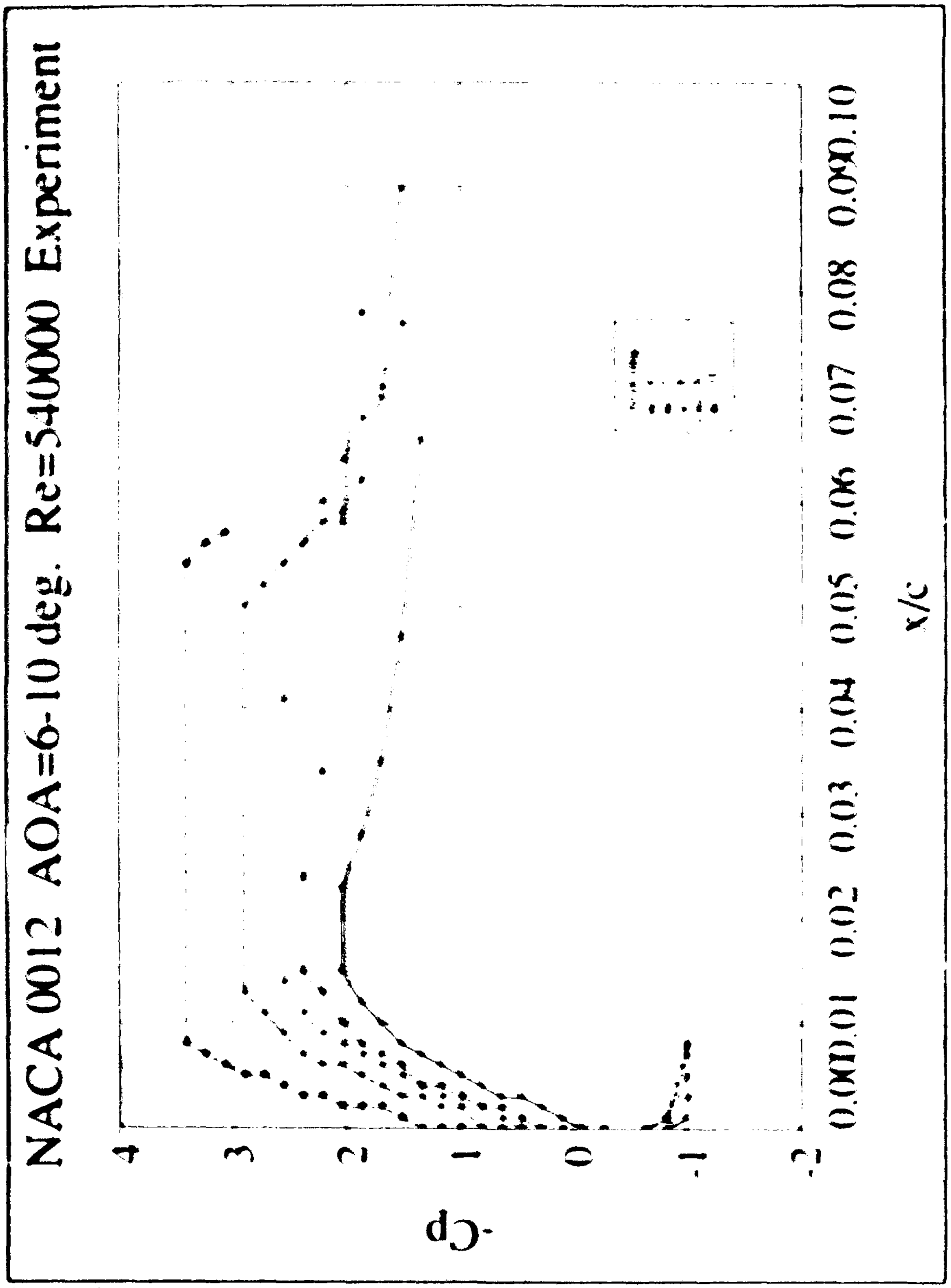

Figure 5.6 C_p Plots Showing Separation Bubble, AOA=6-10°

D. COMPARISON OF RESULTS TO COMPUTATION

Table 5.2 compares the experimental results to those obtained by **bl2d** and **incompbl**. The direct boundary layer code shows a very small separation bubble at 4°; however, none is present in the experiment. At 6°, the bubble's existence is correctly detected, but the computational location is slightly aft of the experimental value. The higher angles of attack show excellent agreement, with differences of only 0.3% of chord. In contrast, the viscous-inviscid code failed to predict any separation until 13.3°. Since this is past the steady stall angle of 12.4°, there is no experimental bubble to compare with. Even if stall had not occurred, the trend clearly indicates that the location would be in great error as well.

Table 5.2 COMPARISON OF BUBBLE START LOCATIONS

AOA	BL2D	INCOMPBL	EXPERIMENT
0	---	---	---
2	---	---	---
4	.245	---	---
6	.038	---	.015
8	.017	---	.014
10	.012	---	.009
13.3	NO SOLN	.024	STALL

VI. NAVIER-STOKES CODE

A. OVERVIEW

A time-averaged Navier-Stokes (NS) code called ns2 was used for a final study. This method has the advantage of including compressibility effects. While at a Mach number of 0.3 the effects are small, it is just on the border of the region that is usually considered acceptable for the assumption of incompressibility. A disadvantage of this method, as mentioned in the transition discussion of Chapter 2, is the lack of a transition model. Turbulent flow is assumed throughout the flowfield. The code also takes three to four hours to run on a Stardent computer. Nevertheless, Navier-Stokes codes are often regarded as the best computation method currently available and warrant consideration. The details of the time-averaged NS equations, their derivation, and their discretization are well documented elsewhere [Ref. 9] and will not be reviewed here. The particular code used for this investigation was developed by J. A. Ekaterinaris of the Navy-NASA Joint Institute of Aeronautics.

B. RESULTS

The code was run for a NACA 0012 airfoil, with a Mach number of 0.3 and a Reynolds number of 540,000. A 161x64 C-type grid with a very fine distribution normal to the surface in the viscous region was used. The grid is shown in Figure 6.1. The Baldwin-Lomax model was used for turbulence modeling. The program was run to 2000

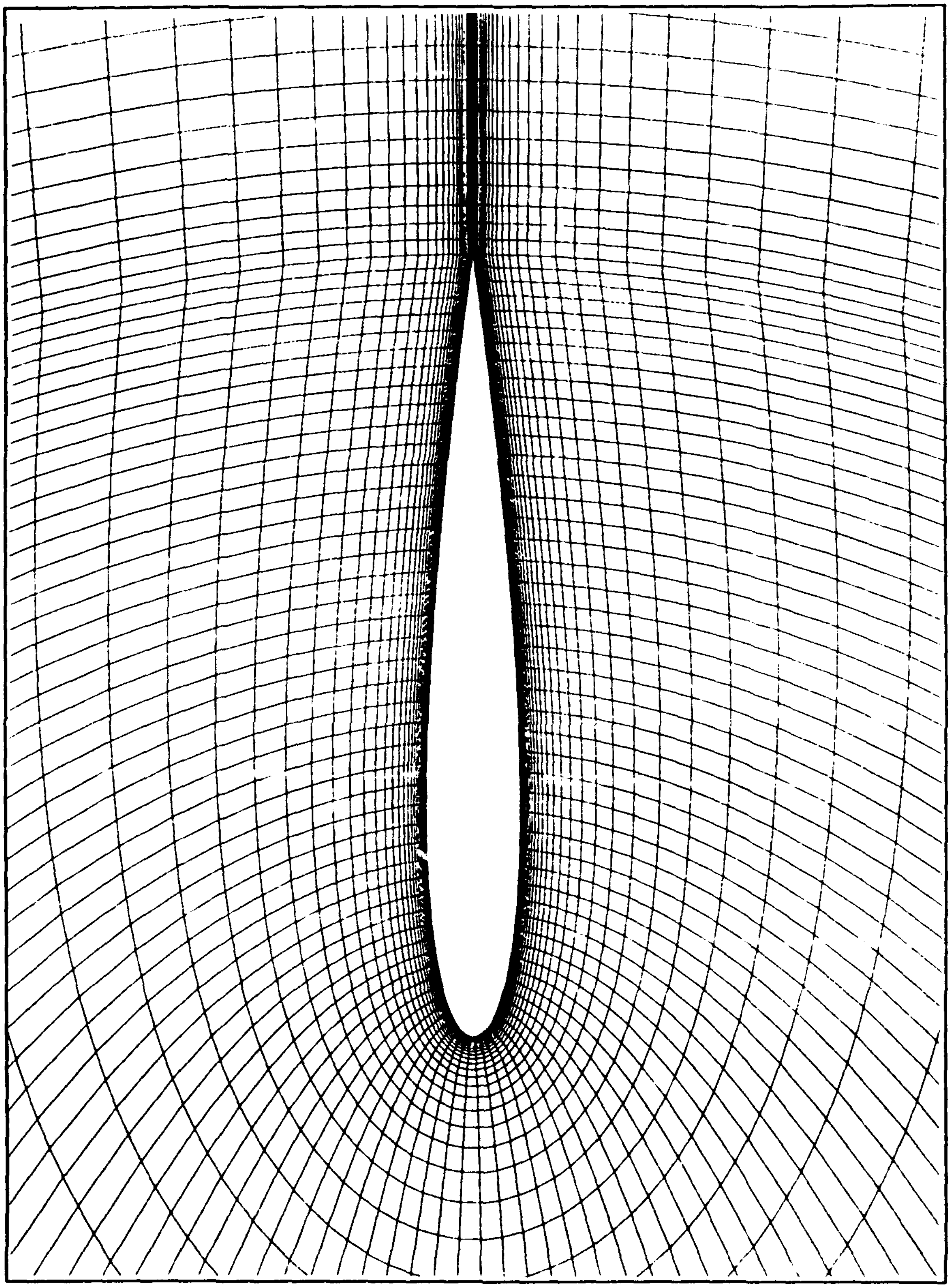

Figure 6.1 161x64 Viscous Grid for NACA 0012 Airfoil

iterations, where the residuals had dropped two orders of magnitude. Figures 6.2 to 6.4 show comparisons of the computational density fields to the experimental interferograms. At 0°, there is no separation bubble and the agreement appears to be good. At 6°, the experimental bubble first appears at $x/c = 0.015$. All of the computational density curves emanate from the leading edge. This is consistent with the lack of a transition model. At 10°, both methods show a large bubble. The NS bubble is much further aft. There is also a small extra bubble near the leading edge. It appears that the "state of the art" method is not always best for representing reality, particularly for low Reynolds number flows. The author is, however, currently working on a version of the program which will incorporate transition calculations and preliminary work indicates that results will be much better.

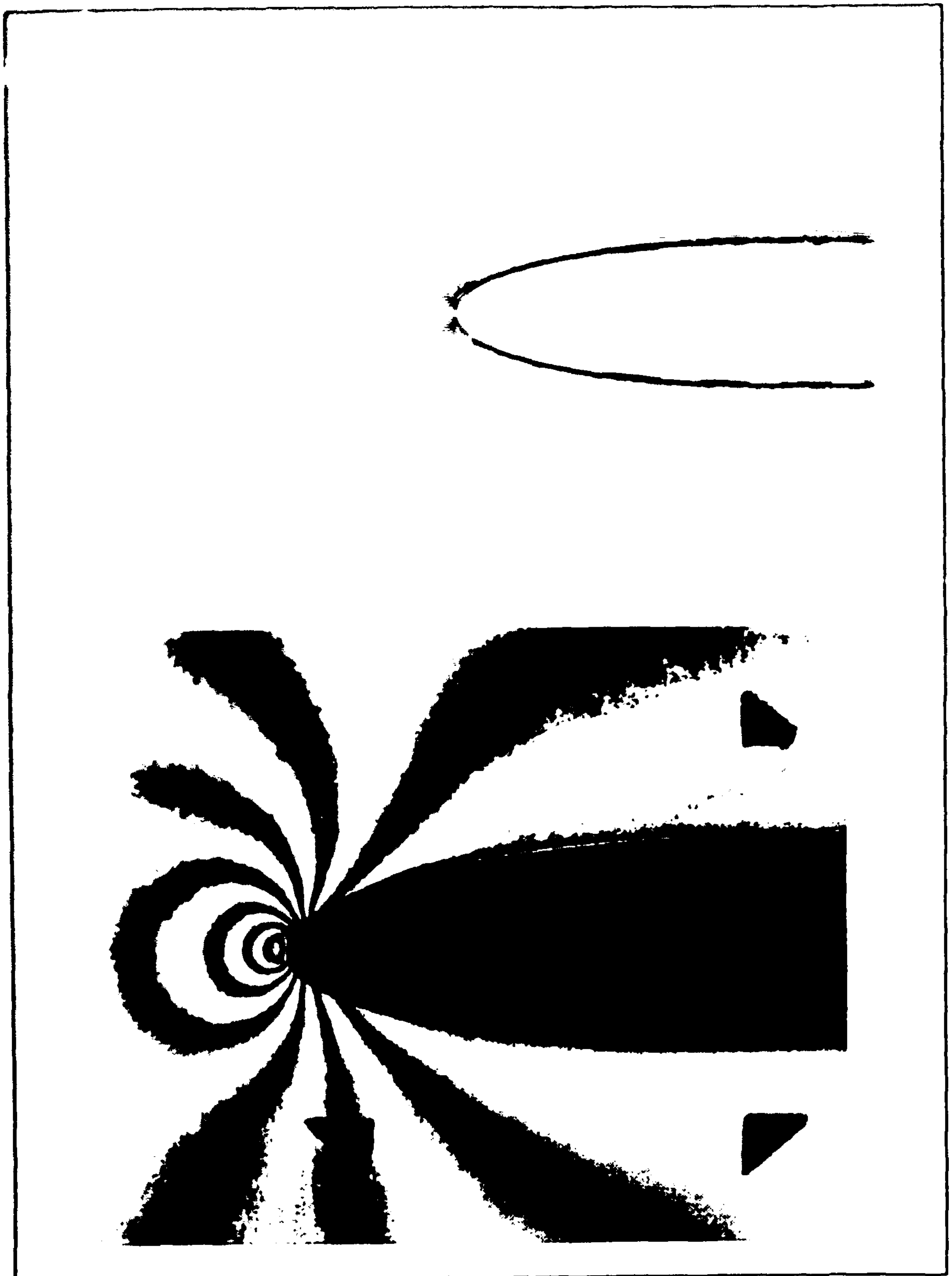

Figure 6.2 Comparison of NS and Experiment, AOA=0°

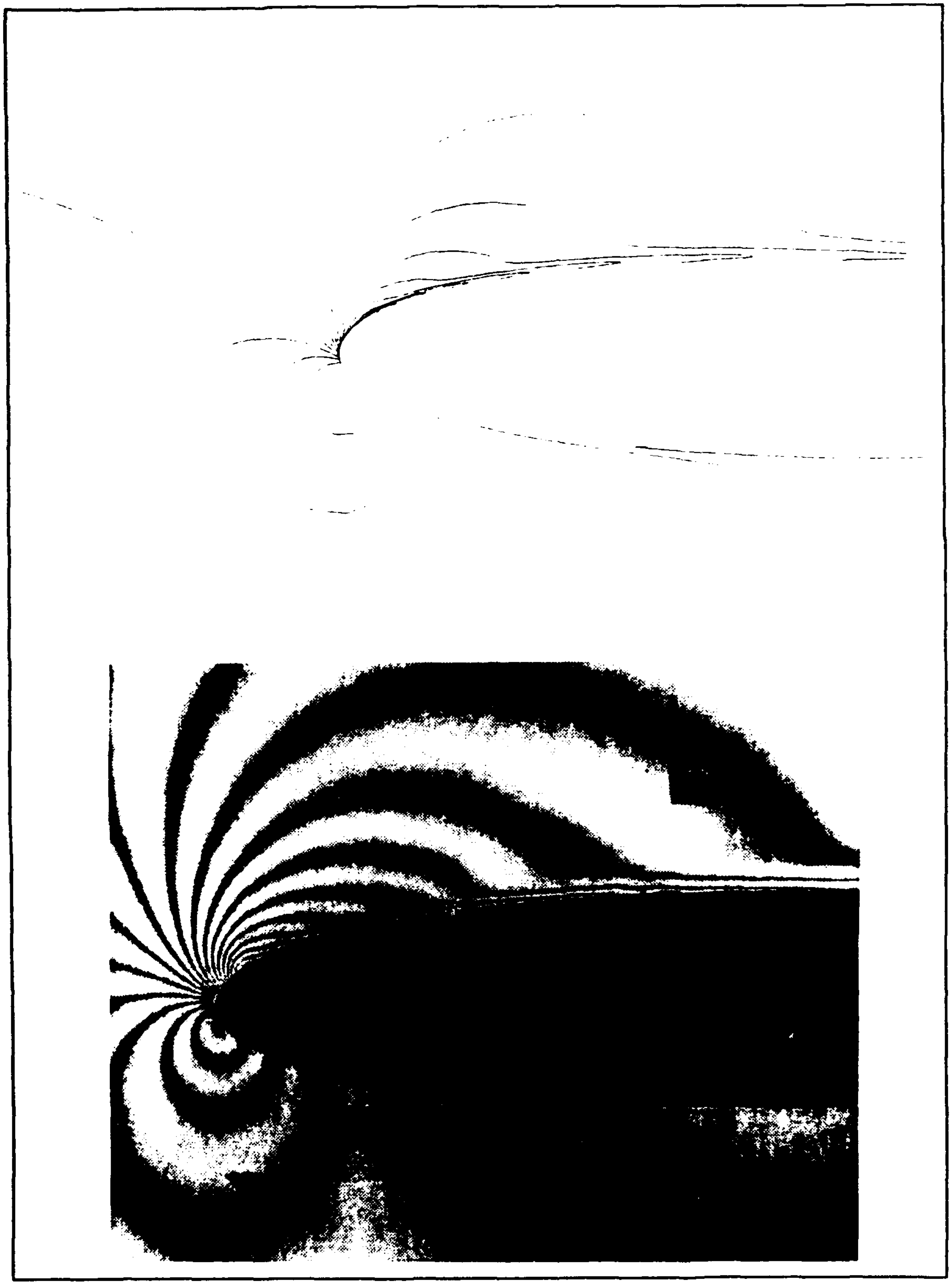

Figure 6.3 Comparison of NS and Experiment, AOA=6°

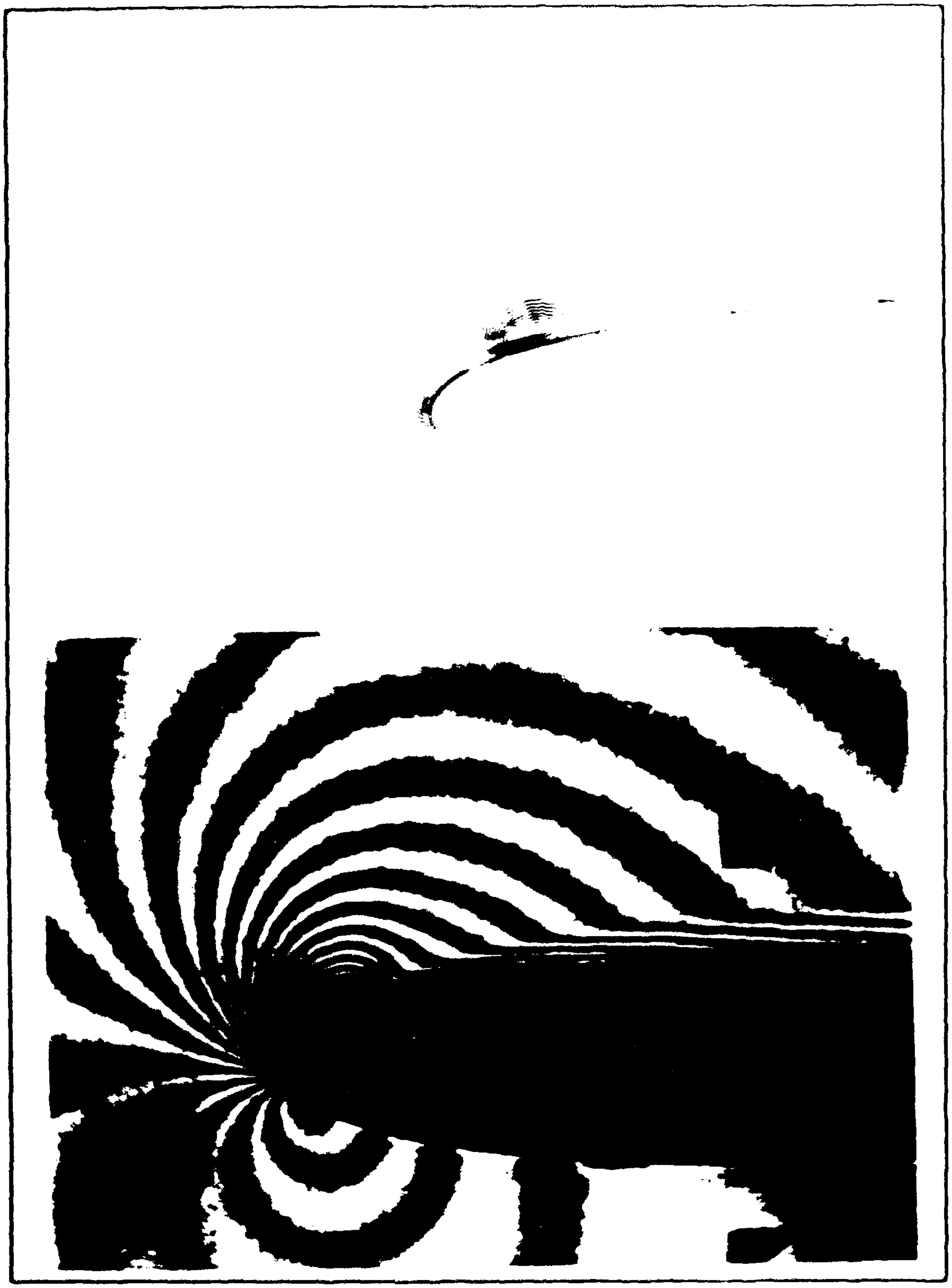

Figure 6.4 Comparison of NS and Experiment, AOA = 10°

VII. CONCLUSIONS

In the computational investigations of a NACA 0012 airfoil in low Reynolds number flows, several important discoveries have been made. First, even though Navier-Stokes codes are the most advanced computational method currently available, they are not always appropriate. Specifically, a transition model is necessary for the case of low Reynolds number flows. Even if accurate results may be obtained, the method is not practical for many applications due to its high cost in time and money. The viscous-inviscid method seems to offer very advanced calculations at a very inexpensive price. The problem with this code used alone, however, is that it simply does not give correct results for separation bubbles in low Reynolds number flows. A compressibility correction in a future version may alleviate the problem. Using a stability/transition method in conjunction with the code will certainly provide more refined transition estimates, which may influence separation bubble results. Finally, the relatively simple and often overlooked direct boundary layer method can provide meaningful information about separation bubbles in low Reynolds number flows. Given that the code is also very fast, it may be used efficiently in the design stages and quality assurance checking of many aeronautical applications.

Successful or not, all current computational methods have several important limitations. All are dependent on empirical models for transition onset, transition length, and turbulence. The models are often formulated for specific parameter ranges outside of which agreement is poor. Until there is sufficient computing power available to solve the *full* Navier-Stokes codes, it is imperative to check the applicability of the models and the assumptions of the equations on which a method is based before using a program and counting on the results to be reliable.

LIST OF REFERENCES

1. Eppler, R., *Airfoil Design and Data*, Springer-Verlag, 1990.

2. Anderson, J. D., *Fundamentals of Aerodynamics*, 2d ed., McGraw-Hill, Inc., 1991.

3. Cebeci, T., and Bradshaw, P., *Momentum Transfer in Boundary Layers*, Hemisphere Publishing Corporation, 1977.

4. Neace, K. S., *A Computational and Experimental Investigation of the Propulsive and Lifting Characteristics of Oscillating Airfoils and Airfoil Combinations in Incompressible Flow*, Engineer's Thesis, Naval Postgraduate School, Monterey, CA, September 1992.

5. Walker, G. J., Subroto, P. H., and Platzer, M. F., "Transition Modeling Effects on Viscous/Inviscid Interaction Analysis of Low Reynolds Number Airfoil Flows Involving Laminar Separation Bubbles," ASME Paper No. 88-GT-32 presented at the Gas Turbine and Aeroengine Congress, Amsterdam, The Netherlands, 6-9 June, 1988.

6. California State University Aerospace Engineering Department Report AE-90-2, *An Interactive Boundary-Layer Stability-Transition Method for Calculating Flow Over Hydrofoils*, by T. Cebeci, H. H. Chen, and H. M. Jang, August 1990.

7. Carr, L. W., Chandrasekhara, M. S., Brock, N. J., and Ahmed, S., "A Study of Dynamic Stall Using Real-Time Interferometry," AIAA Paper No. 91-0007 presented at the 29th Aerospace Sciences Meeting, Reno, NV, 7-10 January, 1991.

8. Goldstein, R. J., *Fluid Mechanics Measurements*, Hemisphere Publishing Corporation, 1983.

9. Anderson, D. A., Tannehill, J. C., and Pletcher, R. H., *Computational Fluid Mechanics and Heat Transfer*, Hemisphere Publishing Corporation, 1984.

APPENDIX A: COMPUTER PROGRAMS AND FILES

The following programs and input/output files are presented in logical order of use. The input and output files are examples only. The input values may be modified as described in the appropriate User's Guide section. The program **incompbl** comprises 90 pages in length and was not modified other than inserting WRITE statements to extract data for plotting; thus, the source code is not included.

Input file: points.dat

```
1.000000  0.000000
0.999013 -0.000141
0.996057 -0.000562
0.991144 -0.001258
0.984292 -0.002222
0.975528 -0.003443
0.964888 -0.004909
0.952414 -0.006603
0.938153 -0.008510
0.922164 -0.010610
0.904508 -0.012883
0.885257 -0.015310
0.864484 -0.017868
0.842274 -0.020535
0.818712 -0.023291
0.793893 -0.026111
0.767913 -0.028974
0.740877 -0.031856
0.712890 -0.034733
0.684062 -0.037582
0.654508 -0.040378
0.624345 -0.043094
0.593691 -0.045705
0.562667 -0.048182
0.531395 -0.050499
0.500000 -0.052625
0.468605 -0.054534
0.437333 -0.056195
0.406309 -0.057581
0.375655 -0.058666
0.345491 -0.059424
0.315938 -0.059834
0.287110 -0.059876
0.259123 -0.059535
0.232087 -0.058799
0.206107 -0.057661
0.181288 -0.056119
0.157726 -0.054176
0.135516 -0.051839
0.114743 -0.049121
0.095491 -0.046037
0.077836 -0.042608
0.061847 -0.038854
0.047586 -0.034800
0.035112 -0.030471
0.024472 -0.025893
0.015708 -0.021088
0.008856 -0.016078
0.003943 -0.010884
0.000987 -0.005521
0.000000  0.000000
0.000987  0.005521
0.003943  0.010884
0.008856  0.016078
0.015708  0.021088
0.024472  0.025893
0.035112  0.030471
0.047586  0.034800
0.061847  0.038854
0.077836  0.042608
0.095491  0.046037
0.114743  0.049121
0.135516  0.051839
0.157726  0.054176
0.181288  0.056119
0.206107  0.057661
0.232087  0.058799
0.259123  0.059535
0.287110  0.059876
0.315938  0.059834
0.345492  0.059424
```

```
0.375655  0.058666
0.406309  0.057581
0.437333  0.056195
0.468605  0.054534
0.500000  0.052625
0.531395  0.050499
0.562667  0.048182
0.593691  0.045705
0.624345  0.043094
0.654509  0.040378
0.684062  0.037582
0.712890  0.034733
0.740877  0.031856
0.767913  0.028974
0.793893  0.026111
0.818712  0.023291
0.842274  0.020535
0.864484  0.017868
0.885257  0.015310
0.904509  0.012883
0.922164  0.010610
0.938153  0.008510
0.952414  0.006603
0.964888  0.004909
0.975528  0.003443
0.984292  0.002222
0.991144  0.001258
0.996057  0.000562
0.999013  0.000141
1.000000  0.000000
```

Source code: panel.f


```
      PROGRAM PANEL

*  AUTHOR:  L. M. NOWAK
*    DATE:  6 NOV 91    modified:  MAY, AUG 1992
* PURPOSE:  CALCULATE THE VELOCITIES ON AN AIRFOIL USING A PANEL METHOD.

*     LIM:  Arrays currently dimensioned for maximum of N=200 panels
*           Input data file points.dat will have N+1 points
*           Output velocities are referenced to freestream, ie. V/Vinf

*  METHOD:  FLOWFIELD CONSISTS OF THREE SIMPLER FLOWS:  FREESTREAM, SOURCE,
*           AND VORTICITY.  SOURCE DISTRIBUTIONS q(j) VARY FROM PANEL TO
*           PANEL.  VORTICITY STRENGTH GAMMA IS THE SAME FOR ALL PANELS.
*           BOUNDARY CONDITIONS INCLUDE FLOW TANGENCY AT CONTROL POINTS AND
*           KUTTA CONDITION FOR FIRST AND LAST PANELS.  INFLUENCE
*           COEFFICIENTS COMBINED TO FORM NEW COEFFICIENTS IN LINEAR SYSTEM
*           OF n+1 EQUATIONS, n+1 UNKNOWNS (q(1)...q(n), GAMMA).  VELOCITIES
*           AT CONTROL POINTS EVALUATED FROM q(j) AND GAMMA.

      REAL      X(1:202),Y(1:202),XM(1:202),YM(1:202),
     :          At(1:202,1:202),Bt(1:202,1:202),
     :          a(1:202,1:202),b(1:202),
     :          q(1:202),Vt(1:202),ALPHA,V,VtC(1:202),
     :          PI,GAMMA,THETA(1:202),NUM,DEN,
     :          R(1:202,1:202),BETA(1:202,1:202),NUM1,DEN1,NUM2,DEN2,
     :          AAUG(1:202,1:202),An(1:202,1:202),Bn(1:202,1:202)


* NUMBER OF NODES ON AIRFOIL SURFACE:
*       PRINT*,'INPUT NO. OF PANELS (1 less than #lines in points.dat):'
*       READ *,N

      PI=ACOS(-1.)

      OPEN (UNIT=88,FILE='points.dat',STATUS='UNKNOWN')
      OPEN (UNIT=89,FILE='veloc.dat',STATUS='UNKNOWN')
      OPEN (UNIT=91,FILE='cp.dat',STATUS='UNKNOWN')
      OPEN (UNIT=90,FILE='bl2d.dat',STATUS='UNKNOWN')

      print *,'INPUT REYNOLDS NUMBER:'
      READ *,RL

      print *,'ENTER 0 IF TRANSITION LOCATIONS UNKNOWN'
      PRINT *,'      1 IF TRANSITION LOCATIONS KNOWN:'
      READ *,IANS

      IF(IANS.EQ.1) THEN
         PRINT *,'INPUT X/C TRANSITION LOCATION FOR UPPER SURFACE:'
         READ *, TRANSUPPER
         PRINT *,'INPUT X/C TRANSITION LOCATION FOR LOWER SURFACE:'
         READ *, TRANSLOWER
      ELSE
***These are arbitrary values intended to be downstream of the
***  actual transition points, for use with Michel's criterion in BL2D
         TRANSUPPER=.8
         TRANSLOWER=.999
      ENDIF

      WRITE (90,50) RL,TRANSUPPER,TRANSLOWER
      WRITE (94,50) RL,TRANSUPPER,TRANSLOWER
 50   FORMAT (F10.0,F10.4,F10.4)

      PRINT *,'INPUT ANGLE OF ATTACK IN DEGREES:'
      READ *,ALPHA
      ALPHA=ALPHA*PI/180.0


      DO 30 I=1,N+1
        READ (88,25) X(I),Y(I)
 25     FORMAT (2(F8.6,2x))
 30   CONTINUE
```

```

*This section defines the influence coefficients:

      DO I=1,N
        XM(I)=0.5*(X(I)+X(I+1))
        YM(I)=0.5*(Y(I)+Y(I+1))
        R(I,1)=(XM(I)-X(1))**2.+(YM(I)-Y(1))**2.
        DO J=1,N
          NUM=Y(J+1)-Y(J)
          DEN=X(J+1)-X(J)
          THETA(J)=ATAN2(NUM,DEN)
          NUM1=YM(I)-Y(J+1)
          DEN1=XM(I)-X(J+1)
          NUM2=YM(I)-Y(J)
          DEN2=XM(I)-X(J)
          BETA(I,J)=ATAN2((NUM1*DEN2-DEN1*NUM2),(DEN1*DEN2+NUM1*NUM2))
          R(I,J+1)=(XM(I)-X(J+1))**2.+(YM(I)-Y(J+1))**2.
          THETADIF=THETA(I)-THETA(J)
          IF (I.EQ.J)
     :      THEN
                 An(I,J)=0.5
                 Bn(I,J)=0.0
            ELSE
                An(I,J)=(1/(2*PI))*(SIN(THETADIF)*ALOG(R(I,J+1)/R(I,J))
     :                   *.5+COS(THETADIF)*BETA(I,J))
                Bn(I,J)=(1/(2*PI))*(COS(THETADIF)*ALOG(R(I,J+1)/R(I,J))
     :                   *.5-SIN(THETADIF)*BETA(I,J))
          END IF
          At(I,J)=-Bn(I,J)
          Bt(I,J)=An(I,J)
        END DO
      END DO

* Matrix coefficients of linear system defined (a's and b's):

      a(N+1,N+1)=0.0
      DO I=1,N
        a(I,N+1)=0.0
        DO J=1,N
          a(I,J)=An(I,J)
          a(I,N+1)=a(I,N+1)+Bn(I,J)
        END DO
        b(I)=-1.0*SIN(ALPHA-THETA(I))
        a(N+1,I)=At(1,I)+At(N,I)
        a(N+1,N+1)=a(N+1,N+1)+Bt(1,I)+Bt(N,I)
      END DO
      b(N+1)=-1.0*(COS(ALPHA-THETA(1))+COS(ALPHA-THETA(N)))

* Define augmented matrix for input to linear solver subroutine GAUSS

      DO I=1,N+1
        DO J=1,N+1
          AAUG(I,J)=a(I,J)
        END DO
        AAUG(I,N+2)=b(I)
      END DO

      CALL GAUSS(N+1,AAUG)

* Define source and vorticity strengths:

      DO I=1,N
        q(I)=AAUG(I,N+2)
      END DO

      GAMMA=AAUG(N+1,N+2)

* Calculate velocity on each panel at control point

      NSTAGFLAG=0
      ISTAG=0
```

```
      DO I=1,N
        Vt(I)=0.0
        DO J=1,N
          Vt(I)=At(I,J)*q(J)+GAMMA*Bt(I,J)+Vt(I)
        END DO
        Vt(I)=Vt(I)+COS(ALPHA-THETA(I))
        Cp=1.0-Vt(I)**2
c        WRITE (25,*) XM(I),Vt(I)
        IF ((Vt(I).GT.0) .AND. (NSTAGFLAG.EQ.0)) THEN
           ISTAG=I
           NSTAGFLAG=1
        ENDIF
        IF (Vt(I).LT.0) Vt(I)=-Vt(I)
        WRITE (89,45) XM(I),Vt(I)
        WRITE (91,45) XM(I),-Cp
      END DO

45    FORMAT (2(F10.5,2X))
48    FORMAT (3(F10.5))
49    FORMAT (3I5)

      WRITE (90,49) N,ISTAG,IANS
      DO I=1,N
         WRITE (90,48) XM(I),YM(I),Vt(I)
      END DO

      print *,'CALCULATIONS COMPLETE'
      PRINT *,'OUTPUT FILES ARE veloc.dat, cp.dat, bl2d.dat'

      END

*************************************************************
* Gauss elimination procedure obtained from Numerical Methods text Ch.6

      SUBROUTINE GAUSS(N,Z)
      INTEGER PV
      REAL Z(1:202,1:203),E

      E=1.0
10    IF (1.0+E.GT.1.0) THEN
         E=E/2.0
         GOTO 10
      END IF
      E=E*2
      EPS2=2*E
      PRINT *,'                MACHINE EPSILON=',E

1005  DET=1

      DO 1010 I=1,N-1
         PV=I

         DO 1020 J=I+1,N
            IF (ABS(Z(PV,I))  .LT.  ABS(Z(J,I))) PV=J
1020     CONTINUE

         IF (PV.EQ.I) GOTO 1050

         DO 1040 JC=1,N+1
            TM=Z(I,JC)
            Z(I,JC)=Z(PV,JC)
            Z(PV,JC)=TM
1040     CONTINUE

1045     DET=-1*DET


1050     IF (Z(I,I).EQ.0) THEN
            GOTO 1200
         END IF
```

```
              DO 1060 JR=I+1, N
                 IF (Z(JR,I).NE.0) THEN
                    R=Z(JR,I)/Z(I,I)
                    DO 1075 KC=I+1,N+1
                       TEMP=Z(JR,KC)
                       Z(JR,KC)=Z(JR,KC)-R*Z(I,KC)
                       IF (ABS(Z(JR,KC)).LT.EPS2*TEMP) Z(JR,KC)=0.0

C                          !-- if the result of subtraction is smaller than
C                          !-- 2 times machine epsilon times the original
C                          !-- value, it is set to zero.

 1075               CONTINUE
                 END IF
 1060       CONTINUE
 1010    CONTINUE

         DO 1084 I=1,N
            DET=DET*Z(I,I)
 1084    CONTINUE

         PRINT *
         PRINT *,'DETERMINANT = ',DET
         PRINT *

         IF (Z(N,N).EQ.0) GOTO 1200
         Z(N,N+1)=Z(N,N+1)/Z(N,N)

         DO 1130 NV=N-1,1,-1
            VA=Z(NV,N+1)
            DO 1120 K=NV+1,N
               VA=VA-Z(NV,K)*Z(K,N+1)
 1120       CONTINUE
            Z(NV,N+1)=VA/Z(NV,NV)
 1130    CONTINUE
         RETURN

 1200    PRINT *,'MATRIX IS SINGULAR'
         PRINT *,'I=',I,'Z(I,I)=',Z(I,I)
         STOP
         END
```

Output/input file: bl2d.dat

```
540000.       0.380      0.762
 100   49      1
  0.99950   -0.00005    0.75382
  0.99750   -0.00030    0.80956
  0.99355   -0.00085    0.84584
  0.98765   -0.00170    0.87525
  0.97985   -0.00280    0.89440
  0.97015   -0.00415    0.91199
  0.95860   -0.00575    0.92807
  0.94525   -0.00755    0.94058
  0.93010   -0.00955    0.95235
  0.91330   -0.01170    0.96041
  0.89485   -0.01405    0.97159
  0.87480   -0.01655    0.98018
  0.85330   -0.01915    0.98850
  0.83045   -0.02185    0.99553
  0.80625   -0.02465    1.00410
  0.78085   -0.02750    1.01099
  0.75435   -0.03035    1.01700
  0.72680   -0.03325    1.02456
  0.69840   -0.03610    1.03023
  0.66925   -0.03890    1.03612
  0.63940   -0.04165    1.04173
  0.60895   -0.04435    1.04948
  0.57810   -0.04690    1.05502
  0.54695   -0.04925    1.05902
  0.51565   -0.05150    1.06629
  0.48430   -0.05355    1.07290
  0.45295   -0.05530    1.07664
  0.42180   -0.05680    1.08126
  0.39095   -0.05805    1.08640
  0.36050   -0.05900    1.09167
  0.33065   -0.05960    1.09584
  0.30150   -0.05980    1.09756
  0.27310   -0.05965    1.10065
  0.24555   -0.05910    1.10082
  0.21905   -0.05815    1.10060
  0.19365   -0.05685    1.10136
  0.16945   -0.05510    1.09665
  0.14660   -0.05295    1.09134
  0.12510   -0.05045    1.08489
  0.10505   -0.04755    1.07358
  0.08660   -0.04430    1.05957
  0.06980   -0.04070    1.03695
  0.05465   -0.03680    1.01271
  0.04130   -0.03260    0.96970
  0.02975   -0.02810    0.91371
  0.02005   -0.02340    0.82914
  0.01225   -0.01850    0.70489
  0.00635   -0.01340    0.49893
  0.00240   -0.00815    0.19469
  0.00045   -0.00275    0.24808
  0.00045    0.00275    0.67814
  0.00240    0.00815    1.03123
  0.00635    0.01340    1.20822
  0.01225    0.01850    1.29430
  0.02005    0.02340    1.32604
  0.02975    0.02810    1.33560
  0.04130    0.03260    1.33815
  0.05465    0.03680    1.33195
  0.06980    0.04070    1.32173
  0.08660    0.04430    1.31302
  0.10505    0.04755    1.30239
  0.12510    0.05045    1.29178
  0.14660    0.05295    1.27918
  0.16945    0.05510    1.27004
  0.19365    0.05685    1.25919
  0.21905    0.05815    1.24575
  0.24555    0.05910    1.23599
  0.27310    0.05965    1.22431
  0.30150    0.05980    1.21308
  0.33065    0.05960    1.20278
  0.36050    0.05900    1.19052
  0.39095    0.05805    1.17947
  0.42180    0.05680    1.16662
  0.45295    0.05530    1.15685
  0.48430    0.05355    1.14723
  0.51565    0.05150    1.13455
  0.54695    0.04925    1.12330
  0.57810    0.04690    1.11552
  0.60895    0.04435    1.10402
  0.63940    0.04165    1.09349
  0.66925    0.03890    1.08354
  0.69840    0.03610    1.07480
  0.72680    0.03325    1.06496
  0.75435    0.03035    1.05419
  0.78085    0.02750    1.04603
  0.80625    0.02465    1.03433
  0.83045    0.02185    1.02466
  0.85330    0.01915    1.01330
  0.87480    0.01655    1.00407
  0.89485    0.01405    0.99028
  0.91330    0.01170    0.97919
  0.93010    0.00955    0.96712
  0.94525    0.00755    0.95337
  0.95860    0.00575    0.93767
  0.97015    0.00415    0.91836
  0.97985    0.00280    0.89968
  0.98765    0.00170    0.87407
  0.99355    0.00085    0.83978
  0.99750    0.00030    0.79799
  0.99950    0.00005    0.75382
```

Source code: bl2d.f


```
*****************************************************************
*     Modifications:  L. M. NOWAK                               *
*     ver. 2                                                    *
*     16 July 1992:                                             *
*        Added write statements(30-40) in the "DO 175" loop to  *
*        output boundary layer profiles for plotting            *
*          (100 panel airfoil only)                             *
*     20 July 1992:  Added write statements(20-21) to output    *
*        CF(skin friction) and DLS(delta star) for plotting     *
*     ver. 3                                                    *
*     31 Aug 1992:  Added calculation for onset of transition   *
*        based on Michel's criterion, added input ITRANS        *
*     ver. 4                                                    *
*      3 Sept 1992:  Modified boundary layer profile output     *
*        to be compatible with airfoil of any number points     *
*     ver. 5                                                    *
*      3 Sept 1992:  Redimensioned all arrays to accept         *
*        airfoil up to 200 panels                               *
*****************************************************************
      SUBROUTINE BL
      COMMON /BLC2/ NX,NXT,NP,NPT,NTR,IT,ISF
      COMMON /BLC3/ X(200),UE(200),P1(200),P2(200),GMTR(200)
      COMMON /BLC7/ ETA(201),DETA(201),A(201)
      COMMON /BLC8/ F(201,2),U(201,2),V(201,2),B(201,2)
      COMMON /BLC6/ DELF(201),DELU(201),DELV(201)
C
C
      NX     = 0
      ITMAX  = 10
      IGROWT = 2
      EPSL   = 0.0001
      EPST   = 0.01
      NPT    = 101
C
C     ETA-GRID
      ETAE     = 8.0
      VGP      = 1.10
      DETA(1)  = 0.01
      NP = ALOG((ETAE/DETA(1))*(VGP-1.0)+1.0)/ALOG(VGP)+1.001
      ETA(1) = 0.0
      DO 10 J=2,NPT
         ETA(J)  = ETA(J-1) + DETA(J-1)
         DETA(J)= VGP*DETA(J-1)
         A(J)    = 0.5*DETA(J-1)
   10 CONTINUE
C
C     INITIAL LAMINAR VELOCITY PROFILE
      DO 20  J=1,NP
         ETAB = ETA(J)/ETA(NP)
         ETAB2 = ETAB**2
         F(J,2) = 0.25*ETA(NP)*ETAB2*(3.0 - 0.5*ETAB2)
         U(J,2) = 0.5*ETAB*(3.0 - ETAB2)
         V(J,2) = 1.5*(1.0 - ETAB2)/ETA(NP)
         B(J,2) = 1.0
   20 CONTINUE
C
C
    1 NX      = NX+1
      IT      =0
      IGROW   = 0
C
    5 IT      = IT+1
      IF (IT .GT. ITMAX) GO TO 101
      IF (NX .GE. NTR) CALL EDDY
      CALL COEF
      CALL SOLV3
C
C     CHECK FOR CONVERGENCE
      IF (NX .LT. NTR) THEN
         IF(ABS(DELV(1)) .GT. EPSL) GO TO 5
      ELSE
```

```
            IF(ABS(DELV(1)/V(1,2)) .GT. EPST) GO TO 5
         ENDIF
C
C        PROFILES FOR GROWTH
         DO 30 J=NP+1,NPT
            F(J,2) = F(J-1,2) + DETA(J-1)*U(J-1,2)
            U(J,2) = U(J-1,2)
            V(J,2) = 0.0
            B(J,2) = B(J-1,2)
   30    CONTINUE
C
C        CHECK FOR GROWTH
         IF (ABS(V(NP,2)) .GT. 0.0005 .OR. ABS(1.0-U(NP-2,2)/U(NP,2))
     +                .GT. 0.005) THEN
            NP = NP+2
            IGROW  = IGROW+1
            IF (NP .LE. NPT .AND. IGROW .LE. IGROWT) THEN
                IT     = 0
                GO TO 5
            ENDIF
         ENDIF
C
  101    CALL OUTPUT
         IF (NX .LT. NXT) GO TO 1
C
         RETURN
         END
         SUBROUTINE COEF
         COMMON /BLC2/ NX,NXT,NP,NPT,NTR,IT,ISF
         COMMON /BLC3/ X(200),UE(200),P1(200),P2(200),GMTR(200)
         COMMON /BLC7/ ETA(201),DETA(201),A(201)
         COMMON /BLC8/ F(201,2),U(201,2),V(201,2),B(201,2)
         COMMON /BLC9/ S1(201),S2(201),S3(201),S4(201),S5(201),S6(201),
     +                 S7(201),S8(201),R1(201),R2(201),R3(201),R4(201)
C
C
         P1H = 0.5 * P1(NX)
         IF (NX .EQ. 1) THEN
            CEL = 0.0
            CELH= 0.0
            DO 5 J=1,NP
               F(J,1) = 0.0
               U(J,1) = 0.0
               V(J,1) = 0.0
               B(J,1) = 0.0
    5       CONTINUE
         ELSE
            CEL = 0.5 * (X(NX)+X(NX-1))/(X(NX)-X(NX-1))
            CELH= 0.5 * CEL
         ENDIF
C
         DO 100 J= 2,NP
C           CURRENT STATION
            FB      = 0.5*(F(J,2) + F(J-1,2))
            UB      = 0.5*(U(J,2) + U(J-1,2))
            FVB     = 0.5*(F(J,2)*V(J,2)+F(J-1,2)*V(J-1,2))
            VB      = 0.5*(V(J,2) + V(J-1,2))
            USB     = 0.5*(U(J,2)**2 + U(J-1,2)**2)
            DERBV   =(B(J,2)*V(J,2) - B(J-1,2)*V(J-1,2))/DETA(J-1)
C
C           PREVIOUS STATION
            CFB     = 0.5*(F(J,1) + F(J-1,1))
            CUB     = 0.5*(U(J,1) + U(J-1,1))
            CVB     = 0.5*(V(J,1) + V(J-1,1))
            CUSB    = 0.5*(U(J,1)**2 + U(J-1,1)**2)
            CFVB    = 0.5*(F(J,1)*V(J,1)+F(J-1,1)*V(J-1,1))
            CDERBV = (B(J,1)*V(J,1) - B(J-1,1)*V(J-1,1))/DETA(J-1)
C
C           S- COEFFICIENTS
            S1(J)   = CELH*(F(J,2) - CFB) + P1H*F(J,2) + B(J,2)/DETA(J-1)
            S2(J)   = CELH*(F(J-1,2)-CFB) + P1H*F(J-1,2)-B(J-1,2)/DETA(J-1)
```

```
            IF(ABS(DELV(1)/V(1,2)) .GT. EPST) GO TO 5
         ENDIF
C
C        PROFILES FOR GROWTH
         DO 30 J=NP+1,NPT
            F(J,2) = F(J-1,2) + DETA(J-1)*U(J-1,2)
            U(J,2) = U(J-1,2)
            V(J,2) = 0.0
            B(J,2) = B(J-1,2)
   30    CONTINUE
C
C        CHECK FOR GROWTH
         IF (ABS(V(NP,2)) .GT. 0.0005 .OR. ABS(1.0-U(NP-2,2)/U(NP,2))
     *                .GT. 0.005) THEN
            NP    NP+2
            IGROW  = IGROW+1
            IF (NP .LE. NPT .AND. IGROW .LE. IGROWT) THEN
               IT      = 0
               GO TO 5
            ENDIF
         ENDIF
C
  101    CALL OUTPUT
         IF (NX .LT. NXT) GO TO 1
C
         RETURN
         END
         SUBROUTINE COEF
         COMMON /BLC2/ NX,NXT,NP,NPT,NTR,IT,ISF
         COMMON /BLC3/ X(200),UE(200),P1(200),P2(200),GMTR(200)
         COMMON /BLC7/ ETA(201),DETA(201),A(201)
         COMMON /BLC8/ F(201,2),U(201,2),V(201,2),B(201,2)
         COMMON /BLC9/ S1(201),S2(201),S3(201),S4(201),S5(201),S6(201),
     *                 S7(201),S8(201),R1(201),R2(201),R3(201),R4(201)
C
C
         P1H = 0.5 * P1(NX)
         IF (NX .EQ. 1) THEN
            CEL = 0.0
            CELH= 0.0
            DO 5 J=1,NP
               F(J,1) = 0.0
               U(J,1) = 0.0
               V(J,1) = 0.0
               B(J,1) = 0.0
    5       CONTINUE
         ELSE
            CEL = 0.5 * (X(NX)+X(NX-1))/(X(NX)-X(NX-1))
            CELH= 0.5 * CEL
         ENDIF
C
         DO 100 J= 2,NP
C           CURRENT STATION
            FB     = 0.5*(F(J,2) + F(J-1,2))
            UB     = 0.5*(U(J,2) + U(J-1,2))
            FVB    = 0.5*(F(J,2)*V(J,2)+F(J-1,2)*V(J-1,2))
            VB     = 0.5*(V(J,2) + V(J-1,2))
            USB    = 0.5*(U(J,2)**2 + U(J-1,2)**2)
            DERBV  =(B(J,2)*V(J,2) - B(J-1,2)*V(J-1,2))/DETA(J-1)
C
C           PREVIOUS STATION
            CFB    = 0.5*(F(J,1) + F(J-1,1))
            CUB    = 0.5*(U(J,1) + U(J-1,1))
            CVB    = 0.5*(V(J,1) + V(J-1,1))
            CUSB   = 0.5*(U(J,1)**2 + U(J-1,1)**2)
            CFVB   = 0.5*(F(J,1)*V(J,1)+F(J-1,1)*V(J-1,1))
            CDERBV = (B(J,1)*V(J,1) - B(J-1,1)*V(J-1,1))/DETA(J-1)
C
C           S- COEFFICIENTS
            S1(J)  = CELH*(F(J,2) - CFB) + P1H*F(J,2) + B(J,2)/DETA(J-1)
            S2(J)  = CELH*(F(J-1,2)-CFB) + P1H*F(J-1,2)-B(J-1,2)/DETA(J-1)
```

```
              S3(J)   = CELH*(V(J,2) + CVB) + P1H*V(J,2)
              S4(J)   = CELH*(V(J-1,2) + CVB) + P1H*V(J-1,2)
              S5(J)   = -(CEL+P2(NX))*U(J,2)
              S6(J)   = -(CEL+P2(NX))*U(J-1,2)
C
C             R- COEFFICIENTS
              IF (NX .EQ. 1) THEN
                 CRB     = -P2(NX)
                 R2(J)   = CRB - (DERBV + P1(NX)*FVB - P2(NX)*USB)
              ELSE
                 CLB     = CDERBV  + P1(NX-1)*CFVB - P2(NX-1)*CUSB + P2(NX-1)
                 CRB     = -CLB - CEL*CUSB - P2(NX)
                 R2(J)   = CRB - (DERBV + P1(NX)*FVB- (CEL+P2(NX))*USB  + CEL*
     *                    (FVB + CVB*FB - VB*CFB - CFVB))
              ENDIF
              R1(J)   = F(J-1,2) - F(J,2) + DETA(J-1)*UB
              R3(J-1)= U(J-1,2) - U(J,2) + DETA(J-1)*VB
  100     CONTINUE
C
C         BOUNDARY CONDITIONS
C
          R1(1)   = 0.0
          R2(1)   = 0.0
          R3(NP)  = 0.0
C
          RETURN
          END
          COMMON /BLC0/ RL,NBL(2),XCTRI(2),ntflag,transnew(2),"
          COMMON /BLC1/ ITR,XCTR,XC(200),YC(200)
          COMMON /BLC2/ NX,NXT,NP,NPT,NTR,IT,ISF
          COMMON /BLC3/ X(200),UE(200),P1(200),P2(200),GMTR(200)
          COMMON /BLC5/ DLS(200),VW(200),CF(200),THT(200)
          DIMENSION     NXTSF(2),XI(200),YI(200),VEI(200)
C
C ???
          OPEN (UNIT=9,FILE='bl2d.dat',STATUS='UNKNOWN')
          OPEN (UNIT=8,FILE='bl2d.out',STATUS='UNKNOWN')
          OPEN (UNIT=20,FILE='cf.dat',STATUS='UNKNOWN')
          OPEN (UNIT=21,FILE='dls.dat',STATUS='UNKNOWN')

          WRITE(6,*) 'READING THE DATA...'
          READ ( 9,15 ) RL,XCTRI(1),XCTRI(2)
          READ ( 9,10 ) NI,IS,ITRANS
          READ ( 9,15 ) (XI(I),YI(I),VEI(I),I=1,NI)
          WRITE(6,*) 'INPUT OF DATA COMPLETE.'
C
C
          WRITE(8,90) RL,XCTRI(1),XCTRI(2)
          NXTSF(1)= NI - IS + 1
          NXTSF(2)= IS
C
C         DATA FOR EACH SURFACE
          DO 200 ISF = 1,2
          ntflag=0
          NXT = NXTSF(ISF)
          GO TO (201,202),ISF
C
C         UPPER SURFACE
      201 II = IS-1
          DO 211 I=1,NXT
             II = II+1
             XC(I) = XI(II)
             YC(I) = YI(II)
             UE(I) = VEI(II)
      211 CONTINUE
          GO TO 300
C
C         LOWER SURFACE
      202 II = IS+1
          DO 212 I=1,NXT
             II = II-1
```

```
            XC(I) = XI(II)
            YC(I) = YI(II)
            UE(I) = VEI(II)
  212 CONTINUE
C
  300 X(1) = 0.0
      DO 301 I=2,NXT
  301 X(I) = X(I-1)+SQRT((XC(I)-XC(I-1))**2+(YC(I)-YC(I-1))**2)
C
C
C     TRANSITION LOCATION
      DO 320 I=1,NXT
         GMTR(I) = 0.0
         IF (XC(I) .GE. XCTRI(ISF)) GO TO 321
  320 CONTINUE
  321 NTR = I
      PGAMTR  = 1200.
      RXNTR   = X(NTR-1)* UE(NTR-1) * RL
      GGFT    = RL**2/RXNTR**1.34*UE(NTR-1)**3
      UEINTG  = 0.0
      U1      = 0.5/UE(NTR-1)/ PGAMTR
      DO 322 I = NTR,NXT
         U2       = 0.5/UE(I)/PGAMTR
         UEINTG   = UEINTG+(U1+U2)*(X(I)-X(I-1))
         U1       = U2
         GG       = GGFT*UEINTG*(X(I)-X(NTR-1))
         IF(GG .GT. 10.0) GO TO 323
         GMTR(I)= 1.0-EXP(-GG)
  322 CONTINUE
  323 DO 324 II=I,NXT
  324 GMTR(II) = 1.0
C
C     PRESSURE GRADIENT PARAMETERS
      DX      = X(2)-X(1)
      DUE     = UE(2)-UE(1)
      ANG2    = ATAN2(DUE,DX)
      DL2     = DX
      DO 331 I = 2,NXT-1
         ANG1     = ANG2
         DL1      = DL2
         DX       = X(I+1)-X(I)
         DUE      = UE(I+1)-UE(I)
         ANG2     = ATAN2(DUE,DX)
         DL2      = DX
         ANG      = (DL2*ANG1+DL1*ANG2)/(DL1+DL2)
         P2(I)    = TAN(ANG)
  331 CONTINUE
      P2(NXT) = 2.*DUE/DL2 - P2(NXT-1)
      DO 330 I = 2,NXT
      P2(I)   = X(I) * P2(I) /UE(I)
      P1(I)   = 0.5 * (1.0 + P2(I))
  330 CONTINUE
      P2(1)   = 1.0
      P1(1)   = 0.5 * (1.0 + P2(1))
C
C
C     BOUNDARY LAYER CALCULATION
      WRITE(6,*) 'BOUNDARY LAYER COMPUTATIONS IN PROGRESS..'
      CALL BL
      WRITE(8,910)ISF,(I,XC(I),X(I),VW(I),CF(I),DLS(I),THT(I),I=1,NXT)
      if(ISF.EQ.1) then
         write(20,905) (XC(I),CF(I),I=2,NXT)
         write(21,905) (XC(I),DLS(I),I=2,NXT)
      end if
  905 FORMAT(F8.4,4X,E11.4)
  200 CONTINUE
C
***IF AOA is 0 deg., make trans. locs. equal:
      if(vei(2).eq.vei(ni-1)) transnew(1)=transnew(2)

      if(ITRANS.eq.0) then
```

```
            print *,'Estimate for upper transition:',transnew(1)
            print *,'Estimate for lower transition:',transnew(2)
         endif

         CLOSE(UNIT=8)
         CLOSE(UNIT=9)
         STOP
C
   10    FORMAT(3I5)
   15    FORMAT(3F10.0)
   90    FORMAT(//5X,'RL=',E12.5,5X,'XCTRI(1) =',F8.3,5X,'XCTR(2) =',F8.3)
  910    FORMAT(///2X,'*** SUMMARY OF BOUNDARY LAYER SOLUTIONS OF ISF =',I2
        +//2X,'NX',4X,'XC',8X,'S',8X,'VW',8X,'CF',8X,'DLS',8X,'THT'
        +/(I5,2F8.4,4F11.4))
         END
         SUBROUTINE EDDY
         COMMON /BLC0/ RL,NBL(2),XCTRI(2),ntflag,transnew(2),NI
         COMMON /BLC2/ NX,NXT,NP,NPT,NTR,IT,ISF
         COMMON /BLC3/ X(200),UE(200),P1(200),P2(200),GMTR(200)
         COMMON /BLC7/ ETA(201),DETA(201),A(201)
         COMMON /BLC8/ F(201,2),U(201,2),V(201,2),B(201,2)
         DIMENSION EDVI(201)
C
C
         RL2     = SQRT(RL*UE(NX)*X(NX))
         RL4     = SQRT(RL2)
         RL216   = 0.16 * RL2
C
         ALFA    = 0.0168
         EDVO    = ALFA*RL2*GMTR(NX)*(U(NP,2)*ETA(NP)-F(NP,2))
         EDVI(1)= 0.0
         YBAJ    = RL4*SQRT(ABS(V(1,2)))/26.0
         DO 70 J=2,NP
            JJ      = J
            YBA     = YBAJ*ETA(J)
            EL      = 1.0
            IF(YBA .LT. 10.0) EL = 1.0 - EXP(-YBA)
            EDVI(J)  = RL216*GMTR(NX)*(EL*ETA(J))**2 * ABS(V(J,2))
            IF(EDVI(J) .GT. EDVO) GO TO 90
            IF (EDVI(J) .LE. EDVI(J-1)) EDVI(J)= EDVI(J-1)
            B(J,2) = 1.0 + EDVI(J)
   70    CONTINUE
   90    DO 100 JJ=J,NPT
  100    B(JJ,2) = 1.0 + EDVO
         B(1,2) = 1.0
C
         RETURN
         END
         SUBROUTINE OUTPUT
         COMMON /BLC0/ RL,NBL(2),XCTRI(2),ntflag,transnew(2),NI
         COMMON /BLC2/ NX,NXT,NP,NPT,NTR,IT,ISF
         COMMON /BLC3/ X(200),UE(200),P1(200),P2(200),GMTR(200)
         COMMON /BLC7/ ETA(201),DETA(201),A(201)
         COMMON /BLC8/ F(201,2),U(201,2),V(201,2),B(201,2)
         COMMON /BLC9/ DLS(200),VW(200),CF(200),THT(200)
         dimension rdiff(201),rdlow(201)
C
C
         IF(NX.EQ.1 ) THEN
            DLS(NX)= 0.0
            THT(NX)= 0.0
            CF(NX) = 0.0
            VW(NX) = V(1,2)
            rdifflow=1000
            nstop=0
         ELSE
C
            SQRX    = SQRT(UE(NX)*X(NX)*RL)
            CF(NX) = 2.0 * V(1,2) * B(1,2) /SQRX
            VW(NX) = V(1,2)
            DLS(NX)= X(NX)/SQRX * (ETA(NP)-F(NP,2))
```

```
            U1      = U(1,2) * (1.0 -U(1,2))
            SUM     = 0.0
            DO 20 J=2,NP
               U2      = U(J,2) * (1.0 -U(J,2))
               SUM     = SUM + A(J) * (U1 + U2)
               U1      = U2
   20       CONTINUE
            THT(NX)= X(NX)/SQRX * SUM
            rex=UE(NX)*X(NX)*RL
            rtheta=UE(NX)*THT(NX)*RL
            rtrans=1.174*(1.0+22400.0/rex)*rex**0.46
            rdiff(nx)=abs(rtheta-rtrans)
            if ((NX.gt.2) .and. (rdlow(nx-1).eq.rdlow(nx-2))) then
                if (rdlow(nx-2).eq.rdlow(nx-3)) nstop=1
            endif
            if (ISF.eq.2) then
              if ((ntflag.eq.1) .and. (nstop.eq.0)) then
                  transnew(ISF)=rex/(RL*UE(NX))
                  ntflag=0
              endif
            endif
            if((rdiff(nx).LT.rdifflow) .and. (nstop.eq.0)) then
                 transnew(ISF)=rex/(RL*UE(NX))
                 rdifflow=rdiff(nx)
                 ntflag=1
            endif
            rdlow(nx)=rdifflow
         ENDIF
c         print *,nstop,nx,rdiff( nx-1),rdiff(nx-2)
C         print *, isf,rex,transnew(isf)
c         print *,rtheta,rtrans,rdiff(nx),rdifflow
C
C
C        SHIFT PROFILES FOR THE NEXT STATION
C
         ymark=.0005
         DO 175 J=1,NPT
            if(ISF.EQ.1) then
               if(U(J,1).LT.'0.995)) then
                  lasty=1
                  yplot=ETA(J)*SQRT(X(NX)/(RL*UE(NX)))
                  do nxloop=5,NI/2-1,5
                     if(NX.EQ.nxloop) then
   91                   markx=NX/5
                        numw=markx+30
                        write (numw,*) U(J,1)+markx,yplot
                        write (60,*) U(J,1)+markx,yplot
                        if(yplot.gt.ymark) then
                          write(55,*) markx,ymark
                          ydiff=yplot-yplotold
                          udiff=U(J,1)-U(J-1,1)
                          xvalue=U(J-1,1)+udiff*(ymark-yplotold)/ydiff
                          write(55,*) xvalue+markx,ymark
                          write(55,92)
   92                     format (/)
                          ymark=ymark+.0005
                          if(yplot.GT.ymark) goto 91
                        endif
                     endif
                  end do
               else
                  if (lasty.EQ.1) then
                     lasty=0
                     do m=1,2
                        do nxloop=5,NI/2-1,5
                           if(NX.EQ.nxloop) then
                              markx=NX/5
                              numw=markx+30
                              write (numw,*) markx,yplot
                              write (60,*) markx,yplot
                           endif
```

```
                     end do
                     yplot=0.0
                  end do
               endif
            endif
         endif
         yplotold=yplot
         F(J,1)   = F(J,2)
         U(J,1)   = U(J,2)
         V(J,1)   = V(J,2)
         B(J,1)   = B(J,2)
 175  CONTINUE
C
      RETURN
      END
      SUBROUTINE SOLV3
      COMMON /BLC2/ NX,NXT,NP,NPT,NTR,IT,ISF
      COMMON /BLC7/ ETA(201),DETA(201),A(201)
      COMMON /BLC8/ F(201,2),U(201,2),V(201,2),B(201,2)
      COMMON /BLC9/ S1(201),S2(201),S3(201),S4(201),S5(201),S6(201),
     +              S7(201),S8(201),R1(201),R2(201),R3(201),R4(201)
      COMMON /BLC6/ DELF(201),DELU(201),DELV(201)
      DIMENSION     A11(201),A12(201),A13(201),A14(201),
     +              A21(201),A22(201),A23(201),A24(201)
C
C
      A11(1)= 1.0
      A12(1)= 0.0
      A13(1)= 0.0
      A21(1)= 0.0
      A22(1)= 1.0
      A23(1)= 0.0
      G11    =-1.0
      G12    =-A(2,
      G13    = 0.0
      G21    = S4(2)
      G23    =-S2(2)/A(2)
      G22    = G23+S6(2)
      A11(2)= 1.0
      A12(2)=-A(2)-G13
      A13(2)= A(2)*G13
      A21(2)= S3(2)
      A22(2)= S5(2)-G23
      A23(2)= S1(2)+A(2)*G23
      R1(2)  = R1(2)-(G11*R1(1)+G12*R2(1)+G13*R3(1))
      R2(2)  = R2(2)-(G21*R1(1)+G22*R2(1)+G23*R3(1))
C
C  FORWARD SWEEP
-
      DO 500 J=2,NP
         DEN    = (A13(J-1)*A21(J-1)-A23(J-1)*A11(J-1)-A(J)*
     +                   (A12(J-1)*A21(J-1)-A22(J-1)*A11(J-1)))
         DEN1   = A22(J-1)*A(J)-A23(J-1)
         G11    = (A23(J-1)+A(J)*(A(J)*A21(J-1)-A22(J-1)))/DEN
         G12    = -(A(J)*A(J)+G11*(A12(J-1)*A(J)-A13(J-1)))/DEN1
         G13    = (G11*A13(J-1)+G12*A23(J-1))/A(J)
         G21    = (S2(J)*A21(J-1)-S4(J)*A23(J-1)+A(J)*(S4(J)*
     +                   A22(J-1)-S6(J)*A21(J-1)))/DEN
         G22    = (-S2(J)+S6(J)*A(J)-G21*(A(J)*A12(J-1)-A13(J-1)))/DEN1
         G23    = G21*A12(J-1)+G22*A22(J-1)-S6(J)
         A11(J)= 1.0
         A12(J)=-A(J)-G13
         A13(J)= A(J)*G13
         A21(J)= S3(J)
         A22(J)= S5(J)-G23
         A23(J)= S1(J)+A(J)*G23
         R1(J)  = R1(J)-(G11*R1(J-1)+G12*R2(J-1)+G13*R3(J-1))
         R2(J)  = R2(J)-(G21*R1(J-1)+G22*R2(J-1)+G23*R3(J-1))
 500  CONTINUE
C
C  BACKWARD SWEEP
```

```
C
       DELU(NP)  = R3(NP)
       E1        = R1(NP)-A12(NP)*DELU(NP)
       E2        = R2(NP)-A22(NP)*DELU(NP)
       DELV(NP)  = (E2*A11(NP)-E1*A21(NP))/(A23(NP)*A11(NP)-A13(NP)*
     +                  A21(NP))
       DELF(NP)  = (E1-A13(NP)*DELV(NP))/A11(NP)
       DO 600 J = NP-1,1,-1
          E3        = R3(J)-DELU(J+1)+A(J+1)*DELV(J+1)
          DEN2      = A21(J)*A12(J)*A(J+1)-A21(J)*A13(J)-A(J+1)*A22(J)*
     +                A11(J)+A23(J)*A11(J)
          DELV(J)   = (A11(J)*(R2(J)+E3*A22(J))-A21(J)*R1(J)-E3*A21(J)*
     +                A12(J))/DEN2
          DELU(J)   =-A(J+1)*DELV(J)-E3
          DELF(J)   = (R1(J)-A12(J)*DELU(J)-A13(J)*DELV(J))/A11(J)
  600  CONTINUE
C
       DO 700 J=1,NP
          F(J,2)= F(J,2)+DELF(J)
          U(J,2)= U(J,2)+DELU(J)
          V(J,2)= V(J,2)+DELV(J)
  700  CONTINUE
       U(1,2)= 0.0
C
       RETURN
       END
```

Output file: bl2d.out

RL= 0.54000E+06 XCTR(1) = 0.300 XCTR(2) = 0.[illegible]

*** SUMMARY OF BOUNDARY LAYER SOLUTIONS OF ISF = 1

NX	XC	S	VW	CF	DLS	THT
1	0.0024	0.0000	0.1232E+01	0.0000E+00	0.0000E+00	0.0000E+00
2	0.0005	0.0057	0.8456E 00	0.6090E-01	0.1815E-03	0.3901E-04
3	0.0005	0.0112	0.1242E+01	0.3073E-01	0.1215E-03	0.5674E-04
4	0.0024	0.0170	0.1083E+01	0.2220E-01	0.1219E-03	0.5399E-04
5	0.0064	0.0236	0.8010E+00	0.1292E-01	0.1656E-03	0.6995E-04
6	0.0123	0.0314	0.6520E+00	0.8809E-02	0.2164E-03	0.8960E-04
7	0.0201	0.0406	0.5141E+00	0.6033E-02	0.2828E-03	0.1131E-03
8	0.0298	0.0513	0.4639E+00	0.4821E-02	0.3479E-03	0.1377E-03
9	0.0413	0.0637	0.4023E+00	0.3749E-02	0.4201E-03	0.1631E-03
10	0.0547	0.0777	0.3660E+00	0.3096E-02	0.4966E-03	0.1901E-03
11	0.0698	0.0934	0.3224E+00	0.7498E-02	0.5820E-03	0.2185E-03
12	0.0866	0.1106	0.3097E+00	0.2212E-02	0.6613E-03	0.2467E-03
13	0.1050	0.1293	0.2804E+00	0.1860E-02	0.7490E-03	0.2755E-03
14	[illegible]	0.1496	0.2644E+00	0.1637E-02	0.8387E-03	0.3051E-03
15	0.1466	0.1712	0.2426E+00	0.1411E-02	0.9339E-03	0.3353E-03
16	0.1695	0.1941	0.2393E+00	0.1311E-02	0.1020E-02	0.3646E-03
17	0.1937	0.2184	0.2091E+00	0.1085E-02	[illegible]	0.3954E-03
18	0.2191	0.2438	0.1926E+00	0.9513E-03	0.1241E-02	0.4278E-03
19	0.2456	0.2704	0.1784E+00	0.8401E-03	0.1349E-02	0.4593E-03
20	0.2731	0.2979	0.1608E+00	0.7248E-03	0.1469E-02	0.4917E-03
21	0.3015	0.3263	0.1398E+00	0.6046E-03	0.1602E-02	0.5247E-03
22	0.3307	0.3555	0.1202E+00	0.5001E-03	0.1746E-02	0.5590E-03
23	0.3605	0.3853	0.7410E-01	0.2978E-03	0.1965E-02	0.5940E-03
24	0.3909	0.4158	0.4473E-01	0.1739E-03	0.2177E-02	0.6315E-03
25	0.4219	0.4467	0.1177E+00	0.4439E-03	0.1994E-02	0.6692E-03
26	0.4530	0.4778	0.2823E+00	0.1033E-02	0.1746E-02	0.7035E-03
27	0.4843	0.5092	0.4308E+00	0.1534E-02	0.1655E-02	0.7520E-03
28	0.5156	0.5407	0.5688E+00	0.1976E-02	0.1642E-02	0.8127E-03
29	0.5469	0.5720	0.7246E+00	0.2460E-02	0.1656E-02	0.8794E-03
30	0.5781	0.6033	0.8574E+00	0.2845E-02	0.1700E-02	0.9503E-03
31	0.6090	0.6342	0.9414E+00	0.3062E-02	0.1782E-02	0.1031E-02
32	0.6394	0.6648	0.1017E+01	0.3247E-02	0.1881E-02	0.1118E-02
33	0.6693	0.6948	0.1077E+01	0.3379E-02	0.1987E-02	0.1206E-02
34	0.6984	0.7241	0.1119E+01	0.3454E-02	0.2100E-02	0.1296E-02
35	0.7269	0.7526	0.1133E+01	0.3443E-02	0.2229E-02	0.1390E-02
36	0.7544	0.7803	0.1146E+01	0.3438E-02	0.2364E-02	0.1486E-02
37	0.7808	0.8070	0.1152E+01	0.3411E-02	0.2499E-02	0.1581E-02
38	0.8062	0.8325	0.1131E+01	0.3317E-02	0.2655E-02	0.1684E-02
39	0.8304	0.8569	0.1107E+01	0.3216E-02	0.2813E-02	0.1787E-02
40	0.8533	0.8799	0.1089E+01	0.3139E-02	0.2975E-02	0.1891E-02
41	0.8748	0.9016	0.1045E+01	0.2989E-02	0.3148E-02	0.1999E-02
42	0.8949	0.9218	0.9844E+00	0.2804E-02	0.3351E-02	0.2119E-02
43	0.9133	0.9404	0.9361E+00	0.2655E-02	0.3557E-02	0.2239E-02
44	0.9301	0.9573	0.8514E+00	0.2408E-02	0.3787E-02	0.2366E-02
45	0.9452	0.9726	0.7083E+00	0.2002E-02	0.4079E-02	0.2513E-02
46	0.9586	0.9860	0.4504E+00	0.1275E-02	0.4490E-02	0.2688E-02
47	0.9701	0.9977	0.2796E+00	0.7951E-03	0.4909E-02	0.2877E-02
48	0.9798	1.0075	-0.1193E+01	-0.3410E-02	0.7082E-02	0.3153E-02
49	0.9876	1.0154	-0.9638E+03	-0.2784E+01	0.1712E+03	-0.4403E+07
50	0.9936	1.0213	0.2408E+12	0.7077E+09	-0.5044E-08	-0.3973E+19
51	0.9975	1.0253	0.3259E+12	0.9806E+09	-0.2188E-09	-0.4494E+20
52	0.9995	1.0273	0.9400E+12	0.2907E+10	-0.3171E+09	-0.5061E+20

*** SUMMARY OF BOUNDARY LAYER SOLUTIONS OF ISF = 2

NX	XC	S	VW	CF	DLS	THT
1	0.0024	0.0000	0.1232E+01	0.0000E+00	0.0000E+00	0.0000E+00
2	0.0064	0.0066	0.8911E+00	0.4236E-01	0.1316E-03	0.5759E-04
3	0.0123	0.0144	0.7976E+00	0.2157E-01	0.1814E-03	0.7867E-04
4	0.0201	0.0236	0.7397E+00	0.1440E-01	0.2262E-03	0.9751E-04

5	0.0298	0.0344	0.6680E+00	0.1026E-01	0.2790E-03	0.1184E-03
6	0.0413	0.0468	0.6251E+00	0.7990E-02	0.3328E-03	0.1401E-03
7	0.0547	0.0607	0.5723E+00	0.6280E-02	0.3921E-03	0.1629E-03
8	0.0698	0.0764	0.5379E+00	0.5202E-02	0.4550E-03	0.1874E-03
9	0.0866	0.0936	0.5121E+00	0.4426E-02	0.5160E-03	0.2112E-03
10	0.1050	0.1123	0.4870E+00	0.3818E-02	0.5802E-03	0.2358E-03
11	0.1251	0.1326	0.4627E+00	0.3321E-02	0.6472E-03	0.2609E-03
12	0.1466	0.1542	0.4419E+00	0.2932E-02	0.7162E-03	0.2867E-03
13	0.1695	0.1772	0.4283E+00	0.2644E-02	0.7834E-03	0.3121E-03
14	0.1937	0.2014	0.4090E+00	0.2364E-02	0.8539E-03	0.3378E-03
15	0.2191	0.2269	0.3842E+00	0.2092E-02	0.9338E-03	0.3653E-03
16	0.2456	0.2534	0.3742E+00	0.1928E-02	0.1008E-02	0.3923E-03
17	0.2731	0.2809	0.3582E+00	0.1753E-02	0.1084E-02	0.4191E-03
18	0.3015	0.3093	0.3411E+00	0.1593E-02	0.1166E-02	0.4469E-03
19	0.3307	0.3385	0.3272E+00	0.1462E-02	0.1247E-02	0.4743E-03
20	0.3605	0.3683	0.3032E+00	0.1301E-02	0.1341E-02	0.5032E-03
21	0.3909	0.3988	0.2810E+00	0.1162E-02	0.1442E-02	0.5333E-03
22	0.4218	0.4297	0.2691E+00	0.1075E-02	0.1535E-02	0.5629E-03
23	0.4530	0.4609	0.2643E+00	0.1021E-02	0.1618E-02	0.5909E-03
24	0.4843	0.4923	0.2476E+00	0.9272E-03	0.1711E-02	0.6187E-03
25	0.5156	0.5237	0.2102E+00	0.7655E-03	0.1844E-02	0.6497E-03
26	0.5469	0.5551	0.1983E+00	0.7038E-03	0.1962E-02	0.6808E-03
27	0.5781	0.5863	0.1997E+00	0.6912E-03	0.2043E-02	0.7082E-03
28	0.6090	0.6173	0.1641E+00	0.5549E-03	0.2182E-02	0.7375E-03
29	0.6394	0.6478	0.1279E+00	0.4237E-03	0.2366E-02	0.7697E-03
30	0.6693	0.6778	0.1173E+00	0.3810E-03	0.2504E-02	0.7996E-03
31	0.6984	0.7071	0.9714E-01	0.3098E-03	0.2658E-02	0.8285E-03
32	0.7268	0.7356	0.3892E-01	0.1220E-03	0.2973E-02	0.8592E-03
33	0.7544	0.7633	0.7604E-01	0.2349E-03	0.2607E-02	0.7663E-03
34	0.7808	0.7900	0.5406E-02	0.1646E-04	0.3036E-02	0.7969E-03
35	0.8062	0.8155	-0.8287E-01	-0.2493E-03	0.3390E-02	0.7639E-03
36	0.8304	0.8399	-0.9041E-01	-0.2691E-03	0.3384E-02	0.7997E-03
37	0.8533	0.8629	-0.1562E+00	-0.4602E-03	0.3787E-02	0.8365E-03
38	0.8748	0.8846	-0.2511E+00	-0.7339E-03	0.4468E-02	0.8788E-03
39	0.8949	0.9048	-0.3360E+00	-0.9753E-03	0.5324E-02	0.9352E-03
40	0.9133	0.9234	-0.7438E+01	-0.2150E-01	0.2542E-01	-0.5421E+00
41	0.9301	0.9403	-0.2989E+07	-0.8597E+04	0.7034E+03	-0.2021E+09
42	0.9452	0.9556	-0.3229E+07	-0.9271E+04	-0.1513E+07	-0.1239E+16
43	0.9586	0.9691	-0.1935E+12	-0.5553E+09	0.1354E+09	-0.2746E+22
44	0.9701	0.9807	-0.8049E+14	-0.2316E+12	0.2118E+10	-0.5502E+23
45	0.9798	0.9905	0.1412E+17	0.4082E+14	0.3070E+14	-0.7058E+30
46	0.9876	0.9984	NaN	NaN	NaN	-NaN
47	0.9936	1.0044	NaN	NaN	NaN	-NaN
48	0.9975	1.0083	NaN	NaN	NaN	-NaN
49	0.9995	1.0104	NaN	NaN	NaN	-NaN

Gnuplot command file: **profile**

```
set terminal tek40xx
set nogrid
set nolabel
set size 1,1
set data style lines
set noxtics
set ytics
set title "Velocity Profiles  AOA=10 deg." 0,0
set nokey
set xlabel "Airfoil Upper Surface Station" 0,0
set xrange [0 : 10]
set ylabel "y/c" 0,.5
set yrange [0 : .010]
set size .6,.6
set label "Station 0 is" at .8,.008
set label "stagnation" at .8,.007
set label "point" at .8, .006
set label "5" at 1,-.0003 center
set label "10" at 2,-.0003 center
set label "15" at 3,-.0003 center
set label "20" at 4,-.0003 center
set label "25" at 5,-.0003 center
set label "30" at 6,-.0003 center
set label "35" at 7,-.0003 center
set label "40" at 8,-.0003 center
set label "45" at 9,-.0003 center
plot "pro10","lin10"
```

Input file: FOR001.DAT (Stardent) or fort.1 (IRIS)

```
        3
C
C    NACA 0012 AIRFOIL
C
      ALPI      PIVOT
  6.000000   0.250000
    NLOWER     NUPPER
        50         50
X/C
   1.00000    0.98000    0.96000    0.94000    0.92000    0.90000
   0.88000    0.86000    0.84000    0.82000    0.80000    0.78000
   0.76000    0.74000    0.72000    0.70000    0.68000    0.66000
   0.64000    0.62000    0.60000    0.58000    0.56000    0.54000
   0.52000    0.50000    0.48000    0.46000    0.44000    0.42000
   0.40000    0.38000    0.36000    0.34000    0.32000    0.30000
   0.28000    0.26000    0.24000    0.22000    0.20000    0.18000
   0.16000    0.14000    0.12000    0.10000    0.08000    0.06000
   0.04000    0.02000    0.00000    0.02000    0.04000    0.06000
   0.08000    0.10000    0.12000    0.14000    0.16000    0.18000
   0.20000    0.22000    0.24000    0.26000    0.28000    0.30000
   0.32000    0.34000    0.36000    0.38000    0.40000    0.42000
   0.44000    0.46000    0.48000    0.50000    0.52000    0.54000
   0.56000    0.58000    0.60000    0.62000    0.64000    0.66000
   0.68000    0.70000    0.72000    0.74000    0.76000    0.78000
   0.80000    0.82000    0.84000    0.86000    0.88000    0.90000
   0.92000    0.94000    0.96000    0.98000    1.00000
Y/C
  -0.00126   -0.00403   -0.00674   -0.00938   -0.01196   -0.01448
  -0.01694   -0.01935   -0.02170   -0.02399   -0.02623   -0.02842
  -0.03056   -0.03264   -0.03467   -0.03664   -0.03856   -0.04042
  -0.04222   -0.04396   -0.04563   -0.04723   -0.04878   -0.05026
  -0.05165   -0.05294   -0.05415   -0.05530   -0.05634   -0.05726
  -0.05803   -0.05868   -0.05923   -0.05966   -0.05995   -0.06006
  -0.05997   -0.05966   -0.05911   -0.05838   -0.05737   -0.05607
  -0.05444   -0.05236   -0.04990   -0.04683   -0.04309   -0.03842
  -0.03231   -0.02382    0.00000    0.02382    0.03231    0.03842
   0.04309    0.04683    0.04990    0.05236    0.05444    0.05607
   0.05737    0.05838    0.05911    0.05966    0.05997    0.06006
   0.05995    0.05966    0.05923    0.05868    0.05803    0.05726
   0.05634    0.05530    0.05415    0.05294    0.05165    0.05026
   0.04878    0.04723    0.04563    0.04396    0.04222    0.04042
   0.03856    0.03664    0.03467    0.03264    0.03056    0.02842
   0.02623    0.02399    0.02170    0.01935    0.01694    0.01448
   0.01196    0.00938    0.00674    0.00403    0.00126
```

Input file: incompbl.dat

```
     IWAKE       NXT        NW    ITREND
         1       161        37        40
    ITR(1)    ITR(2)    ISWPMX        RL   XCTR(1)
         0         0         1  540000.0   0.30000
        IP
         1
```

Output file: incompbl.out

```
C
C     NACA 0012 AIRFOIL
C


INPUT DATA FOR INVISCID-FLOW CALCULATIONS


  ALPI=     6.0000   PIVOT=     0.2500

NLOWER=         50  NUPPER=         50



COORDINATES OF THE BODY
X/C
 1.000000  0.980000  0.960000  0.940000  0.920000  0.900000
 0.880000  0.860000  0.840000  0.820000  0.800000  0.780000
 0.760000  0.740000  0.720000  0.700000  0.680000  0.660000
 0.640000  0.620000  0.600000  0.580000  0.560000  0.540000
 0.520000  0.500000  0.480000  0.460000  0.440000  0.420000
 0.400000  0.380000  0.360000  0.340000  0.320000  0.300000
 0.280000  0.260000  0.240000  0.220000  0.200000  0.180000
 0.160000  0.140000  0.120000  0.100000  0.080000  0.060000
 0.040000  0.020000  0.000000  0.020000  0.040000  0.060000
 0.080000  0.100000  0.120000  0.140000  0.160000  0.180000
 0.200000  0.220000  0.240000  0.260000  0.280000  0.300000
 0.320000  0.340000  0.360000  0.380000  0.400000  0.420000
 0.440000  0.460000  0.480000  0.500000  0.520000  0.540000
 0.560000  0.580000  0.600000  0.620000  0.640000  0.660000
 0.680000  0.700000  0.720000  0.740000  0.760000  0.780000
 0.800000  0.820000  0.840000  0.960000  0.880000  0.900000
 0.920000  0.940000  0.960000  0.980000  1.000000

Y/C

-0.001260 -0.004030 -0.006740 -0.009380 -0.011960 -0.014480
-0.016940 -0.019350 -0.021700 -0.023990 -0.026230 -0.028420
-0.030560 -0.032640 -0.034670 -0.036640 -0.038560 -0.040420
-0.042220 -0.043960 -0.045630 -0.047230 -0.048780 -0.050260
-0.051650 -0.052940 -0.054150 -0.055300 -0.056340 -0.057260
-0.058030 -0.058680 -0.059230 -0.059660 -0.059950 -0.060060
-0.059970 -0.059660 -0.059110 -0.058380 -0.057370 -0.056070
-0.054440 -0.052360 -0.049900 -0.046830 -0.043090 -0.038420
-0.032310 -0.023820  0.000000  0.023820  0.032310  0.038420
 0.043090  0.046830  0.049900  0.052360  0.054440  0.056070
 0.057370  0.058380  0.059110  0.059660  0.059970  0.060060
 0.059950  0.059660  0.059230  0.058680  0.058030  0.057260
 0.056340  0.055300  0.054150  0.052940  0.051650  0.050260
 0.048780  0.047230  0.045630  0.043960  0.042220  0.040420
 0.038560  0.036640  0.034670  0.032640  0.030560  0.028420
 0.026230  0.023990  0.021700  0.019350  0.016940  0.014480
 0.011960  0.009380  0.006740  0.004030  0.001260
1                               INVISCID FLOW RESULTS

PANEL          XP                YP               CP
    1     0.99000E+00     -0.14950E-02     0.25966E+00
    2     0.97000E+00     -0.44813E-02     0.18892E+00
    3     0.95000E+00     -0.74415E-02     0.14703E+00
    4     0.93000E+00     -0.10321E-01     0.11735E+00
    5     0.91000E+00     -0.13061E-01     0.98371E-01
    6     0.89000E+00     -0.15646E-01     0.87319E-01
    7     0.87000E+00     -0.18113E-01     0.79369E-01
    8     0.85000E+00     -0.20502E-01     0.72019E-01
    9     0.83000E+00     -0.22829E-01     0.65720E-01
   10     0.81000E+00     -0.25104E-01     0.58929E-01
   11     0.79000E+00     -0.27325E-01     0.53452E-01
   12     0.77000E+00     -0.29490E-01     0.48109E-01
   13     0.75000E+00     -0.31600E-01     0.43794E-01
```

14	0.73000E+00	-0.33655E-01	0.39437E-01
15	0.71000E+00	-0.35655E-01	0.35961E-01
16	0.69000E+00	-0.37600E-01	0.32250E-01
17	0.67000E+00	-0.39490E-01	0.29030E-01
18	0.65000E+00	-0.41320E-01	0.26174E-01
19	0.63000E+00	-0.43090E-01	0.23331E-01
20	0.61000E+00	-0.44795E-01	0.21248E-01
21	0.59000E+00	-0.46430E-01	0.20161E-01
22	0.57000E+00	-0.48005E-01	0.17719E-01
23	0.55000E+00	-0.49520E-01	0.15010E-01
24	0.53000E+00	-0.50955E-01	0.13363E-01
25	0.51000E+00	-0.52295E-01	0.13779E-01
26	0.49000E+00	-0.53545E-01	0.14543E-01
27	0.47000E+00	-0.54725E-01	0.12579E-01
28	0.45000E+00	-0.55820E-01	0.11691E-01
29	0.43000E+00	-0.56800E-01	0.11746E-01
30	0.41000E+00	-0.57645E-01	0.15506E-01
31	0.39000E+00	-0.58355E-01	0.20332E-01
32	0.37000E+00	-0.58955E-01	0.23743E-01
33	0.35000E+00	-0.59445E-01	0.26773E-01
34	0.33000E+00	-0.59805E-01	0.29743E-01
35	0.31000E+00	-0.60005E-01	0.34896E-01
36	0.29000E+00	-0.60015E-01	0.42571E-01
37	0.27000E+00	-0.59815E-01	0.53307E-01
38	0.25000E+00	-0.59385E-01	0.69110E-01
39	0.23000E+00	-0.58745E-01	0.81783E-01
40	0.21000E+00	-0.57875E-01	0.99470E-01
41	0.19000E+00	-0.56720E-01	0.12038E+00
42	0.17000E+00	-0.55255E-01	0.14436E+00
43	0.15000E+00	-0.53400E-01	0.18096E+00
44	0.13000E+00	-0.51130E-01	0.21706E+00
45	0.11000E+00	-0.48365E-01	0.26992E+00
46	0.90000E-01	-0.44960E-01	0.33458E+00
47	0.70000E-01	-0.40755E-01	0.42103E+00
48	0.50000E-01	-0.35365E-01	0.54734E+00
49	0.30000E-01	-0.28065E-01	0.70537E+00
50	0.10000E-01	-0.11910E-01	0.92260E+00
51	0.10000E-01	0.11910E-01	-0.22412E+01
52	0.30000E-01	0.28065E-01	-0.22809E+01
53	0.50000E-01	0.35365E-01	-0.18259E+01
54	0.70000E-01	0.40755E-01	-0.15937E+01
55	0.90000E-01	0.44960E-01	-0.14321E+01
56	0.11000E+00	0.48765E-01	-0.13171E+01
57	0.13000E+00	0.51130E-01	-0.12149E+01
58	0.15000E+00	0.53400E-01	-0.11409E+01
59	0.17000E+00	0.55255E-01	-0.10709E+01
60	0.19000E+00	0.56720E-01	-0.10079E+01
61	0.21000E+00	0.57875E-01	-0.95329E+00
62	0.23000E+00	0.58745E-01	-0.89952E+00
63	0.25000E+00	0.59385E-01	-0.85976E+00
64	0.27000E+00	0.59815E-01	-0.81[illegible]2E+00
65	0.29000E+00	0.60015E-01	-0.78028E+00
66	0.31000E+00	0.60005E-01	-0.74122E+00
67	0.33000E+00	0.59805E-01	-0.70214E+00
68	0.35000E+00	0.59445E-01	-0.66601E+00
69	0.37000E+00	0.58955E-01	-0.63339E+00
70	0.39000E+00	0.58355E-01	-0.60614E+00
71	0.41000E+00	0.57645E-01	-0.58144E+00
72	0.43000E+00	0.56800E-01	-0.55344E+00
73	0.45000E+00	0.55820E-01	-0.52502E+00
74	0.47000E+00	0.54725E-01	-0.49560E+00
75	0.49000E+00	0.53545E-01	-0.47125E+00
76	0.51000E+00	0.52295E-01	-0.44926E+00
77	0.53000E+00	0.50955E-01	-0.42597E+00
78	0.55000E+00	0.49520E-01	-0.40167E+00
79	0.57000E+00	0.48005E-01	-0.37797E+00
80	0.59000E+00	0.46430E-01	-0.35733E+00
81	0.61000E+00	0.44795E-01	-0.33679E+00
82	0.63000E+00	0.43090E-01	-0.31577E+00
83	0.65000E+00	0.41320E-01	-0.29529E+00
84	0.67000E+00	0.39490E-01	-0.27507E+00

85	0.69000E+00	0.37600E-01	-0.25461E+00
86	0.71000E+00	0.35655E-01	-0.23509E+00
87	0.73000E+00	0.33655E-01	-0.21494E+00
88	0.75000E+00	0.31600E-01	-0.19540E+00
89	0.77000E+00	0.29490E-01	-0.17526E+00
90	0.79000E+00	0.27319E-01	-0.15370E+00
91	0.81000E+00	0.25096E-01	-0.13335E+00
92	0.83000E+00	0.22823E-01	-0.11255E+00
93	0.85000E+00	0.20492E-01	-0.91752E-01
94	0.87000E+00	0.18084E-01	-0.69232E-01
95	0.89000E+00	0.15564E-01	-0.40632E-01
96	0.91000E+00	0.12899E-01	-0.29571E-02
97	0.93000E+00	0.10105E-01	0.42584E-01
98	0.95000E+00	0.72366E-02	0.94521E-01
99	0.97000E+00	0.43440E-02	0.15781E+00
100	0.99000E+00	0.14480E-02	0.25966E+00

1 INVISCID WAKE RESULTS

PANEL	XP	YP	CP
101	0.10069E+01	0.83952E-04	0.30200E+00
102	0.10222E+01	0.30601E-03	0.21875E+00
103	0.10413E+01	0.65883E-03	0.17465E+00
104	0.10649E+01	0.11943E-02	0.14324E+00
105	0.10942E+01	0.19821E-02	0.11851E+00
106	0.11305E+01	0.31169E-02	0.98103E-01
107	0.11755E+01	0.47269E-02	0.80872E-01
108	0.12313E+01	0.69828E-02	0.66183E-01
109	0.13005E+01	0.10111E-01	0.53640E-01
110	0.13863E+01	0.14408E-01	0.42979E-01
111	0.14926E+01	0.20260E-01	0.33995E-01
112	0.16244E+01	0.28163E-01	0.26515E-01
113	0.17878E+01	0.38747E-01	0.20374E-01
114	0.19903E+01	0.52814E-01	0.15419E-01
115	0.22414E+01	0.71365E-01	0.11487E-01
116	0.25526E+01	0.95653E-01	0.84237E-02
117	0.29393E+01	0.12723E+00	0.60806E-02
118	0.34165E+01	0.16902E+00	0.43224E-02
119	0.38364E+01	0.20480E+00	0.33384E-02

CL = 0.71900E+00

INPUT DATA FOR BOUNDARY-LAYER CALCULATIONS

IWAKE	NXT	NW	ITREND	
1	161	37	40	
ITR(1)	ITR(2)	ISWPMX	10**-6*RL	XOTR(1)
0	0	1	0.54	0.0
IP				
1				

0ITERATIONS EXCEEDED MAX IN WAKE B.L. CALCULATIONS AT
0INTERA= 4NX= 162IT= 20
0CALCULATIONS ARE ALLOWED TO CONTINUE

************** CYCLE 40 **************

BOUNDARY_LAYER PROPERTIES FOR THE LAST CYCLE

----- UPPER SURFACE ------

XCTR= 0.850E-01

NX	XC	XS	CF	DLS	UE	CP	IT
74	0.00888	0.003702	0.15032	0.00014	0.13991	0.98042	2
75	0.00540	0.008237	0.08157	0.00013	0.32881	0.89189	3
76	0.00294	0.012084	0.04906	0.00012	0.52154	0.72800	3
77	0.00140	0.015235	0.03732	0.00012	0.70635	0.50107	2
78	0.00056	0.017694	0.03117	0.00011	0.86148	0.25786	2
79	0.00018	0.019500	0.02762	0.00011	0.97807	0.04338	0
80	0.00003	0.020776	0.02539	0.00011	1.06098	-0.12567	3
81	0.00000	0.021748	0.02381	0.00011	1.12282	-0.26072	3
82	0.00003	0.022721	0.02239	0.00011	1.18295	-0.39937	3
83	0.00018	0.023996	0.02073	0.00011	1.25953	-0.58640	3
84	0.00056	0.025802	0.01865	0.00011	1.36203	-0.85513	3
85	0.00140	0.028261	0.01614	0.00012	1.48647	-1.20958	3
86	0.00294	0.031413	0.01327	0.00013	1.61299	-1.60172	3
87	0.00540	0.035260	0.01023	0.00015	1.70750	-1.91554	3
88	0.00888	0.039795	0.00757	0.00019	1.75175	-2.06861	3
89	0.01333	0.045011	0.00579	0.00023	1.76511	-2.11562	3
90	0.01841	0.050900	0.00474	0.00026	1.76972	-2.13192	3
91	0.02387	0.057456	0.00394	0.00030	1.77030	-2.13396	3
92	0.03052	0.064668	0.00300	0.00035	1.76054	-2.09950	3
93	0.03806	0.072526	0.00184	0.00044	1.73169	-1.99875	4
94	0.04609	0.081020	0.00082	0.00059	1.69096	-1.85935	4
95	0.05484	0.090139	0.00026	0.00076	1.65398	-1.73564	5
96	0.06424	0.099869	0.00004	0.00089	1.62448	-1.63892	4
97	0.07430	0.110198	0.00026	0.00093	1.58915	-1.52539	9
98	0.08498	0.121110	0.00137	0.00082	1.53150	-1.34548	2
99	0.09627	0.132591	0.00306	0.00075	1.49382	-1.23149	4
100	0.10814	0.144626	0.00438	0.00075	1.47910	-1.18775	3
101	0.12058	0.157198	0.00495	0.00078	1.47268	-1.16879	2
102	0.13357	0.170288	0.00507	0.00083	1.46562	-1.14805	2
103	0.14708	0.183881	0.00497	0.00090	1.45392	-1.11389	2
104	0.16108	0.197955	0.00481	0.00099	1.43861	-1.06960	2
105	0.17557	0.212494	0.00466	0.00108	1.42270	-1.02407	2
106	0.19052	0.227475	0.00453	0.00117	1.40737	-0.98069	2
107	0.20589	0.242880	0.00441	0.00126	1.39279	-0.93987	2
108	0.22168	0.258685	0.00431	0.00136	1.37837	-0.89991	2
109	0.23786	0.274871	0.00421	0.00146	1.36513	-0.86358	2
110	0.25439	0.291414	0.00413	0.00155	1.35299	-0.83059	2
111	0.27127	0.308292	0.00404	0.00165	1.34127	-0.79901	2
112	0.28846	0.325482	0.00394	0.00176	1.32926	-0.76693	2
113	0.30593	0.342959	0.00384	0.00187	1.31691	-0.73426	2
114	0.32367	0.360699	0.00375	0.00198	1.30442	-0.70151	2
115	0.34164	0.378667	0.00367	0.00210	1.29217	-0.66971	2
116	0.35980	0.396830	0.00361	0.00222	1.28062	-0.63999	2
117	0.37814	0.415179	0.00356	0.00234	1.26995	-0.61277	1
118	0.39667	0.433716	0.00351	0.00245	1.26014	-0.58796	1
119	0.41535	0.452411	0.00345	0.00256	1.25058	-0.56394	2
120	0.43414	0.471223	0.00338	0.00269	1.24058	-0.53903	2
121	0.45303	0.490131	0.00330	0.00282	1.23017	-0.51331	2
122	0.47197	0.509108	0.00324	0.00296	1.21986	-0.48805	2
123	0.49095	0.528118	0.00320	0.00309	1.21022	-0.46463	1
124	0.50994	0.547147	0.00316	0.00322	1.20121	-0.44292	1
125	0.52891	0.566158	0.00310	0.00335	1.19231	-0.42159	2
126	0.54783	0.585126	0.00304	0.00350	1.18320	-0.39997	2
127	0.56668	0.604025	0.00298	0.00364	1.17413	-0.37859	2
128	0.58542	0.622828	0.00294	0.00379	1.16549	-0.35836	2
129	0.60405	0.641511	0.00289	0.00393	1.15718	-0.33906	2
130	0.62252	0.660047	0.00285	0.00408	1.14896	-0.32011	2
131	0.64081	0.678411	0.00279	0.00423	1.14078	-0.30139	2
132	0.65890	0.696577	0.00274	0.00438	1.13270	-0.28302	2
133	0.67677	0.714521	0.00269	0.00454	1.12481	-0.26519	2

134	0.69439	0.732218	0.00264	0.00470	1.11693	-0.24753	2
135	0.71173	0.749643	0.00259	0.00487	1.10914	-0.23019	2
136	0.72878	0.766773	0.00253	0.00504	1.10145	-0.21318	2
137	0.74550	0.783584	0.00248	0.00521	1.09380	-0.19640	2
138	0.76188	0.800052	0.00242	0.00539	1.08618	-0.17979	2
139	0.77789	0.816156	0.00235	0.00558	1.07851	-0.16318	2
140	0.79351	0.831873	0.00228	0.00578	1.07085	-0.14672	2
141	0.80873	0.847182	0.00222	0.00598	1.06328	-0.13056	2
142	0.82351	0.862061	0.00215	0.00619	1.05581	-0.11473	2
143	0.83785	0.876489	0.00207	0.00641	1.04838	-0.09910	2
144	0.85171	0.890448	0.00199	0.00663	1.04096	-0.08359	2
145	0.86509	0.903918	0.00189	0.00688	1.03340	-0.06792	2
146	0.87795	0.916880	0.00177	0.00715	1.02559	-0.05184	2
147	0.89029	0.929317	0.00162	0.00745	1.01739	-0.03507	2
148	0.90208	0.941211	0.00147	0.00778	1.00885	-0.01779	2
149	0.91331	0.952546	0.00131	0.00815	1.00016	-0.00032	2
150	0.92397	0.963307	0.00115	0.00854	0.99160	0.01672	3
151	0.93404	0.973478	0.00100	0.00895	0.98336	0.03299	3
152	0.94351	0.983046	0.00087	0.00937	0.97557	0.04826	3
153	0.95237	0.991998	0.00075	0.00979	0.96835	0.06229	3
154	0.96061	1.000321	0.00064	0.01021	0.96173	0.07507	3
155	0.96821	1.008003	0.00054	0.01062	0.955[illegible]	0.08662	3
156	0.97517	1.015035	0.00044	0.01102	0.95029	0.09694	3
157	0.98147	1.021406	0.00036	0.01140	0.94548	0.10606	3
158	0.98712	1.027108	0.00029	0.01176	0.94126	0.11403	3
159	0.99209	1.032134	0.00023	0.01209	0.93761	0.12088	3
160	0.99639	1.036475	0.00017	0.01238	0.93453	0.12666	3
161	1.00000	1.040126	0.00011	0.01265	0.93195	0.13147	3

----- LOWER SURFACE ------

XCTR= 0.944E+00

NX	XC	XS	CF	DIS	UE	CP	IT
89	0.01333	0.001514	0.35755	0.00014	0.05723	0.99672	2
90	0.01841	0.007404	0.13898	0.00016	0.26180	0.93146	3
91	0.02387	0.013959	0.04285	0.00018	0.44075	0.80574	3
92	0.03052	0.021171	0.02499	0.00021	0.56141	0.68482	3
93	0.03806	0.029030	0.01728	0.00026	0.62934	0.60393	3
94	0.04609	0.037524	0.01352	0.00030	0.67535	0.54390	0
95	0.05484	0.046643	0.01117	0.00034	0.71733	0.48543	3
96	0.06424	0.056373	0.00957	0.00037	0.75644	0.42790	3
97	0.07430	0.066701	0.00837	0.00040	0.79111	0.37414	3
98	0.08498	0.077614	0.00734	0.00044	0.81986	0.32783	3
99	0.09627	0.089095	0.00653	0.00048	0.84452	0.28679	3
100	0.10814	0.101130	0.00590	0.00051	0.86609	0.24990	3
101	0.12058	0.113701	0.00536	0.00054	0.88542	0.21604	3
102	0.13357	0.126792	0.00489	0.00058	0.90187	0.18664	3
103	0.14708	0.140385	0.00449	0.00062	0.91600	0.16095	3
104	0.16108	0.154459	0.00417	0.00065	0.92896	0.13704	3
105	0.17557	0.168998	0.00387	0.00069	0.94062	0.11524	3
106	0.19052	0.183979	0.00359	0.00073	0.95018	0.09716	3
107	0.20589	0.199384	0.00334	0.00077	0.95838	0.08151	3
108	0.22168	0.215190	0.00314	0.00081	0.96566	0.06751	3
109	0.23786	0.231375	0.00295	0.00084	0.97198	0.05526	3
110	0.25439	0.247918	0.00281	0.00088	0.97777	0.04396	3
111	0.27127	0.264797	0.00267	0.00092	0.98337	0.03298	3
112	0.28846	0.281986	0.00252	0.00095	0.98811	0.02363	3
113	0.30593	0.299464	0.00237	0.00100	0.99165	0.01662	2
114	0.32367	0.317203	0.00222	0.00104	0.99417	0.01163	2
115	0.34164	0.335171	0.00210	0.00109	0.99595	0.00808	2
116	0.35980	0.353334	0.00200	0.00113	0.99732	0.00535	2
117	0.37814	0.371683	0.00193	0.00117	0.99871	0.00258	2
118	0.39667	0.390221	0.00187	0.00121	1.00031	-0.00062	2
119	0.41535	0.408916	0.00181	0.00124	1.00197	-0.00395	2
120	0.43414	0.427728	0.00173	0.00128	1.00311	-0.00624	2
121	0.45303	0.446636	0.00163	0.00133	1.00340	-0.00682	3
122	0.47198	0.465613	0.00154	0.00138	1.00306	-0.00612	3

123	0.49095	0.484624	0.00149	0.00143	1.00262	-0.00524	3
124	0.50994	0.503652	0.00145	0.00147	1.00243	-0.00486	2
125	0.52891	0.522663	0.00140	0.00150	1.00230	-0.00460	2
126	0.54783	0.541631	0.00134	0.00155	1.00178	-0.00357	3
127	0.56668	0.560530	0.00128	0.00160	1.00087	-0.00173	3
128	0.58543	0.579334	0.00123	0.00164	0.99985	0.00030	3
129	0.60405	0.598017	0.00120	0.00168	0.99895	0.00210	2
130	0.62252	0.616553	0.00116	0.00172	0.99806	0.00387	2
131	0.64081	0.634916	0.00112	0.00177	0.99702	0.00595	3
132	0.65891	0.653083	0.00108	0.00181	0.99581	0.00836	3
133	0.67677	0.671027	0.00104	0.00186	0.99450	0.01096	3
134	0.69439	0.688723	0.00100	0.00190	0.99310	0.01375	3
135	0.71173	0.706149	0.00096	0.00195	0.99162	0.01670	3
136	0.72878	0.723279	0.00092	0.00199	0.99006	0.01978	3
137	0.74550	0.740090	0.00088	0.00204	0.98842	0.02303	3
138	0.76188	0.756558	0.00084	0.00209	0.98668	0.02647	3
139	0.77789	0.772662	0.00079	0.00215	0.98483	0.03011	3
140	0.79352	0.788379	0.00074	0.00221	0.98287	0.03397	3
141	0.80873	0.803688	0.00069	0.00227	0.98080	0.03804	3
142	0.82352	0.818567	0.00064	0.00233	0.97862	0.04231	3
143	0.83785	0.832996	0.00059	0.00240	0.97637	0.04670	3
144	0.85172	0.846954	0.00053	0.00248	0.97407	0.05119	3
145	0.86509	0.860424	0.00047	0.00255	0.97173	0.05573	3
146	0.87796	0.873387	0.00041	0.00264	0.96935	0.06037	3
147	0.89030	0.885823	0.00034	0.00274	0.96693	0.06505	3
148	0.90210	0.897717	0.00026	0.00287	0.96448	0.06978	3
149	0.91334	0.909052	0.00018	0.00302	0.96204	0.07448	3
150	0.92400	0.919813	0.00009	0.00320	0.95970	0.07897	3
151	0.93407	0.929985	0.00002	0.00341	0.95757	0.08306	4
152	0.94354	0.939553	-0.00006	0.00366	0.95569	0.08665	9
153	0.95240	0.948504	-0.00016	0.00391	0.95402	0.08985	8
154	0.96063	0.956827	-0.00030	0.00417	0.95231	0.09310	8
155	0.96823	0.964510	-0.00050	0.00445	0.95021	0.09711	2
156	0.97518	0.971542	-0.00074	0.00469	0.94748	0.10229	7
157	0.98149	0.977913	-0.00106	0.00488	0.94382	0.10921	9
158	0.98713	0.983615	-0.00144	0.00501	0.93889	0.11849	8
159	0.99210	0.988640	-0.00181	0.00509	0.93267	0.13013	7
160	0.99639	0.992982	-0.00208	0.00512	0.92576	0.14296	6
161	1.00000	0.996632	-0.00223	0.00511	0.91935	0.15480	4

----- WAKE ------

I	XS	UE	CP	DLS	H	ITR
161	1.04119	0.92565	0.14318	0.01781	2.33014	0
162	1.04769	0.91681	0.15945	0.01604	2.00843	2
163	1.05492	0.92128	0.15124	0.01449	1.83997	3
164	1.06297	0.92771	0.13935	0.01336	1.75191	3
165	1.07193	0.93190	0.13157	0.01264	1.68424	3
166	1.08191	0.93520	0.12541	0.01208	1.63208	3
167	1.09303	0.93674	0.12251	0.01169	1.58876	2
168	1.10541	0.93908	0.11813	0.01129	1.54734	2
169	1.11919	0.94130	0.11394	0.01091	1.50866	2
170	1.13454.	0.94431	0.10828	0.01051	1.47057	2
171	1.15163	0.94776	0.10175	0.01012	1.43422	2
172	1.17066	0.95176	0.09415	0.00974	1.39982	2
173	1.19185	0.95609	0.08589	0.00937	1.36780	2
174	1.21545	0.96029	0.07784	0.00903	1.33817	2
175	1.24173	0.96418	0.07035	0.00873	1.31099	2
176	1.27099	0.96780	0.06337	0.00846	1.28600	2
177	1.30357	0.97110	0.05696	0.00822	1.26354	2
178	1.33985	0.97422	0.05090	0.00800	1.24320	2
179	1.38025	0.97700	0.04547	0.00781	1.22443	2
180	1.42523	0.97960	0.04038	0.00763	1.20706	2
181	1.47533	0.98193	0.03582	0.00748	1.19108	2
182	1.53111	0.98409	0.03157	0.00733	1.17633	2
183	1.59322	0.98602	0.02777	0.00720	1.16280	2
184	1.66239	0.98778	0.02428	0.00709	1.15040	2
185	1.73941	0.98935	0.02118	0.00699	1.13904	2

186	1.82517	0.99078	0.01835	0.00689	1.12864	2
187	1.92067	0.99204	0.01585	0.00681	1.11910	2
188	2.02702	0.99317	0.01361	0.00673	1.11030	2
189	2.14543	0.99416	0.01164	0.00666	1.10217	2
190	2.27730	0.99504	0.00990	0.00660	1.09469	2
191	2.42413	0.99580	0.00839	0.00654	1.08789	2
192	2.58763	0.99647	0.00705	0.00649	1.08171	2
193	2.76969	0.99705	0.00590	0.00645	1.07607	2
194	2.97243	0.99755	0.00490	0.00641	1.07090	2
195	3.19818	0.99797	0.00405	0.00638	1.06616	2
196	3.44956	0.99833	0.00333	0.00635	1.06177	2
197	3.72949	0.99863	0.00273	0.00632	1.05778	2
198	4.04119	0.99859	0.00281	0.00000	0.00000	0

SUMMARY OF THE DRAG, LIFT AND PITCHING MOMENT
COEFFICIENTS WITH THE CYCLE

CD IS EVALUATED FROM FAR-WAKE FORMULA
CL & CM FROM INTEGRATION OF CP

CYCLE	CD	CL	CM
1	0.012592	0.011899	0.665033
2	0.011748	0.012118	0.665977
3	0.010915	0.011455	0.665906
4	0.012261	0.011898	0.665108
5	0.012459	0.012117	0.666036
6	0.011849	0.011452	0.666010
7	0.012065	0.011898	0.665143
8	0.012272	0.012116	0.666060
9	0.011606	0.011458	0.666076
10	0.011963	0.011896	0.665151
11	0.012167	0.659153	0.002399
12	0.011474	0.658606	0.002286
13	0.011860	0.658523	0.002802
14	0.012045	0.658816	0.002517
15	0.011458	0.657755	0.002500
16	0.011853	0.658142	0.002906
17	0.012176	0.658669	0.002550
18	0.011600	0.659903	0.002041
19	0.011900	0.661304	0.002273
20	0.012125	0.662937	0.001650
21	0.011497	0.663051	0.001381
22	0.011887	0.663583	0.001815
23	0.012100	0.664779	0.001284
24	0.011466	0.664628	0.001075
25	0.011872	0.664564	0.001613
26	0.012094	0.665672	0.001124
27	0.011457	0.665504	0.000910
28	0.011898	0.664903	0.001520
29	0.012119	0.665869	0.001091
30	0.011455	0.665745	0.000863
31	0.011899	0.665033	0.001489
32	0.012118	0.665977	0.001061
33	0.011455	0.665906	0.000835
34	0.011898	0.665108	0.001471
35	0.012117	0.666036	0.001050
36	0.011452	0.666010	0.000917
37	0.011898	0.665143	0.001461
38	0.012116	0.666060	0.001046
39	0.011458	0.666076	0.000806
40	0.011896	0.665151	0.001457

Source code: **pres.f**


```
      program pres

      REAL cp(100)

      OPEN(unit=20,file='cp.in',status='unknown')

      print *,'no. points before bubble?'
      read *,N

      print *,'no. points after bubble?'
      read *,M

c     cp(1) corresponds to stagnation for M=.3, dark fringe 5.5
c     cp(2) corresponds to next dark fringe 4.5, etc.

      do i=1,N
        read(20,*) cp(i)
      end do

      do i=1,N
        read(1,*) k,x,y
        write(2,*) x,-cp(i)
      end do

      do i=N,N-M+1,-1
        read(1,*) k,x,y
        write (2,*) x,-cp(i)
      end do

      end
```

APPENDIX B: BASIC COMPUTER COMMANDS

The following collection of commands is intended as a very basic user's guide to the various support programs which are necessary to do research on a UNIX system. It is by no means a comprehensive list, simply enough to get started without wasting a great deal of time on finding elementary procedures and syntax. In some commands, the arbitrary word *filename* or abbreviation *fn* is used. In others, specific examples are used when it aids the clarity of the explanation. Both methods, however, indicate that the user may substitute an appropriate name.

BASIC UNIX COMMANDS

man *filename*	obtain on-line help information for a program or command from the on-line manual
cp fn1 fn2	copy filename1 to filename2, both in current directory
cp /alpha/nowak/bl/fn1 .	copy fn1 from another directory, path specified, to the name in the current directory
cp ../fn1 .	copy fn1 from the directory above to the same name
mv fn1 fn2	move, or rename, fn1 to fn2 ; fn1 will no longer exist
more fn	type the text of the file on the screen, read only
!v	repeat the last command that started with v
cd *paneldir*	change directory to paneldir
rm fn	delete (remove) fn
mkdir *paneldir*	make directory (example name paneldir)
rmdir *paneldir*	remove (delete) directory
ls	list contents of directory (like dir on a pc)
pwd	print working directory
batch < fn	execute a command file in batch mode (runs even after logging off) - useful for long run-time programs

telnet 131.120.254.92 (suzqt -Stardent)
131.120.254.91 (madmax - IRIS)

BASIC EDITING COMMANDS FOR THE VI EDITOR

NOTE: ALL commands in VI are **case sensitive**, type exactly as shown. Check the status of the CAPS LOCK key if a command does not seem to work properly.

vi *fn1*	(invokes editor, calls old file if it exists, otherwise creates new file)

To get started into text mode from command mode:

a	(add to document, cursor moves to right and input is enabled)
i	(insert, cursor does not move and input is enabled)
o	(open a new line below the cursor, input is enabled)

To return to command mode:

Esc	(disables input, enables move, write, save, etc.)

NOTE: All of the following commands assume command mode

To move around:

ctrl-h	move 1 left
ctrl-l	move 1 right
ctrl-k	move 1 up
ctrl-l	move 1 down

The above commands will always work. On some machines, the arrow keys MAY work as well. Other commands:

1(shift)G	go to first line
(shift)G	go to last line
/bl2D	search for the next text string "bl2D" after the cursor location, CASE SENSITIVE
/(Enter key)	search for another occurrence of the previous search string

When done editing or viewing:

(Esc):w	write, or save, but do not exit
(Esc):q	quit, exits only if no modifications were made
(Esc):q!	quit, exits without saving even if modifications were made
(Esc):wq	write quit, exits and saves all modifications to the original fn
(Esc):wq newfn	write quit, exits and saves all modifications to a new fn

To delete:

dd	delete current line
2dd	delete current line and next line
10dd	delete current line and next 9 lines

To cut (copy) and paste:

5yy	yanks 5 lines to buffer (leaves original 5 lines also)
move cursor to desired location	
p	pastes the 5 lines

BASIC GNUPLOT COMMANDS

NOTE: The following is a list of some example commands. Extensive on-line help is available by typing **help** while in gnuplot.

set term tek40xx (Stardent)
plot "VEL.DAT"
plot "VEL.DAT" with lines
set xrange [0:1]
set yrange [0:5]
plot "cf.dat", "dls.dat" (two different data files)
plot "cf.dat" with lines, "dls.dat" with lines
plot "bl.dat" using 2:5 (one data file with multiple columns)
plot "bl.dat" using 2:5, "bl.dat" using 2:6
set data style lines (option *with lines* will then not be needed after each plot command)
set key (legend)
set nokey (no legend)
set grid
set nogrid
set nozeroaxis

To print (these commands can be put in a command file):
set term postscript
set output "gnuout"
replot
set term tek40xx (reset to terminal being used)
set output

FILE TRANSFER USING FTP

ftp suzqt
ftp madmax (numerical computer address may be used instead)

get fn
get oldfn newfn
put fn
quit

INITIAL DISTRIBUTION LIST

1.	Defense Technical Information Center Cameron Station Alexandria, Virginia 22304-6145	2
2.	Library, Code 0142 Naval Postgraduate School Monterey, California 93943-5002	2
3.	Dr. M. F. Platzer Dept. of Aeronautics and Astronautics, Code AA/PL Naval Postgraduate School Monterey, California 93943-5002	5
4.	Professor M. S. Chandrasekhara M. S. 260-1 NASA Ames Research Center Moffett Field, California 94035	1
5.	Dr. L. W. Carr M. S. 260-1 NASA Ames Research Center Moffett Field, California 94035	1
6.	Dr. T. L. Doligalski Chief, Fluid Dynamics Division U. S. Army Research Office P. O. Box 12211 Research Triangle Park, North Carolina 27709	1
7.	MAJ D. B. Fant Program Manager, AFOSR/NA Billing AFB Washington, District of Columbia 20032	1
8.	LT L. M. Nowak, USN 701 Wilson Court Lexington Park, Maryland 20653	2