NASA TECHNICAL NOTE

NASA TN D-7630

NASA TN D-7630

SUBSONIC ANNULAR WING THEORY WITH APPLICATION TO FLOW ABOUT NACELLES

by Michael J. Mann

Langley Research Center
Hampton, Va. 23665

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION • WASHINGTON, D. C. • SEPTEMBER 1974

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
NASA TN D-7630		

4. Title and Subtitle	5. Report Date
SUBSONIC ANNULAR WING THEORY WITH APPLICATION TO FLOW ABOUT NACELLES	September 1974
	6. Performing Organization Code

7. Author(s)	8. Performing Organization Report No.
Michael J. Mann	L-9274
	10. Work Unit No.
9. Performing Organization Name and Address	760-64-60-01
NASA Langley Research Center Hampton, Va. 23665	11. Contract or Grant No.
	13. Type of Report and Period Covered
12. Sponsoring Agency Name and Address	Technical Note
National Aeronautics and Space Administration Washington, D.C. 20646	14. Sponsoring Agency Code

15. Supplementary Notes

16. Abstract

A method has recently been developed for calculating the flow over a subsonic nacelle at zero angle of attack. The method makes use of annular wing theory and boundary-layer theory and has shown good agreement with both experimental data and more complex theoretical solutions. The method permits variation of the mass flow by changing the size of a center body.

17. Key Words (Suggested by Author(s))	18. Distribution Statement
Nacelle design Annular wing Ring wing Propeller shroud	Unclassified – Unlimited STAR Category 01

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price*
Unclassified	Unclassified	91	$4.00

*For sale by the National Technical Information Service, Springfield, Virginia 22151

SUBSONIC ANNULAR WING THEORY WITH APPLICATION TO FLOW ABOUT NACELLES

By Michael J. Mann
Langley Research Center

SUMMARY

Jet engine nacelles have generally been designed with the assumption that the intake and nozzle can be treated separately. However, advancements in engine development for subsonic cruise airplanes have tended toward high-bypass-ratio fan engines. This type of engine usually has a short fan nacelle which necessitates analyzing the complete nacelle. These nacelles can be treated as annular wings on which the circulation developed determines both the internal and external flow.

A method has recently been developed for calculating the flow over such a nacelle at zero angle of attack and at subsonic Mach numbers. The method makes use of annular wing theory and boundary-layer theory and has shown good agreement with both experimental data and more complex theoretical solutions. The method permits variation of the mass flow by changing the size of a center body.

INTRODUCTION

Jet engines have generally been designed with the assumption that the inlet and nozzle can be designed separately. Advancements in engine technology have recently tended toward high-bypass-ratio fan engines with short fan nacelles. For these nacelles, the complete nacelle must be examined. A complete nacelle can be viewed as an annular wing, the thickness and camber of which determine the internal and external flow.

Several methods have been developed for calculating the flow over nacelles or annular wings. The Douglas-Neumann method (ref. 1) has been used to predict accurately the pressures on the surface of a nacelle inlet. Trulin and Iversen (ref. 2), Geissler (ref. 3), and Young (ref. 4) have recently published methods which use surface singularities and show good agreement between theory and experiment for complete nacelles. Geissler's method includes the effects of angle of attack.

Young (ref. 5) also examined the accuracy of a method which distributes the singularities along a mean cylinder of constant diameter. However, Young did not present the details of the theory. Agreement between the theory and experiment is good at low Mach numbers but is inadequate at the higher Mach numbers.

Mascitti (ref. 6) has developed a rapid numerical technique for calculating the subsonic flow over planar and axisymmetric profiles at zero angle of attack. The method solves the exact incompressible potential-flow equations written in terms of the stream function.

Belotserkovskii (ref. 7) and Weissinger (ref. 8) have developed much of the theory of annular wing flow. Their application of the theory has been the determination of overall force and moment characteristics rather than a study of surface pressure distribution. However, much of their theoretical work has been useful in the present study.

Hess (ref. 9) discusses some of the early solutions for axisymmetric flow which made use of vortex rings on the surface. Keith and others (ref. 10), and Grossman and Moretti (ref. 11) have developed numerical methods for solving the exact inviscid flow over two-dimensional and axisymmetric nacelles at subsonic and transonic speeds. Reference 10 uses a stream-tube curvature relaxation technique, and reference 11 uses a time-dependent technique. However, the technique of reference 11 was only successful for subsonic flow. To the author's knowledge, only the present paper and references 3 and 10 have included the effects of the boundary layer.

The purpose of the present study is twofold. First, an inviscid theory, which uses vortex rings to represent the camber and a combination of source rings and vortex rings to represent thickness has been developed. Three-dimensional sources are distributed along the center line to control the mass flow through the wing. This method appears to have some advantage over each of the other methods in terms of either simplicity, a reduction of computer core storage or central processor time requirements, ease of extension to more complex flows, accuracy, or quantities computed. Secondly, an analysis of the importance of including the displacement effect of the boundary layer in the calculations has been made. A boundary-layer iteration procedure has been included in the computer program.

Results of the present theory have been compared with experimental data obtained in tests on four nacelles and good agreement has been found over a range of mass-flow ratios and Mach numbers.

SYMBOLS

A internal-flow cross-sectional area, m^2 $\left(ft^2\right)$

c wing chord, m (ft)

C_p pressure coefficient, $\frac{p - p_\infty}{q_\infty}$

d	intake diameter equals highlight diameter less twice inlet lip radius, m (ft)
D	average of minimum inner diameter and maximum outer diameter of wing; when subscripted, refers to diameter of wing, m (ft)
$\hat{e}_t, \hat{e}_r, \hat{e}_x$	unit vectors in meridional, r- and x-directions, respectively
E	complete elliptic integral of second kind, $\int_0^{\pi/2} \sqrt{1 - k^2 \sin^2 \upsilon}\, d\upsilon$, dimensionless
$\bar{g}_q(\xi)$	dimensionless correction term for γ_t (see eq. (32))
H	shape factor, δ^*/θ
K	complete elliptic integral of first kind, $\int_0^{\pi/2} \frac{1}{\sqrt{1 - k^2 \sin^2 \upsilon}}\, d\upsilon$, dimensionless
k	modulus of complete elliptic integrals (see eq. (11)), dimensionless
k'	complementary modulus, $\sqrt{1 - k^2}$
M_∞	free-stream Mach number
m	mass flow through annular wing, kg/sec (slugs/sec); iteration number or summation index
N	number of chordwise divisions for numerical solution
p	static pressure, N/m^2 (lbf/ft^2)
$Q(\xi)$	thickness function defined by equation (29), dimensionless
$q(\xi)$	source strength per unit axial distance, m/sec (ft/sec)
r	radial coordinate attached to wing, m (ft)
q_∞	free-stream dynamic pressure, $\rho_\infty V_\infty^2/2$, N/m^2 (lbf/ft^2)
$t(\xi)$	wing thickness, $r_{outer} - r_{inner}$, m (ft)

$U_o(\xi - \xi_{o,1})$ dimensionless kernel defined by equation (31) for $\rho_o = 1$

$V_q(\xi - \xi_{o,1})$ dimensionless kernel defined by equation (30) for $\rho_o = 1$

V total velocity, free stream plus perturbation, m/sec (ft/sec)

V_∞ free-stream velocity, m/sec (ft/sec)

$u(\xi_o,\rho_o)$, $v(\xi_o,\rho_o)$ perturbation velocity components in x- and r-directions, respectively, induced at ξ_o,ρ_o, m/sec (ft/sec)

$\hat{u}(\xi - \xi_o,\rho_o)$, $\hat{v}(\xi - \xi_o,\rho_o)$ dimensionless perturbation velocity influence functions for x- and r-directions, respectively; these influence functions are for velocity induced at ξ_o,ρ_o by a ring singularity at $\xi,\rho = 1$ or by a point singularity at $\xi,\rho = 0$

X inlet length, m (ft)

x, y, z, r coordinates attached to wing (see fig. 1 and app. A for positive directions), m (ft)

α_o slope of mean camber surface with respect to x-axis (wing centerline), radians

α_t slope of symmetrical thickness profile with respect to x-axis, $(dt/dx)/2$ radians

$\beta = \sqrt{1 - M_\infty^2}$

Γ_o dimensionless bound vortex strength of camber solution, $\gamma_o \Delta\xi$

γ_o dimensionless bound vortex sheet strength of camber solution, $\gamma_+ / V_\infty \alpha_o$

γ_+ bound vortex sheet strength of camber solution, m/sec (ft/sec)

γ_t vortex sheet strength for thickness, m/sec (ft/sec)

γ ratio of specific heats

$\Delta(\ )$ inner minus outer of ( ); $\Delta\xi$ = dimensionless panel length

δ^* boundary-layer displacement thickness, m (ft)

ϵ small positive number

θ momentum thickness, m (ft)

λ aspect ratio, D/c

μ mass-flow ratio (see eq. (41)

ξ dimensionless distance along axis, $\frac{x}{D/2}$

ξ' dummy variable form of ξ

ρ dimensionless radius, $\frac{r}{D/2}$; gas density, kg/m^3 $(slugs/ft^3)$

υ dummy variable

ψ digamma function (see app. B)

Superscripts:

$(\ )'$ incompressible quantity except ξ' and k'

$(\hat{\ })$ unit vector; velocity influence function

$(\bar{\ })$ vector, except $\bar{g}_q(\xi)$

$*$ critical Mach number condition; boundary-layer displacement thickness

Subscripts:

$+$ dimensional bound vortex sheet strength of camber solution

∞ free stream

b center body

exp	experimental
h	highlight station (most forward point on inlet lip, $x = c/2$ in fig. 1)
i	location of vortex and source singularities
j	location of point at which velocity is induced
LE	leading edge
m	iteration number (see eq. (44))
max	maximum
min	minimum
o	point at which velocity is induced in ξ_o,ρ_o; effect due to camber only without angle-of-attack or pitching-motion effects in α_o, γ_o, and Γ_o; zeroth order in Fourier expansions in U_o; also see equation (44)
q	due to source ring of thickness solution
r	radial direction
TE	trailing edge
t	thickness solution; tangential
th	theory
w	wing surface
x	wing axial direction
γ	due to vortex ring
ξ_o	axial location
ρ_o	radial location

THEORETICAL SOLUTION

The subsonic flow over an axisymmetric nacelle or annular wing with a center body of revolution will be determined by use of small perturbation theory. This problem is analogous in many ways to airfoil theory and biplane theory. Linearization of the problem permits computation of the compressibility effects by transforming the wing coordinates x and r into an equivalent incompressible wing x' and r' by using the Göthert transformation

$$x' = x \tag{1}$$

$$r' = \beta r \tag{2}$$

where $\beta = \sqrt{1 - M_\infty^2}$. The incompressible flow over this equivalent wing is then computed, and the compressible perturbation velocities u and v in the x- and r-directions, respectively, are found by

$$u = u'/\beta^2 \tag{3}$$

$$v = v'/\beta \tag{4}$$

while the free-stream velocity V_∞ remains unchanged. Compressible flow equations can then be used to determine the pressure distribution. (All primes are dropped from coordinates and velocities from this point on, and these variables are understood to be incompressible, unless otherwise stated.)

Also as a consequence of the linearization of the equations of motion and the boundary conditions, the problem can be broken up into wing camber, wing thickness, and center-body portions. Furthermore, each of these problems can be solved by superimposing elementary singular solutions of Laplace's equation, such as vortexes and sources. The linearized boundary condition on the wing surface at the point ξ_o,ρ_o is

$$\frac{v\left(\xi_o,\rho_o\right)}{V_\infty} = -\left(\frac{dr}{dx}\right)_{\xi_o,\rho_o} \tag{5}$$

where dr/dx is the local slope of the wing and ξ_o and ρ_o are dimensionless x- and r-coordinates. The minus sign arises because the positive x-direction has been chosen to be upstream (fig. 1). This boundary condition is written in two parts, one part for the camber and center body and one part for the thickness

$$\frac{v_\gamma(\xi_o,\rho_o)}{V_\infty} + \frac{v_b(\xi_o,\rho_o)}{V_\infty} = -\alpha_o(\xi_o,\rho_o) \tag{6}$$

$$\frac{v_q(\xi_o,\rho_o)}{V_\infty} + \frac{v_{t,\gamma}(\xi_o,\rho_o)}{V_\infty} = \mp\alpha_t(\xi_o,\rho_o) \tag{7}$$

where α_o and α_t are the slopes of the mean camber line and the symmetrical thickness, respectively. The subscripts γ, b, q, and t,γ refer to camber vorticity, center body, thickness source, and thickness vorticity, respectively. Two terms arise in the thickness equation because sources alone will not produce a symmetrically thick wing. This point is discussed further when the thickness solution is outlined. When plus and minus signs are used, the upper and lower signs refer to the outer and inner wing surfaces, respectively.

For design purposes it is convenient to separate the problem into its camber and thickness components. Thus, all singularities (except the center body) are placed on a cylinder of constant diameter D as shown in figure 1 and the boundary conditions are satisfied at control points on this same cylinder (counterpart of the planar wing approximation). A correction for approximating the surface velocities by velocities on this cylinder, which is called the Riegels factor, is discussed subsequently.

The strengths of the singularities are found by solving the boundary conditions given by equations (6) and (7). The solution of these boundary conditions involves evaluating improper integrals. The technique used is simply to replace the integrals by summations, thereby replacing the continuous vortex and source sheets and the source line by distributions of singularities. In the camber solution, proper placement of the vortex

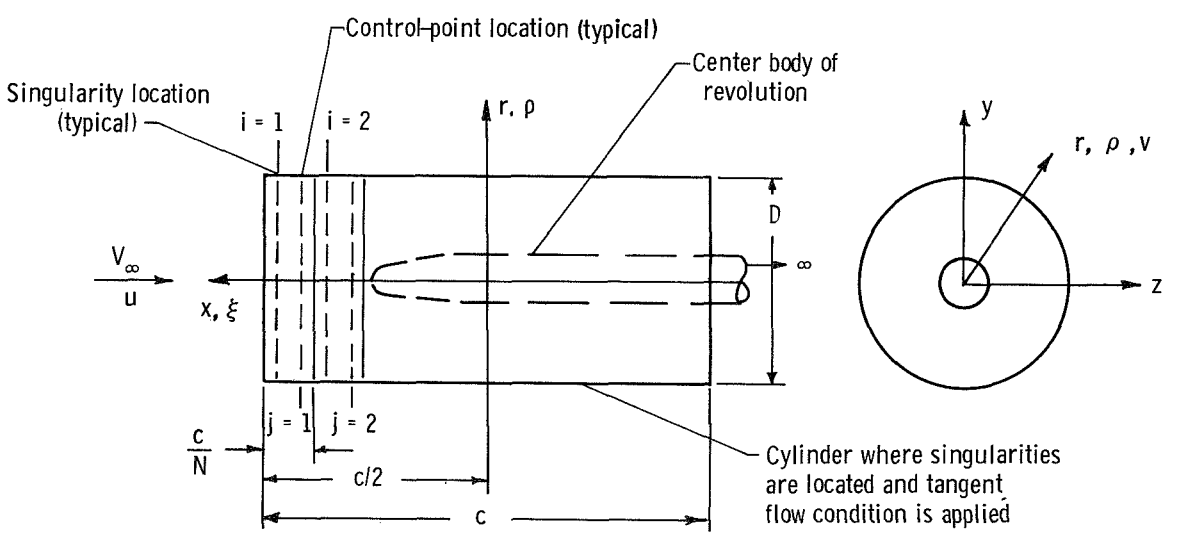


Figure 1.- Coordinates, geometry, and singularity and control-point location.

singularities and control points will satisfy both the principal value of the integral and the Kutta condition (ref. 7, p. 106). The same lattice arrangement is used for the sources as for the vortexes.

Once the velocity expressions are substituted into equations (6) and (7), a set of linear equations results with unknown singularity strengths. The vortex strengths are found by solving these equations by the method of successive orthogonalization. The source strengths for the wing thickness and the center body are proportional to the local body slopes.

In the following sections the major equations are given. The camber—center-body solution, the thickness solution, the complete wing solution, and the boundary-layer calculation are discussed.

Camber—Center-Body Solution

The flow to be calculated is subsonic flow over an axisymmetric mean camber line at zero angle of attack. Along the center line of this infinitesimally thin wing is mounted a body of revolution. The mass flow inside the camber surface is varied by varying the size of the center body.

In order to obtain a unique solution for the flow over a camber surface, the basic flow conservation equations must be satisfied, boundary conditions on the body and at infinity must be satisfied, and the circulation specified (accomplished by the Kutta condition). As mentioned previously, these conditions can be met by distributing a continuous vortex sheet along a cylinder the length of which is equal to the wing chord. The influence of the center body can be found by use of a source line along the wing axis.

In this study the influence of the wing on the center body is ignored. There are several reasons for ignoring this interference. First, owing to the wide variation in internal structure, such as stings, struts, rakes, and so forth, no attempt was made to simulate any particular internal structural geometry. Secondly, inclusion of the mutual interference would greatly increase the storage requirements of the computer program (see app. A), thereby losing one of the attractive features of this method. Furthermore, the primary objective has been to obtain an accurate solution for the external flow. Thus, a simple body of revolution on the wing axis was selected to represent the internal parts. The size of this body is selected to either match the desired mass flow or match the internal blockage (and compute the mass flow). Both methods have shown good success as is demonstrated in the next section.

The vortex sheet strength is found by first substituting the expressions for the vortex velocities (refs. 7 and 12) and source velocities (refs. 12 and 13) into the boundary condition, equation (6). When this is done, the result is

$$\frac{1}{2\pi}\int_{-1/\lambda}^{1/\lambda} \alpha_o(\xi)\gamma_o(\xi)\hat{v}_\gamma\left(\xi - \xi_o, 1\right) d\xi = -\alpha_o\left(\xi_o\right) - \int_{\xi_{b,TE}}^{\xi_{b,LE}} \rho_b(\xi)\left(\frac{d\rho_b(\xi)}{d\xi}\right)\hat{v}_b\left(\xi - \xi_o, 1\right) d\xi \quad (8)$$

where the influence functions are defined by

$$\hat{v}_\gamma\left(\xi - \xi_o, \rho_o\right) = -\frac{\left(\xi - \xi_o\right)k}{4\rho_o^{3/2}}\left(\frac{2 - k^2}{1 - k^2}E - 2K\right) \quad (9)$$

$$\hat{v}_b\left(\xi - \xi_o, \rho_o\right) = \frac{\rho_o}{2\left[\left(\xi - \xi_o\right)^2 + \rho_o^2\right]^{3/2}} \quad (10)$$

and K and E are complete elliptic integrals of the first and second kinds with modulus

$$k^2 = \frac{4\rho_o}{\left(\xi - \xi_o\right)^2 + \left(1 + \rho_o\right)^2} \quad (11)$$

The variables ξ and ρ are the dimensionless x- and r-locations of singularities ($\rho = 1$), and ξ_o, ρ_o is the point at which the velocity is calculated $\left(\rho_o = 1\right)$. The variable λ is the wing aspect ratio; $\gamma_o(\xi)$ is the vortex strength and is related to the usual definition of vortex sheet strength by

$$\gamma_+ = V_\infty \alpha_o \gamma_o$$

where γ_+ is proportional to $\Delta p/\rho V_\infty$.

The subscript o on γ_o and α_o is used in reference 7 to indicate camber effects without angle of attack or pitching motion. For the sake of consistency this notation is retained. When angle-of-attack effects are included, the vortex sheet is no longer of constant strength around its circumference and a trailing vortex system must also be included (ref. 7). The camber effect alone will give a nonzero radial force; however, owing to symmetry the net lift is zero.

Equation (8) is solved numerically by replacing the continuous vortex sheet by N equally spaced vortex rings. The source line along the center-body axis is replaced by three-dimensional point sources at the same axial locations as the vortex rings.[1] The

[1]Center-body length can be less than the wing chord c. The case of a semi-infinite center body can be simulated by a constant body radius to downstream infinity because the local source strength is zero where $d\rho_b/d\xi = 0$.

singularity location is i and the control-point location is j. When the quantity $\gamma_o \, d\xi$ is replaced by

$$\Gamma_{oi} = \gamma_{oi} \, \Delta\xi$$

equation (8) becomes

$$\sum_{i=1}^{N} \alpha_{oi} \Gamma_{oi} \hat{v}_\gamma \left(\xi_i - \xi_{oj}, 1 \right) = -2\pi \left[\alpha_{oj} + \sum_{i=1}^{N} \rho_{bi} \left(\frac{d\rho_b}{d\xi} \right)_i \hat{v}_b \left(\xi_i - \xi_{oj}, 1 \right) \Delta\xi \right] \tag{12}$$

Because the vortex rings and control points are located on a constant-diameter right circular cylinder of diameter D, then $\rho_i = \rho_{oj} = 1$ in equation (12).[2] Each circular panel has a chord length of c/N. The vortex ring is located at one-fourth of this length from the panel leading edge and the control point is located at three-fourths of this length from the panel leading edge (fig. 1).

As previously mentioned, the influence of the wing on the body is neglected. Thus the body source strengths are given by the body slopes and everything on the right-hand side of equation (12) is known. Equation (12) is then a set of N simultaneous linear equations with N unknown vortex strengths Γ_{oi} (see definition of CAM in app. A for discussion of case when $\alpha_o = 0$).

The set of equations resulting from equation (12) was solved for the unknown vortex strengths Γ_{oi} by the successive orthogonalization procedure outlined in reference 1. This procedure gave the same results as an alternate matrix inversion technique but has a great advantage on computer storage requirements. The method of reference 1 requires storage for a single N-dimensional array, the elements of which are $\hat{v}_\gamma \left(\xi_i - \xi_{oj}, 1 \right) \alpha_{oi}$ as opposed to other matrix techniques which require N- by N-array storage.

Once the vortex and source strengths are computed, the perturbation velocities can be found at each location ξ_{oj} on the cylinder $\rho_{oj} = 1$ by use of the following expressions

$$\frac{u_\gamma \left(\xi_{oj}, 1 \right)}{V_\infty} = \sum_{i=1}^{N} \frac{\alpha_{oi} \Gamma_{oi}}{2\pi} \hat{u}_\gamma \left(\xi_i - \xi_{oj}, 1 \right) \pm \frac{\alpha_{oj}}{2} \frac{\Gamma_{oj}}{\Delta\xi} \tag{13}$$

$$\frac{v_\gamma \left(\xi_{oj}, 1 \right)}{V_\infty} = \sum_{i=1}^{N} \frac{\alpha_{oi} \Gamma_{oi}}{2\pi} \hat{v}_\gamma \left(\xi_i - \xi_{oj}, 1 \right) \tag{14}$$

---

[2]The radius ρ of a singularity ring always has a value of 1 in this paper; however, the radius of the point at which the velocity is computed ρ_o varies between 0 and 1 in the mass-flow computation but has a value of 1 elsewhere.

$$\frac{u_b(\xi_{oj},1)}{V_\infty} = \sum_{i=1}^{N} \rho_{bi}\left(\frac{d\rho_b}{d\xi}\right)_i \hat{u}_b(\xi_i - \xi_{oj},1)\Delta\xi \tag{15}$$

$$\frac{v_b(\xi_{oj},1)}{V_\infty} = \sum_{i=1}^{N} \rho_{bi}\left(\frac{d\rho_b}{d\xi}\right)_i \hat{v}_b(\xi_i - \xi_{oj},1)\Delta\xi \tag{16}$$

where the influence functions are

$$\hat{u}_\gamma(\xi_i - \xi_{oj},1) = \frac{k}{2}(E - K) \tag{17}$$

$$\hat{v}_\gamma(\xi_i - \xi_{oj},1) = -(\xi_i - \xi_{oj})\frac{k}{4}\left(\frac{2 - k^2}{1 - k^2}E - 2K\right) \tag{18}$$

$$\hat{u}_b(\xi_i - \xi_{oj},1) = \frac{\xi_i - \xi_{oj}}{2\left[(\xi_i - \xi_{oj})^2 + 1\right]^{3/2}} \tag{19}$$

$$\hat{v}_b(\xi_i - \xi_{oj},1) = \frac{1}{2\left[(\xi_i - \xi_{oj})^2 + 1\right]^{3/2}} \tag{20}$$

For points off the cylinder[3] $\rho_{oj} = 1$ (necessary when computing mass flow), equations (13) to (16) apply with the $\pm$term dropped in equation (13) and using the influence functions

$$\hat{u}_\gamma(\xi_i - \xi_{oj},\rho_{oj}) = \frac{k}{4\sqrt{\rho_{oj}}}\left[\left(\frac{2 - k^2}{1 - k^2} - \frac{1}{\rho_{oj}}\frac{k^2}{1 - k^2}\right)E - 2K\right] \tag{21}$$

$$\hat{u}_b(\xi_i - \xi_{oj},\rho_{oj}) = \frac{\xi_i - \xi_{oj}}{2\left[(\xi_i - \xi_{oj})^2 + \rho_{oj}^2\right]^{3/2}} \tag{22}$$

---

[3]The variables ξ and ρ locate a singularity, and ξ_o,ρ_o is the point at which the velocity is calculated. When going from integrals and continuous distributions to summations and discrete distributions, it is necessary to indicate these quantities by ξ_i,ρ_i and ξ_{oj},ρ_{oj}, respectively. When $\rho_{oj} = 1$, the location ξ_{oj},ρ_{oj} is a control point; but when $\rho_{oj} \neq 1$, then ξ_{oj},ρ_{oj} is meant to indicate any arbitrary point.

$\hat{v}_{\gamma}\left(\xi_i - \xi_{oj}, \rho_{oj}\right)$ and $\hat{v}_{b}\left(\xi_i - \xi_{oj}, \rho_{oj}\right)$ as given in equations (9) and (10). Equation (13) is derived in appendix B by using the influence function in equation (21).

A linear interpolation of the Γ_{oi} was used to find the Γ_{oj}. A check of the Γ_{oi} distributions showed that the distribution approaches zero at the trailing edge, thereby satisfying the Kutta condition. Note that the pressure distribution over the camber surface depends on three quantities: camber surface slopes, aspect ratio, and mass flow (controlled by the size of the center body).

Thickness Solution

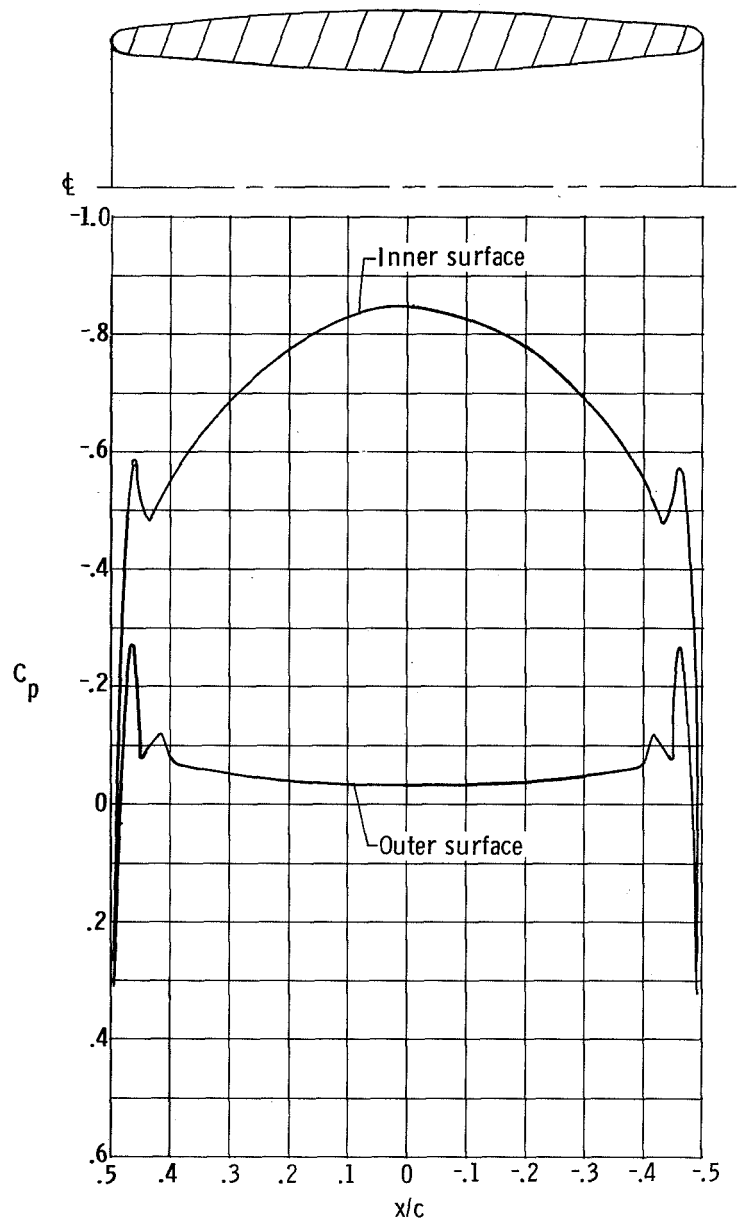
The flow to be calculated in this case is the subsonic flow over an annular wing with a symmetrical airfoil section at zero angle of attack. The flow must satisfy the same basic equations and conditions as the camber solution. The tangent flow condition is now equation (7).

In the case of planar problems, thickness can be accounted for by a distribution of sources on a plane. Such a distribution has the necessary symmetry property and discontinuous change of sign in the vertical velocity for the planar wing thickness. However, if two planar source sheets are placed in close proximity to form a biplane, the interference between the two sets of source distributions will cause a mutual induction of camber. The vertical velocity at some point on either sheet will no longer depend solely on the local source strength but will also depend on the complete source distribution of the other sheet. This interference problem also exists for an annular wing. The induced camber effects can be canceled by a vortex distribution, the strength of which is adjusted to just cancel the interference effects on the wing. Then the total source and vortex radial velocity will have the necessary discontinuous property across the singularity sheet.

Note that, because of the interference effects, an annular wing with thickness only at zero angle of attack has a negative radial force; that is, it tends to collapse. The internal flow area reduction due to thickness accelerates the flow much more than the same flow area reduction externally. Figure 2 shows the pressure distributions for an annular wing with thickness only and the same wing with enough camber to make the inside diameter constant. The radial force changes sign when geometric camber is added, and with the proper amount of geometric camber the inner and outer pressures can be equalized.

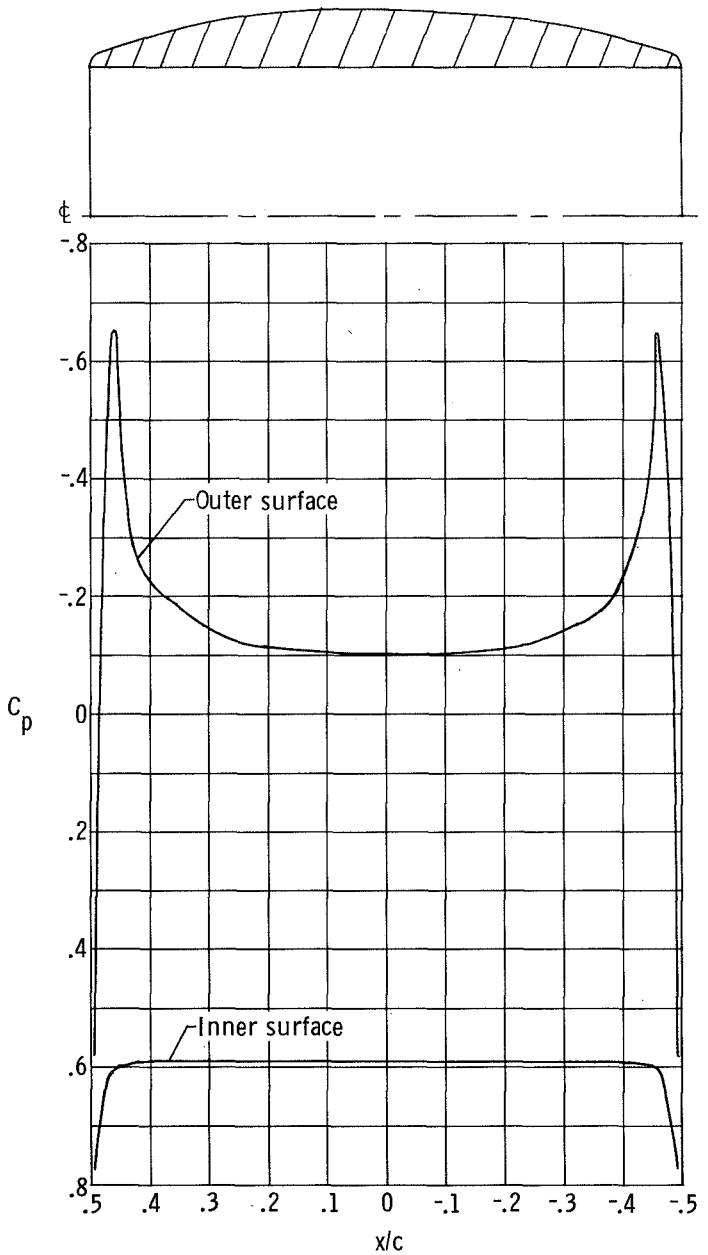
Because the annular wing with thickness only has the effect of camber in its flow, namely a nonzero radial force, it would be convenient to refer to this effect as an aerodynamic camber. Then the variations of the mean camber line from the chord could be referred to as geometric camber.

The radial velocity induced at ξ_0 and $\rho_0 = 1$ by a continuous distribution of source rings of diameter D (radius $\rho = 1$) over an axial length of c is (see app. B)



(a) 4.3-percent thickness and zero camber; $\mu = 0.87$.

Figure 2.- Pressure distribution on an annular wing without and with camber at $M_\infty = 0.50$.



(b) 4.3-percent thickness and constant inside diameter; μ = 0.70.

Figure 2.- Concluded.

$$\frac{v_q(\xi_o,1)}{V_\infty} = \pm\frac{q(\xi_o)}{2V_\infty} - \int_{-\frac{1}{\lambda}}^{\frac{1}{\lambda}} \frac{q(\xi)}{4\pi V_\infty}\hat{v}_q(\xi - \xi_o,1)\,d\xi \tag{23}$$

where $q(\xi)$ is the source strength at ξ and the influence function $\hat{v}_q(\xi - \xi_o,1)$ is related to the vortex influence function for the velocity u by

$$\hat{v}_q(\xi - \xi_o,1) = 2\hat{u}_\gamma(\xi - \xi_o,1) \tag{24}$$

The integral term in equation (23) is the induced camber due to interference effects. This integral can be canceled by a vortex distribution the strength of which is determined by

$$\frac{v_{t,\gamma}(\xi_o,1)}{V_\infty} = \frac{1}{2\pi}\int_{-\frac{1}{\lambda}}^{\frac{1}{\lambda}} \frac{\gamma_t(\xi)}{V_\infty}\hat{v}_{t,\gamma}(\xi - \xi_o,1)\,d\xi$$

$$= \int_{-\frac{1}{\lambda}}^{\frac{1}{\lambda}} \frac{q(\xi)}{4\pi V_\infty}\hat{v}_q(\xi - \xi_o,1)\,d\xi \tag{25}$$

where $\gamma_t(\xi)$ has the dimensions of γ_+, and $\hat{v}_{t,\gamma}$ is computed by using equation (18). When this is done, equation (23) can be written as

$$\frac{v_q(\xi_o,1)}{V_\infty} + \frac{v_{t,\gamma}(\xi_o,1)}{V_\infty} = \pm\frac{q(\xi_o)}{2V_\infty} \tag{26}$$

The flow constructed in this way will satisfy the required boundary condition in equation (7). Thus from equations (7) and (26) the source strength is determined by

$$q(\xi_o) = -2V_\infty\alpha_t(\xi_o) \tag{27}$$

where the boundary condition is being satisfied at points $(\xi_o,\rho_o = 1)$.

The solution of equation (25) for the unknown vortex strength is the same type of numerical problem found in the camber solution. An alternate method of finding $\gamma_t(\xi)$ has been outlined by Weissinger (ref. 8). By examination of the boundary conditions for a vortex flow, the vortex strength necessary to cancel the integral of equation (23) is determined by

$$\frac{1}{2}\int_{-\frac{1}{\lambda}}^{\frac{1}{\lambda}} Q(\xi)V_q\left(\xi - \xi_o,1\right)d\xi = -\int_{-\frac{1}{\lambda}}^{\frac{1}{\lambda}} \frac{\bar{g}_q(\xi)}{\lambda\left(\xi - \xi_o\right)}\,d\xi + \frac{1}{\lambda}\int_{-\frac{1}{\lambda}}^{\frac{1}{\lambda}} \bar{g}_q(\xi)U_o\left(\xi - \xi_o,1\right)d\xi \tag{28}$$

where

$$Q(\xi) = \lambda\int_{-\frac{1}{\lambda}}^{\xi} \frac{q(\xi')}{V_\infty}\,d\xi' = 2\,\frac{t(\xi)}{c} \tag{29}$$

$$V_q\left(\xi - \xi_o,1\right) = \frac{\sqrt{1 - k^2}}{2}\,\frac{\left(\xi - \xi_o\right)}{\left|\xi - \xi_o\right|}\left[2\left(1 + k^2\right)E - \left(2 + k^2\right)K\right] \tag{30}$$

$$U_o\left(\xi - \xi_o,1\right) = \frac{1}{\xi - \xi_o}\left\{\frac{1}{k}\left[\left(k^2 - 2\right)E + 2\left(1 - k^2\right)K\right] + 1\right\} \tag{31}$$

and $t(\xi)$ is the wing thickness and $\bar{g}_q(\xi)$ is related to the vortex strength by

$$\frac{\gamma_t(\xi)}{V_\infty} = -\frac{1}{2\lambda}Q(\xi) - \frac{1}{\lambda^2}\bar{g}_q(\xi) \tag{32}$$

(The symbol $-\gamma_t(\xi)/V_\infty$ is called $g_q(\xi)$ in ref. 8.) The kernel functions V_q and U_o are continuous and antisymmetrical. The solutions of equations (25) and (28) for $\gamma_t(\xi)$ should require about the same computing time; equation (28) was chosen for this study. In numerical form, equations (28) and (29) yield

$$\sum_{i=1}^{N} \frac{t_i}{c}V_q\left(\xi_i - \xi_{oj},1\right)\Delta\xi = \sum_{i=1}^{N}\left[U_o\left(\xi_i - \xi_{oj},1\right) - \frac{1}{\xi_i - \xi_{oj}}\right]\frac{\bar{g}_{qi}}{\lambda}\,\Delta\xi \tag{33}$$

The source and vortex-ring locations $\left(\xi_i,\rho_i = 1\right)$ and control-point locations $\left(\xi_{oj},\rho_{oj} = 1\right)$ are the same as in the camber solution. The same method of solution of linear equations discussed previously was used for equation (33).

Once the source and vortex strengths are computed, the perturbation velocities can be computed at any point $\left(\xi_{oj},\rho_{oj}\right)$ Equations (13) and (14) give the vortex velocities $u_{t,\gamma}$ and $v_{t,\gamma}$ when $\alpha_{oi}\Gamma_{oi}$ is replaced by $\left(\gamma_{ti}/V_\infty\right)\Delta\xi$ and similarly for $\alpha_{oj}\Gamma_{oj}$. Recall that for points off the cylinder $\rho_{oj} = 1$, the $\pm$ term in equation (13) must be dropped. The source velocities at points on the cylinder $\rho_{oj} = 1$ are computed from (refs. 8 and 12)

$$\frac{u_q(\xi_{oj},1)}{V_\infty} = \sum_{i=1}^{N} \frac{\alpha_{ti}\,\Delta\xi}{2\pi}\,\hat{u}_q(\xi_i - \xi_{oj},1) \tag{34}$$

$$\frac{v_q(\xi_{oj},1)}{V_\infty} = \sum_{i=1}^{N} \frac{\alpha_{ti}\,\Delta\xi}{2\pi}\,\hat{v}_q(\xi_i - \xi_{oj},1) \mp \alpha_{tj} \tag{35}$$

where the influence function $\hat{v}_q$ is given by equation (24) and $\hat{u}_q$ is

$$\hat{u}_q(\xi_i - \xi_{oj},1) = \frac{-2kE}{\xi_i - \xi_{oj}} \tag{36}$$

Equation (35) can be obtained from equations (23) and (27). For points off the cylinder defined by $\rho_{oj} = 1$ (necessary when computing the mass flow), equations (34) and (35) apply with the term $\mp\alpha_{tj}$ in equation (35) dropped and by using the influence functions

$$\hat{u}_q(\xi_i - \xi_{oj},\rho_{oj}) = \frac{-4(\xi_i - \xi_{oj})E}{\sqrt{(\xi_i - \xi_{oj})^2 + (1 + \rho_{oj})^2}\left[(\xi_i - \xi_{oj})^2 + (1 - \rho_{oj})^2\right]} \tag{37}$$

$$\hat{v}_q(\xi_i - \xi_{oj},\rho_{oj}) = \left[\frac{2}{\rho_{oj}\sqrt{(\xi_i - \xi_{oj})^2 + (1 + \rho_{oj})^2}}\right] \times \left\{\left[1 - \frac{2\rho_{oj}(\rho_{oj} - 1)}{\left[(\xi_i - \xi_{oj})^2 + (1 - \rho_{oj})^2\right]}\right]E - K\right\} \tag{38}$$

Note that the thickness solution depends on two quantities: thickness distribution $t(\xi)$ and aspect ratio.

Pressure Distribution and Mass Flow

Once the singularity strengths are determined, the velocities due to camber, center body, and thickness can be added to give the total surface velocity and pressure distribution. The internal flow velocities can be integrated over a surface perpendicular to the center line at some ξ-location to calculate the mass flow through the annular wing.

In regions of high body curvature, such as the nose, approximation of the surface velocity by the velocity on a constant-diameter cylinder is not satisfactory. A correction for this approximation called the Riegels factor, is shown by Weber (ref. 14) to improve the calculated velocities in the nose region of a two-dimensional airfoil.

The Riegels factor can be developed from geometry as follows. By writing the equation for the wing surface[4] as

$$f(x,r) = r - r_w(x) = 0$$

the unit tangential vector in the meridional direction is

$$\hat{e}_t = \frac{\hat{e}_x + \frac{dr_w}{dx}\hat{e}_r}{\sqrt{1 + \left(\frac{dr_w}{dx}\right)^2}}$$

where $\hat{e}_x$ and $\hat{e}_r$ are unit vectors in the x- and r-directions, respectively. Writing the total compressible surface velocity as

$$\overline{V} = \left(V_\infty + u\right)\hat{e}_x + v\hat{e}_r$$

the total tangential surface velocity is

$$V_t = \overline{V} \cdot \hat{e}_t = \frac{\left(V_\infty + u\right) + v\frac{dr_w}{dx}}{\sqrt{1 + \left(\frac{dr_w}{dx}\right)^2}}$$

Consistent with linear theory the total surface velocity is approximated by

$$V = \frac{\left(V_\infty + u\right)_{\rho=1}}{\sqrt{1 + \left(\frac{dr_w}{dx}\right)^2}} \tag{39}$$

The quantity in the denominator is the Riegels factor. Equation (39) shows that the total velocity on the wing surface is found by computing the total x-velocity on the cylinder $\rho = 1$ and multiplying this result by the Riegels factor.

---

[4]All velocities from now on are compressible and coordinates are for the actual (untransformed) geometric shape.

The surface pressure distribution is computed by making compressibility and Riegels factor corrections to the incompressible velocities and calculating the pressure from

$$C_p = \frac{2}{\gamma M_\infty^2}\left\{\left[1 + \frac{\gamma - 1}{2} M_\infty^2\left(1 - \frac{V^2}{V_\infty^2}\right)\right]^{\frac{\gamma}{\gamma - 1}} - 1\right\} \tag{40}$$

where γ is the ratio of specific heats.

The mass flow is computed in the form of a ratio

$$\mu \equiv \frac{m}{\rho_\infty A_h V_\infty} = \left(\frac{\rho_{\xi_o}}{\rho_\infty}\right)\left(\frac{A_{\xi_o}}{A_h}\right)\left(\frac{V_{\xi_o}}{V_\infty}\right) \tag{41}$$

where m is the mass flow through the annular wing, V_{ξ_o} is the total x-velocity at some ξ_o location, and A_h is a reference area based on the highlight diameter (diameter at $x = c/2$ in fig. 1). The area A_{ξ_o} is the internal cross-sectional area of the wing at ξ_o less the local center-body cross-sectional area. The velocity ratio is computed by dividing the flow area into small annuli and calculating the mass flow through each annulus and summing the result as follows (assuming density constant over the cross section)

$$\frac{V_{\xi_o}}{V_\infty} = \sum_j \frac{V\left(\xi_o,\rho_{oj}\right)}{V_\infty}\frac{A\left(\xi_o,\rho_{oj}\right)}{A_{\xi_o}} \tag{42}$$

The $V\left(\xi_o,\rho_{oj}\right)$ are the total x-velocities computed at a fixed ξ_o and the particular ρ_{oj} location. The annulus area $A\left(\xi_o,\rho_{oj}\right)$ is at the ξ_o,ρ_{oj} location. The density ratio is calculated from the one-dimensional relation

$$\frac{\rho_{\xi_o}}{\rho_\infty} = \left[1 + \frac{\gamma - 1}{2} M_\infty^2\left(1 - \frac{V_{\xi_o}^2}{V_\infty^2}\right)\right]^{\frac{1}{\gamma - 1}} \tag{43}$$

Boundary-Layer Solution

The boundary layer was approximated by a turbulent boundary layer over the entire chord length. The incompressible turbulent boundary-layer method of Truckenbrodt (ref. 15) was used. John B. Peterson, Jr., of NASA Langley Research Center developed the necessary equations for computing the boundary layer and programed these equations for the boundary-layer subroutine used in this study.

The inviscid velocity distribution at the trailing edge is greatly in error since a stagnation point is computed there. Hence, the velocity was extrapolated to the trailing edge (ref. 16). A procedure similar to that used in reference 17 was employed. In each case an estimate was made of the point beyond which the inviscid velocity became unreasonable because of the trailing-edge stagnation point. Then a least-square-curve fit of the velocities was made between that point (97-percent-chord station) and the 80- or 90-percent-chord station. This least-square-curve fit was then used to recompute the velocities between the 80- or 90-percent station and the trailing edge. The outer surface was computed first so that the Nth value of the recomputed outer surface velocity (value closest to trailing edge) could also be included in the inner surface curve fit. A linear-curve fit was used on the outer surface and a parabolic fit on the inner surface. This procedure produced smooth and reasonable velocities for the boundary-layer calculation. It was determined that uncertainties in this method gave 2-percent or less error in μ. The figures to be presented indicate the range in which the curve fit was made.

It should be clearly understood, however, that this trailing-edge procedure is simply an approximate scheme in place of calculating the true effect of the correct wake. (For a nonzero trailing-edge thickness the total source strength is nonzero, and therefore some wake is simulated.) It may be possible to simulate the correct wake effect by use of a wake annular thickness distribution which has just enough geometric camber to cancel the ΔC_p due to thickness, thereby making the inner and outer pressures equal.

In order to obtain convergence of the boundary-layer displacement thickness, it was necessary to modify the wing coordinates, on the mth iteration, with the following effective displacement thickness

$$\delta^* = \frac{\delta_o^* + 2\delta_1^* + \ldots + (m+1)\delta_m^*}{1 + 2 + \ldots + (m+1)} \tag{44}$$

where δ_o^* is based on velocities computed by using the actual wing geometry. On the first iteration the wing coordinates were modified by $\delta_o^*/2$, rather than δ_o^* as indicated in equation (44). When separation occurred during the iteration process (shape factor $H \geq 2.4$), the shape factor was limited to 2.55 to permit convergence.

Computer Program

The complete computer program is described in appendix A. Input and output quantities and the program listing are included. One output quantity is the critical Mach number, which is determined by an approximate method of von Kármán (ref. 18). The printed value of the critical Mach number is valid only when M_∞ is low enough for the flow to be assumed incompressible.

The number of chordwise divisions N was 150. For the nacelles of this study the magnitude of the leading-edge suction peak had reached a stationary value by this value of N. For the mass-flow computation the radial increments were $r_{inner}/170$. Since the velocity generally had a very small variation across the flow area (substantiating the usual one-dimensional-flow assumption), these radial increments were more than sufficient.

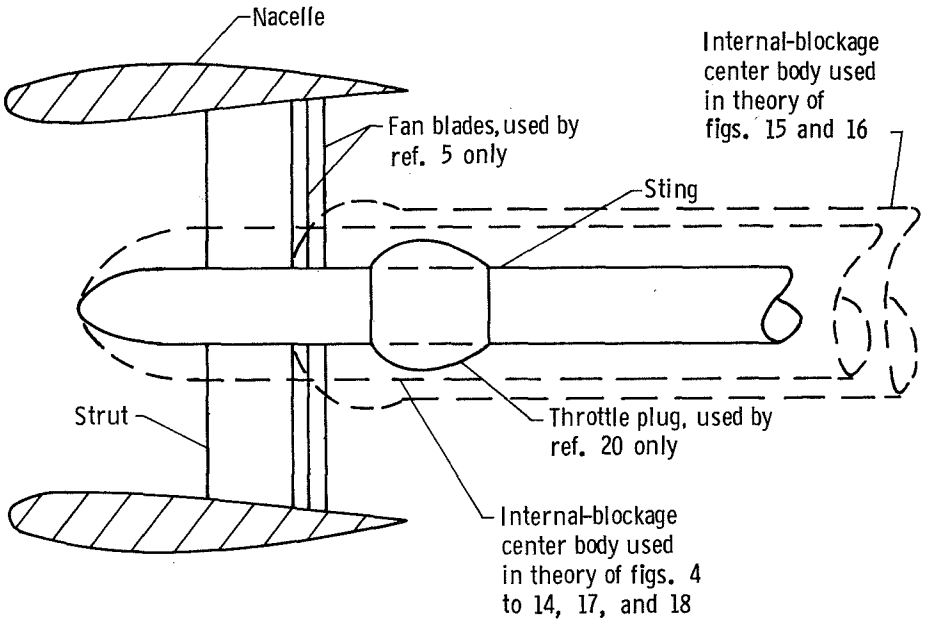
Calculations on a CDC-6600 computer required 2 minutes for an inviscid calculation and 4 minutes for a viscous calculation. Each additional iteration on the viscous calculation required about 2 minutes. This time could be reduced by use of a cosine distribution of singularities which would increase the singularity density in regions of strong pressure gradients and reduce the density in the center of the body. Such a distribution of vortexes will produce the same downwash at points midway between the vortexes as a continuous distribution of vorticity (app. B of ref. 19).

COMPARISON OF THEORY AND EXPERIMENT

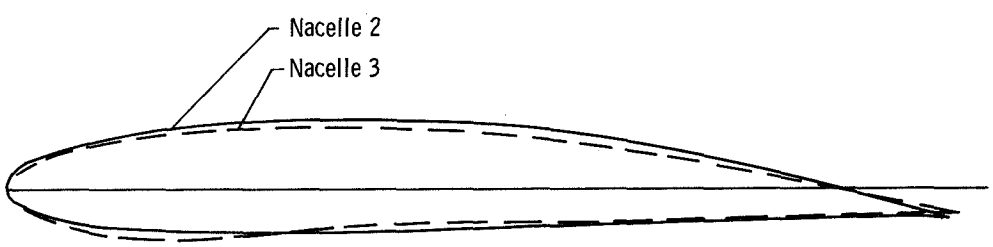
Calculations were compared with experimental data obtained in two wind tunnels (refs. 5 and 20) to provide a check on the present theory. The comparison with the tests of reference 5 is discussed first. These tests covered a range of subsonic Mach numbers at primarily zero angle of attack. Figure 3(a) shows a sketch of the experimental setup and its relation to the theoretical model used to represent it. The mass flow in the experiment was controlled by the insertion of stationary sets of fan blades. The blockage of the sting, struts, and fan blades was represented in the theory by the blockage of a center body. In the experiment, the highest mass flow was obtained without a set of blades installed. In this condition the experimental setup was more faithfully modeled by the theory than with sets of blades installed, and the best agreement between measured and calculated internal pressures would be expected. This expected result was achieved as will be discussed.

The pressure distribution and mass flow are not sensitive to variations in the diameter D of the vortex and source rings (see fig. 1). The value of D used in this study was the average of the minimum inside diameter and the maximum outside diameter of the wing.

Nacelles 2 and 3 of reference 5 were selected for the present study (fig. 3(b)). These nacelles were designed to have prescribed external pressure distributions and internal contraction ratios. Comparisons between theory and experiment were made for three Mach numbers and several mass-flow ratios. The exact sting leading edge and nose shape were not given. Thus the nose of the center body in the theoretical model was placed at approximately the position of the sting leading edge and the shape was arbitrarily selected to be that of a NASA supercritical body of revolution (ref. 21). At the maximum diameter



(a) Relation of experimental setup to theoretical model. (Not to scale.)



(b) Profile shapes for nacelles 2 and 3.

Figure 3.- Geometry of experimental setup and theoretical model.

station of the center body, a constant-diameter cylinder was attached and extended downstream. As seen in figure 3(a), the constant diameter portion of the center body extended to infinity for the high-mass-flow cases of figures 4 to 14. However, for the low-mass-flow cases of figures 15 and 16 the center body was expanded at the approximate location of the fan blades. The necessity of this expansion for the low-mass-flow cases is discussed subsequently.

The quoted accuracy of the measured mass flows in reference 5 is 1.5 percent based on boundary layer and instrument errors. However, Young also states that the presence of the fan blades tended to separate the flow. Hence, for the lower mass flows tested, larger errors in the mass-flow ratio μ may be present. Errors in μ would be reflected in a lack of agreement of measured and calculated pressures, especially on the inner surface and at the leading-edge suction peak.

In the experiments the transition bands were applied from 6.25 to 10.4 percent of the chord. Therefore in the theory the boundary layer was approximated by a turbulent boundary layer over the entire chord length, as discussed previously. The Reynolds number based on chord length was fixed at the experimental value of 2×10^6.

In all the pressure-distribution curves shown herein, with the exception of figure 2(a), the upper and lower curves correspond to the outer and inner surfaces, respectively.

A comparison of the theory and experiment is presented in figures 4 to 12. Mach numbers of 0.3, 0.5, and 0.7 were examined. In selecting the center-body diameter, two approaches are available. First, if the desired mass flow is known, the procedure is simply to vary the body size until the desired mass flow is achieved. Only a few trial solutions are necessary to establish a curve of mass flow versus maximum body radius from which the maximum body radius can be selected to yield the desired mass flow. The second approach is to match the internal blockage area between the experimental geometry and a body of revolution and calculate the mass flow. Figures 4 to 7 used the first approach, and figures 8 to 11 used the second. Each figure gives the number of iterations on the boundary layer and the chordwise range from which velocities were used to obtain the least-square-curve fits. The iteration was continued until the maximum change in δ^* on either surface was less than 2 percent of the maximum δ^* for the respective surfaces. Good agreement is obtained between measured and calculated pressures when the boundary layer is included.

Note that figures 6, 7, 10, and 11 have slightly supercritical flow as shown by the value of the critical pressure coefficient C_p^*. A weak shock seems to occur near the leading edge. However, aft of this region agreement is good.

Insight into the effect of the boundary layer on the mass flow is provided by examining figures 8 to 11. These figures indicate that for a fixed geometry, the presence of the boundary layer increases the mass flow, contrary to what might at first be expected. Recall the case of a two-dimensional airfoil where the boundary layer causes an effective decrease in angle of attack; that is, the boundary layer reduces the lift and therefore the area under the curve of C_p versus ξ (ref. 17). As seen from figures 8 to 11, the boundary layer has the same effect on an annular wing, only in this case the radial force is being lowered. As this happens, the leading-edge stagnation point moves more toward

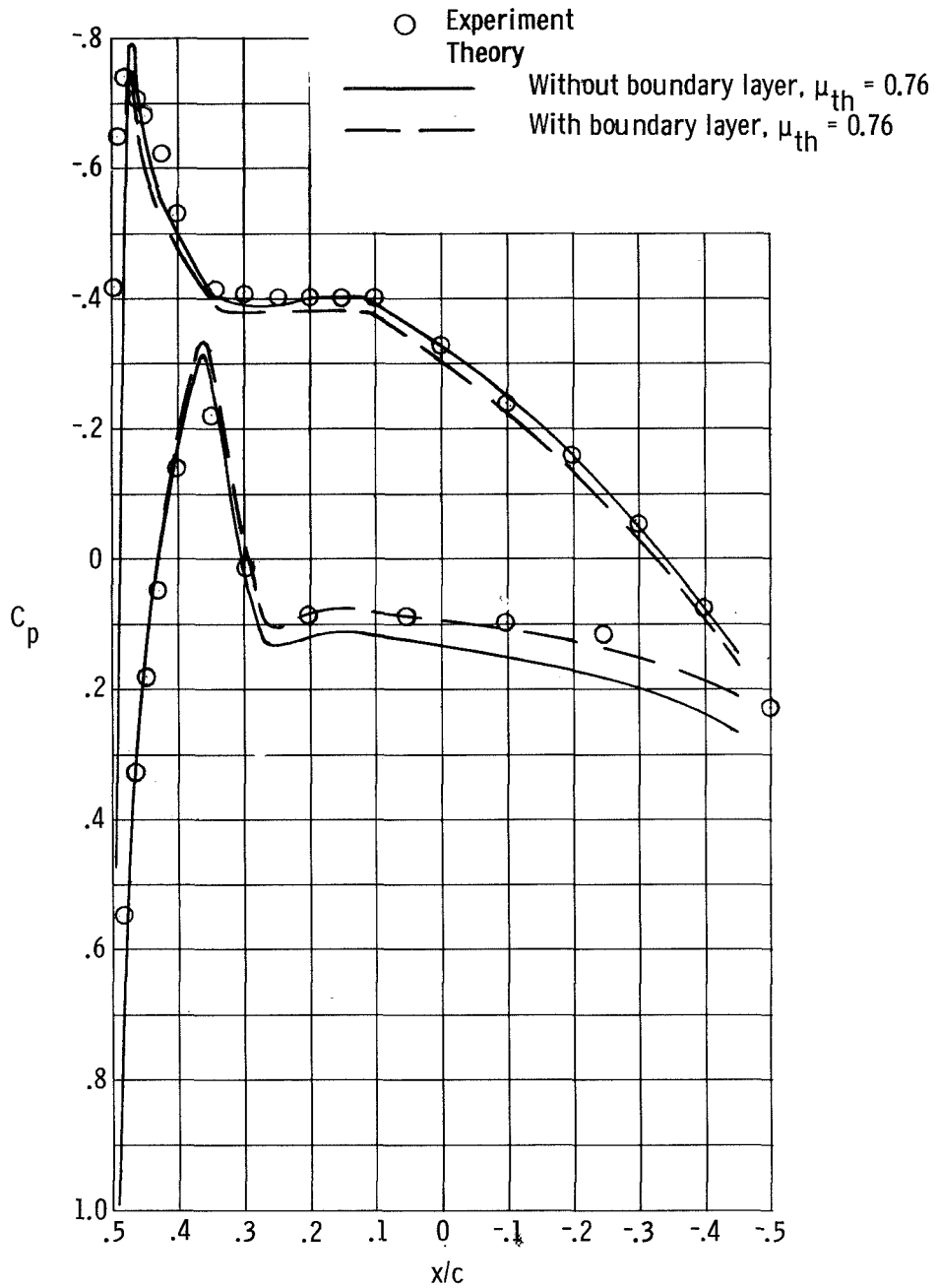


Figure 4.- Pressure distribution on nacelle 3; M_∞ = 0.3; $\mu_{th} = \mu_{exp}$ = 0.76. Boundary layer: Six iterations; least square 90- to 97-percent c.

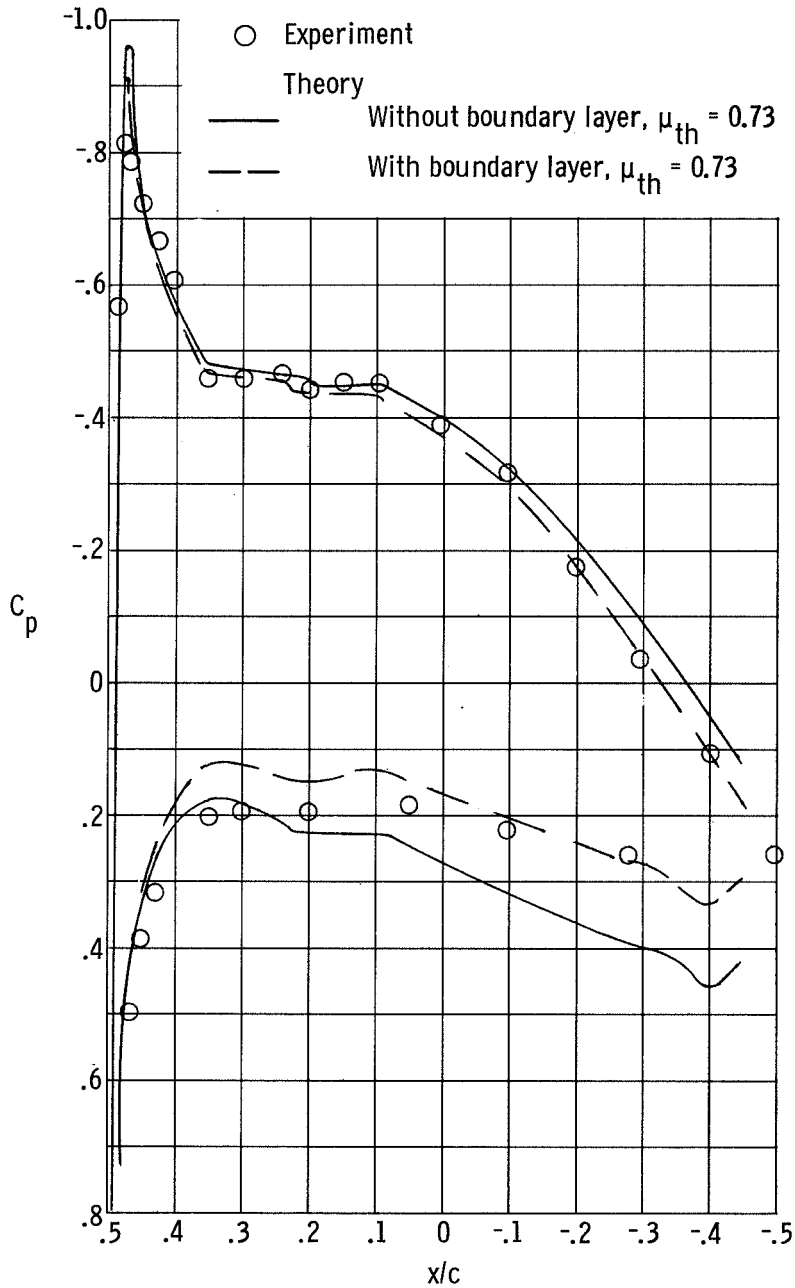


Figure 5.- Pressure distribution on nacelle 2; M_∞ = 0.5; $\mu_{th} = \mu_{exp}$ = 0.73. Boundary layer: Six iterations; least square 90- to 97-percent c.

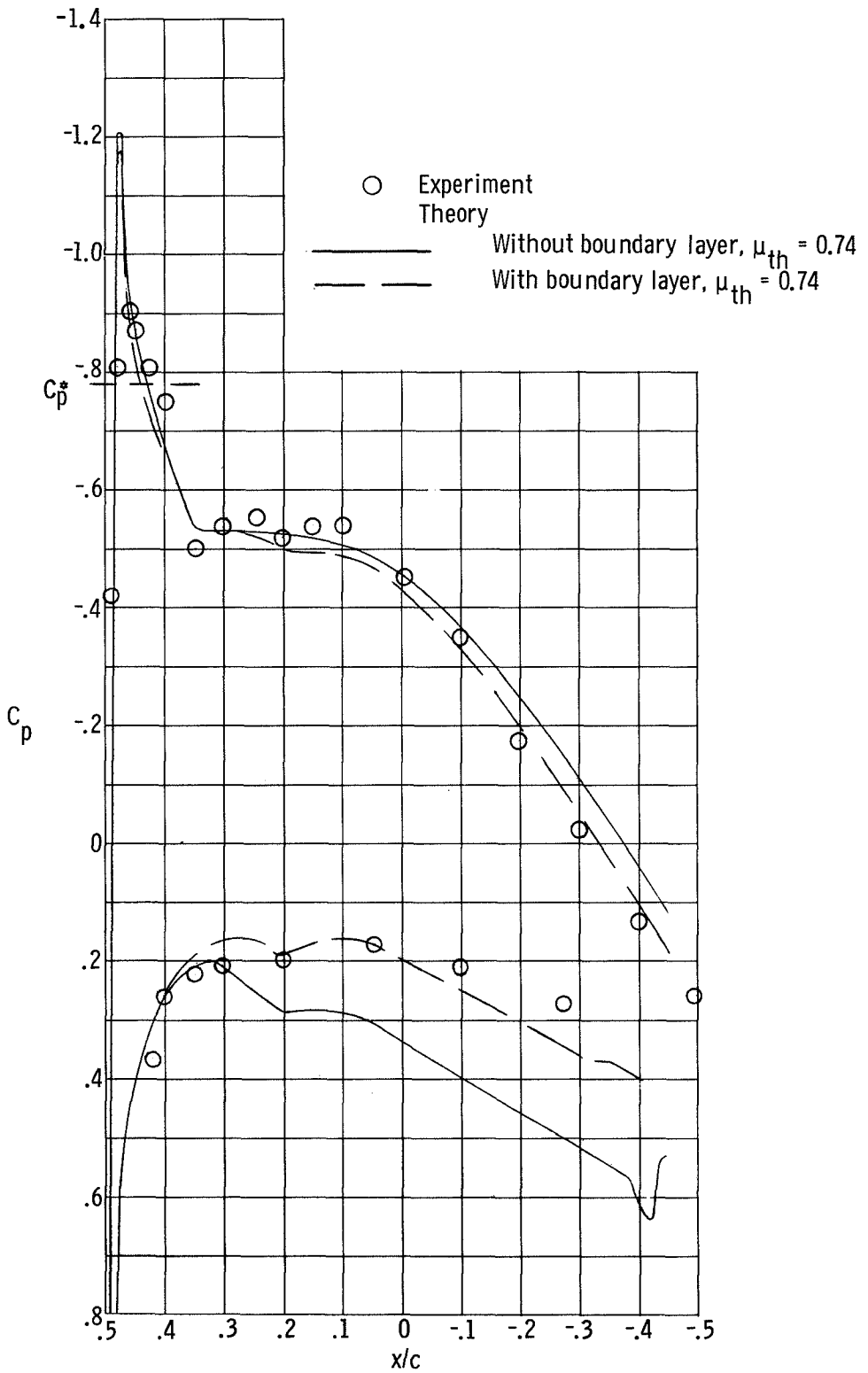


Figure 6.- Pressure distribution on nacelle 2; $M_\infty = 0.7$; $\mu_{th} = \mu_{exp} = 0.74$. Boundary layer: Five iterations; least square 80- to 97-percent c.

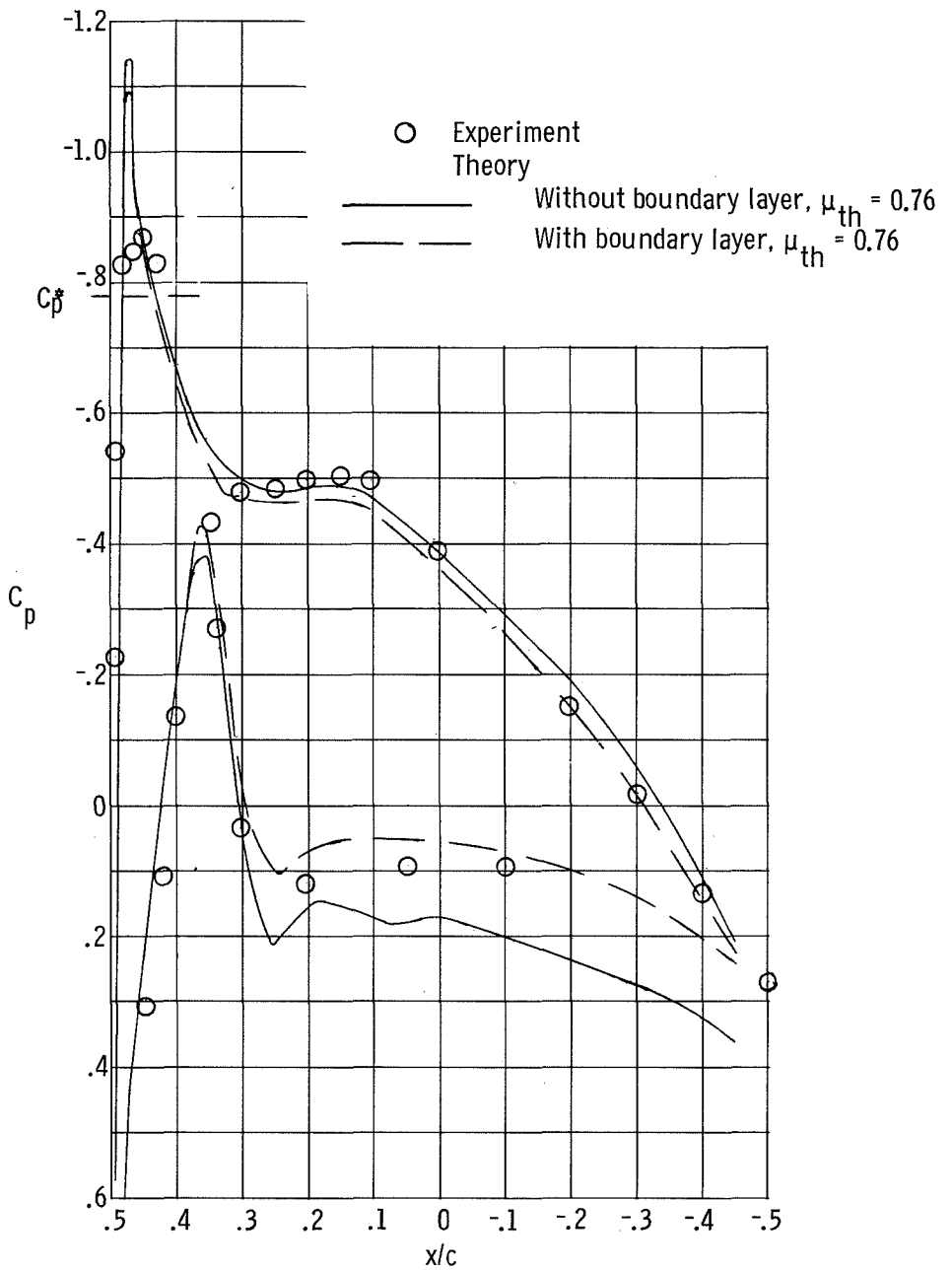


Figure 7.- Pressure distribution on nacelle 3; $M_\infty = 0.7$; $\mu_{th} = \mu_{exp} = 0.76$. Boundary layer: Five iterations; least square 80- to 97-percent c.

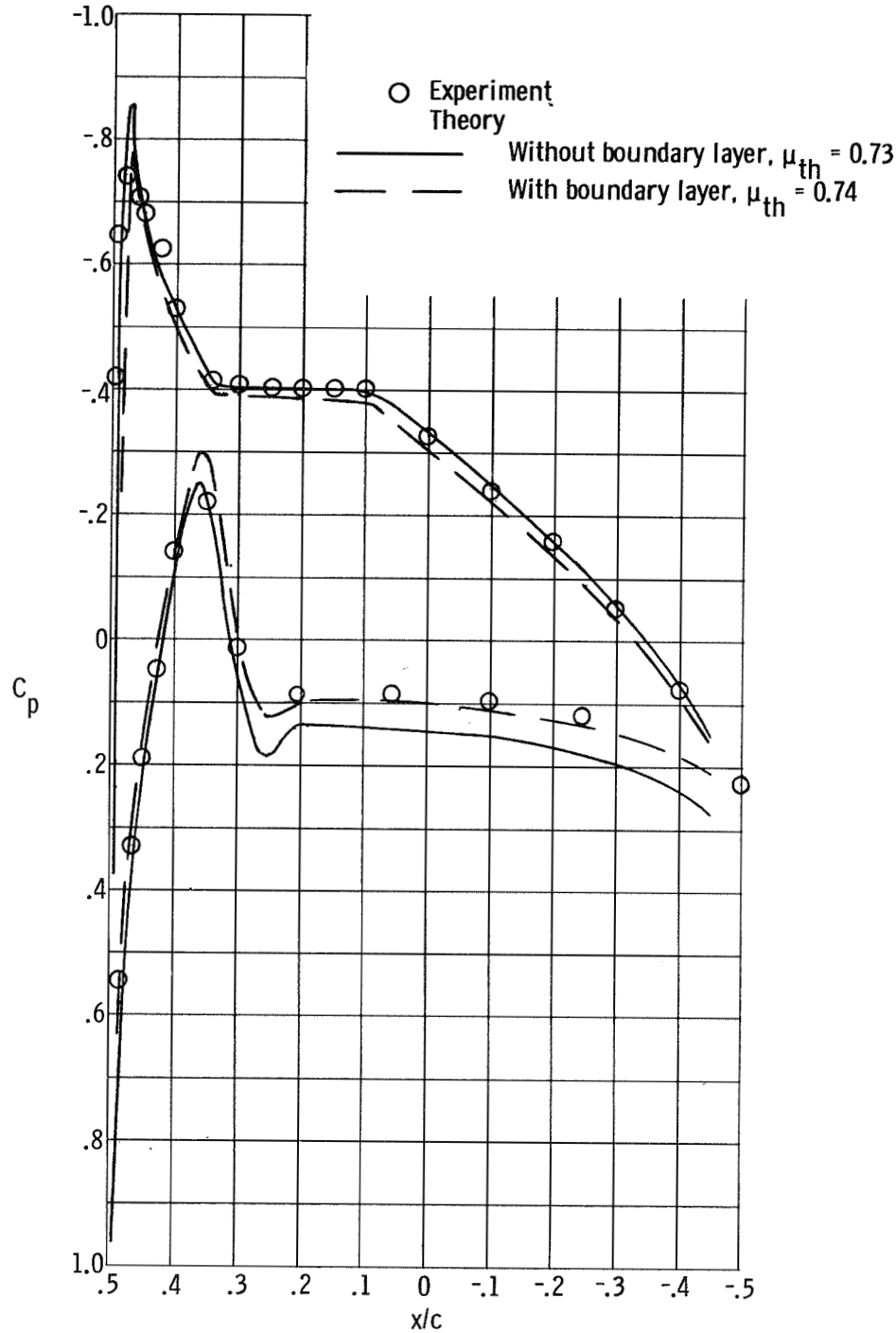


Figure 8.- Pressure distribution on nacelle 3; $M_\infty = 0.3$; $\left(r_b/c\right)_{max} = 12.79$ percent; $\mu_{exp} = 0.76$. Boundary layer: Six iterations; least square 90- to 97-percent c.

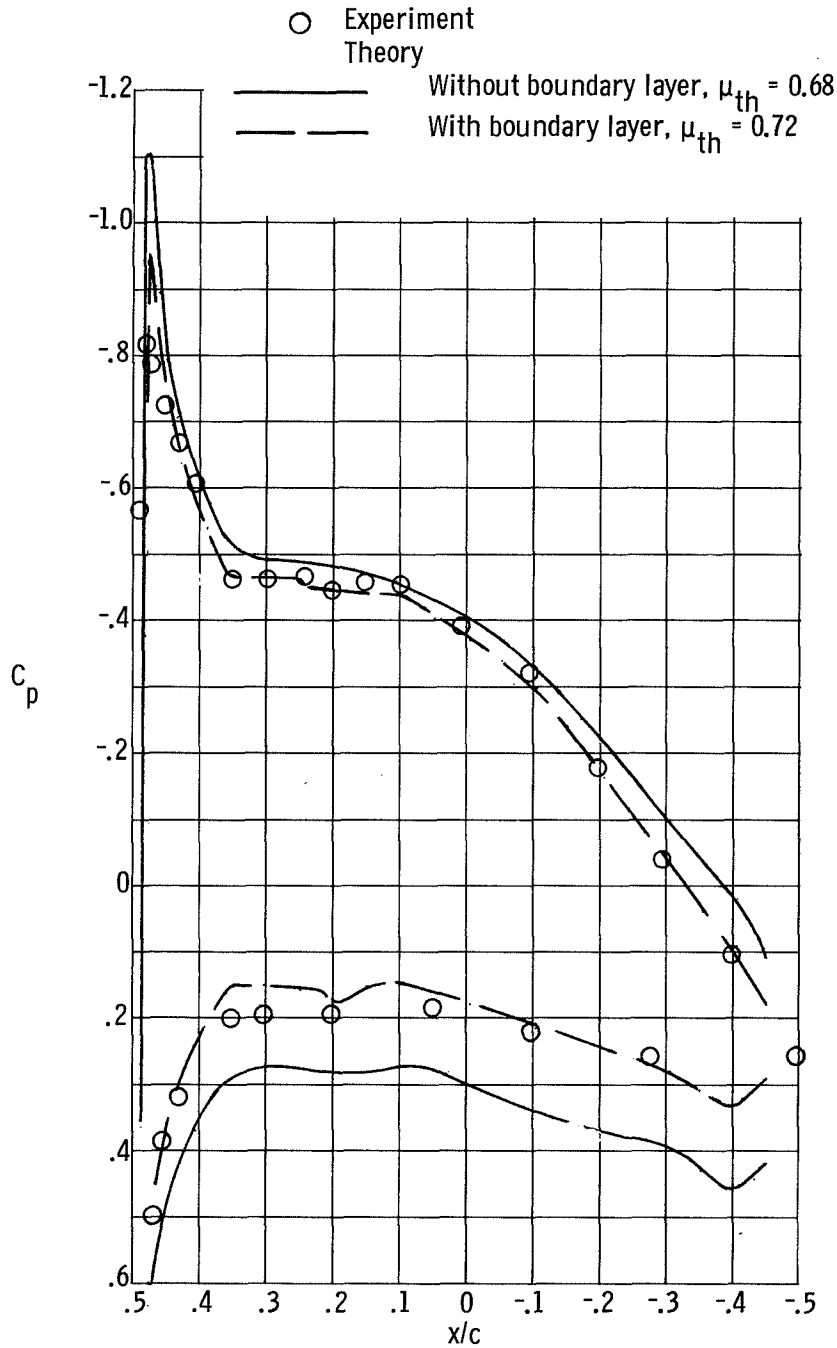


Figure 9.- Pressure distribution on nacelle 2; M_∞ = 0.5; $\left(r_b/c\right)_{max}$ = 12.61 percent; μ_{exp} = 0.73. Boundary layer: Six iterations; least square 90- to 97-percent c.

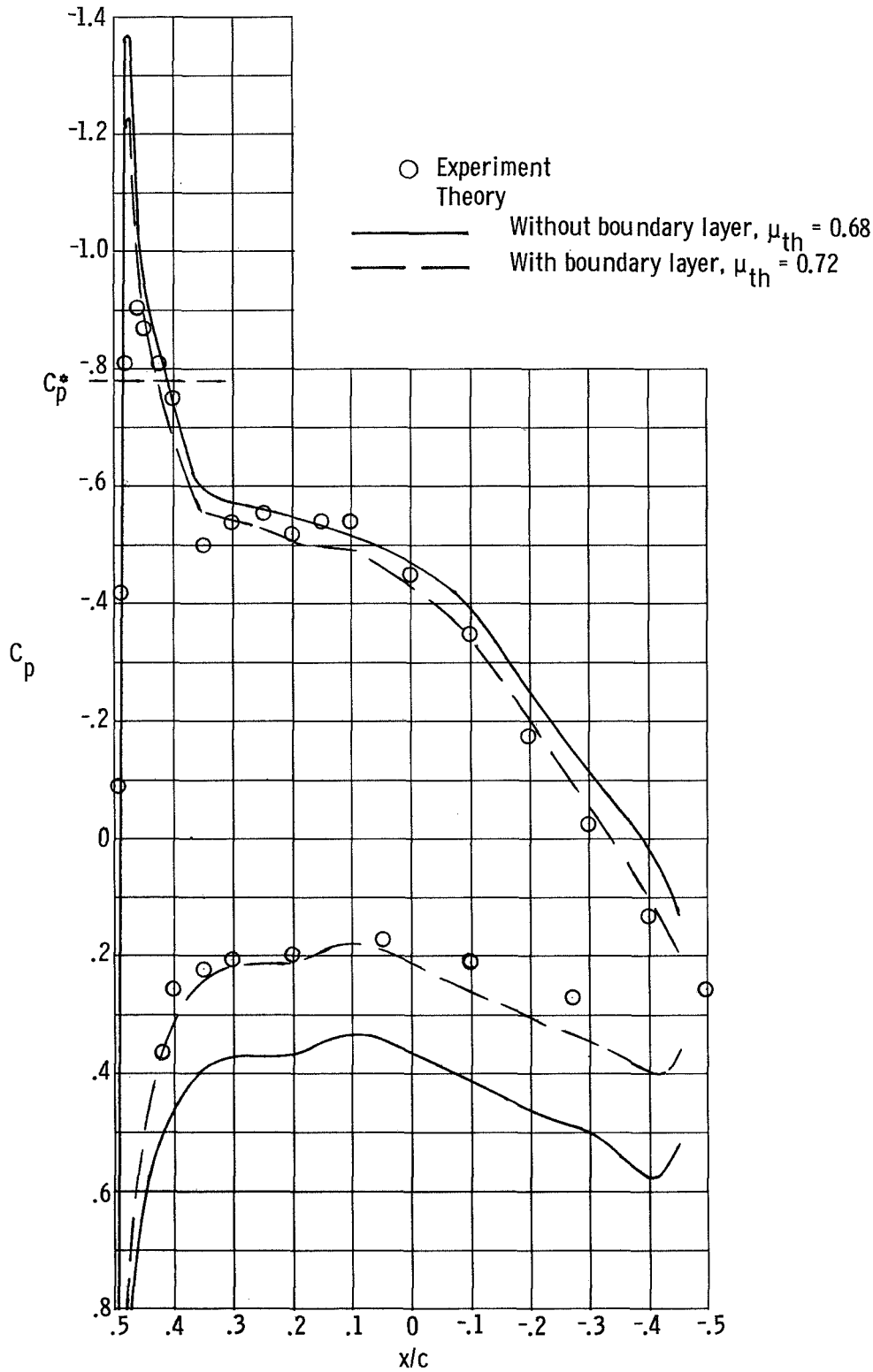


Figure 10.- Pressure distribution on nacelle 2; $M_\infty = 0.7$; $\left(r_b/c\right)_{max}$ = 12.61 percent; $\mu_{exp} = 0.74$. Boundary layer: Five iterations, least square 80- to 97-percent c.

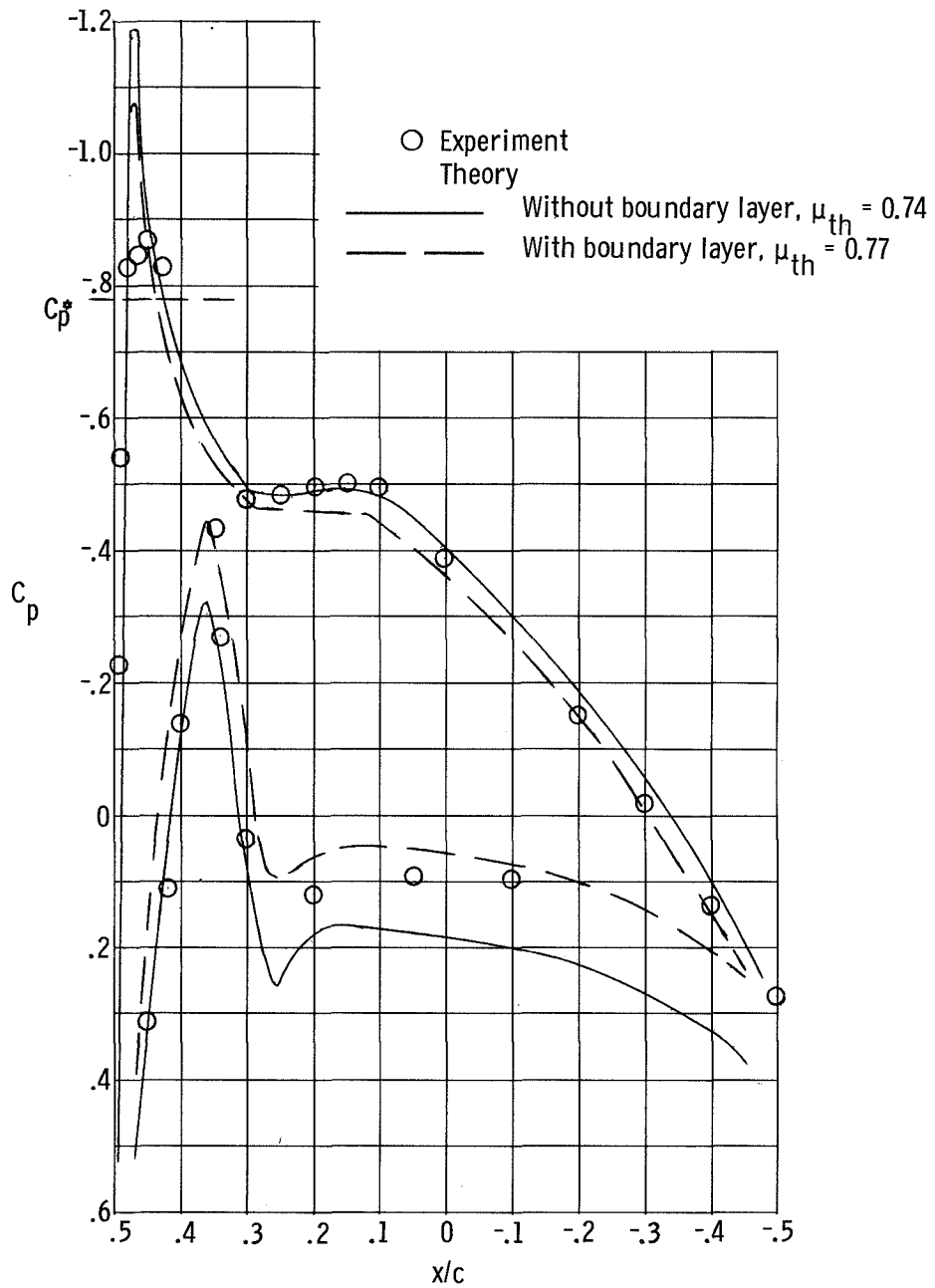


Figure 11.- Pressure distribution on nacelle 3; M_∞ = 0.7; $\left(r_b/c\right)_{max}$ = 12.79 percent; μ_{exp} = 0.76. Boundary layer: Five iterations; least square 80- to 97-percent c.

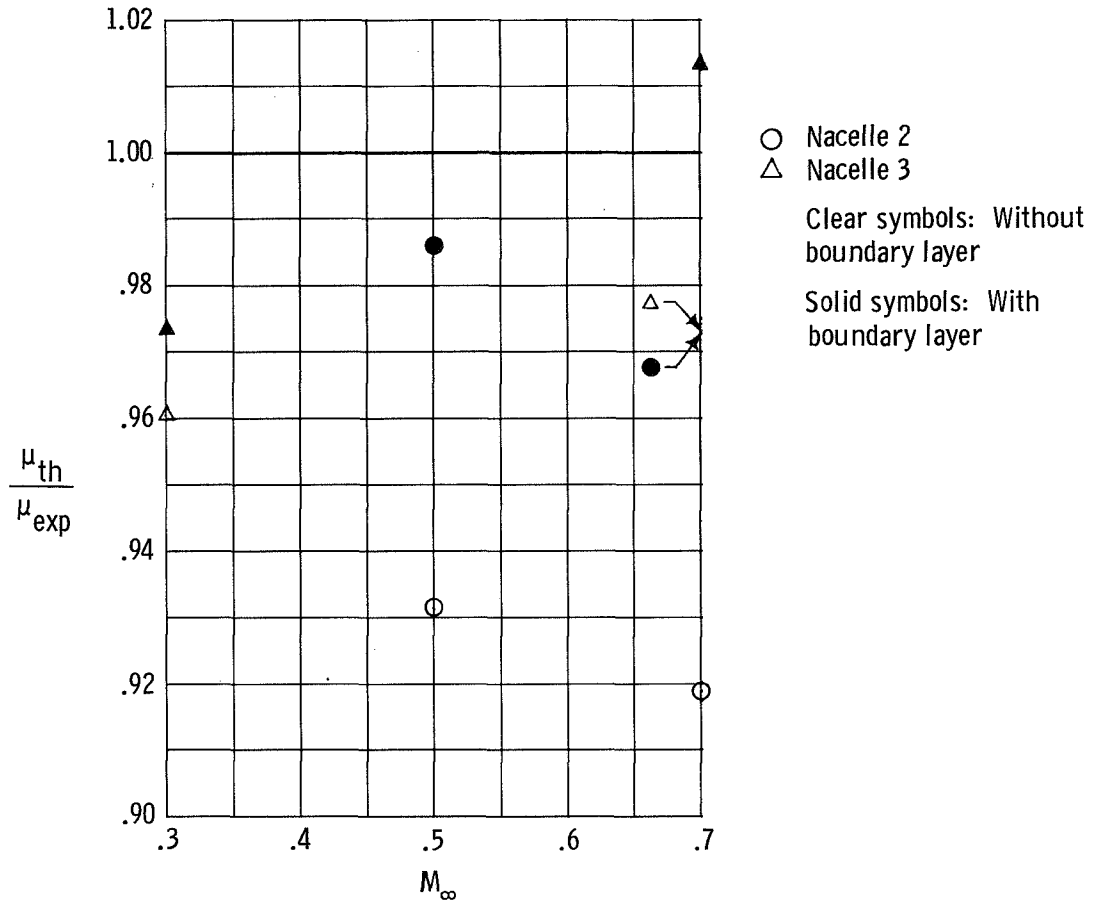
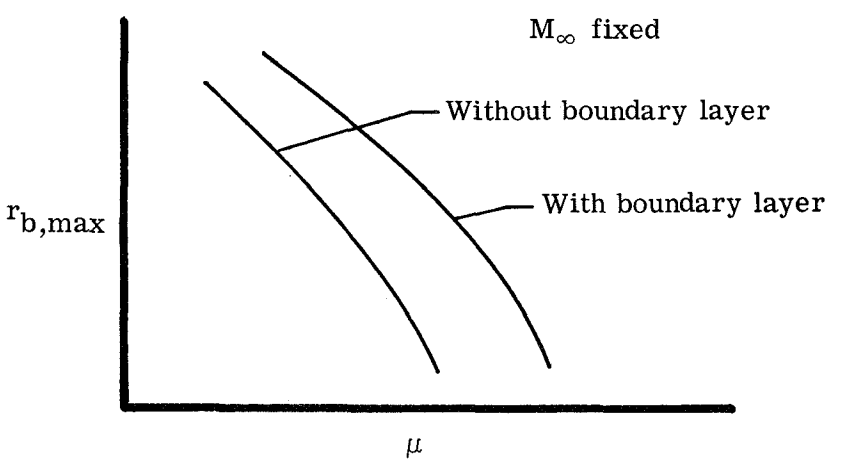


Figure 12.- Summary of calculated mass-flow ratios for cases where internal blockage was estimated, including effect of boundary layer.

the outer surface. Thus far ahead of the body the capture streamline for the internal mass flow is at a larger radius and more flow is swallowed by the wing. A summary of the effect of the boundary layer on the mass flow is shown in the following sketch:



Both methods of selecting the center-body size show good results and neither seems superior to the other. The center-body size used in figures 8 to 11 was estimated from information given in reference 5. The accuracy with which the mass flow was calculated is summarized in figure 12 which shows that inclusion of the boundary layer in the calculations improved this accuracy.

Figures 13 and 14 compare the present theory with the methods of references 5 and 10 for the flow condition of figure 4. Figure 13 is without the boundary layer and figure 14 is with it. All methods are seen to give very close to the same results; therefore, the use of surface singularities gives essentially the same result in this case as the present method, which uses singularities placed on a mean diameter cylinder.

When comparing different theoretical solutions of the flow over a nacelle, caution must be exercised to ensure that the mass flow from each solution is the same. Because the mass flow is a function of the circulation, different methods of approximating the Kutta condition can result in different mass flows on the same geometry.

The experiments of figures 4 to 11 were run without fan blades present (highest mass-flow runs). When fan blades were present, it was found that the straight center-body model used previously was not sufficient to calculate accurately the internal nacelle surface pressures. Hence the center body was modified to approximate the experimental sting up to the fan-blade location, which was estimated to be at about the 75-percent-chord location (the exact sting diameter and fan-blade location were not given in ref. 5). In this region the fan blade blockage was simulated by expanding the center body into an ellipsoid with a circular cross section (see fig. 3(a)). The ellipse in the x-r plane had a semi-minor axis of about 13 percent of the chord and a semimajor axis (r-direction), which was adjusted to give the experimental mass flow. Where the ellipsoid intersected the plane of the nozzle exit, a constant-diameter cylinder was extended to infinity. Figures 15 and 16 show the results of using this model for two low-mass-flow cases. The external pressure distributions are well predicted in spite of the discrepancies on the internal surface. Furthermore a fairly reasonable estimate of the internal pressure distribution is obtained. In figure 16 some large experimental errors may occur in μ because of internal-flow separation as mentioned previously. This would explain the discrepancy at the leading-edge suction peak.

Calculations were also compared with experimental data obtained from reference 20. The investigation of reference 20 was made on several NACA 1-series inlet contours tested over a range of subsonic and transonic Mach numbers, mass-flow ratios, and angles of attack. The thickness ratio $(t/c)_{max}$ of the two nacelles of reference 20 selected for the present study was much smaller than the thickness ratio of nacelles 2 and 3 (2 percent compared with 11 percent). As shown in figure 3(a), the nacelles were sting supported with a throttle plug at the rear of the model for controlling the mass flow (the strut arrangement

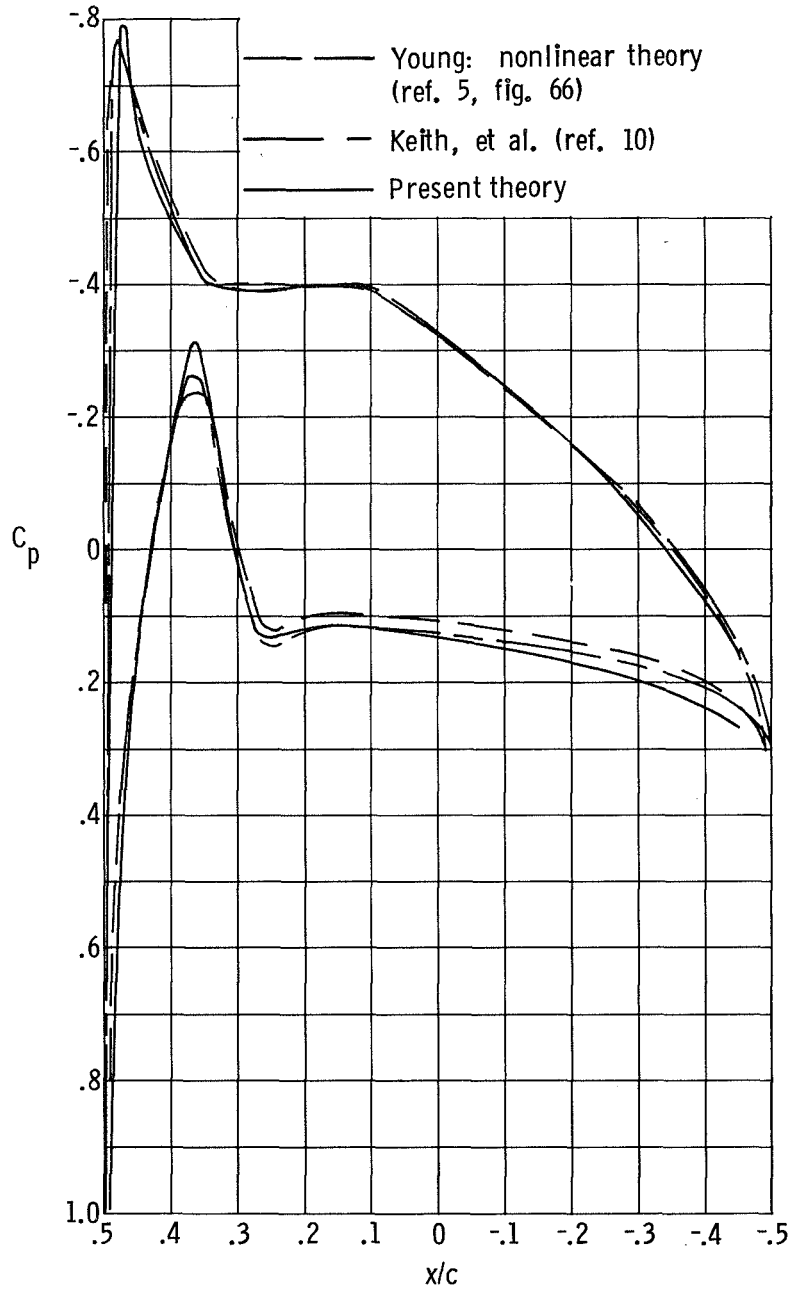


Figure 13.- Pressure distribution on nacelle 3; $M_\infty = 0.3$; $\mu_{th} = 0.76$. All theories without boundary layer.

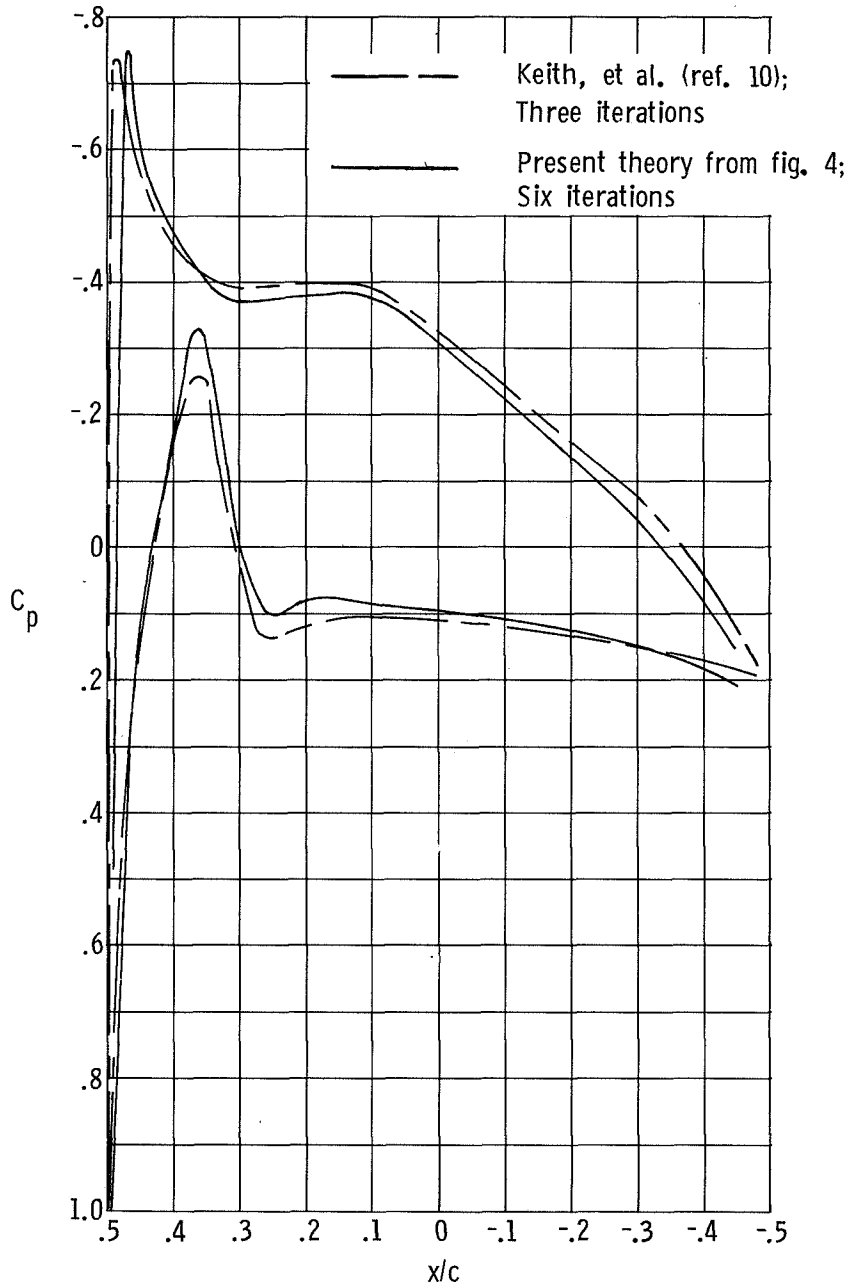


Figure 14.- Pressure distribution on nacelle 3; $M_\infty = 0.3$; $\mu_{th} = 0.76$. Both theories with boundary layer.

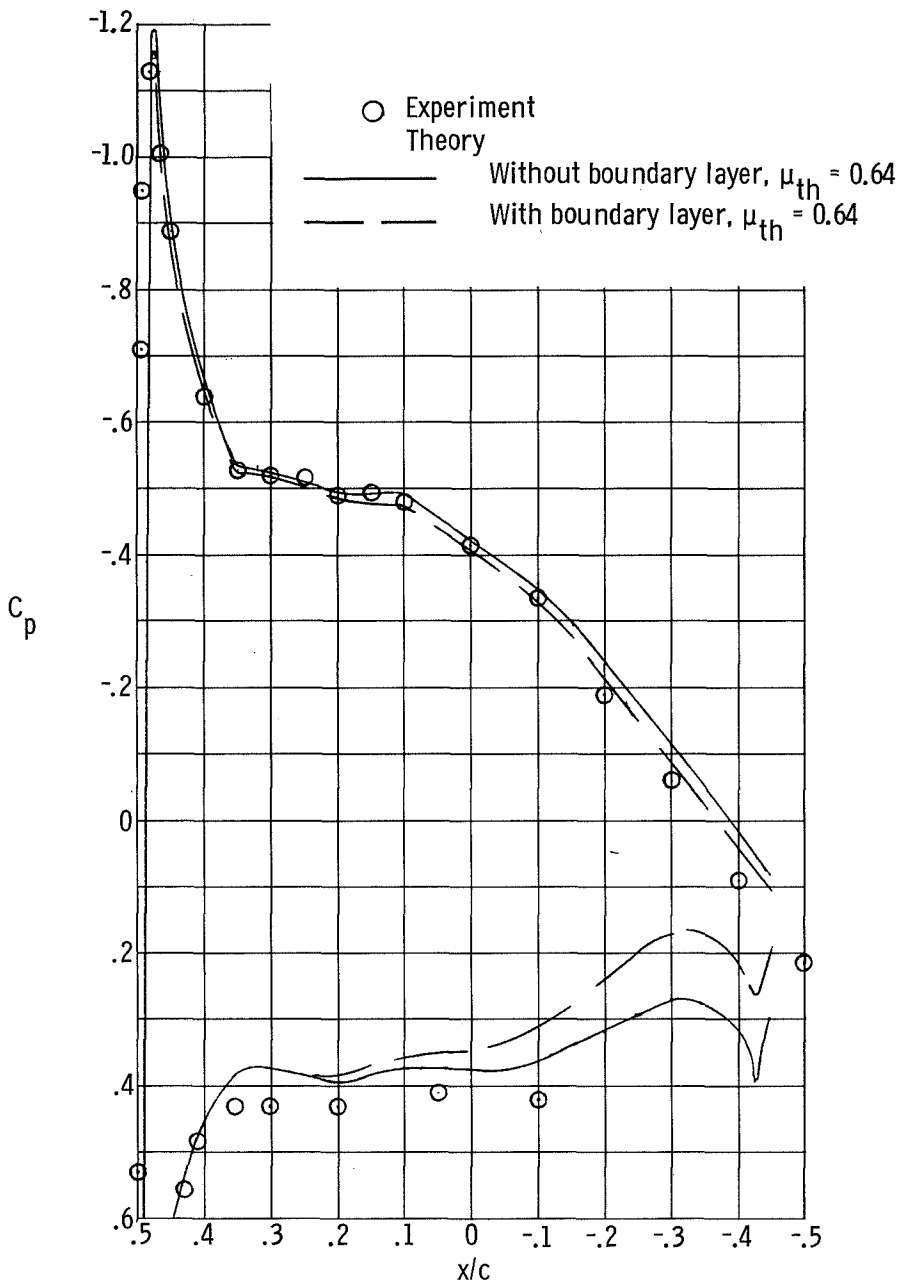


Figure 15.- Pressure distribution on nacelle 2; $M_\infty = 0.5$; $\mu_{th} = \mu_{exp} = 0.64$. Boundary layer: Five iterations; least square 90- to 97-percent c.

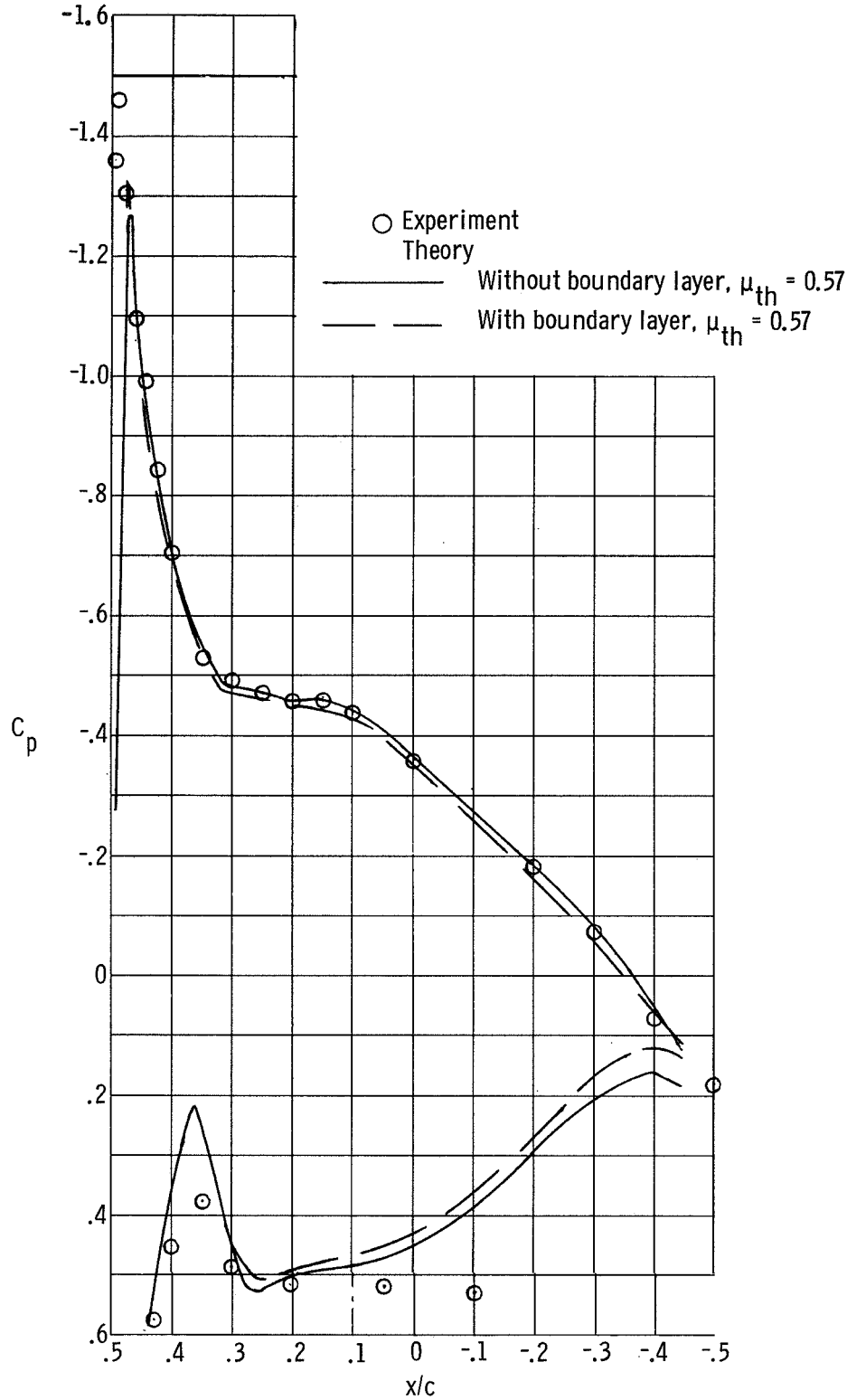


Figure 16.- Pressure distribution on nacelle 3; M_∞ = 0.30; $\mu_{th} = \mu_{exp}$ = 0.57. Boundary layer: Five iterations; least square 90- to 97-percent c.

differed from that of fig. 3(a) in that fore and aft sets of struts were used). The plug was moved forward and backward to vary the mass flow. As previously, the blockage of the sting, struts, and plug was represented in the theory by the blockage of a center body. High-mass-flow ratios were selected for study to minimize the effect of the plug on the afterbody pressures.

The models of reference 20 consisted of the same afterbody attached to various inlets. The inlets had NACA 1-series outer profiles and various inner profiles depending upon the contraction ratio. The two inlets selected for the present study were the NACA 1-81-100 with a contraction ratio of 1.012 and the NACA 1-85-100 with a contraction ratio of 1.009. The numbers in the NACA designation indicate first the series, then d/D_{max} in percent, and finally X/D_{max} in percent. The quantity d is the highlight diameter less twice the inlet lip radius; X is the inlet length; and D_{max} is the inlet (and nacelle, in this case) maximum diameter.

The nose of the center body in the theoretical model was again placed at the position of the sting leading edge and had the same shape as used previously. (See fig. 3(a).)

The tests of reference 20 were conducted without artificial boundary-layer transition. A limited amount of data obtained with a transition band located 2.54 cm aft of the lip (the maximum nacelle diameter was 45.72 cm) showed no difference in pressure when compared with free transition data. Thus, it again seemed reasonable to assume a turbulent boundary layer over the entire chord length for the viscous calculations. The Reynolds number was a function of Mach number. For M_∞ of 0.40 and 0.80 the Reynolds number was $7.437 \times 10^4/\text{cm}$ and $1.159 \times 10^5/\text{cm}$, respectively.

The comparison between theory and experiment is shown in figures 17 and 18 for M_∞ of 0.80 and 0.40. The theoretical center-body size was selected to give the mass-flow ratio measured in the experiment. The data on the inner surface and the data labeled "other" were obtained from the author of reference 20. The data points labeled "other" are an average of measurements made at that x/c location on the sting, mass-flow rake, and inner surface (these measurements were not separately available). Good agreement was obtained on the outer surface for both cases when the boundary layer was included. The very large adverse pressure gradient on the inner surface of the inlet in figure 17 occurs at the higher Mach numbers. The boundary-layer calculation indicated flow separation in this region. Thus, the boundary layer on the outer surface did not converge to any better than a maximum change in δ^* of 12 percent of the maximum δ^*. This lack of convergence was due to erroneous changes in the thickness and camber caused primarily by incorrect values of δ^* on the inner surface. Hence, flow separation and a possible influence of the mass-flow plug on the measured pressures at the trailing edge could explain the discrepancies between the theory and experiment in figure 17.

In figure 18 the boundary layer converged on both surfaces to a maximum change in δ^* of less than 0.2 percent of the maximum δ^* for the respective surfaces. The

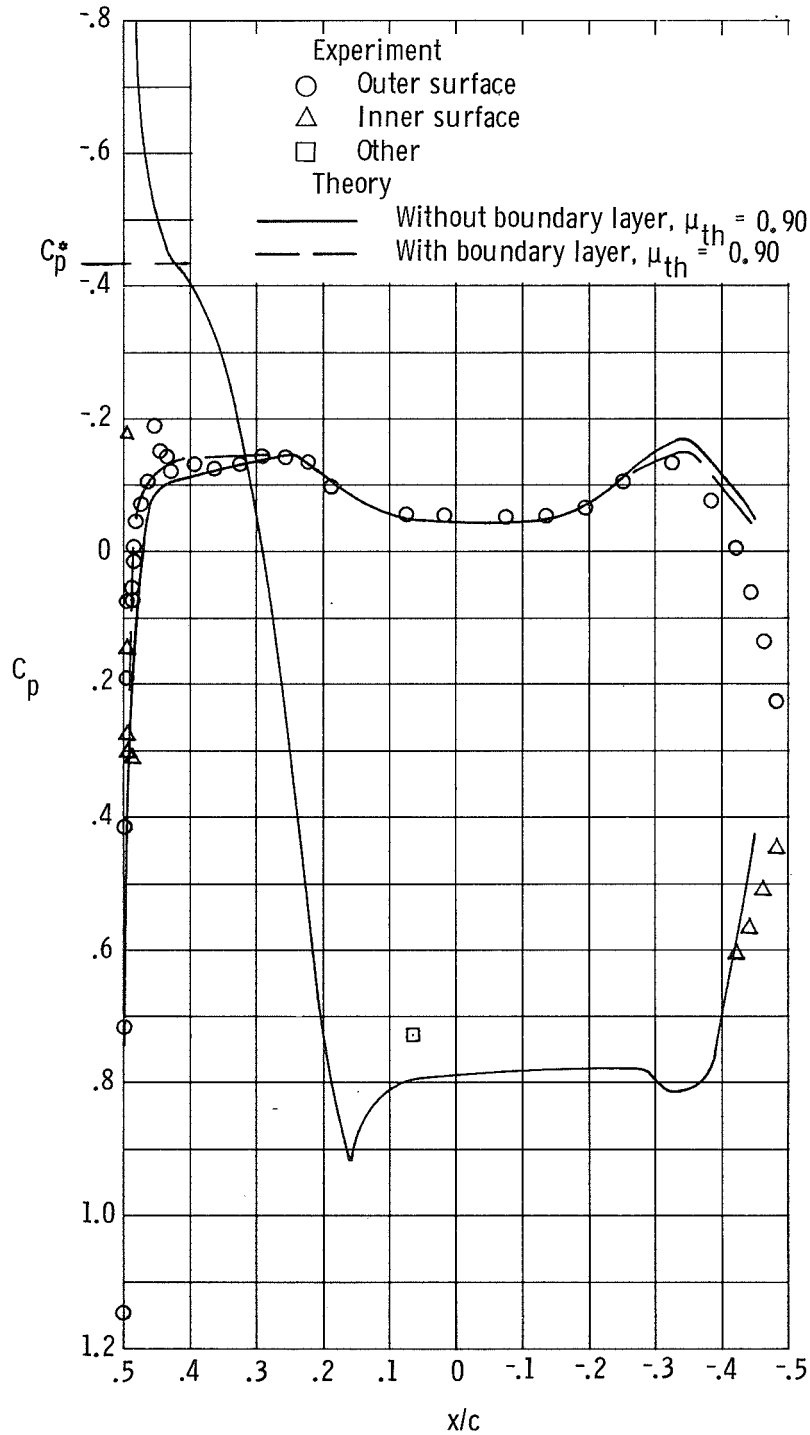


Figure 17.- Pressure distribution on NACA nacelle 1-81-100; contraction ratio, 1.012; M_∞ = 0.80; $\mu_{th} = \mu_{exp}$ = 0.901. Boundary layer: Three iterations; least square 90- to 97-percent c.

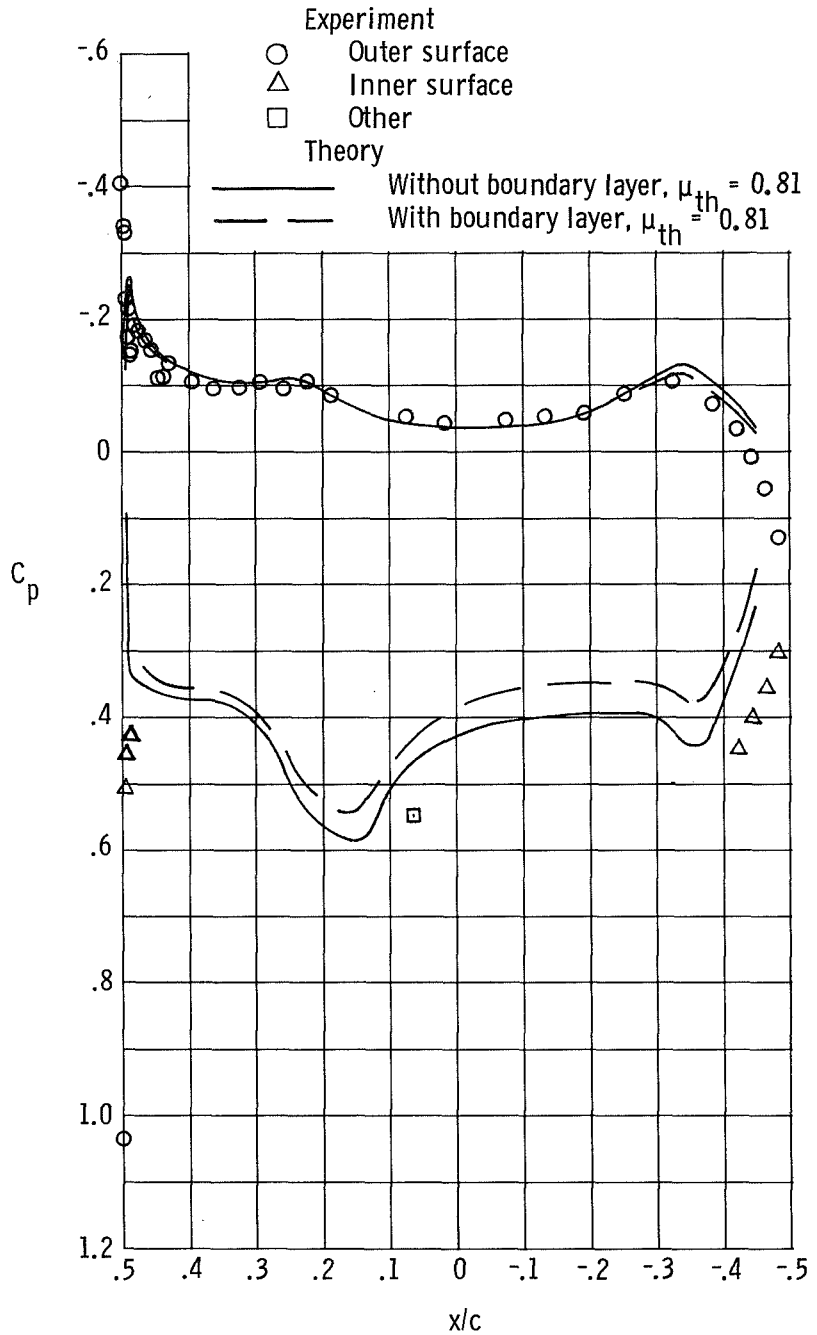


Figure 18.- Pressure distribution on NACA nacelle 1-85-100; contraction ratio, 1.009; M_∞ = 0.40; $\mu_{th} = \mu_{exp}$ = 0.815. Boundary layer: Six iterations; least square 90- to 97-percent c.

discrepancy at the leading-edge suction point could be due to inaccuracies in the leading-edge shape. The pressure in this region is very sensitive to slight errors in the manufacture. The inner surface pressures are approximately predicted by the theory. The calculated velocities in this region are somewhat higher than the actual velocities because the center-body radius used in the theory was larger than the sting radius.

Finally, figure 19 shows the accuracy of calculating the critical Mach number on nacelle 2 by using von Kármán's method (ref. 18). The free-stream Mach number must be low enough to assume incompressible flow. The critical conditions are based on the external pressure distribution. The curve labeled "Minimum C_p" is from the leading-edge suction peak. The intersection of this curve and the curve for C_p^* gives a critical Mach number for nacelle 2 of 0.632. The method of von Kármán yields a critical Mach number of 0.658.

In the case of a propulsive device the addition of energy to the internal flow will have an effect on the pressure distribution over the nacelle. Energy addition is treated in references 10 and 22. Shollenberger (ref. 22) has made a theoretical and experimental study of a two-dimensional propulsive lifting system. The external pressure distribution was found to have a relatively weak dependence on the intensity of energy addition to the internal flow; however, the inner surface pressure distribution was strongly affected. The theoretical formulation of the present study could be extended to include energy addition by use of a method similar to that of reference 22.

CONCLUSIONS

A method analogous to classical airfoil theory has been developed for computing the pressure distribution on an annular wing with a center body at zero angle of attack and subsonic speeds. The following conclusions are drawn:

1. For the geometries investigated, this method was found to give essentially the same results as a method using a surface distribution of singularities and a stream-tube method.

2. Comparison of the results with experimental data for a range of flow conditions and thickness ratios showed that when the boundary-layer displacement effects are included in the calculations, the pressure distribution and mass flow can be calculated with good accuracy. An accurate prediction is dependent on the existence of attached flow and a fairly accurate modeling of the internal-blockage geometry.

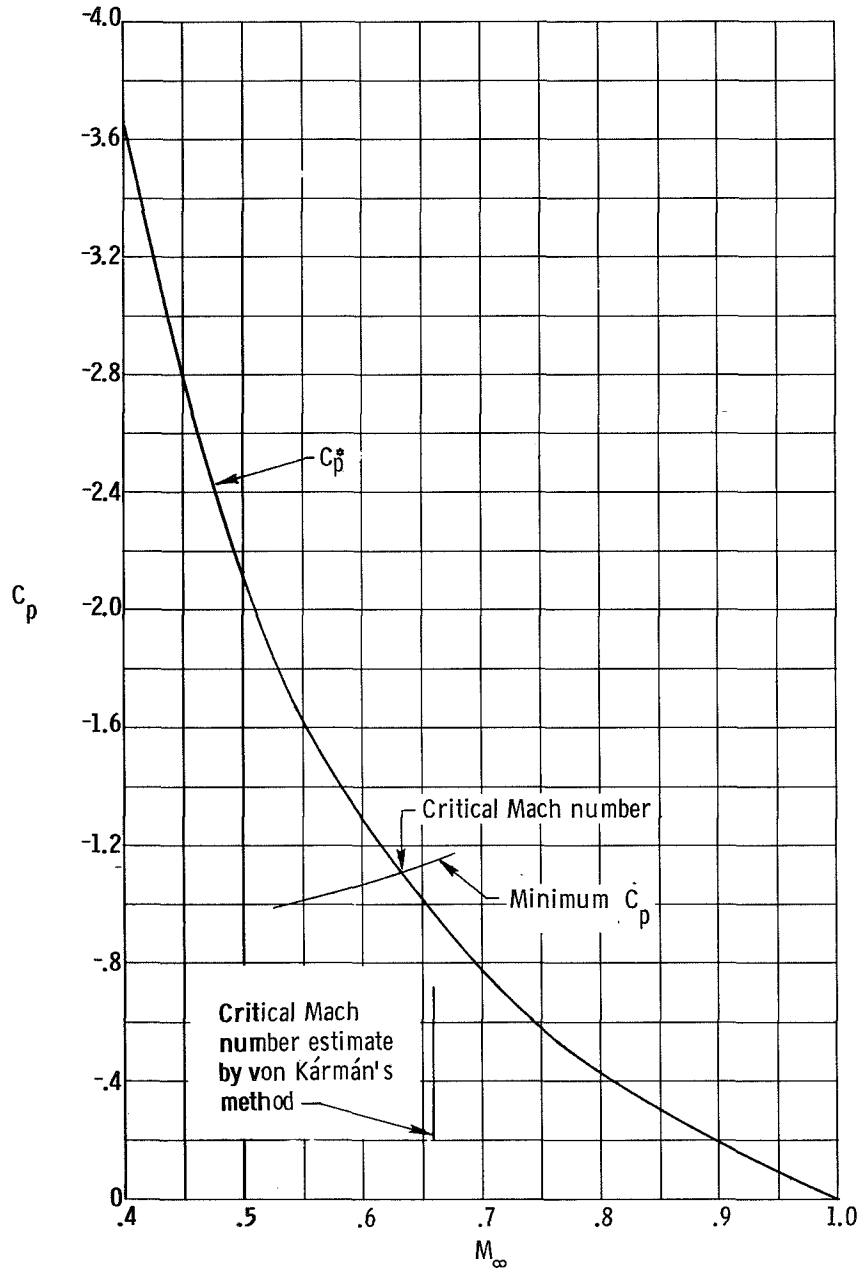


Figure 19.- Critical Mach number for nacelle 2 with $\left(r_b/c\right)_{max}$ = 3.354 percent.

3. For cases in which the internal blockage was not accurately modeled or flow separation may have been present, the inner pressures were not accurately calculated. However, the outer pressures were still calculated to a fairly good degree of accuracy. If internal flow separation occurs, the experimentally measured mass flow will be in error, and this error must be accounted for when comparing experimental and theoretical results.

4. When calculating the boundary-layer thickness, the inviscid velocities were extrapolated to the trailing edge. A more accurate calculation would include the effect of the wake.

5. The method requires modest computer running times and storage and can be generalized to include such effects as lift and internal energy addition.

Langley Research Center,
National Aeronautics and Space Administration,
Hampton, Va., May 9, 1974.

APPENDIX A

COMPUTER PROGRAM

Input Variables

The input data format is the Namelist method used by the Control Data series 6600 computer system at the Langley Research Center. The input variables are listed in this appendix and the input format can easily be modified to any other format. The body radius and wing inner and outer coordinates are each listed in sequential order to correspond with the x-values listed. The x-values are input from the leading edge to the trailing edge. The first quantities in the arrays X, RB, RI, and RO are X(1), RB(1), RI(1), and RO(1), respectively. The positive direction for a radius is shown in figure 1; however x is measured from the leading edge, positive downstream. Any length units can be used so long as the same units are used for all variables. Meters are indicated, however, for dimensional quantities.

The program uses 1.4 for γ, the ratio of specific heats; however, a gas which has a value of γ different from 1.4 can be treated by changing γ in the pressure-coefficient equations and the mass-flow equation. Incompressible cases can be run by simply inputting zero for the free-stream Mach number. Special cases, such as camber with zero thickness or thickness with zero camber, can be treated by simply inputting the correct outer and inner surface coordinates. Some limitations are discussed. (See CAM.) For no center body, input all values of RB equal to 0 or set SCALE equal to 0 (both RB and SCALE must be input).

AIAM $A_h/A_{min} = \left(D_h/D_{min}\right)^2$

C wing chord, m

CAM use 1. when there is camber; use 0. when desire all $\Gamma_0 = 0$, which is an inviscid thickness-only case with no center body. Note that zero geometric camber $\left(\text{all } \alpha_0 = 0\right)$ is just a special case of equation (6). However, because of the method of defining the vortex strengths, v_γ/V_∞ would become indeterminant if $\alpha_0 = 0$. Thus the form of the equation for v_γ/V_∞ will not permit use of equation (6) (as given by eqs. (8) or (12)) to account for the center-body interference for zero camber. Therefore a geometry which has a center body but no geometric camber $\left(\text{all } \alpha_0 = 0\right)$ cannot be treated by this computer program. Likewise, if portions of a cambered wing have a mean camber line with zero slope, a very small slope must be used in these regions. These limitations could be removed by replacing $\alpha_0\gamma_0$ by γ_+/V_∞ in the vortex equations $\left(\text{see definition of } \gamma_0\right)$.

DMAX	maximum wing outside diameter, m
DMIN	minimum wing inside diameter, m
FINI	use 1. for the last data set and 0. for preceding data sets when more than one set in input; one data set is shown in sample input
ITMAX	number of times boundary layer is computed; potential flow is computed ITMAX + 1 times; use 0. for inviscid solution
M	free-stream Mach number
N	number of wing panels in chordwise direction (same N used in text, see pages 10 and 17); maximum value is 170
NN	number of x-stations where quantities in arrays X, RI, RO, and RB are input; maximum value is 170
NRHO	number of annuli internal flow area is divided into to compute mass flow; maximum value is 170
RB	array of center-body radii; for $X \leqq XBN$, use .0 for values of RB, m
RI	array of inner wing radii, m
RO	array of outer wing radii, m
SCALE	scale factor to expand or shrink elements of RB so as to vary mass flow; SCALE is ratio of desired body radius to input body radius
UNRREF	Reynolds number per meter
X	array of x-stations where arrays RI, RO, and RB are input, m
XBN	x-station of center-body nose, m

XAFT, XFOR — Between the x/c values of XFOR and XAFT, a linear least-square-curve fit is applied to outer surface velocities. Then the outer surface velocities are recomputed from XFOR to the trailing edge with this curve fit. Similarly, a parabolic least-square-curve fit is applied to the inner surface velocities between XFOR and XAFT and the Nth value of recomputed outer surface velocity. Then the inner surface velocities are recomputed from XFOR to the trailing edge with this curve fit.

Sample Input

The input for the viscous calculation of figure 4 is shown below. If more than one case is to be run, the complete data set must be repeated by using the proper values for FINI. The last card is identification information and uses format (6A10).

```
$GEOM N=150, NN=55, M=.30, C=4.4101, DMIN=3.9395, DMAX=5., NRHO=170,
XFOR=.90, XAFT=.97,
SCALE=3.125,
ITMAX=6,
AIAM=1.2580,
FINI=1., CAM=1.,
XBN=.8,
UNRREF=453500.,
X=    0.0000,  .0250,  .0450,  .0700,  .0882,  .1200,  .1500,  .1764,  .2646,
       .3528,  .4410,  .5292,  .6174,  .7056,  .7938,  .8000,  .8057,  .8114,
       .8171,  .8228,  .8820,  .9702, 1.0584, 1.1466, 1.2348, 1.3230, 1.4112,
      1.4994, 1.5876, 1.6758, 1.7640, 1.8522, 1.9404, 2.0286, 2.1168, 2.2051,
      2.2933, 2.4256, 2.5138, 2.6020, 2.7343, 2.8666, 2.9548, 3.0871, 3.2194,
      3.3517, 3.5281, 3.6604, 3.7927, 3.9691, 4.1014, 4.1896, 4.2778, 4.3660,
      4.4101,
RI=   2.2093, 2.1758, 2.1467, 2.1170, 2.1019, 2.0822, 2.0694, 2.0603, 2.0307,
      2.0073, 1.9888, 1.9760, 1.9698, 1.9711, 1.9773, 1.9782, 1.9787, 1.9792,
      1.9797, 1.9802, 1.9852, 1.9927, 1.9985, 2.0016, 2.0038, 2.0056, 2.0073,
      2.0095, 2.0117, 2.0140, 2.0162, 2.0184, 2.0206, 2.0228, 2.0246, 2.0268,
      2.0290, 2.0321, 2.0338, 2.0360, 2.0396, 2.0427, 2.0449, 2.0480, 2.0511,
      2.0542, 2.0586, 2.0617, 2.0648, 2.0683, 2.0710, 2.0723, 2.0736, 2.0749,
      2.0749,
RO=   2.2093, 2.2428, 2.2717, 2.3010, 2.3153, 2.3344, 2.3459, 2.3537, 2.3811,
      2.4019, 2.4174, 2.4302, 2.4412, 2.4509, 2.4602, 2.4611, 2.4616, 2.4621,
      2.4626, 2.4631, 2.4682, 2.4757, 2.4823, 2.4876, 2.4920, 2.4956, 2.4982,
      2.4995, 2.5000, 2.4991, 2.4969, 2.4934, 2.4889, 2.4832, 2.4761, 2.4686,
      2.4602, 2.4456, 2.4355, 2.4240, 2.4054, 2.3855, 2.3714, 2.3493, 2.3263,
      2.3020, 2.2671, 2.2389, 2.2106, 2.1730, 2.1443, 2.1257, 2.1067, 2.0877,
      2.0785,
RB=   0.0000, 0.0000, 0.0000, 0.0000, 0.0000, 0.0000, 0.0000, 0.0000, 0.0000,
      0.0000, 0.0000, 0.0000, 0.0000, 0.0000, 0.0000, 0.0000,  .0193,  .0263,
       .0315,  .0358,  .0620,  .0840,  .0980,  .1080,  .1160,  .1230,  .1290,
       .1345,  .1390,  .1415,  .1450,  .1460,  .1480,  .1495,  .1498,  .1500,
       .1500,  .1500,  .1500,  .1500,  .1500,  .1500,  .1500,  .1500,  .1500,
       .1500,  .1500,  .1500,  .1500,  .1500,  .1500,  .1500,  .1500,  .1500,
       .1500,
$
YOUNGS COWL 3
```

Output Variables

Most of the input quantities are printed and identified with the symbols of the preceding section. Other quantities have the meanings listed in this section. In the output for the wing surface solution, all perturbation velocities, including center-body velocities are for the equivalent incompressible flow and are on the wing surface $\left(\rho_o = 1\right)$. In the mass-flow computation output the perturbation velocities and the RAD are also for the equivalent incompressible flow. The pressure coefficients, total velocities, terms of equation (42), MFC and MU are for the compressible flow. Other output sections are either labeled as to compressibility or the nature is obvious (for example, GQ, etc. are obviously for the incompressible flow). The coordinate system of figure 1 is used except in the case of the center-body slopes as noted in the following list. Output which follows the boundary layer output is for the wing shape modified with the final δ^* distribution.

ALPHO α_o at ξ_i, rad

ALPHOO α_o at ξ_{oj} in first listing, rad; $-2\pi\left(\alpha_{oj} + \frac{v_{bj}}{V_\infty}\right)$ in second listing

ALPHQBB left-hand side of equation (33) at ξ_{oj}

ALPT α_t at ξ_i, rad

ALPTH α_t at ξ_{oj}, rad

CPO C_p on wing outer surface

CPI C_p on wing inner surface

CPBA C_p on center body (only approximate because the source strength does not correspond to the input center-body shape owing to neglect of wing interference on the center body)

CPBI isolated center body C_p; this C_p would be theoretically correct if the wing were absent

DDEL $\delta^*_m - \delta^*_{m-1}$, m

DELI δ^* from equation (44) on wing inner surface, m

DELO δ^* from equation (44) on wing outer surface, m

DELST δ^*_m, m

DRDX dr/dx of center body at ξ_i, with x positive downstream

DRDXH dr/dx of center body at ξ_{oj}, with x positive downstream

GAM Γ_{oi}

GQ γ_t / V_∞ at ξ_i

GQB $\bar{g}_q$ at ξ_i

H shape factor, δ^*/θ

I i

IT number of times boundary layer has been computed; IT - 1 equals iteration number

J j

LAM λ on equivalent incompressible wing

MCR critical Mach number

MFC $m / \rho_\infty A_{min} V_\infty$ where $A_{min} = \pi D^2_{min} / 4$

MU μ

RAD radii of annuli in mass-flow computation, m

RIN RI - DELI (input values of RI used), m

ROUT RO + DELO (input values of RO used), m

RHOO ρ_{oj}

RHOO-BODY ρ_b at ξ_{oj}

TEMP10 quantity under summation sign of equation (42) at a given value of ρ_{oj}

THE θ, m

UOURO V/V_∞ on wing outer surface at X-stations, used to compute boundary layer

UOURI V/V_∞ on wing inner surface at X-stations, used to compute boundary layer; for $x/c < 0.02$, UOURI is set equal to UOURO(1) to obtain more reasonable values

VTO $\left(V/V_\infty\right)^2$ on outer surface of wing

VTI $\left(V/V_\infty\right)^2$ on inner surface of wing first time printed; $\left[\left(V_\infty + u\right)/V_\infty\right]^2$ at ρ_{oj} second time printed (square of velocity term under summation sign in eq. (42))

VIVIN summation of equation (42) evaluated from $j = 1$ to the value of J listed for any given value of VIVIN. The summation progresses from wing inner surface toward center line, or center body if one is present, at fixed axial location. This sum is evaluated at $\xi_o = 0$ (midchord) presently but this station can be changed by changing card numbers T170, T320, T650, and T870 in program MASFLOW. The last value of VIVIN equals VI/VIN.

VI/VIN $V_{\xi o}/V_\infty$ of equation (42)

WX1JUOO u_γ/V_∞ at ξ_{oj} on wing outer surface $\left(\rho_{oj} = 1\right)$

WX1JUOI u_γ/V_∞ at ξ_{oj} on wing inner surface $\left(\rho_{oj} = 1\right)$

WX1JUO summation in equation (13); for points where $\rho_{oj} \neq 1$ (required for computing mass flow), this quantity is u_γ/V_∞ at ξ_{oj},ρ_{oj}

WR1JUO v_γ/V_∞ at ξ_{oj},ρ_{oj}

WX1QJUO $u_{t,\gamma}/V_\infty$ at ξ_{oj} on wing outer surface $\left(\rho_{oj} = 1\right)$

WX1QJUI $u_{t,\gamma}/V_\infty$ at ξ_{oj} on wing inner surface $\left(\rho_{oj} = 1\right)$

WX1QJU same as WX1JUO only with respect to $u_{t,\gamma}/V_\infty$

WR1QJU $v_{t,\gamma}/V_\infty$ at ξ_{oj},ρ_{oj}

WXQJU u_q/V_∞ at ξ_{oj},ρ_{oj}

WRQJUO v_q/V_∞ at ξ_{oj} on wing outer surface $\left(\rho_{oj} = 1\right)$

WRQJUI v_q/V_∞ at ξ_{oj} on wing inner surface $\left(\rho_{oj} = 1\right)$

WRQJU summation in equation (35); for points where $\rho_{oj} \neq 1$ (required for mass flow), this quantity is v_q/V_∞ at ξ_{oj},ρ_{oj}

WXBQJU u_b/V_∞ at ξ_{oj},ρ_{oj}

WRBQJU v_b/V_∞ at ξ_{oj},ρ_{oj}

XI ξ_i

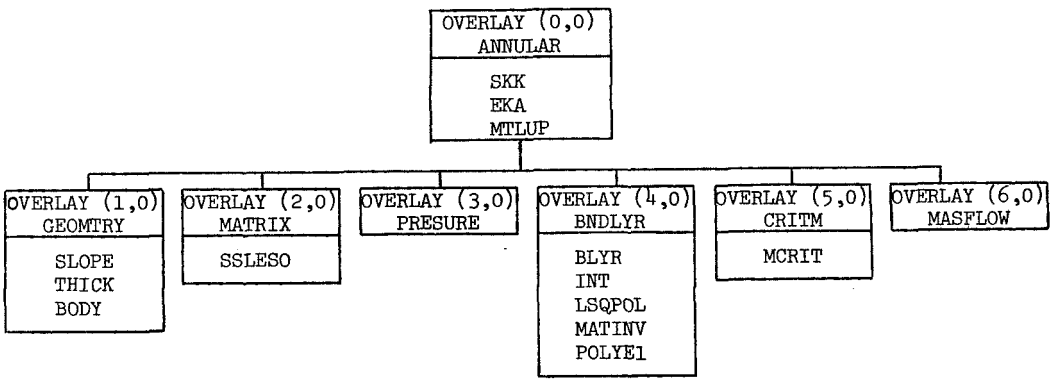
XIO ξ_{oj}

XOJ/C x_{oj}/c (x of fig. 1)

Program Structure and Listing

This program is written in FORTRAN IV, Version 2.3 for the Control Data series 6600 computer system with the run compiler and execution routines operating under SCOPE 3.0. The program requires $65{,}000_8$ words of storage.

The program operates in the OVERLAY mode. The main overlay program is called ANNULAR and has six primary overlay programs - GEOMTRY, MATRIX, PRESURE, BNDLYR, CRITM, and MASFLOW. The overlay structure and subprogram arrangement are shown in the following sketch:



The function LOCF(A) used by subroutine MTLUP is a library function which returns the address of argument A. No listing is provided for LOCF.

Numerous comment cards have been inserted in the main overlay to assist the user in understanding the flow of the computations. The purpose of each overlay and subprogram is described by comment cards at the beginning of the overlay or subprogram.

```
      OVERLAY(RING,0,0)                                                    A  10
      PROGRAM ANNULAR(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT,TAPE10)        A  20
C                                                                          A  30
C                                                                          A  40
C     PROGRAM ANNULAR EMPLOYS ANNULAR WING THEORY TO COMPUTE THE           A  50
C     SUBSONIC FLOW OVER NACELLES INCLUDING THE COMPRESSIBILITY EFFECTS     A  60
C     AND VISCOUS EFFECTS                                                  A  70
C                                                                          A  80
C                                                                          A  90
C     ANY ERRORS OR PROBLEMS ENCOUNTERED IN USING THE PROGRAM SHOULD BE    A 100
C     DIRECTED TO DR. MICHAEL J. MANN AT NASA LANGLEY, AREA CODE 804-827   A 110
C     -3711.                                                               A 120
C                                                                          A 130
C                                                                          A 140
C     A CARD DECK AND DOCUMENTATION FOR THE PROGRAM ARE AVAILABLE FROM     A 150
C     COSMIC, UNIVERSITY OF GEORGIA, ATHENS, GEORGIA, 30601.               A 160
C                                                                          A 170
C                                                                          A 180
C     THIS PROGRAM IS WRITTEN IN CDC FORTRAN IV, VERSION 2.3, TO RUN ON    A 190
C     CDC 6600 SERIES COMPUTERS WITH THE SCOPE 3.0 OPERATING SYSTEM AND    A 200
C     LIBRARY TAPE.                                                        A 210
C                                                                          A 220
C                                                                          A 230
      COMMON /CORD/ X(170),R(170),C,RAVG,NN,N,DMIN,T(170),RB(170),XBN      A 240
      COMMON /ONE/ ROUT(170),RIN(170),M,DMAX,XI(170),XIO(170),ALPHO(170)   A 250
     1,ALPHOO(170,1),XBL(170),ALPT(170),ALPTH(170),DRDX(170),DRDXH(170),   A 260
     2LAM,IT,ITMAX                                                         A 270
      COMMON /TWO/ IP,CAM,GAM(170),GQ(170),ALPOO(170,1)                    A 280
      COMMON /THREE/ VOUT(170),VIN(170)                                    A 290
      COMMON /FOUR/ XOVC(170),RO(170),RI(170),DELO(170),DELI(170),WLS(40   A 300
     1),DOLDO(170),DOLDI(170),UNRREF,SAM,XFOR,XAFT                         A 310
      COMMON /FIVE/ CPO(170),MCR                                           A 320
      COMMON /SIX/ NRHO,RAD(170),VIVIN,RXX,RBXI                            A 330
      DIMENSION IDENT(6)                                                   A 340
      REAL LAM,KSQ,M,MFC,MCR,JOHN,MU                                       A 350
C                                                                          A 360
C     INPUT                                                                A 370
C                                                                          A 380
      NAMELIST /GEOM/ N,NN,M,C,DMIN,DMAX,X,RO,RI,FINI,CAM,NRHO,RB,XBN,SC   A 390
     1ALE,UNRREF,ITMAX,AIAM,XFOR,XAFT                                      A 400
10    READ (5,GEOM)                                                        A 410
      READ (5,100) IDENT                                                   A 420
      WRITE (6,80)                                                         A 430
      WRITE (6,100) IDENT                                                  A 440
      WRITE (6,90) C,DMIN,DMAX,NRHO,UNRREF,ITMAX,N,M,AIAM,XFOR,XAFT,(X(I   A 450
     1),RO(I),RI(I),RB(I),I=1,NN)                                          A 460
      DO 20 I=1,40                                                         A 470
      WLS(I)=1.                                                            A 480
20    CONTINUE                                                             A 490
      DO 30 I=1,NN                                                         A 500
      XOVC(I)=X(I)/C                                                       A 510
      ROUT(I)=RO(I)                                                        A 520
      RIN(I)=RI(I)                                                         A 530
      DELO(I)=0.                                                           A 540
      DELI(I)=0.                                                           A 550
      DOLDI(I)=0.                                                          A 560
      DOLDO(I)=0.                                                          A 570
30    CONTINUE                                                             A 580
C                                                                          A 590
C     SCALE CENTER BODY                                                    A 600
```

```
C                                                                       A 610
      DO 40 I=1,NN                                                      A 620
      RB(I)=RB(I)*SCALE                                                 A 630
40    CONTINUE                                                          A 640
      WRITE (6,120) SCALE,(X(I),RB(I),I=1,NN)                           A 650
C                                                                       A 660
C     COMPUTE EQUIVALENT INCOMPRESSIBLE CENTER BODY                     A 670
C                                                                       A 680
      DO 50 I=1,NN                                                      A 690
      RB(I)=RB(I)*SQRT(1.-M**2)                                         A 700
50    CONTINUE                                                          A 710
      IP=-1                                                             A 720
      SAM=0.                                                            A 730
      IT=1                                                              A 740
C                                                                       A 750
C     COMPUTE NACELLE MEAN CAMBER AND THICKNESS AND EQUIVALENT          A 760
C     INCOMPRESSIBLE SHAPE - THEN COMPUTE NACELLE CAMBER AND THICKNESS  A 770
C     SLOPES AND CENTER BODY SLOPES                                     A 780
C                                                                       A 790
60    CONTINUE                                                          A 800
      CALL OVERLAY (4HRING,1,0,6HRECALL)                                A 810
C                                                                       A 820
C     COMPUTE CAMBER AND THICKNESS VORTEX STRENGTHS - GAM(I) AND GQ(I)  A 830
C                                                                       A 840
      CALL OVERLAY (4HRING,2,0,6HRECALL)                                A 850
C                                                                       A 860
C     COMPUTE NACELLE AND CENTER BODY SURFACE PRESSURES                 A 870
C                                                                       A 880
      CALL OVERLAY (4HRING,3,0,6HRECALL)                                A 890
C                                                                       A 900
C     TEST FOR INVISCID ONLY SOLN OR END OF BOUNDARY LAYER ITERATION    A 910
C                                                                       A 920
      IF (IT.GT.ITMAX) GO TO 70                                         A 930
C                                                                       A 940
C     EXTRAPOLATE FOR TRAILING EDGE VELOCITIES AND COMPUTE BOUNDARY     A 950
C     LAYER. MODIFY NACELLE COORDINATES FOR DISPLACEMENT THICKNESS.     A 960
C                                                                       A 970
      CALL OVERLAY (4HRING,4,0,6HRECALL)                                A 980
      GO TO 60                                                          A 990
70    CONTINUE                                                          A1000
C                                                                       A1010
C     CRITICAL MACH NUMBER                                              A1020
C                                                                       A1030
      CALL OVERLAY (4HRING,5,0,6HRECALL)                                A1040
C                                                                       A1050
C     COMPUTE VI/VIN BY NACELLE INTERNAL MASS FLOW INTEGRATION          A1060
C                                                                       A1070
      CALL OVERLAY (4HRING,6,0,6HRECALL)                                A1080
C                                                                       A1090
C     CALCULATE MFC                                                     A1100
C                                                                       A1110
      MFC=((RXX**2-RBXI**2)/((DMIN/2.)**2))*VIVIN*((1.+(.2*(M**2)*(1.-VI A1120
     1VIN**2)))**2.5)                                                   A1130
      MU=MFC/AIAM                                                       A1140
      WRITE (6,110) MCR,VIVIN,MFC,MU                                    A1150
C                                                                       A1160
C     CHECK TO SEE IF THERE IS ANOTHER CASE                             A1170
C                                                                       A1180
      IF (FINI.EQ.0.) GO TO 10                                          A1190
      STOP                                                              A1200
```

```
C                                                                               A1210
C                                                                               A1220
80     FORMAT (1H1)                                                             A1230
90     FORMAT (//16X,14HINPUT GEOMETRY//4X,2HC=,F9.3,4X,5HDMIN=,F9.3,4X,5       A1240
      1HDMAX=,F9.3,4X,5HNRHO=,I9/4X,7HUNRREF=,E11.4,4X,6HITMAX=,I3,4X,2HN       A1250
      2=,I3,4X,2HM=,F9.4,4X,5HAIAM=,F9.4,4X,5HXFOR=,F9.4,4X,5HXAFT=,F9.4/      A1260
      3/9X,1HX,8X,2HRO,8X,2HRI,8X,2HRB//(4F10.4))                               A1270
100    FORMAT (6A10)                                                            A1280
110    FORMAT (1H0,4X,4HMCR=,F9.4,4X,7HVI/VIN=,F9.4,4X,4HMFC=,F9.4,4X,3HM       A1290
      1U=,F9.4//4X,56HMCR IS CORRECT ONLY IF M IS LOW ENOUGH FOR INCOMP A       A1300
      2SSUMP)                                                                   A1310
120    FORMAT (1H1/2X,29HCENTER BODY EXPANDED BY SCALE,5X,6HSCALE=,F10.4/       A1320
      1///9X,1HX,8X,2HRB//(2F10.4))                                             A1330
       END                                                                      A1340-
```

```
      FUNCTION SKK (Z)                                                    B  10
C                                                                         B  20
C     FUNCTION SKK COMPUTES THE COMPLETE ELLIPTICAL INTEGRAL OF THE       B  30
C     FIRST KIND K(K)                                                     B  40
C                                                                         B  50
C     S. WAGNER, NASA AMES RESEARCH CENTER, MOFFETT FIELD CALIF. 94035    B  60
C                                                                         B  70
C     REFERENCE C. HASTINGS, APPROXIMATIONS FOR DIGITAL COMPUTERS, P.172  B  80
C                                                                         B  90
C     INPUT Z = K**2                                                      B 100
C                                                                         B 110
      IF (Z) 10,20,30                                                     B 120
10    WRITE (6,60) Z                                                      B 130
      STOP                                                                B 140
20    SKK=1.570796326794896                                               B 150
      RETURN                                                              B 160
30    E=1.0-Z                                                             B 170
      IF (E) 10,40,50                                                     B 180
40    WRITE (6,70) Z                                                      B 190
      STOP                                                                B 200
50    SKK=(((E*.1451196212E-01+.3742563713E-01)*E+.3590092383E-01)*E+.96   B 210
     166344259E-01)*E+1.38629436112-((((E*.441787012E-02+.3328355346E-01   B 220
     2)*E+.6880248576E-01)*E+.12498593597)*E+.5)*ALOG(E)                   B 230
      RETURN                                                              B 240
C                                                                         B 250
C                                                                         B 260
60    FORMAT (1H1,10X,21HFUNCTION SKK  K**2 = ,E16.9)                     B 270
70    FORMAT (1H1,10X,7HK**2 = ,E16.9,13HK(K) INFINITE)                   B 280
      END                                                                 B 290-
```

```
      FUNCTION EKA (Z)                                                    C  10
C                                                                         C  20
C     FUNCTION EKA COMPUTES THE COMPLETE ELLIPTICAL INTEGRAL OF THE       C  30
C     SECOND KIND E(K)                                                    C  40
C                                                                         C  50
C     S. WAGNER, NASA AMES RESEARCH CENTER, MOFFETT FIELD CALIF. 94035    C  60
C                                                                         C  70
C     REFERENCE C. HASTINGS, APPROXIMATIONS FOR DIGITAL COMPUTERS, P.175  C  80
C                                                                         C  90
C     INPUT Z = K**2                                                      C 100
C                                                                         C 110
      IF (Z) 10,20,30                                                     C 120
10    WRITE (6,60) Z                                                      C 130
      STOP                                                                C 140
20    EKA=1.570796326794896                                               C 150
      RETURN                                                              C 160
30    E=1.0-Z                                                             C 170
      IF (E) 10,40,50                                                     C 180
40    EKA=1.0                                                             C 190
      RETURN                                                              C 200
50    EKA=1.+(((.1736506451E-01*E+.4757383546E-01)*E+.6260601220E-01)*E+  C 210
     1.44325141463)*E-(((.526449639E-02*E+.4069697526E-01)*E+.9200180037  C 220
     2E-01)*E+.24998368310)*E*ALOG(E)                                     C 230
      RETURN                                                              C 240
C                                                                         C 250
C                                                                         C 260
60    FORMAT (1H1,10X,21HFUNCTION EKA  K**2 = ,E16.9)                     C 270
      END                                                                 C 280-
```

```
      SUBROUTINE MTLUP (X,Y,M,N,MAX,NTAB,I,VARI,VARD)                    D  10
C                                                                        D  20
C     MULTIPLE TABLE LOOK-UP SUBROUTINE. IT COMPUTES Y=F(X) FROM         D  30
C     A TABLE USING M=1 OR 2 FOR FIRST OR SECOND ORDER INTERPOLATION.    D  40
C                                                                        D  50
C       MULTIPLE TABLE LOOK-UP ON ONE INDEPENDENT VARIABLE TABLE         D  60
C                                                                        D  70
C      USES AN EXTERNAL INTERVAL POINTER (I) TO START SEARCH             D  80
C      I LESS THAN 0 WILL CHECK MONOTONICITY                             D  90
C                                                                        D 100
      DIMENSION VARI(1), VARD(MAX,1), Y(1), V(3), YY(2)                  D 110
      LOGICAL EX                                                         D 120
      IF (M.EQ.0) GO TO 170                                              D 130
      IF (N.LE.1) GO TO 170                                              D 140
      EX=.F.                                                             D 150
      IF (I.GE.0) GO TO 60                                               D 160
      IF (N.LT.2) GO TO 60                                               D 170
C                                                                        D 180
C     MONOTONICITY CHECK                                                 D 190
C                                                                        D 200
      IF (VARI(2)-VARI(1)) 20,20,40                                      D 210
C                                                                        D 220
C     ERROR IN MONOTONICITY                                              D 230
C                                                                        D 240
10    K=LOCF(VARI(1))                                                    D 250
C                                                                        D 260
C     LOCF(X) IS A SYSTEM ROUTINE THAT RETURNS THE ADDRESS OF X          D 270
C                                                                        D 280
      PRINT 190, J,K,(VARI(J),J=1,N)                                     D 290
      STOP                                                               D 300
C                                                                        D 310
C     MONOTONIC DECREASING                                               D 320
C                                                                        D 330
20    DO 30 J=2,N                                                        D 340
      IF (VARI(J)-VARI(J-1)) 30,10,10                                    D 350
30    CONTINUE                                                           D 360
      GO TO 60                                                           D 370
C                                                                        D 380
C     MONOTONIC INCREASING                                               D 390
C                                                                        D 400
40    DO 50 J=2,N                                                        D 410
      IF (VARI(J)-VARI(J-1)) 10,10,50                                    D 420
50    CONTINUE                                                           D 430
C                                                                        D 440
C     INTERPOLATION                                                      D 450
C                                                                        D 460
60    IF (I.LE.0) I=1                                                    D 470
      IF (I.GE.N) I=N-1                                                  D 480
C                                                                        D 490
C     LOCATE I INTERVAL (X(I).LE.X.LT.X(I+1))                            D 500
C                                                                        D 510
      IF ((VARI(I)-X)*(VARI(I+1)-X)) 100,100,70                          D 520
C                                                                        D 530
C     IN GIVES DIRECTION FOR SEARCH OF INTERVALS                         D 540
C                                                                        D 550
70    IN=SIGN(1.0,(VARI(I+1)-VARI(I))*(X-VARI(I)))                       D 560
C                                                                        D 570
C     IF X OUTSIDE ENDPOINTS, EXTRAPOLATE FROM END INTERVAL              D 580
C                                                                        D 590
80    IF ((I+IN).LE.0) GO TO 90                                          D 600
```

```
      IF ((I+IN).GE.N) GO TO 90                                          D 610
      I=I+IN                                                             D 620
      IF ((VARI(I)-X)*(VARI(I+1)-X)) 100,100,80                          D 630
C                                                                        D 640
C     EXTRAPOLATION                                                      D 650
C                                                                        D 660
90    EX=.T.                                                             D 670
100   IF (M.EQ.2) GO TO 120                                              D 680
C                                                                        D 690
C     FIRST ORDER                                                        D 700
C                                                                        D 710
      DO 110 NT=1,NTAB                                                   D 720
110   Y(NT)=(VARD(I,NT)*(VARI(I+1)-X)-VARD(I+1,NT)*(VARI(I)-X))/(VARI(I+ D 730
     11)-VARI(I))                                                        D 740
      IF (EX) I=I+IN                                                     D 750
      RETURN                                                             D 760
C                                                                        D 770
C     SECOND ORDER                                                       D 780
C                                                                        D 790
120   IF (N.EQ.2) GO TO 10                                               D 800
      IF (I.EQ.(N-1)) GO TO 140                                          D 810
      IF (I.EQ.1) GO TO 130                                              D 820
C                                                                        D 830
C     PICK THIRD POINT                                                   D 840
C                                                                        D 850
      SK=VARI(I+1)-VARI(I)                                               D 860
      IF ((SK*(X-VARI(I-1))).LT.(SK*(VARI(I+2)-X))) GO TO 140            D 870
130   L=I                                                                D 880
      GO TO 150                                                          D 890
140   L=I-1                                                              D 900
150   V(1)=VARI(L)-X                                                     D 910
      V(2)=VARI(L+1)-X                                                   D 920
      V(3)=VARI(L+2)-X                                                   D 930
      DO 160 NT=1,NTAB                                                   D 940
      YY(1)=(VARD(L,NT)*V(2)-VARD(L+1,NT)*V(1))/(VARI(L+1)-VARI(L))      D 950
      YY(2)=(VARD(L+1,NT)*V(3)-VARD(L+2,NT)*V(2))/(VARI(L+2)-VARI(L+1))  D 960
160   Y(NT)=(YY(1)*V(3)-YY(2)*V(1))/(VARI(L+2)-VARI(L))                  D 970
      IF (EX) I=I+IN                                                     D 980
      RETURN                                                             D 990
C                                                                        D1000
C     ZERO ORDER                                                         D1010
C                                                                        D1020
170   DO 180 NT=1,NTAB                                                   D1030
180   Y(NT)=VARD(1,NT)                                                   D1040
      RETURN                                                             D1050
C                                                                        D1060
C                                                                        D1070
190   FORMAT (1H1,49H TABLE BELOW OUT OF ORDER FOR MTLUP AT POSITION  ,I  D1080
     15,/31H X TABLE IS STORED IN LOCATION ,06,//(8G15.8))               D1090
      END                                                                D1100
```

```
      OVERLAY(RING,1,0)                                                    E  10
      PROGRAM GEOMTRY                                                      E  20
C                                                                          E  30
C     THIS OVERLAY COMPUTES NACELLE MEAN CAMBER RADII AND THICKNESS        E  40
C     FOR THE EQUIVALENT INCOMPRESSIBLE SHAPE.  THEN THE SLOPES ARE        E  50
C     COMPUTED ON THE EQUIVALENT NACELLE CAMBER AND THICKNESS SHAPES       E  51
C     AND ON THE EQUIVALENT CENTER BODY.                                   E  52
C                                                                          E  60
      COMMON /CORD/ X(170),R(170),C,RAVG,NN,N,DMIN,T(170),RB(170),XRN      E  70
      COMMON /ONE/ ROUT(170),RIN(170),M,DMAX,XI(170),XIO(170),ALPHO(170)   E  80
     1,ALPHOO(170,1),XBL(170),ALPT(170),ALPTH(170),DRDX(170),DRDXH(170),   E  90
     2LAM,IT,ITMAX                                                         E 100
      REAL LAM,M                                                           E 110
C                                                                          E 111
C     COMPUTE NACELLE MEAN CAMBER RADII AND THICKNESS AND                  E 112
C     COMPUTE NACELLE EQUIVALENT INCOMPRESSIBLE BODY                       E 113
C                                                                          E 114
      DO 10 I=1,NN                                                         E 120
      R(I)=(ROUT(I)+RIN(I))/2.                                             E 130
      T(I)=ROUT(I)-RIN(I)                                                  E 140
10    CONTINUE                                                             E 150
      DO 20 I=1,NN                                                         E 160
      R(I)=R(I)*SQRT(1.-M**2)                                              E 170
      T(I)=T(I)*SQRT(1.-M**2)                                              E 180
20    CONTINUE                                                             E 190
      DMAX=DMAX*SQRT(1.-M**2)                                              E 200
      DMIN=DMIN*SQRT(1.-M**2)                                              E 210
      RAVG=(DMIN+DMAX)/4.                                                  E 220
C                                                                          E 230
C     COMPUTE NACELLE CAMBER AND THICKNESS SLOPES AND CENTER BODY SLOPES   E 240
C                                                                          E 250
      CALL SLOPE (XI,XIO,ALPHO,ALPHOO)                                     E 260
      DO 30 J=1,N                                                          E 270
      XBL(J)=(C/2.)-(XIO(J)*RAVG)                                          E 280
30    CONTINUE                                                             E 290
      CALL THICK (ALPT,ALPTH)                                              E 300
      CALL BODY (DRDX,DRDXH)                                               E 310
      LAM=(DMAX+DMIN)/(2.*C)                                               E 320
      IF (IT.LE.ITMAX) GO TO 40                                            E 330
      WRITE (6,50) LAM,(ALPHO(I),ALPHOO(I,1),XI(I),XIO(I),ALPT(I),ALPTH(   E 340
     1I),DRDX(I),DRDXH(I),I,I=1,N)                                         E 350
40    CONTINUE                                                             E 360
      RETURN                                                               E 370
C                                                                          E 380
C                                                                          E 390
50    FORMAT (1H1,25X,42HGEOMETRY ON EQUIVALENT INCOMPRESSIBLE BODY//7X,   E 400
     14HLAM=,F9.4//5X,5HALPHO,4X,6HALPHOO,8X,2HXI,7X,3HXIO,6X,4HALPT,5X,   E 410
     25HALPTH,6X,4HDRDX,5X,5HDRDXH,4X,6HI OR J//(8F10.5,I10))              E 420
      END                                                                  E 430-
```

```
      SUBROUTINE SLOPE (XI,XIO,ALPHO,ALPHOO)                              F  10
C                                                                         F  20
C     THIS SUBROUTINE COMPUTES XI(I), XIO(J), ALPHO(I), ALPHOO(J)         F  30
C                                                                         F  40
C     THE XI AND XIO ARE FOR NACELLE AND CENTER BODY .                    F  50
C     THE SLOPES ARE NACELLE CAMBER SLOPES.                               F  60
C                                                                         F  70
      COMMON /CORD/ X(170),R(170),C,RAVG,NN,N,DMIN                        F  80
      DIMENSION ALPHO(170), ALPHOO(170,1), XI(170), XIO(170)              F  90
      IP=-1                                                               F 100
      XXX=-.25*(C/N)                                                      F 110
      XX=(C/N)*.75                                                        F 120
      DO 30 I=1,N                                                         F 130
      IF (I.EQ.1) GO TO 10                                                F 140
      CALL MTLUP (XXX,RXI1,2,NN,170,1,IP,X,R)                             F 150
      CALL MTLUP (XX,RXI,2,NN,170,1,IP,X,R)                               F 160
      TEMP4=(RXI1-RXI)/(C/N)                                              F 170
      GO TO 20                                                            F 180
10    PIP=0.                                                              F 190
      POP=(C/N)*.5                                                        F 200
      CALL MTLUP (PIP,RXI1,2,NN,170,1,IP,X,R)                             F 210
      CALL MTLUP (POP,RXI,2,NN,170,1,IP,X,R)                              F 220
      TEMP4=(RXI1-RXI)/((C/N)*.5)                                         F 230
20    CONTINUE                                                            F 240
      ALPHO(I)=ATAN(TEMP4)                                                F 250
      XXX=XXX+(C/N)                                                       F 260
      XX=XX+(C/N)                                                         F 270
30    CONTINUE                                                            F 280
      DO 50 I=1,N                                                         F 290
      IF (I.EQ.N) GO TO 40                                                F 300
      ALPHOO(I,1)=(ALPHO(I)+ALPHO(I+1))*.5                                F 310
      GO TO 50                                                            F 320
40    ALPHOO(I,1)=ALPHO(I)+(.5*(ALPHO(I)-ALPHO(I-1)))                     F 330
50    CONTINUE                                                            F 340
      TEMP5=.5*C/RAVG                                                     F 350
      TEMP6=(C/N)/RAVG                                                    F 360
      DO 70 I=1,N                                                         F 370
      IF (I.EQ.1) GO TO 60                                                F 380
      XI(I)=XI(I-1)-TEMP6                                                 F 390
      XIO(I)=XIO(I-1)-TEMP6                                               F 400
      GO TO 70                                                            F 410
60    XI(I)=TEMP5-(.25*TEMP6)                                             F 420
      XIO(I)=TEMP5-(.75*TEMP6)                                            F 430
70    CONTINUE                                                            F 440
      RETURN                                                              F 450
      END                                                                 F 460-
```

```
      SUBROUTINE THICK (ALPT,ALPTH)                                    G  10
C                                                                      G  20
C     THIS SUBROUTINE COMPUTES NACELLE THICKNESS SLOPES.               G  30
C                                                                      G  40
      COMMON /CORD/ X(170),R(170),C,RAVG,NN,N,DMIN,T(170)              G  50
      DIMENSION ALPT(170), ALPTH(170)                                  G  60
      IP=-1                                                            G  70
      XXX=-.25*(C/N)                                                   G  80
      XX=(C/N)*.75                                                     G  90
      DO 30 I=1,N                                                      G 100
      IF (I.EQ.1) GO TO 10                                             G 110
      CALL MTLUP (XXX,TXI1,2,NN,170,1,IP,X,T)                          G 120
      CALL MTLUP (XX,TXI,2,NN,170,1,IP,X,T)                            G 130
      TEMP4=((TXI1-TXI)/2.)/(C/N)                                      G 140
      GO TO 20                                                         G 150
10    X14=(C/N)*.5                                                     G 160
      CALL MTLUP (X14,TXI,2,NN,170,1,IP,X,T)                           G 170
      TEMP4=((0.-TXI)/2.)/((C/N)*.5)                                   G 180
20    CONTINUE                                                         G 190
      ALPT(I)=ATAN(TEMP4)                                              G 200
      XXX=XXX+(C/N)                                                    G 210
      XX=XX+(C/N)                                                      G 220
30    CONTINUE                                                         G 230
      DO 50 I=1,N                                                      G 240
      IF (I.EQ.N) GO TO 40                                             G 250
      ALPTH(I)=(ALPT(I)+ALPT(I+1))*.5                                  G 260
      GO TO 50                                                         G 270
40    ALPTH(I)=ALPT(I)+(.5*(ALPT(I)-ALPT(I-1)))                        G 280
50    CONTINUE                                                         G 290
      RETURN                                                           G 300
      END                                                              G 310-
```

```
      SUBROUTINE BODY (DRDX,DRDXH)                                          H  10
C                                                                           H  20
C     THIS SUBROUTINE COMPUTES CENTER BODY SLOPES.                          H  30
C                                                                           H  40
      COMMON /CORD/ X(170),R(170),C,RAVG,NN,N,DMIN,T(170),RB(170),XBN       H  50
      DIMENSION DRDX(170), DRDXH(170)                                       H  60
      IP=-1                                                                 H  70
      XXX=-.25*(C/N)                                                        H  80
      XX=(C/N)*.75                                                          H  90
      DO 30 I=1,N                                                           H 100
      IF (I.EQ.1) GO TO 10                                                  H 110
      CALL MTLUP (XXX,RBXI1,2,NN,170,1,IP,X,RB)                             H 120
      IF (XXX.LE.XBN) RBXI1=0.                                              H 130
      CALL MTLUP (XX,RBXI,2,NN,170,1,IP,X,RB)                               H 140
      IF (XX.LE.XBN) RBXI=0.                                                H 150
      DRDX(I)=(RBXI-RBXI1)/(C/N)                                            H 160
      GO TO 20                                                              H 170
10    X14=(C/N)*.5                                                          H 180
      CALL MTLUP (X14,RBXI,2,NN,170,1,IP,X,RB)                              H 190
      IF (X14.LE.XBN) RBXI=0.                                               H 200
      DRDX(I)=RBXI/((C/N)*.5)                                               H 210
20    XXX=XXX+(C/N)                                                         H 220
      XX=XX+(C/N)                                                           H 230
30    CONTINUE                                                              H 240
      DO 50 I=1,N                                                           H 250
      IF (I.EQ.N) GO TO 40                                                  H 260
      DRDXH(I)=(DRDX(I)+DRDX(I+1))/2.                                       H 270
      GO TO 50                                                              H 280
40    DRDXH(I)=DRDX(I)+(.5*(DRDX(I)-DRDX(I-1)))                             H 290
50    CONTINUE                                                              H 300
      IF (DRDXH(1).NE.0.) GO TO 80                                          H 310
      DO 60 I=2,N                                                           H 320
      IF (DRDXH(I).NE.0.) GO TO 70                                          H 330
60    CONTINUE                                                              H 340
70    DRDXH(I)=0.                                                           H 350
80    CONTINUE                                                              H 360
      RETURN                                                                H 370
      END                                                                   H 380-
```

```
      OVERLAY(RING,2,0)                                                    I  10
      PROGRAM MATRIX                                                       I  20
C                                                                          I  30
C     THIS OVERLAY COMPUTES NACELLE CAMBER AND THICKNESS VORTEX            I  40
C     STRENGTHS - GAM(I) AND GQ(I)                                         I  41
C                                                                          I  50
      COMMON /CORD/ X(170),R(170),C,RAVG,NN,N,DMIN,T(170),RB(170),XBN      I  60
      COMMON /ONE/ ROUT(170),RIN(170),M,DMAX,XI(170),XIO(170),ALPHO(170)   I  70
     1,ALPHOO(170,1),XBL(170),ALPT(170),ALPTH(170),DRDX(170),DRDXH(170),   I  80
     2LAM,IT,ITMAX                                                         I  90
      COMMON /TWO/ IP,CAM,GAM(170),GQ(170),ALPOO(170,1)                    I 100
      REAL KSQ,LAM,M                                                       I 110
      DIMENSION WA(170), WY(170), GQB(170), ANS(170), ALPHQBB(170,1), RH   I 120
     100(170)                                                              I 130
C                                                                          I 131
C     COMPUTE SECOND ALPHOO WHICH IS PRINTED - DO 20                       I 132
C                                                                          I 133
      DO 20 J=1,N                                                          I 140
      WRBQJU=0.                                                            I 150
      RHOO(J)=1.                                                           I 160
      XX=(C/N)*.25                                                         I 170
      DO 10 I=1,N                                                          I 180
      CALL MTLUP (XX,RBXI,2,NN,170,1,IP,X,RB)                              I 190
      IF (XX.LE.XBN) RBXI=0.                                               I 200
      DELXI=XI(I)-XIO(J)                                                   I 210
      TEMP11=RHOO(J)/(2.*((DELXI**2+RHOO(J)**2)**1.5))                     I 220
      WRBQJU=WRBQJU+TEMP11*(RBXI/RAVG)*DRDX(I)*(XI(1)-XI(2))               I 230
      XX=XX+(C/N)                                                          I 240
10    CONTINUE                                                             I 250
      ALPOO(J,1)=ALPHOO(J,1)                                               I 260
      ALPHOO(J,1)=-2.*3.14159*(ALPHOO(J,1)+WRBQJU)                         I 270
20    CONTINUE                                                             I 280
C                                                                          I 290
C     SOLVE MATRIX EQUATION FOR THE GAM(I) MATRIX                          I 300
C                                                                          I 310
      IF (CAM.EQ.0.) GO TO 50                                              I 320
      REWIND 10                                                            I 330
      DO 40 J=1,N                                                          I 340
      DO 30 I=1,N                                                          I 350
      DELXI=XI(I)-XIO(J)                                                   I 360
      KSQ=4./(DELXI**2+4.)                                                 I 370
      EK=EKA(KSQ)                                                          I 380
      SK=SKK(KSQ)                                                          I 390
30    WA(I)=(((-1.)*DELXI*SQRT(KSQ))/4.)*((((2.-KSQ)/(1.-KSQ))*EKA(KSQ))  I 400
     1-(2.*SKK(KSQ)))*ALPHO(I)                                             I 410
      WRITE (10) (WA(IT),IT=1,N),ALPHOO(J,1)                               I 420
40    CONTINUE                                                             I 430
      CALL SSLESO (N,1,GAM)                                                I 440
      REWIND 10                                                            I 450
      GO TO 70                                                             I 460
50    DO 60 I=1,N                                                          I 470
      GAM(I)=0.                                                            I 480
60    CONTINUE                                                             I 490
70    CONTINUE                                                             I 500
C                                                                          I 510
C     COMPUTE ALPHQBB - DO 80                                              I 520
C                                                                          I 530
      DO 80 J=1,N                                                          I 540
      ALPHQBB(J,1)=0.                                                      I 550
      XX=C/N*.25                                                           I 560
```

```
      DO 80 I=1,N                                                        I 570
      DELXI=XI(I)-XIO(J)                                                 I 580
      KSQ=4./(DELXI**2+4.)                                               I 590
      EK=EKA(KSQ)                                                        I 600
      SK=SKK(KSQ)                                                        I 610
      CALL MTLUP (XX,TX,2,NN,170,1,IP,X,T)                               I 620
      ALPHQBB(J,1)=ALPHQBB(J,1)+((TX/C)*(((SQRT(1.-KSQ))/2.)*(-DELXI/ABS I 630
     1(DELXI))*(((KSQ+2.)*SK)-(2.*(KSQ+1.)*EK)))*(XI(1)-XI(2)))         I 640
      XX=XX+(C/N)                                                        I 650
80    CONTINUE                                                           I 660
C                                                                        I 670
C     SOLVE MATRIX EQUATION FOR GQB(I) MATRIX                            I 680
C                                                                        I 690
      REWIND 10                                                          I 700
      DO 100 J=1,N                                                       I 710
      DO 90 I=1,N                                                        I 720
      DELXI=XI(I)-XIO(J)                                                 I 730
      KSQ=4./(DELXI**2+4.)                                               I 740
      EK=EKA(KSQ)                                                        I 750
      SK=SKK(KSQ)                                                        I 760
90    WY(I)=(((KSQ-2.)*EK+2.*(1.-KSQ)*SK)/SQRT(KSQ))*((XI(1)-XI(2))/DELX I 770
     1I)*(1./LAM)                                                        I 780
      WRITE (10) (WY(IT),IT=1,N),ALPHQBB(J,1)                            I 790
100   CONTINUE                                                           I 800
      CALL SSLESO (N,1,GQB)                                              I 810
C                                                                        I 820
C     COMPUTE GQ                                                         I 830
C                                                                        I 840
      XX=C/N*.25                                                         I 850
      DO 110 I=1,N                                                       I 860
      CALL MTLUP (XX,TX,2,NN,170,1,IP,X,T)                               I 870
      GQ(I)=((TX/C)/LAM)+(((C/(2.*RAVG))**2)*GQB(I))                     I 880
      GQ(I)=-GQ(I)                                                       I 890
      XX=XX+(C/N)                                                        I 900
110   CONTINUE                                                           I 910
      IF (IT.LE.ITMAX) GO TO 130                                         I 920
      PRINT 140                                                          I 930
      DO 120 I=1,N                                                       I 940
      WRITE (6,150) I,GAM(I),GQ(I),GQB(I),ALPHQBB(I,1),ALPHOO(I,1)       I 950
120   CONTINUE                                                           I 960
130   CONTINUE                                                           I 970
      RETURN                                                             I 980
C                                                                        I 990
C                                                                        I1000
140   FORMAT (1H1,11X,1HI,9X,3HGAM,11X,2HGQ,10X,3HGQB,6X,7HALPHQBB,7X,6H I1010
     1ALPHOO)                                                            I1020
150   FORMAT (9X,I3,2X,E12.5,4(2X,E11.4))                                I1030
      END                                                                I1040-
```

```
      SUBROUTINE SSLESO (NT,NCFLG,ANS)                                  J  10
C                                                                       J  20
C     THIS SUBROUTINE SOLVES SIMULTANEOUS LINEAR EQUATIONS BY           J  30
C     SUCCESSIVE ORTHOGONALIZATION **                                   J  40
C                                                                       J  50
C     IF MATRIX ORDER IS = N, AND NUMBER OF RIGHT-HAND SIDES = M, THEN  J  60
C     THE AMOUNT OF CORE THAT MUST BE AVAILABLE FOR EACH ARRAY IS --    J  70
C     R = N+M, RV = N+M-1, CV = N-1, V =J(J+1) WHERE J IS THE INTEGER   J  80
C     PORTION OF (N+M)/2                                                J  90
C                                                                       J 100
C     DIMENSIONS BASED ON THE VALUES N+M .LE. 283, MAX N IS 281         J 110
C                                                                       J 120
      DIMENSION RV(170), R(170), CV(170), V(8100), ANS(NT)              J 130
      REWIND 10                                                         J 140
      N1=NT+NCFLG                                                       J 150
      J=N1-1                                                            J 160
C                                                                       J 170
C     READ (10)                                                         J 180
C                                                                       J 190
      READ (10) (R(I),I=1,N1)                                           J 200
      DO 10 I=1,J                                                       J 210
10    V(I)=-R(I+1)/R(1)                                                 J 220
      IN=1                                                              J 230
C                                                                       J 240
C     READ (10)                                                         J 250
C                                                                       J 260
20    READ (10) (R(I),I=1,N1)                                           J 270
      I2=0                                                              J 280
      DO 40 I=1,J                                                       J 290
      RV(I)=0.                                                          J 300
      DO 30 II=1,IN                                                     J 310
      I2=I2+1                                                           J 320
30    RV(I)=RV(I)+R(II)*V(I2)                                           J 330
      N2=I+IN                                                           J 340
40    RV(I)=RV(I)+R(N2)                                                 J 350
      I2=IN+1                                                           J 360
      NN=J*IN+1                                                         J 370
      KK=J*I2                                                           J 380
      J=J-1                                                             J 390
      DO 60 I=1,J                                                       J 400
      DO 50 II=1,IN                                                     J 410
      NN=NN-1                                                           J 420
      KK=KK-1                                                           J 430
50    V(KK)=V(NN)                                                       J 440
60    KK=KK-1                                                           J 450
      DO 70 I=1,IN                                                      J 460
70    R(I)=V(I)                                                         J 470
      K=0                                                               J 480
      DO 90 I=1,J                                                       J 490
      C=-RV(I+1)/RV(1)                                                  J 500
      DO 80 II=1,IN                                                     J 510
      CV(II)=C*R(II)                                                    J 520
      NN=K+II                                                           J 530
      I2=I2+1                                                           J 540
80    V(NN)=CV(II)+V(I2)                                                J 550
      K=NN+1                                                            J 560
      I2=I2+1                                                           J 570
90    V(K)=C                                                            J 580
      IF (J.EQ.NCFLG) GO TO 100                                         J 590
      IN=IN+1                                                           J 600
```

```
      GO TO 20                                                    J 610
100   K=1-NT                                                      J 620
      KK=0                                                        J 630
      DO 110 J=1,NCFLG                                            J 640
      K=K+NT                                                      J 650
      KK=KK+NT                                                    J 660
110   CONTINUE                                                    J 670
      DO 120 J=1,NT                                               J 680
120   ANS(J)=-V(J)                                                J 690
      RETURN                                                      J 700
      END                                                         J 710-
```

```
      OVERLAY(RING,3,0)                                                    K  10
      PROGRAM PRESURE                                                      K  20
C                                                                          K  30
C     THIS OVERLAY COMPUTES NACELLE AND CENTER BODY SURFACE PRESSURES       K  40
C                                                                          K  50
      COMMON /CORD/ X(170),R(170),C,RAVG,NN,N,DMIN,T(170),RB(170),XBN      K  60
      COMMON /ONE/ ROUT(170),RIN(170),M,DMAX,XI(170),XIO(170),ALPHO(170)   K  70
     1,ALPHOO(170,1),XBL(170),ALPT(170),ALPTH(170),DRDX(170),DRDXH(170),   K  80
     2LAM,IT,ITMAX                                                         K  90
      COMMON /TWO/ IP,CAM,GAM(170),GQ(170),ALPOO(170,1)                    K 100
      COMMON /THREE/ VOUT(170),VIN(170)                                    K 110
      COMMON /FIVE/ CPO(170),MCR                                           K 120
      DIMENSION RHOO(170), CPI(170)                                        K 130
      REAL KSQ,LAM,M                                                       K 140
      IF (IT.LE.ITMAX) GO TO 10                                            K 150
      PRINT 190                                                            K 160
      PRINT 130                                                            K 170
      PRINT 140                                                            K 180
      PRINT 170                                                            K 190
10    CONTINUE                                                             K 200
      XXX=(C/N)*.75                                                        K 210
      DO 120 J=1,N                                                         K 220
      XOJ=XIO(J)*RAVG/C                                                    K 230
      RHOO(J)=1.                                                           K 240
      XX=(C/N)*.25                                                         K 250
C                                                                          K 260
C     COMPUTE VELOCITIES ON NACELLE SURFACE - DO 20                        K 270
C                                                                          K 280
      WX1JUO=0.                                                            K 290
      WR1JUO=0.                                                            K 300
      WX1QJU=0.                                                            K 310
      WR1QJU=0.                                                            K 320
      WXQJU=0.                                                             K 330
      WRQJU=0.                                                             K 340
      WXBQJU=0.                                                            K 350
      WRBQJU=0.                                                            K 360
      DO 20 I=1,N                                                          K 370
      DELXI=XI(I)-XIO(J)                                                   K 380
      KSQ=4./(DELXI**2+4.)                                                 K 390
      EK=EKA(KSQ)                                                          K 400
      SK=SKK(KSQ)                                                          K 410
      CALL MTLUP (XX,RBXI,2,NN,170,1,IP,X,RB)                              K 420
      IF (XX.LE.XBN) RBXI=0.                                               K 430
      TEMP1=(1./(2.*SQRT(DELXI**2+4.)))*((((2.-KSQ)/(1.-KSQ))*EKA(KSQ))-  K 440
     1(2.*SKK(KSQ))-((KSQ/(1.-KSQ))*EKA(KSQ)))                            K 450
      TEMP2=(((-1.)*DELXI*SQRT(KSQ))/4.)*((((2.-KSQ)/(1.-KSQ))*EKA(KSQ))  K 460
     1-(2.*SKK(KSQ)))                                                     K 470
      TEMP7=(4.*EK)/((SQRT(4.+DELXI**2))*DELXI)                            K 480
      TEMP8=2.*TEMP1                                                       K 490
      TEMP11=RHOO(J)/(2.*((DELXI**2+RHOO(J)**2)**1.5))                     K 500
      TEMP12=(DELXI)/(2.*((DELXI**2+RHOO(J)**2)**1.5))                     K 510
      WX1JUO=WX1JUO+TEMP1*ALPHO(I)*GAM(I)*(1./(2.*3.14159))                K 520
      WR1JUO=WR1JUO+TEMP2*ALPHO(I)*GAM(I)*(1./(2.*3.14159))                K 530
      WX1QJU=WX1QJU+(TEMP1*GQ(I)*(XI(1)-XI(2))/(2.*3.14159))               K 540
      WR1QJU=WR1QJU+(TEMP2*GQ(I)*(XI(1)-XI(2))/(2.*3.14159))               K 550
      WXQJU=WXQJU-(TEMP7*ALPT(I)*(XI(1)-XI(2))/(2.*3.14159))               K 560
      WRQJU=WRQJU+(TEMP8*ALPT(I)*(XI(1)-XI(2))/(2.*3.14159))               K 570
      WXBQJU=WXBQJU+TEMP12*(RBXI/RAVG)*DRDX(I)*(XI(1)-XI(2))               K 580
      WRBQJU=WRBQJU+TEMP11*(RBXI/RAVG)*DRDX(I)*(XI(1)-XI(2))               K 590
      XX=XX+(C/N)                                                          K 600
```

```
20    CONTINUE                                                                K 610
      IF (J.EQ.N) GO TO 30                                                    K 620
      TEMP3=(ALPHO(J)*GAM(J)+ALPHO(J+1)*GAM(J+1))*.25*(1./(XI(1)-XI(2)))      K 630
      TEMP31=.25*(GQ(J)+GQ(J+1))                                              K 640
      GO TO 40                                                                K 650
30    TEMP3=ALPHO(J)*GAM(J)*.50*(1./(XI(1)-XI(2)))*.3333333333                K 660
      TEMP31=.50*(GQ(J)*.3333333333)                                          K 670
40    CONTINUE                                                                K 680
      WX1JUOO=WX1JUO+TEMP3                                                    K 690
      WX1JUOI=WX1JUO-TEMP3                                                    K 700
      WX1QJUI=WX1QJU-TEMP31                                                   K 710
      WX1QJUO=WX1QJU+TEMP31                                                   K 720
      WRQJUI=WRQJU+ALPTH(J)                                                   K 730
      WRQJUO=WRQJU-ALPTH(J)                                                   K 740
C                                                                             K 750
C     CALCULATE TOTAL VELOCITY SQUARED AND APPLY COMPRESSIBILITY FACTOR       K 760
C                                                                             K 770
      VTO=(1.+(WX1JUOO+WX1QJUO+WXQJU+WXBQJU)/(1.-M**2))**2                    K 780
      VTI=(1.+(WX1JUOI+WX1QJUI+WXQJU+WXBQJU)/(1.-M**2))**2                    K 790
C                                                                             K 800
C     APPLY RIEGELS FACTOR                                                    K 810
C                                                                             K 820
      VTO=VTO/(1.+(TAN((ALPOO(J,1)+ALPTH(J))/SQRT(1.-M**2)))**2)              K 830
      VTI=VTI/(1.+(TAN((ALPOO(J,1)-ALPTH(J))/SQRT(1.-M**2)))**2)              K 840
      VOUT(J)=SQRT(VTO)                                                       K 850
      VIN(J)=SQRT(VTI)                                                        K 860
C                                                                             K 870
C     COMPUTE CPO AND CPI                                                     K 880
C                                                                             K 890
      IF (M.EQ.0.) GO TO 50                                                   K 900
      CPO(J)=(2./(1.4*(M**2)))*((1.+(.2*(M**2)*(1.-VTO)))**(1.4/.4)-1.)       K 910
      CPI(J)=(2./(1.4*(M**2)))*((1.+(.2*(M**2)*(1.-VTI)))**(1.4/.4)-1.)       K 920
      GO TO 60                                                                K 930
50    CPO(J)=(1.-VTO)                                                         K 940
      CPI(J)=(1.-VTI)                                                         K 950
60    CONTINUE                                                                K 960
C                                                                             K 970
C     COMPUTE CENTER BODY SURFACE VELOCITIES - DO 70                          K 980
C                                                                             K 990
      WX1J=0.                                                                K1000
      WR1J=0.                                                                 K1010
      WX1QJ=0.                                                                K1020
      WR1QJ=0.                                                                K1030
      WXQJ=0.                                                                 K1040
      WRQJ=0.                                                                 K1050
      WXBQJ=0.                                                                K1060
      WRBQJ=0.                                                                K1070
      XX=(C/N)*.25                                                            K1080
      CALL MTLUP (XXX,RBXJ,2,NN,170,1,IP,X,RB)                                K1090
      IF (XXX.LE.XBN) RBXJ=0.                                                 K1100
      RHOO(J)=RBXJ/RAVG                                                       K1110
      IF (RHOO(J).LE.0.) GO TO 80                                             K1120
      DO 70 I=1,N                                                             K1130
      DELXI=XI(I)-XIO(J)                                                      K1140
      CALL MTLUP (XX,RBXI,2,NN,170,1,IP,X,RB)                                 K1150
      IF (XX.LE.XBN) RBXI=0.                                                  K1160
      KSQ=(4.*RHOO(J))/((DELXI**2)+((1.+RHOO(J))**2))                         K1170
      EK=EKA(KSQ)                                                             K1180
      SK=SKK(KSQ)                                                             K1190
      TEMP1=(1./(2.*SQRT((DELXI**2)+((1.+RHOO(J))**2))))*((((2.-KSQ)/(1.      K1200
```

```
     1-KSQ))*EK)-(2.*SK)-((1./RHOO(J))*(KSQ/(1.-KSQ))*EK))                    K1210
      TEMP2=(((-1.)*DELXI)/(2.*RHOO(J)*SQRT((DELXI**2)+((1.+RHOO(J))**2)      K1220
     1)))*((((2.-KSQ)/(1.-KSQ))*EK)-(2.*SK))                                  K1230
      TEMP7=(4.*DELXI*EK)/((SQRT((DELXI**2)+((1.+RHOO(J))**2)))*((DELXI*      K1240
     1*2)+((1.-RHOO(J))**2)))                                                 K1250
      TEMP8=(2./(RHOO(J)*SQRT((DELXI**2)+((1.+RHOO(J))**2))))*(((1.-((2.      K1260
     1*RHOO(J)*(RHOO(J)-1.))/((DELXI**2)+((1.-RHOO(J))**2))))*EK)-SK)         K1270
      TEMP11=RHOO(J)/(2.*((DELXI**2+RHOO(J)**2)**1.5))                        K1280
      TEMP12=(DELXI)/(2.*((DELXI**2+RHOO(J)**2)**1.5))                        K1290
      WX1J=WX1J+TEMP1*ALPHO(I)*GAM(I)*(1./(2.*3.14159))                       K1300
      WR1J=WR1J+TEMP2*ALPHO(I)*GAM(I)*(1./(2.*3.14159))                       K1310
      WX1QJ=WX1QJ+(TEMP1*GQ(I)*(XI(1)-XI(2))/(2.*3.14159))                    K1320
      WR1QJ=WR1QJ+(TEMP2*GQ(I)*(XI(1)-XI(2))/(2.*3.14159))                    K1330
      WXQJ=WXQJ-(TEMP7*ALPT(I)*(XI(1)-XI(2))/(2.*3.14159))                    K1340
      WRQJ=WRQJ+(TEMP8*ALPT(I)*(XI(1)-XI(2))/(2.*3.14159))                    K1350
      WXBQJ=WXBQJ+TEMP12*(RBXI/RAVG)*DRDX(I)*(XI(1)-XI(2))                    K1360
      WRBQJ=WRBQJ+TEMP11*(RBXI/RAVG)*DRDX(I)*(XI(1)-XI(2))                    K1370
      XX=XX+(C/N)                                                             K1380
70    CONTINUE                                                                K1390
80    VBA=(1.+(WX1J+WX1QJ+WXQJ+WXBQJ)/(1.-M**2))**2+((WR1J+WR1QJ+WRQJ+WR      K1400
     1BQJ)/SQRT(1.-M**2))**2                                                  K1410
      VBI=(1.+WXBQJ/(1.-M**2))**2+(WRBQJ/SQRT(1.-M**2))**2                    K1420
C                                                                             K1430
C     COMPUTE CENTER BODY PRESSURES                                           K1440
C                                                                             K1450
      IF (M.EQ.0.) GO TO 90                                                   K1460
      CPBA=(2./(1.4*(M**2)))*((1.+(.2*(M**2)*(1.-VBA)))**(1.4/.4)-1.)         K1470
      CPBI=(2./(1.4*(M**2)))*((1.+(.2*(M**2)*(1.-VBI)))**(1.4/.4)-1.)         K1480
      GO TO 100                                                               K1490
90    CPBA=(1.-VBA)                                                           K1500
      CPBI=(1.-VBI)                                                           K1510
100   CONTINUE                                                                K1520
      IF (IT.LE.ITMAX) GO TO 110                                              K1530
      WRITE (6,150) J,XOJ,CPO(J),CPI(J),VTO,VTI,WX1JUOO,WX1JUOI,WX1JUO,W      K1540
     1R1JUO                                                                   K1550
      WRITE (6,160) WX1QJUO,WX1QJUI,WX1QJU,WR1QJU,WXQJU,WRQJUO,WRQJUI,WR      K1560
     1QJU                                                                     K1570
      WRITE (6,180) CPBA,CPBI,WXBQJU,WRBQJU,RHOO(J)                           K1580
110   CONTINUE                                                                K1590
      XXX=XXX+(C/N)                                                           K1600
120   CONTINUE                                                                K1610
      RETURN                                                                  K1620
C                                                                             K1630
C                                                                             K1640
130   FORMAT (1X,1HJ,7X,5HXOJ/C,8X,3HCPO,9X,3HCPI,8X,3HVTO,10X,3HVTI,6X,      K1650
     17HWX1JUOO,6X,7HWX1JUOI,7X,6HWX1JUO,7X,6HWR1JUO)                         K1660
140   FORMAT (19X,7HWX1QJUO,6X,7HWX1QJUI,7X,6HWX1QJU,7X,6HWR1QJU,8X,5HWX      K1670
     1QJU,7X,6HWRQJUO,7X,6HWRQJUI,8X,5HWRQJU)                                 K1680
150   FORMAT (1X,I3,3(2X,F10.5),6(2X,E11.4))                                  K1690
160   FORMAT (13X,8(2X,E11.4))                                                K1700
170   FORMAT (19X,4HCPBA,9X,4HCPBI,7X,6HWXBQJU,7X,6HWRBQJU,4X,9HRHOO-80D      K1710
     1Y/)                                                                     K1720
180   FORMAT (10X,2(2X,F11.5),3X,2(2X,E11.4),F10.4/)                          K1730
190   FORMAT (1H1,20HNACELLE SURFACE SOLN,10X,48HALL PERTURB VEL ARE ON       K1740
     1EQUIV INCOMP NACELLE SURF,10X,26HCENTER BODY RHOO AND PRESS//)          K1750
      END                                                                     K1760-
```

```
      OVERLAY(RING,4,0)                                                    L  10
      PROGRAM BNDLYR                                                       L  20
C                                                                          L  30
C     THIS OVERLAY EXTRAPOLATES THE VELOCITIES IN THE TRAILING EDGE        L  40
C     REGION OF THE NACELLE AND COMPUTES THE BOUNDARY LAYER ON THE         L  41
C     NACELLE.  THEN THE NACELLE COORDINATES ARE MODIFIED FOR THE          L  42
C     DISPLACEMENT THICKNESS.                                              L  43
C                                                                          L  50
      COMMON /CORD/ X(170),R(170),C,RAVG,NN,N,DMIN,T(170),RB(170),XBN      L  60
      COMMON /ONE/ ROUT(170),RIN(170),M,DMAX,XI(170),XIO(170),ALPHO(170)   L  70
     1,ALPHOO(170,1),XBL(170),ALPT(170),ALPTH(170),DRDX(170),DRDXH(170),   L  80
     2LAM,IT,ITMAX                                                         L  90
      COMMON /TWO/ IP,CAM,GAM(170),GQ(170),ALPOO(170,1)                    L 100
      COMMON /THREE/ VOUT(170),VIN(170)                                    L 110
      COMMON /FOUR/ XOVC(170),RO(170),RI(170),DELO(170),DELI(170),WLS(40   L 120
     1),DOLDO(170),DOLDI(170),UNRREF,SAM,XFOR,XAFT                         L 130
      DIMENSION UOURO(170), UOURI(170), THE(170), H(170), DELST(170), XL   L 140
     1S(40), YLS(40,1), RESID(40,1), SUM(1), ALS(3,3), BLS(3,1), CLSO(40   L 150
     2,2), CLSI(40,3), DDEL(170)                                           L 160
      REAL JOHN,LAM,M                                                      L 170
C                                                                          L 180
C     EXTRAPOLATE OUTER SURFACE VELOCITIES FROM XFOR PERCENT CHORD TO      L 190
C     TRAILING EDGE BY LINEAR LEAST SQUARE FIT OF VELOCITIES               L 200
C     BETWEEN XFOR AND XAFT PERCENT CHORD                                  L 210
C                                                                          L 220
      L=0                                                                  L 230
      DO 10 J=1,N                                                          L 240
      JOHN=XBL(J)/C                                                        L 250
      IF (JOHN.LT.XFOR) GO TO 10                                           L 260
      IF (JOHN.GT.XAFT) GO TO 10                                           L 270
      L=L+1                                                                L 280
      XLS(L)=XBL(J)                                                        L 290
      YLS(L,1)=VOUT(J)                                                     L 300
10    CONTINUE                                                             L 310
      CALL LSQPOL (XLS,YLS,WLS,RESID,L,SUM,1,ALS,BLS,2,CLSO,40,3)          L 320
      DO 20 J=1,N                                                          L 330
      JOHN=XBL(J)/C                                                        L 340
      IF (JOHN.LT.XFOR) GO TO 20                                           L 350
      VOUT(J)=BLS(1,1)+BLS(2,1)*XBL(J)                                     L 360
20    CONTINUE                                                             L 370
C                                                                          L 380
C     EXTRAPOLATE INNER SURFACE VELOCITIES FROM XFOR PERCENT CHORD TO      L 390
C     TRAILING EDGE BY PARABOLIC LEAST SQUARE FIT OF VELOCITIES            L 400
C     BETWEEN XFOR AND XAFT PERCENT CHORD AND LAST OUTER SURFACE           L 410
C     VELOCITY FROM ABOVE                                                  L 420
C                                                                          L 430
      L=0                                                                  L 440
      DO 30 J=1,N                                                          L 450
      JOHN=XBL(J)/C                                                        L 460
      IF (JOHN.LT.XFOR) GO TO 30                                           L 470
      IF (JOHN.GT.XAFT) GO TO 30                                           L 480
      L=L+1                                                                L 490
      XLS(L)=XBL(J)                                                        L 500
      YLS(L,1)=VIN(J)                                                      L 510
30    CONTINUE                                                             L 520
      L=L+1                                                                L 530
      XLS(L)=XBL(N)                                                        L 540
      YLS(L,1)=VOUT(N)                                                     L 550
      CALL LSQPOL (XLS,YLS,WLS,RESID,L,SUM,1,ALS,BLS,3,CLSI,40,3)          L 560
      DO 40 J=1,N                                                          L 570
```

```
      JOHN=XBL(J)/C                                                          L 580
      IF (JOHN.LT.XFOR) GO TO 40                                             L 590
      VIN(J)=BLS(1,1)+(BLS(2,1)*XBL(J))+(BLS(3,1)*(XBL(J)**2))               L 600
40    CONTINUE                                                               L 610
C                                                                            L 620
C     INTERPOLATE FOR VEL AT INPUT X                                         L 630
C                                                                            L 640
      LUG=NN-1                                                               L 650
      DO 50 I=2,LUG                                                          L 660
      BLAIR=X(I)                                                             L 670
      CALL MTLUP (BLAIR,GLOSS,2,N,170,1,IP,XBL,VOUT)                         L 680
      CALL MTLUP (BLAIR,TIM,2,N,170,1,IP,XBL,VIN)                            L 690
      UOURO(I)=GLOSS                                                         L 700
      UOURI(I)=TIM                                                           L 710
50    CONTINUE                                                               L 720
      UOURO(1)=VOUT(1)                                                       L 730
      UOURI(1)=VIN(1)                                                        L 740
      UOURO(NN)=VOUT(N)                                                      L 750
      UOURI(NN)=VIN(N)                                                       L 760
      DO 60 I=1,NN                                                           L 770
      JOHN=X(I)/C                                                            L 780
      IF (JOHN.LT..02) UOURI(I)=UOURO(1)                                     L 790
60    CONTINUE                                                               L 800
C                                                                            L 810
C     COMPUTE OUTER BOUNDARY LAYER AND MODIFY OUTER WING COORDINATES FOR      L 820
C     DISPLACEMENT THICKNESS                                                 L 830
C                                                                            L 840
      CALL BLYR (X,RO,NN,UOURO,UNRREF,THE,H,DELST)                           L 850
      DO 100 I=1,NN                                                          L 860
      DELO(I)=((DELO(I)*SAM)+(IT*DELST(I)))/(SAM+IT)                         L 870
      IF (IT.EQ.1) 70,80                                                     L 880
70    ROUT(I)=RO(I)+(DELO(I)/2.)                                             L 890
      GO TO 90                                                               L 900
80    ROUT(I)=RO(I)+DELO(I)                                                  L 910
90    DDEL(I)=DELST(I)-DOLDO(I)                                              L 920
      DOLDO(I)=DELST(I)                                                      L 930
100   CONTINUE                                                               L 940
C                                                                            L 950
C     FIND NEW DMAX                                                          L 960
C                                                                            L 970
      DO 140 I=1,NN                                                          L 980
      IF (I.EQ.1) 110,120                                                    L 990
110   ELIZ=ROUT(I)                                                           L1000
      GO TO 140                                                              L1010
120   IF (ROUT(I).GT.ELIZ) 130,140                                           L1020
130   ELIZ=ROUT(I)                                                           L1030
140   CONTINUE                                                               L1040
      DMAX=2.*ELIZ                                                           L1050
      WRITE (6,230) IT,DMAX,(XOVC(I),X(I),UOURO(I),DELST(I),DDEL(I),DELO     L1060
     1(I),ROUT(I),THE(I),H(I),I=1,NN)                                        L1070
C                                                                            L1080
C     COMPUTE INNER BOUNDARY LAYER AND MODIFY INNER WING COORDINATES FOR      L1090
C     DISPLACEMENT THICKNESS                                                 L1100
C                                                                            L1110
      CALL BLYR (X,RI,NN,UOURI,UNRREF,THE,H,DELST)                           L1120
      DO 180 I=1,NN                                                          L1130
      DELI(I)=((DELI(I)*SAM)+(IT*DELST(I)))/(SAM+IT)                         L1140
      IF (IT.EQ.1) 150,160                                                   L1150
150   RIN(I)=RI(I)-(DELI(I)/2.)                                              L1160
      GO TO 170                                                              L1170
```

```
160   RIN(I)=RI(I)-DELI(I)                                                  L1180
170   DDEL(I)=DELST(I)-DOLDI(I)                                             L1190
      DOLDI(I)=DELST(I)                                                     L1200
180   CONTINUE                                                              L1210
C                                                                           L1220
C     FIND NEW DMIN                                                         L1230
C                                                                           L1240
      DO 220 I=1,NN                                                         L1250
      IF (I.EQ.1) 190,200                                                   L1260
190   ELIZ=RIN(I)                                                           L1270
      GO TO 220                                                             L1280
200   IF (RIN(I).LT.ELIZ) 210,220                                           L1290
210   ELIZ=RIN(I)                                                           L1300
220   CONTINUE                                                              L1310
      DMIN=2.*ELIZ                                                          L1320
      WRITE (6,240) IT,DMIN,(XOVC(I),X(I),UOURI(I),DELST(I),DDEL(I),DELI    L1330
     1(I),RIN(I),THE(I),H(I),I=1,NN)                                        L1340
      SAM=SAM+IT                                                            L1350
      IT=IT+1                                                               L1360
      RETURN                                                                L1370
C                                                                           L1380
C                                                                           L1390
230   FORMAT (1H1,4X,44HOUTER WING SHAPE MODIFIED BY BOUNDARY LAYER./4X,    L1400
     150HVELOCITIES ARE ONES USED TO COMPUTE BOUNDARY LAYER//4X,3HIT=,I3    L1410
     2,4X,5HDMAX=,F9.4//7X,3HX/C,9X,1HX,5X,5HUOURO,5X,5HDELST,6X,4HDDEL,    L1420
     36X,4HDELO,6X,4HROUT,7X,3HTHE,9X,1HH//(9F10.4))                        L1430
240   FORMAT (1H1,4X,44HINNER WING SHAPE MODIFIED BY BOUNDARY LAYER./4X,    L1440
     150HVELOCITIES ARE ONES USED TO COMPUTE BOUNDARY LAYER//4X,3HIT=,I3    L1450
     2,4X,5HDMIN=,F9.4//7X,3HX/C,9X,1HX,5X,5HUOURI,5X,5HDELST,6X,4HDDEL,    L1460
     36X,4HDELI,7X,3HRIN,7X,3HTHE,9X,1HH//(9F10.4))                         L1470
      END                                                                   L1480
```

```
      SUBROUTINE BLYR (S,R,NX,UOUR,UNRREF,THE,H,DELST)                          M  10
C                                                                               M  20
C     THIS SUBROUTINE COMPUTES THE INCOMPRESSIBLE TURBULENT AXISYMMETRIC        M  30
C     BOUNDARY LAYER USING A SLIGHT MODIFICATION OF TRUCKENBRODTS METHOD        M  40
C     PAGE 584 - 585 OF BOUNDARY LAYER THEORY BY SCHLICHTING,4TH EDITION        M  50
C                                                                               M  60
C     TRANSITION IS ASSUMED AT THE LEADING EDGE.                                M  70
C                                                                               M  80
      REAL IARG,IARGL,IUOUR,L                                                   M  90
      DIMENSION R(200), S(200), UOUR(200), IUOUR(200), UAOUR(200), REY(2        M 100
     100), CF(200), ARG(200), IARG(200), THE(200), RTHE(200), B(200), AR        M 110
     2GL(200), IARGL(200), L(200), TL(13), TH(13), H(200), DELST(200), S        M 120
     3QRTXI(200)                                                                M 130
      DO 10 I=1,NX                                                              M 140
      IF (UOUR(I).LT..00001) UOUR(I)=0.00001                                    M 150
10    CONTINUE                                                                  M 160
C                                                                               M 170
C     U AVE OVER U REF                                                          M 180
C                                                                               M 190
      CALL INT (S,UOUR,NX,IUOUR)                                                M 200
      UAOUR(1)=UOUR(1)                                                          M 210
      DO 20 I=2,NX                                                              M 220
20    UAOUR(I)=IUOUR(I)/S(I)                                                    M 230
C                                                                               M 240
C     CF                                                                        M 250
C                                                                               M 260
      REY(1)=0.                                                                 M 270
      CF(1)=1.                                                                  M 280
      DO 30 I=2,NX                                                              M 290
      REY(I)=UNRREF*UAOUR(I)*S(I)                                               M 300
      IF (REY(I).LE.1.) REY(I)=1.                                               M 310
      CF(I)=.0745/(REY(I)**.218)+.00072                                         M 320
30    CONTINUE                                                                  M 330
C                                                                               M 340
C     THETA                                                                     M 350
C                                                                               M 360
      DO 40 I=1,NX                                                              M 370
40    ARG(I)=UOUR(I)**(10./3.)*R(I)**(7./6.)                                    M 380
      CALL INT (S,ARG,NX,IARG)                                                  M 390
      THE(1)=0.                                                                 M 400
      DO 50 I=2,NX                                                              M 410
      IF (IARG(I).LE.0.) IARG(I)=0.                                             M 420
50    THE(I)=(UAOUR(I)*S(I))**(1./7.)*CF(I)/2.*IARG(I)**(6./7.)/(UOUR(I)        M 430
     1**3*R(I))                                                                 M 440
C                                                                               M 450
C     SHAPE FACTOR                                                              M 460
C                                                                               M 470
C     XI                                                                        M 480
C                                                                               M 490
      SQRTXI(1)=0.                                                              M 500
      DO 60 I=1,NX                                                              M 510
      SQRTXI(I)=(IARG(I))**2                                                    M 520
60    CONTINUE                                                                  M 530
C                                                                               M 540
C     L                                                                         M 550
C                                                                               M 560
      DO 70 I=1,NX                                                              M 570
      RTHE(I)=UNRREF*UOUR(I)*THE(I)                                             M 580
      IF (RTHE(I).LT.1.) RTHE(I)=1.                                             M 590
      B(I)=.07*ALOG10(RTHE(I))-.23                                              M 600
```

```
      IF (B(I).LT.0.) B(I)=0.                                          M 610
70    ARGL(I)=(B(I)-ALOG(UOUR(I)))*2.*SQRTXI(I)                        M 620
      CALL INT (SQRTXI,ARGL,NX,IARGL)                                  M 630
      L(1)=0.                                                          M 640
      DO 100 I=2,NX                                                    M 650
      IF (SQRTXI(I).EQ.0.) GO TO 80                                    M 660
      L(I)=ALOG(UOUR(I))*IARGL(I)/SQRTXI(I)**2                         M 670
      GO TO 90                                                         M 680
80    L(I)=0.                                                          M 690
90    IF (L(I).GT.1.) L(I)=1.                                          M 700
100   IF (L(I).LT.-.19) L(I)=-.19                                      M 710
C                                                                      M 720
C     H AND DELTA STAR                                                 M 730
C                                                                      M 740
      DATA TL/-.183,-.172,-.157,-.131,-.1,-.05,.0,.05,.1,.3,.5,.7,1./,TH M 750
     1/2.4,2.2,2.0,1.8,1.645,1.495,1.4,1.335,1.295,1.2,1.15,1.12,1.07/,N M 760
     2TL/13/                                                           M 770
      IP=-1                                                            M 780
      DO 110 I=1,NX                                                    M 790
      TEP1=L(I)                                                        M 800
      CALL MTLUP (TEP1,TEP2,2,NTL,13,1,IP,TL,TH)                       M 810
      H(I)=TEP2                                                        M 820
      DELST(I)=THE(I)*H(I)                                             M 830
110   CONTINUE                                                         M 840
      RETURN                                                           M 850
      END                                                              M 860-
```

```
      SUBROUTINE INT (X,Y,N,ANS)                                          N  10
C                                                                         N  20
C     THIS SUBROUTINE IS USED BY SUBROUTINE BLYR TO INTEGRATE Y DX FROM   N  30
C     X(1) TO X(I). RESULT EQUALS ANS(I). I RUNS FROM 1 TO N.             N  40
C                                                                         N  50
      DIMENSION X(100), Y(100), ANS(100)                                  N  60
      SUM=0.0                                                             N  70
      ANS(1)=0.                                                           N  80
      IF (N.EQ.2) GO TO 90                                                N  90
      I=1                                                                 N 100
      GO TO 40                                                            N 110
10    XA=X(I)-X(I-1)                                                      N 120
      XB=X(I+1)-X(I)                                                      N 130
      YA=Y(I)-Y(I-1)                                                      N 140
      YB=Y(I+1)-Y(I)                                                      N 150
      IF (XA.EQ.0..OR.XB.EQ.0.) GO TO 20                                  N 160
      IF (YA/XA.GE.2.*YB/XB.AND.YA/XA.GE.0.) GO TO 20                     N 170
      IF (YA/XA.LE.2.*YB/XB.AND.YA/XA.LE.0.) GO TO 20                     N 180
      SUMFW=XB*(Y(I)+YB/3.*(XA**2*YB+XB**2*YA)/(6.*XA*(XA+XB)))           N 190
      GO TO 30                                                            N 200
20    SUMFW=XB*((Y(I)+Y(I+1))/2.)                                         N 210
30    IF (I.EQ.N-1) GO TO 80                                              N 220
40    I=I+1                                                               N 230
      XA=X(I)-X(I-1)                                                      N 240
      XB=X(I+1)-X(I)                                                      N 250
      YA=Y(I)-Y(I-1)                                                      N 260
      YB=Y(I+1)-Y(I)                                                      N 270
      IF (XA.EQ.0..OR.XB.EQ.0.) GO TO 50                                  N 280
      IF (YB/XB.GE.2.*YA/XA.AND.YB/XB.GE.0.) GO TO 50                     N 290
      IF (YB/XB.LE.2.*YA/XA.AND.YB/XB.LE.0.) GO TO 50                     N 300
      SUMBK=XA*(Y(I)-YA/3.-(XB**2*YA+XA**2*YB)/(6.*XB*(XA+XB)))           N 310
      GO TO 60                                                            N 320
50    SUMBK=XA*((Y(I)+Y(I-1))/2.)                                         N 330
60    IF (I.EQ.2) GO TO 70                                                N 340
      SUM=SUM+(SUMFW+SUMBK)/2.                                            N 350
      ANS(I)=SUM                                                          N 360
      GO TO 10                                                            N 370
70    SUM=SUMBK                                                           N 380
      ANS(2)=SUM                                                          N 390
      GO TO 10                                                            N 400
80    SUM=SUM+SUMFW                                                       N 410
      ANS(N)=SUM                                                          N 420
      GO TO 100                                                           N 430
90    ANS(2)=((Y(1)+Y(2))/2.)*(X(2)-X(1))                                 N 440
100   CONTINUE                                                            N 450
      RETURN                                                              N 460
      END                                                                 N 470-
```

```
      SUBROUTINE LSQPOL (X,Y,W,RESID,N,SUM,L,A,B,M,C,NMAX,MMAX)                  0  10
C                                                                               0  20
C     THIS SUBROUTINE IS A LEAST SQUARE POLYNOMIAL FIT.  GIVEN A SET OF         0  30
C     VALUES OF AN INDEPENDENT VARIABLE X  WITH ASSOCIATED WEIGHTS W AND        0  40
C     A SET OF CORRESPONDING VALUES OF Y.  THE ROUTINE DETERMINES THE           0  50
C     COEFFICIENTS OF THE POLYNOMIAL OF DEGREE M-1 WHICH GIVES THE BEST         0  60
C     FIT IN THE LEAST SQUARES SENSE TO THE SET OF Y.                           0  70
C                                                                               0  80
      DIMENSION X(NMAX), Y(NMAX,L), RESID(NMAX,L), A(MMAX,MMAX), B(MMAX,        0  90
     1L), C(NMAX,M), SUM(L), W(NMAX)                                            0 100
10    DO 20 I=1,N                                                               0 110
20    C(I,1)=1.0                                                                0 120
      DO 30 J=2,M                                                               0 130
      DO 30 I=1,N                                                               0 140
30    C(I,J)=C(I,J-1)*X(I)                                                      0 150
      DO 40 I=1,M                                                               0 160
      DO 40 J=1,M                                                               0 170
      A(I,J)=0.0                                                                0 180
      DO 40 K=1,N                                                               0 190
40    A(I,J)=A(I,J)+C(K,I)*C(K,J)*W(K)                                          0 200
      DO 50 J=1,L                                                               0 210
      DO 50 I=1,M                                                               0 220
      B(I,J)=0.0                                                                0 230
      DO 50 K=1,N                                                               0 240
50    B(I,J)=B(I,J)+C(K,I)*Y(K,J)*W(K)                                          0 250
      CALL MATINV (A,M,B,L,DETERM,RESID,C,MMAX,ISCALE)                          0 260
      DO 70 J=1,L                                                               0 270
      SUM(J)=0.0                                                                0 280
      KK=M                                                                      0 290
      DO 60 K=1,M                                                               0 300
      C(K,1)=B(KK,J)                                                            0 310
60    KK=KK-1                                                                   0 320
      DO 70 I=1,N                                                               0 330
      RESID(I,J)=POLYE1(X(I),M,C)-Y(I,J)                                        0 340
70    SUM(J)=SUM(J)+RESID(I,J)**2*W(I)                                          0 350
      RETURN                                                                    0 360
      END                                                                       0 370-
```

```
      SUBROUTINE MATINV (A,N,B,M,DETERM,IPIVOT,INDEX,NMAX,ISCALE)          P  10
C                                                                          P  20
C     THIS SUBROUTINE SOLVES THE MATRIX EQUATION AX=B WHERE A IS A         P  30
C     SQUARE COEFFICIENT MATRIX AND B IS A MATRIX OF CONSTANT VECTORS.     P  40
C                                                                          P  50
      DIMENSION IPIVOT(N), A(NMAX,N), B(NMAX,M), INDEX(NMAX,2)             P  80
      EQUIVALENCE (IROW,JROW), (ICOLUM,JCOLUM), (AMAX,T,SWAP)              P  90
C                                                                          P 100
C     INITIALIZATION                                                       P 110
C                                                                          P 120
10    ISCALE=0                                                             P 130
      R1=10.0**100                                                         P 140
      R2=1.0/R1                                                            P 150
      DETERM=1.0                                                           P 160
      DO 20 J=1,N                                                          P 170
20    IPIVOT(J)=0                                                          P 180
      DO 360 I=1,N                                                         P 190
C                                                                          P 200
C     SEARCH FOR PIVOT ELEMENT                                             P 210
C                                                                          P 220
      AMAX=0.0                                                             P 230
      DO 70 J=1,N                                                          P 240
      IF (IPIVOT(J)-1) 30,70,30                                            P 250
30    DO 60 K=1,N                                                          P 260
      IF (IPIVOT(K)-1) 40,60,400                                           P 270
40    IF (ABS(AMAX)-ABS(A(J,K))) 50,60,60                                  P 280
50    IROW=J                                                               P 290
      ICOLUM=K                                                             P 300
      AMAX=A(J,K)                                                          P 310
60    CONTINUE                                                             P 320
70    CONTINUE                                                             P 330
      IF (AMAX) 90,80,90                                                   P 340
80    DETERM=0.0                                                           P 350
      ISCALE=0                                                             P 360
      GO TO 400                                                            P 370
90    IPIVOT(ICOLUM)=IPIVOT(ICOLUM)+1                                      P 380
C                                                                          P 390
C     INTERCHANGE ROWS TO PUT PIVOT ELEMENT ON DIAGONAL                    P 400
C                                                                          P 410
      IF (IROW-ICOLUM) 100,140,100                                         P 420
100   DETERM=-DETERM                                                       P 430
      DO 110 L=1,N                                                         P 440
      SWAP=A(IROW,L)                                                       P 450
      A(IROW,L)=A(ICOLUM,L)                                                P 460
110   A(ICOLUM,L)=SWAP                                                     P 470
      IF (M) 140,140,120                                                   P 480
120   DO 130 L=1,M                                                         P 490
      SWAP=B(IROW,L)                                                       P 500
      B(IROW,L)=B(ICOLUM,L)                                                P 510
130   B(ICOLUM,L)=SWAP                                                     P 520
140   INDEX(I,1)=IROW                                                      P 530
      INDEX(I,2)=ICOLUM                                                    P 540
      PIVOT=A(ICOLUM,ICOLUM)                                               P 550
      IF (PIVOT) 150,80,150                                                P 560
C                                                                          P 570
C     SCALE THE DETERMINANT                                                P 580
C                                                                          P 590
150   PIVOTI=PIVOT                                                         P 600
```

```
      IF (ABS(DETERM)-R1) 180,160,160                                  P 610
160   DETERM=DETERM/R1                                                 P 620
      ISCALE=ISCALE+1                                                  P 630
      IF (ABS(DETERM)-R1) 210,170,170                                  P 640
170   DETERM=DETERM/R1                                                 P 650
      ISCALE=ISCALE+1                                                  P 660
      GO TO 210                                                        P 670
180   IF (ABS(DETERM)-R2) 190,190,210                                  P 680
190   DETERM=DETERM*R1                                                 P 690
      ISCALE=ISCALE-1                                                  P 700
      IF (ABS(DETERM)-R2) 200,200,210                                  P 710
200   DETERM=DETERM*R1                                                 P 720
      ISCALE=ISCALE-1                                                  P 730
210   IF (ABS(PIVOTI)-R1) 240,220,220                                  P 740
220   PIVOTI=PIVOTI/R1                                                 P 750
      ISCALE=ISCALE+1                                                  P 760
      IF (ABS(PIVOTI)-R1) 270,230,230                                  P 770
230   PIVOTI=PIVOTI/R1                                                 P 780
      ISCALE=ISCALE+1                                                  P 790
      GO TO 270                                                        P 800
240   IF (ABS(PIVOTI)-R2) 250,250,270                                  P 810
250   PIVOTI=PIVOTI*R1                                                 P 820
      ISCALE=ISCALE-1                                                  P 830
      IF (ABS(PIVOTI)-R2) 260,260,270                                  P 840
260   PIVOTI=PIVOTI*R1                                                 P 850
      ISCALE=ISCALE-1                                                  P 860
270   DETERM=DETERM*PIVOTI                                             P 870
C                                                                      P 880
C     DIVIDE PIVOT ROW BY PIVOT ELEMENT                                P 890
C                                                                      P 900
      A(ICOLUM,ICOLUM)=1.0                                             P 910
      DO 280 L=1,N                                                     P 920
280   A(ICOLUM,L)=A(ICOLUM,L)/PIVOT                                    P 930
      IF (M) 310,310,290                                               P 940
290   DO 300 L=1,M                                                     P 950
300   B(ICOLUM,L)=B(ICOLUM,L)/PIVOT                                    P 960
C                                                                      P 970
C     REDUCE NON-PIVOT ROWS                                            P 980
C                                                                      P 990
310   DO 360 L1=1,N                                                    P1000
      IF (L1-ICOLUM) 320,360,320                                       P1010
320   T=A(L1,ICOLUM)                                                   P1020
      A(L1,ICOLUM)=0.0                                                 P1030
      DO 330 L=1,N                                                     P1040
330   A(L1,L)=A(L1,L)-A(ICOLUM,L)*T                                    P1050
      IF (M) 360,360,340                                               P1060
340   DO 350 L=1,M                                                     P1070
350   B(L1,L)=B(L1,L)-B(ICOLUM,L)*T                                    P1080
360   CONTINUE                                                         P1090
C                                                                      P1100
C     INTERCHANGE COLUMNS                                              P1110
C                                                                      P1120
      DO 390 I=1,N                                                     P1130
      L=N+1-I                                                          P1140
      IF (INDEX(L,1)-INDEX(L,2)) 370,390,370                           P1150
370   JROW=INDEX(L,1)                                                  P1160
      JCOLUM=INDEX(L,2)                                                P1170
      DO 380 K=1,N                                                     P1180
      SWAP=A(K,JROW)                                                   P1190
      A(K,JROW)=A(K,JCOLUM)                                            P1200
```

```
      A(K,JCOLUM)=SWAP                                          P1210
380   CONTINUE                                                  P1220
390   CONTINUE                                                  P1230
400   RETURN                                                    P1240
      END                                                       P1250-
```

```
      FUNCTION POLYE1 (X,M,C)                                          Q  10
C                                                                      Q  20
C     THIS IS A FUNCTION USED BY SUBROUTINE LSQPOL                     Q  30
C                                                                      Q  40
      DATA BIG/0377777777777/                                          Q  50
      DIMENSION C(M)                                                   Q  60
      IF (M-1) 30,40,10                                                Q  70
10    N=M-1                                                            Q  80
      POLYE1=C(1)                                                      Q  90
      DO 20 I=1,N                                                      Q 100
20    POLYE1=X*POLYE1+C(I+1)                                           Q 110
      RETURN                                                           Q 120
30    POLYE1=BIG                                                       Q 130
      RETURN                                                           Q 140
40    POLYE1=C(1)                                                      Q 150
      RETURN                                                           Q 160
      END                                                              Q 170-
```

```
      OVERLAY(RING,5,0)                                                    R  10
      PROGRAM CRITM                                                        R  20
C                                                                          R  30
C     THIS OVERLAY COMPUTES CRITICAL MACH NUMBER                           R  40
C                                                                          R  50
      COMMON /CORD/ X(170),R(170),C,RAVG,NN,N,DMIN,T(170),RB(170),XBN      R  60
      COMMON /FIVE/ CPO(170),MCR                                           R  70
      REAL MCR                                                             R  80
      CALL MCRIT (CPO,MCR)                                                 R  90
      RETURN                                                               R 100
      END                                                                  R 110-
```

```
      SUBROUTINE MCRIT (CPO,MCR)                                          S  10
C                                                                         S  20
C     THIS SUBROUTINE COMPUTES CRITICAL MACH NUMBER OF THE NACELLE BASED  S  30
C     ON THE OUTER SURFACE PRESSURE                                       S  40
C                                                                         S  50
      COMMON /CORD/ X(170),R(170),C,RAVG,NN,N                             S  60
      DIMENSION CPO(170)                                                  S  70
      REAL MCR                                                            S  80
      A=0.                                                                S  90
      DO 30 J=1,N                                                         S 100
      IF (CPO(J)) 10,30,30                                                S 110
10    IF (A.EQ.0.) GO TO 20                                               S 120
      IF (CPO(J).LT.A) A=CPO(J)                                           S 130
      GO TO 30                                                            S 140
20    A=CPO(J)                                                            S 150
30    CONTINUE                                                            S 160
      J=1                                                                 S 170
      MCR=.999                                                            S 180
40    CP=(-2./MCR)*((1.-MCR)**1.5)*(SQRT(1.+MCR))                         S 190
      IF (CP.LT.A) GO TO 50                                               S 200
      MCR=MCR-.001                                                        S 210
      J=J+1                                                               S 220
      IF (J.GE.999) GO TO 50                                              S 230
      GO TO 40                                                            S 240
50    CONTINUE                                                            S 250
      RETURN                                                              S 260
      END                                                                 S 270-
```

```
      OVERLAY(RING,6,0)                                                    T  10
      PROGRAM MASFLOW                                                      T  20
C                                                                          T  30
C     THIS OVERLAY COMPUTES VI/VIN BY INTEGRATING THE INTERNAL MASS FLOW   T  40
C     AT SOME X STATION.  THE X STATION IS PRESENTLY MIDCHORD BUT CAN      T  41
C     BE CHANGED BY CHANGING CARDS T 170, T 320, T 650, T 970.             T  42
C                                                                          T  50
      COMMON /CORD/ X(170),R(170),C,RAVG,NN,N,DMIN,T(170),RB(170),XB(      T  60
      COMMON /ONE/ ROUT(170),RIN(170),M,DMAX,XI(170),XIO(170),ALPHO(170)   T  70
     1,ALPHOO(170,1),XBL(170),ALPT(170),ALPTH(170),DRDX(170),DRDXH(170),   T  80
     2LAM,IT,ITMAX                                                         T  90
      COMMON /TWO/ IP,CAM,GAM(170),GQ(170),ALPOO(170,1)                    T 100
      COMMON /SIX/ NRHO,RAD(170),VIVIN,RXX,RBXI                            T 110
      DIMENSION RHOO(170)                                                  T 120
      REAL LAM,KSQ,M                                                       T 130
      PRINT 150                                                            T 140
      PRINT 130                                                            T 150
      PRINT 140                                                            T 160
      XXX=C/2.                                                             T 170
      CALL MTLUP (XXX,RXX,2,NN,170,1,IP,X,R)                               T 180
      CALL MTLUP (XXX,TXX,2,NN,170,1,IP,X,T)                               T 190
      RXX=RXX-(TXX/2.)                                                     T 200
      TEMP9=RXX/NRHO                                                       T 210
C                                                                          T 220
C     COMPUTE ALL RAD(J)                                                   T 230
C                                                                          T 240
      DO 20 J=1,NRHO                                                       T 250
      IF (J.EQ.NRHO) GO TO 10                                              T 260
      RAD(J)=RXX-(J*TEMP9)                                                 T 270
      GO TO 20                                                             T 280
10    RAD(J)=0.                                                            T 290
20    CONTINUE                                                             T 300
      VIVIN=0.                                                             T 310
      XX=C/2.                                                              T 320
      CALL MTLUP (XX,RBXI,2,NN,170,1,IP,X,RB)                              T 330
      IF (XX.LE.XBN) RBXI=0.                                               T 340
      TEST=RBXI/RAVG                                                       T 350
C                                                                          T 360
C     MASS FLOW - DO 90                                                    T 370
C                                                                          T 380
      DO 90 J=1,NRHO                                                       T 390
C                                                                          T 400
C     COMPUTE THE RHOO(J)                                                  T 410
C                                                                          T 420
      IF (J.EQ.1) GO TO 30                                                 T 430
      IF (J.EQ.NRHO) GO TO 40                                              T 440
      RHOO(J)=(RAD(J-1)+RAD(J))/(2.*RAVG)                                  T 450
      GO TO 50                                                             T 460
30    RHOO(J)=(RXX+RAD(J))/(2.*RAVG)                                       T 470
      GO TO 50                                                             T 480
40    RHOO(J)=RAD(J-1)/(2.*RAVG)                                           T 490
50    CONTINUE                                                             T 500
      IF (RHOO(J).LE.TEST) GO TO 100                                       T 510
C                                                                          T 520
C     VELOCITIES AT RHOO(J) - DO 60                                        T 530
C                                                                          T 540
      WX1JUO=0.                                                            T 550
      WR1JUO=0.                                                            T 560
      WX1QJU=0.                                                            T 570
      WR1QJU=0.                                                            T 580
```

```
      WXQJU=0.                                                          T 590
      WRQJU=0.                                                          T 600
      WXBQJU=0.                                                         T 610
      WRBQJU=0.                                                         T 620
      XX=(C/N)*.25                                                      T 630
      DO 60 I=1,N                                                       T 640
      DELXI=XI(I)                                                       T 650
      KSQ=(4.*RHOO(J))/((DELXI**2)+((1.+RHOO(J))**2))                   T 660
      EK=EKA(KSQ)                                                       T 670
      SK=SKK(KSQ)                                                       T 680
      CALL MTLUP (XX,RBXI,2,NN,170,1,IP,X,RB)                           T 690
      IF (XX.LE.XBN) RBXI=0.                                            T 700
      TEMP1=(1./(2.*SQRT((DELXI**2)+((1.+RHOO(J))**2))))*((((2.-KSQ)/(1.  T 710
     1-KSQ))*EK)-(2.*SK)-((1./RHOO(J))*(KSQ/(1.-KSQ))*EK))              T 720
      TEMP2=(((-1.)*DELXI)/(2.*RHOO(J)*SQRT((DELXI**2)+((1.+RHOO(J))**2)  T 730
     1)))*((((2.-KSQ)/(1.-KSQ))*EK)-(2.*SK))                           T 740
      TEMP7=(4.*DELXI*EK)/((SQRT((DELXI**2)+((1.+RHOO(J))**2)))*((DELXI*  T 750
     1*2)+((1.-RHOO(J))**2)))                                           T 760
      TEMP8=(2./(RHOO(J)*SQRT((DELXI**2)+((1.+RHOO(J))**2))))*(((1.-((2.  T 770
     1*RHOO(J)*(RHOO(J)-1.))/((DELXI**2)+((1.-RHOO(J))**2))))*EK)-SK)   T 780
      TEMP11=RHOO(J)/(2.*((DELXI**2+RHOO(J)**2)**1.5))                  T 790
      TEMP12=(DELXI)/(2.*((DELXI**2+RHOO(J)**2)**1.5))                  T 800
      WX1JUO=WX1JUO+TEMP1*ALPHO(I)*GAM(I)*(1./(2.*3.14159))             T 810
      WR1JUO=WR1JUO+TEMP2*ALPHO(I)*GAM(I)*(1./(2.*3.14159))             T 820
      WX1QJU=WX1QJU+(TEMP1*GQ(I)*(XI(1)-XI(2))/(2.*3.14159))            T 830
      WR1QJU=WR1QJU+(TEMP2*GQ(I)*(XI(1)-XI(2))/(2.*3.14159))            T 840
      WXQJU=WXQJU-(TEMP7*ALPT(I)*(XI(1)-XI(2))/(2.*3.14159))            T 850
      WRQJU=WRQJU+(TEMP8*ALPT(I)*(XI(1)-XI(2))/(2.*3.14159))            T 860
      WXBQJU=WXBQJU+TEMP12*(RBXI/RAVG)*DRDX(I)*(XI(1)-XI(2))            T 870
      WRBQJU=WRBQJU+TEMP11*(RBXI/RAVG)*DRDX(I)*(XI(1)-XI(2))            T 880
      XX=XX+(C/N)                                                       T 890
60    CONTINUE                                                          T 900
      WX1JUOI=WX1JUO                                                    T 910
      WX1QJUI=WX1QJU                                                    T 920
      VTI=(1.+(WX1JUOI+WX1QJUI+WXQJU+WXBQJU)/(1.-M**2))**2              T 930
C                                                                       T 940
C     SUMMATION FOR MASS FLOW - ADD ONE TERM EACH TIME THRU DO 90       T 950
C                                                                       T 960
      XX=C/2.                                                           T 970
      CALL MTLUP (XX,RBXI,2,NN,170,1,IP,X,RB)                           T 980
      IF (XX.LE.XBN) RBXI=0.                                            T 990
      IF (J.EQ.1) GO TO 70                                             T1000
      TEMP10=SQRT(VTI)*((RAD(J-1)**2-RAD(J)**2)/(RXX**2-RBXI**2))      T1010
      GO TO 80                                                         T1020
70    TEMP10=SQRT(VTI)*((RXX**2-RAD(J)**2)/(RXX**2-RBXI**2))           T1030
80    CONTINUE                                                         T1040
      VIVIN=VIVIN+TEMP10                                               T1050
      WRITE (6,120) J,RAD(J),RHOO(J),WX1JUO,WR1JUO,WX1QJU,WR1QJU,WXQJU,W T1060
     1RQJU                                                             T1070
      WRITE (6,110) VTI,VIVIN,TEMP10,WXBQJU,WRBQJU                     T1080
90    CONTINUE                                                         T1090
100   CONTINUE                                                         T1100
      RETURN                                                           T1110
C                                                                      T1120
C                                                                      T1130
110   FORMAT (13X,5(2X,E11.4)/)                                        T1140
120   FORMAT (1X,I3,3(2X,F10.5),6(2X,E11.4))                           T1150
130   FORMAT (1X,1HJ,9X,3HRAD,8X,4HRHOO,6X,6HWX1JUO,7X,6HWR1JUO,7X,6HWX1 T1160
     1QJU,7X,6HWR1QJU,8X,5HWXQJU,8X,5HWRQJU)                           T1170
140   FORMAT (23X,3HVTI,8X,5HVIVIN,7X,6HTEMP10,7X,6HWXBQJU,7X,6HWRBQJU/) T1180
```

```
150   FORMAT (1H1,21HMASS FLOW COMPUTATION,10X,49HALL PERTURB VEL AND RA  T1190
     1D ARE FOR EQUIV INCOMP GEOM//)                                     T1200
      END                                                                T1210-
```

APPENDIX B

DERIVATION OF EQUATIONS (13) AND (23)

The velocity induced in the x-direction at ξ_o,ρ_o by a distribution of vortex rings on a cylinder of radius $\rho = 1$ and of chord length c is found by use of equation (21). The result is

$$\frac{u_\gamma(\xi_o,\rho_o)}{V_\infty} = \frac{1}{2\pi}\int_{-1/\lambda}^{1/\lambda} \alpha_o(\xi)\gamma_o(\xi)\frac{k}{4\sqrt{\rho_o}}\left[\left(\frac{2-k^2}{1-k^2} - \frac{1}{\rho_o}\frac{k^2}{1-k^2}\right)E - 2K\right]d\xi \qquad \text{(B1)}$$

Replacing the modulus k by its definition in equation (11) and rearranging the result, equation (B1) becomes

$$\frac{u_\gamma(\xi_o,\rho_o)}{V_\infty} = \frac{1}{2\pi}\int_{-1/\lambda}^{1/\lambda} \alpha_o(\xi)\gamma_o(\xi)\frac{E-K}{\sqrt{(\xi-\xi_o)^2+(1+\rho_o)^2}}d\xi$$

$$+\frac{1}{2\pi}\int_{-1/\lambda}^{1/\lambda} \frac{\alpha_o(\xi)\gamma_o(\xi)}{\sqrt{(\xi-\xi_o)^2+(1+\rho_o)^2}}\frac{2(\rho_o-1)E}{(\xi-\xi_o)^2+(\rho_o-1)^2}d\xi \qquad \text{(B2)}$$

Equation (B2) must now be evaluated at $\rho_o = 1$ to obtain equation (13). Considering the first integral, at $\rho_o = 1$ and $\xi = \xi_o$, a singularity exists in the term containing the elliptic integral of the first kind K. To investigate this singularity, examine the integral in a small region extending ϵ on each side of ξ_o

$$\lim_{\epsilon\to 0} -\frac{1}{2\pi}\int_{\xi_o-\epsilon}^{\xi_o+\epsilon} \frac{\alpha_o(\xi)\gamma_o(\xi)K\,d\xi}{\sqrt{(\xi-\xi_o)^2+4}} \approx \frac{-\alpha_o(\xi_o)\gamma_o(\xi_o)}{2\pi}\lim_{\epsilon\to 0}\int_{\xi_o-\epsilon}^{0} K\,d\xi \qquad \text{(B3)}$$

where the terms multiplying K are approximated by setting $\xi = \xi_o$. The elliptic integral K can be expanded in a series as a function of the complimentary modulus k' (ref. 23) in the following form

$$K = \sum_{m=0}^{\infty} \frac{\left(\frac{1}{2}\right)_m\left(\frac{1}{2}\right)_m}{m!m!}\left[\ln\left(\frac{1}{k'}\right) - \psi\left(m+\frac{1}{2}\right) + \psi(m+1)\right](k')^{2m} \qquad \left((k')^2 < 1\right) \qquad \text{(B4)}$$

where

$$k' = \sqrt{1 - k^2} = \sqrt{\frac{(\xi - \xi_o)^2}{(\xi - \xi_o)^2 + 4}}$$

and Pochhammer's symbol is defined by

$$\left(\frac{1}{2}\right)_m = \frac{1}{2}\left(\frac{1}{2} + 1\right)\left(\frac{1}{2} + 2\right) \cdots \left(\frac{1}{2} + m - 1\right)$$

$$\left(\frac{1}{2}\right)_o = 1$$

and ψ is the digamma function (ref. 24). By using equation (B4) and letting $\epsilon \to 0$, and therefore $k' \to 0$, it can be shown that the integral in equation (B3) goes to 0. Thus the principal value of the first integral in equation (B2) is found by treating it as an ordinary integral.

For $\rho_o = 1$, the value of the second integral in equation (B2) can be determined as follows. For $\xi \neq \xi_o$, the integrand is 0 when $\rho_o = 1$. Hence the limit must be found as $\xi \to \xi_o$ simultaneously with $\rho_o \to 1$

$$\lim_{\rho_o \to 1} \frac{1}{2\pi} \int_{-1/\lambda}^{1/\lambda} \frac{\alpha_o(\xi)\gamma_o(\xi)}{\sqrt{(\xi - \xi_o)^2 + (1 + \rho_o)^2}} \frac{2(\rho_o - 1)E}{(\xi - \xi_o)^2 + (\rho_o - 1)^2} d\xi$$

$$\approx \frac{\alpha_o(\xi_o)\gamma_o(\xi_o)}{2\pi} \lim_{\substack{\epsilon \to 0 \\ \rho_o \to 1}} \int_{\xi_o - \epsilon}^{\xi_o + \epsilon} \frac{\rho_o - 1}{(\xi - \xi_o)^2 + (\rho_o - 1)^2} d\xi$$

where $E \approx 1$ because $k \approx 1$, and again the first term in the integrand is approximated by setting $\xi = \xi_o$. By defining a new variable

$$\xi' \equiv \frac{\xi - \xi_o}{\rho_o - 1}$$

and requiring $\rho_o \to 1$ faster than $\epsilon \to 0$ for a nontrivial solution, it is found that

$$\lim_{\substack{\epsilon \to 0 \\ \rho_o \to 1}} \int_{\xi_o - \epsilon}^{\xi_o + \epsilon} \frac{\rho_o - 1}{\left(\xi - \xi_o\right)^2 + \left(\rho_o - 1\right)^2} \, d\xi = \lim_{\substack{\epsilon \to 0 \\ \rho_o \to 1}} \int_{-\frac{\epsilon}{\rho_o - 1}}^{\frac{\epsilon}{\rho_o - 1}} \frac{d\xi'}{(\xi')^2 + 1} = \pm\pi$$

so that

$$\lim_{\rho_o \to 1} \frac{1}{2\pi} \int_{-1/\lambda}^{1/\lambda} \frac{\alpha_o(\xi)\gamma_o(\xi)}{\sqrt{\left(\xi - \xi_o\right)^2 + \left(1 + \rho_o\right)^2}} \frac{2\left(\rho_o - 1\right)E}{\left(\xi - \xi_o\right)^2 + \left(\rho_o - 1\right)^2} \, d\xi = \pm \frac{\alpha_o\left(\xi_o\right)}{2} \frac{\Gamma_o\left(\xi_o\right)}{\Delta\xi} \tag{B5}$$

where $\gamma_o\left(\xi_o\right) = \Gamma_o\left(\xi_o\right)/\Delta\xi$. By substituting equation (B5) into equation (B2) and setting $\rho_o = 1$

$$\frac{u_\gamma\left(\xi_o, 1\right)}{V_\infty} = \int_{-1/\lambda}^{1/\lambda} \frac{\alpha_o(\xi)\,\gamma_o(\xi)}{2\pi} \frac{E - K}{\sqrt{\left(\xi - \xi_o\right)^2 + 4}} \, d\xi \pm \frac{\alpha_o\left(\xi_o\right)}{2} \frac{\Gamma_o\left(\xi_o\right)}{\Delta\xi} \tag{B6}$$

Writing equation (B6) in summation form yields equation (13). Equation (23) is derived by a procedure analogous to the preceding derivation.

REFERENCES

1. Hess, J. L.; and Smith, A. M. O.: Calculation of Potential Flow About Arbitrary Bodies. Progress in Aeronautical Sciences, Vol. 8, D. Küchemann, ed., Pergamon Press, Ltd., c.1967, pp. 1-138.

2. Trulin, Darryl J.; and Iversen, James D.: Pressure Distribution About a Finite Axisymmetric Nacelle. J. Aircraft, vol. 7, no. 1, Jan.-Feb. 1970, pp. 85-87.

3. Geissler, Wolfgang: Berechnung der Potentialströmung um rotationssymmetrische Rümpfe, Ringprofile und Triebwerkseinläufe. Z. Flugwiss., Jahrg. 20, Heft 12, Dec. 1972, pp. 457-462.

4. Young, C.: A Computer Program to Calculate the Pressure Distribution on an Annular Aerofoil. C.P. No. 1217, Brit. A.R.C., 1972.

5. Young, C.: An Investigation of Annular Aerofoils for Turbofan Engine Cowls. R. & M. No. 3688, Brit. A.R.C., 1972.

6. Mascitti, Vincent R.: A Rapid Numerical Solution to Subsonic Flow Over Planar and Axisymmetric Profiles at an Angle of Attack of 0°. NASA TN D-7410, 1974.

7. Belotserkovskii, Sergei Mikhailovich (Maurice Holt, transl. ed.): The Theory of Thin Wings in Subsonic Flow. Plenum Press, 1967, pp. 1-17, 91-104, 105-126.

8. Weissinger, Johannes: The Influence of Profile Thickness on Ring Airfoils in Steady Incompressible Flow. AFOSR TR 57-8, DDC No. AD 120 424, U.S. Air Force, Jan. 1957.

9. Hess, J. L.: Numerical Solution of the Integral Equation for the Neumann Problem With Application to Aircraft and Ships. Paper presented at Symposium on Numerical Solution of Integral Equations With Physical Applications, SIAM Fall Meeting (Univ. of Wisconsin), Oct. 1971.

10. Keith, J. S.; Ferguson, D. R.; Merkle, C. L.; Heck, P. H.; and Lahti, D. J.: Analytical Method for Predicting the Pressure Distribution About a Nacelle at Transonic Speeds. NASA CR-2217, 1973.

11. Grossman, Bernard; and Moretti, Gino: Development of Analytical Methods of Predicting the Pressure Distribution About a Nacelle at Transonic Speeds – Exact Solution. NASA CR-112271, 1973.

12. Küchemann, Dietrich; and Weber, Johanna: Aerodynamics of Propulsion. McGraw-Hill Book Co., Inc., 1953.

13. Ashley, Holt; and Landahl, Marten: Aerodynamics of Wings and Bodies. Addison-Wesley Pub. Co., Inc., c.1965.

14. Weber, J.: The Calculation of the Pressure Distribution Over the Surface of Two-Dimensional and Swept Wings With Symmetrical Aerofoil Sections. R. & M. No. 2918, Brit. A.R.C., 1956.

15. Schlichting, Hermann (J. Kestin, transl.): Boundary Layer Theory. Fourth ed., McGraw-Hill Book Co., Inc., c.1960.

16. Cebeci, Tuncer; Mosinskis, G. J.; and Smith, A. M. O.: Calculation of Viscous Drag of Two-Dimensional and Axisymmetric Bodies in Incompressible Flows. AIAA Paper No. 72-1, Jan. 1972, p. 5.

17. Stevens, W. A.; Goradia, S. H.; and Braden, J. A.: Mathematical Model for Two-Dimensional Multi-Component Airfoils in Viscous Flow. NASA CR-1843, 1971.

18. von Kármán, Th.: Compressibility Effects in Aerodynamics. J. Aeronaut. Sci., vol. 8, no. 9, July 1941, pp. 337-356.

19. Tulinius, J. R.: Theoretical Prediction of Thick Wing Aerodynamic Characteristics at Subsonic Speeds. NA-70-104, Los Angeles Div., North American Rockwell Corp., Oct. 9, 1970.

20. Re, Richard J.: An Investigation of Several NACA 1-Series Axisymmetric Inlets at Mach Numbers From 0.4 to 1.29. NASA TM X-2917, 1974.

21. Chudyk, D. W.: Transonic Wind Tunnel Tests of Several NASA Bodies of Revolution. Rep. No. AA-4018-W-5 (Contract No, NAS-1-10649), Cornell Aeronaut. Lab., Inc., Jan./Mar. 1971. (Available as NASA CR-112069.)

22. Shollenberger, Carl A.: An Investigation of a Two-Dimensional Propulsive Lifting System. Ph. D. Thesis, California Inst. Technol., 1971.

23. Byrd, Paul F.; and Friedman, Morris D.: Handbook of Elliptic Integrals for Engineers and Physicists. Lange, Maxwell & Springer, Ltd., 1954, p. 298.

24. Davis, Philip J.: Gamma Function and Related Functions. Handbook of Mathematical Functions, Milton Abramowitz and Irene A. Stegun, eds., Dover Publ., Inc., Fifth printing, pp. 253-293.

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
WASHINGTON, D.C. 20546

OFFICIAL BUSINESS
PENALTY FOR PRIVATE USE $300

SPECIAL FOURTH-CLASS RATE BOOK

POSTAGE AND FEES PAID
NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
451

U.S.MAIL

POSTMASTER: If Undeliverable (Section 158 Postal Manual) Do Not Return

"The aeronautical and space activities of the United States shall be conducted so as to contribute . . . to the expansion of human knowledge of phenomena in the atmosphere and space. The Administration shall provide for the widest practicable and appropriate dissemination of information concerning its activities and the results thereof."

—NATIONAL AERONAUTICS AND SPACE ACT OF 1958

NASA SCIENTIFIC AND TECHNICAL PUBLICATIONS

TECHNICAL REPORTS: Scientific and technical information considered important, complete, and a lasting contribution to existing knowledge.

TECHNICAL NOTES: Information less broad in scope but nevertheless of importance as a contribution to existing knowledge.

TECHNICAL MEMORANDUMS: Information receiving limited distribution because of preliminary data, security classification, or other reasons. Also includes conference proceedings with either limited or unlimited distribution.

CONTRACTOR REPORTS: Scientific and technical information generated under a NASA contract or grant and considered an important contribution to existing knowledge.

TECHNICAL TRANSLATIONS: Information published in a foreign language considered to merit NASA distribution in English.

SPECIAL PUBLICATIONS: Information derived from or of value to NASA activities. Publications include final reports of major projects, monographs, data compilations, handbooks, sourcebooks, and special bibliographies.

TECHNOLOGY UTILIZATION PUBLICATIONS: Information on technology used by NASA that may be of particular interest in commercial and other non-aerospace applications. Publications include Tech Briefs, Technology Utilization Reports and Technology Surveys.

Details on the availability of these publications may be obtained from:

SCIENTIFIC AND TECHNICAL INFORMATION OFFICE

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION

Washington, D.C. 20546