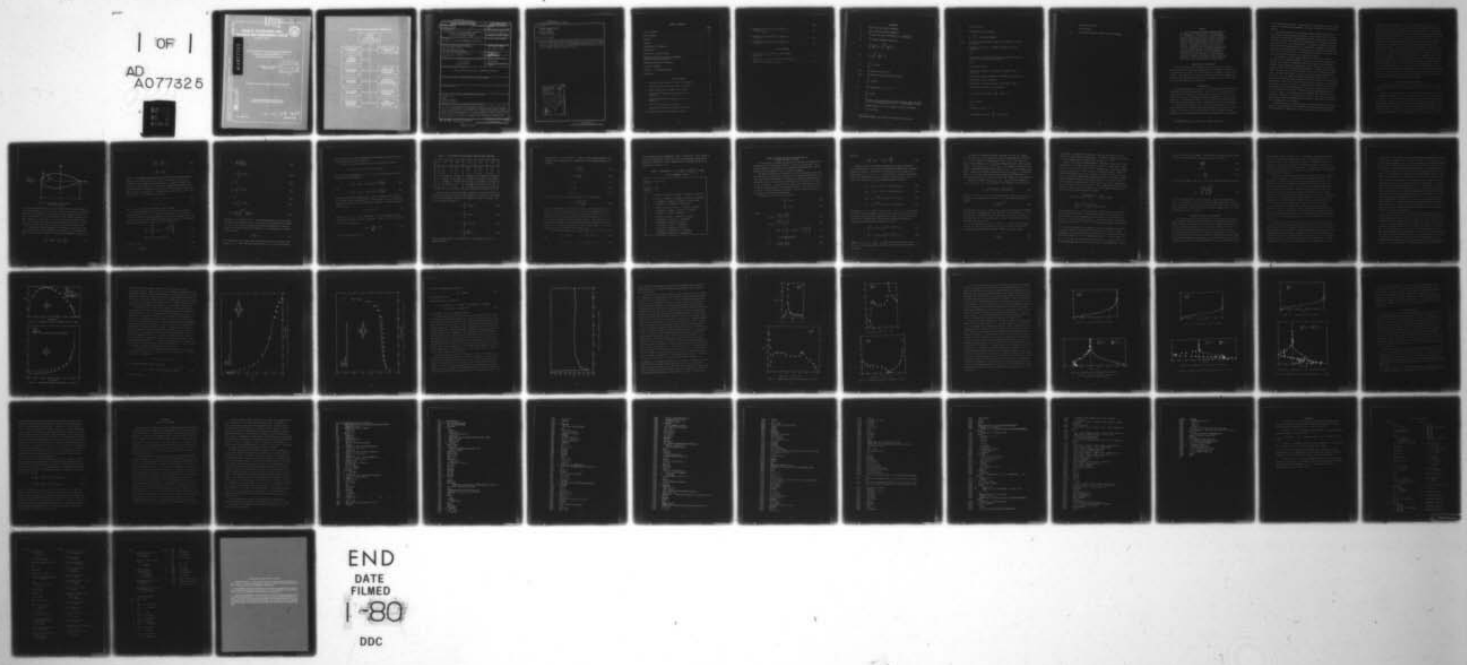
AD-A077 325 DAVID W TAYLOR NAVAL SHIP RESEARCH AND DEVELOPMENT CE--ETC F/G 20/4
AN EVALUATION OF THE TWO-PARAMETER MOMENTUM-INTEGRAL LAMINAR BO--ETC(U)
NOV 79 C H VON KERCZEK , N C GROVES

UNCLASSIFIED DTNSRDC-79/019 NL

| OF |
AD
A077325



END
DATE
FILMED
1-80
DDC

LEVEL II

DAVID W. TAYLOR NAVAL SHIP RESEARCH AND DEVELOPMENT CENTER

Bethesda, Md. 20084

AN EVALUATION OF THE TWO-PARAMETER MOMENTUM-INTEGRAL LAMINAR BOUNDARY LAYER CALCULATION METHOD OF HEAD

by

Christian H. von Kerczek
Nancy C. Groves

APPROVED FOR PUBLIC RELEASE: DISTRIBUTION UNLIMITED

SHIP PERFORMANCE DEPARTMENT
RESEARCH AND DEVELOPMENT REPORT

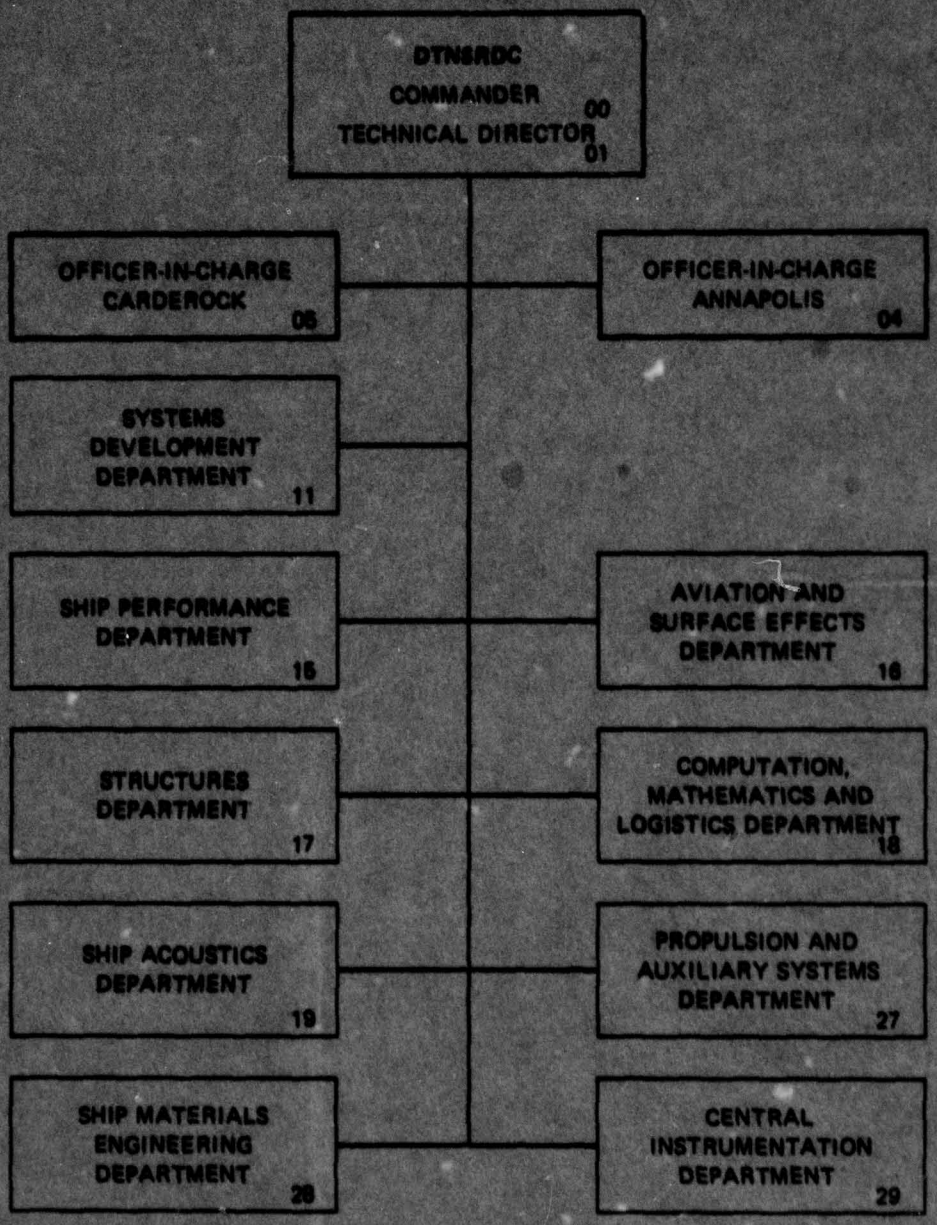
November 1979

DTNSRDC-79/019

DTNSRDC-79/019

AN EVALUATION OF THE TWO-PARAMETER MOMENTUM-INTEGRAL LAMINAR BOUNDARY LAYER CALCULATION METHOD OF HEAD

MAJOR DTNSRDC ORGANIZATIONAL COMPONENTS



UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER: DTNSRDC-79/019	2. GOVT ACCESSION NO.
3. RECIPIENT'S CATALOG NUMBER	
4. TITLE (and Subtitle): AN EVALUATION OF THE TWO-PARAMETER MOMENTUM-INTEGRAL LAMINAR BOUNDARY LAYER CALCULATION METHOD OF HEAD	5. TYPE OF REPORT & PERIOD COVERED: Final rept.
	6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s): Christian H. von Kerczek and Nancy C. Groves	8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS: David W. Taylor Naval Ship Research and Development Center, Bethesda, Maryland 20084	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS: (See reverse side)
11. CONTROLLING OFFICE NAME AND ADDRESS: Naval Sea Systems Command (SEA-037), Washington, D.C. 20362	12. REPORT DATE: November 1979
	13. NUMBER OF PAGES: 56
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)	15. SECURITY CLASS. (of this report): UNCLASSIFIED
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

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

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

Boundary Layer
Laminar
Momentum Integral
Numerical Methods

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

The two-parameter momentum-integral method of M.R. Head for the calculation of two-dimensional or axisymmetric, incompressible, laminar boundary layers is reworked into an analytical form, thereby eliminating the graphical data of Head's original method. The revised method is coded for digital computer solution and a number of test cases are computed. The numerical results for these test cases are compared with very accurate boundary layer

(Continued on reverse side)

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-014-6601

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

387682

(Block 10)

Program Element 25634N
Project S0218011
Task Area 20052
Work Unit 1942-087

(Block 20 continued)

solutions obtained by a finite-difference method and also results obtained using the Pohlhausen momentum-integral method. It was found that the Head method is vastly superior to the Pohlhausen method yet requires only about the same computational expense. Head's method yields very accurate results even for highly nonsimilar boundary layers.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

NOTATION*

a	Basic velocity profile parameter
b	Basic velocity profile parameter
C_f	The skin friction coefficient, $C_f = \tau_w / \left(\frac{1}{2} \rho U_0^2\right)$
C_p	The surface pressure coefficient
D	$\int_0^1 \left(\frac{\partial f}{\partial \eta}\right)^2 d\eta = \int_0^{\delta/\theta} \left(\frac{\partial f}{\partial \eta}\right)^2 d\eta$
D^*	$(\theta/\delta) \int_0^{\delta/\theta} \left(\frac{\partial f}{\partial \eta}\right)^2 d\eta$
E	$\int_0^1 (1-f^2) f d\eta$
$f(\eta)$	The velocity profile, u/U
$f_1(\eta)$	The Blasius flat plate velocity profile
G	$\int_0^1 (1-f) f d\eta$
H	The shape factor $= \delta_1/\theta = J/G$
J	$\int_0^1 (1-f) d\eta$
j	Exponent in the dimensionless laminar boundary layer equations ($j = 0$ for two-dimensional flow; $j = 1$ for axisymmetric flow)
L	Overall length scale of an airfoil or body of revolution
ℓ	$(\theta/\delta)\ (\partial f/\partial \eta)_0$

*Additional symbols are defined as they appear in the text.

m	$(\theta/\delta)^2(\partial^2 f/\partial\eta^2)_0$
p	Dimensionless fluid pressure
R	$R_L = \frac{U_\infty L}{\nu}$, the Reynolds number
$r_0(s)$	Function describing body's profile as a function of surface arc-length s
s	Dimensionless surface arc-length starting at the front stagnation
t^*	$\theta^2 R_L$
U	Dimensionless inviscid slip velocity at the body's surface (usually the potential flow velocity)
U'	dU/ds
U_∞	Free-stream velocity
u	Dimensionless tangential component of boundary layer velocities
v	Dimensionless normal component of boundary layer velocities
x	Dimensionless axial coordinate
z	Dimensionless coordinate normal to the body's surface
δ	Dimensionless boundary layer thickness
δ_1	The displacement thickness $\delta \int_0^1 (1-f)d\eta = \delta J$
ε	$\delta \int_0^1 (1-f^2)fd\eta$
η	Similarity variable, z/δ
θ	The momentum thickness, $\delta \int_0^1 (1-f)fd\eta = \delta G$

ν	Kinematic viscosity
ρ	Fluid density
τ_w	The dimensionless wall shear stress, $\rho\nu(\partial u/\partial z)_0$

ABSTRACT

The two-parameter momentum-integral method of M.R. Head for the calculation of two-dimensional or axisymmetric, incompressible, laminar boundary layers is reworked into an analytical form, thereby eliminating the graphical data of Head's original method. The revised method is coded for digital computer solution and a number of test cases are computed. The numerical results for these test cases are compared with very accurate boundary layer solutions obtained by a finite-difference method and also results obtained using the Pohlhausen momentum-integral method. It was found that the Head method is vastly superior to the Pohlhausen method yet requires only about the same computational expense. Head's method yields very accurate results even for highly nonsimilar boundary layers.

ADMINISTRATIVE INFORMATION

The work described in this report was carried out by the David W. Taylor Naval Ship Research and Development Center (DTNSRDC), Ship Performance Department, in support of the Ship Acoustic Department's Laminar Flow Program. This program is sponsored by the Naval Sea Systems Command (SEA-037) under Program Element 25634N, Project SO218011, Task Area 20052, and was performed under Work Unit 1942-087.

INTRODUCTION

The accurate numerical solution of the two-dimensional and axisymmetric incompressible laminar boundary-layer equations is now a routine matter. Finite difference methods are normally used for such calculations. These methods require fairly substantial computing equipment for efficient implementation. Representative examples of finite difference laminar boundary-layer calculation methods are given by Blottner[1*] and Cebeci and Bradshaw.[2]

Simpler laminar boundary layer calculation methods that are of comparable accuracy to the finite difference methods are desirable for several reasons. For instance, it would be useful to have an accurate laminar boundary layer calculation method that can be implemented on a programmable

*A complete listing of references is given on page 45.

desk or hand-held calculator. When used on a large computer, such a method would be considerably less expensive to execute than the finite difference methods.

The simplest methods of calculating laminar boundary layers are the one-parameter momentum integral methods; typically Pohlhausen's method as described by Rosenhead.[3] These momentum integral methods yield acceptable accuracy only over a limited range of variations of the external pressure distribution and are not always capable of accurately predicting the laminar boundary layer separation point. Furthermore, although one-parameter momentum integral methods can be extended easily to include extraneous effects such as surface suction or free-stream unsteadiness, the results computed by these extensions are not always very accurate.

Two-parameter momentum integral methods for calculating laminar boundary layers, although somewhat more complicated than the one-parameter methods, are, however, considerably simpler than the finite difference methods. Many two-parameter methods have been considered in the past (see Rosenhead[3] and Waltz[4] for extensive discussions of momentum integral methods). However, only the method of Head[5] seems to have fulfilled the expectation of considerably higher accuracy in predicting the boundary layer properties than the simpler one-parameter momentum integral methods. Head's two-parameter momentum integral method (henceforth referred to as simply the Head method) seems to have received little attention because it was intended for manual calculation and thus it required the use of much graphical data. The graphical data of the Head method is based on a seemingly complicated set of graphical velocity profile shapes that discouraged analytical description. Gadd, Jones and Watson (see Rosenhead[3]) and Waltz[4] give only superficial coverage to the Head method in their notable reviews of momentum integral methods even though Head had shown by some sample calculations that his method was very accurate.

A reexamination of the Head two-parameter momentum integral calculation method for two-dimensional and axisymmetric laminar boundary layers shows that his graphical construction of the velocity profiles is fairly easily

transformed into an analytical form. From this analytical form of the velocity profiles, all the necessary relationships between the parameters of the Head method were easily computed, thereby replacing all the graphical data by polynomial expressions. The method can then be easily programmed for machine calculations. This report briefly outlines and recasts the Head two-parameter method into an analytical form. The report contains a comparison of some sample numerical results which were obtained using the Head method, the Pohlhausen method, and a finite difference method described by Blottner.[1] These results show the remarkable accuracy of the Head method. Although the modification of the Head two-parameter method that is given in this report has not been reduced to a compact form so that it can be programmed for a hand-held calculator, it seems that this could be done. Furthermore, the accuracy of the Head method encourages a belief that it can be extended for the accurate calculation of certain unsteady laminar boundary layers. The calculation of unsteady boundary layers by the two-parameter momentum integral method would be considerably less expensive than the calculation of such boundary layers using finite difference methods. A brief discussion of this point is given in the last section of this report. This report concludes with an Appendix in which the computer program for the Head two-parameter method is given.

DESCRIPTION OF HEAD'S METHOD

In this section a brief outline of the Head method is given. For the complete details of the method, the reader is referred to Head's original report.[5]

Consider the two-dimensional or axisymmetric laminar boundary layer on the surface of an airfoil or body of revolution in axial flow. The problem is scaled as follows: Distances are referred to the overall length scale L of the body. Velocities are referred to the uniform flow velocity U_∞ at infinity and stresses are referred to the total head $1/2\rho U_\infty^2$ where ρ is the constant density of the fluid. Suppose the configuration in Figure 1 depicts a meridian cut through a body of revolution where x is the dimensionless axial coordinate, s is the dimensionless surface arc-length starting

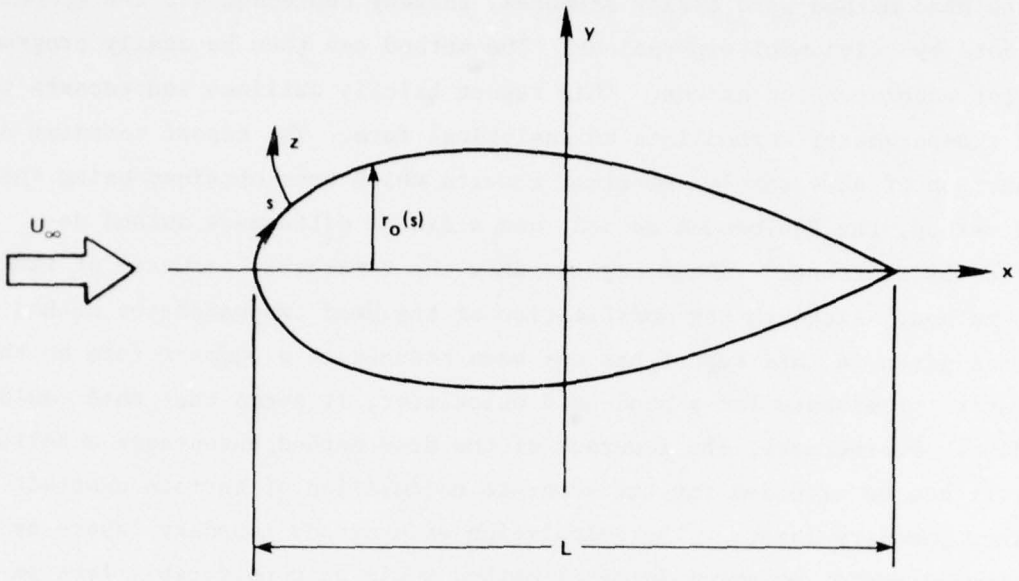


Figure 1 - Coordinates and Geometry of a Body of Revolution

at the front stagnation point, z is the dimensionless coordinate normal to the body's surface, and $r_0(s)$ describes the body's profile as a function of the surface arc-length s. If an airfoil is being considered instead of a body of revolution, s and z can be interpreted in a manner similar to that shown in Figure 1. The value of s is zero at the front stagnation point and increases downstream. The thickness distribution $r_0(s)$ of an airfoil measures the distance between the airfoil offsets and the camber line.

The appropriate laminar boundary layer equations can be found in any one of many standard works (Rosenhead,[3] for example). The dimensionless laminar boundary layer equations for planar and axisymmetric external flows are

$$u \frac{\partial u}{\partial s} + w \frac{\partial u}{\partial z} = - \frac{\partial p}{\partial s} + \frac{1}{R} \frac{\partial^2 u}{\partial z^2} \qquad (1a)$$

$$\frac{\partial r_0^j u}{\partial s} + \frac{\partial r_0^j w}{\partial z} = 0 \tag{1b}$$

$$-\frac{\partial p}{\partial s} = U \frac{\partial U}{\partial s} \tag{2}$$

where U is the dimensionless inviscid slip velocity at the body's surface (usually the potential-flow velocity) and j is either 0 or 1 for two-dimensional or axisymmetric flow, respectively. Standard boundary layer notation is used in which (u, w) denotes the dimensionless tangential and normal components of boundary layer velocities, respectively, p denotes the dimensionless fluid pressure, ν is kinematic viscosity and $R = U_\infty L/\nu$ is the Reynolds number. The boundary conditions are

$$u = w = 0 \quad \text{at } z = 0 \tag{3a}$$

$$u = U \quad \text{at } z = \delta \tag{3b}$$

where δ is the dimensionless boundary layer thickness.

By integrating Equation (1a) with respect to z across the boundary layer from the surface $z = 0$ of the body to the edge of the boundary layer $\delta(s)$, and similarly integrating Equation (1a) after first multiplying it by u, one obtains, after some algebraic manipulations (see Head[5] or Rosenhead[3]) the momentum and energy integral equations, respectively;

$$\frac{dt^*}{ds} = \frac{2}{U}\left[\ell + m(2+H) - \frac{t^* U \dfrac{dr_0^j}{ds}}{r_0^j}\right] \tag{4a}$$

$$\frac{dh}{ds} = \frac{1}{Ut^*}\{2D^* - h[\ell + m(H-1)]\} \tag{4b}$$

where $t^* = \theta^2 R_L$ (5a)

$$\ell = \frac{\theta}{\delta}\left(\frac{\partial f}{\partial \eta}\right)_0 \tag{5b}$$

$$m = \left(\frac{\theta}{\delta}\right)^2 \left(\frac{\partial^2 f}{\partial \eta^2}\right)_0 \tag{5c}$$

$$\theta = \delta \int_0^1 (1-f) f d\eta \tag{5d}$$

$$H = \frac{\delta_1}{\theta} \tag{5e}$$

$$\delta_1 = \delta \int_0^1 (1-f) d\eta \tag{5f}$$

$$h = \frac{\varepsilon}{\theta} \tag{5g}$$

$$\varepsilon = \delta \int_0^1 (1-f^2) f d\eta \tag{5h}$$

$$D^* = \left(\frac{\theta}{\delta_1}\right) \int_0^{\delta_1/\theta} \left(\frac{\partial f}{\partial \eta}\right)^2 d\eta \tag{5i}$$

and where $\eta = z/\delta$ and $f(\eta) = u/U$. In deriving Equations (4) it is assumed that the velocity profile u/U is a function $f(\eta)$ of the similarity variable $\eta = z/\delta(s)$ only, so that the partial differential Equations (1) reduce to the pair of ordinary differential Equations (4). Furthermore, the compatibility condition

$$t^* \frac{dU}{ds} = - m \tag{6}$$

is a consequence of the requirement that Equation (1a) be satisfied on the boundary $z = 0$. Thus, the three Equations (4a), (4b), and (6) are used to

obtain the values of the two parameters that determine the velocity profile $f(\eta)$ and the boundary layer scale $\delta(s)$.

The form of the velocity profile $f(\eta)$, that was given by Head, is

$$\frac{u}{U} = f(\eta) = f_1(\eta) + af_2(\eta;a) + bf_3(\eta;a) \tag{7}$$

where a and b are parameters, $f_1(\eta) \equiv 1 + g_1(\eta)$ is the Blasius flat plate velocity profile (Rosenhead[3])

$$f_2(\eta;a) \equiv g_2(\eta) - [g_2(\eta)-g_3(\eta)] \frac{0.493 + a}{1.493} \tag{8a}$$

and

$$f_3(\eta;a) \equiv g_4(\eta) - [g_4(\eta)-g_5(\eta)] \frac{0.493 + a}{1.493} \tag{8b}$$

The shape functions g_1 through g_5 were given in graphical form by Head. These graphs are not reproduced here. Instead, only the set of analytical interpolation functions of these graphs is given here. It was found that the functions

$$g_i(\eta) = p_i(\eta)e^{-4\eta^2} \tag{9}$$

where $p_i(\eta)$, $i = 1,\ldots, 5$ are polynomials in η, well represent the functions $g_i(\eta)$ and their first two derivatives. Table 1 lists the coefficients $C_i^{(k)}$ of the polynomials

$$p_i(\eta) \equiv \sum_{k=0}^{7} C_i^{(k)} \eta^k \tag{10}$$

that are used in Equation (9).

TABLE 1 - COEFFICIENTS OF THE VELOCITY PROFILE SHAPE FUNCTIONS

k	$C_1^{(k)}$	$C_2^{(k)}$	$C_3^{(k)}$	$C_4^{(k)}$	$C_5^{(k)}$
0	-1.0	0.0	0.0	0.0	0.0
1	1.747	3.5	3.5	0.0	0.0
2	-4.0	0.0	0.0	-15.0	-15.0
3	6.4820091	11.525525	-61.488444	49.494362	112.00692
4	-6.053572	-34.55105	223.78189	-167.59281	-354.96093
5	3.5642032	20.025525	-353.94145	340.86788	561.74806
6	-1.1218303	0.0	261.49099	-305.15578	-434.862
7	0.0	0.0	-73.342991	97.331351	131.01295

With the aid of the profile family, Equation (7), and the functions of Equations (8), (9), and (10), all the parameters of Equations (5a) through (5i) can be evaluated in terms of the parameters a and b as follows. Let G, J, E, and D be defined by

$$G \equiv \int_0^1 (1-f)f\,d\eta \tag{11a}$$

$$J \equiv \int_0^1 (1-f)\,d\eta \tag{11b}$$

$$E \equiv \int_0^1 (1-f^2)f\,d\eta \tag{11c}$$

$$D \equiv \int_0^1 \left(\frac{\partial f}{\partial \eta}\right)^2 d\eta \tag{11d}$$

where the upper limit δ_1/θ in Equation (5i) can be taken as 1 in the definition (11d)

because for $\delta_1/\theta > 1$, $(\partial f/\partial \eta)' \doteq 0$. Then, by substituting Equations (11a) through (11d) in Equations (5a) through (5i), the following formulas are obtained:

$$m = G^2 \left(\frac{\partial^2 f}{\partial \eta^2}\right)_0 \tag{12a}$$

$$\ell = G \left(\frac{\partial f}{\partial \eta}\right)_0 \tag{12b}$$

$$H = \frac{J}{G} \tag{12c}$$

$$h = \frac{E}{G} \tag{12d}$$

$$D^* = GD \tag{12e}$$

Equation (6) yields the following formula for the variable t*;

$$t^* = -G^2 \frac{\left(\partial^2 f/\partial \eta^2\right)_0}{(dU/ds)} \tag{12f}$$

All of the quantities m to D* given by Equations (12a) through (12e) are functions (either ordinary polynomials or rational polynomials) only of a and b, and t* is a function of a and b and the known function (dU/ds). Equations (12) are easily evaluated by using the polynomials G, J, E, D, $(\partial f/\partial \eta)_0$ and $(\partial^2 f/\partial \eta^2)_0$ which are listed in Table 2. Thus, Equations (4) and (6) can be reduced to equations determining the parameters a and b as a function of s for given distributions of the inviscid surface velocity U(s) and body radius (or thickness) distribution $r_0(s)$.

At the front stagnation point of blunt bodies, a and b have the values

$$a = 0.445, \qquad b = 0.240 \quad \text{at } s = 0 \tag{13a}$$

and

$$a = 0.3225, \qquad b = 0.1605 \quad \text{at } s = 0 \tag{13b}$$

for two-dimensional and axisymmetric flows, respectively. These values for a and b are obtained by finding the roots of the algebraic equations resulting from Equations (4) upon assuming that both dt*/ds and dh/ds are bounded in the limit $s \to 0$.

TABLE 2 - FUNCTIONS J, G, D, E, $\left(\frac{\partial f}{\partial \eta}\right)_0$, AND $\left(\frac{\partial^2 f}{\partial \eta^2}\right)_0$ IN TERMS OF THE PARAMETERS a AND b

$$\left(\frac{\partial f}{\partial \eta}\right)_0 = 1.747 + 3.5a$$

$$\left(\frac{\partial^2 f}{\partial \eta^2}\right)_0 = -30b$$

$$J = 0.32832 - 0.31897a + 0.14931a^2 - 0.1876b - 0.14947ab$$

$$\begin{aligned} G = {} & 0.12733 - 0.05005a - 0.10089a^2 + 0.13915a^3 - 0.035073a^4 \\ & + (0.049054 + 0.13403a - 0.23383a^2 + 0.072643a^3)b \\ & + (-0.059455 + 0.093916a - 0.037702a^2)b^2 \end{aligned}$$

$$\begin{aligned} D = {} & 1.3713 + 1.2124a + 1.8754a^2 - 1.8268a^3 + 0.60524a^4 \\ & + (-0.83784 - 1.9728a + 3.3493a^2 - 1.2919a^3)b \\ & + (0.94797 - 1.5608a + 0.6957a^2)b^2 \end{aligned}$$

$$\begin{aligned} E = {} & 0.20069 - 0.054971a - 0.18242a^2 + 0.1758a^3 \\ & + 0.043227a^4 - 0.055451a^5 + 0.0095503a^6 \\ & + (0.060072 + 0.26478a - 0.29465a^2 \\ & - 0.11611a^3 + 0.152a^4 - 0.029892a^5)\, b \\ & + (-0.11365 + 0.089165a + 0.12215a^2 \\ & - 0.13673a^3 + 0.031219a^4)b^2 + (0.020669 \\ & - 0.049679a + 0.040141a^2 - 0.010879a^3)b^3 \end{aligned}$$

NUMERICAL METHOD FOR SOLVING THE MOMENTUM AND ENERGY INTEGRAL EQUATIONS

In this section, the numerical method for computing the boundary layer characteristics for given distributions of U and r_0 is described. This numerical method is implemented in the computer program HEAD given in the Appendix and is intended for use on a digital computer.

The numerical solution of Equations (4) can be approached in two ways. The first way is by the direct calculation of t* and h using Equations (4). This would require the solution of a system of algebraic equations for the parameters a and b in terms of the known values of t* and h at each step of the calculations. The second way to numerically solve Equations (4) is to rewrite them in terms of the parameters a and b. This eliminates the necessity of solving algebraic equations at each step and is the basis of the numerical procedure described below.

Equations (4) can be written in the form

$$\frac{da}{ds} = F(a,b) \tag{14a}$$

$$\frac{db}{ds} = G(a,b) \tag{14b}$$

where

$$F(a,b) \equiv \frac{\left(A \frac{\partial h}{\partial b} - B \frac{\partial m}{\partial b}\right)}{\mathcal{D}} \tag{15a}$$

$$G(a,b) \equiv \frac{\left(B \frac{\partial m}{\partial a} - A \frac{\partial h}{\partial a}\right)}{\mathcal{D}} \tag{15b}$$

$$A \equiv \frac{mU''}{U'} - \frac{2}{U'}\left[U'\{\ell+m(2+H)\} + \frac{mU(r_0^j)'}{r_0^j}\right] \tag{15c}$$

$$B \equiv \frac{-U'\{2D^*-h[\ell+m(H+1)]\}}{mU} \tag{15d}$$

$$\mathcal{D} \equiv \frac{\partial m}{\partial a}\frac{\partial h}{\partial b} - \frac{\partial m}{\partial b}\frac{\partial h}{\partial a} \tag{15e}$$

and where

$$U' \equiv \frac{dU}{ds} \quad \text{and,} \qquad (r_0^j)' \equiv \frac{dr_0^j}{ds} \tag{15f}$$

Equations (14) can be integrated numerically using the predictor-corrector method that is based on the generalized midpoint and trapezoidal quadrature formulas. The values of a(s) and b(s) at the points s_i of a general grid $\{s_i\}$ i = 0, 1, 2, ... are denoted by a_i and b_i, respectively. Then, given the values a_i and b_i of a and b, respectively, at s_i, the values a_{i+1} and b_{i+1} at s_{i+1} are obtained using the formulas

$$\bar{a}_{i+1} = c^2 a_{i-1} + (1-c^2)a_i + \Delta s_i(1+c)F(a_i,b_i) \tag{16a}$$

$$\bar{b}_{i+1} = c^2 b_{i-1} + (1-c^2)b_i + \Delta s_i(1+c)G(a_i,b_i) \tag{16b}$$

$$a_{i+1} = a_i + \frac{\Delta s_{i+1}}{2}\left(F(a_i,b_i) + F(\bar{a}_{i+1},\bar{b}_{i+1})\right) \tag{17a}$$

$$b_{i+1} = b_i + \frac{\Delta s_{i+1}}{2}\left(G(a_i,b_i) + G(\bar{a}_{i+1},\bar{b}_{i+1})\right) \tag{17b}$$

where $\Delta s_{i+1} = s_{i+1} - s_i$ and $c = \Delta s_{i+1}/\Delta s_i$ for i = 2, 3, ... For the first station s_1 past the stagnation point $s_0 = 0$, a special procedure is used. In order to avoid the front stagnation point where $r_0 = U_0 = 0$ and da/ds and db/ds are indeterminate, the numerical integration is started by iterating the backward Euler formulas as follows:

$$a_1^{(k)} = a_0 + \Delta s_1\, F(a_1^{(k-1)},\, b_1^{(k-1)},\, s_1) \tag{18a}$$

$$b_1^{(k)} = b_0 + \Delta s_1\, G(a_1^{(k-1)},\, b_1^{(k-1)},\, s_1) \tag{18b}$$

where k = 1, 2, 3, ..., $a_1^{(0)} = a_0$, $b_1^{(0)} = b_0$ and the third or fourth (k = 3 or 4) iterates usually are sufficiently accurate approximations of a_1 and b_1.

The numerical integration method, Equations (16) and (17) changes the step size Δs_{i+1} as the integration proceeds downstream from the point $s = s_1$, thus automatically generating the grid $\{s_i\}$ in such a way as to keep a certain measure e of the numerical integration error below a preset value $\bar{E}$. Each new step $s_{i+1} = s_i + \Delta s_{i+1}$ of the numerical integration procedure is obtained from the previous step $s_i = s_{i-1} + \Delta s_i$ by setting $\Delta s_{i+1} = A\Delta s_i$. In the first attempt at moving from step s_i to s_{i+1}, A is either set equal to 1 or is the value given by the procedure, described below, for increasing the step size if higher accuracy than desired is being obtained.

Suppose the values of $\bar{a}_{i+1}$, $\bar{b}_{i+1}$, a_{i+1}, b_{i+1} have been computed at $s_{i+1} = s_i + A\Delta s_i$. Let e be the measure of the absolute error, defined by

$$e = \max \left\{ \frac{|\bar{a}_{i+1} - a_{i+1}|}{d}, \frac{|\bar{b}_{i+1} - b_{i+1}|}{d} \right\} \tag{19}$$

where $d = 3 + 2/c^3$. If $e > \bar{E}$ the current values of s_{i+1}, a_{i+1}, $\bar{a}_{i+1}$, b_{i+1} and $\bar{b}_{i+1}$ are discarded and a new value of A is estimated by using the formula

$$A = \left(\frac{0.97\bar{E}}{e} \right)^{1/3} \tag{20}$$

and new values of $s_{i+1} = s_i + A\Delta s_i$, a_{i+1}, $\bar{a}_{i+1}$, b_{i+1} and $\bar{b}_{i+1}$ are computed. The procedure is repeated until the inequality $e \leq \bar{E}$ is satisfied. In all the numerical experiments using this procedure to satisfy the accuracy requirement that $e \leq \bar{E}$ it has never been necessary to repeat the procedure more than once at each step.

If too much accuracy, say, for example, $e \leq \bar{E}/10$, is being obtained at step s_{i+1} then the current value of Δs_{i+1} is accepted but the next location s_{i+2} is obtained by using the value $\Delta s_{i+2} = A\Delta s_{i+1}$ where

$$A = \left(\frac{\bar{E}}{10e} \right)^{1/3} \tag{21}$$

This method of changing the integration step size has proven to be very effective in many numerical experiments. The value of E that has been used in all the computations presented in this report is $E = 10^{-5}$.

Special care must be exercised when U' vanishes. Formula (12f) shows that when U' = 0, m must also be zero in order for t* to be finite. Whenever U' is close to zero (say $\approx 10^{-3}$) at location s_{i+1}, a special procedure is used to evaluate t* = -m/U' in order to maintain accuracy. Instead of simply evaluating the ratio m/U' in order to obtain t* at the point s_{i+1}, t^*_{i+1} is obtained by numerically integrating Equation (4a) using a special explicit procedure based on the known values t^*_{i-1}, t^*_i (and the rest of the data needed in Equation (20)) at the points s_{i-1} and s_i, respectively. The value $t^{*\prime}_i$ of dt*/ds at the point s_i can be computed using Equation (4a). Then, the value of t^*_{i+1} at the point s_{i+1} can be computed using the two-point explicit Hermite extrapolation formula

$$t^*_{i+1} = t^*_{i-1} + \left[\frac{2(t^*_i - t^*_{i-1})}{\Delta s_i} - t^{*\prime}_i\right](\Delta s_i + \Delta s_{i+1})$$

$$+ \left[\frac{t^{*\prime}_i}{\Delta s_i} - \frac{(t^*_{i-1} - t^*_{i-1})}{\Delta s_i^2}\right](\Delta s_i + \Delta s_{i+1})^2 \qquad (22)$$

If the values of t^*_{i-1}, t^*_i and the values of the rest of the data needed in Equation (4a) are known to second-order accuracy in the step sizes Δs_i, then the value of t^*_{i+1} computed using Equation (22) is of second-order accuracy in the step sizes Δs_i and Δs_{i+1}. Thus, even when U' and m become very small, their ratio t* can still be calculated accurately using Equation (22).

The characteristics of the boundary layer are determined once the distributions along the body's profile of the values of a and b are known. The boundary layer parameters that are usually of main interest are the displacement thickness $\delta_1 = \delta J$, the momentum thickness $\theta = \delta G$, the shape factor $H = \delta_1/\theta$, and the skin friction coefficient $c_f = \tau_w / \left[\frac{1}{2}(\rho U_0^2)\right]$ where τ_w is the dimensional wall shear stress $\rho\nu(\partial u/\partial z)_0$. These parameters, scaled

only by the overall body length L, overall velocity U_0 and the fluid density ρ are given in terms of the calculated values of t* and H by

$$\theta = \sqrt{\frac{t^*}{R_L}} \tag{23a}$$

$$\delta_1 = \theta H \tag{23b}$$

$$C_f = \frac{2U\ell}{\sqrt{t^* R_L}} \tag{23c}$$

The intrinsic boundary layer scale $\delta(s)$ can be computed using the relation

$$\delta(s) = -\left[\frac{\left(\frac{\partial^2 f}{\partial \eta^2}\right)_0}{R_L U'} \right]^{1/2} \tag{24}$$

once the distributions of a and b are known. Thus, the boundary layer velocity profiles $f(\eta) = u/U$ can be obtained using Equations (8) through (10). The velocity profiles $f(\eta)$ can be plotted in terms of any other body normal coordinate $\tilde{\eta}$, that is scaled by some other length scale L, by simply multiplying η by $L\delta/L$ to obtain $\tilde{\eta}$.

NUMERICAL RESULTS AND DISCUSSION

The laminar boundary layer characteristics were calculated for three sample bodies of revolution by three methods; the Pohlhausen and Head momentum integral methods and the finite-difference method designated DB, that is described and referred to as the Davis-B scheme by Blottner.[1] The DB finite-difference method provides second-order accuracy in the step sizes Δs and $\Delta\eta$ along and normal to the body, respectively. The values of the step sizes Δs and $\Delta\eta$ that were used in the DB method were small enough for each body considered in this report that further reductions in the step sizes made less than a one-percent difference in any of the computed values

of the boundary layer characteristics. Accordingly, the results obtained using the DB method are presumed to be exact for comparison purposes. As a further check of these computational results, the boundary layer characteristics were also calculated for several bodies using a computer program based on the Keller-Cebeci box method.[2] The numerical results obtained with the box method in all cases agreed to within two percent of the results obtained with the DB method.

The boundary layer characteristics were computed by all three methods for a number of bodies (both cylindrical and axisymmetric bodies). It is sufficient for comparison of the accuracy and computational cost of the Pohlhausen, Head, and DB methods to illustrate the numerical results on only three sample bodies since, in all the cases considered, Head's method gave equally accurate results. The three illustrative bodies that were chosen are fairly extreme cases for which the prediction of the laminar boundary layer can be classified as either fairly easy or hard. The easiest possible case is probably the flat plate boundary layer but this was not deemed a fair test because Head's method has an accurate interpolation of the flat plate laminar boundary layer velocity profile built into it.

The two easy cases considered for this report are the flow past a sphere and the axial flow past a spheroid with a length-to-diameter ratio of 4 to 1. These are common test cases for which many approximate boundary layer solutions have been published. Any viable calculation method must be able to predict the laminar boundary layer on these bodies with reasonable accuracy.

The hard test case considered for this report is a rather contrived mathematically defined body of revolution with a flat nose. Such extremely blunt bodies have very rapid and large variations of the surface pressure distribution. These variations can cause approximate laminar boundary layer calculation methods to predict catastrophically inaccurate boundary layer characteristics such as separation where it does not occur or no separation where it does occur. The hard test case chosen for illustration is designated body B. On body B, the boundary layer nearly separates at a point

near the nose of the body but recovers and eventually culminates in actual boundary layer separation near the tail of the body. It seems that body B is a sufficiently extreme case for laminar boundary layer calculation methods so that if a method is successful for body B, then one can expect it to be successful on most other bodies (two-dimensional or axisymmetric ones).

The first test case to be considered is the boundary layer flow on a sphere. It is sufficient to compare only the surface distribution of the values of the shape factor H and skin friction coefficient $C_f\sqrt{R_L}$. Figures 2 and 3 show the distributions of the surface values of $C_f\sqrt{R_L}$ and of H, respectively, plotted as a function of the axial coordinate x. The front stagnation point is located at the point x = -1. The values of $C_f\sqrt{R_L}$ and H that are obtained by the Pohlhausen, Head, and DB methods are denoted by the triangular, circular, and diamond shaped symbols, respectively. This notation is used in all the comparisons shown in this report. On the front of the sphere, from the stagnation point at x = -1.0 to the location x = -0.3, the numerical results that were obtained by the Pohlhausen, Head, and DB methods are plotted at the different station values x for clarity. From the station x = -0.3 to the point of separation near $x \approx 0.26$ the numerical results for $C_f\sqrt{R_L}$ and H obtained by each of the three methods are plotted at the same station values x.

It can be seen in Figures 2 and 3 that the laminar boundary layer calculation method of Head is considerably more accurate than the Pohlhausen method although the latter seems to be adequate for rough estimates. The separation point is predicted by the DB method to occur at the location x = +0.255. Head's method predicts the separation point at x = 0.265 and Pohlhausen's method predicts it at x = 0.305. The price that one has to pay for Head's method, over the price of the Pohlhausen method, for the more accurate boundary layer predictions is very low. The calculations for the sphere, given by the computer program of the Head method in the Appendix, required 3.5 seconds to execute on the CDC 6700 computer. The Pohlhausen method required 4.7 seconds using a less efficient integration routine. A comparable numerical integration method to the one used for

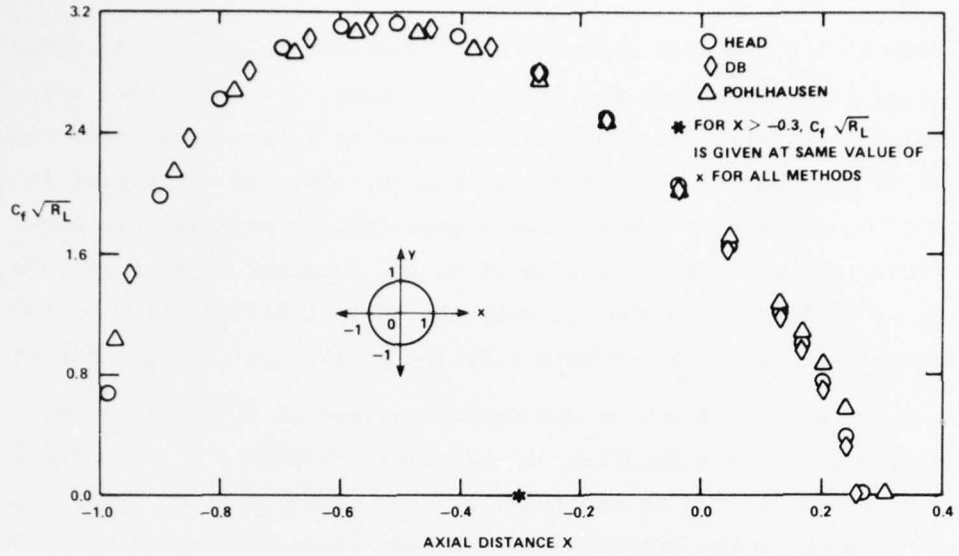


Figure 2 - Computed Skin Friction Coefficient on a Sphere

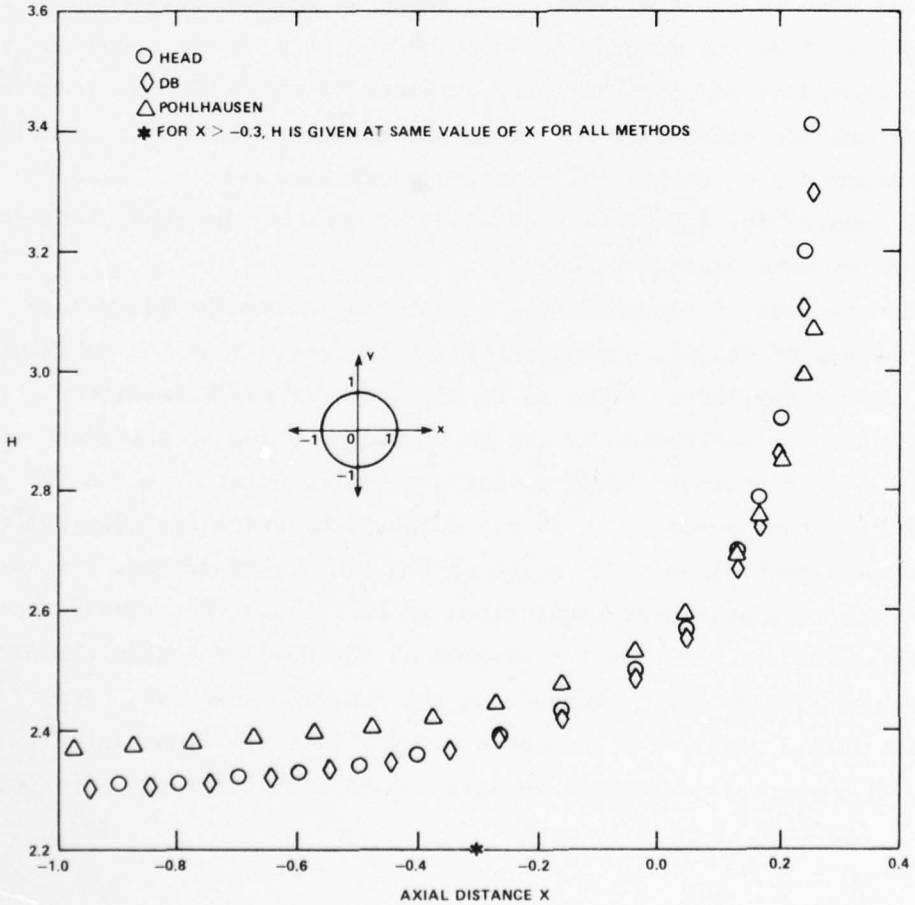


Figure 3 - Computed Shape Factor Parameter on a Sphere

Head's method could probably reduce by half the execution time of Pohlhausen's method. Thus, there is little to gain on a computer by using Pohlhausen's method in favor of Head's method. Furthermore, there is little difference in computer storage requirements between Pohlhausen's and Head's method, but this will be discussed later. The DB method required 25 seconds to execute for the sphere. (The Cebeci-Keller box method required slightly more execution time than the DB method for all the bodies considered.)

The comparison of the surface distributions of values of $C_f\sqrt{R_L}$ and H on the spheroid as calculated by the Pohlhausen, Head, and DB methods are shown in Figures 4 and 5. Essentially the same comments can be made concerning these comparisons as the ones for the sphere. The Pohlhausen predictions for the spheroid are better than for the sphere, although still not as good as Head's predictions. One must not be misled by the seemingly better predictions of the values of H near separation on the spheroid by Pohlhausen's method than by Head's method. It should be noted that the graph of H predicted by Pohlhausen's method crosses the exact DB graph of H very close to separation and, thus, the good agreement between values of H there is somewhat fortuitous. In contrast, separation (near point x = 0.665 in Figure 4) is predicted slightly more accurately by Head's method than by Pohlhausen's method. The computer execution times for calculating the boundary layer characteristics on the spheroid were 3, 7, and 35 seconds for the Head, Pohlhausen, and DB methods, respectively.

The final test case for comparing the laminar boundary layer calculation methods is body B which is described by the following set of equations. The flat nose is given by

$$x = 0, \text{ for } r_0 \in [0.0, 0.025] \tag{25a}$$

the spheroidal blending region is given by

$$r_0 = 0.025 + 0.02137 \left[\frac{2x}{0.113} - \left(\frac{x}{0.113}\right)^2\right]^{1/2} \tag{25b}$$

for $x \in [0.0, 0.113]$

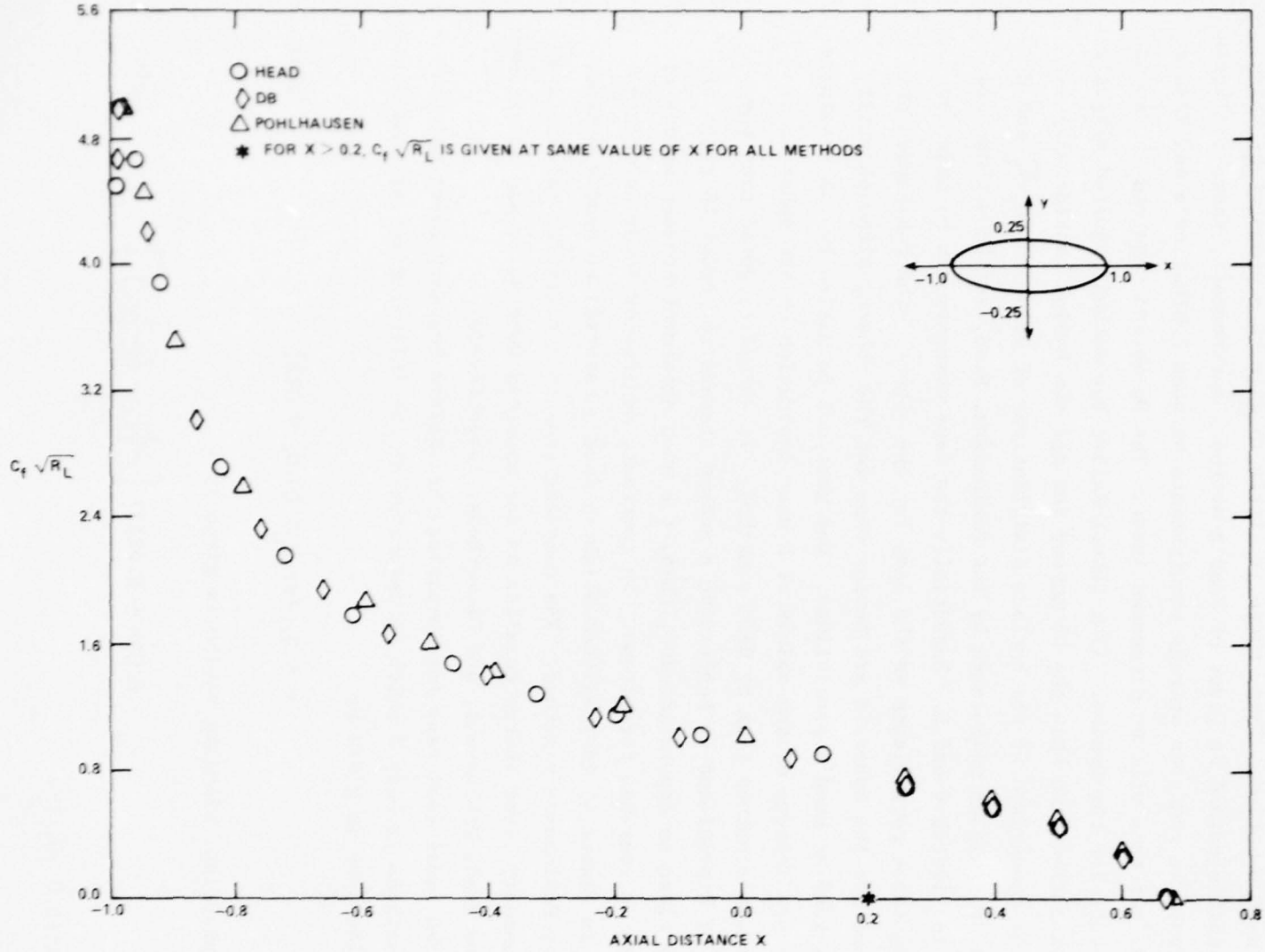


Figure 4 - Computed Skin Friction Coefficient on a 4 to 1 Spheroid

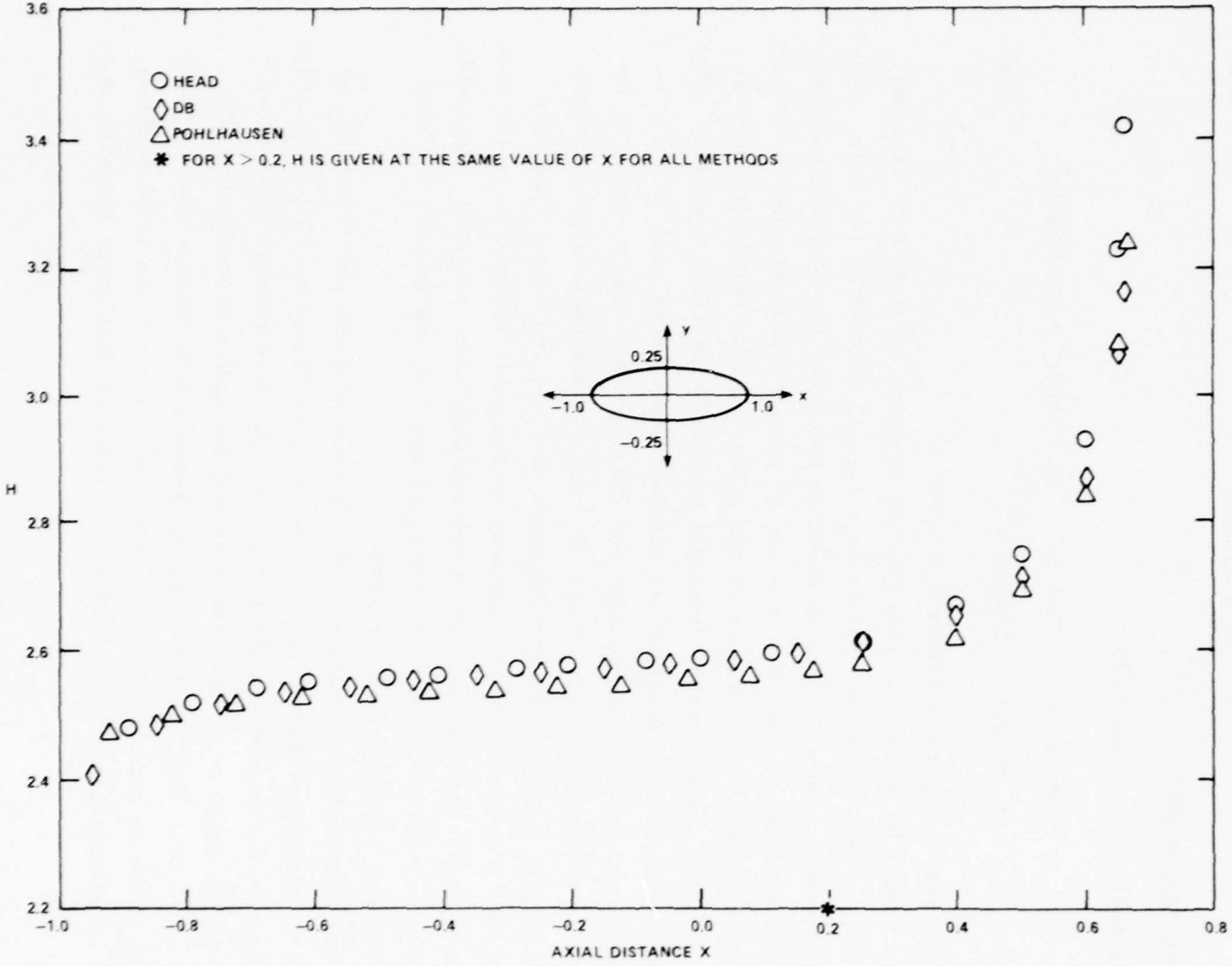


Figure 5 - Computed Shape Factor Parameter on a 4 to 1 Spheroid

the parallel middle body is given by

$$r_0 = 0.04637 \tag{25c}$$

for $x \in [0.113, 0.592]$

and the tail piece is given by

$$r_0 = 0.09274\ (z + 1.137153z^2 - 10.774885z^3 + 19.784286z^4 - 16.792534z^5 + 5.645977z^6)^{1/2} \tag{25d}$$

for $x \in [0.592, 1.0]$, $z = 1.47049x - 0.47049$

It should be noted that this profile equation for body B is used only to compute a set of discrete points $\{x_i, r_0(x_i)\}$ describing the body. This set of discrete points is used as input to calculate the body surface potential flow velocity values $\{U_j\}$ at a set of points midway between the defining points $\{r_0(x_i)\}$. The values of the potential flow velocities $\{U_j\}$ are computed using a panel method developed at DNTSRDC by Chang and Pien.[6] The laminar boundary layer programs for the Pohlhausen, Head, and DB methods that were used for the calculations described in this report, used, as input, the set of values $\{r_0(x_i)\}$ and $\{U_j\}$ and derived cubic spline interpolation polynominals for the distributions of $r_0(s)$ and $U(s)$. Thus, the actual body B used for the calculations is the cubic spline fit to the data points $\{r_0(x_i)\}$ that is implicit in the boundary layer programs. The cubic spline interpolation polynomials for $r_0(s)$ and $U(s)$ are exactly the same for all three boundary layer programs.

The distribution of values of the surface pressure coefficient C_p on body B is plotted versus surface arc-length s in Figure 6. The front stagnation point corresponds to the point s = 0.0. Note the extremely steep slope of the graph of C_p near the nose and its sudden reversal near the point $s \approx 0.03$. This rapidly changing pressure distribution near the nose of body B can be expected to cause large deviations of the local boundary layer characteristics from the characteristics of similarity boundary layer

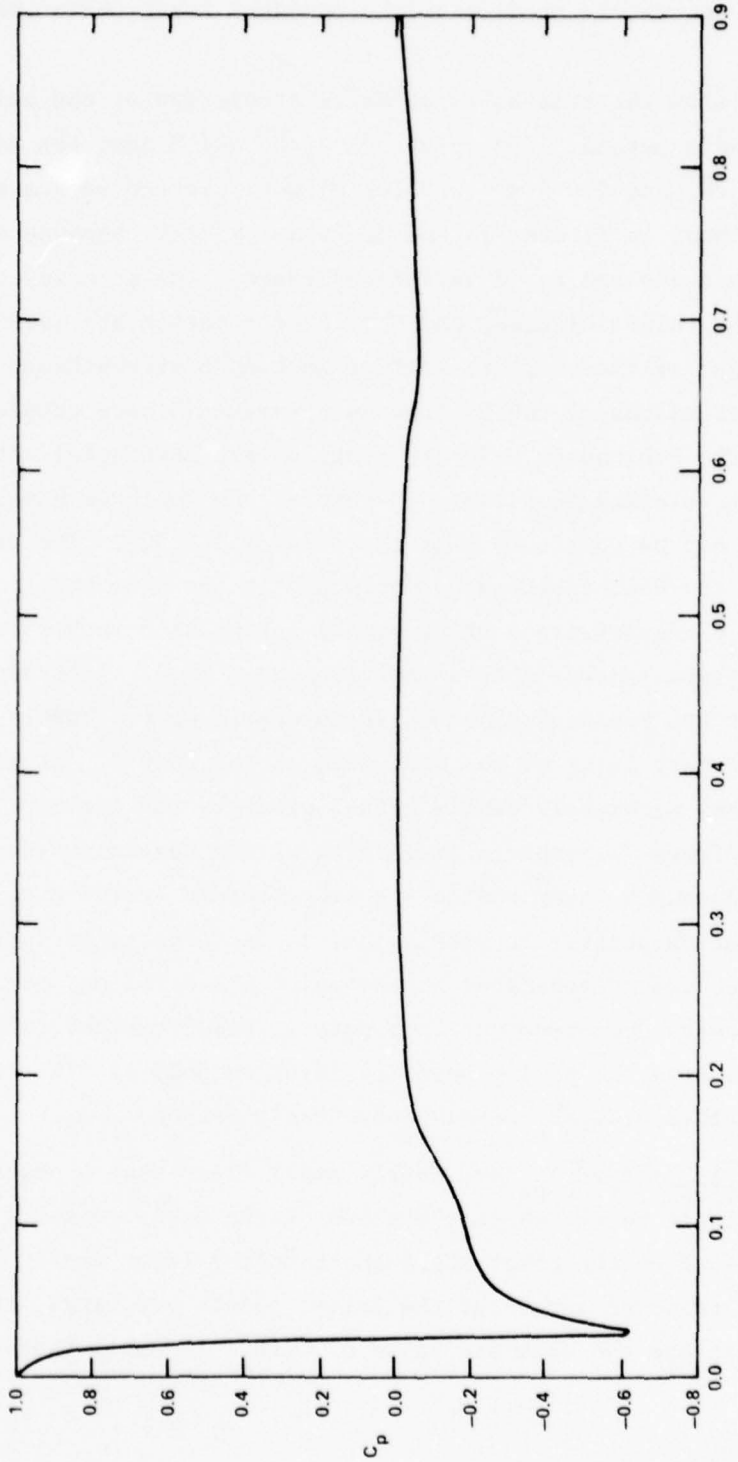


Figure 6 - Pressure Coefficient C_p on Body B

flow. Thus, one would expect most low order momentum integral methods to have difficulties in accurately predicting the boundary layer characteristics for body B.

Figures 7 and 8 show the remarkably accurate prediction of the values of $C_f\sqrt{R_L}$ and H by Head's method. The values of $C_f\sqrt{R_L}$ and H near the nose ($0.0 \leq s \leq 0.1$ for $C_f\sqrt{R_L}$ and $0.0 \leq s \leq 0.2$ for H) were plotted separately from the rest of the body in Figures 7a and 8a, respectively, because of their very large magnitudes and rapid variations there. The accuracy of the predictions of the values of $C_f\sqrt{R_L}$ and H by Head's method are even more remarkable in the light of the complete failure on body B of Pohlhausen's method. The predicted values of the Pohlhausen parameter Λ were outside the region in which the Pohlhausen velocity profiles are meaningful approximations of the exact velocity profiles. Therefore, the Pohlhausen method broke down and could not be continued past the point $s \approx 0.026$. The same type of breakdown of the Pohlhausen method occurred at the same point, $s \approx 0.026$, using two different types of numerical integration methods on a number of different fixed and variable integration nets $\{s_i\}$. Accordingly, it was concluded that the Pohlhausen method is incapable of reasonably approximating the boundary layer on the flat face of the body B. It is interesting to note how accurately Head's method predicts the laminar separation point in Figure 7b, especially in view of the severe pressure variations that the boundary layer had to negotiate before arriving at the actual laminar separation point. In particular, it is to be noted that the laminar boundary layer nearly separates at the point $s \approx 0.145$ on body B but then rapidly recovers downstream of this point. Head's method faithfully reproduces this behavior of the boundary layer on body B. The relative errors of the values of $C_f\sqrt{R_L}$ obtained by Head's method, i.e., $((C_f\sqrt{R_L})_{HEAD} - (C_f\sqrt{R_L})_{DB})/(C_f\sqrt{R_L})_{DB}$ are fairly small, less than 5 percent except near some isolated points where the slope of the $C_f\sqrt{R_L}$ versus s curve is nearly vertical at the point where the boundary layer nearly separates. Although the relative errors at the latter points are large, it is not deemed serious because the absolute value of $C_f\sqrt{R_L}$ is small there.

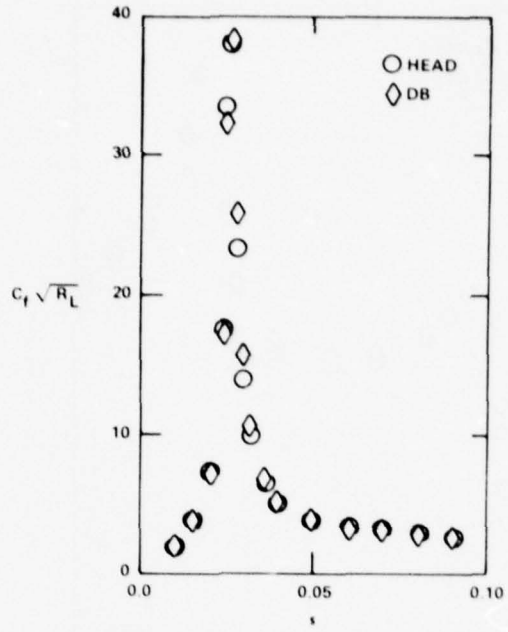


Figure 7a - $0.0 \leq s \leq 0.10$

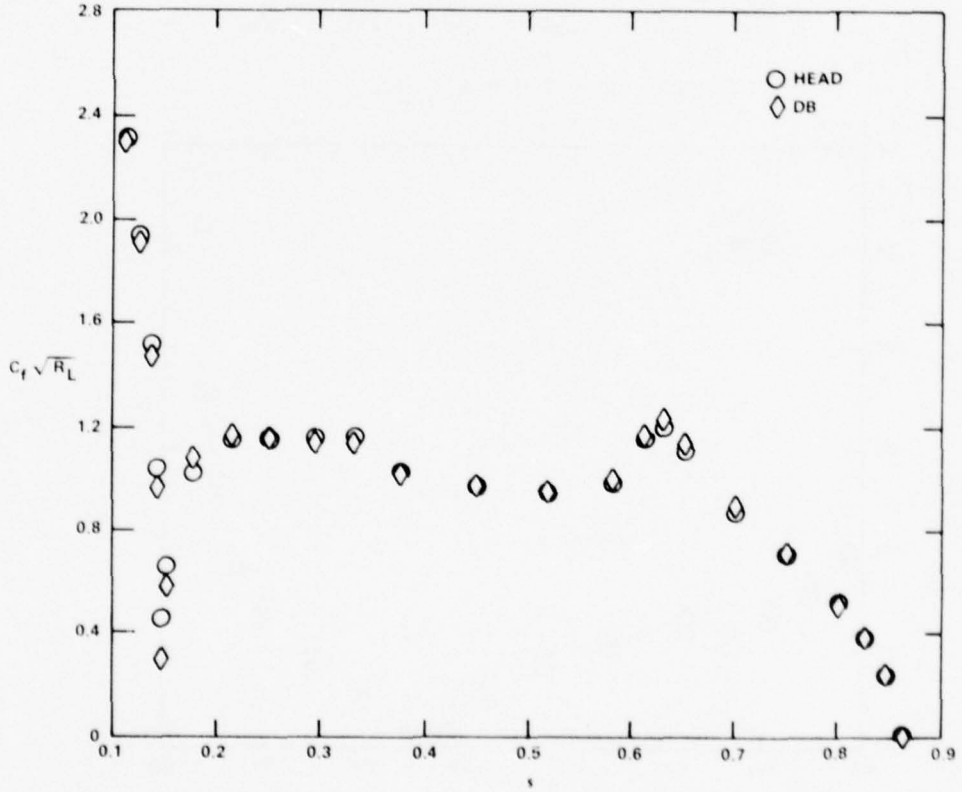


Figure 7b - $0.1 \leq s \leq 0.9$

Figure 7 - Computed Skin Friction Coefficient on Body B

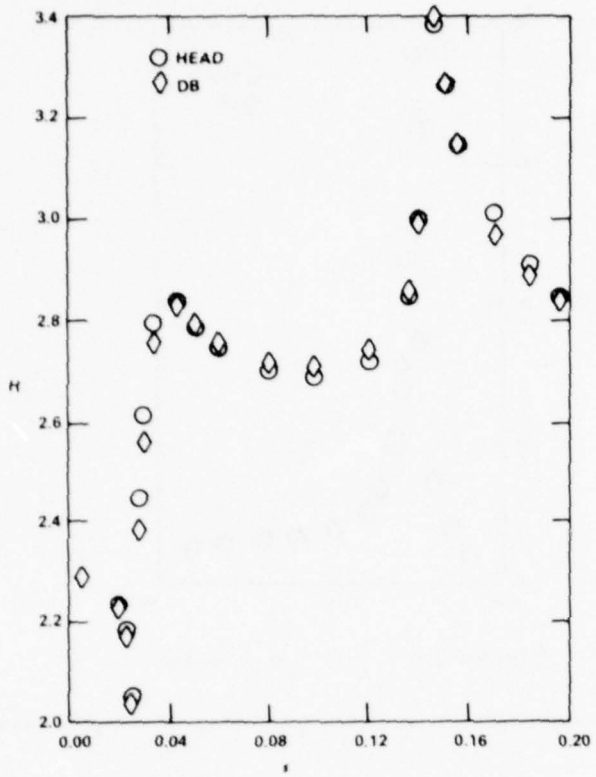


Figure 8a - $0.0 \leq s \leq 0.2$

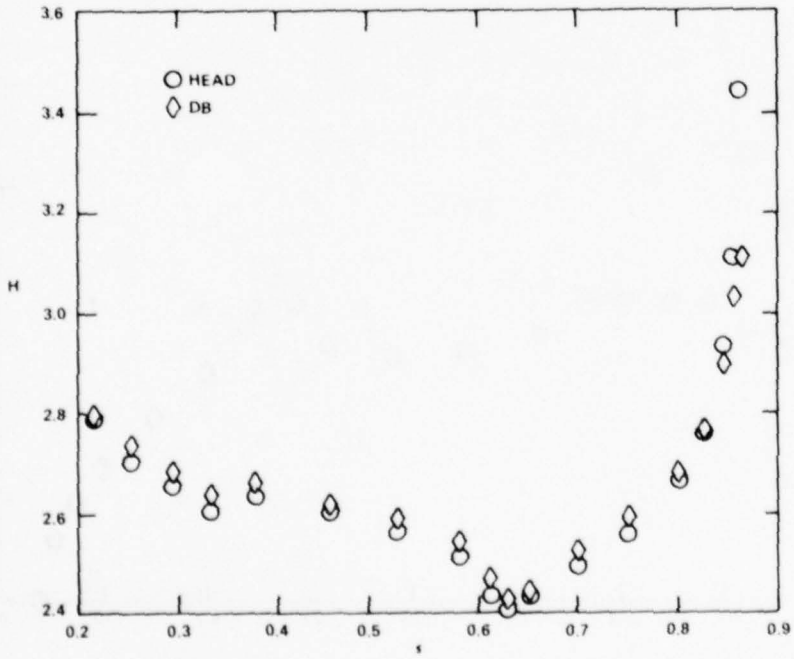


Figure 8b - $0.2 \leq s \leq 0.9$

Figure 8 - Computed Shape Factor Parameter on Body B

It is interesting to note that for the case of body B, Head's method required 19 seconds of execution time and the DB method required 217 seconds. By requiring less accuracy of the DB method, for example, accuracy comparable to the results of Head's method, the step sizes required by the DB method can be increased to large values that require only about half the execution time of the accurate step-size DB method. Thus, for equivalent accuracy levels, Head's method would still be about five times as fast as the DB method. It is also interesting to note that the automatic step size selection mechanism of the numerical method that was used to implement Head's method reduced the step sizes Δs_i to values on the order of 10^{-5} in order to accurately integrate the boundary layer Equations (14) near the nose of body B. On the parallel middle body portion of body B, the step sizes Δs_i automatically increased to values on the order of 10^{-2}.

There are several reasons why it may also be necessary to compute the boundary layer velocity profile $f(\eta) = u/U$ and its first two derivatives, e.g., for boundary-layer instability analysis. Thus, a comparison of the computed values of the velocity profile f and its first two derivatives f' and f" that were obtained from Head's method and the DB method is made in Figures 9 through 11. These velocity profiles are plotted versus the coordinate η_* that is normal to the body surface and that is made dimensionless by the local value of the displacement thickness $\delta_1(s)$. Figures 9 through 11 show the body B velocity profiles and their derivatives $df/d\eta_*$ and $d^2f/d\eta_*^2$ at locations: $s \approx 0.026$, where C_p is a minimum and is rapidly varying; at $s \approx 0.146$, where boundary layer nearly separates; and at $s \approx 0.862$, where the boundary layer is just beginning to separate. The values of the velocity profiles $f(\eta)$ are very accurately predicted by Head's method at all of the above three points s on the body. The values of the first derivatives $df/d\eta_*$ are less accurate but still fairly good. Substantial inaccuracy is found only in the values of the second derivative $d^2f/d\eta_*^2$. The major discrepancies between the values of $d^2f/d\eta_*^2$ predicted by Head's method and the DB method are mainly confined to a region very close to the body surface.

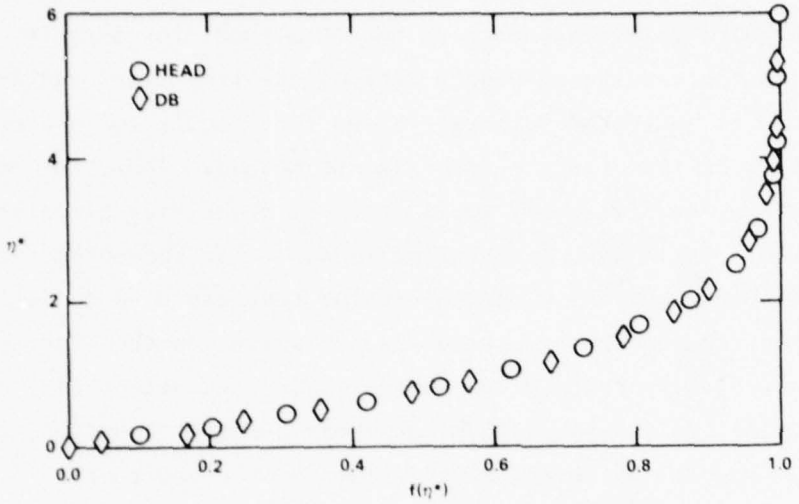


Figure 9a - Comparison of f(η*) Profiles

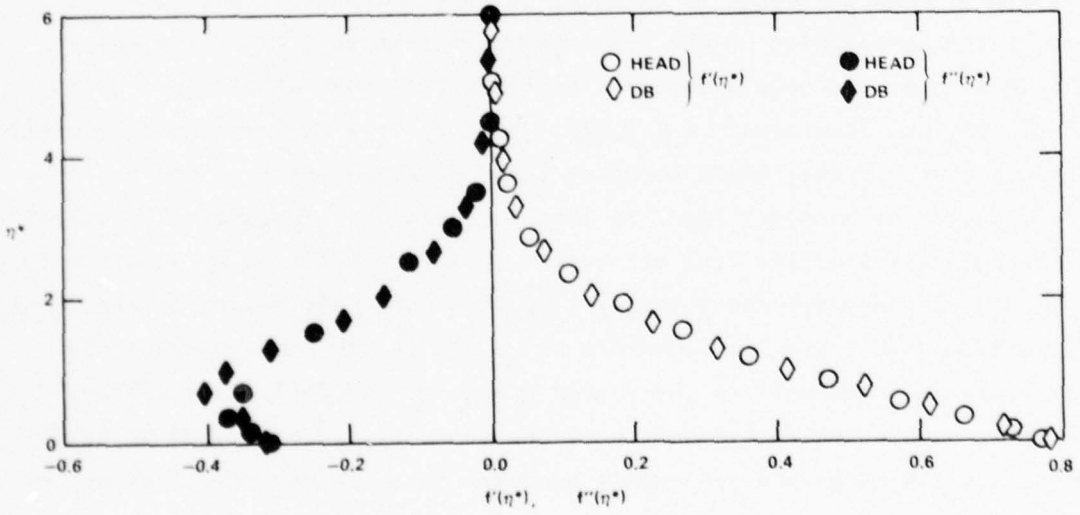


Figure 9b - Comparison of f'(η*) and f''(η*) Profiles

Figure 9 - Computed Velocity Profiles on Body B at s = 0.026 ($C_{p_{min}}$)

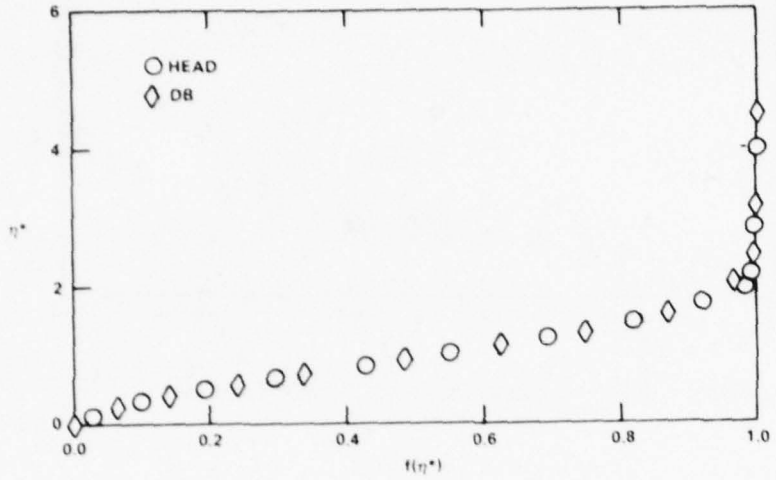


Figure 10a - Comparison of f(η*) Profiles

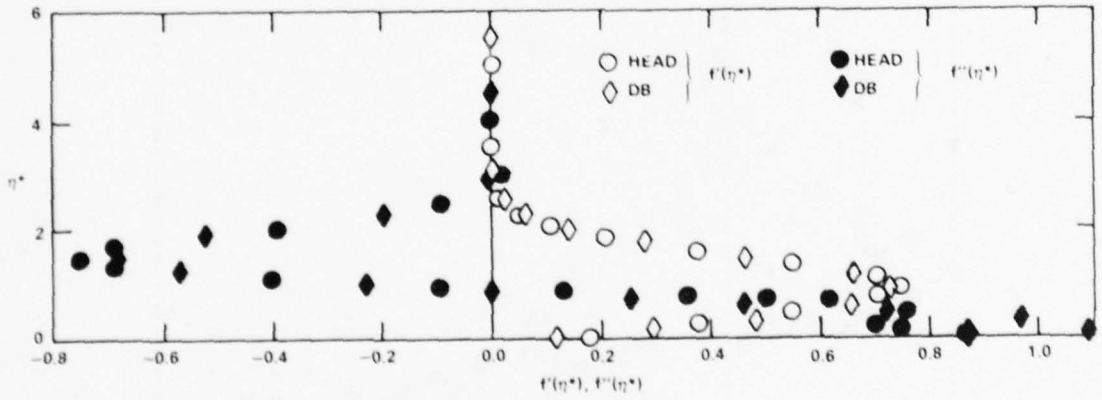


Figure 10b - Comparison of f'(η*) and f''(η*) Profiles

Figure 10 - Computed Velocity Profiles on Body B at s = 0.146

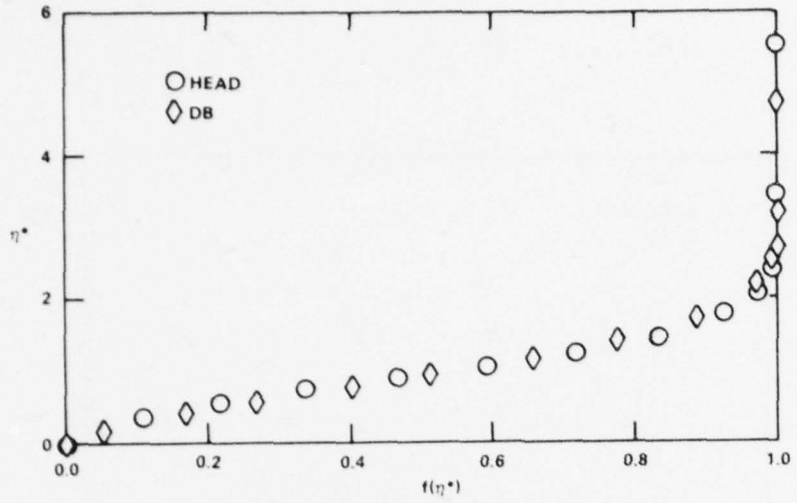


Figure 11a - Comparison of f(η*) Profiles

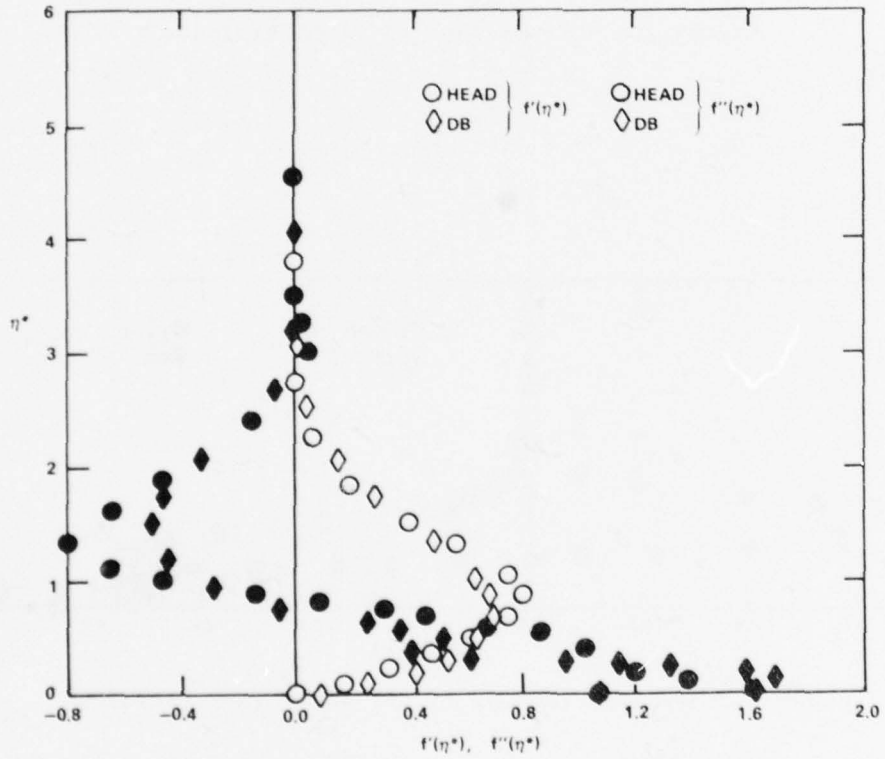


Figure 11b - Comparison of f'(η*) and f''(η*) Profiles

Figure 11 - Computed Velocity Profiles on Body B at s = 0.862

It is really remarkable how accurately Head's method predicts the velocity profiles $f(\eta)$. It is probably impossible to obtain much better predictions of a general velocity profile shape $f(\eta)$ and its derivatives by any other two-parameter method. This can be surmized by noting in Figures 10b and 11b that the second derivative $d^2f/d\eta_*^2$ has several local minimums and maximums across the boundary layer. Such variable functions would seem, in general, to require more than two parameters to accurately interpolate.

CONCLUDING REMARKS

The numerical results of the laminar boundary layer calculations using the Pohlhausen, Head, and DB methods shown in Figures 2 through 11 show that Head's method is very accurate and vastly superior to Pohlhausen's method. Since Head's method requires no more computer execution time and very little more computer core storage than Pohlhausen's method, there seems little reason for using the latter method when the former is available. However, with the availability of accurate finite-difference methods, such as the DB method, what advantages are there in using Head's method? If, for example, extremely high accuracy, essentially exact, laminar boundary layer computations are required, then one must use the finite difference methods. However, if only moderately accurate results, as obtained from Head's method shown here, are required, then Head's method has several advantages.

The first advantage is, of course, that Head's method requires only about one-fifth the computer execution time of the DB and similar finite-difference methods for the same level of accuracy. Although computer execution times are not large even for the finite-difference methods, the savings gained by using Head's method can be substantial for design where a parametric study of the boundary layer behavior on a large number of bodies must be made.

Secondly, there is a substantial difference in computer-core storage requirements between Head's method and finite-difference methods. In fact,

the small computer-core storage requirements of Head's method raise the possibility that with further refinements the method can probably be executed on a programmable hand calculator. However, for large computers, there is also an advantage of the small computer-core storage requirements of Head's method when one needs complete information of the boundary layer characteristics over the whole body, including boundary layer velocity profiles at each of many locations on the body. For finite-difference methods such a requirement would necessitate the storage of many vector arrays that describe the velocity profiles by points. But Head's method requires only the storage of the two arrays $\{a_i\}$ and $\{b_i\}$ of the values of the parameters a and b at the locations $\{s_i\}$ i = 0, 1, 2, 3,...I on the body. Furthermore, if the boundary layer characteristics for a certain body must be stored permanently for some future purpose it is very easy to simply punch the arrays $\{a_i\}$ and $\{b_i\}$ on a thin card-deck.

Since it has been found that Head's method is very accurate, it seems that the method can be extended to accurately predict unsteady boundary layers of low and moderate values of frequency (or low and moderate values of inverse time scale). Such an investigation is worthwhile and should be fairly easy to carry out. The unsteady versions of Equations (14) can be shown to have the form

$$A_{11} \frac{\partial a}{\partial t} + A_{12} \frac{\partial a}{\partial s} + B_{11} \frac{\partial b}{\partial t} + B_{12} \frac{\partial b}{\partial s} = C_1 \tag{26a}$$

$$A_{21} \frac{\partial a}{\partial t} + A_{22} \frac{\partial a}{\partial s} + B_{21} \frac{\partial b}{\partial t} + B_{22} \frac{\partial b}{\partial s} = C_2 \tag{26b}$$

where the coefficients A_{ij}, B_{ij} and C_i are functions of a, b, $r_0(s)$ and U(s,t). Equations (26) are in a natural form to which one can apply the method of characteristics. The method of characteristics is probably the most effective available numerical method for solving a pair of first order hyperbolic equations in two independent variables. In contrast to this, a finite difference time dependent calculation of the momentum Equation (1a) involves <u>three</u> independent variables and thus would require much larger storage arrays and much more computer time.

APPENDIX

THE COMPUTER PROGRAM

The FORTRAN IV computer program that implements Head's[5] two-parameter, two-dimensional and axisymmetric, incompressible laminar boundary layer calculation method is listed in this appendix. The program, called HEAD, requires, as input for an axisymmetric body, a discrete set of offsets $\{x_i, r_{0_i}\}$ where $\{x_i\}$ is the axial coordinate and $\{r_{0_i}\}$ is the radius, and the inviscid slip velocity $\{U_j\}$. The arc-length $\{s_j\}$ is computed by the program. For a two-dimensional body, the sets of arc-length values $\{s_j\}$ and inviscid slip velocities $\{U_j\}$ are the required input. Also for two-dimensional cases, the arc-length location s_0 of the front stagnation point must be provided.

The program HEAD is presently set up to run on the Burroughs B7000 computer directly in the interactive mode from a terminal. Thus all one has to do to run the program is to give the command to EXECUTE HEAD. The input must be typed in at the terminal, in free format, in the order in which the program calls for it. In each case, the program prompts the user as to the input that must be provided. For example: the first input card required by the program is the set of parameters NP, IPOTF, SCALE, IAN. The program first types a message requesting that these parameters be defined. NP is the number of input offsets defining the body ($1 \leq NP \leq 200$); IPOTF equals zero (0) for axisymmetric flow and one (1) for two-dimensional flow; SCALE is an arbitrary scale factor to rescale the input offsets at the discretion of the user (normally SCALE = 1.0); IAN equals zero (0) if the body and potential flow is provided by the user, equals one (1) if the user wishes to use a built-in potential flow for the sphere, and equals two (2) if the user wishes to use the built-in potential flow for a circular cylinder with no circulation (the front stagnation point is at 0.0). In the latter two cases one can set NP=1 and input for body offsets and potential flow velocities are not required.

The input variable "initial step size" for which the program will ask serves a dual role. The first role is precisely that of setting the first integration step at the front stagnation point. However, this is not important and the program is not sensitive to the value of initial step size since it will automatically be adjusted to maintain a certain integration accuracy. The second, more important, role of initial step size is that it determines the approximate interval along the body surface at which output is provided. Near separation, more output may be provided than far in front of separation. Thus, the density of the output of boundary layer characteristics along the surface of the body is controlled by the number put in under the heading "initial step size."

The HEAD program yields the following output. First, a summary of the geometric ($r_0(s)$) and inviscid flow ($U(s)$) data is printed under the columns of R, S, UB, A, RI which are the input values of body radius r_0, the axial locations of the input values of body radius, the input values of U, the arc-length values of s at which the input values of U are specified, and the body radius r_0 at the positions of the input values of U, respectively. In the version of the HEAD program given in this report, the input values of U correspond to locations along the body which are midway between consecutive locations at which the input values of r_0 are given. Following the above output, the boundary layer characteristics are printed. The numerical values of the quantities arc-length location s_j, step size Δs_j, body radius r_{0j}, first derivative with respect to s of U at s_j, second derivative with respect to s of U at s_j, H_j, $(C_f\sqrt{R_L})_j$, $(\delta_1\sqrt{R_L})_j$, and $(\theta\sqrt{R_L})_j$ are printed under the column headings x, HI, ROB, UB, C, D, DUBDS, DUBDS2, H, CT, DELST, and J, respectively. The program terminates either at the end of the body or just in front of separation with the message that the step size of the numerical integration procedure is too small ($0(10^{-7})$).

In this program no provisions for computing the boundary layer velocity profiles have been made. Given values of a and b (i.e., C and D in the program) one can easily write programs using formulas (7)-(10) to construct velocity profiles.

```
290 C*********PROGRAM HEAD******************
300        COMMON /BODY/ R(200),S(200),UB(200),A(200),RI(200),NP
320          COMMON/IPOT/IPOTF
400        DIMENSION Y(10),DY(10),PRMT(10)
600          EPS1=1.0E-4
700          EPS2=1.0E-5
800        WRITE(6,1)
900    1     FORMAT(1H1)
1100         TEND=100.0
1600 2     FORMAT(2I5,F10.5)
1720         WRITE(6,15)
1725         READ(5,/) NP,IPOTF,SCALE,IAN
1727         NP1=NP-1
1730    15   FORMAT(26H  INPUT NP,IPOTF,SCALE,IAN)
1740         IF(IAN.GT.0) GO TO 19
1750         WRITE(6,16)
1760    16   FORMAT(27H  INPUT THE BODY STATIONS S)
2000         READ(5,/) (S(I),I=1,NP)
2025         WRITE(6,17)
2050    17   FORMAT(27H  INPUT THE BODY OFFSETTS R)
2100         READ(5,/) (R(I),I=1,NP)
2200     5 FORMAT(6F10.5)
2300         IF(SCALE.EQ.0.0)  SCALE=1.0
2400         DO 10 I=1,NP
2500         S(I)=S(I)/SCALE
2600         R(I)=R(I)/SCALE
2700    10 CONTINUE
2950    13       CONTINUE
2960         WRITE(6,18)
2970    18   FORMAT(30H  INPUT THE SURFACE VELOCITIES)
3000         READ(5,/) (A(I),I=1,NP1)
3100     3 FORMAT(6F13.5)
3200         UB(1)=0.0
3300         DO 9 I=1,NP1
3400         J=I+1
3500         UB(J)=A(I)
3600         A(I)=0.0
3700     9 CONTINUE
3800         CALL ARC
3810            GOTO 100
3900         WRITE(6,6)
4000     6   FORMAT(/53H    R           S           UB          A       \
            \   RI)
4200         DO 7 I=1,NP
4300         WRITE(6,8) R(I),S(I),UB(I),A(I),RI(I)
4400     7 CONTINUE
```

```
4500      8 FORMAT(5E12.4)
4600  100   CALL SPLINE(A,UB,S,NP)
4700        CALL SPLINE(A,RI,R,NP)
4750     19  CONTINUE
4800        N=2
4850          ISTEP=1
4900          PRMT(1)=0.0
4910          IF(IPOTF.EQ.0) GO TO 14
4920          WRITE(6,20)
4925     20   FORMAT(38H  PUT IN LOCATION OF FRONT STAG. POINT)
4930          READ(5,/) SSTP
4940          PRMT(1)=SSTP
4950     14   CONTINUE
5000        PRMT(2)=TEND
5050          WRITE(6,21)
5100     21   FORMAT(27H  READ IN INITIAL STEP SIZE)
5150          READ(5,/) PRMT(3)
5200        PRMT(4)=EPS1
5300        PRMT(5)=EPS2
5400        PRMT(6)=0.97
5500        PRMT(7)=2.0
5600        PRMT(8)=0.1
5650           PRMT(10)=1.0
5700        IF(IPOTF.EQ.1)  GO TO 11
5800        C=0.3225
5900        D=0.1605
6000        GO TO 12
6100  11    CONTINUE
6200        C=0.445
6300        D=0.240
6400  12    CONTINUE
6500        Y(1)=C
6600        Y(2)=D
6700        WRITE(6,1)
6800        WRITE(6,4)
6900     4        FORMAT(43H  X,HI,ROB,UB,C,D,DUBDS,DUBDS2,H,CF,DELST,T/)
7100         CALL MTPC(Y,DY,PRMT,N,ISTEP,IAN)
7200        END
7300         SUBROUTINE MTPC(Y,DY,PRMT,N,ISTEP,IAN)
7400        DIMENSION Y(10),DY(10),Z(10,5),PRMT(10)
7500        NOUT=0
7600        IEND=1
7650           NZ=1
7800        HI=PRMT(3)
7850           X=PRMT(1)+HI
7900        HO=HI
8000        ISTEP=1
8500        NOUT=NOUT+1
8600     18   CONTINUE
8710          DO 1 I=1,N
8720          Z(I,3)=Y(I)
```

```
8735          Z(I,1)=Y(I)
8740       1  CONTINUE
8750          K=1
8760      14  CONTINUE
8765           CALL CALC(Y,DY,X,HI,N,IAN)
8770          E=0.
8780          DO 2 I=1,N
8790          Y(I)=Z(I,3)+HI*DY(I)
8800          E=ABS(Z(I,1)-Y(I))+E
8810      2   CONTINUE
8812           IF(K.EQ.1) GO TO 17
8815          IF(E.LT.1.0E-4) GO TO3
8820          IF(K.GT.10) GO TO 4
8825      17   CONTINUE
8830          K=K+1
8840          DO  13 I=1,N
8850          Z(I,1)=Y(I)
8860      13  CONTINUE
8880          GO TO 14
8890       4  CONTINUE
8900          HI=HI/2.
8910          HO=HI
8912           DO 21 I=1,N
8914           Y(I)=Z(I,3)
8916      21   CONTINUE
8920          IF(HI.LT.1.0E-5) WRITE(6,19)
8930      19  FORMAT(30H  NO CONVERGENCE AT FIRST STEP)
8940          K=1
8950          GO TO 14
8960       3  CONTINUE
8970          CALL CALC(Y,DY,X,HI,N,IAN)
8980          DO 20 I=1,N
8990          Z(I,1)=Y(I)
9000          Z(I,2)=DY(I)
9010      20  CONTINUE
9050           CALL OUTPUT(X,Z,PRNT,IEND,HO,HI,NOUT,NZ)
11100      10 CONTINUE
11300         C=HI/HO
11400         C1=C*C
11500         C2=1.-C1
11600         C3=HI*(1.+C)
11700         C4=3.+2./(C*C1)
11800         HI2=HI/2.
11900         DO 5 I=1,N
12000         Y(I)=C1*Z(I,3)+C2*Z(I,1)+C3*Z(I,2)
12100       5 CONTINUE
12300         X=X+HI
12400          CALL CALC(Y,DY,X,HI,N,IAN)
12500         E=0.0
12600         DO 6 I=1,N
12700         Z(I,5)=Y(I)
```

```
12800        Y(I)=Z(I,1)+HI2*(DY(I)+Z(I,2))
12900          AE=ABS(Z(I,5)-Y(I))/C4
13000          IF(AE.GT.E) E=AE
13100      6 CONTINUE
13300          IF(E.GT.PRMT(4)) GO TO 7
13400         CALL CALC(Y,DY,X,HI,N,IAN)
13500        DO 9 I=1,N
13600        Z(I,3)=Z(I,1)
13700        Z(I,4)=Z(I,2)
13800        Z(I,1)=Y(I)
13900        Z(I,2)=DY(I)
14000      9 CONTINUE
14100        ISTEP=2
14200        NOUT=NOUT+1
14300        CALL OUTPUT(X,Z,PRMT,IEND,HO,HI,NOUT,NZ)
14400        IF(IEND.EQ.-1) GO TO 15
14500           IF(E.LT.PRMT(5)) GO TO 8
14600        HO=HI
14700        GO TO 10
14800      7 CONTINUE
15000        IF(ISTEP.EQ.1) GO TO 11
15100        A=(PRMT(6)*PRMT(4)/E)**(1./3.)
15200        X=X-HI
15300        HI=A*HI
15400        IF(HI.LT.1.0E-7) GO TO 12
15500        GO TO 10
15600     11 CONTINUE
15800        X=X-2.*HI
15900        HI=HI/2.
16000        IF(HI.LT.1.0E-10) GO TO 12
16100          IF(E.LT.1.0E-10) E=PRMT(5)
16200      8      CONTINUE
16400        A=(PRMT(5)*PRMT(7)/E)**(1./3.)
16500        HO=HI
16600        HI=A*HI
16700        IF(HI.GT.PRMT(8)) HI=PRMT(8)
16800        GO TO 10
16900     12 CONTINUE
16920         PRMT(10)=-1.0
16940         PRMT(1)=0.0
16960         CALL OUTPUT(X,Z,PRMT,IEND,HO,HI,NOUT,NZ)
17000        WRITE(6,16) X,HI
17100 16    FORMAT(/27H     STEP SIZE TOO SMALL X=,E12.4,4H HI=,E12.4/)
17200        RETURN
17300     15 CONTINUE
17400        RETURN
17500        END
17600        SUBROUTINE ARC
17700        COMMON /BODY/ R(200),S(200),UB(200),A(200),RI(200),NP
17800        RI(1)=0.0
17900        A(1)=0.0
```

```
18000       IFLAT=2
18100       IFLAT1=1
18200       DO 1 I=2,NP
18300       IF(S(I)-S(I-1).EQ.0.0) GO TO 1
18400       IFLAT=I
18500       GO TO 2
18600     1 CONTINUE
18700     2 CONTINUE
18800       IF(IFLAT.LT.3) GO TO 3
18900       IFLAT1=IFLAT-1
19000       DO 4 I=2,IFLAT1
19100       A(I)=0.5*(R(I)+R(I-1))
19200       RI(I)=A(I)
19300     4 CONTINUE
19400     3 CONTINUE
19500       XI1=R(IFLAT1)
19600       DO 5 I=IFLAT,NP
19700       F=(S(I)-S(I-1))*SQRT(1.+((R(I)-R(I-1))/(S(I)-S(I-1)))**2)
19800       RI(I)=0.5*(R(I)+R(I-1))
19900       XI=XI1+F
20000       A(I)=0.5*(XI+XI1)
20100       XI1=XI
20200     5 CONTINUE
20300       RETURN
20400       END
20500       SUBROUTINE SPLINE(X,Y,C,N)
20600       DIMENSION X(200),Y(200),A(200),B(200),D(200),C(200)
20700       N1=N-1
20800       N2=N-2
20900       DO 1 I=1,N2
21000       J=I+1
21100       HI=X(J)-X(J-1)
21200       HI1=X(J+1)-X(J)
21300       D(I)=3.*(((Y(J+1)-Y(J))/HI1)-(Y(J)-Y(J-1))/HI) /(HI+HI1)
21400       B(I)=HI1/(HI+HI1)
21500       A(I)=1.-B(I)
21600       A(I)=0.5*A(I)
21700       B(I)=0.5*B(I)
21800     1 CONTINUE
21900       C(1)=1.0
22000       B(1)=B(1)/C(1)
22100       DO 2 I=2,N2
22200       C(I)=1.0-A(I)*B(I-1)
22300       B(I)=B(I)/C(I)
22400     2 CONTINUE
22500       A(1)=D(1)/C(1)
22600       DO 3 I=2,N2
22700       A(I)=(D(I)-A(I)*A(I-1))/C(I)
22800     3 CONTINUE
22900       N3=N2-1
23000       DO 4 I=1,N3
```

```
23100       J=N2-I
23200       C(J)=A(J)-B(J)*C(J+1)
23300     4 CONTINUE
23400       C(N1)=0.0
23500       C(N)=0.0
23600       DO 5 I=1,N3
23700       J=N2-I
23800       K=J+1
23900       C(K)=C(J)
24000     5 CONTINUE
24100       Y(1)=0.0
24200       RETURN
24300       END
24400        SUBROUTINE POTF(X,U,DU,DU2,RO,DRO,IAN)
24500       COMMON /BODY/ R(200),S(200),UB(200),A(200),RI(200),NP
24550        IF(IAN.GT.0) GO TO 3
24600       N=NP
24700       DO 1 I=1,NP
24800       IF(X.GE.A(I)) GO TO 1
24900       N=I
25000       GO TO 2
25100     1 CONTINUE
25200     2 CONTINUE
25300       HJ=A(N)-A(N-1)
25400       X3=(HJ**2)/6.
25500       X4=(A(N)-X)/HJ
25600       X5=(X-A(N-1))/HJ
25700       Y1=((A(N)-X)**2)/(2.*HJ)
25800       Y2=((X-A(N-1))**2)/(2.*HJ)
25900       X1=(A(N)-X)*Y1/3.
26000       X2=(X-A(N-1))*Y2/3.
26100       U=S(N-1)*X1+S(N)*X2+(UB(N-1)-S(N-1)*X3)*X4+(UB(N)-S(N)*X3)\
      \*X5
26200       RO=R(N-1)*X1+R(N)*X2+(RI(N-1)-R(N-1)*X3)*X4+(RI(N)-R(N)*X3)\
      \*X5
26300       DU=-S(N-1)*Y1+S(N)*Y2+(UB(N)-UB(N-1))/HJ-(S(N)-S(N-1))*HJ/6.
26400       DRO=-R(N-1)*Y1+R(N)*Y2+(RI(N)-RI(N-1))/HJ-(R(N)-R(N-1))*HJ\
      \/6.
26500       DU2=S(N-1)*X4+S(N)*X5
26510     3   CONTINUE
26520        IF(IAN.GT.1) GO TO 4
26522        RO=SIN(X)
26524        DRO=COS(X)
26526        U=1.5*RO
26528        DU=1.5*DRO
26530        DU2=-U
26531        RETURN
26532     4  CONTINUE
26534        RO=1.
26536        DRO=0.
26538        U=2.*SIN(X)
```

```
26540         DU=2.*COS(X)
26542         DU2=-U
26600       RETURN
26700       END
26800       SUBROUTINE OUTPUT(X,Z,PRMT,IEND,HO,HI,NOUT,NZ)
26900       DIMENSION Z(10,5),PRMT(10)
27000          COMMON /BL1/ J,SL,SM,H,DSTAR,HE,HT,UB,DUBDS,ROB,DROBDS\
     \,DUBDS2
27100          COMMON /BL2/ Y(4),RATIO,G,DTDS
27200       REAL J
27225         IF(NOUT.EQ.1) Y(1)=1.
27250          NZ1=500
27300         Y(4)=DTDS
27400         Y(2)=Y(1)
27500         Y(3)=HI
27600         Y(1)=-RATIO
27820         XRATIO=-RATIO
27830         IF(XRATIO.LT.0)GOTO 10
27840          T=SQRT(XRATIO)
28000       CF=2.*UB*SL/T
28100       DELST=H*T
28150         IF(PRMT(10).LT.0.0) GO TO 12
28200           IF(CF.GT.0.1) GO TO 12
28275         PRMT(3)=PRMT(3)/10.
28280         PRMT(10)=-1.0
28290    12   CONTINUE
28300         A=X-PRMT(1)
28400         IF(A.LT.PRMT(3)) GO TO 11
28500         PRMT(1)=X
28900 2     CONTINUE
28950         DUB2=DUBDS2
29000       WRITE(6,1) X,HI,ROB,UB,Z(1,1),Z(2,1),DUBDS,DUB2    ,H,CF\
     \,DELST,T
29100    1     FORMAT(12E10.3)
29150    11   CONTINUE
29200 8     IF(CF.LT.0.) IEND=-1
29300       IF(X.GT.PRMT(2)) IEND=-1
29400       RETURN
29450    10       WRITE(6,20)X
29470    20       FORMAT(1X,//,3X,10HCF NEGATIV  ,3X,3HX= ,F10.4)
29480             STOP
29500       END
29600        SUBROUTINE CALC(Y,DY,X,DX,N,IAN)
29650        DIMENSION Y(N),DY(N)
29700         COMMON /BL1/ J,SL,SM,H,DSTAR,HE,HT,UB,DUBDS,ROB,DROBDS\
     \,DUBDS2
29750         COMMON /BL2/ Z(4),RATIO,G,DTDS
29900       REAL J,L,M,MPD,MPC
29950         C=Y(1)
29960         D=Y(2)
30000       CALL POTF(X,UB,DUBDS,DUBDS2,ROB,DROBDS,IAN)
```

```
30100       J=.32832+(-.31897+.14931*C)*C+D*(.18760-.14947*C)
30200       L=1.747+3.5*C
30300       ESC=.20069+(-.054971+(-.18242+(.17580+(.043227+(-.055451+\
      \.0095503*
30400       $C)*C)*C)*C)*C)*C
30500       ESD=.060072+(.26478+(-.29465+(-.11611+(.15200-.029892*C)*C)\
      \*C)*C)*
30600       $C
30700       ESD2=-.11365+(.089165+(.12215+(-.13673+.031219*C)*C)*C)*C
30800       ESD3=.020669+(-.049679+(.040141-.010879*C)*C)*C
30900       E=ESC+(ESD+(ESD2+ESD3*D)*D)*D
31000       EPD=ESD+(2.*ESD2+3.*ESD3*D)*D
31100       EPCC=-.054971+(-.36484+(.52740+(.17291+(-.27726+.057302*C)\
      \*C)*C)*C)*C
31200       $)*C
31300       EPCD=.26478+(-.58930+(-.34833+(.60800-.14946*C)*C)*C)*C
31400       EPCD2=.089165+(.2443+(-.41019+.12488*C)*C)*C
31500       EPCD3=-.049679+(.080282-.032637*C)*C
31600       EPC=EPCC+(EPCD+(EPCD2+EPCD3*D)*D)*D
31700       GSC=.12733+(-.050050+(-.10089+(.13915-.035073*C)*C)*C)*C
31800       GSD=.049054+(.13403+(-.23383+.072643*C)*C)*C
31900       GSD2=-.059455+(.093916-.037702*C)*C
32000       G=GSC+(GSD+GSD2*D)*D
32100       GPD=GSD+2.*D*GSD2
32200       GPCC=-.050050+(-.20178+(.41745-.14029*C)*C)*C
32300       GPCD=.13403+(-.46766+.21793*C)*C
32400       GPCD2=.093916-.075404*C
32500       GPC=GPCC+(GPCD+GPCD2*D)*D
32600       M=-30.*D
32700       MPC=0.
32800       MPD=-30.
32900       HE=E/G
33000       G2=G*G
33100       SM=G2*M
33500 10    SL=G*L
33600         H=J/G
33700       AI=1.3713+(1.2124+(1.8754+(-1.8268+.60524*C)*C)*C)*C
33800       AI=AI+D*(-.83784+(-1.9728+(3.3493-1.2919*C)*C)*C)
33900       AI=AI+D*D*(.94797+(-1.5608+.69570*C)*C)
34000       DSTAR=G*AI
34100       GM=G*M
34200       HEPD=(EPD*G-GPD*E)/G2
34300       HEPC=(EPC*G-GPC*E)/G2
34400       SMPD=2.*GM*GPD+G2*MPD
34500       SMPC=2.*GM*GPC
34550        AO2D=0.
34600       IF(ABS(DUBDS).LE.1.0E-2)   GO TO 5
34660         IF(IPOTF.EQ.0)AO2D=SM*UB*DROBDS/(DUBDS*ROB)
34700       DTDS=2.*(SL+SM*(2.+H)+AO2D)/UB
34800       RATIO=SM/DUBDS
34900       GO TO 6
```

```
35000 5      CONTINUE
35050 C        PRIVIOUS VALUE IN Z(4)
35060          TSIM=Z(1)
35070          TSI=Z(2)
35080          DTDSI=Z(4)
35100        TS=TSIM+(2.*(TSI-TSIM)/Z(3)-DTDSI)*(Z(3)+DX)
35200        TS=TS+(DTDSI/Z(3)-(TSI-TSIM)/(Z(3)*Z(3)))*(DX+Z(3))**2
35300        RATIO=-TS
35350          IF(IPOTF.EQ.0)A02D=-TS*UB*DROBDS/ROB
35400        DTDS=2.*(SL+SM*(2.+H)+A02D)/UB
35500      6 CONTINUE
35600        R1=-2.*DUBDS*(SL+SM*(2.+H))/UB
35650         IF(IPOTF.EQ.0)A02D=SM*DROBDS/ROB
35700        R1=R1-2.*A02D+RATIO*DUBDS2
35895 210       R2=HE*(SL+SM*(H-1))
36000        DS=SMPC*HEPD-SMPD*HEPC
36050          R2=-(2.*DSTAR-R2)/(RATIO*UB)
36100         DY(1)=(R1*HEPD-R2*SMPD)/DS
36200         DY(2)=(R2*SMPC-R1*HEPC)/DS
36300        RETURN
36400        END
#
```

REFERENCES

1. Blottner, F.G., "Investigation of Some Finite-Difference Techniques for Solving the Boundary Layer Equations," Computational Methods Applied Mechanical Engineering, Vol. 6, pp. 1-30 (1975).

2. Cebeci, T. and P. Bradshaw, "Momentum Transfer in Boundary Layers," McGraw-Hill Book Co., New York (1977).

3. Rosenhead, L. (ed.), "Laminar Boundary Layers," Oxford U. Press (1963).

4. Waltz, A., "Boundary Layers of Flow and Temperature," (translation by H.J. Oser), The M.I.T. Press, Cambridge, Mass. (1969).

5. Head, M.R., "An Approximate Method of Calculating the Laminar Boundary Layer in Two-Dimensional Incompressible Flow," Reports and Memoranda 3123, Aeronautical Research Council of Great Britain (1957).

6. Chang, M.S. and P.C. Pien, "Hydrodynamic Forces on a Body Moving Beneath a Free Surface," Proceedings of the First International Conference on Numerical Ship Hydrodynamics, ed. by J. Schot and N. Salvesen, Gaithersburg, Md., pp. 539-559 (1975).

INITIAL DISTRIBUTION

Copies		Copies	
1	DARPA	1	SEA 32
	1 P. Selwyn	1	SEA 32R, T.E. Peirce
		1	SEA 321
1	U.S. ARMY TRAS R&D	1	SEA 52
	Marine Trans. Div.	1	SEA 521
		1	SEA 524
3	CHONR	1	PMS 393
	1 Code 468, H.M. Fitzpatrick	2	PMS 395, J. Granet, R. Manning
	1 Code 438, R.D. Cooper		
	1 Code 474, N. Perrone	5	NOSC, San Diego
			1 A. Fabula
1	ONR/Boston		1 J. Hoyt
			1 Dr. M. Reischman
1	ONR/Chicago		1 T.G. Lang
			1 G. Donahue
1	ONR/New York		
		5	NUSC, Newport
1	ONR/Pasadena		1 B. Myers
			1 D. Goodrich
1	ONR/San Francisco		1 R. Nadolink
			1 F. White
3	NRL		1 Library
	1 Dr. Hansen		
	1 O.M. Griffin	1	NUSC, NLONLAB
	1 R.A. Skop		1 H. Schloemer
1	NORDA	1	NSWC, Dahlgren/Lib
3	USNA	2	NSWC, White Oak
	1 Dr. B. Johnson		1 Dawson
	1 Nav. Sys. Engr. Dept		1 Library
	1 Tech Library		
		1	NAVSEC, NORVA/660.03 Blount
3	NAVPGSCOL		
	1 Library	1	NCEL/Code 131
	1 T. Sarpkaya		
	1 J. Miller	1	NAVFAC/Code 032C
1	NADC	1	NAVSHIPYD PTSMH/Lib
1	NCSC/712, D. Humphreys	1	NAVSHIPYD PHILA/Lib
17	NAVSEA	1	NAVSHIPYD NORVA/Lib
	1 SEA 03		
	2 SEA 03D	1	NAVSHIPYD CHASN/Lib
	2 SEA 05H, A.R. Paladino		
	1 SEA 05R	1	NAVSHIPYD LBEACH/Lib
	2 SEA 312		

Copies	
2	NAVSHIPYD MARE 1 Library 1 Code 250
1	NAVSHIPYD PUGET/Lib
1	NAVSHIPYD PEARL/Code 202.32
12	DTIC
1	AFOSR/NAM
1	AFFOL/FYS, J. Olsen
1	National Bureau of Standards 1 P.S. Klebanoff
2	MARAD 1 Div of Ship R&D 1 Library
1	NASA, HQS/Lib
1	NSF/Eng. Lib
1	LC/SCI & TECH
1	DOT/Lib. TAD-491.1
2	MMA 1 Nat'l Maritime Res. Cen. 1 Library
1	Cal. Inst. Technology 1 A. Costa
1	Cal. Inst. Technology Jet Propulsion Lab., Pasadena, Cal. 1 L.M. Mack
1	Case Western University 1 E. Reshotko
2	Univ. of California, Los Angeles 1 S.J. Barker 1 A.R. Wazzan

Copies	
1	Illinois Inst. of Tech. 1 M.V. Morkovin
1	Univ. of Iowa 1 L. Landweber
1	Penn State University 1 J.J. Eisenhuth
2	Penn State Univ./ARL 1 B. Parkin 1 G. Lauchle
1	Princeton University 1 S.I. Cheng
2	Massachusetts Inst. of Tech. 1 S.A. Orszag 1 P. Leehey
1	Univ. of Minn., SAFAHL 1 R. Arndt
1	Virginia Polytechnic Inst. and State Univ. 1 W.S. Saric 1 D.P. Telionis
1	Univ. of Rhode Island 1 F.M. White
1	SIT Davidson Lab., 1 J. Breslin
1	Adv. Tech. Center, Inc., Texas 1 C.S. Wells
1	DCW Industries, Inc. 1 D.C. Wilcox
1	Lockheed Aircraft, Sunnyvale 1 R. Waid
2	McDonnell Douglas Corp. 1 Lib 1 T. Cebeci

Copies	
1	Physical Dynamics, Inc., Torrance, Cal. 1 W. Haigh
2	Rand Corp., Santa Monica, Cal. 1 C. Cazley 1 J. Aroesty
3	Rockwell Intern. Autonetics Group 1 D.B. Moody 1 C. Jennings 1 B. Carmichael
1	Westinghouse Electric Corp. Annapolis, Md. 1 Library
1	Westinghouse Research Corp. Pittsburgh, Pa. 1 F.R. Goldschmied

CENTER DISTRIBUTION

Copies	Code	Name
1	15	
1	1504	V.J. Monacella
1	1522	J.H. Robinson
1	1524	
1	1532	G. Dobay
1	154	W.B. Morgan
1	1552	J.H. McCarthy
1	1552	N. Groves
1	1552	T.T. Huang
30	1552	C. von Kerczek
1	184	J.W. Schot
1	1843	C. Dawson
1	19	M.M. Sevik
1	1901	M. Strasberg
1	1902	G. Maidanik
1	1903	G. Chertock
1	1905	R.E. Biancardi
1	192	
1	193	R.C. Olson
1	1933	J.P. Lee
1	1942	J.T. Shen
1	1942	R.E. Bowers
1	1942	M.J. Casarella
1	1946	J.R. Spina
10	5211.1	Reports Distribution
1	522.1	Unclassified Lib (C)
1	522.2	Unclassified Lib (A)

DTNSRDC ISSUES THREE TYPES OF REPORTS

1. DTNSRDC REPORTS, A FORMAL SERIES, CONTAIN INFORMATION OF PERMANENT TECHNICAL VALUE. THEY CARRY A CONSECUTIVE NUMERICAL IDENTIFICATION REGARDLESS OF THEIR CLASSIFICATION OR THE ORIGINATING DEPARTMENT.

2. DEPARTMENTAL REPORTS, A SEMIFORMAL SERIES, CONTAIN INFORMATION OF A PRELIMINARY, TEMPORARY, OR PROPRIETARY NATURE OR OF LIMITED INTEREST OR SIGNIFICANCE. THEY CARRY A DEPARTMENTAL ALPHANUMERICAL IDENTIFICATION.

3. TECHNICAL MEMORANDA, AN INFORMAL SERIES, CONTAIN TECHNICAL DOCUMENTATION OF LIMITED USE AND INTEREST. THEY ARE PRIMARILY WORKING PAPERS INTENDED FOR INTERNAL USE. THEY CARRY AN IDENTIFYING NUMBER WHICH INDICATES THEIR TYPE AND THE NUMERICAL CODE OF THE ORIGINATING DEPARTMENT. ANY DISTRIBUTION OUTSIDE DTNSRDC MUST BE APPROVED BY THE HEAD OF THE ORIGINATING DEPARTMENT ON A CASE-BY-CASE BASIS.

1 O.M. Griffin
1 R.A. Skop

RDA

NA
1 Dr. B. Johnson
1 Nav. Sys. Engr. Dept
1 Tech Library

VPGSCOL
1 Library
1 T. Sarpkaya
1 J. Miller

DC

SC/712, D. Humphreys

VSEA
1 SEA 03
2 SEA 03D
2 SEA 05H, A.R. Paladino
1 SEA 05R
2 SEA 312

1 NUSC, NLONLAB
1 H. Schloemer

1 NSWC, Dahlgren/Lib

2 NSWC, White Oak
1 Dawson
1 Library

1 NAVSEC, NORVA/660.03 Blount

1 NCEL/Code 131

1 NAVFAC/Code 032C

1 NAVSHIPYD PTSMH/Lib

1 NAVSHIPYD PHILA/Lib

1 NAVSHIPYD NORVA/Lib

1 NAVSHIPYD CHASN/Lib

1 NAVSHIPYD LBEACH/Lib

PRECEDING PAGE BLANK