AEDC-TR-76-27

EVALUATION OF A METHOD FOR COMPUTATION OF SEPARATED, TURBULENT, COMPRESSIBLE BOUNDARY LAYERS

PROPULSION WIND TUNNEL FACILITY
ARNOLD ENGINEERING DEVELOPMENT CENTER
AIR FORCE SYSTEMS COMMAND
ARNOLD AIR FORCE STATION, TENNESSEE 37389

February 1976

Final Report for Period January 2, 1974 — February 28, 1975

Prepared for

DIRECTORATE OF TECHNOLOGY (DY)
ARNOLD ENGINEERING DEVELOPMENT CENTER
ARNOLD AIR FORCE STATION, TENNESSEE 37389

NOTICES

When U. S. Government drawings specifications, or other data are used for any purpose other than a definitely related Government procurement operation, the Government thereby incurs no responsibility nor any obligation whatsoever, and the fact that the Government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data, is not to be regarded by implication or otherwise, or in any manner licensing the holder or any other person or corporation, or conveying any rights or permission to manufacture, use, or sell any patented invention that may in any way be related thereto.

Qualified users may obtain copies of this report from the Defense Documentation Center.

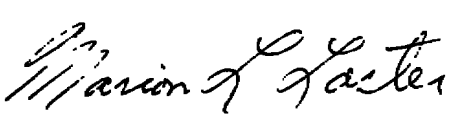
References to named commercial products in this report are not to be considered in any sense as an endorsement of the product by the United States Air Force or the Government.

This report has been reviewed by the Information Office (OI) and is releasable to the National Technical Information Service (NTIS). At NTIS, it will be available to the general public, including foreign nations.

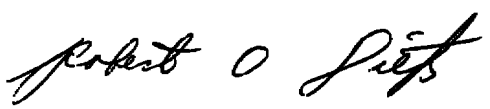
APPROVAL STATEMENT

This technical report has been reviewed and is approved for publication.

FOR THE COMMANDER



MARION L. LASTER
Research and Development
Division
Directorate of Technology



ROBERT O. DIETZ
Director of Technology

REPORT DOCUMENTATION PAGE	READ INSTRUCTIONS BEFORE COMPLETING FORM
1 REPORT NUMBER: AEDC-TR-76-27	2 GOVT ACCESSION NO. / 3 RECIPIENT'S CATALOG NUMBER
4 TITLE (and Subtitle): EVALUATION OF A METHOD FOR COMPUTATION OF SEPARATED, TURBULENT, COMPRESSIBLE BOUNDARY LAYERS	5 TYPE OF REPORT & PERIOD COVERED: Final Report-January 2, 1974-February 28, 1975 / 6 PERFORMING ORG REPORT NUMBER
7 AUTHOR(s): M. C. Altstatt, ARO, Inc.	8 CONTRACT OR GRANT NUMBER(s)
9 PERFORMING ORGANIZATION NAME AND ADDRESS: Arnold Engineering Development Center (DY) Arnold Air Force Station, Tennessee 37389	10 PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS: Program Element 65807F
11 CONTROLLING OFFICE NAME AND ADDRESS: Arnold Engineering Development Center (DYFS) Arnold Air Force Station, Tennessee 37389	12 REPORT DATE: February 1976 / 13 NUMBER OF PAGES: 44
14 MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)	15 SECURITY CLASS. (of this report): UNCLASSIFIED / 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE: N/A

16 DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17 DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18 SUPPLEMENTARY NOTES

Available in DDC

19 KEY WORDS (Continue on reverse side if necessary and identify by block number)

coding
computer programs
computations
turbulent boundary layer
compressible flow
separated flow
velocity (external)

20 ABSTRACT (Continue on reverse side if necessary and identify by block number)

A computer code for a turbulent, compressible boundary-layer method, capable of carrying out computations in a region of separated flow, is developed and tested. The procedure after separation is to specify either friction velocity or boundary-layer thickness as an independent variable and obtain external velocity as a dependent variable. This requires a trial and error alteration of the specified variable in order to match

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

20. ABSTRACT (Continued)

the desired experimental or computed external velocity. Satisfactory results were obtained by this method in the analysis of certain specialized cases of separated flow.

AFSC
Arnold AFS Tenn

PREFACE

The work reported herein was conducted by the Arnold Engineering Development Center (AEDC), Air Force Systems Systems Command (AFSC), under Program Element 65807F. The results of the research presented were obtained by ARO, Inc. (a subsidiary of Sverdrup & Parcel and Associates, Inc.), contract operator of AEDC, AFSC, Arnold Air Force Station, Tennessee. The work was conducted under ARO Project Nos. PF409 and P32A-31A. The author of this report was M. C. Altstatt, ARO, Inc. The manuscript (ARO Control No. ARO-PWT-TR-75-103) was submitted for publication on June 27, 1975.

CONTENTS

ILLUSTRATIONS

Figure

Figure Page

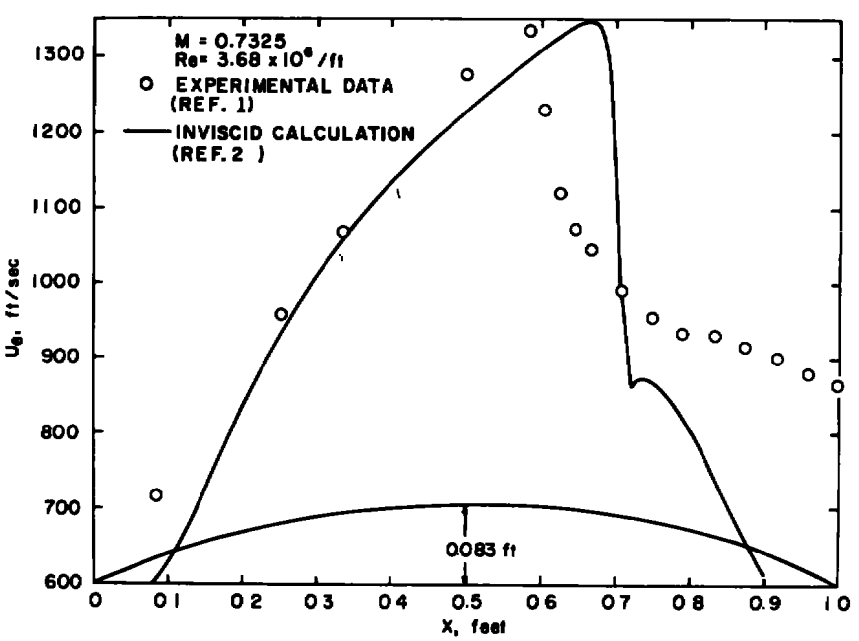
APPENDIXES

1.0 INTRODUCTION

The computation of the flow field with viscous effects over an airfoil or similar body is usually carried out by an iterative method. The first step is to obtain an inviscid outer solution of the basic profile. A displacement thickness is then calculated by using the resulting velocity or pressure distribution in a boundary-layer solution. This displacement thickness is added to the initial body profile to form an equivalent body and the process is repeated. Convergence is normally achieved with a few iterations.

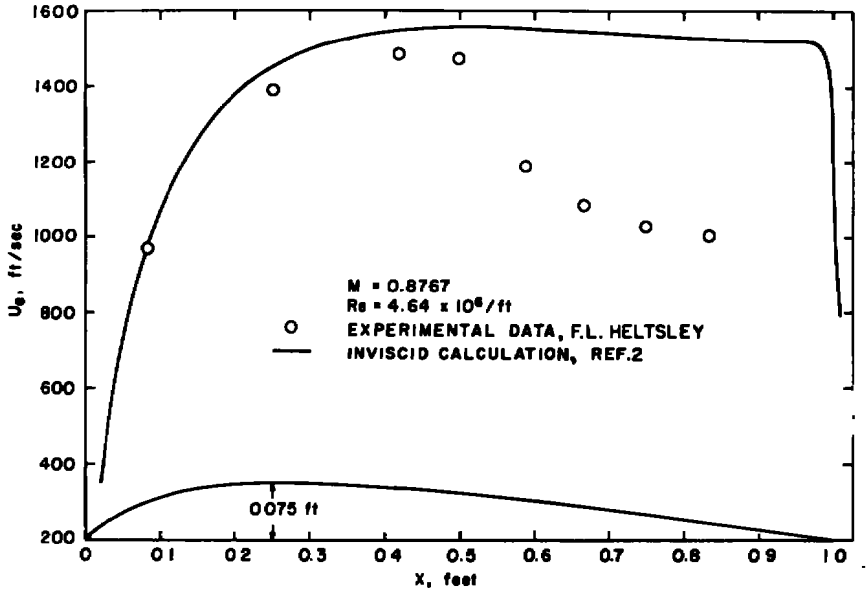
This iterative procedure is not often successful, however, if significant flow separation occurs. Standard boundary-layer methods, if used in a separated flow region, produce highly inaccurate displacement thicknesses. This is particularly true in transonic flows in which a shock appears, due to the strong pressure gradients induced by the shock and possible shock-induced separation.

Two examples of transonic flows exhibiting the shock boundary-layer interaction are shown in Fig. 1. The first case, a modified circular arc bump mounted on a wind tunnel floor, was investigated by Alber. et al. (Ref. 1). The second velocity distribution consists of data from the test of an airfoil-shaped bump mounted on the floor of AEDC Aerodynamic Wind Tunnel (1T), a test performed by F. L. Heltsley in May 1974. While



a. Circular arc bump

Figure 1. Comparison of experimental velocity distributions with velocities obtained by inviscid computation.



b. Airfoil-shaped bump
Figure 1. Concluded.

this second flow is not fully separated, it presents the same difficulties due to the greatly increased boundary-layer thickness downstream of the shock.

A comparison of the experimentally measured velocities with those obtained using Murman's inviscid solution (Ref. 2) clearly indicates the error resulting from ignoring the shock boundary-layer interaction. The large increase in boundary-layer thickness caused by the shock forms an equivalent body greatly different from the original body.

As part of an effort to extend an iterative method of solution to separated flows of the type illustrated in Fig. 1, a computer program for a turbulent, compressible, boundary-layer method, capable of carrying out computations in a separated flow region, has been developed and tested. The procedure used for extension through the separation point follows that used by Kuhn and Nielsen (Ref. 3). Instead of specifying a pressure or external velocity distribution, friction velocity is specified, so that the external velocity is obtained as a result. The prescribed friction velocity is adjusted so that the external velocity computed by the boundary-layer method matches a velocity measured experimentally or an inviscid computation. Alternately, in the present method, the boundary-layer thickness can be specified in the same manner.

By prescribing either the friction velocity or the boundary-layer thickness, the singularity occurring as skin friction goes to zero is avoided, more accurate boundary-layer

velocity profiles are produced, and the displacement thicknesses obtained compare well with experimental results.

2.0 BOUNDARY-LAYER METHOD FOR ATTACHED FLOW

The turbulent, compressible, boundary-layer method used is a variation of the method of Nash and Hicks (Ref. 4) using a modification of Cole's wall-wake velocity representation (Ref. 5) to replace the shear stress equation proposed by Nash. The Stewartson Transformations (Ref. 6) are included for compressibility effects. This method was modified by Kuhn for boundary-layer analysis in a region of separated flow (Ref. 3).

The reasons for choosing this method are speed, flexibility, and the previous experience gained by Kuhn in and near separated regions. The speed is particularly important as the future use of this method is part of an iterative solution.

2.1 DEVELOPMENT OF THE BOUNDARY-LAYER EQUATIONS

Starting with the continuity, momentum and energy equations for a turbulent, compressible boundary layer (Ref. 3)

$$\frac{\partial}{\partial \tilde{x}} (\rho u) + \frac{\partial}{\partial \tilde{y}} (\rho v) = 0 \tag{1}$$

$$\rho u \frac{\partial u}{\partial \tilde{x}} + \rho v \frac{\partial u}{\partial \tilde{y}} = \frac{\partial p}{\partial \tilde{y}} \left(\mu\beta \frac{\partial u}{\partial \tilde{y}} \right) \tag{2}$$

$$\rho u \frac{\partial S}{\partial \tilde{x}} + \rho v \frac{\partial S}{\partial \tilde{y}} = \frac{\partial p}{\partial \tilde{x}} + \frac{\partial}{\partial \tilde{y}} \left(\mu\beta \frac{\partial S}{\partial \tilde{y}} \right) \tag{3}$$

where

$$S = T/T_t - 1 \tag{4}$$

The Stewartson Transformations (Ref. 6) are applied in the following form

$$x = \int_0^{\tilde{x}} \frac{p_e a_e}{p_\infty a_\infty} d\tilde{x} \tag{5}$$

$$y = \int_0^{\tilde{y}} \frac{\rho_e a_e}{\rho_\infty a_\infty} \frac{\rho}{\rho_e} d\tilde{y} \tag{6}$$

$$U = \frac{a_\infty}{a_e} u \tag{7}$$

$$V = \frac{p_\infty}{p_e}\left(\frac{a_\infty}{a_e}\right)^2 u \frac{\partial}{\partial x} \int_0^{\tilde{y}} \frac{\rho_e a_e \rho}{\rho_\infty a_\infty \rho_e} d\tilde{y} + \frac{\rho_\infty a_\infty}{p_e a_e} \frac{\rho}{\rho_e} v \tag{8}$$

With the assumptions that the fluid is a perfect gas, the laminar and turbulent Prandtl numbers are unity, viscosity is linear with temperature, and the wall is adiabatic, the set of Eqs. (1 through 4) can be transformed to

$$U_x \pm V_y = 0 \tag{9}$$

$$UU_x + VU_y - (S + 1) U_e U_{e_x} - \nu(\beta U_y)_y = 0 \tag{10}$$

$$S = \frac{T_{AW}}{T_t} - 1 = \text{constant} \tag{11}$$

Equations (9 through 11) are combined using the usual boundary-layer integral approach (Ref. 4).

$$\int_0^\delta \left\{ UU_x - U_y \int_0^y U_x d\eta - \left(\frac{T_{AW}}{T_t}\right) U_e U_{e_x} - \nu(\beta\ U_y)_y \right\} y^n dy = 0 \tag{12}$$

Equation (12) is the integral of the momentum across the boundary layer for n = 0, and the moment of momentum for n = 1.

With the addition of a boundary-layer velocity profile representation and an expression for eddy viscosity, a closed set of equations can be formed. Crocco's theorem with a relaxation factor of 0.89 is used to determine density gradient across the boundary layer. The velocity profile is taken in the form given by Coles (Ref. 5) with an exponential term added for the viscous sublayer:

$$U = u_\tau[2.5\ \ell n\ (1 + y^+) + 5.1 - (3.39y^+ + 5.1)e^{-0.37y^+}] + \frac{u_\beta}{2}\left(1 - \cos\frac{\pi y}{\delta}\right) \tag{13}$$

where

$$y^+ = |u_\tau| y/\nu \tag{14}$$

and

$$u_\tau = \frac{\tau_w}{|\tau_w|}\left(\frac{|\tau_w|}{\rho}\right)^{1/2} \tag{15}$$

The friction velocity, u_τ, is defined in this way to allow for reverse flow in a separated region. The wake velocity, $u\beta$, can be eliminated by setting $y = \delta$ in Eq. (13):

$$u\beta = U_e - u_\tau(2.5\ \ell n\ (1 + \delta^+) + 5.1) \tag{16}$$

For the unseparated flow, the eddy viscosities are expressed in two forms, for an inner and outer region (Refs. 7 and 8).

$$\beta_i = 1 + 0.0533\left\{e^{0.41U^+} - \left[1 + 0.41U^+ + \frac{1}{2}(0.41U^+)^2\right]\right\} \tag{17}$$

where

$$U^+ = U/u_\tau \tag{18}$$

and

$$\beta_o = 1 + K\left(1 + 5.5\left(\frac{y}{\delta}\right)^6\right)^{-1} U_e\delta^*/\nu \tag{19}$$

with K taken as 0.0168 for favorable pressure gradients and

$$K = 0.013 + 0.0038\ \exp\left(-\delta^*\ \frac{dp}{dx}/15\tau_w\right) \tag{20}$$

for dp/dx positive (Ref. 3).

In a separated flow region, following Alber (Ref. 9), the eddy viscosity is given by

$$\beta = 0.013\left[1 + 5.5\left(\frac{y}{\delta}\right)^6\right]^{-1}\ \frac{U_e}{\nu}\int_{y\,u=0}^{\delta}\left(1 - \frac{U}{U_e}\right)dy \tag{21}$$

Substituting the velocity, Eq. (13), and its derivatives into Eq. (12) results in the two equations

$$A_{11}\ u_\tau' + A_{12}\ \delta' + A_{13}\ U_e' = -u_\tau|u_\tau| \tag{22}$$

$$A_{21}\ u_\tau' + A_{22}\ \delta' + A_{23}\ U_e' = -\nu\int_0^\delta \beta U_y\,dy \tag{23}$$

The prime denotes differentiation with respect to x. The A_{ij} coefficients are functions of u_τ, U_e and δ as defined in Appendix A.

2.2 METHOD OF SOLUTION

Given the inviscid velocity distribution, Eqs. (22) and (23) form an initial value problem in the dependent variables u_T and δ. In addition to the velocity, initial values of u_T, δ, and the free-stream thermodynamic state are required. Input is in terms of the physical variables and both compressible and incompressible results are output.

Equations (22) and (23) were integrated using a fourth order Runge-Kutta method (Ref. 10) with the step size in x of the order of δ. The interval for integration across the boundary layer was $\delta/20$ for the cases shown. Accuracy is not very sensitive to the step size chosen if it is in that range. The time required for each streamwise integration step is approximately 0.05 sec on an IBM 370/165. A listing of the computer program is given as Appendix B.

Note that the problem is formally the same if either u_T or δ is specified and the external velocity is treated as a dependent variable.

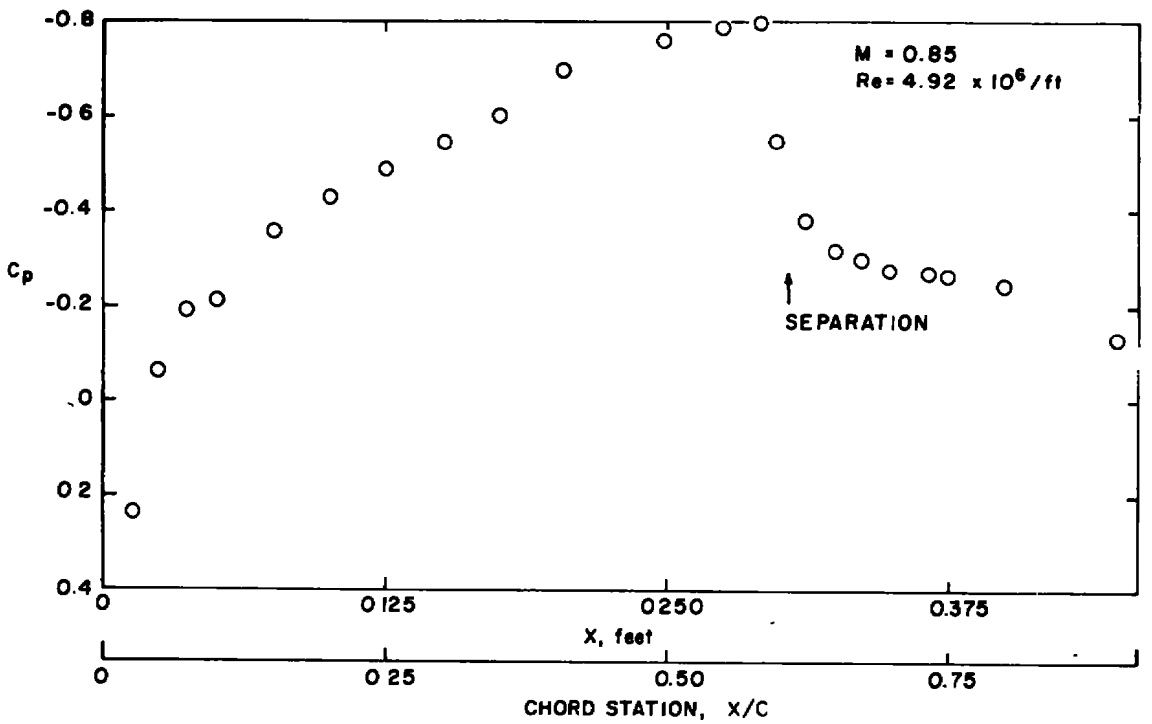
2.3 TESTING OF THE VALIDITY OF THE METHOD

The performance of the present boundary-layer method was checked by comparison with other boundary-layer methods and with experimental data. Figure 2a shows the pressure distribution on the upper surface of a 6-in. C-141 airfoil taken in Tunnel 16T (Ref. 11). The boundary-layer characteristics shown in Figs. 2b through e were calculated, using the C-141 data, by the present method, the method of Nash and Hicks (Ref. 4), the method of Patankar and Spalding (Ref. 12) as modified by High and Felderman (Ref. 13), and by Adams (Ref. 14).

There is considerable difference in values for some of the results, particularly shape factor and friction coefficient (Figs. 2b and c). This difference, with the exception of shape factor variation, is not unusually large, even for comparison of incompressible

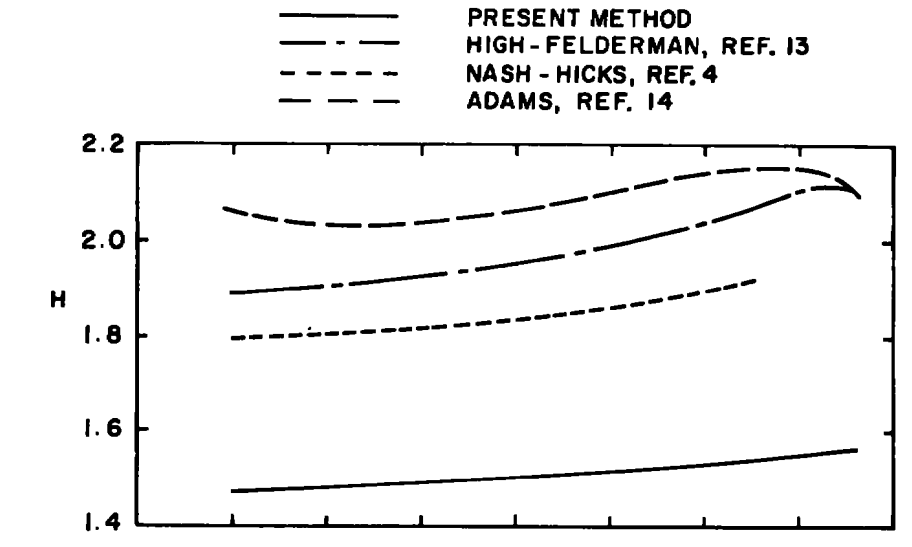
methods in moderate pressure gradients (Ref. 15). The momentum thicknesses compared in Fig. 2d are in better agreement, and the displacement thicknesses (Fig. 2e) are quite close. Since the result of principal interest for present purposes is displacement thickness, the agreement is considered satisfactory.

The comparison with experimental data is shown in Figs. 3a and b. The computations are based on the experimental velocity distribution over the circular arc shown in Fig. 1a. The experimental data were presented by Alber, et al. (Ref. 1). The agreement with experiment for both the friction velocity and the displacement thickness is quite good up to the strong pressure gradient near the shock. Then the accuracy becomes poor, especially after separation.

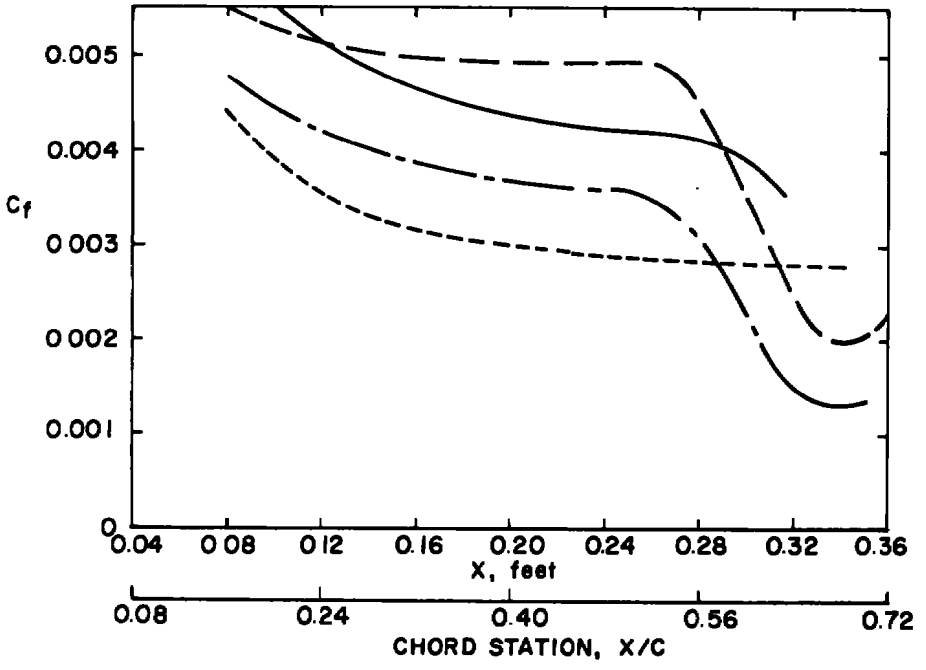


a. Pressure coefficient on the airfoil

Figure 2. Comparison of results from four boundary-layer methods using the pressure distribution on a C-141 airfoil.

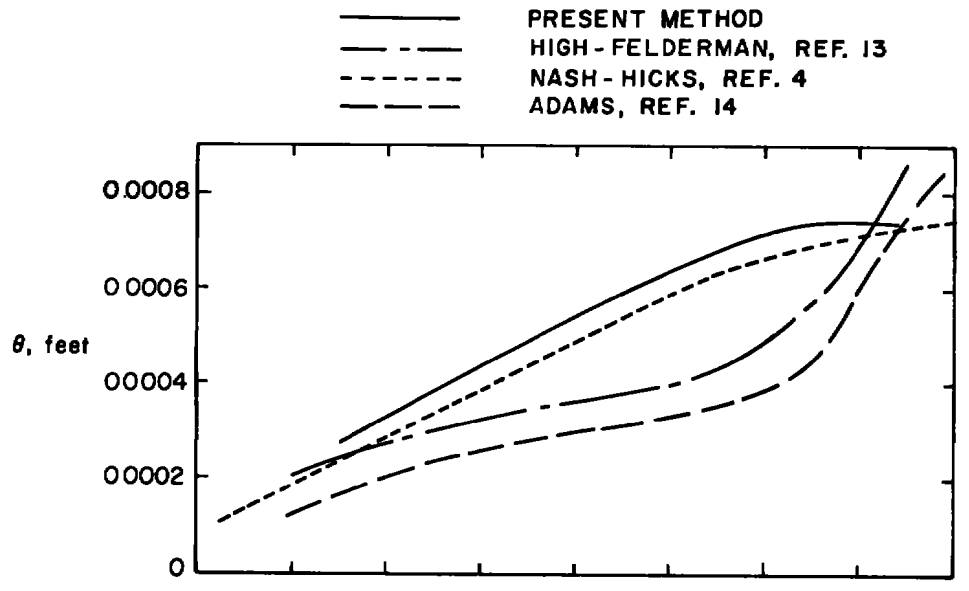


b. Shape factor

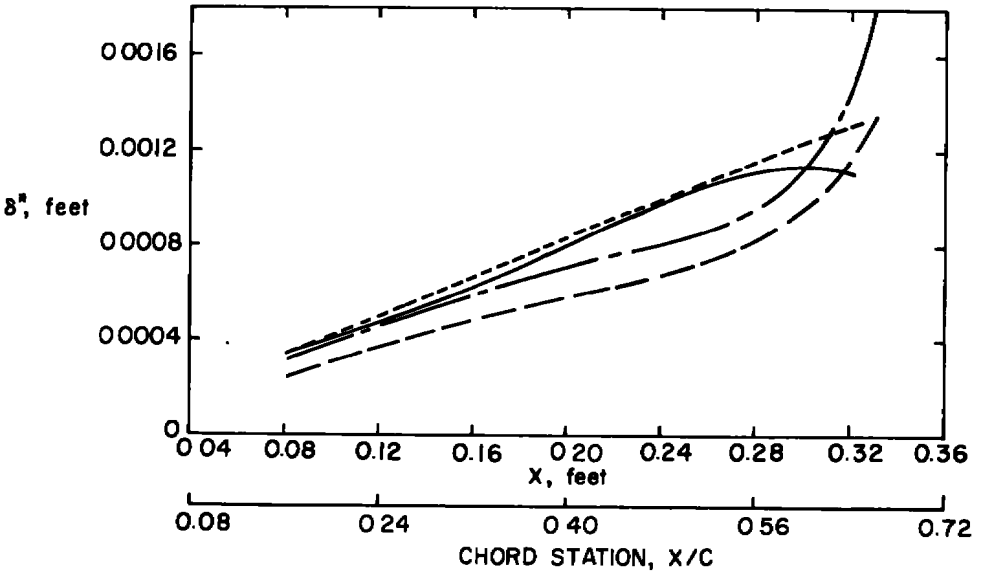


c. Skin-friction coefficient

Figure 2. Continued.



d. Momentum thickness



e. Displacement thickness

Figure 2. Concluded.

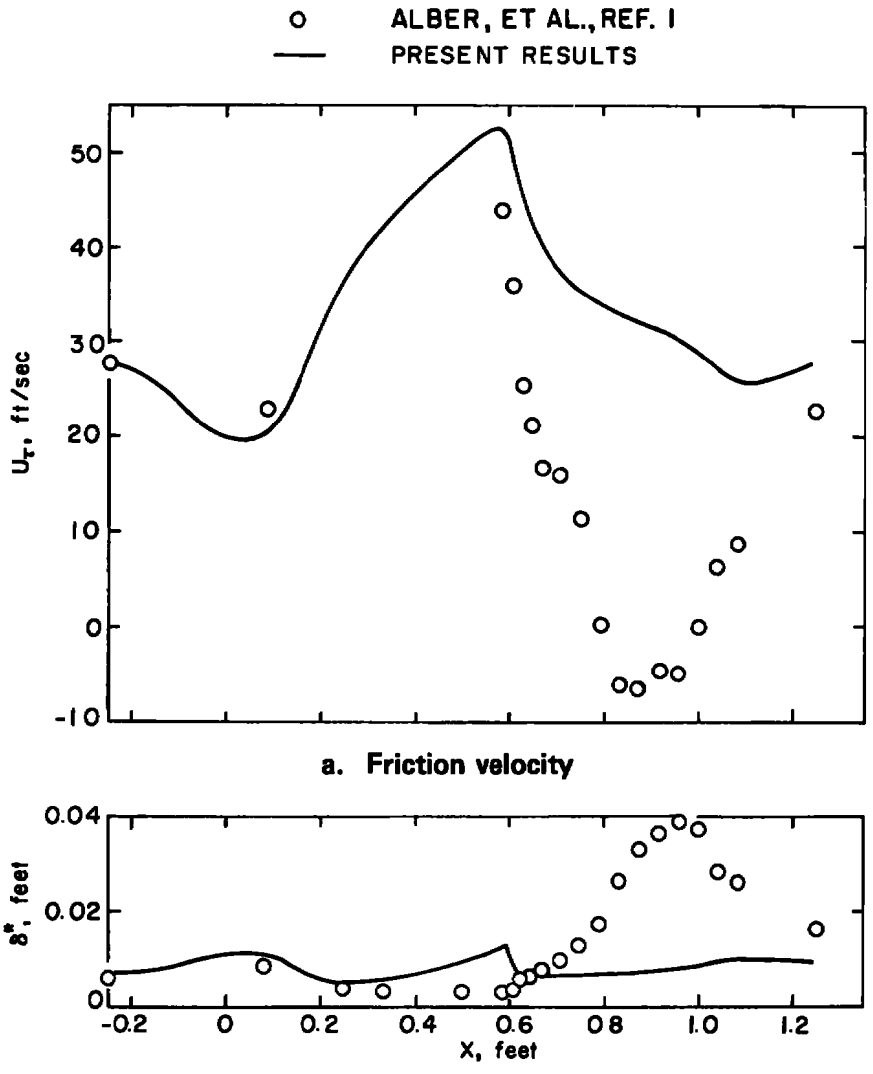


Figure 3. Boundary-layer characteristics computed from the experimental velocity distribution on the circular arc bump.

3.0 PROCEDURE IN A SEPARATED FLOW REGION

Three problems are most evident in boundary-layer computations involving a shock and separation. The first is the strong pressure gradients in the vicinity of the shock. The boundary-layer equations are not developed for use in strong pressure gradients and the assumptions made in the derivations are not very appropriate. The second problem is prediction of the point of separation. Boundary-layer methods typically do not predict

separation accurately, as the equations are singular at that point (Ref. 16), and while some prediction schemes such as Alber's (Ref. 9) and Stratford's (Ref. 17) are better, the accuracy in the transonic range is still inadequate. Finally, computing boundary-layer characteristics in a separated region with velocity specified normally results in very poor accuracy. This third problem is treated here by specifying either the friction velocity or the boundary-layer thickness in Eqs. (22) and (23) and solving for external velocity as a dependent variable.

3.1 METHOD OF COMPUTATION

Since Eqs. (22) and (23) are in terms of three unknowns, external velocity, boundary-layer thickness, and friction velocity, one of these variables must be prescribed. Usually the external velocity, either measured experimentally or from an inviscid solution, is specified and δ and u_τ are calculated. Near separation, this results in large inaccuracies. Typically, the skin-friction coefficient (or friction velocity) decreases in value but does not reach zero (Ref. 16). Similarly, other boundary-layer parameters fail to reach accurate values. Figure 3 is an example of this behavior as exhibited by the present method.

The accuracy of the results can be improved by specifying friction velocity or boundary-layer thickness and solving for external velocity as a dependent variable. In order to produce the desired experimental or calculated (inviscid) external velocity, a trial and error procedure is necessary. The specified friction velocity or boundary-layer thickness is varied until the resulting external velocity matches that desired.

Accuracy is improved both by forcing the boundary-layer velocity profiles to assume a more correct shape and by avoiding the singular behavior which occurs as skin friction approaches zero when external velocity is specified.

3.2 RESULTS

Figure 4 illustrates a case in which the friction velocity is specified. The body is the circular arc profile shown in Fig. 1a. Figure 4a gives the specified u_τ as compared with experimental values. This u_τ distribution is close to the experimentally determined value, and the computed external velocity is also near the experimental value (Fig. 4b). From Fig. 4c it can be seen that the displacement thickness is determined with good accuracy.

Figure 5 illustrates the same process with boundary-layer thickness specified. Figure 5a is the prescribed boundary-layer thickness, Fig. 5b is the resulting external velocity, and Fig. 5c compares the computed displacement thickness with experimental values. Again, the agreement is quite close.

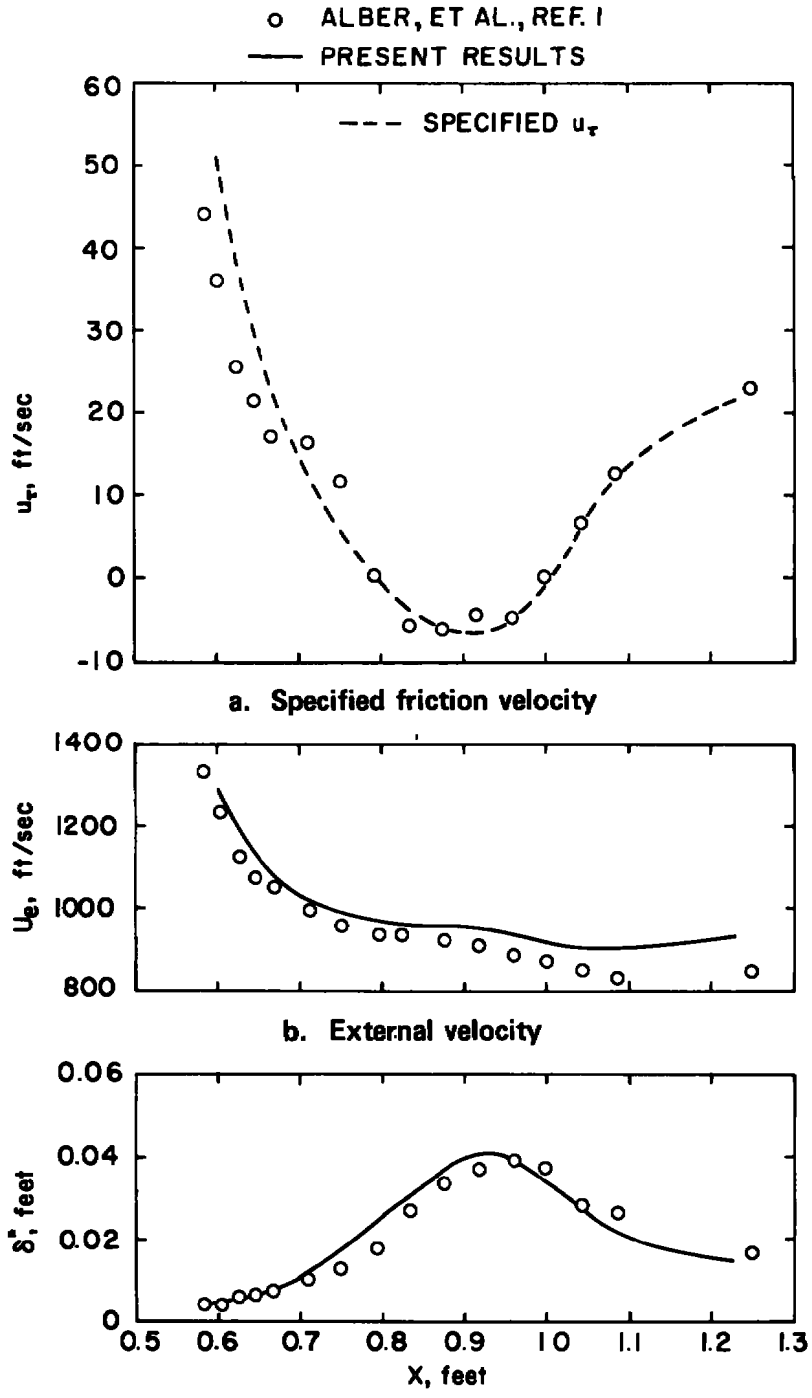


Figure 4. Boundary-layer characteristics computed for the circular arc bump with friction velocity specified.

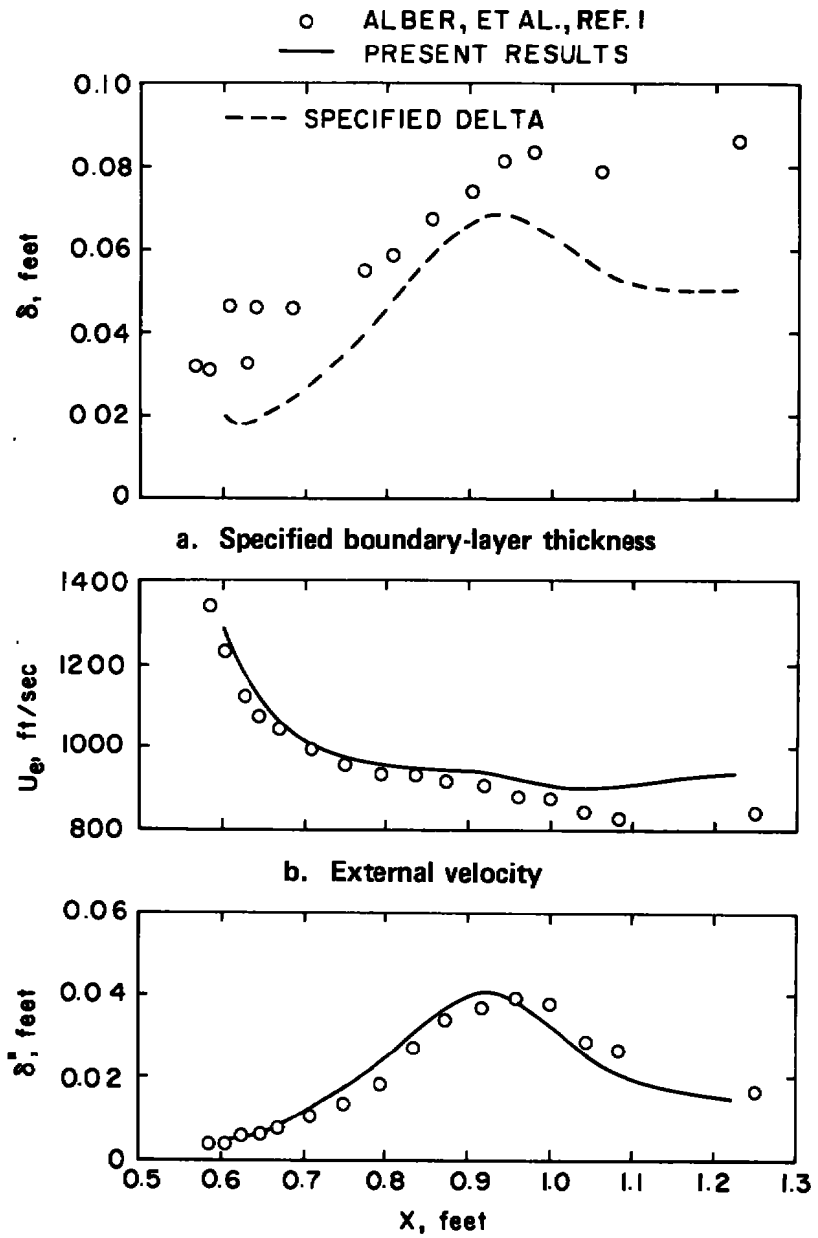
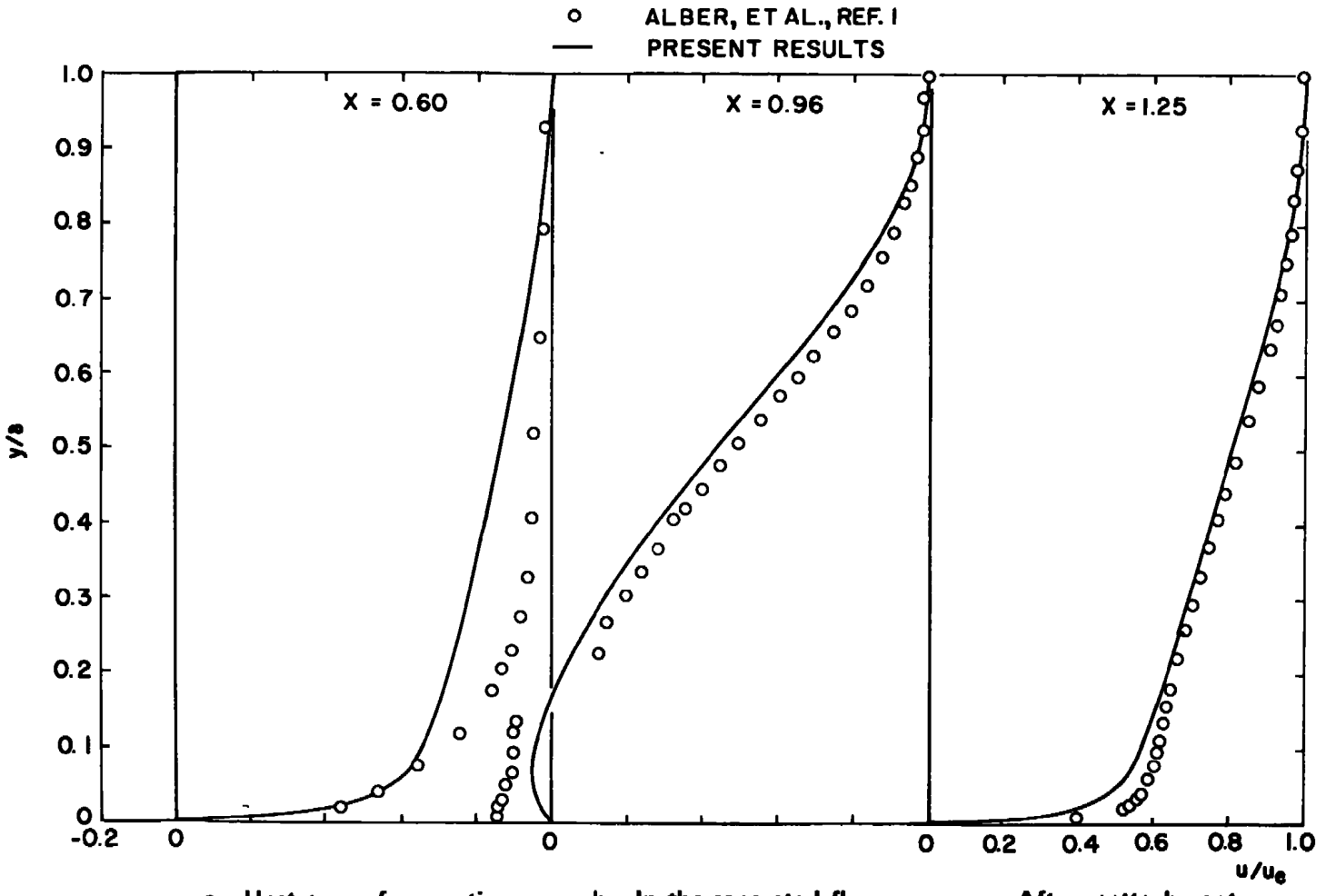


Figure 5. Boundary-layer characteristics computed for the circular arc bump with boundary-layer thickness specified.

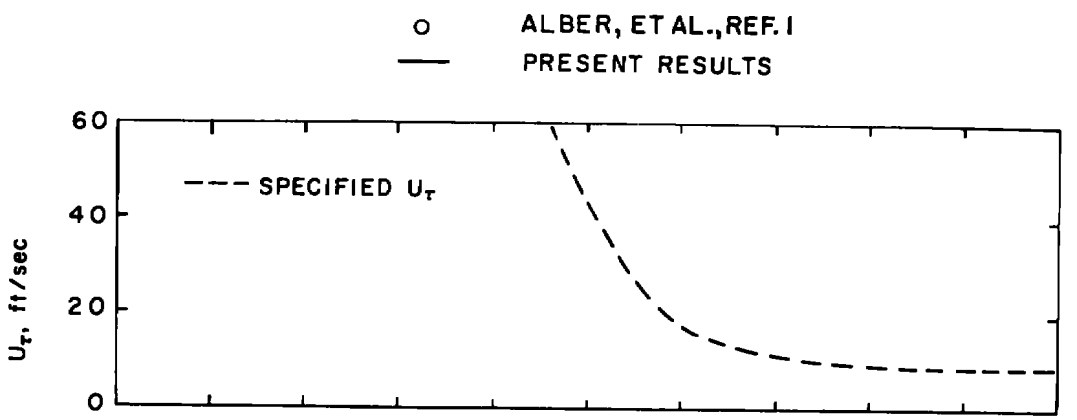
Figure 6 compares computed and experimental boundary-layer velocity profiles for the flow when boundary-layer thickness is specified. Figure 6a is a profile upstream of separation, Fig. 6b is in the separated flow, and Fig. 6c is after reattachment. Similar accuracy is obtained when friction velocity is specified.



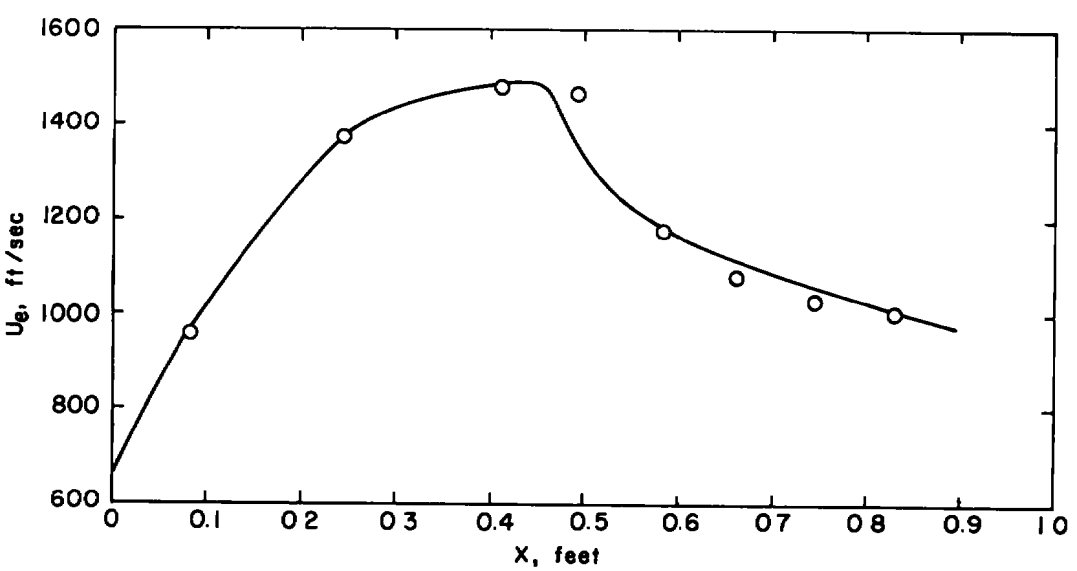
a. Upstream of separation b. In the separated flow c. After reattachment

Figure 6. Computed and experimental boundary-layer velocity profiles for the circular arc bump.

Figure 7. however. shows a difficulty that can arise while using this method. The computations for this figure are based on the experimental velocity distribution shown in Fig. 1b. Figures 7a and b demonstrate a specified friction velocity and the computed external velocity. Similarly. Figs. 7c and d give a specified boundary-layer thickness and the associated velocity. Both calculated velocities agree with experimental data; however, Fig. 7e shows the discrepancy between the displacement thickness resulting from these two computations. Even though the practical difference between these displacement thicknesses is not large. Fig. 7e indicates an ambiguity that can occur in these solutions.

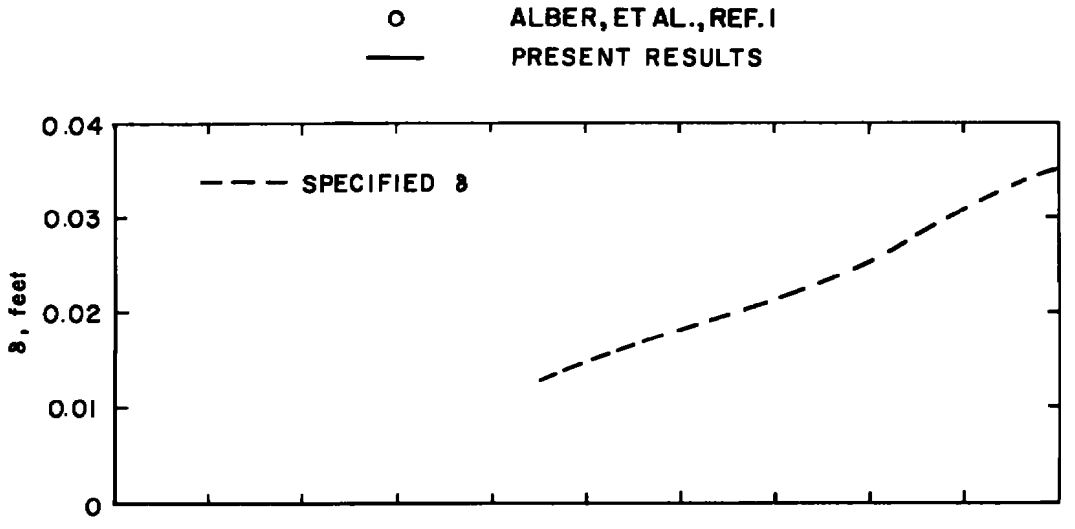


a. Specified friction velocity

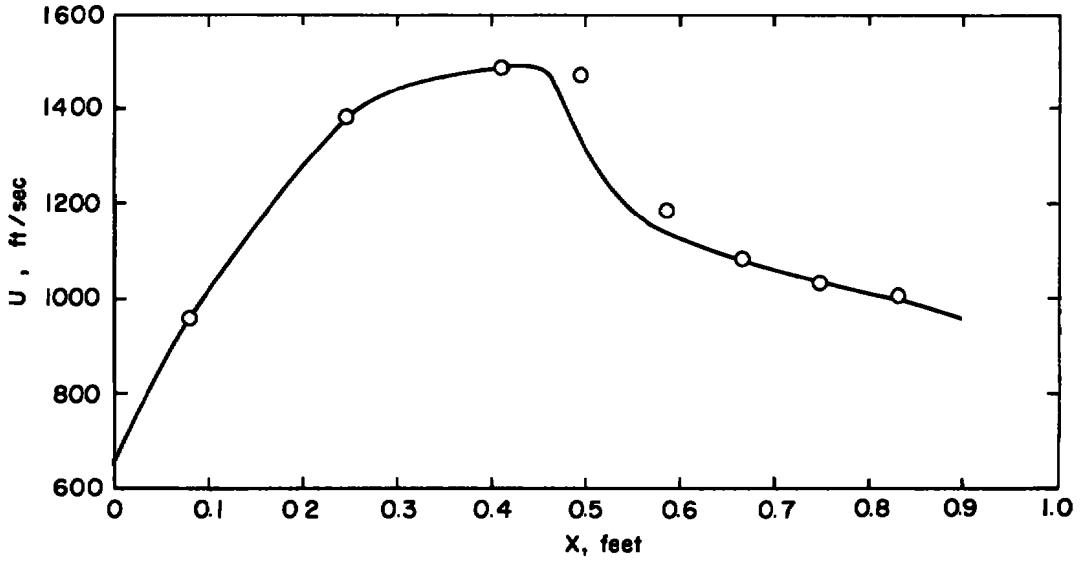


b. Resulting velocity profile

Figure 7. Calculation of boundary-layer characteristics for a flow near separation over an airfoil-shaped bump.

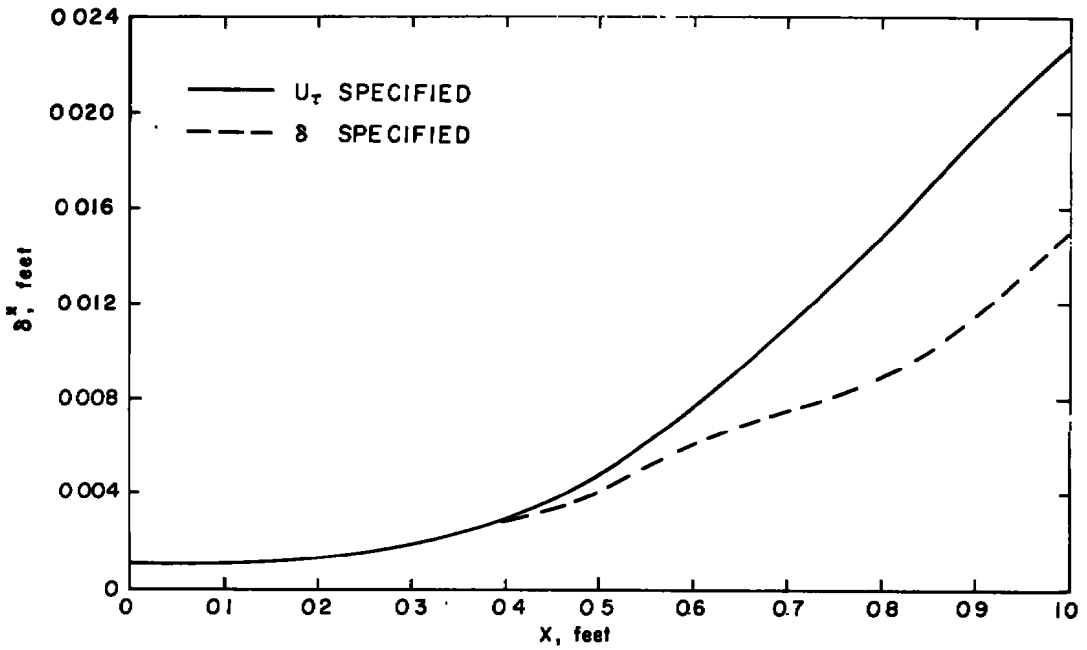


c. Specified boundary-layer thickness



d. Resulting velocity profile

Figure 7. Continued.



e. Comparison of displacement thicknesses computed by this method
Figure 7. Concluded.

4.0 CONCLUSIONS

The purpose of this work was to develop a computer code for a compressible, turbulent, boundary-layer method which could be used for computations in a region of separated flow.

The boundary-layer method presented here is fast and relatively accurate for attached flows in moderate pressure gradients. As with most available techniques. the accuracy is poor in very strong pressure gradients, such as those in the vicinity of a shock.

Following Kuhn and Nielsen, the solutions can be extended to separated flow regions by specification of friction velocity as the independent variable in the boundary-layer equations and solving for external velocity as a dependent variable. In the present program, the boundary-layer thickness can also be specified instead of friction velocity. When either friction velocity or boundary-layer thickness is specified so that the resulting external velocity matches an experimental or calculated inviscid external velocity, accurate displacement thicknesses can be obtained.

REFERENCES

1. Alber, I. E., Bacon, J. W., Masson, B. S., and Collins, D. J. "An Experimental Investigation of Turbulent Transonic Viscous-Inviscid Interactions." AIAA Journal, Vol. 11, No. 5, May 1973, pp. 620-627.

2. Murman, E. M. and Cole, J. D. "Calculation of Plane Steady Transonic Flows." AIAA Journal, Vol. 9, No. 1, January 1971, pp. 114-121.

3. Kuhn, G. D. and Nielsen, J. N. "Prediction of Turbulent Separated Boundary Layers." AIAA Paper No. 73-663, AIAA 6th Fluid and Plasma Dynamics Conference, Palm Springs, California, July 1973.

4. Nash, J. F. and Hicks, J. G. "An Integral Method Including the Effect of Upstream History on the Turbulent Shear Stress." Proceedings 1968 AFOSR-IPF-Stanford Conference, Stanford University, Vol. 1, 1969, pp. 37-45.

5. Coles, D. "The Law of the Wake in the Turbulent Boundary Layer." Journal of Fluid Mechanics, Vol. 1, Part 2, July 1956, pp. 191-226.

6. Stewartson, K. "Correlated Incompressible and Compressible Boundary Layers." Proceedings of the Royal Society, A, Vol. 200, 1949, pp. 85-100.

7. Spalding, D. B. "A Single Formula for the "Law of the Wall." Journal of Applied Mechanics, Vol. 28, Sept. 1961, pp. 455-458.

8. Klebanoff, P. S. "Characteristics of Turbulence in a Boundary Layer with Zero Pressure Gradient." NACA Report 1247, 1955.

9. Alber, I. E. "Similarity Solutions for a Family of Separated Turbulent Boundary Layers." AIAA Paper No. 71-203, 9th Aerospace Sciences Meeting, New York, N. Y., January 1971.

10. Handbook of Mathematical Functions. Edited by M. Abramowitz and I. A. Stegun, U. S. Government Printing Office, June 1964, p. 897.

11. Lo, C. F. and Carleton, W. E. "Transonic Scaling Effect on a Quasi, Two-Dimensional C-141 Airfoil Model." AEDC-TR-73-61 (AD762285), June 1973.

12. Patankar, S. V. and Spalding, D. B. Heat and Mass Transfer in Boundary Layers, CRC Press, Cleveland, Ohio, 1968.

13. High, M. D. and Felderman, E. J. "Turbulent MHD Boundary Layers with Electron Thermal Nonequilibrium and Finite Rate Ionization." AIAA Journal, Vol. 10, No. 1, January 1972, pp. 98-103.

14. Adams, J. C., Jr. "Numerical Calculation of the Subsonic and Transonic Turbulent Boundary Layer on an Infinite Yawed Airfoil." AEDC-TR-73-112 (AD763730), July 1973.

15. "Summary of Predicted Results." Proceedings, 1968 AFOSR-IPF-Stanford Conference, Stanford University, Vol. 1, 1969, pp. 533-569.

16. Gerhart, P. M. and Bober, L. J. "Comparison of Several Methods for Predicting Separation in a Compressible Turbulent Boundary Layer." NASA TMX-3102, August 1974.

17. Stratford, B. S. "The Prediction of Separation of the Turbulent Boundary Layer." Journal of Fluid Mechanics, Vol. 5, Pt. 1, January 1959, pp. 1-16.

APPENDIX A
DERIVATION OF DIFFERENTIAL EQUATIONS

Using Eqs. (12) and (13) one obtains:

$$\int_0^{\delta} \left\{ UU_x - U_y \int_0^{y} U_x \, d\eta - (S + 1)\, U_e\, U_{e_x} - \nu\, (\beta U_y)_y \right\} y^n dy = 0 \qquad \text{(A-1)}$$

$$n = 0,\ 1$$

and

$$U = u_\tau \left[2.5\ \ell n\ (1 + y^+) + 5.1 - (3.39\ y^+ + 5.1)\ e^{-0.37y^+}\right] + \frac{U_\beta}{2}\left(1 - \cos\frac{\pi y}{\delta}\right) \qquad \text{(A-2)}$$

where

$$u_\tau = \frac{\tau_w}{|\tau_w|}\left(\frac{|\tau_w|}{\rho}\right)^{1/2} \qquad \text{(A-3)}$$

$$u_\beta = U_e - u_\tau\ (2.5\ \ell n\ (1 + \delta^+) + 5.1) \qquad \text{(A-4)}$$

$$y^+ = \frac{|u_\tau|\, y}{\nu} \qquad \text{(A-5)}$$

$$\delta^+ = \frac{|u_\tau|\, \delta}{\nu} \qquad \text{(A-6)}$$

Then let

$$F_1(y) = 2.5\ \ell n\ (1 + y^+) + 5.1 - (3.39\ y^+ + 5.1)\ e^{-0.37y^+} \qquad \text{(A-7)}$$

$$F_2(y) = \frac{2.5}{1 + y^+} + (1.254\ y^+ - 1.502)\ e^{-0.37y^+} \qquad \text{(A-8)}$$

$$F_3(y) = \frac{1}{2}\left(1 - \cos\left(\frac{\pi y}{\delta}\right)\right) \qquad \text{(A-9)}$$

$$F_4(y) = \frac{\pi}{2}\sin\frac{\pi y}{\delta} \qquad \text{(A-10)}$$

so that Eq. (A-2) becomes

$$U = u_\tau F_1(y) + u_\beta F_3(y) \tag{A-11}$$

Taking the derivatives of Eq. (A-11) with the additional definition

$$F_5(y) = F_1(y) + y^+ F_2(y) \tag{A-12}$$

$$U_x = u'_\tau (F_5(y) - F_3(y) F_5(\delta)) - \delta' \left(\frac{y u_\beta}{\delta^2} F_4(y) + u_\tau F_3(y) F_5(y)\right) + U'_e F_3(y) \tag{A-13}$$

$$U_y = u^+_\tau F_2(y) + \frac{u_\beta}{\delta} F_4(y) \tag{A-14}$$

Defining

$$F_6 = F_5(y) - F_5(\delta)F_3(y) \tag{A-15}$$

$$F_7 = - \frac{y u_\beta}{\delta^2} F_4(y) - u^+_\tau F_2(\delta)F_3(y) \tag{A-16}$$

Equation (A-1) can then be written

$$\int_0^\delta \Big\{U[u'_\tau F_6 + U'_e F_3(y) + \delta' F_7] + U_y \int_0^y [u'_\tau + U'_e F_3(y) + \delta' F_7]\, d\eta - (s + 1)U_e U'_e - \nu (\beta U_y)_y \Big\} y^n dy = 0 \tag{A-17}$$

Then the coefficients for Eqs. (19) and (20) become

$$A_{k1} = \int_0^\delta \left(UF_6 + U_y \int_0^y F_6\, d\eta\right) y^{(k-1)}\, dy \tag{A-18}$$

$$A_{k2} = \int_0^\delta \left(UF_7 + U_y \int_0^y F_7\, d\eta\right) y^{(k-1)}\, dy \tag{A-19}$$

$$A_{k3} = \int_0^\delta \left(UF_3(y) + U_y \int_0^y F_3(y) d\eta - (s + 1)\, U_e U'_e\right) y^{(k-1)}\, dy \tag{A-20}$$

These expressions were evaluated numerically.

APPENDIX B
FORTRAN IV PROGRAM

B-1 DESCRIPTION OF PROGRAM FUNCTIONS

Given initial values and the external velocity, this program computes characteristics of a turbulent, compressible boundary layer. For a given region in the flow, either friction velocity or boundary-layer thickness can be specified and external velocity computed as a dependent variable.

B-2 INPUT

All input except specification of friction velocity or boundary-layer thickness is read in Subroutine Input. The specified variables are read in Subroutine Loop.

Card	Variables	Format	Description
1	M, N	(3I5)	M is the number of streamwise grid points, corresponding to the number of external velocities input. N is the number of points taken across the boundary layer. N = 21 is recommended.
2	LS,LD,LT	(3I5)	LS is the first streamwise grid point at which friction velocity is specified. LD is the last point specified. If LS = LD then boundary-layer thickness is specified and LT is the last point. If LS is greater than M neither variable is specified.
3	C,PI,RHOI, UI,DLC(1) UM,XO, UTAU(1)	(8F10.0)	C is the length of the computed area in feet. C/(M-1) is the streamwise computational interval. PI, RHOI and UI are the reference free-stream pressure, density and velocity in lb/sq ft, slugs/cu ft and ft/sec. DLC(1) is the initial boundary-layer thickness

Card	Variables	Format	Description
3 (Continued)			in feet. UM is the reference absolute viscosity in lb sec/sq ft. XO is the streamwise location of the starting point. UTAU(1) is the initial value of the friction velocity, $\sqrt{\tau_w/\rho}$, in ft/sec.
4-K	UEC(I)	(7F10.0)	External velocity in ft/sec. Must be given for each X station, but dummy values may be used if UTAU or DEL is specified at that location.
(K+1)-END	UTAU(I) or DEL(I)	(7F10.0)	Either friction velocity (ft/sec) or incompressible boundary-layer thickness (ft) as determined by card 2. Input values for alternate points; the program will interpolate.

B-3 OUTPUT

Output is in the same units as input. Print interval is controlled for variables K4 and K6 in Input. On the first page are starting conditions and initial profiles, in compressible and incompressible form.

The next block gives results at the designated streamwise locations. The following variables are printed:

N	Number of the station
XC	Compressible (physical) streamwise location
UC	Compressible external velocity input, or computed if UTAU or DEL is specified
DU/DXC	Derivative of the input external velocity, does not correspond to UC in regions where UTAU or DEL is specified
DELC	Compressible boundary-layer thickness

DEL*	Compressible displacement thickness
TH	Compressible momentum thickness
H	Shape factor
CF	Skin-friction coefficient
NU	Kinematic viscosity
X	Transformed incompressible streamwise location
UEI	Incompressible external velocity, computed value if UTAU or DELI is specified
DU/DXI	Incompressible velocity derivative, also computed if UTAU or DELI is specified
DELI	Incompressible boundary-layer thickness, this is one of the variables that can be specified
UTAU	Friction velocity - $\frac{\tau_w}{\lvert\tau_w\rvert}\ \frac{\tau_w}{\rho}$, this is the other variable that can be specified.
UBET	Wake velocity from Eq. (16)
TW	Wall temperature

The last print block gives compressible velocity profiles in the boundary layer. The profiles are given at streamwise locations corresponding to those having the same number in the preceding print block. Values printed are U/U_e in thousandths, with the decimal and leading zeros omitted. The spacing across the boundary layer is constant in compressible coordinates.

```
C   THIS PROGRAM COMPUTES A TURBULENT BOUNDARY LAYER BY THE METHOD
C   GIVEN BY KUHN AND NIELSEN.  INSTEAD OF SPECIFYING THE VELOCITY
C   PROFILE, UTAU(FRICTION VELOCITY) OR DELTA(BOUNDARY LAYER THICKNESS)
C   CAN BE SPECIFIED IN DESIGNATED REGIONS.  INPUT IS IN FEET, SLUGS
C   AND SECONDS OR A SIMILAR COMPATIBLE SYSTEM.
C   INPUT VARIABLES:  M IS THE NUMBER OF X VELOCITIES INPUT.  THESE
C   MUST BE SPACED AT EVEN INTERVALS.  N IS THE NUMBER OF GRID POINTS
C   IN THE Y DIRECTION.  FORMAT (2I5).  ENTER LS,LT,LD (3I5).  LS IS
C   THE X GRID POINT AT WHICH UTAU IS FIRST SPECIFIED.  LT IS THE LAST
C   GRID POINT FOR SPECIFYING UTAU.  IF LS=LT THEN DELTA IS SPECIFIED
C   AND LD IS THE LAST GRID POINT.  NEXT CARD:  CHORD LENGTH, FREE
C   STREAM PRESSURE,DENSITY AND VELOCITY, INITIAL BOUNDARY LAYER
C   THICKNESS, ABSOLUTE VISCOSITY, INITIAL X STATION AND INITIAL UTAU
C   (FRICTION VELOCITY).  FORMAT(8F10.0).  INPUT VELOCITY AT EACH X
C   STATION.  FORMAT(7F10.0).  ENTER (STARTING NEXT CARD) UTAU OR DELTA
C   IF APPLICABLE.  ENTER EVERY OTHER POINT--PROGRAM WILL INTERPOLATE.
C   FORMAT(7F10.0).

C   MAIN PROGRAM

      COMMON YT(101),Y(101),UC(101),U(101),RHO(101),XT(201),X(201),
     1 UEC(201),UE(201),UXC(201),UX(201),ULC(201),UEL(201),UTAU(201),
     2 UTP(201),K,K1,K4,K6,K7,UI,PI,AI,RHOI,PE,AE,RHOE,N,M,C,DX,DY,TH,
     3 DELD,TW,UBET,GNU,E(4),UM,RR,TT,TI,TE,H,TS
C   ESTABLISH CONSTANTS
      RO(ZC,ZE)=1+.178*(ZE/AE)**2*(1-(ZC/ZE)**2)
      E(1)=1
      E(2)=2
      E(3)=2
      E(4)=1
      RR=.00058278
      JC=0
      K=1
      K1=0
      Y(1)=0
      U(1)=0
      YT(1)=0.
      UC(1)=0.
      J=1
C   CALL INPUT AND ESTABLISH INITIAL VALUES
      CALL INPUT
      GAM=1.4
      GAM1=GAM-1
      TI=RR*PI/RHOI
      AI=SQRT(1.4*PI/RHOI)
      CALL EXT(1)
      RHOS=(RHOI**GAM+GAM1*(RHOI**GAM/PI)*(UI**2)/(2.*GAM))**(1/GAM1)
      AS=SQRT(.2*(UI**2)+AI**2)
      PS=RHOS*AS**2/GAM
      TS=RR*PS/RHOS
      AW=SQRT(GAM*PE/RHO(1))
      UE(1)=UEC(1)*AI/AE
      UBET=UE(J)-UTAU(J)*F1(UEL(J)*UTAU(J)/GNU)
C   ITERATE FOR INITIAL VELOCITY PROFILE
      DO 11 I=2,N
      YT(I)=YT(I-1)+DY
      Y(I)=Y(I-1)+DY
      U(I)=UTAU(J)*F1(ABS(UTAU(J))*Y(I)/GNU)+.5*UBET*(1-COS(3.14159*
     1   Y(I)/DEL(J)))
      UC(I)=U(I)*AE/AI
   11 RHO(I)=RHOE/RO(UC(I),UEC(1))
      NC=N/5
   14 UTES=UC(NC)
```

```
      DO 12 I=2,N
   12 Y(I)=Y(I-1)+(RHO(I)+RHO(I-1))*AE*DY/(RHOI*AI*2)
      DEL(1)=Y(N)
      UBET=UE(1)-UTAU(1)*FI(DEL(1)*UTAU(1)/GNU)
      DO 13 I=2,N
      U(I)=UTAU(J)*FI(ABS(UTAU(J))*Y(I)/GNU)+.5*UBET*(1-COS(3.14159*
     1   Y(I)/DEL(J)))
      UC(I)=U(I)*AE/AI
   13 RHO(I)=RHOE/RO(UC(I),UEC(1))
      JC=JC+1
      IF (JC.GT.20) GO TO 15
      IF (ABS(UTES-UC(NC)).GT..01) GO TO 14
   15 DEL(1)=Y(N)
      UBET=UE(1)-UTAU(1)*FI(DEL(1)*UTAU(1)/GNU)
      WRITE(6,2001) JC,UBET,UTAU(1)
      DO 6 I=2,N
      RHO(I)=RHOE/RO(UC(I),UEC(1))
      Y(I)=Y(I-1)+(RHO(I)+RHO(I-1))*AE*DY/(RHOI*AI*2)
    6 U(I)=UTAU(J)*FI(ABS(UTAU(J))*Y(I)/GNU)+.5*UBET*(1-COS(3.14159*
     1   Y(I)/DEL(J)))
      CALL BL
      UX(1)=UXC(1)*AI*(1+.2*UEC(1)**2/AE**2)/AE
      UX(2)=UXC(2)*AI*(1+.2*UEC(2)**2/AE**2)/AE
C   PRINT INITIAL VALUES
      WRITE(6,2002) DLC(1),UEC(1),UXC(1),DFL(1),UE(1),UX(1)
      TK=0.
      WRITE(6,2008)
      WRITE(6,2003) UI,PI,AI,RHOI,TI,UEC(1),PE,AE,RHOE,TE,TK,PE,A*,RHO(1
     1  ),TT,TK,PS,AS,RHOS,TS
      WRITE(6,2004) DELD,TM,M,TW
      WRITE(6,2006)
      WRITE(6,2007)((Y(I),U(I),YT(I),UC(I),RHO(I)),I=1,N)
      WRITE(6,2005)
      CALL PRINT(-1,&3)
C   CALL EXECUTIVE SUBROUTINE
    3 CALL LOOP
 2001 FORMAT(//I6,' ITERATIONS   UBET=',F10.4,'   UTAU=',F10.5/)
 2002 FORMAT(/15X,'   DELTA        UE          UEX'/,' COMPRESSIBLE  ',F10.6,
     1  2F10.2,/' INCOMPRESSIBLE',F10.6,2F10.2)
 2003 FORMAT('              U           P           A          RHO          T'
     1    / ' INF ',3F10.2,F10.7,F10.2
     2    / ' EDGE',3F10.2,F10.7,F10.2
     3    / ' WALL',3F10.2,F10.7,F10.2
     4    / ' STAG',3F10.2,F10.7,F10.2 )
 2004 FORMAT(/'   DEL*        THETA          H          CF'/,4F10.6)
 2005 FORMAT('0  K    X C    U C   DU/DX C   DEL C    DEL*       TM       H
     1CF          NU         X      UE I   DU/DX I   DEL I    UTAU     UBET     TW'/)
 2006 FORMAT('       Y             U             YT            UC          RHO')
 2007 FORMAT(2(F10.6,F10.2),F10.7)
 2008 FORMAT(////)
      STOP
      END
```

```
      SUBROUTINE INPUT
C     INPUT READS ALL INPUT EXCEPT THE UTAU OR DELTA.  THE VELOCITY
C     DERIVATIVES AND X GRID ARE SET UP AND PRINT INTERVAL DECIDED.
      COMMON YT(101),Y(101),UC(101),U(101),RHO(101),XT(201),X(201),
     1 UEC(201),UE(201),UXC(201),UX(201),DLC(201),DEL(201),UTAU(201),
     2 UTP(201),K,K1,K4,K6,K7,UI,P1,A1,RHOI,PE,AE,RHOE,N,M,C,DX,DY,TH,
     3 DELO,I,UBET,GNU,E(4),UM,RR,TT,TI,TE,H
      COMMON /MMM/ LS,LD,LT
      READ (5,801) M,N
      READ(5,801)LS,L1,LD
      READ (5,802) C,P1,RHOI,UI,DLC(1),UM,XO,UTAU(1)
      READ(5,803)(UEC(I),I=1,M)
C     SETS PRINT INTERVAL--K4 FOR X AND K6 FOR Y
      K4=1
      K6=N/15
      DEL(1)=DLC(1)
      DY=DLC(1)/(N-1)
      DX=C/(M-1)
      XT(1)=XO
      XT(M)=XO+C
      X(1)=XO
      UXC(1)=(-1.5*UEC(1)+2.*UEC(2)-.5*UEC(3))/DX
      M1=M-1
      DO  1 J=2,M1
      XT(J)=XT(J-1)+DX
    1 UXC(J)=(UEC(J+1)-UEC(J-1))/(2.*DX)
      UXC(M)=(1.5*UEC(M)-2.*UEC(M-1)+.5*UEC(M-2))/DX
      UEC(M+1)=2.*UEC(M)-UEC(M-1)
  801 FORMAT(3I5)
  802 FORMAT(8F10.0)
  803 FORMAT(7F10.0)
      RETURN
      END
```

```
      SUBROUTINE LOOP
C     EXECUTIVE SUBROUTINE.  CALLS NADV(ADVANCE WITH VELOCITY SPECIFIED)
C     SADV(ADVANCE WITH UTAU SPECIFIED) AND DADV(ADVANCE WITH DELTA
C     SPECIFIED).  ALSO READS UTAU OR DELTA AND CALLS INTERPOLATION
C     SUBROUTINE.  EXIT IS FROM MAIN ON RETURN.
      COMMON YT(101),Y(101),UC(101),U(101),RHO(101),XT(201),X(201),
     1 UEC(201),UE(201),UXC(201),UX(201),DLC(201),DEL(201),UTAU(201),
     2 UTP(201),K,K1,K4,K6,K7,U1,P1,A1,RHOI,PE,AE,RHOE,N,M,C,DX,DY,TH,
     3 DELD,TW,UWEI,GNU,E(4),UM,RR,TI,TI,TE,H
      COMMON /MMM/ LS,LD,LT
    1 CALL NADV(&2)
    2 K=K+1
      K1=K1+1
      X(K)=X(K-1)+DX*PE*AE/P1/A1
      IF (K.GE.M )CALL PRINT(1,&3)
      IF (K1.GE.K4) CALL PRINT(-1,&4)
    4 IF (K.EQ.LS) GO TO 5
      CALL NADV(&2)
    5 IF(LS.EQ.LT)GO TO 50
C     UTAU INPUT
      READ(5,101)(UTAU(J),J=LS,LT,2)
      CALL LAGR4(UTAU,LT,LS)
      LT=LT-1
      LN=LS+1
      DO 18 I=LN,LT
   18 UTAU(I)=(UTAU(I+1)+2.*UTAU(I)+UTAU(I-1))/4.
      LT=LT-1
      CALL SADV
      WRITE(6,102)
      GO TO 1
C     DELTA INPUT
   50 READ(5,101)(DEL(J),J=LS,LD,2)
      CALL LAGR4(DEL,LD,LS)
      LD=LD-1
      DO 51 I=LS,LD
   51 DEL(I)=(DEL(I+1)+2.*DEL(I)+DEL(I-1))/4.
      LD=LD-1
      CALL DADV
      WRITE(6,102)
      GO TO 1
  101 FORMAT(/F10.0)
  102 FORMAT(' VELOCITY SPECIFIED')
    3 RETURN
      END
```

```
      SUBROUTINE DERIV(DELQ,UTAP,UEP,K9,II)
C     FORMS COEFFICIENTS FOR USE IN THE RUNGE-KUTTA SUBROUTINES.
      COMMON YT(101),Y(101),UC(101),U(101),RHO(101),XT(201),X(201),
     1 UEC(201),UE(201),UXC(201),UX(201),DLC(201),DEL(201),UTAU(201),
     2 UTP(201),K,K1,K4,K6,K7,UI,PI,AI,RHOI,PE,AE,RHOE,N,M,C,DX,DY,IH,
     3 DELO,TW,UBET,GNU,E(4),UM,RR,TT,TI,TE,H,TS
      COMMON /DD/ DLPR(201)
      B5=0.
      B6=0.
      B7=0.
      F3Y=0.
      F6=0.
      F7=0.
      A11=0.
      A12=0.
      A13=0.
      A21=0.
      A22=0.
      A23=0.
      S=TT/TS
      D=DELQ/(N-1)
      DLQ=DELQ*ABS(UTAP)/GNU
      F1D=F1(DLQ)
      B20=0.
      F2D=(2.5/(1+DLQ))-B20
      F5D=F1D+DLQ*F2D
      A13L=UEP*DELQ*S
      A23L=A13L*DELQ/2
      B14=      -UTAP*ABS(UTAP)
      B24=B14
      PIE=3.14159
      UBET=UEP-UTAP*F1D
      UTQ=UTAP*ABS(UTAP)/GNU
      DO 1 I=2,N
      Y(I)=Y(I-1)+D
      YLQ=Y(I)*ABS(UTAP)/GNU
      U(I)=UTAP*F1(YLQ)+.5*UBET*(1-COS(PIE*Y(I)/DELQ))
      AMU=3+(-1)**I
      IF(I.EQ.N) AMU=1.
      B5A=F6
      B6A=F7
      B7A=F3Y
      YLQ=Y(I)*ABS(UTAP)/GNU
      F1Y=F1(YLQ)
      B20=0
       IF(YLQ.LT.60) B20=-(1.254*YLQ-1.502)*EXP(-.37*YLQ)
      F2Y=(2.5/(1+YLQ))-B20
      F3Y=.5*(1-COS(PIE*Y(I)/DELQ))
      F4Y=PIE*SIN(PIE*Y(I)/DELQ)/2.
      F5Y=F1Y+YLQ*F2Y
      F6=F5Y-F3Y*F5D
      F7=-UBET*Y(I)*F4Y/DELQ**2-UTQ*F3Y*F2D
      UY=UTQ*F2Y+UBET*F4Y/DELQ
      B5=B5+(B5A+F6)*D/2
      B6=B6+(B6A+F7)*D/2
      B7=B7+(B7A+F3Y)*D/2
      A11=A11+(U(I)*F6-UY*B5)*AMU
      A21=A21+(U(I)*F6-UY*B5)*AMU  *Y(I)
      A12=A12+(U(I)*F7-UY*B6)*AMU
      A22=A22+(U(I)*F7-UY*B6)*AMU      *Y(I)
      A13=A13+(U(I)*F3Y-UY*B7)*AMU
      A23=A23+(U(I)*F3Y-UY*B7)*AMU  *Y(I)
      BET=     BETA(I,DELQ,UTAP,UEP,II)
```

```
      B24=B24-BET*DY*AMU*GND
    1 CONTINUE
      A11=A11*D/3
      A12=A12*D/3
      A21=A21*D/3
      A22=A22*D/3
      B24=B24*D/3
      A13=A13*D/3-A13L
      A23=A23*D/3-A23L
      A21=A21/DELQ
      A22=A22/DELQ
      A23=A23/DELQ
      B24=B24/DELQ
      IF (DY.LT.0) GO TO 2
      IF(K9.GT.2)  GO TO 3
      UXD=UX(K+1)-UX(K)
      IF (II.EQ.1) UXD=0
      UXD=UX(K)+UXD/E(II)
      B14=-A13*UXD+B14
      B24=B24-A23*UXD
      UEP=UXD
      DELQ=(A21*B14-B24*A11)/(A12*A21-A11*A22)
      UTAP=(A22*B14-B24*A12)/(A11*A22-A21*A12)
      RETURN
    2 UTD=UTP(K+1)-UTP(K)
      IF (II.EQ.1) UTD=0
      UTPP=UTP(K)+UTD/E(II)
      B14=B14-A11*UTPP
      B24=B24-A21*UTPP
      UTAP=UTPP
      DELQ=(B14*A23-B24*A13)/(A12*A23-A22*A13)
      UEP=(B14*A22-B24*A12)/(A13*A22-A23*A12)
      RETURN
    3 DLU=DLPR(K+1)-DLPR(K)
      IF(II.EQ.1) DLU=0.
      DLPP=DLPR(K)+DLU/E(II)
      B14=B14-A12*DLPP
      B24=B24-A22*DLPP
      DELQ=DLPP
      UTAP=(A23*B14-A13*B24)/(A11*A23-A13*A21)
      UEP=(A21*B14-A11*B24)/(A13*A21-A11*A23)
      RETURN
      END
```

```
      SUBROUTINE MADV(*)
C     RUNGA KUTTA ADVANCE WHEN EXTERNAL VELOCITY IS SPECIFIED.  CALLS
C     DERIV FOR COEFFICIENTS OF THE DERIVATIVES.
      COMMON YT(101),Y(101),UC(101),U(101),RHO(101),AT(201),X(201),
     1 UEC(201),UE(201),UXC(201),UX(201),DLC(201),DEL(201),UTAU(201),
     2 UTP(201),K,K1,K4,K6,K7,U1,PI,AI,RHOI,PE,AE,RHOE,N,M,C,DX,DY,TH,
     3 DELD,TW,UBET,UNU,E(4),UM,RR,II,JJ,IE,M
      DELP=0.
      UEP=0.
      UTAUP=0.
      DDI=0.
      UUI=0.
      D=DX*PE*AE/(PI*AI)
C     RUNGE-KUTTA ADVANCE
      DO 1 I=1,4
      DELP=DEL(K)+DELP*D/E(I)
      UTAUP=UTAU(K)+UTAUP*D/E(I)
      UEP=UE(K)+UEP*D/E(I)
      CALL DERIV(DELP,UTAUP,UEP,1,1)
      DDI=DDI+E(I)*DELP
    1 UUI=UUI+E(I)*UTAUP
      J=K+1
      DEL(J)=DEL(K)+DDI*D/6
      UTAU(J)=UTAU(K)+UUI*D/6
      CALL EAT(J)
      UE(J)=UEC(J)*AI/AE
      UX(J)=UXC(J)*AI*(1+.2*UEC(J)**2/AE**2)/AE
      IF(J.GE.M)GO TO 3
      L=J+1
    3 UX(L)=UXC(L)*AI*(1+.2*UEC(L)**2/AE**2)/AE
      RETURN 1
      END
```

```
      SUBROUTINE DADV
C     RUNGA-KUTTA ADVANCE WHEN BOUNDARY LAYER THICKNESS IS SPECIFIED.
      COMMON YT(101),Y(101),UC(101),U(101),RHO(101),XT(201),X(201),
     1 UEC(201),UE(201),UXC(201),UX(201),DLC(201),DEL(201),UTAU(201),
     2 UTP(201),K,K1,K4,K6,K7,UI,PI,AI,RHOI,PE,AE,RHOE,N,M,C,DX,DY,TH,
     3 DELO,TW,UBET,GNU,E(4),UM,RR,TT,TI,TE,H
      COMMON /MMM/ LS,LD,LT
      COMMON /DD/ DLPR(201)
      WRITE(6,901)
      J=K
      UBET=UE(J)-UTAU(J)*F1(ABS(UTAU(J)))*DEL(J)/GNU)
      CALL PRINT(-1,&8)
    8 DLPR(K)=(DEL(K+1)-DEL(K))/(DX*PE*AE/(PI*AI))
      DLPR(K+1)=(DEL(K+2)-DEL(K+1))/(DX*PE*AE/(PI*AI))
      DO 2 KK=LS,LD
      K=KK
      D=DX*PE*AE/(PI*AI)
      DELO=0.
      UEP=0.
      UTAUP=0.
      UTI=0.
      UUI=0.
      DO 5 I=1,4
      DELO=DEL(K)+DELO*D/E(I)
      UTAUP=UTAU(K)+UTAUP*D/E(I)
      UEP=UE(K)+UEP*D/E(I)
      CALL DERIV(DELO,UTAUP,UEP, 3,I)
      UTI=UTI+E(I)*UTAUP
    5 UUI=UUI+E(I)*UEP
      J=K+1
      UX(J)=UUI/6.
      UTAU(J)=UTAU(K)+UTI*D/6.
      UE(J)=UE(K)+UUI*D/6.
      CALL EXT(J)
      DLPR(J)=(DEL(J+1)-DEL(J))/(DX*PE*AE/(PI*AI))
      IF(J.GE.M)GO TO 3
      L=J+1
    3 DLPR(L)=(DEL(L+1)-DEL(L))/(DX*PE*AE/(PI*AI))
      UEC(J)=UE(J)*AE/AI
      K=J
      K1=K1+1
      X(K)=X(K-1)+DX*PE*AE/PI/AI
      IF (K1.GE.K4) CALL PRINT(-1,&2)
    2 CONTINUE
      RETURN
  901 FORMAT(//'  SEPARATED REGION   DELTA SPECIFIED'/)
      END
```

```
      SUBROUTINE SAUV
C     RUNGE-KUTTA ADVANCE WHEN FRICTION VELOCITY IS SPECIFIED.
      COMMON YT(101),Y(101),UC(101),U(101),RHO(101),XT(201),X(201),
     1 UEC(201),UE(201),UXC(201),UX(201),DLC(201),DEL(201),UTAU(201),
     2 UTP(201),K,K1,K4,K6,K7,UI,PI,AI,RHOI,PE,AE,RHOE,N,M,C,DX,DY,TH,
     3 DELD,TW,UBET,GNU,E(4),UM,RR,TT,TI,TE,H
      COMMON /MMM/ LS,LD,LT
      WRITE(6,901)
      J=K
      UBET=UE(J)-UTAU(J)*FI(ABS(UTAU(J))*DEL(J)/GNU)
      CALL PRINT(-1,&8)
    8 UTP(K)=(UTAU(K+1)-UTAU(K ))/(DX*PE*AE/(PI*AI))
      UTP(K+1)=(UTAU(K+2)-UTAU(K+1))/(DX*PE*AE/(PI*AI))
      DO 2 KK=LS,LT
      K=KK
      D=DX*PE*AE/(PI*AI)
      DELQ=0.
      UEP=0.
      UTAUP=0.
      DDI=0.
      UUI=0.
      DO 5 I=1,4
      DELQ=DEL(K)+DELW*D/E(I)
      UTAUP=UTAU(K)+UTAUP*D/E(I)
      UEP=UE(K)+UEP*D/E(I)
      CALL DERIV(DELQ,UTAUP,UEP,-1,1)
      DDI=DDI+E(I)*DELW
    5 UUI=UUI+E(I)*UEP
      J=K+1
      UX(J)=UUI/6.
      DEL(J)=DEL(K)+DDI*D/6.
      UE(J)=UE(K)+UUI*D/6.
      CALL FAI(J)
      UTP(J)=(UTAU(J+1)-UTAU(J))/(DX*PE*AE/(PI*AI))
      UEC(J)=UE(J)*AE/AI
      IF(J.GE.M)GO TO 3
      L=J+1
    3 UTP(L)=(UTAU(L+1)-UTAU(L))/(DX*PE*AE/(PI*AI))
      K=J
      K1=K1+1
      X(K)=X(K-1)+UX*PE*AE/PI/AI
      IF (K1.GE.K4) CALL PRINT(-1,&2)
    2 CONTINUE
      RETURN
  901 FORMAT(// ' SEPARATED REGION    UTAU SPECIFIED'/)
      END
```

```
      SUBROUTINE BL
C     COMPUTES DISPLACEMENT AND MOMENTUM THICKNESSES,SHAPE FACTOR AND
C     SKIN FRICTION
      COMMON YT(101),Y(101),UC(101),U(101),RHO(101),XT(201),X(201),
     1 UEC(201),UE(201),UXC(201),UX(201),DLC(201),DEL(201),UTAU(201),
     2 UTP(201),K,K1,K4,K6,K7,U1,PI,AI,RHOI,PE,AE,RHOE,N,M,C,DX,DY,TH,
     3 DELD,TW,UBET,GNU,E(4),UM,RH,TT,TI,TE,H
      DELD=DY
      TH=0.
      DO 7 I=2,N
      Z3=1.-(RHO(I)*UC(I))/(RHOE*UEC(K))
      Z1=Z3*(RHO(I)*UC(I))/(RHOE*UEC(K))
      Z3=Z3*(3+(-1)**I)
      IF (I.EQ.N) Z3=Z3/4
      Z1=Z1*(3+(-1)**I)
      IF (I.EQ.N) Z1=Z1/4
      DELD=DELD+Z3*DY
    7 TH=TH+Z1*DY
      DELD=DELD/3
      TH=TH/3
      H=DELD/TH
      TW=UTAU(K)*ABS(UTAU(K))*2./U1**2
      RETURN
      END
```

```
      FUNCTION BETA(J,DP,UP,UU,KK)
C     COMPUTES EDDY VISCOSITY
      COMMON YT(101),Y(101),UC(101),U(101),RHO(101),XT(201),X(201),
     1 UEC(201),UE(201),UXC(201),UX(201),DLC(201),DEL(201),UTAU(201),
     2 UIP(201),K,K1,K4,K5,K7,U1,P1,A1,RHO1,PE,AE,RHOE,N,M,C,DX,DY,IM,
     3 DELD,T4,UBET,GNU,E(4),U4,RR,T1,TI,TE,M
      IF (DP.LE.0)  GO TO 1
      IF (J.GT.2)  GO TO 2
      D=DP/(N-1)
      TH=0.
      DELD=1.
      DO 3 I=2,N
      TH=TH+(1.-U(I)/UU)*(3+(-1)**I)*(U(I)/UU)
    3 DELD=DELD+(1.-U(I)/UU)*(3+(-1)**I)
      TH=TH*D/3.
      DELD=DELD*D/3
      PK=.013
      TW=UP*ABS(UP)*RHOE
      PX=0
      IF(UX(K).LT.0.)
     1PX=-RHOE*UU*UX(K)
      P1=ABS(DELD*PX/(TW*15.))
      IF(P1.LT.25.)
     1PK=.013+.0038*EXP(-(DELD/TW)*PX/15)
      K7=-1
    2 IF(K7.GT.0) GO TO 4
      T1=U(J)/ABS(UP)
      IF(T1.GT.25.) GO TO 4
      T2=1+.0533*(EXP(.41*T1)-(1+.41*T1+.5*(.41*T1)**2))
      T3=PK*(1+5.5*(Y(J)/DP)**6)**(-1)*UU*DELD/GNU+1
      IF(T3.LE.T2) K7=1
      BETA=AMIN1(T2,T3)
      RETURN
    4 BETA=PK*(1+5.5*(Y(J)/DP)**6)**(-1)*UU*DELD/GNU +1
      RETURN
    1 IF (J.GT.2)  GO TO 6
      DO 5 I=2,N
      II=I
      IF(U(II).GE.0.) GO TO 7
    5 CONTINUE
    7 DELD=0
      D=DP/(N-1)
      DO 8 I=II,N
    8 DELD=DELD+((1-(U(I)/UU))+(1-(U(I-1)/UU)))
      DELD=DELD*.013*UU*D*.5/GNU
    6 BETA=DELD*(1+5.5*(Y(J)/DP)**6)**(-1)
      RETURN
      END
```

```
      SUBROUTINE PRINT(K2,*)
C     COMPUTES COMPRESSIBLE VELOCITY PROFILES AND PRINTS OUTPUT
      COMMON YT(101),Y(101),UC(101),U(101),RHO(101),XT(201),X(201),
     1 UEC(201),UE(201),UXC(201),UX(201),DLC(201),DEL(201),UTAU(201),
     2 UTP(201),K,K1,K4,K6,K7,UI,PI,AI,RHOI,PE,AE,RHOE,N,M,C,DX,DY,TH,
     3 DELD,TW,UBET,GNU,E(4),UM,RH,TT,TI,TE,H
      DIMENSION  L(101,201)
      DIMENSION KOU(24)
      RO(ZC,ZE)=1+.178*(ZE/AE)**2*(1-(ZC/ZE)**2)
      IF (K2) 1,1,3
    3 K1=100
    1 DY=DEL(K)/(N-1)
      UBET=UE(K)-UTAU(K)*F1(ABS(UTAU(K)*DEL(K)/GNU))
      RHO(1)=RHOE/RO(0,UEC(K))
      DO 4 I=2,N
      RHO(I)=RHOE/RO(UC(I),UEC(K))
    4 YT(I)=YT(I-1)+AI*RHOI*DY*(((1/RHO(I))+(1/RHO(I-1)))/AE*.5)
      DLC(K)=YT(N)
      DY=DLC(K)/(N-1)
      DO 6 I=2,N
      Y(I)=Y(I-1)+(RHO(I)+RHO(I-1))*AE*DY/(RHOI*AI*2)
      U(I)=UTAU(K)*F1(ABS(UTAU(K))*Y(I)/GNU)+.5*UBET*(1-COS(3.14159*
     1   Y(I)/DEL(K)))
    6 UC(I)=U(I)*AE/AI
      CALL BL
      DO 8 I=1,N,K6
    8 L(I,K)= U(I)/ UE(K)  *1000
      L(N,K)=999
      WRITE(6,2001)K,XT(K),UEC(K),UXC(K),DLC(K),DELD,TH,H,TW,GNU,X(K),
     1  UE(K),UX(K),DEL(K),UTAU(K),UBET,TT
      IF (K1.GT.50) GO TO 2
      K1=0
      RETURN 1
    2 CONTINUE
      II=0
      DO 14 I=1,N,K6
      II=II+1
   14 KOU(II)=I
      WRITE(6,2002)(KOU(I),I=1,N,K6),(II,(L(I,II),I=1,N,K6),II=1,M,K4)
 2001 FORMAT(14,F7.4,F6.0,F8.0,3F8.5,F5.2,2F9.6,F6.3,F6.0,F8.0,
     1  F8.5,F7.3,F7.1,F5.0)
 2002 FORMAT('1                              VELOCITY PROFILES'//,5X,21I5/210
     1  (2215/))
      RETURN 1
      END
```

```
      SUBROUTINE LAGR4(X,M,L)
C     FOURTH ORDER LAGRANGIAN INTERPOLATION
      DIMENSION  X(201)
      L=L+3
      N=M-3
      DO 1 J=L,N,2
    1 X(J)=-.0625*(X(J-3)+X(J+3))+.5625*(X(J-1)+X(J+1))
      L=L-2
      X(L)=(X(L+1)+X(L-1))/2.
      X(M-1)=(X(M)+X(M-2))/2.
      L=L-1
      RETURN
      END


      SUBROUTINE EXI(J)
C     COMPUTES EDGE CONDITIONS AND INCOMPRESSIBLE VELOCITY PROFILES
      COMMON YT(101),Y(101),UC(101),U(101),RHO(101),XT(201),X(201),
     1 UEC(201),UE(201),UXC(201),UX(201),DLC(201),DEL(201),UTAU(201),
     2 UTP(201),K,K1,K4,K6,K7,U1,P1,A1,RHO1,PE,AE,RHOE,N,M,C,DX,DY,IM,
     3 DELD,TW,UBET,GNU,E(4),UM,RR,TT,TI,TE,H
      RO(ZC,ZE)=1+.178*(ZE/AE)**2*(1-(ZC/ZE)**2)
      GAM=1.4
      GAM1=GAM-1
      AE=SQRT(.2*(U1**2-UEC(J)**2)+A1**2)
      RHOE=(RHO1**GAM1+GAM1*(RHO1**GAM/P1)*(U1**2-UEC(J)**2)/(2*GAM))**
     1 (1/GAM1)
      PE=RHOE*AE**2/1.4
      TE=RR*PE/RHOE
      GNU=UM*TE/TI/RHOE
      RHO(1)=RHOE/RO(0.,UEC(J))
      TT=PE*RR/RHO(1)
      RETURN
      END


      FUNCTION F1(Z)
C     LOGRITHMIC FUNCTION FOR VELOCITY
      F1    =2.5*ALOG(1+Z)+5.1
      IF(Z.GT.60.) GO TO 1
                   F1=F1      -(3.39*Z+5.1)*EXP(-.37*Z)
    1 RETURN
      END
```

NOMENCLATURE

a	Sound speed
A_{ij}	Coefficients (Eqs. 19 and 20)
C_f	Skin-friction coefficient
C_p	Pressure coefficient
F_i	Defined functions (Appendix B)
H	Shape factor
P	Pressure
S	Enthalpy parameter
T	Temperature
U,V	Incompressible velocities
u,v	Compressible velocities
u_β	Wake velocity
u_τ	Friction velocity
u^+	u/u_τ
X,Y	Incompressible flow coordinates
$\tilde{x}, \tilde{y}$	Compressible flow coordinates
y^+	$\lvert u_\tau \rvert y/\nu$
β	$1 + \epsilon/\nu$
δ	Boundary-layer thickness
δ^*	Displacement thickness
ϵ	Eddy viscosity
θ	Momentum thickness

ν	Kinematic viscosity
ρ	Density
τ_w	Shear stress at wall

SUBSCRIPTS

AW	Adiabatic wall
e	Edge of boundary layer
t	Stagnation
∞	Free stream