AD-A046 461 STANFORD UNIV CALIF THERMOSCIENCES DIV F/G 20/4

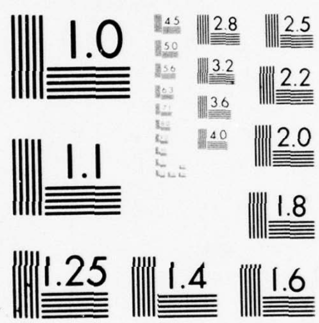
PREDICTION OF TRANSITORY STALL IN TWO-DIMENSIONAL DIFFUSERS.(U)

DEC 76 S GHOSE, S J KLINE F44620-74-C-0016

UNCLASSIFIED MD-36 AFOSR-TR-77-1278 NL

1 OF 2

AD A046461



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

AFOSR-TR- 77 - 1278

PREDICTION OF TRANSITORY STALL
IN TWO-DIMENSIONAL DIFFUSERS

by

S. Ghose and S. J. Kline

Prepared from work sponsored by the
U. S. Air Force Office of Scientific Research
Mechanics Division, Contract [redacted] F44620-74-C-0

Report MD-36

THERMOSCIENCES DIVISION
DEPARTMENT OF MECHANICAL ENGINEE
STANFORD UNIVERSITY
STANFORD, CALIFORNIA

December 1976

6

PREDICTION OF TRANSITORY STALL IN TWO-DIMENSIONAL DIFFUSERS.

by

10

S. Ghose [redacted] S. J. Kline

9 Interim rept.

Prepared from work sponsored by the
U. S. Air Force Office of Scientific Research,
Mechanics Division, Contract [redacted] F44620-74-C-0016

15

14 Report MD-36

16 2307

17 A4

18 AFOSR

19 TR-77-1278

12 184p.

Thermosciences Division
Department of Mechanical Engineering
Stanford University
Stanford, California

11 December 1976

1473

401 973

Acknowledgments

This research project was sponsored by the Mechanics Division of the U. S. Air Force Office of Scientific Research, Contract Number [redacted]F44620-74-C-0016.

Thanks are extended to Professor J. P. Johnston and Professor J. H. Ferziger for reviewing the manuscript and for their helpful suggestions throughout the project.

The authors also adknowledge the excellent job of typing done by Ms. Ruth Korb.

PREDICTION OF TRANSITORY STALL IN TWO-DIMENSIONAL DIFFUSERS

A method has been developed that predicts the performance of diffusers operating in the transitory stall mode of the flow regime chart of Fox and Kline. The calculations are accurate within ± 6%, which is of the same order as the uncertainty in the data for diffusers with divergence angles that are 1.2 times that at which line a-a occurs. This corresponds approximately to the line of appreciable stall.

Singular behavior in the neighborhood of detachment is avoided by simultaneous calculation of the inviscid core and the shear layers. A new boundary layer scheme using Bradshaw's entrainment-maximum shear correlation is developed that is valid for both attached and detached flows. The irrotational core is first assumed one-dimensional and then extended to the two-dimensional case by an iterative scheme consisting of alternate calculations of the boundary layers and Laplace's equation in the core.

The basic boundary layer method is shown to be of comparable accuracy as the best calculation presented in the 1968 Stanford Conference on Computation of Turbulent Boundary Layers. When compared against the data maps of Reneau et al. and the measurements of Carlson et al., the one-dimensional core model gives excellent agreement for the streamwise distribution of the shape factor H, the displacement thickness, and skin friction coefficient $C_f/2$, as well as for the locations of intermittent detachment and time-averaged zero wall shear. The two-dimensional model predicts the same quantities to the accuracy in the data for the flow of Strickland and Simpson. However, in the reversed flow portion, the predicted skin friction is somewhat low, and the entrainment much too high. In all cases, the largest deviation from data occurs in the region between intermittent detachment and the location of time-averaged zero wall shear. Complete verification of the method, or its improvement in this zone, must await further data.

Table of Contents

Table of Contents (cont.)

List of Figures

Figure		Page
20	Effect of lag parameter λ_a on the predictions of the unstalled diffuser flow of Carlson et al. [27]	85
21	Effect of lag parameter λ_d on the calculations of a diffuser in transitory stall as measured by Carlson et al. [27]	86
22	Predicted variation of boundary layer quantities H, δ^* and $C_f/2$ along the walls of a diffuser operating in the transitory stall regime	87
23	Predicted performance of $N/W_1 = 6$, $B_1 = .030$ diffuser family, as compared against the data of Carlson et al. [27], and the data maps of Reneau et al. [45]	88
24	Predicted exit conditions for $N/W_1 = 6$, $B_1 = .030$ diffuser family	89
25	Predicted variation of exit C_p, location of intermittent detachment $(H = H_{sep})$, and zero wall shear $(C_f = 0)$ location as fraction of length (X/L)	90
26	Summary of $N/W_1 = 12$ diffusers, comparing the data maps of of Reneau et al. [45] with 1-D core diffuser prediction .	91
27	Summary of performance of all tested diffusers	92
28	Comparison of the data of Moses [7] on an asymmetrical diffuser with the 2-D core calculation	93
29	Comparison of the data of Moses [7] on an asymmetrical diffuser with the 2-D core calculation	94
30	Strickland-Simpson [32] flow, comparing data with predictions	95
31	Strickland-Simpson [32] flow	96

Nomenclature

AR	Area ratio of diffuser, W2/W1
B	Boundary layer blockage, $2\delta^*/W$
c	Constant for the law of the wall (= 5.0)
$\hat{c}$	Redefined c (= 2.05)
C_D	Dissipation integral, Eqn. (2-5)
C_f	Skin-friction coefficient, $\tau_w/\frac{1}{2}\rho U_\infty^2$
$\overline{Cp}$	Diffuser exit Cp, averaged over space and time
Cp^*	Averaged value of the peak exit Cp
Cp(x)	Local pressure recovery, $1 - (U_\infty(x)/U_o)^2$
C_τ	Shear stress integral across the boundary layer, Eqn. (2-7)
H	Boundary layer shape factor, δ^*/θ
$\overline{H}$	Energy shape factor, δ^{**}/θ
H_s	Senoo-Nishi separation criterion, Eqn. (1-10)
H_{sep}	Sandborn-Kline separation criterion, Eqn. (1-9)
$H_{\delta-\delta^*}$	Mass defect shape factor, Eqn. (2-10)
IT	Location of intermittent transitory stall
K_e	Clauser's outer layer eddy-viscosity constant, Eqn. (2-38)
PL	Left hand side of normalized momentum integral, Eq. (3-1)
PR	Right hand side of normalized momentum integral, Eq. (3-1)
p	Static pressure
Q	Volumetric flow rate
Re_δ	Reynolds number based on δ, $U_\infty\delta/\nu$
Re_θ	Reynolds number based on θ, $U_\infty\theta/\nu$
TI	Location of incipient transitory stall

u	Mean velocity in the streamwise direction
u'	Turbulent velocity fluctuation in the streamwise direction
U_∞	Streamwise velocity at the edge of the shear layer
U_e	Effective core velocity, Eqn. (5-4)
u_τ	Shear velocity, Eqn. (2-15)
u_β	Wake amplitude, Eqn. (2-15)
v	Mean velocity in cross-stream direction, normal to the wall
v'	Turbulent fluctuation in the cross-stream direction
V_T	Non-dimensional shear velocity, Eqn. (2-16)
V_B	Non-dimensional wake amplitude, Eqn. (2-16)
w	Mean velocity in the spanwise direction
W	Width of diffuser
W_e	Effective width available to the throughflow, Eqn. (5-3)
x	Streamwise coordinate along the diffuser walls
x_c	Location of fictitious source or sink, Eqn. (3-2)
y	Cross-stream coordinate, normal to the wall
y^+	Non-dimensional cross-stream distance, Eqn. (2-15)
z	Coordinate location in the complex plane
2θ	Total divergence angle of the diffuser

Greek Symbols

α	Angle between streamlines and the positive x direction, Eqn. (5-13)
β	Clauser's equilibrium parameter, Eqn. (2-39)
γ	Intermittency in the layer, Eqn. (2-39)
δ	Boundary layer thickness
δ^*	Displacement thickness, $\int_0^\infty \left(1 - \frac{u}{U_\infty}\right) dy$

δ^{**}	Energy thickness, $\int_0^\infty \left(\frac{u}{U_\infty}\right)^2 \left(1 - \frac{u}{U_\infty}\right) dy$
ε	Eddy viscosity, Eqn. (2-37)
η	Nondimensional distance in the layer, y/δ
ϕ	Functional form for turbulent shear stress model, Eqn. (2-12)
κ	von Kármán constant (≈ 0.41)
λ	Lag parameter
λ_a	Attached flow lag parameter
λ_d	Lag parameter for detached flows
ν	Kinematic viscosity
Π	Coles' wake parameter
ρ	Mass density
τ	Turbulent shear stress $(= -\rho\overline{u'v'})$
θ	Momentum thickness, $\int_0^\infty \left(\frac{u}{U_\infty}\right)\left(1 - \frac{u}{U_\infty}\right) dy$
Λ	Geometry coefficients for solution of Laplace's equation (5-10)

Subscripts

a-a	Evaluated at the location of line a-a on flow regime chart		
max	Maximum value		
τ_{max}	Value at which the maximum shear occurs		
w	Evaluated at the wall		
eq	Equilibrium values		
1D	One-dimensional core model		
2D	Two-dimensional core model	0	Reference condition
U	Upper wall values	1	Inlet values
S	Lower wall values	2	Exit values

CHAPTER ONE

INTRODUCTION

A. Objective

The objective of this investigation is to develop a method for predicting the performance of two-dimensional (2-D) diffusers operating in the "unstalled" and "transitory stalled" regimes of the diffuser performance chart of Fox and Kline [1], as shown in Fig. 1.

A typical curve of static pressure recovery, Cp, as a function of the divergence angle 2θ is shown in Fig. A.† Line a-a on both figures represents the approximate dividing line between the unstalled and the transitory stalled regimes. Current calculation methods [4,5] can make successful predictions in the shaded zones only. These consist of the fully stalled regime and the unstalled zone for diffusers with $2\theta/2\theta_{a-a} \leq 0.6$ to 0.8. It will be noted that the peak pressure recovery, Cp^*, occurs in the transitory stall regime, where the flow is unsteady and the boundary layers (b.l.'s) along the diffuser walls are partially detached. (Separated or stalled b.l.'s shall be referred to as detached b.l.'s to avoid confusion between stalled b.l.'s and stalled diffusers.)

The ability to predict diffuser performance in the region near Cp^* is of obvious interest to the designer of flow equipment. However, a prerequisite to being able to do this is the calculation of b.l.'s which are attached and partially detached. Accordingly, the first few chapters of this report are devoted to the development of such a turbulent boundary layer prediction method (TBLPM), which is then used to predict diffusers operating in the region near Cp^*.

B. Cyclic Iteration

The classical method for predicting the development of b.l.'s is to prescribe the pressure gradient dp/dx in the flow direction and calculate the dependent b.l. parameters through a parabolic marching scheme

† Figures with lettered titles (Fig. A, Fig. B, etc.) are embodied within the text. Numbered figures (Fig. 1, Fig. 2, etc.) are collected at the end.

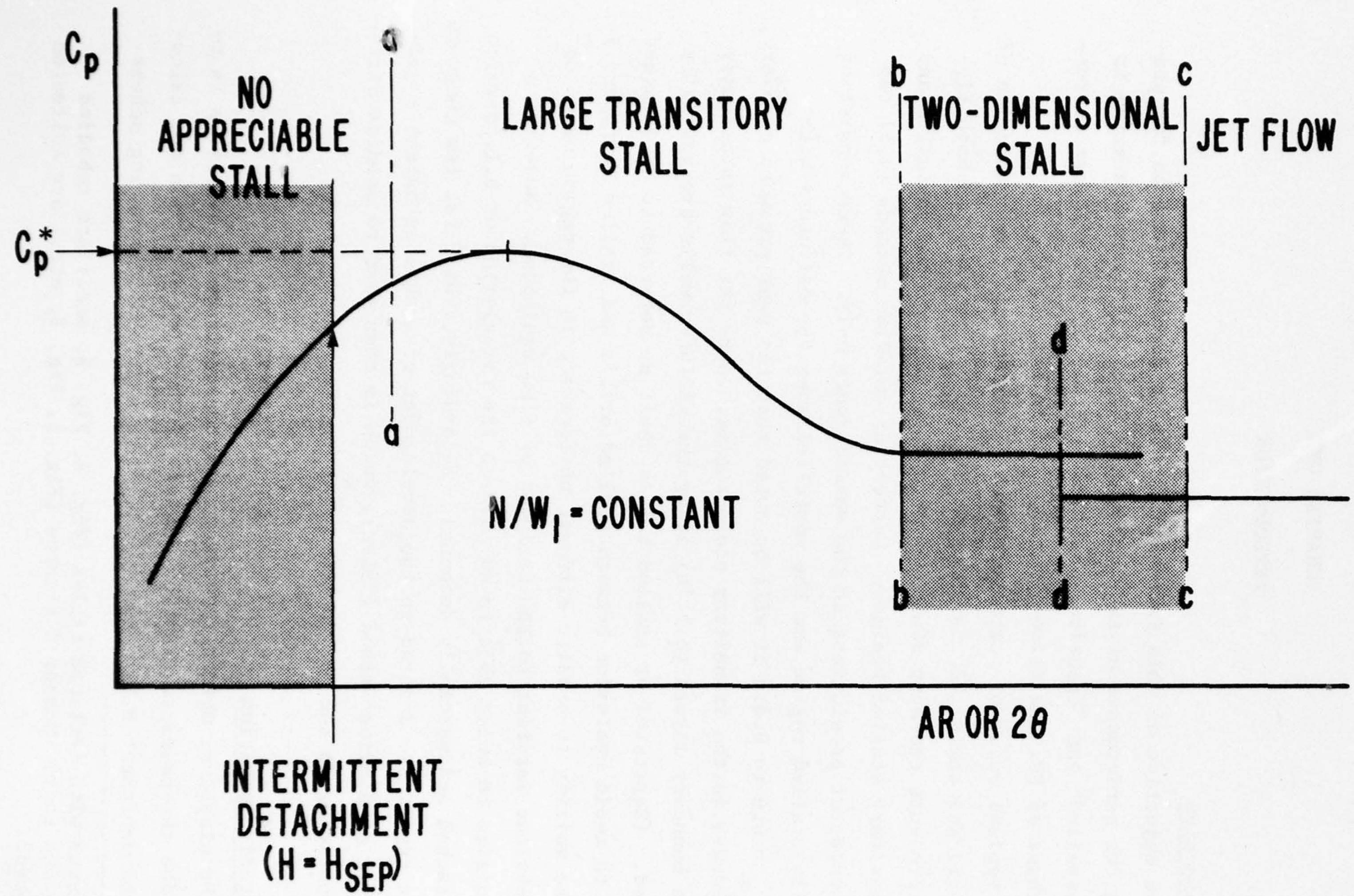


Fig. A. Behavior of C_p with increasing area ratio or 2θ.

[2,3]. The a priori imposition of the pressure gradient implies that the b.l. does not greatly affect the free stream velocity, an assumption that is true only for attached flows with b.l.'s that are thin compared to passage height.

The pressure gradient acting on the b.l. is in fact the result of mutual interaction between itself and the adjacent irrotational fluid. This interaction assumes an increasingly important role in adverse pressure gradients, as the "blockage" of the b.l. becomes greater and greater. Finally, for flows at and near detachment, it will be shown to be the controlling factor in determining whether or not such calculations can be made to converge.

This problem is much more serious for internal flow than for external flow, because in internal flow the irrotational core is confined between b.l.'s growing on the bounding walls and the blockage is large enough in typical passages to cause a substantial amount of mutual interaction.

Several schemes for fully stalled and attached flows have recently appeared [4,5], wherein the classical turbulent boundary layer prediction methods (TBLPM's) with prescribed pressure gradient have been coupled with an inviscid core and the calculation iterated to closure using a scheme such as shown in Fig. B. An initial estimate of the pressure gradient is impressed upon and used to calculate the b.l.'s, which in turn supply an estimate of the displacement thickness, δ^*. The blockage is subtracted from the channel width, giving a new body shape, which is then used to calculate the new pressure gradient and so on, hopefully to convergence. We shall call such schemes "CYCLIC ITERATION".

Consider the case of a rapidly detaching flow, such as in a stalled diffuser. As the b.l. approaches detachment, δ^* grows very rapidly. In a real physical situation, this rapid increase in blockage will result in a simultaneous decrease in pressure gradient. This is so because the mutual interaction between the b.l. and the potential core decreases the effective flow channel (EFC) available to the outer flow, thereby relaxing the pressure gradient as shown in Fig. C.

In cyclic iteration, however, the pressure gradient is fixed beforehand for the entire iteration, and there is no mechanism available to

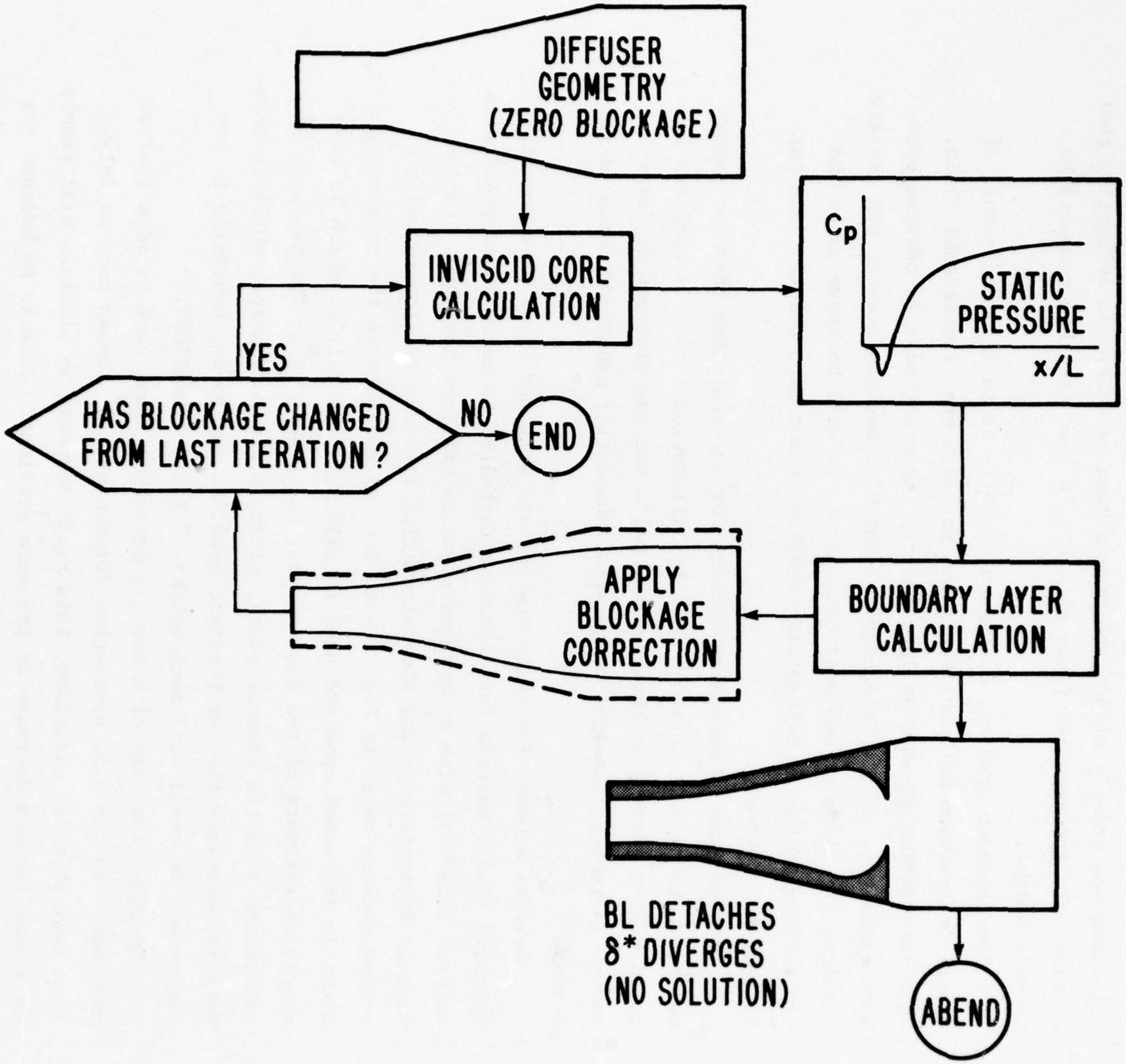


Fig. B. Flowchart of cyclic iteration as applied to the calculation of diffusers.

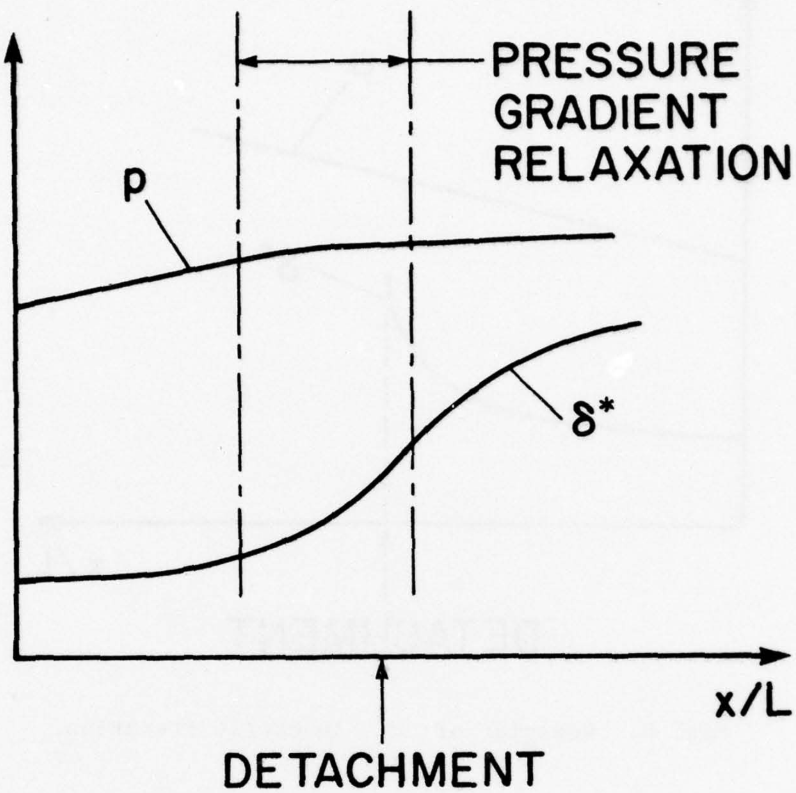


Fig. C. Behavior of δ^* in a real flow.

reduce the pressure gradient in reaction to the sudden growth of δ^* (Fig. D) in that iteration. The adverse pressure gradient is therefore maintained unchanged, resulting in runaway growth of δ^* and catastrophic failure of the prediction scheme. This is the so-called "separation singularity", the effects of which can be seen rather dramatically in many of the "separating flows" of reference [3].

A related effect is the inability of the calculation method to predict detachment, even though the prescribed pressure gradient was obtained experimentally from a separated flow in which pressure gradient relaxation has occurred. This, too, may be observed in several of the predictions in [3], and is again the result of not including the free-stream interaction explicitly into the calculation.

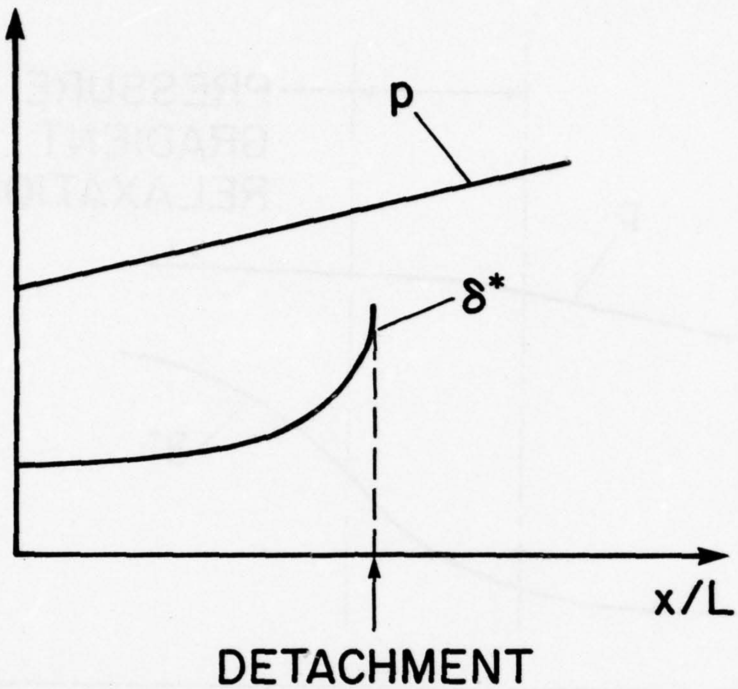


Fig. D. Behavior of δ^* in cyclic iteration.

Because of the unavoidable approximations involved in modeling the turbulent shear stresses and small errors in measurement, the phase relationship between the pressure gradient and the dependent b.l. variables in the actual flow can never be exactly duplicated in a calculation using this very same pressure gradient as a boundary condition. Therefore the experimental growth of δ^* does not exactly match the calculated value. If the calculated value is slightly ahead of the measured one, runaway growth of δ^* will occur. Conversely, if the calculated δ^* lags, the freestream pressure gradient will relax prematurely and the b.l. will not detach; near detachment, the classical b.l. procedure tends to become unstable.

Several methods are used in practice to avoid this problem of non-detachment of a calculated b.l. from experimental data. A very popular scheme is the "frozen dp/dx" method, wherein the pressure gradient is maintained at its maximum value and prescribed on the b.l. until it detaches. Cebeci et al. [6] present several comparisons of this method against the separation criteria of Head, Goldschmied and Stratford.

There are several objections to the use of methods such as the frozen dp/dx method. A primary one is that it will predict detachment in cases where none should occur, such as in the case of a flow which is decelerated and then allowed to relax. In addition, there is no physical basis for the method, even though it gives good answers for the location of zero wall shear for rapidly detaching flows.

C. Simultaneous Iteration

The heuristic explanation given above suggests that the "separation singularity" and the inability to predict detachment is nothing more than prescription of the wrong boundary conditions on the b.l. equations.

If a method could be devised wherein the pressure gradient at any given point is the result of mutual interaction between the b.l. and the inviscid core, then no such singular behavior should occur. In this type of calculation, the pressure gradient (or equivalently the core velocity at the edge of the b.l., u_∞) is assumed unknown, and an additional equation, commonly a 1-D continuity equation in the core, is added. This set of equations is solved simultaneously at each step along the flow. We shall call this scheme "SIMULTANEOUS ITERATION"; its main features are outlined in Fig. E. In Chapter Five the method will be extended to the case where the edge velocity is obtained from a solution of the 2-D LaPlace's equation in the inviscid core.

A mathematical description of cyclic and simultaneous iteration follows. For steady, two-dimensional, incompressible flow, the b.l. equations are

x momentum:
$$u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} = -\frac{1}{\rho}\frac{\partial p}{\partial x} + \frac{\partial \tau}{\partial y} , \tag{1-1}$$

continuity:
$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0 , \tag{1-2}$$

y momentum:
$$\frac{\partial p}{\partial y} \simeq 0 , \quad \text{so that} \quad -\frac{1}{\rho}\frac{\partial p}{\partial x} = u_\infty \frac{du_\infty}{dx} . \tag{1-3}$$

Consider cyclic iteration number n. The dependent variables are calculated using the pressure gradient obtained in iteration (n-1).

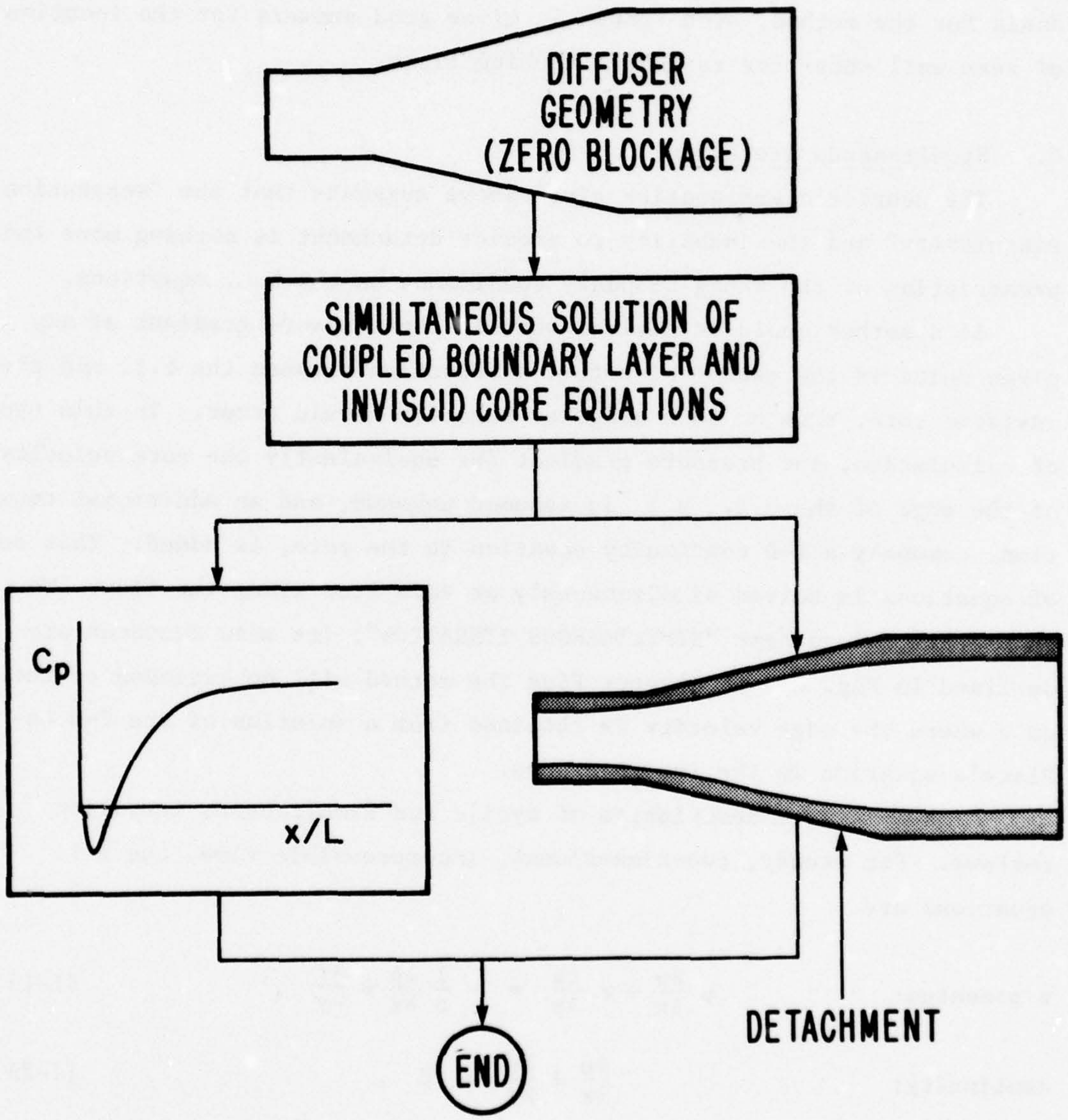


Fig. E. Flowchart of simultaneous iteration as applied to the calculation of diffusers.

$$\left[u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y}\right]^{(n)} = \left[u_\infty \frac{du_\infty}{dx}\right]^{(n-1)} + \left[\frac{\partial \tau}{\partial y}\right]^{(n)} \qquad (1\text{-}4)$$

If the pressure gradient is approximately the same for both iterations, such as for attached flows, then this set of equations gives a good solution. If, however, dp/dx varies greatly between iterations, i.e.,

$$[p_x]^{(n)} \neq [p_x]^{(n-1)} ,$$

then either one obtains the solution to the wrong problem, or the equations diverge, giving no solution at all.

In simultaneous iteration, the pressure gradient is replaced by that at the current iteration, so that all quantities are now at step n. That is, Eqn. (1-4) becomes

$$\left[u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y}\right]^{(n)} = \left[u_\infty \frac{du_\infty}{dx} + \frac{\partial \tau}{\partial y}\right]^{(n)} . \qquad (1\text{-}5)$$

$[u_\infty \, du_\infty/dx]^{(n)}$ is now unknown and must be supplied from an additional relationship involving the potential core. In Chapter Two a one-dimensional core equation is used, while in Chapter Five the method will be extended to include a two-dimensional core. In either case, all quantities are expressed in terms of values at step n. In effect, we have converted from an explicit to an implicit set of equations, with a corresponding gain in numerical stability.

A clear explanation of simultaneous iteration as applied to a dissipation integral type b.l. calculation can be found in Gerhart [50].

D. Previous Work

Moses [7] used the simultaneous iteration concept to calculate the flow in incompressible diffusers with detached b.l.'s. Even though he used a power-law velocity profile and rather simple prediction schemes, he was able to obtain fair agreement with data for diffusers operating in the early portions of the transitory stall regime. The most significant contribution of his work was to recognize the need for including a

1-D core equation, thereby enabling him to avoid the singular behavior of the equations near detachment.

Moses was much criticized for having the audacity to attempt the calculation of flows at and beyond detachment [8]. Unfortunately, some of the correlations used by him were questionable in the light of the then available data. The discussion of his work in the literature quickly became bogged down in arguments over the validity of details of these correlations, and the central idea, that of simultaneous iteration, was largely forgotten.

The next few years saw tremendous activity in the development of prediction methods for turbulent b.l.'s, as exemplified by the 1968 Stanford Conference [3] on Computation of Turbulent Boundary Layers. In all, 28 methods for the calculation of TBL's with prescribed pressure gradient were presented. These varied from simple correlative integral methods to rather complicated differential methods using sophisticated turbulence closure schemes. None of the methods presented was able to calculate separating flows near detachment adequately. This is not surprising, since they were all calculated with prescribed pressure gradient without taking the freestream interaction into account.

It is apparent from the discussions that some predictors were acutely aware of the need for including this interaction for separating flows. Nevertheless, the majority of attendees bypassed this in favor of discussions involving the validity of the b.l. equations, the contributions from normal stresses, curvature effects, etc.

The controversy regarding the inability of the b.l. equations with prescribed pressure gradient to predict detaching flows is still raging. As late as 1975, attendees at the AGARD Separating Flow Conference [9] were still debating the same questions as at the '68 Stanford Conference. The idea of simultaneous iteration being the key to removing the singular behavior near detachment is still far from being universally accepted.

In 1972, Bower [10] extended Moses' calculation to include compressible flow in axisymmetric diffusers. He retained the 1-D core assumption and used a dissipation integral b.l. method. The dissipation integral, C_D was related to the shape factor H through an empirical correlation due to Alber [11]. The energy shape factor, $\overline{H}$, w... elated to H through the Escudier-Nicoll correlation [3],

$$\overline{H} = 1.431 - .0971/H + .775/H^2 \quad . \tag{1-6}$$

A limiting form of Coles' velocity profile was used, with $Re_\delta \to \infty$, giving a one-parameter family.

$$\frac{u}{u_\infty} = \left(\frac{3-H}{2H}\right)\left[1 + \ell n\left(\frac{y}{\delta}\right)\Big/\ell n\ (.565\ Re_\delta)\right] + \frac{3}{4}\left(\frac{H-1}{H}\right)\left(1 - \cos\pi\frac{y}{\delta}\right) \tag{1-7}$$

Skin friction, $C_f/2$, was obtained through the Ludweig-Tillmann correlation,

$$\frac{C_f}{2} = 0.123\ Re_\theta^{-.268}\ 10^{-.678\ H} \quad . \tag{1-8}$$

Bower's predictions for the diffusers operating in the early portions of the transitory stall regime are quite good. Nevertheless, his calculation method can be criticized on several grounds.

The one-parameter velocity profile, Eqn. (1-7), is a poor representation of actual b.l.'s in adverse pressure gradients, even though it does permit backflow. The empirical $\overline{H}$ vs. H relationship, Eqn. (1-6), is valid only for $1.25 \leq H \leq 2.8$ (Ref. [3], pp. 136-138), but is used in this method for H up to 12.0. The Ludweig-Tillman correlation, Eqn. (1-8), is always positive, so that zero or negative wall shear values cannot be represented, no matter how large the values of Re_θ and H. As a result, the location of zero wall shear cannot be determined, and the rather arbitrary value of $H = 1.8$ was used as an indication of detachment.

However, it is well known that detachment is not a unique function of H, being in fact a stronger function of the blockage, δ^*. This is apparent in the work of Sandborn and Kline [51], who postulate the beginning of intermittent detachment at a point where $H = H_{sep}$, where

$$H_{sep} = 1 + \frac{1}{1-\delta^*/\delta} \quad . \tag{1-9}$$

Also, Senoo and Nishi [13] obtained an empirical stall limit relation for diffusers,

$$H_s = 1.8 + 3.75\,B \quad , \tag{1-10}$$

where B is the local value of the blockage factor, $2\delta^*/W$.

The use of any empirical correlation to determine detachment is clearly undesirable, since it limits the probable generality of the procedure and is hence to be avoided if possible.

In view of these residual difficulties in the work of Bower, the relatively good results achieved strongly support (but do not definitely prove) the idea that the central difficulties in predicting detachment and separated flows can be cured or strongly alleviated by simultaneous iteration. To put this differently, as already found by Woolley [4] and White [5] for fully stalled flows, the crucial matter is to get the interaction between the blockage effects of the separated zone and the outer flow modeled adequately; all other effects are less important to adequate predictions. What is suggested here, then, is that the same is true of detachment and detaching flows, and that for such cases simultaneous rather than cyclic iteration is necessary. It is this idea plus the specific details needed to alleviate the problems relative to Bower's work that are central to the work that follows.

CHAPTER TWO

UNIFIED INTEGRAL METHOD

A. Requirements for Boundary Layer Prediction Method

The survey of currently available prediction methods for turbulent b.l.'s presented in the last chapter showed that calculation methods for attached b.l.'s are highly developed. The converse is true for detached flows and b.l.'s that are in the process of detaching, both of which must be calculated in simultaneous fashion with the bounding freestream. It was felt that a new calculation method was needed to be able to extend the diffuser calculations deeper into the transitory stall regime. The requirements for such a method are that:

(a) The equations simultaneously solve for the boundary layer and the freestream.

(b) The velocity profile family be capable of representing both attached and detached flows.

(c) The auxiliary equation, turbulent shear stress model and its associated correlations be valid for attached and detached flows.

(d) The set of equations should not introduce any singularities at the detachment point.

(e) Detachment should occur "naturally" and be perceptible as having occurred without recourse to any empiricism such as the frozen dp/dx method or a detachment criterion. That is, the desirable detachment criterion is $C_f = 0$ (on the average).

(f) The method should be fast, since we expect to use it in an iterative fashion.

(g) The core velocity should be obtained from a solution of the elliptic irrotational core, so as to include downstream effects on the upstream flow.

The rest of this chapter develops such a method, the "Unified Integral Method " (UIM), with a 1-D core. The extension to the 2-D case is deferred to Chapter Five.

B. Integral Methods in General

A brief summary of integral b.l. methods will be presented before proceeding with the development of the UIM equations.

All integral methods use the von Kármán momentum integral equation,

$$\frac{d\theta}{dx} + (2+H)\frac{\theta}{u_\infty}\frac{du_\infty}{dx} = \frac{C_f}{2} + \frac{1}{u_\infty^2}\int_o^\delta \frac{\partial}{\partial x}\left(\overline{u'^2} - \overline{v'^2}\right)dy \quad , \qquad (2\text{-}1)$$

The normal stress term is usually neglected, although there is some evidence that its value may be large near detachment. This equation may be parametrically expressed as

$$\frac{d\theta}{dx} = f_1(H, \theta, u_\infty, C_f/2) \quad . \qquad (2\text{-}2)$$

Consider the case of a prescribed pressure gradient calculation where u_∞ is a known function of the streamwise coordinate x. Two more equations are needed to solve for the three unknowns θ, H, and $C_f/2$. The differences in the various methods arise in the procedure used to close the set of equations.

Many methods use an empirical equation relating the skin friction $C_f/2$ to the calculation variables. The most commonly used is the Ludweig-Tillmann correlation,

$$\frac{C_f}{2} = 0.123\ Re_\theta^{-.128}\ 10^{-.678\ H} \quad . \qquad (2\text{-}3)$$

The last equation remaining is called the "auxiliary" equation and relates the growth of the shape factor H to the other b.l. parameters. One method of obtaining this equation is by taking moments in u or y of the two-dimensional b.l. equations before integrating across the layer. The first moment in u gives the "mechanical energy" equation,

$$\theta\frac{d\overline{H}}{dx} = (H-1)\frac{\overline{H}\theta}{u_\infty}\frac{du_\infty}{dx} - \overline{H}\frac{C_f}{2} + C_D \quad , \qquad (2\text{-}4)$$

where

$$C_D = \frac{2}{\rho u_\infty^3}\int_o^\delta \tau\frac{\partial u}{\partial y}dy \quad . \qquad (2\text{-}5)$$

The first moment in y gives the "moment of momentum" equation,

$$\int_o^\delta \left[y \frac{\partial u^2}{\partial x} - y \frac{\partial}{\partial y} \left(u \int_o^y \frac{\partial u}{\partial x} dy \right) \right] dy = \frac{\delta^2}{2} u_\infty \frac{du_\infty}{dx} - C_\tau \qquad (2\text{-}6)$$

where

$$C_\tau = \int_o^\delta \frac{\tau}{\rho} dy \quad . \qquad (2\text{-}7)$$

Additional unknowns $\overline{H}$, C_D, C_τ have appeared in both auxiliary moment equations (2-4) and (2-6), and these must be related back to the primary variables H, θ, and $C_f/2$. At this stage a model equation for the turbulent shear stresses and a velocity profile family must be introduced. The turbulence model relates C_D or C_τ to the mean flow parameters, while the velocity profile family allows $\overline{H}$ to be expressed in terms of H for Eqn. (2-4) and permits Eqn. (2-6) to be integrated. For details of this process, see the review papers by Reynolds [3], Rotta [14], and the introductory sections of Hirst and Reynolds [15].

Head [16] used the growth rate of the turbulent-nonturbulent front to derive an auxiliary equation. The rate at which the b.l. spreads into the irrotational fluid is the entrainment rate dQ/dx and may be expressed as a function of a new shape factor $H_{\delta-\delta^*}$,

$$Q = \int_o^\delta u \, dy = u_\infty(\delta - \delta^*) \quad , \qquad (2\text{-}8)$$

$$\frac{dQ}{dx} = F_1(H_{\delta-\delta^*}, u_\infty, \delta-\delta^*) \quad , \qquad (2\text{-}9)$$

where

$$H_{\delta-\delta^*} = \frac{\delta-\delta^*}{\theta} \quad . \qquad (2\text{-}10)$$

$H_{\delta-\delta^*}$ is in turn related back to H through another correlation, closing this set of equations.

A survey of the literature will show the large variety of auxiliary equations that have been used. This is a consequence of the fact that no "exact" independent equation is available. It is therefore important to understand exactly what the auxiliary equation provides in the way of new information.

We note that there is no term involving the turbulent shear stresses in the momentum integral equation (2-1). Therefore, the most important function of the auxiliary equation is to supply information regarding the shear stresses in the b.l. The second requirement is that it truly contain independent information. For instance, Hirst et al. [15] found that the mechanical energy equation may not be completely independent of the momentum integral equation. This is so since u is fairly constant across the layer, and the resulting set of equations is almost redundant.

Studies by Hirst et al. [15] and Thompson [35] showed that the entrainment method of Head appeared to work better than other available methods for a large variety of b.l.'s. They hypothesized that perhaps this technique contained "more" independent information regarding the turbulence. We shall therefore use the entrainment concept, but extend its applicability to enable calculation of detached flows.

To summarize, the auxiliary equation is of the form

$$\frac{dH}{dx} = f_2(H, \theta, u_\infty, C_f/2, \phi(\tau)) \quad . \tag{2-11}$$

The turbulence model equation is of the form

$$\phi(\tau) = f_3(H, \theta, u_\infty, C_f/2) \quad . \tag{2-12}$$

$\phi(\tau)$ is a functional representation of shear stress integrals such as C_D or C_τ in Eqns. (2-4) and (2-6). The closure model for $\phi(\tau)$ relates it back to known quantities such as H, θ, u_∞, and $C_f/2$, as shown through Eqn. (2-12).

The skin friction equation is obtained from a correlation of the form

$$C_f/2 = f_4(\theta, H, u_\infty) \quad . \tag{2-13}$$

Equations (2-2) and (2-11) through (2-13) permit the b.l. parameters to be calculated in a stepwise marching fashion along the flow.

As mentioned before, all of the above methods work quite well for accelerating and flat-plate flows, and reasonably well for decelerating flows which are far from detachment. Neither the velocity profile family nor the auxiliary equations are valid at or beyond detachment. We proceed therefore to tailor the UIM to be able to do this by first examining a velocity profile family that is capable of representing both attached and detached flows, and then developing an auxiliary equation that works over this entire range.

C. Velocity Profile Family

It is generally accepted that typical TBL velocity profiles can be represented by the combination of an inner-wall-dominated layer plus an outer "wake-like" structure. One such velocity profile family is the log law of the wall matched to Coles' [17] "law of the wake" outer profile,

$$\frac{u}{u_\tau} = \frac{1}{\kappa} \ell n \left(\frac{yu_\tau}{\nu}\right) + c + \frac{\Pi}{\kappa} \left(1 - \cos \pi \frac{y}{\delta}\right) , \qquad (2\text{-}14)$$

where

κ is the von Karman constant = 0.41,

c is the wall constant = 5.0,

u_τ is the shear velocity = $\sqrt{\tau_\omega/\rho}$,

Π is the wake amplitude.

This profile gives excellent results for attached flows, but cannot be used without modification for either detached or detaching flows. The difficulty in representing detaching flows may be seen by taking the limit as $u_\tau \to 0$ after setting $u = u_\infty$ at $y = \delta$.

$$u_\infty = \lim_{u_\tau \to 0} \left[\frac{u_\tau}{\kappa} \ell n \left(\frac{yu_\tau}{\nu}\right) + cu_\tau + \frac{2\Pi u_\tau}{\kappa}\right] ,$$

i.e.,

$$u_\infty = \frac{2}{\kappa} \lim_{u_\tau \to 0} (\Pi u_\tau) .$$

So $\Pi \to \infty$ as $u_\tau \to 0$ in order to keep the limit finite.

Beyond detachment, the wall shear τ_ω is negative and u_τ is not even defined.

However, a simple modification of (2-14) together with redefinition of u_τ for reversed flows permit the desired representation.

Let

$$u_\tau \triangleq (\operatorname{sgn} \tau_\omega)\sqrt{\frac{|\tau_\omega|}{\rho}}$$

and

$$y^+ \triangleq \frac{y|u_\tau|}{\nu} .$$

The modified velocity profile family is

$$u = \frac{u_\tau}{\kappa}\left[\ell n \frac{y|u_\tau|}{\nu} + \hat{c}\right] + \frac{u_\beta}{2}\left(1 - \cos\frac{\pi y}{\delta}\right) , \qquad (2\text{-}15)$$

where u_β is the redefined wake amplitude, and $\hat{c}$ is a new constant, $\hat{c} = c\kappa = 2.05$.

Define

$$v_T \triangleq \frac{u_\tau}{\kappa u_\infty} ,$$

$$v_B \triangleq \frac{u_\beta}{u_\infty} , \qquad (2\text{-}16)$$

$$\eta \triangleq \frac{y}{\delta} .$$

At the edge of the b.l., $\eta = 1$, and $u = u_\infty$.

$$\therefore \quad u_\infty = \frac{u_\tau}{\kappa}\left[\ell n\left(\frac{\delta|u_\tau|}{\nu}\right) + \hat{c}\right] + u_\beta . \qquad (2\text{-}17)$$

Subtracting Eqn. (2-17) from (2-15) and substituting from Eqn. (2-16), we get the desired profile,

$$\frac{u}{u_\infty} = \underset{(a)}{1} + \underbrace{V_T \ln \eta}_{(b)} - \underbrace{V_B \cos^2 \frac{\pi\eta}{2}}_{(c)} \quad . \tag{2-18}$$

Fig. F shows the contribution of each of the terms in this equation to the velocity profile.

We note from Eqn. (2-18) and the above sketch that u_β is normally positive and that $u_\beta < u_\infty$ for attached flows while $u_\beta > u_\infty$ for detached flows. This form of the velocity profile has been used by McDonald and Stoddard [18] and Nash and Hicks [3] for attached flows. Kuhn and Nielsen [19] attempted to calculate detached flows using this profile. The ability of Eqn. (2-15) to represent attached and detached flows is shown in Figures 2 and 3.

Alber et al. [20] extended the applicability of this profile to represent compressible flows, and concluded that the Coles type formulation is an adequate representation of the velocity field both upstream and downstream of detachment. The measured detachment profile does not, however, correspond to Coles' zero wall-friction profile. Over most of the flow, however, a fair to good fit with experimental data was obtained. It should therefore be adequate for use in an integral prediction method.

Using the velocity profile, Eqn. (2-18), it is possible to develop relationships among the b.l. integral parameters, δ^*, θ, and V_T, V_B. Substituting Eqn. (2-18) into the definitions of δ^* and θ and on integrating across the b.l., we get

$$\frac{\delta^*}{\delta} = V_T + \frac{V_B}{2} \quad , \tag{2-19}$$

$$\frac{\delta^* - \theta}{\delta} = 2V_T^2 + \frac{3}{8} V_B^2 + 1.58949\, V_T V_B \quad . \tag{2-20}$$

These two equations give an unambiguous definition of δ.

D. Momentum Integral Equation

The momentum integral equation for steady, 2-D, incompressible flow is

$$\frac{d\theta}{dx} + (2+H)\frac{\theta}{u_\infty}\frac{du_\infty}{dx} = \frac{C_f}{2} + \frac{1}{u_\infty^2}\int_0^\delta \frac{\partial}{\partial x}\left(\overline{u'^2} - \overline{v'^2}\right) dy \quad . \tag{2-21}$$

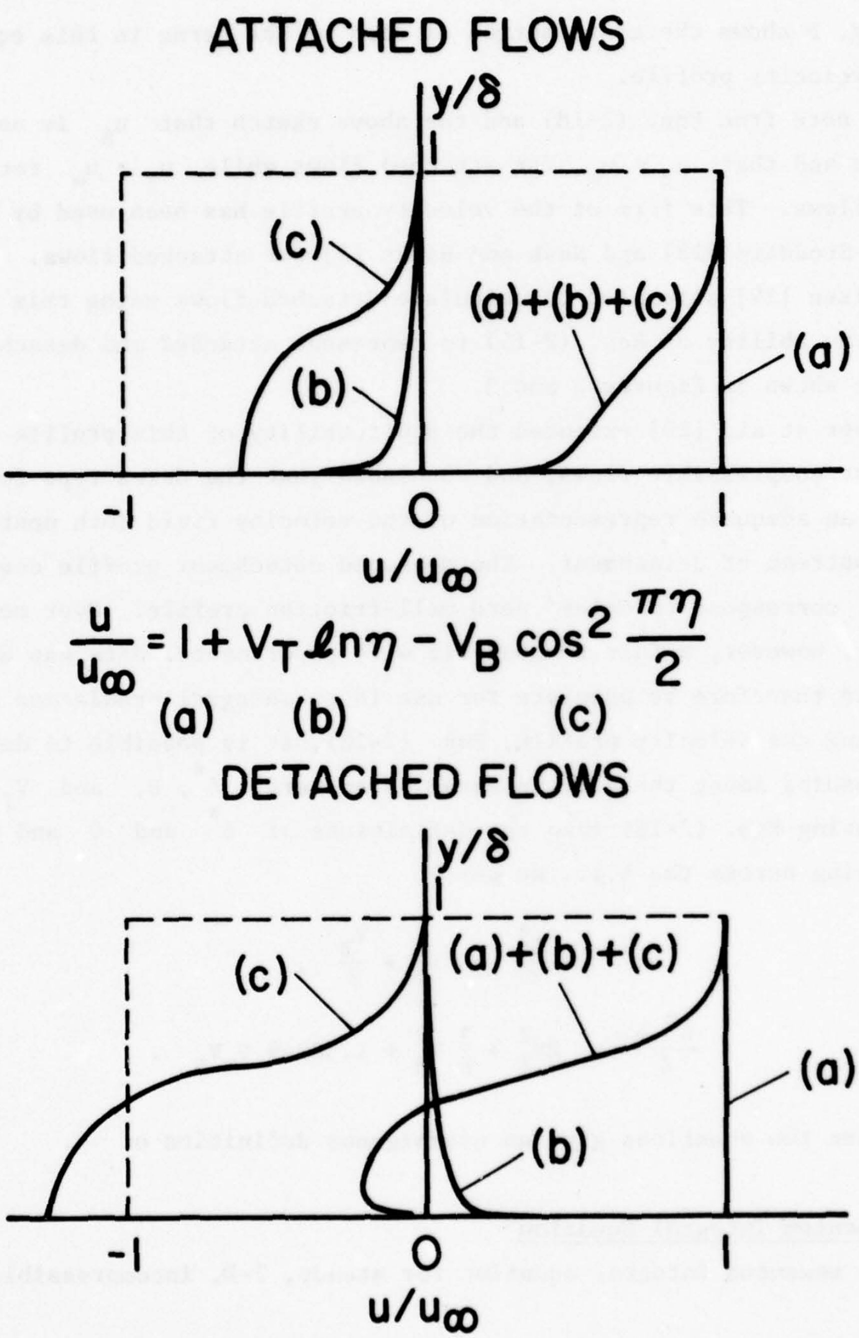


Fig. F. Components of the velocity profile for attached and detached flows.

This may be rewritten in terms of the proposed dependent variables δ, u_β, u_τ, and U_∞ by defining

$$V_N \triangleq \frac{1}{U_\infty^2}\int_o^\delta \frac{\partial}{\partial x}\left(\overline{u'^2}-\overline{v'^2}\right) dy \tag{2-22}$$

and noting that

$$\frac{C_f}{2} = \left(\frac{u_\tau}{U_\infty}\right)^2 . \tag{2-23}$$

Differentiating (2-19) with respect to x,

$$\frac{d\delta^*}{dx} = \frac{\delta^*}{\delta}\frac{d\delta}{dx} + \frac{\delta}{\kappa U_\infty}\frac{du_\tau}{dx} - \frac{\delta}{U_\infty}\left(\frac{u_\tau}{\kappa U_\infty} + \frac{u_\beta}{2U_\infty}\right)\frac{dU_\infty}{dx} + \frac{\delta}{2U_\infty}\frac{du_\beta}{dx} . \tag{2-24}$$

Similarly, differentiating (2-20) and rearranging gives

$$\begin{aligned}\frac{d\theta}{dx} = {} & \left(\frac{\delta^*}{\delta} - 2V_T^2 - \frac{3}{8}V_B^2 - 1.58949\, V_T V_B\right)\frac{d\delta}{dx} - \left(\frac{4\delta}{\kappa}V_T^2 + \frac{1.58949\, V_B\delta}{\kappa U_\infty} - \frac{\delta}{\kappa U_\infty}\right)\frac{du_\tau}{dx} \\ & + \left(\frac{4V_T^2\delta}{U_\infty} + \frac{3}{4}V_B^2\frac{\delta}{U_\infty} + \frac{3.17898}{\kappa U_\infty^3}u_\beta u_\tau\delta - \frac{\delta V_T}{U_\infty} - \frac{\delta V_B}{2U_\infty}\right)\frac{dU_\infty}{dx} \\ & + \left(\frac{\delta}{2U_\infty} - \frac{3}{4}\frac{\delta u_\beta}{U_\infty^2} - \frac{1.58949}{\kappa U_\infty^2}u_\tau\delta\right)\frac{du_\beta}{dx} .\end{aligned} \tag{2-25}$$

Substituting (2-22) through (2-25) into (2-21), rearranging, and using (2-19) and (2-20) gives

$$\begin{aligned}&\left(\frac{\theta}{\delta}\right)\left\{\frac{d\delta}{dx}\right\} + \left(\frac{\delta}{U_\infty}\right)\left(\frac{1}{2} - \frac{3}{4}V_B - 1.58949\, V_T\right)\left\{\frac{du_\beta}{dx}\right\} \\ &+ \left(\frac{\delta}{\kappa U_\infty}\right)\left(1 - 4V_T - 1.58949\, V_B\right)\left\{\frac{du_\tau}{dx}\right\} + \left(\frac{2\delta^*}{U_\infty}\right)\left\{\frac{dU_\infty}{dx}\right\} = \kappa^2 V_T^2 + V_N .\end{aligned} \tag{2-26}$$

Equation (2-26) is the form of the momentum integral equation used in the computations. The normal stress term, V_N, has been carried along for completeness and is not used further in this investigation.

E. Outer-Edge Matching Equation

Differentiating Eqn. (2-17) in the streamwise direction and rearranging gives

$$\frac{dU_\infty}{dx} = \frac{1}{\kappa}\left(\ln\frac{\delta|u_\tau|}{\nu} + \hat{c}\right)\left\{\frac{du_\tau}{dx}\right\} + \frac{(\operatorname{sgn} u_\tau)}{\kappa\delta}\left(\delta\frac{d|u_\tau|}{dx} + |u_\tau|\frac{d\delta}{dx}\right) + \left\{\frac{du_\beta}{dx}\right\} . \tag{2-27}$$

Now,

$$(\operatorname{sgn} u_\tau)|u_\tau| = u_\tau , \tag{2-28}$$

and

$$(\operatorname{sgn} u_\tau)\frac{d|u_\tau|}{dx} = \frac{du_\tau}{dx} . \tag{2-29}$$

Equation (2-29) is valid everywhere except in the case where u_τ changes sign. There is then an ambiguity in the sign of the resulting value of u_τ, which may be resolved by using physical insight from the velocity profile. When $u_\beta > u_\infty$, then $u_\tau < 0$ and vice-versa.

Therefore,

$$u_\tau = |u_\tau| \operatorname{sgn}(u_\beta - U_\infty) . \tag{2-30}$$

Rearranging Eqn. (2-17), we get

$$\frac{1}{\kappa}\left(\ln\frac{\delta|u_\tau|}{\nu} + \hat{c}\right) = \frac{U_\infty - u_\beta}{u_\tau} . \tag{2-31}$$

Substituting (2-28) through (2-31) in (2-27) gives

$$\left(\frac{u_\tau^2}{\kappa\delta}\right)\left\{\frac{d\delta}{dx}\right\} + (u_\tau)\left\{\frac{du_\beta}{dx}\right\} + \left(\frac{u_\tau}{\kappa} + U_\infty - u_\beta\right)\left\{\frac{du_\tau}{dx}\right\} + (-u_\tau)\left\{\frac{dU_\infty}{dx}\right\} = 0 . \tag{2-32}$$

F. Entrainment Equation

The concept of entrainment will be used to derive the auxiliary equation. The calculation method of Bradshaw et al. [21] uses a correlation between the nondimensional entrainment rate

$$\frac{1}{U_\infty}\frac{d}{dx}\left[U_\infty(\delta-\delta^*)\right]$$

and the maximum shear stress in the b.l., $\tau_{max}/\rho U_\infty^2$. This correlation has been revised in Fig. 4 to include data from recent experiments, and shows that the entrainment rate is about ten times the maximum shear stress. That is,

$$\frac{1}{U_\infty}\frac{d}{dx}\left[U_\infty(\delta-\delta^*)\right] = 10\ \tau_{max}/\rho U_\infty^2 \ . \tag{2-33}$$

The remarkable feature of this correlation is that it seems to apply to both attached and detached flows. It works equally well for b.l.'s in favorable or adverse pressure gradients, provided that for accelerating flows the maximum shear stress is evaluated at $\eta = y/\delta = 1/4$, despite the fact that the maximum shear stress for an accelerating b.l. occurs at the wall.

Differentiating Eqn. (2-33) in the x direction and substituting for $d\delta^*/dx$ from (2-24) gives

$$\left(1-\frac{\delta^*}{\delta}\right)\left\{\frac{d\delta}{dx}\right\} + \left(\frac{-\delta}{2U_\infty}\right)\left\{\frac{du_\beta}{dx}\right\} + \left(\frac{-\delta}{\kappa U_\infty}\right)\left\{\frac{du_\tau}{dx}\right\} + \left(\frac{\delta}{U_\infty}\right)\left\{\frac{dU_\infty}{dx}\right\} = 10\ \tau_{max}/\rho U_\infty^2 \ . \tag{2-34}$$

We require τ_{max}/ρ to be able to use (2-34). The distance from the wall y/δ at which the shear stress is maximum will be obtained from another correlation. The velocity profile can be differentiated and evaluated at this y/δ location to give the value of $\partial u/\partial y$ corresponding to maximum τ. It is then possible to compute τ_{max}/ρ by using an eddy-viscosity model.

A plot of $(u/U_\infty)_{\tau_{max}}$ at which the maximum shear stress occurs (Fig. G) as a function of the ratio of the wall to wake velocities, $2u_\tau/\kappa u_\beta$, is shown in Fig. 5. There is a fair amount of scatter and a clearer picture emerges when only equilibrium b.l.'s and detached flows are plotted (Fig. 6). It is apparent that the velocity ratio at which τ_{max} occurs, denoted by $\left(\frac{u}{U_\infty}\right)_{\tau_{max}}$, may be quite well represented by

equilibrium b.l.'s: $$\left(\frac{u}{U_\infty}\right)_{\tau_{max}} = 0.76 \ ,$$

detached flows: $$\left(\frac{u}{U_\infty}\right)_{\tau_{max}} = 0.60 \ .$$

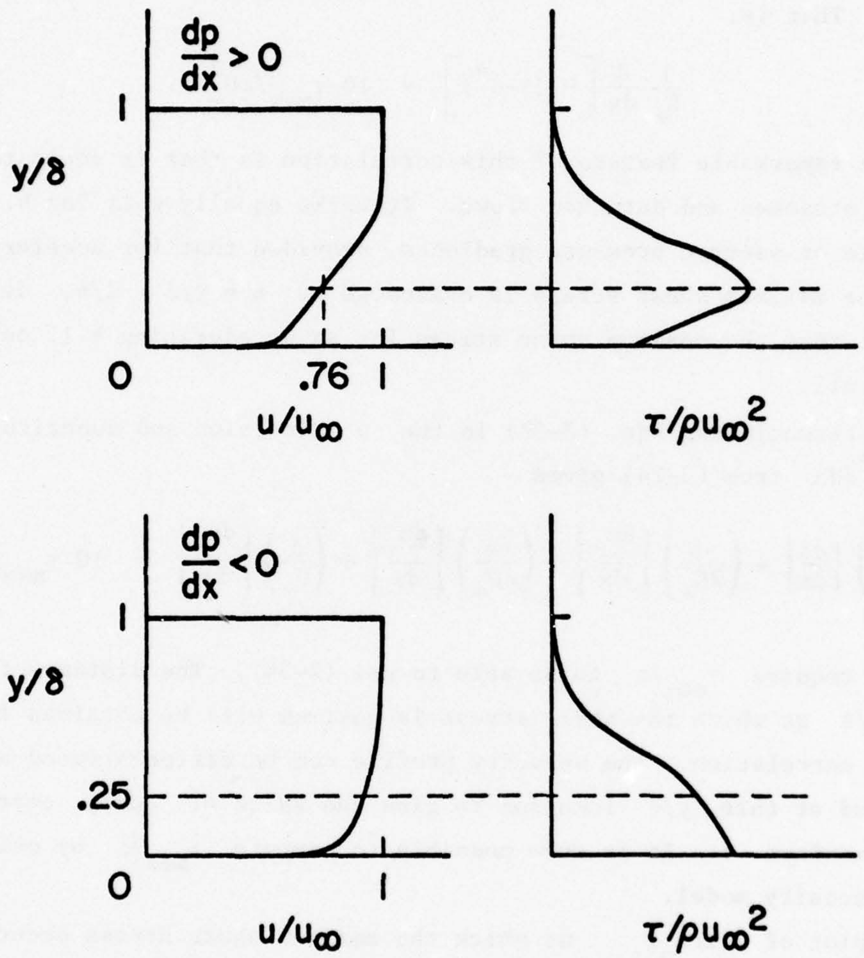


Fig. G. Location of maximum shear for adverse and favorable pressure gradients.

Knowing V_T, V_B, and $(u/U_\infty)_{\tau_{max}}$, Eqn. (2-18) may be used in a straightforward Newton-Raphson scheme to obtain $\eta_{\tau_{max}} = (y/\delta)_{\tau_{max}}$ at which τ_{max} occurs, by solving

$$f\left(\eta_{\tau_{max}}\right) = V_T \ \ln \eta_{\tau_{max}} - V_B \cos^2 \frac{\pi}{2} \eta_{\tau_{max}} + 1 - \left(\frac{u}{U_\infty}\right)_{\tau_{max}} = 0 \ . \tag{2-35}$$

This gives $\eta_{\tau_{max}}$. Differentiating Eqn. (2-18), we get

$$\left(\frac{\partial u}{\partial y}\right)_{\tau_{max}} = \frac{U_\infty}{\delta}\left(\frac{V_T}{\eta_{\tau_{max}}} + \frac{V_B \pi}{2} \sin \pi\eta_{\tau_{max}}\right) \ . \tag{2-36}$$

On substituting $\eta_{\tau_{max}}$ we obtain $(\partial u/\partial y)_{\tau_{max}}$. This may now be used through an eddy viscosity formulation to obtain the maximum shear stress τ_{max}; i.e., we assume

$$\frac{\tau}{\rho} = \varepsilon \frac{\partial u}{\partial y} \ , \tag{2-37}$$

where ε is an eddy viscosity.

For the outer portion of equilibrium b.l.'s $(y/\delta > 0.2)$, Clauser [22] showed that ε may be approximately represented by

$$\varepsilon = K_e U_\infty \delta^* \ , \tag{2-38}$$

where $K_e = .0168$.

For nonequilibrium b.l.'s, the relationship is no longer this simple, and many efforts have been made to obtain a universal formulation for ε. McD. Galbraith and Head [23] present an extensive summary of many of these attempts and compare the results with experiments.

Kuhn and Nielsen [19] included the effect of pressure gradient and intermittency and obtained

$$\frac{\tau}{\rho} = K_e \gamma U_\infty \delta^* \left(\frac{\partial u}{\partial y}\right) \ , \tag{2-39}$$

where

$$K_e = .013 + .0038 \ e^{-\beta/15}$$

$$\beta = \frac{\delta^*}{\tau_\omega} \frac{dp}{dx}$$

and intermittency $\gamma = (1+9\eta^6)^{-1}$.

The pressure gradient parameter β is small for mild pressure gradients and flows far from detachment. This allows K_e to approach Clauser's value of .0168 for attached equilibrium b.l.'s and the limiting value of .013 for free shear flows such as the flow at a free jet boundary (Schlichting [2], pp. 681-707) for which $\beta \to \infty$. For accelerating flows, β is set equal to zero.

The maximum shear stress in an equilibrium b.l. can thus be obtained from

$$\left(\frac{\tau}{\rho}\right)_{max,eq} = (.013 + .0038\ e^{-\beta/15})(1 + 9\eta^6)^{-1}\ U_\infty \delta^* \left(\frac{\partial u}{\partial y}\right)_{\tau_{max}} . \tag{2-40}$$

For a nonequilibrium b.l., this expression has to be modified to take into account upstream history effects. The fluid near the wall in a TBL is in local equilibrium in the sense that it adjusts very rapidly to external changes, such as the pressure gradient. The outer layers, however, are dominated by large eddies that have considerable inertia, so that it has finite adjustment time. The outer layer therefore "lags" behind the local pressure gradient. Typical behavior of the velocity profile in response to sudden removal of the pressure gradient is shown schematically in Fig. H, adapted from White [25].

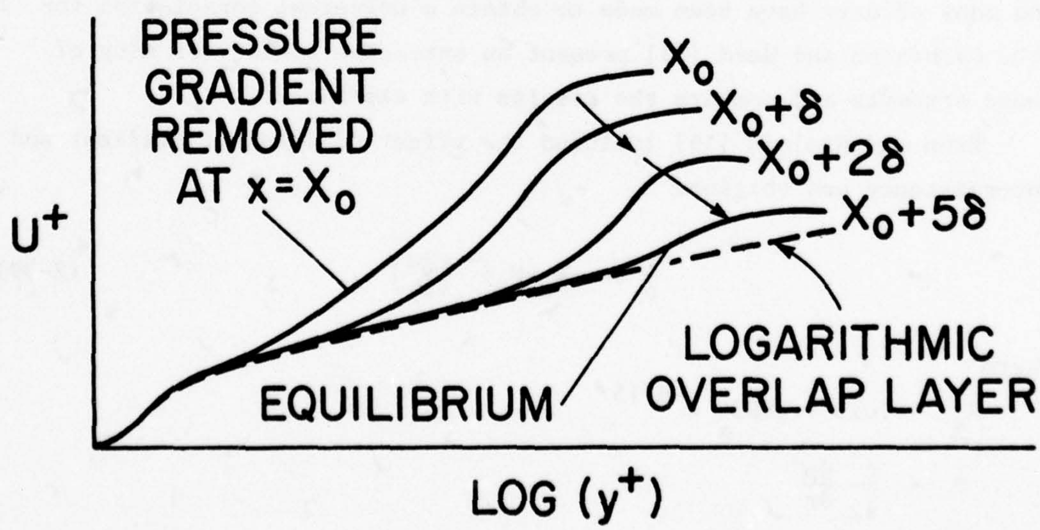


Fig. H. Relaxation following removal of pressure gradient.

For the flow shown in Fig. H, the velocity profile takes 5δ after removal of the pressure gradient before it reaches equilibrium. Another example of shear stress lag may be seen in the relaxing flow of Goldberg [26], as presented by McDonald et al. [18]. Fig. I shows the lag between the measured shear stress integral C_τ and its equilibrium value $\hat{C}_\tau$.

The calculation of shear stresses from an equilibrium condition will therefore give erroneous results.

One method of accounting for the departure from equilibrium is to relate the equilibrium and nonequilibrium values through a first-order differential equation, commonly called a lag equation.

$$\frac{d}{dx}\left(\frac{\tau_{max}}{\rho}\right) = \frac{\lambda}{\delta}\left(\frac{\tau_{max,eq}}{\rho} - \frac{\tau_{max}}{\rho}\right) . \qquad (2\text{-}41)$$

The lag parameter λ is obtained from numerical experiments.

It should be noted that the lag equation does not model a primary term, but only corrects a deviation of what would otherwise be an error in a primary term. Since this deviation is usually small and only significant for rapid changes in the "environment" of the shear layer in the streamwise direction, the form of the lag equation used is not critical. Hence a simple first-order diffusion equation should be sufficient. That this is so is demonstrated by the results in the 1968 Conference [3].

A summary of the process used to obtain the right-hand side of the entrainment equation, which is now expressed entirely in terms of known quantities, is shown in Fig. 7. Fig. 8 compares τ_{max}/ρ measured by Strickland and Simpson [32] with that from Eqn. (2-40). The measured entrainment rate is also shown. The agreement is quite good for the attached flow, and the last few detached flow points, and fair to poor for the rest. Except for the last station, τ_{max}/ρ is overpredicted. This is consistent with our expectations, since these values were calculated from the measured mean velocity profile and correspond to the equilibrium case. Shear lag will decrease these computed values. The worst match is at station 124.3, the location of intermittent detachment, where the uncertainty in both the measured and calculated values is the greatest.

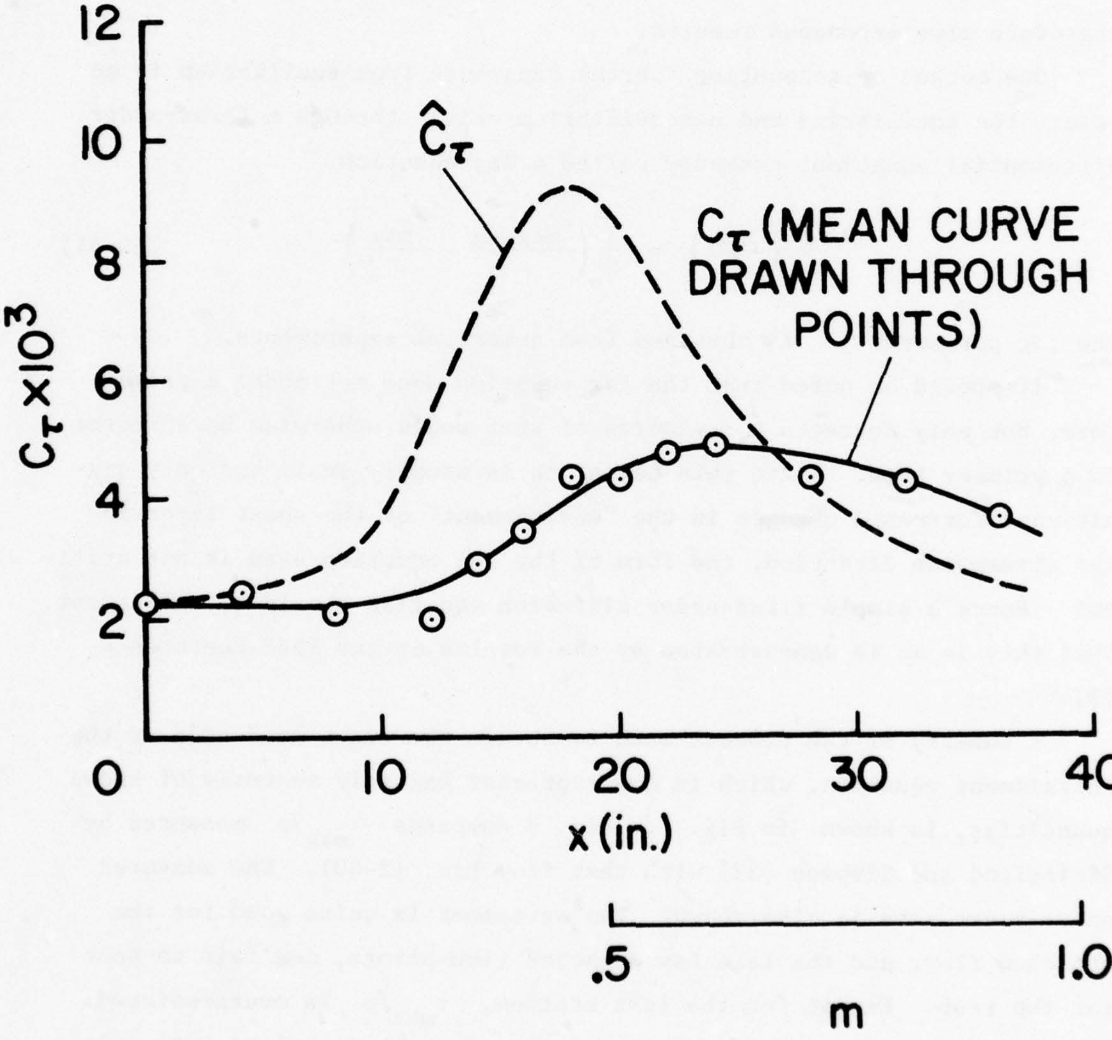


Fig. I. Behavior of equilibrium shear stress integral $\hat{C}_\tau$ and measured value C_τ for the relaxing flow of Goldberg [26].

To calculate a b.l. with prescribed pressure gradient, terms involving dU_∞/dx are moved to the right-hand side of Eqns. (2-26), (2-32), and (2-34), which may then be integrated in a stepwise fashion along the flow.

For detaching flows that must be calculated by simultaneous iteration, an additional equation is needed since U_∞ is now an unknown. In this chapter we shall use a 1-D continuity equation across the diffuser width.

G. One-Dimensional Core Equation

Consider flow in a diffuser of width $W(x)$ with a uniform 1-D velocity distribution in the core (Fig. J).

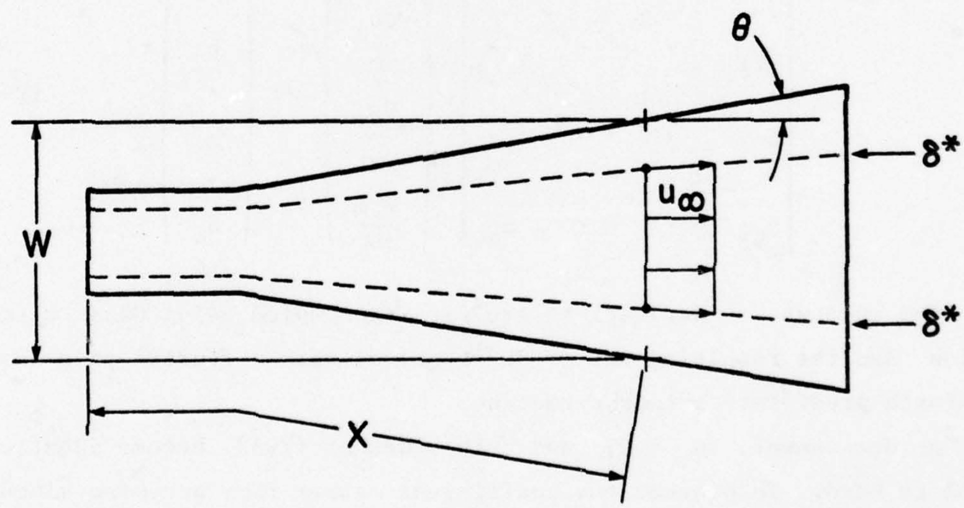


Fig. J. 1-D core diffuser nomenclature.

Assuming symmetrical b.l.'s and ignoring end-wall effects, the continuity equation at any section x is

$$Q = U_\infty(W - 2\delta^*/\cos\theta) \quad . \tag{2-42}$$

On differentiating (2-42) in the x direction, substituting for $d\delta^*/dx$ from (2-24), and manipulating, we get

$$\left(\frac{-\delta^*}{\delta}\right)\left\{\frac{d\delta}{dx}\right\} + \left(\frac{-\delta}{2U_\infty}\right)\left\{\frac{du_\beta}{dx}\right\} + \left(\frac{-\delta}{\kappa U_\infty}\right)\left\{\frac{du_\tau}{dx}\right\} + \left[\frac{\cos\theta}{2U_\infty}\left(W - \frac{2\delta^*}{\cos\theta}\right) + \frac{\delta^*}{U_\infty}\right]\left\{\frac{dU_\infty}{dx}\right\}$$

$$= -\frac{\cos\theta}{2}\frac{dW}{dx} . \qquad (2\text{-}43)$$

H. Solution of the UIM Equations

The addition of Eqn. (2-43) to (2-26), (2-32), and (2-34) closes the set. These may now be written as a 4 × 4 matrix equation at each step along the flow.

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & \cdot & \cdot & \cdot \\ a_{31} & \cdot & \cdot & \cdot \\ a_{41} & \cdot & \cdot & a_{44} \end{bmatrix} \begin{pmatrix} \frac{d\delta}{dx} \\ \frac{du_\beta}{dx} \\ \frac{du_\tau}{dx} \\ \frac{dU_\infty}{dx} \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ \cdot \\ \cdot \\ \cdot \\ b_4 \end{pmatrix} \qquad (2\text{-}44)$$

The unknown x derivatives are first uncoupled using Gauss elimination and the resulting set of ODE's solved with a fourth-order Adams-Bashforth predictor-corrector routine.

At detachment, $u_\tau = 0$, and both sides of (2-32) become identically equal to zero. To prevent the coefficient matrix from becoming singular, this equation is removed for $|u_\tau| < .025$ and the reduced set solved with the value of du_τ/dx frozen at the predetachment value. The results are negligibly affected by varying this threshold value of $|u_\tau|$ between .005 and .050.

In the next few chapters, the TBLPM developed herein will be used to predict diffusers after being calibrated by comparing its performance with that of a large number of currently available calculation methods.

CHAPTER THREE

RESULTS - BOUNDARY LAYER CALCULATIONS WITH PRESCRIBED PRESSURE GRADIENT

A. Determination of Lag Parameter

The lag parameter λ is still not known. The relaxation times for attached and detached flows are expected to be quite different, since the former is wall-dominated while the latter is inertia-bound. Two lag parameters, λ_a for attached and λ_d for detached flows were therefore used. It is expected that the "time constant" for detached flows will be very short compared to that for attached flows, since the effect of the wall is negligible and the large fluctuations present here tend to rapidly destroy any upstream history effects.

It must be noted that the existence of a lag between the equilibrium and actual shear stress values for detached flows is a hypothesis only, and that not enough data are available to rule out or confirm its existence. This is in sharp contrast to the corresponding case for attached flows, for which shear lag is a well-established phenomenon. The results of the present investigation are also ambiguous in this regard and do not conclusively support or rule out the necessity for using a lag equation for detached flows.

Both λ_a and λ_d were varied independently and the resulting predictions compared with data. The effect of varying λ_a for the relaxing flow of Bradshaw et al. [49] is shown in Figs. 9a and 9b. Fig. 20 shows the effect on the unstalled diffuser flow of Carlson and Johnston [27]. Varying λ_d while keeping λ_a constant is shown in Fig. 21 for a diffuser in transitory stall, also measured by the same experimenters. The results of not using any lag equation is shown in all the above cases.

For attached b.l.'s, varying λ_a from .015 to .035 has negligible effect on the flow. Not using a lag equation, however, does cause a small but discernible deviation from the data.

Similar effects are seen for detached flows. As λ_d is varied between 0.3 and 0.7, the exit velocity ratio u/u_{REF} changes from 0.68 to 0.64. This change is of the order of the experimental uncertainty. Not using any lag again causes a small but detectable overprediction of Cp.

We conclude that a lag equation is necessary for accurate predictions, although there is a large latitude in the choice of the numerical values for λ_a and λ_d. The numbers finally used were those giving the best match with a large number of flows, and are

$$\text{attached lag parameter:} \quad \lambda_a = 0.025 \quad ,$$

$$\text{detached lag parameter:} \quad \lambda_d = 0.70 \quad .$$

These results are consistent with the remarks concerning lag equations following Eqn. (2-41).

B. Experimental Data for Turbulent Boundary Layers

The best collection of TBL data is the extensive compilation of Coles and Hirst [44]. These data have been reduced in a consistent manner; moreover, the results of a large number of TBLPM's using prescribed pressure gradient to compute these data are presented in Kline et al. [3]. Any new calculation method must be able to predict satisfactorily all classes of flows in this reference before it can be accepted as a viable prediction tool. This is akin to a "calibration" technique for TBLPM's.

One way of classifying the available TBL data is in terms of the sign of the applied pressure gradient and whether or not the flow is in equilibrium.

Reference [3] has ranked the data according to the difficulty encountered by the 28 calculation methods in predicting the flows. In order of increasing difficulty, these were

(a) zero and mild favorable pressure gradients,

(b) strong favorable and mild adverse pressure gradients,

(c) separating, relaxing, and reattaching flows.

All the data were checked for two-dimensionality by normalizing the momentum integral equation and integrating in the x direction, giving

$$\underbrace{\frac{U_\infty^2\theta}{\left(U_\infty^2\theta\right)_o} - 1 + \frac{1}{2}\int_{x_o}^{x} \frac{\delta^*}{\theta_o}\, d\left(\frac{U_\infty^2}{U_{\infty_o}^2}\right)}_{PL} = \underbrace{\int_{x_o}^{x} \left(\frac{u_\tau}{U_{\infty_o}}\right)^2 d\left(\frac{x}{\theta_o}\right)}_{PR} . \qquad (3\text{-}1)$$

The subscript o indicates quantities at the start of the flow. The left and right sides of this equation were called PL and PR, respectively. If the measured values are exact, the b.l. two-dimensional, and the normal stress terms negligible, then PL = PR. Since all these conditions can never be met in practice, about all that can be said is that PL ≈ PR, and that strong departures from this equality suggest that some or all of these conditions are not met.

Interestingly, the degree of difficulty in predicting a flow is directly proportional to the imbalance between PL and PR. The obvious conclusion is that if the data do not satisfy the 2-D momentum integral equation, then a calculation method using this equation cannot predict the data!

C. Three-Dimensional Correction

Assuming that the imbalance in PL and PR is due to sidewall b.l. growth and that it can be modeled as a source or sink placed along a plane of symmetry, Schlichting [2] showed that the momentum integral equation can be balanced by including a crossflow term,

$$\frac{d\theta}{dx} + (2+H)\,\frac{\theta}{U_\infty}\,\frac{dU_\infty}{dx} = \frac{C_f}{2} + \frac{\theta}{x_c - x} , \qquad (3\text{-}2)$$

where x_c is the location of the fictitious source or sink, and may be obtained by solving Eqn. (3-2) for x_c, giving

$$x_c = x + \frac{\theta U_\infty^2/(\theta U_\infty^2)_o}{\frac{d}{dx}\,(PL-PR)} . \qquad (3\text{-}3)$$

Unfortunately, the PL and PR values are quite noisy, leading to violent fluctuations in the value of x_c. A second method is to arbitrarily adjust x_c to give the best match with experiments.

There are serious objections to using this correction term. Both methods for obtaining x_c depend on having experimental data available. This can hardly qualify as a prediction method! This correction term will not be used for a priori diffuser calculations. However, it will be employed in this chapter during the calibration process, so as to enable comparison with TBLPM's in [3].

D. Initial and Boundary Conditions

Values of H, δ^* are known at the starting location, and the initial τ_{max}/ρ is calculated assuming equilibrium conditions. The pressure gradient is prescribed and δ, u_β, u_τ calculated along the flow. Instead of imposing smoothed dp/dx values, as was done in [3], a tensioned spline [48] was fitted through the data and the derivative obtained numerically. Three-dimensional corrections were applied to detaching flows only.

It is again emphasized that detaching diffuser flows have to be calculated in simultaneous iteration, and the present prescribed pressure gradient mode is for comparison purposes only.

E. Comparison with Experiments

Predictions using the UIM equations are shown in Figs. 9 through 19. The numbers in parentheses after each flow is the identification assigned in [3].

Close agreement is obtained with data for accelerating and decelerating flows, including equilibrium b.l.'s of both types (see Figs. 9-13). Figs. 9a and 9b show the prediction for the relaxing flow of Bradshaw et al. (2400). With the attached lag parameter $\lambda_a = .025$, excellent agreement of H, δ^*, and $C_f/2$ are obtained. Fig. 9b shows the development of τ_{max}/ρ along the flow. If the data satisfied Bradshaw's correlation exactly, entrainment and maximum shear would coincide at every station. The predicted entrainment rate is somewhat low at the beginning, but is quite good for the rest of the flow. The low starting value is probably due to the assumption of equilibrium starting conditions, while in fact the flow is far removed from this state.

The program has problems predicting the reattaching Tillmann ledge flow (1500), Fig. 14. H and δ^* are overpredicted, while the skin friction values are too low. Nash and Hicks [3] were able to improve agreement with data by doubling the initial shear stress value. This is expected to improve the present prediction, but has not been attempted since this type of flow would normally be calculated in simultaneous iteration.

Problems encountered in calculating detaching b.l.'s with prescribed pressure gradient (cyclic iteration) were discussed in Chapter One. These are evident in Figs. 15 through 18. In all cases the flow proceeds towards detachment but relaxes prematurely. A 3-D correction term was included and adjusted to give the best possible match with data. The agreement is improved, but premature detachment occurs if too small a value of x_c is used. Moses' asymmetric diffuser flow, Fig 17, and Strickland-Simpson's airfoil flow, Fig. 18, will be recalculated with a full 2-D core in simultaneous iteration to show the improved prediction possible (see Chapter Six).

For contrast, Perry's flow (2900) was recalculated as a diffuser with a 1-D core, as described in the next chapter. The results, Fig. 16, bear out the claims made for the simultaneous iteration concept. The predicted b.l. quantities found by simultaneous iteration are much closer to the data than those obtained for the prescribed pressure gradient method, especially in the detaching region. In calculating the flow, it was assumed that the upper and lower b.l.'s were identical at the starting section. This is not the case, and it is expected that improved agreement would result if the correct starting values were available.

Finally, So and Mellor's [46] b.l. growing over a convex surface is shown in Fig. 19. The results are in accordance with Bradshaw's [47] observation that the turbulence production is suppressed on a convex wall b.l. and enhanced on a concave one. The skin friction falls along the flow as the turbulence level decreases, and the prediction is too high. This flow was included since the curved throat region of many diffusers is a convex surface, even though the flow length is small. It is presently not known how long the curved region has to be in order to have a significant effect on the downstream flow, nor is the magnitude of the curvature effect well established at this time (1976). However, this phenomenon is known to be important in many passage applications, and the effect needs to be pointed out so that improved TBLPM's that properly account for curvature effects will be created.

F. Chapter Conclusions

(a) The current TBLPM is capable of accurately predicting equilibrium flows, as well as accelerating and decelerating b.l.'s. For attached b.l.'s, its performance is as good as the best prediction method presented in the 1968 Stanford Conference. It has the additional advantage that it is capable of calculating detached flows.

(b) The method has no difficulty in predicting accelerating, mildly decelerating, and equilibrium flows. For detaching flows, the inclusion of the 3-D correction term improves the accuracy until the flow nears detachment; after this point the computed values are no longer accurate. Inclusion of the shear-stress lag equation is believed to be the reason for the good prediction of strongly perturbed flows. An exception is Tillmann's reattaching flow, which was not well predicted. B.l.'s over curved surfaces are not well predicted either.

(c) The procedure does extremely well for b.l.'s encountered in a typical diffuser, which exhibit mild acceleration in the inlet, strong acceleration around the throat, and strong deceleration in the diffusing section. Thus this method, when used in cyclic iteration (prescribed pressure gradient), shows the weaknesses seen in all the methods of the 1968 Conference for flows nearing detachment. However, all the methods in the 1968 Conference also use cyclic iteration. As shown in the next chapter, these difficulties near detachment do not occur when the simultaneous iteration procedure is used.

CHAPTER FOUR

RESULTS - ONE-DIMENSIONAL CORE DIFFUSERS

A. Discussion of Available Data

We restrict ourselves to two-dimensional diffusers and briefly review the currently available data.

Diffusers have either straight or curved centerlines, as depicted in Fig. K. They may be symmetric or asymmetric about the centerline.

The most widely studied is type (a), for which flow-regime charts were established by Fox and Kline [1]. Reneau et al. [45] created a set of data maps that can be used to estimate the overall pressure recovery, Cp. Carlson et al. [27] compared the performance of types (a) and (b). Fox and Kline[30] established flow-regime charts for type (c), sometimes called a circular arc diffuser. Sagi et al. [31] made measurements of both types (c) and (d).

In all the above experiments, only "zeroth-order" quantities were measured. These were Cp(x) and the gross qualitative features of the flow, such as the levels of unsteadiness from visualization of wall tufts and whether or not backflow was present in an intermittent or steady basis. The b.l. integral thicknesses at the inlet were recorded. There were no detailed measurements of the b.l. development along the flow, and no skin friction or turbulence data were taken.

Moses [7] measured the variation of Cp(x) and integral parameters along the wall of a type (e) diffuser in transitory stall. Unfortunately, the diffuser throat had a small radius, and it is possible that strong curvature effects may have introduced unexpected behavior in the b.l.

The most extensive data for a single unit available today is the airfoil type flow of Strickland and Simpson [32], also on a type (e) diffuser. Detailed measurements of the b.l. development are available, including details of the turbulence quantities along the flow. These data were taken with a directionally sensitive laser anemometer, so that the measurements are expected to be more accurate than pitot or hot-wire data in regions where the fluctuations are large and the meanflow direction uncertain.

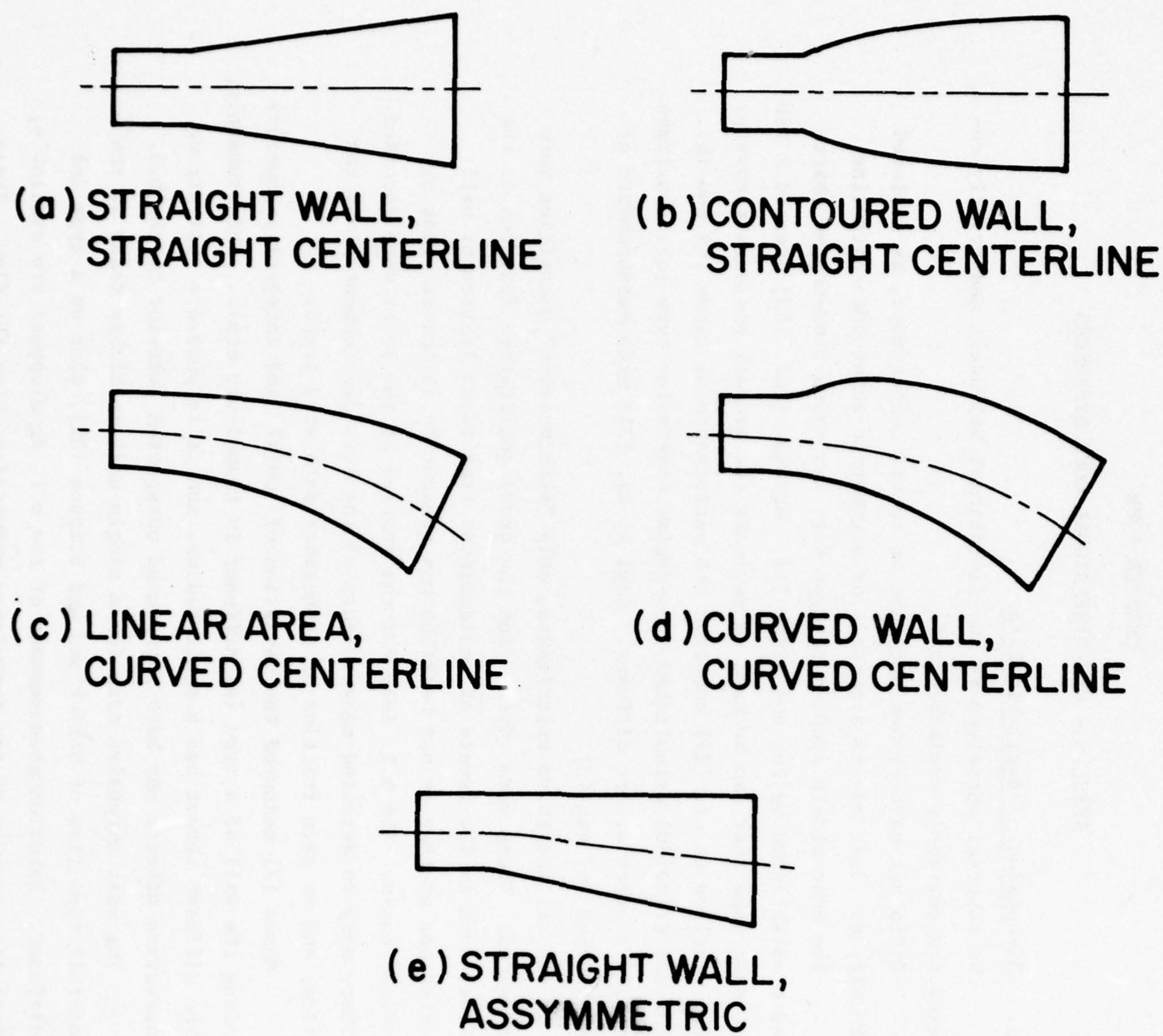


Fig. K. Classification of commonly used types of diffusers.

B. Results and Discussion

Figures 20 through 27 compare predictions of the current method with data. Figs. 20 and 21 compare the variation in velocity ratio u/u_{REF} along the diffuser walls with the data of Carlson et al. [27]. Fig. 20 is for an unstalled diffuser, for which the velocity variation is predicted within the uncertainty of the data. Fig. 21 is for a diffuser in transitory stall, and the calculated values are barely outside the uncertainty band in the region between the intermittent and fully developed detachment points. Fig. 22 shows the predicted variation in H, δ^*, and $C_f/2$ along this diffuser -- no data are available for comparison.

The predictions for a complete series of $N/W1 = 6$, $B1 = .030$ diffusers with area ratios (AR) varying from 1.5 to 2.65 are shown in Fig. 23. The calculations compare very well with the measurements of Carlson et al. [27]. The Cp values from the data maps of Reneau [45] are somewhat higher, but are well within the uncertainty of the data.

The predicted exit conditions for the same series of diffusers are shown in Fig. 24. Only one data point is available, for AR = 1.8. The agreement in this case is excellent.

Figure 25 is a replot of the predicted variation in exit Cp, in addition to which are shown the locations of the intermittent ($H = H_{SEP}$) and fully developed ($C_f/2 = 0$) detachment points as fractions of the diffuser length (x/L). Also shown are the locations designated TI (incipient transitory stall) and IT (intermittent transitory stall) from flow visualization of Carlson et al. [27]. For the few data points available, the TI location is quite well predicted by the intermittent detachment point according to the Sandborn criterion. The location of fully developed detachment occurs somewhat downstream of this point.

Figure 26 is a summary of the performance of $N/W1 = 12$ diffusers as a function of the divergence angle 2θ, for the inlet blockage B1 varying from .007 to .050. The accuracy of the prediction decreases as 2θ and B1 increase. This conclusion is in accordance with the findings of Woolley et al. [4]. For small B1, the b.l. is a correction to the throughflow, so that small errors in δ^* cause even smaller errors in Cp. As the b.l. becomes a significant portion of the flow, the accuracy of the Cp predictions decreases. The data for $B1 = .050$ have a sharp peak, following which it levels off at a constant value of Cp. The beginnings of this peaky behavior can be seen in the curve for $B1 = .030$. The calculation is unable to follow this trend. The behavior of the calculated results is similar for all inlet blockages.

The locations marked X indicate the value of 2θ at which the shear layers from the upper and lower walls begin to interfere with each other. No irrotational region remains, and, viewed strictly, the calculation method is not valid beyond this point.

Finally, Fig. 27 is a summary of all diffusers that have been run. The current method is able to predict Cp of all tested units to about the uncertainty in the data, with the exception of the $B1 = .050$ case. For all B1, the present calculation is capable of predicting Cp within ± 6% of the data for all diffusers whose divergence angle 2θ is less than $1.2 \times 2\theta_{a-a}$. The range of calculation has therefore been doubled from $2\theta/2\theta_{a-a} = 0.6$ in the method of Woolley and Kline [4] to $2\theta/2\theta_{a-a} = 1.2$ in the present method. This extension carries the method well into a region beyond the peak Cp^* -- up to approximately the line of appreciable stall.

C. Chapter Conclusions

(a) The diffuser calculation method assuming a 1-D core and symmetrical b.l.'s is capable of predicting the performance in the transitory stall regime well past the peak in the Cp curve. Accuracies of ± 6% can be obtained up to divergence angles that are 1.2 times the location of $2\theta_{a-a}$, even when the difficult case of $B1 = .050$ is included.

(b) Prediction accuracy decreases with increasing inlet blockage. Neglecting the highest blockage value of .050, the ± 6% accuracy in Cp can be obtained for $2\theta/2\theta_{a-a} = 1.8$.

(c) The predicted location of intermittent detachment $(H = H_{SEP})$ according to the Sandborn criteria occurs very close to the point designated as "incipient transitory stall" (TI) in the flow visualization data of diffusers. The location of zero wall shear occurs a small distance downstream of this point.

(d) The program was tailored to model detached flows using very sparse data. The program output consists of turbulent quantities, wall shear stresses and entrainment values for which experimental data are almost nonexistent. Much more data of detailed b.l. development and turbulence quantities in the detaching regions is needed to extend the applicability and either fully verify or improve the present model.

CHAPTER FIVE

DEVELOPMENT OF PREDICTION METHOD FOR 2-D CORE DIFFUSERS

A. Limitations of the 1-D Core Method

The predictions of symmetric diffusers with symmetric b.l.'s using the 1-D irrotational core model were shown to be quite acceptable for engineering purposes.

There are many cases, however, for which such a 1-D core is obviously a poor approximation, such as for a grossly asymmetrical diffuser. Not so obvious is the fact that the use of the 1-D core approximation and the simultaneous streamwise marching procedure has enabled us to convert from an elliptic problem to a fully parabolic one. Certain essential information has been inevitably lost in this process. The flow of information in the numerical procedure is in the downstream direction only -- all upstream propagation is totally absent.

It is well known that the effect of downstream blockage can play an important role in determining the upstream pressure gradient and hence can control the detachment process. This elliptic field effect can be clearly seen in the experiment of Chui et al. [33] on a fully stalled diffuser. The dominant adverse pressure gradient occurs well ahead of the diffuser throat, and is due mainly to the blockage of the stalled flow downstream of the throat. The elliptic field effect causes the streamlines to diverge ahead of the throat, in a region that is bounded by parallel walls. A 1-D calculation would predict acceleration in this region and could obviously never predict this flow even approximately for the lowest-order quantities.

The importance of the elliptic field effect in the transitory stall regime is not known. It is probably not as pronounced as in the fully stalled regime, on account of the smaller detachment zones and the consequent smaller curvatures of the streamlines.

There is a basic dilemma, however. It was pointed out in Chapter One that detached b.l.'s can only be calculated simultaneously with the adjacent irrotational freestream. Numerical stability requires that the pressure gradient acting on a b.l. at and near detachment be exactly that which occurs as the result of mutual interaction between it and the irrotational core. The solution to Laplace's equation in the core requires information along the entire boundary in order to be well posed. There is thus a basic conflict between the requirements of the b.l. and the inviscid core. The final solution will therefore have to be obtained iteratively, each iteration being designed in such a manner as to satisfy these separate and conflicting requirements.

B. Procedure for Calculation of Diffusers with 2-D Core

The basic outline of the calculation method for diffusers with 2-D core will now be developed. The next few sections present the b.l. and potential flow schemes.

In cyclic iterative calculations of the type used by Woolley [4] and White [5], the solution of Laplace's equation in the domain bounded by the diffuser walls gives an estimate of the velocity gradient in the streamwise direction, which is prescribed to calculate the b.l. growth. The displacement thickness δ^* is subtracted from the diffuser walls to give a new effective flow channel (EFC). Laplace's equation is solved in this new EFC, giving a new estimate of the velocity gradient, which is used to obtain a new δ^*, etc. The process is considered to have converged if the change in δ^* or the velocity gradient between successive iterations is smaller than a preselected tolerance.

This scheme works for unstalled diffusers and for the fully stalled case for which the simplifying assumption of constant pressure in the stalled zone permits the detached b.l. to be modeled as a free streamline problem. In this case of a fully stalled diffuser, the prescribed pressure gradient calculation is terminated before intermittent detachment, and the δ^* line is extrapolated into the stalled zone, where its location is iteratively determined. For detaching b.l.'s, however, cyclic iteration is numerically unstable, and this procedure diverges. Further, the pressure in the transitory stalled zone is not constant, and substantial pressure recovery occurs in this state, so that a free streamline model is inappropriate. A new scheme that avoids these problems is needed.

An examination of the data of Moses [7] and Strickland et al. [32] shows that the velocity profiles in the irrotational core of diffusers operating in the transitory stall regime are quite linear, excluding, of course, regions of sharp curvature in δ^* such as near the throat. That is, data show that there is a linear variation in edge velocity between the upper and lower δ^* lines. In fact, after detachment, the profile becomes almost one-dimensional, which may be why the 1-D core method is so successful.

We note further from the data of Smith and Kline [34] that transitory stall begins and is restricted to one wall, even if the two walls of the diffuser are nominally symmetric. This is not surprising since when one b.l. detaches, the pressure gradient on the opposite wall is immediately relaxed. In an asymmetric diffuser there is no question that detachment is restricted to the diverging wall.

The diffuser will therefore be modeled with an upper wall that has an attached b.l. and a lower wall in which the layer may or may not be detached. The upper b.l. and that portion of the lower b.l. well ahead of detachment can be calculated with prescribed pressure gradient, while the detaching and detached regions have to be simultaneously calculated with the inviscid core.

If the edge velocity, $U_{\infty 1D}$, from the lower wall b.l. calculation is the same as the edge velocity, $U_{\infty 2D}$, obtained from the solution of Laplace's equation in the EFC defined by the new δ^* lines, the solution is considered to have converged. Otherwise, the process is continued by prescribing $U_{\infty 2D}$ on the upper wall and using the δ^*_u so obtained to define a new EFC. Laplace's equation solved in this new domain gives a new $U_{\infty 2D}$ against which the edge velocity $U_{\infty 1D}$ from a new lower wall b.l. calculation may be compared, and so on.

The convergence criterion used is that for all stations, Cp $\leq$ Cperor, where

$$\text{Cperor} = \left| Cp_{1D} - Cp_{2D} \right|$$

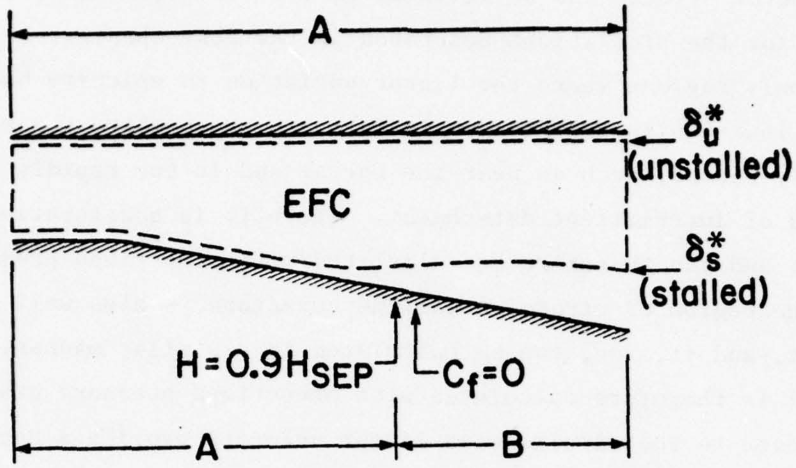
$$\text{where} \quad Cp_{1D} = 1 - (U_{\infty 1D}/U_{REF})^2 \qquad (5\text{-}1)$$

$$\text{and} \quad Cp_{2D} = 1 - (U_{\infty 2D}/U_{REF})^2 \quad .$$

A Cperor = .025 can be achieved in 8 to 10 iterations, and has been used for the predictions described in the next chapter.

The only regions where the linear variation in velocity between the upper and lower walls breaks down is in those areas where the streamlines are sharply curved, such as near the throat and in the rapidly growing zone ahead of intermittent detachment. The b.l. is accelerating around the throat and can therefore be calculated with prescribed pressure gradient. The region of strong streamline curvature is also well ahead of detachment, and it, too, can be calculated in a similar manner. The lower wall is therefore calculated with prescribed pressure gradient and switched over to the simultaneous linear velocity profile scheme for $H \geq 0.9\ H_{SEP}$, where H_{SEP} is the Sandborn criterion. The entire process is shown in Figs. L and M.

In summary, the present scheme is broadly similar to a predictor-corrector method. The linear core profile method is the predictor, which provides an estimate of the lower wall δ_s^* and edge velocity $U_{\infty 1Ds}$. The corrector is the values of the edge velocity $U_{\infty 2Ds}$ obtained from the solution of Laplace's equation in the EFC. When the predicted and corrected C_p values agree within an acceptable tolerance, a converged solution is obtained. The final solution reflects the accuracy of the corrector, and the approximations of the predictor are no longer present.

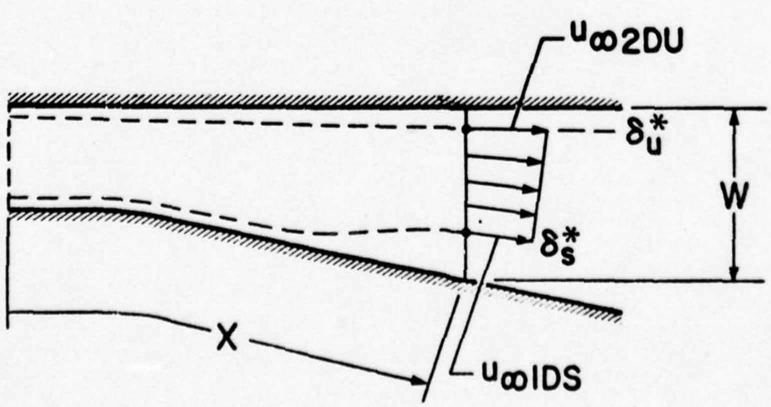


A = prescribed pressure gradient calculation
B = simultaneous linear core profile method

Fig. L. Sketch of the 2-D core diffuser illustrating regions where the two types of calculation methods are used.

C. <u>Simultaneous B.L. Calculation with Linear Core Velocity Profile</u>

The location of the upper wall δ_u^* line is known from the last iteration, as is the velocity distribution $U_{\infty 2Du}$ from the 2-D potential flow calculation in the resulting EFC. The diffuser width W is known from the input geometry. We wish to calculate the lower wall δ_s^* and the corresponding edge velocity, $U_{\infty 1Ds}$, assuming linear variation of velocity between the upper and lower δ^* lines. The figure below shows the situation at location x.



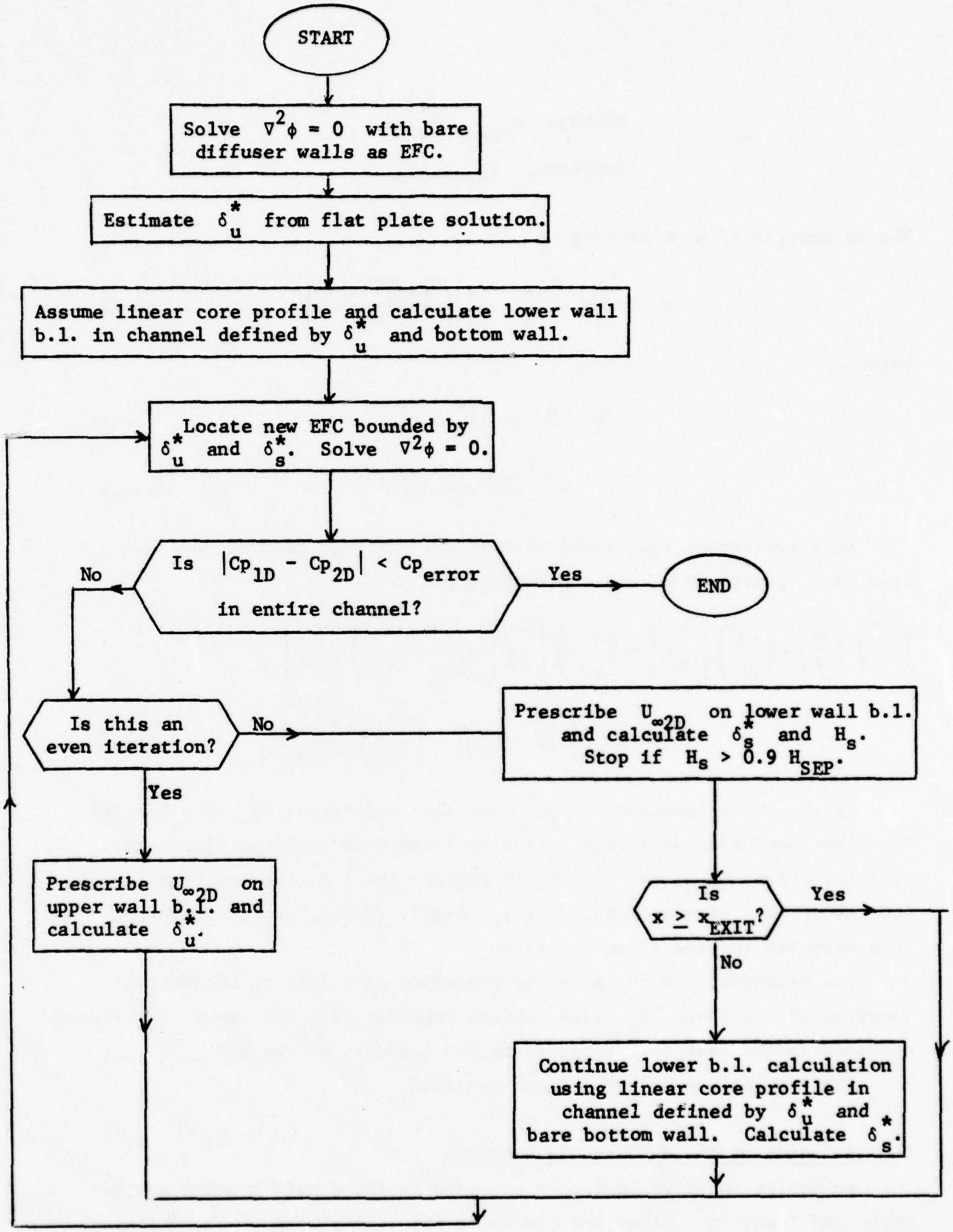


Fig. M. Flowchart illustrating the two-dimensional core calculation method.

Knowns: $U_{\infty 2Du}$, δ_u^*, W

Unknowns: $U_{\infty 1Ds}$, δ_s^* .

The volumetric flow at section x is

$$Q = \left(W - \delta_u^* - \delta_s^*\right)\left(\frac{U_{\infty 2Du} + U_{\infty 1Ds}}{2}\right) . \qquad (5\text{-}2)$$

Define

$$W_e \triangleq W - \delta_u^* - \delta_s^* , \qquad (5\text{-}3)$$

$$U_e = \frac{U_{\infty 2Du} + U_{\infty 1Ds}}{2} . \qquad (5\text{-}4)$$

Differentiating Eqn. (5-2) with respect to x, setting $dQ/dx = 0$ from mass conservation, and rearranging gives

$$\left(\frac{-\delta^*}{\delta}\right)\left\{\frac{d\delta}{dx}\right\} + \left(\frac{-\delta}{2\tilde{U}_\infty}\right)\left\{\frac{du_\beta}{dx}\right\} + \left(\frac{-\delta}{\kappa\tilde{U}_\infty}\right)\left\{\frac{du_\tau}{dx}\right\} + \left(\frac{\delta^*}{\tilde{U}_\infty} + \frac{W_e}{2U_e}\right)\left\{\frac{d\tilde{U}_\infty}{dx}\right\}$$

$$= \left\{\frac{-d}{dx}\left(W - \delta_u^*\right)\right\} + \left(\frac{-W_e}{2U_e}\right)\left\{\frac{dU_{\infty 2DU}}{dx}\right\} . \qquad (5\text{-}5)$$

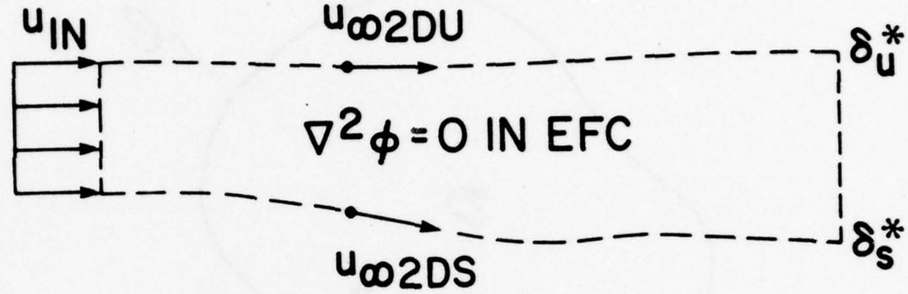
In the above equation, $U_{\infty 1Ds}$ has been written as $\tilde{U}_\infty$ for brevity. The right-hand side of this equation is known from previous iteration. Therefore, Eqn. (5-5) can be used to replace the 1-D core equation (2-43) and the new set of equations, (2-26), (2-32), (2-34), and (5-5) solved in a stepwise fashion along the flow.

The dependent variables can be processed as before to obtain the location of the lower δ_s^* line, which, together with the upper δ_u^* line obtained in the last iteration, forms the boundary of the EFC.

The 2-D Laplace solver is next outlined.

D. *Solution of the 2-D Laplace Equation*

We desire to solve Laplace's equation in the domain bounded by the upper and lower δ^* lines and the inlet and exit planes of the diffuser. The velocity is assumed constant across the inlet, and it is specified that there is no flow across the upper and lower δ^* lines, which are thus approximated as streamlines. The situation is depicted in the following figure.



The edge velocity $U_{\infty 2D}$ is needed along the entire boundary of the EFC. This is similar to the problem solved by Woolley et al. [4], and lends itself naturally to a boundary integral method, since only the values along the boundary are required.

Two shortcomings of the method used in [4] were that the exit velocity was assumed to be one-dimensional and the equation formulation used the Cauchy-Riemann conditions, which necessitated the taking of numerical derivatives with their potential for large errors.

Recently my colleague Rinehart [36] has developed a similar method for solving the 2-D Laplace equation which avoids both these difficulties. Since his work is soon to be published, only an outline of the method will be presented.

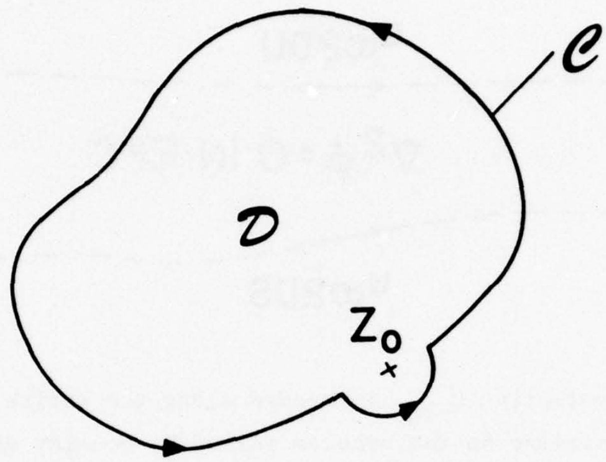
Consider a simply connected domain $\mathcal{D}$ in the complex plane bounded by a smooth closed contour $\mathcal{C}$. Let z_o be an interior point, and, if $f(z)$ is analytic in $\mathcal{D}$, Cauchy's integral formula gives the value of the function at this point as

$$2\pi i\ f(z_o) = \int_C \frac{f(z)}{z-z_o}\, dz \quad . \tag{5-6}$$

Now let z_o approach the contour $\mathcal{C}$. In the limit when z_o is on $\mathcal{C}$, we have the Plemelj formula,

$$i\pi\ f(z_o) = P\int_C \frac{f(z)}{z-z_o}\, dz \quad . \tag{5-7}$$

The integral on the right-hand side is to be interpreted in the Cauchy principal value sense.



If C is not a smooth curve and z_o is a corner point, Eqn. (5-7) is modified to

$$i\alpha\, f(z_o) = P\int_C \frac{f(z)}{z-z_o}\,dz \quad , \tag{5-8}$$

where α is the interior angle at the corner. For a smooth curve, $\alpha = \pi$ and Eqn. (5-8) reduces to (5-7). For further details, see Muskhelishvili [37].

The boundary C is discretized into N segments whose end points are numbered increasing in the counterclockwise direction, as shown on the next page.

Let the boundary point z_o at which the function is to be evaluated be located at node C_m. Then, since the singularity is present at this point alone, Eqn. (5-7) can be rewritten as the sum of an ordinary contour integral plus a principal value integral,

$$i\pi\, f(z_o) = P\int_{C_{PV}} \frac{f(z)}{z-z_o}\,dz + \int_{C-C_{PV}} \frac{f(z)}{z-z_o}\,dz \quad . \tag{5-9}$$

$f(z)$ is expanded in a separate Taylor series expansion along each interval of the boundary, and on performing integrations of the resulting terms we get

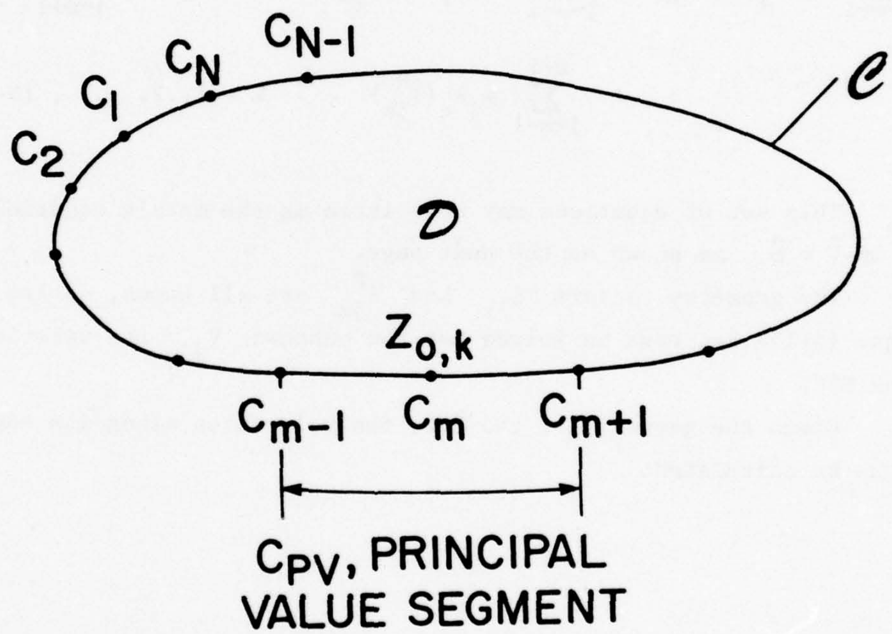
$$i\pi\, f(z_{o,k}) = \sum_{j=m+1}^{m-2} f_j \Lambda_{jk} + \sum_{j=m-1}^{m+1} f_j \Lambda^P_{jk} \;, \qquad k = 1, 2, \ldots, N-1 \tag{5-10}$$

where

$f_j = f(z_j)$,

Λ^P_{jk} = the geometry factors for the principal value segment at the node j when the singularity is at node k,

Λ_{jk} = the geometry factors of the rest of the boundary.



Rewriting Eqn. (5-10) for the m^{th} node and transposing gives

$$\sum_{j=m+1}^{m+N-2} f_j \Lambda_{jk} + f_{m-1}\,\Lambda^P_{m-1,k} + f_m(\Lambda^P_{mk} - i\pi) + f_{m+1}\,\Lambda^P_{m+1,k} = 0 \;,$$

$$\text{for } k = 1, 2, \ldots, (N-1) \;. \tag{5-11}$$

Define

$$\Lambda_{jk} = G_{jk} + iH_{jk} \;,$$
$$\Lambda^P_{jk} = G^P_{jk} + iH^P_{jk} \;. \tag{5-12}$$

Choosing the analytic function as $f(z) = \ln V - i\alpha$, where V is the magnitude of the velocity and α is the local streamline angle, $\alpha = \tan^{-1}\left(\frac{v}{u}\right)$, we have

$$f_j = \ln V_j - i\alpha_j \quad , \qquad j = 1, 2, \ldots , N \quad . \tag{5-13}$$

Substituting Eqns. (5-12) and (5-13) into (5-11) and taking the imaginary part gives

$$\sum_{j=m+1}^{m+N-2} \ln V_j \; Im(\Lambda_{jk}) + \sum_{j=m-1}^{m+1} \ln V_j \; Im(\Lambda^P_{jk}) - \pi \ln V_k = \sum_{j=m+1}^{m+N-2} \alpha_j R_e(\Lambda_{jk})$$

$$+ \sum_{j=m-1}^{m+1} \alpha_j R_e(\Lambda^P_{jk}) \quad , \qquad k = 1, 2, \ldots , (N-1) \quad . \tag{5-14}$$

This set of equations may be written as the matrix equation, $A \ln \vec{V} = \vec{b}$, as shown on the next page.

The geometry factors Λ_{jk} and Λ^P_{jk} are all known, as are the α_j. Eqn. (5-15) can thus be solved for the unknown V_j, the velocities along the EFC.

Given the geometry of the EFC, the velocities along its edge can thus be calculated.

$$
\begin{bmatrix}
(-\pi+H^P_{11}) & (H^P_{12}+H_{12}) & H_{13} & \cdots & H_{1,N-2} & H^P_{1,N-1} \\
H^P_{21} & (-\pi+H^P_{22}) & (H^P_{23}+H_{23}) & & H_{2,N-2} & H_{2,N-1} \\
H_{31} & H^P_{32} & (-\pi+H^P_{33}) & & & \cdot \\
\cdot & & & \cdot & & \cdot \\
\cdot & & & & \cdot & \cdot \\
\cdot & & & & \cdot & \cdot \\
\cdot & & & & \cdot & \cdot \\
H^P_{N-1,1} & \cdot & \cdot & & \cdot & (-\pi+H^P_{N-1,N-1})
\end{bmatrix}
\begin{pmatrix}
\ln V_1 \\ \ln V_2 \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \ln V_{N-1}
\end{pmatrix}
=
\begin{pmatrix}
b_1 \\ b_2 \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ b_{N-1}
\end{pmatrix}
\quad (5\text{-}15)
$$

CHAPTER SIX

RESULTS - TWO-DIMENSIONAL CORE DIFFUSERS

A. Moses' Asymmetric Diffuser Flow

Moses' diffuser was of type (e) with one diverging wall at an angle of 11.31 degrees, $AR = 2.5$, $L/W1 = 7.5$, and the b.l. thickness at the throat, $\theta/W1 = .007$. The sharp throat radius, $R/W1 = .57$, caused convergence problems because of the rapid change of $Cp(x)$ in the throat region. An artificial increase of $R/W1$ to 1.0 allowed convergence without materially affecting the downstream solution. A 3-D correction with $X_c = 100.0$ ft was necessary to match the data. The results are shown in Figs. 28 and 29.

Cp on both walls is predicted to the accuracy in the data, which is estimated to be ± 6%. The qualitative trends of $Cp(x)$ are correct, including the sharp suction peak at the throat of the diverging wall, and the steady increase on the unstalled wall. The suction peak value is considerably underpredicted, but the data here are quite questionable on account of the rapid streamwise variation of Cp in this region. The greatest deviation from the data occurs in the region of detachment. H and δ^* are quite well predicted before detachment, but are considerably overpredicted in the reversed flow region.

B. Strickland-Simpson Airfoil Type Flow

This flow is in a type (e) diffuser with a flat bottom wall. The top wall converges and then diverges, giving a pressure distribution similar to that on the upper surface of an airfoil.

The flow was calculated with prescribed pressure gradient up to 8.11 ft, at which point the experimenters had to remove most of the upper wall b.l. to force the flow to detach on the lower wall. The rest of the flow was calculated simultaneously with a full 2-D inviscid core.

Initial attempts to predict this flow resulted in very rapid growth of δ^* and H after detachment, similar to that for the Moses diffuser flow. To prevent this through a 3-D correction would have required negative values of X_c, which is not realistic for a decelerating flow with growing sidewall b.l.'s. Instead, the lag equation was removed after detachment, giving the results shown in Figs. 30 and 31.

The b.l. growth, H, δ^*, and $C_f/2$ for both the upper and lower walls are very well predicted, except for a small deviation near the exit. The location of both intermittent and fully developed detachment is closely predicted, but the skin-friction values in the reversed flow region are somewhat smaller in magnitude than the data. $C_f/2$ for the upper wall is slightly overpredicted, but the uncertainties in these data are quite large on account of the thinness of this b.l. and the consequent poorer definition the wall regions of the velocity profiles.

Figure 31 shows the variation of Cp(x), U_∞, and the nondimensional entrainment rate $\frac{1}{U_\infty} dQ/dx$. U_∞ is underpredicted by about 5% in the detached region, leading to a 6% overprediction of Cp. The entrainment rate is quite good until detachment, when it abruptly rises in response to the removal of the lag equation. The value is almost 100% too large at detachment, following which the deviation begins to decrease. The reason for the excellent agreement of the mean flow parameters using this incorrect value of the entrainment rate is not known. It is a peculiar coincidence, however, that the values of $\tau_{max}/\rho U_\infty^2$ computed from the data using Eqn. (2-40) and plotted in Fig. 8 display this same trend. The maximum shear stress computed from the data also have their largest deviation near the detachment point.

C. Discussion

Both diffusers used for comparing with the 2-D core calculation are type (e), with one diverging wall, these being the only data available. This is an unfortunate choice, since the flow regimes for asymmetric diffusers are expected to be somewhat different from those for symmetric units. Since the divergence is limited to one wall, the b.l. on this wall begins to detach much earlier than on a symmetric unit with the same 2θ. Line a-a therefore occurs at a lower 2θ and the entire flow regime shifts downward.

Preferential stall occurs and is restricted to the diverging wall. The transitory stall regime is expected to be almost nonexistent for asymmetric diffusers, the flow changing from an essentially unstalled to a quasi-steady fully stalled flow as the divergence angle is increased at constant L/W1. The limited data available support this description.

As a consequence, both diffusers are actually operating in the fully stalled mode with a relatively steady recirculating separation bubble, even though they should both be in the small transitory stall regime, according to the flow regime chart, Fig. 1. The present calculation method was not designed for, and does not give accurate values for, b.l. parameters in the fully stalled zone, even though the zeroth-order quantities, the Cp, and locations of detachment are quite well predicted. The justification for removal of the lag equation in the reversed flow region is that the detached lag parameter λ_d was determined by matching data from a diffuser operating in the transitory stall regime, while the Strickland-Simpson flow is closer to that of a fully stalled case. It appears from the good predictions obtained with no lag equation in detached flow, that perhaps a higher λ_d is appropriate in this zone, since $\lambda_d \to \infty$ corresponds to an instantaneous response between the local velocity profile and the shear stresses.

An interesting feature of the Strickland-Simpson flow is the region in the neighborhood of partial removal of the upper b.l. at the entrance to the diffusing section. The upper b.l. undergoes a severe perturbation and slowly relaxes.

The largest deviation from the data in all the diffusers that have been run occurs in the region between intermittent detachment and the location of zero wall shear. This is evident in Fig. 29 (2-D Moses diffuser) and Figs. 20 and 21 (1-D Carlson diffuser). The present calculation evidently cannot model the flow closely in this region. The agreement improves both upstream and downstream of this zone.

The reasons for this deviation may be:

(a) The Coles' profile does not adequately represent measured velocity profiles in the neighborhood of zero wall shear.

(b) The eddy-viscosity formulation, Fig. 8, has the greatest deviation from data in this region.

(c) The effect of neglected terms in the momentum integral equation, such as the normal stress terms, is greatest in the detachment zone.

(d) The turbulence measurements have the greatest uncertainty in this region on account of the small mean and large fluctuation magnitudes.

Considering all these negative factors, the overall success of the current method is gratifying.

CHAPTER SEVEN

SUMMARY

A. Conclusions

(a) A calculation method has been developed that successfully predicts three types of flows:

- ••• turbulent boundary layers with prescribed pressure gradient,
- ••• symmetric diffusers with one-dimensional core,
- ••• diffusers with two-dimensional inviscid core.

The last two types can have attached or detached boundary layers.

(b) Diffuser predictions to about ± 6% accuracy in Cp can be made up to about the location of the line of appreciable stall in the transitory stall regime. This corresponds to $2\theta/2\theta_{a-a} = 1.2$. Prediction accuracy increases with decreasing inlet blockage.

(c) The mean boundary layer parameters H, δ^*, $C_f/2$, etc., are extremely well predicted. For diffusers, the locations of both intermittent detachment and zero wall shear are also predicted with remarkable accuracy. However, skin friction and entrainment in the reversed flow region are only fair.

(d) Execution times for the program on an IBM 370/168 are on the order of 0.25 seconds for a straight boundary layer calculation, and 1.0 sec for a 2-D Laplace equation solution. A 1-D core diffuser takes about 0.5 sec. A typical full 2-D calculation involves 6 to 10 iterations of the boundary layer and inviscid core and takes about 10 secs to execute.

(e) The overall success of the method legitimizes the concept of simultaneous iteration as a means of preventing the singular behavior of the classical boundary layer calculations in the neighborhood of detachment.

(f) The eddy-viscosity scheme used in this report was based on extremely sparse information. Improved predictions will be possible only when more data on detached and detaching boundary layer behavior become available.

B. Recommendations for Further Work

(a) An understanding of the factors controlling the behavior of detached flows is a prerequisite to being able to predict it. The studies of C_p and flow visualization of diffusers by the Stanford HTTM group over the last 15 years have greatly increased the understanding of the qualitative features of these flows. These studies now need to be extended to include detailed quantitative flowfield information, such as the turbulence field, intermittency and skin-friction along the walls. Because of the complicated nature of the detached flow regions, these measurements will not be easy, and new measurement techniques such as laser Doppler anemometers, etc., may have to be developed.

(b) The currently used eddy viscosity concept is a gross approximation to the actual flow. When new data become available, scaling laws relating the shear stresses to the turbulent kinetic energy or entrainment will permit improved calculation methods to be developed.

(c) The current method can be extended quite readily to the 1-D core axisymmetric case for both incompressible and compressible diffusers. The next step is the case with the incompressible inviscid core calculated from a solution of Laplace's equation in the axisymmetric effective flow channel. The corresponding compressible case must await the development of a fast algorithm for compressible potential flow.

(d) An alternative approach to the iterative matching procedure between the boundary layer and the core, and the inherent convergence problems thereof, is to couple both regions into one large domain and solve the whole flowfield as an elliptic problem. The equations for such a scheme have been developed, but no solution has been attempted. The approach looks promising.

References

[1] Fox, R. W., and Kline, S. J., "Flow Regime Data and Design Methods for Curved Subsonic Diffusers," Trans. ASME, J. Basic Engrg., Ser. D, v84, 1962, pp. 303-312.

[2] Schlichting, H., Boundary Layer Theory, McGraw-Hill, 6th edition, 1968.

[3] Kline, S. J., Morkovin, M. V., Sovran, G., Cockrell, D. J., editors, Computation of Turbulent Boundary Layers - 1968, AFOSR-IFP-Stanford Conference, vol. 1, Thermosciences Div., M. E. Dept., Stanford Univ., 1969.

[4] Woolley, R. L., Kline, S. J., "A Method of Calculation for a Fully Stalled Flow," Report MD-33, Thermosciences Div., M. E. Dept., Stanford University, Nov. 1973.

[5] White, J. W., Kline, S. J., "A Calculation Method for Incompressible Axisymmetric Flows, Including Unseparated, Fully Separated, and Free Surface Flows," Rept. MD-35, Thermosciences Div., M. E. Dept., Stanford University, May 1975.

[6] Cebeci, T., Mosinskis, G. J., and Smith, A. M. O., "Calculation of Separation Points in Incompressible Turbulent Flows," J. Aircraft, vol. 9, No. 9, pp. 618-624, Sept. 1972.

[7] Moses, H. L., Goldberger, T., Chappell, J. R., "Boundary Layer Separation in Internal Flow," Rept. 81, M.I.T. Gas Turbine Lab., Sept. 1965.

[8] Moses, H. L., Chappell, J. R., "Turbulent Boundary Layers in Diffusers Exhibiting Partial Stall," Trans. ASME, J. Basic Engrg., pp. 655-665, Sept. 1967.

[9] AGARD Conf. Proc. No. 168, Flow Separation, pp. RTD-1 to RTD-10, Nov. 1975.

[10] Bower, W. W., "An Analytical Procedure for the Calculation of Attached and Separated Subsonic Diffuser Flows," AIAA Paper 74-1173, 1974.

[11] Alber, I. E., "Similar Solutions for a Family of Separated Turbulent Boundary Layers," AIAA Paper 71-203, Jan. 1971.

[12] Sandborn, V. A., Liu, C. Y., "On Turbulent Boundary Layer Separation," J. Fluid Mech., Vol. 32, Part 2, pp. 293-304, 1968.

[13] Senoo, Y., Nishi, M., "Prediction of Flow Separation in a Diffuser by a Boundary Layer Calculation," submitted to J. Fluids Engrg., ASME, Aug. 1975.

[14] Rotta, J. C., "Critical Evaluation of Methods for Calculating the Development of Turbulent Boundary Layers," Fluid Mechanics of Internal Flow, G. Sovran, editor, Elsevier, 1967.

[15] Hirst, E. A., Reynolds, W. C., "An Integral Prediction Method for Turbulent Boundary Layers Using the Turbulent Kinetic Energy Equation," Report MD-21, Thermosciences Div., M. E. Dept., Stanford Univ., June 1968.

[16] Head, M. R., "Entrainment in the Turbulent Boundary Layer," Aero. Res. Council Rep. and Mem. 3152, 1960.

[17] Coles, D., "The Law of the Wake in the Turbulent Boundary Layer," J. Fluid Mech., Vol. 1, pp. 191-226, 1956.

[18] McDonald, H., Stoddard, J. A. P., "On the Development of the Incompressible Turbulent Boundary Layer," Aero. Res. Council Rep. and Mem. 3484, 1967.

[19] Kuhn, G. D., Nielsen, J. N., "An Analytical Method for Calculating Turbulent Separated Flows Due to Adverse Pressure Gradients," Proj. SQUID, TR NEAR-1-PU, Oct. 1971.

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

[21] Bradshaw, P., Ferriss, D. H., Atwell, N. P., "Calculation of Boundary Layer Development Using the Turbulent Energy Equation," J. Fluid Mech., Vol. 28, Part 3, pp. 593-616, 1967.

[22] Clauser, F. H., "Turbulent Boundary Layers in Adverse Pressure Gradients," J. Aero. Sci., Vol. 21, pp. 91-108, 1954.

[23] McD. Galbraith, R. A., Head, M. R., "Eddy Viscosity and Mixing Length from Measured Boundary Layer Developments," Aero Quart., pp. 133-154, May 1975.

[24] Liepmann, H. W., Laufer, J., "Investigation of Free Turbulent Mixing," NACA TN 1257, Aug. 1947.

[25] White, F. M., Viscous Fluid Flow, McGraw-Hill, 1974, pp. 512-530.

[26] Goldberg, P., "Upstream History and Apparent Stress in Turbulent Boundary Layers," Rept. 85, M.I.T. Gas Turb. Lab., May 1966.

[27] Carlson, J. J., Johnston, J. P., "Effects of Wall Shape on Flow Regimes and Performance in Straight, Two-Dimensional Diffusers," Rept. PD-11, Thermosciences Div., M. E. Dept., Stanford Univ., June 1965.

[28] Waitman, B. A., Reneau, L. R., Kline, S. J., "Effects of Inlet Conditions on Performance of 2-D Diffusers," Rept. PD-5, Thermosciences Div., M. E. Dept., Stanford Univ., Mar. 1960.

[29] Fox, R. W., Abbott, D. E., Kline, S. J., "Optimum Design of Straight Walled Diffusers," Rept. PD-4, Thermosciences Div., M. E. Dept., Stanford Univ., June 1958.

[30] Fox, R. W., Kline, S. J., "Flow Regime Data and Design Methods for Curved Subsonic Diffusers," Rept. PD-6, Thermosciences Div., M. E. Dept., Stanford Univ., Aug. 1960.

[31] Sagi, C. J., Johnston, J. P., Kline, S. J., "The Design and Performance of Two-Dimensional Curved Subsonic Diffusers," Rept. PD-9, Thermosciences Div., M. E. Dept., Stanford Univ., May 1965.

[32] Strickland, J. H., Simpson, R. L., "The Separating Turbulent Boundary Layer: An Experimental Study of an Airfoil Type Flow," Tech. Rept. WT-2, Thermal and Fluid Sci. Center, S. M. Univ., Aug. 1973.

[33] Chui, G., Kline, S. J., "Investigation of a Fully Stalled Turbulent Flowfield," Rept. MD-19, Thermosciences Div., M. E. Dept., Stanford Univ., Aug. 1967.

[34] Smith, C. R., Kline, S. J., "An Experimental Investigation of the Transitory Stall Regime in Two-Dimensional Diffusers Including the Effects of Periodically Disturbed Inlet Conditions," Rept. PD-15, Thermosciences Div., M. E. Dept., Stanford Univ., Aug. 1971.

[35] Thompson, B. G. J., "A Critical Review of Existing Methods of Calculating the Turbulent Boundary Layer," AIAA J., Vol. 3, pp. 746-7, 1964.

[36] Rinehart, F., "A Boundary Integral Method for the Solution of Two-Dimensional Laplace's Equation Using the Plemelj Formula," unpublished report, Thermosciences Div., M. E. Dept., Stanford Univ.

[37] Muskhelishvili, N. I., Singular Integral Equations, P. Nordhoff N. V., Holland, pp. 42-72, 1958.

[38] Tani, I., "Flow Separation on a Step," Boundary Layer Research, Springer-Verlag Publ., Berlin, pp. 377-386, 1958.

[39] Bradshaw, P., Ferriss, D. H., Johnson, R. F., "Turbulence in the Noise Producing Region of a Circular Jet," J. Fluid Mech., Vol. 9, Part 4, pp. 591-625, 1964.

[40] Klebanoff, P. S., "Characteristics of Turbulence in a Boundary Layer with Zero Pressure Gradient," NACA TN 3178, July 1954.

[41] Bradshaw, P., Wong, F. Y. F., "The Reattachment and Relaxation of a Turbulent Shear Layer," J. Fluid Mech., Vol. 52, Part 1, pp. 113-135, 1972.

[42] Harsha, P. T., Glassman, H. N., "Analysis of Turbulent Unseparated Flow in Subsonic Diffusers," presented at the ASME Gas Turbine and Fluids Engrg. Conf., New Orleans, La., Mar. 21-25, 1976.

[44] Coles, D. E., Hirst, E. A., editors, Computation of Turbulent Boundary Layers - 1968, AFOSR-IFP-Stanford Conference, Vol. II, Thermosciences Div., M. E. Dept., Stanford Univ., 1969.

[45] Reneau, L. R., Johnston, J. P., Kline, S. J., "Performance and Design of Straight Two-Dimensional Diffusers," Rept. PD-8, Thermosciences Div., M. E. Dept., Stanford Univ., Sept. 1964.

[46] So, R. M., Mellor, G. L., "Experiment on Convex Curvature Effects in Turbulent Boundary Layers," J. Fluid Mech., Vol. 60, Part 1, pp. 43-62, 1973.

[47] Bradshaw, P., "Effect of Streamline Curvature on Turbulent Flow," AGARDograph No. 169, Aug. 1973.

[48] Cline, A. K., "Curve-Fitting Using Splines in Tension," Atmospheric Tech., NCAR, No. 3, Sept. 1973, pp. 60-65.

[49] Bradshaw, P., Ferriss, D. H., "The Response of a Retarded Equilibrium Boundary Layer to the Sudden Removal of Pressure Gradient," NPL Aero Rept. 1145, 1965.

[50] Gerhart, P. M., "On Prediction of Separated Boundary Layers with Pressure Distribution Specified," AIAA J., pp. 1278-1279, Sept. 1974.

[51] Sandborn, V. A., Kline, S. J., "Flow Models in Boundary Layer Stall Inception," J. Basic Engrg., Vol. 83, No. 3, pp. 317-327, Sept. 1961.

FIGURES

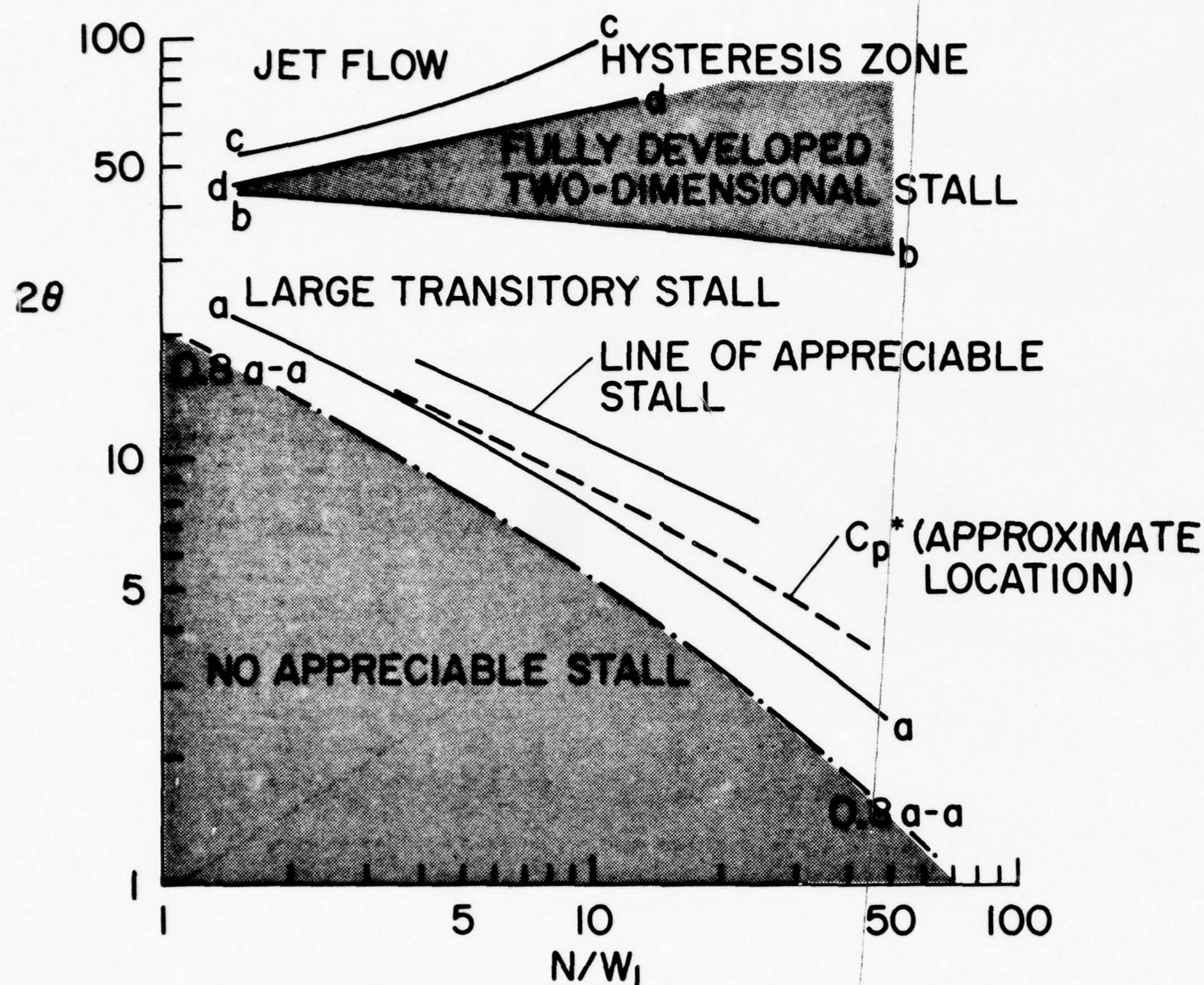


Fig. 1. Straight-walled diffuser flow-regime chart of Fox and Kline [1].

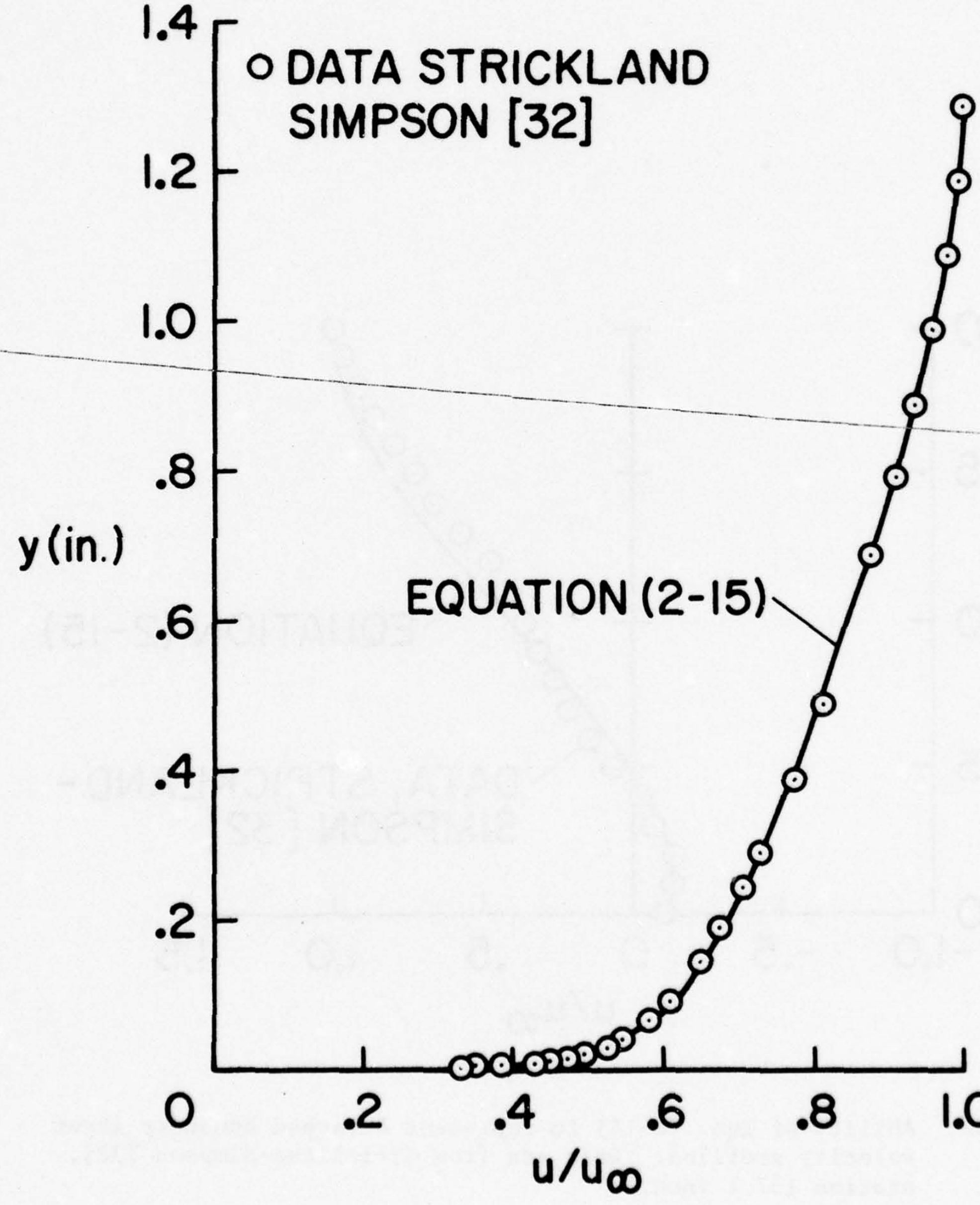


Fig. 2. Ability of Eqn. (2-15) to represent attached boundary layer velocity profiles. Data are from Strickland-Simpson [32], station 88.2 inch.

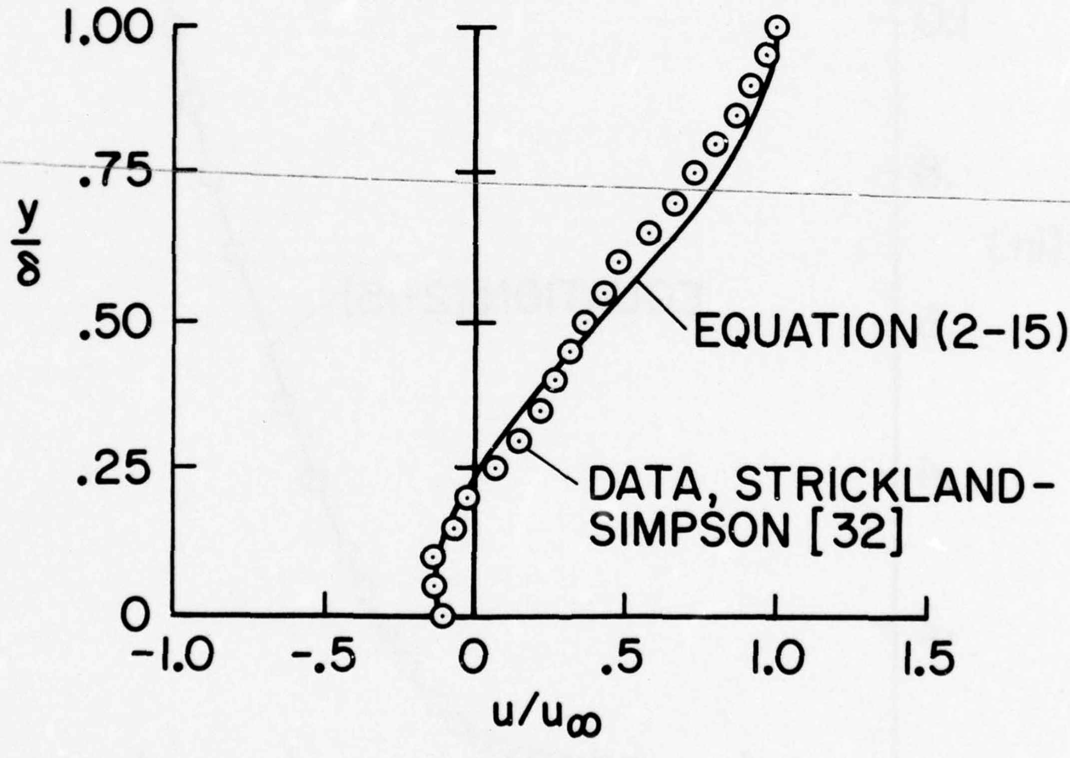


Fig. 3a. Ability of Eqn. (2-15) to represent detached boundary layer velocity profiles. Data are from Strickland-Simpson [32], station 157.1 inch.

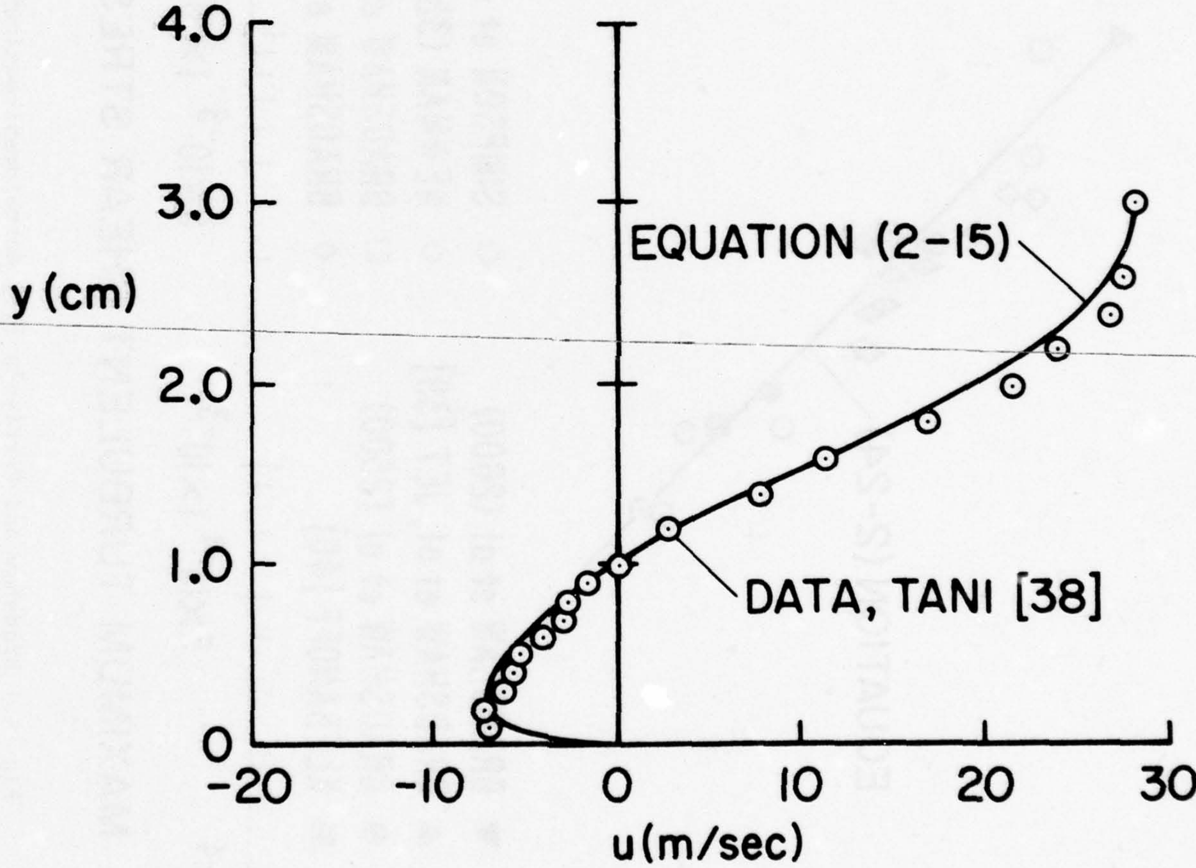


Fig. 3b. Ability of Eqn. (2-15) to represent detached boundary layer velocity profiles. Data are from Tani's [38] flow over a backward-facing step.

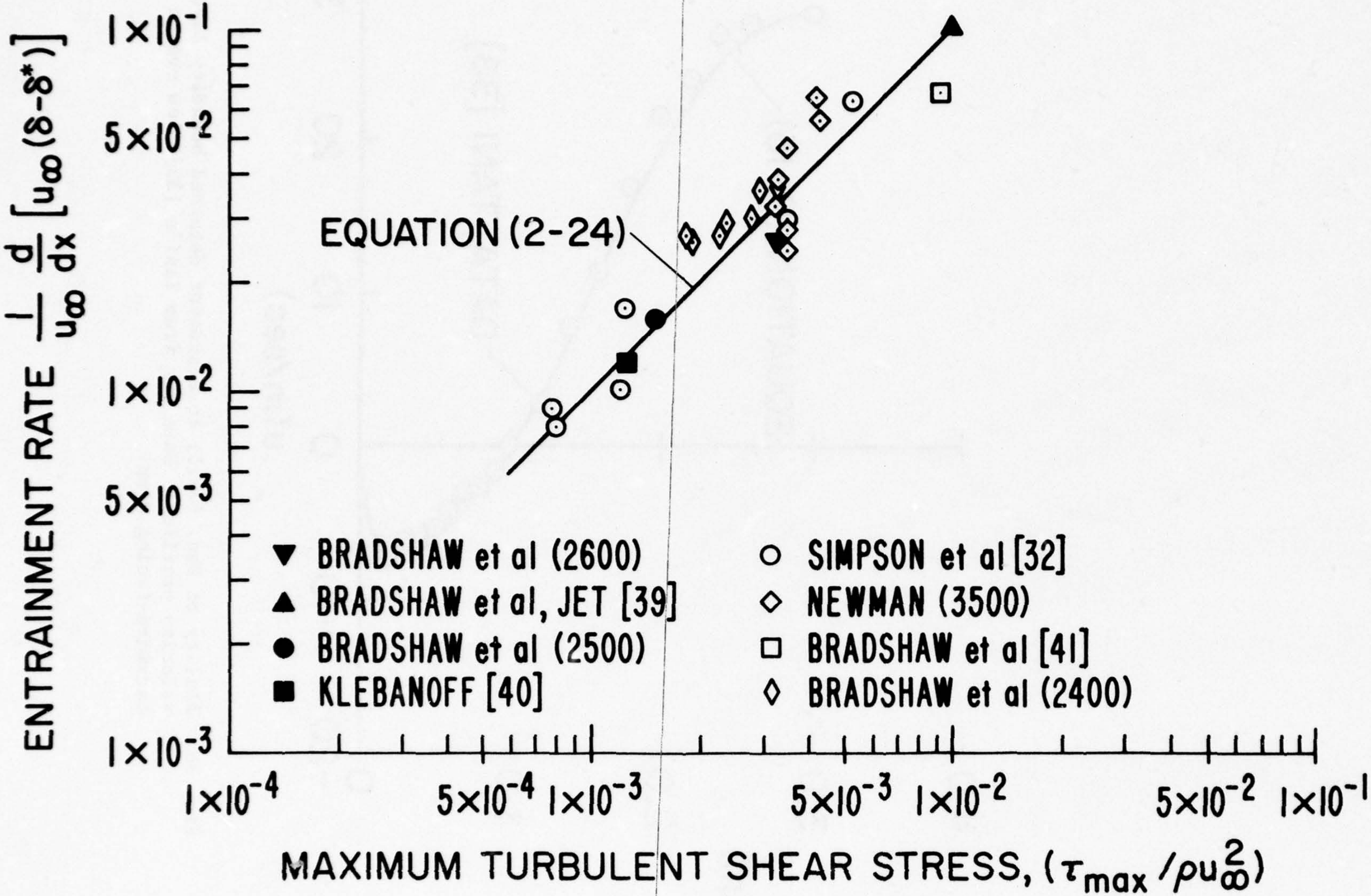


Fig. 4. Bradshaw and Ferriss's [21] entrainment-maximum shear correlation.

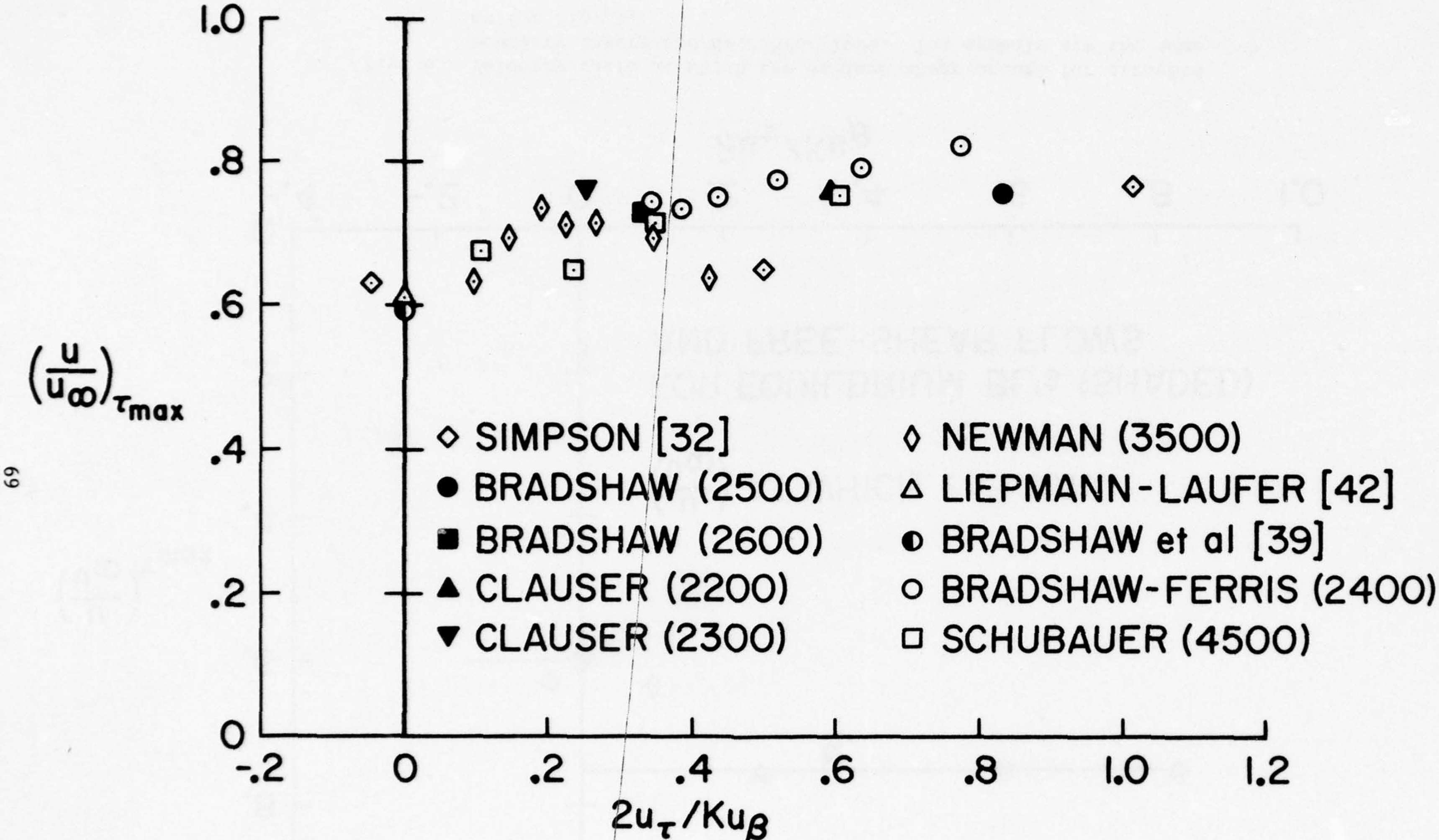


Fig. 5. Velocity ratio at which the maximum shear stress occurs for attached and detached flows.

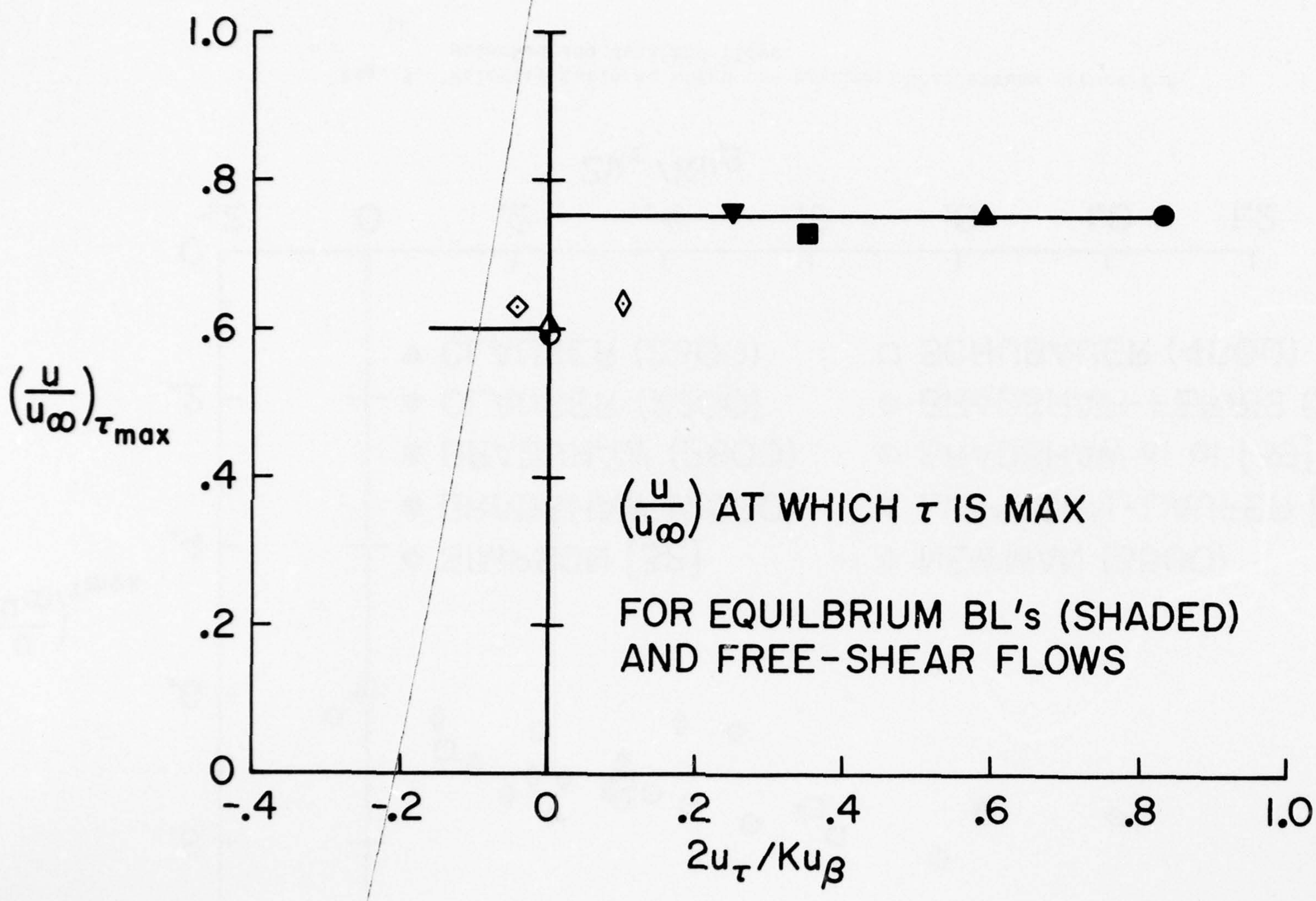


Fig. 6. Velocity ratio at which the maximum shear occurs for attached boundary layers and detached flows. The symbols are the same as for Fig. 5.

Fig. 7. Entrainment Equation Summary

Entrainment Equation

$$\left(1 - \frac{\delta^*}{\delta}\right)\left\{\frac{d\delta}{dx}\right\} + \left(\frac{-\delta}{2u_\infty}\right)\left\{\frac{du_\beta}{dx}\right\} + \left(\frac{-\delta}{\kappa u_\infty}\right)\left\{\frac{du_\tau}{dx}\right\}$$

$$+ \left(\frac{\delta}{u_\infty}\right)\left\{\frac{du_\infty}{dx}\right\} = \frac{10\ \tau_{max}/\rho}{u_\infty^2} .$$

Lag Equation

$$\frac{d}{dx}\left(\tau_{max}/\rho\right) = \frac{\lambda_a}{\delta}\left(\tau_{max,eq}/\rho - \tau_{max}/\rho\right)$$

Equilibrium Maximum Shear

$$\frac{\tau_{max,eq}}{\rho} = \kappa_e u_\infty \delta^* \left(\frac{\partial u}{\partial y}\right)_{\tau_{max}} ,$$

where $\kappa_e = .013 + .0038\ e^{-\beta/15}$

and $\beta = \frac{\delta^*}{\tau_\omega}\frac{dp}{dx}$.

$$\left(\frac{\partial u}{\partial y}\right)_{\tau_{max}} = \frac{u_\infty}{\delta}\left(\frac{V_T}{\eta_{\tau_{max}}} + \frac{V_B \pi}{2} \sin \pi \eta_{\tau_{max}}\right) ,$$

where $\eta_{\tau_{max}}$ is the solution of:

$$f\left(\eta_{\tau_{max}}\right) = V_T \ \ell n\ \eta_{\tau_{max}} - V_B \cos^2 \frac{\pi}{2}\eta_{\tau_{max}} + 1 - \left(\frac{u}{u_\infty}\right)_{\tau_{max}} = 0 ,$$

$$\left(\frac{u}{u_\infty}\right)_{\tau_{max}} = \begin{Bmatrix} 0.76 & \text{attached flows} \\ 0.60 & \text{detached flows} \end{Bmatrix} \text{correlation} .$$

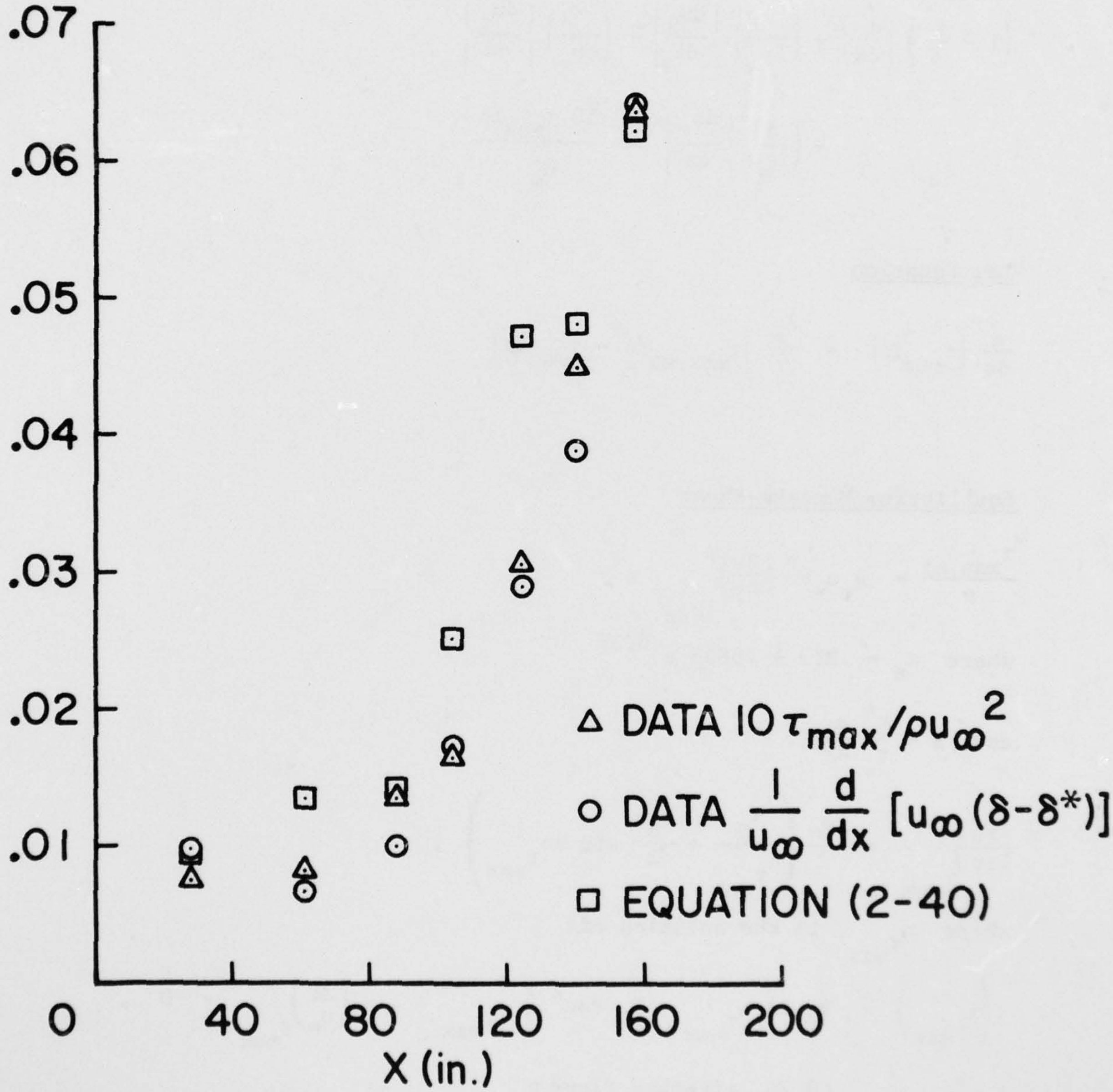


Fig. 8. Comparison of $\tau_{max}/\rho U_\infty^2$ and entrainment rate data with that obtained from Eqn. (2-40). Data are from Strickland-Simpson [32]. Intermittent detachment is at $x = 127$ in and $\tau_\omega = 0$ at $x = 132$ in.

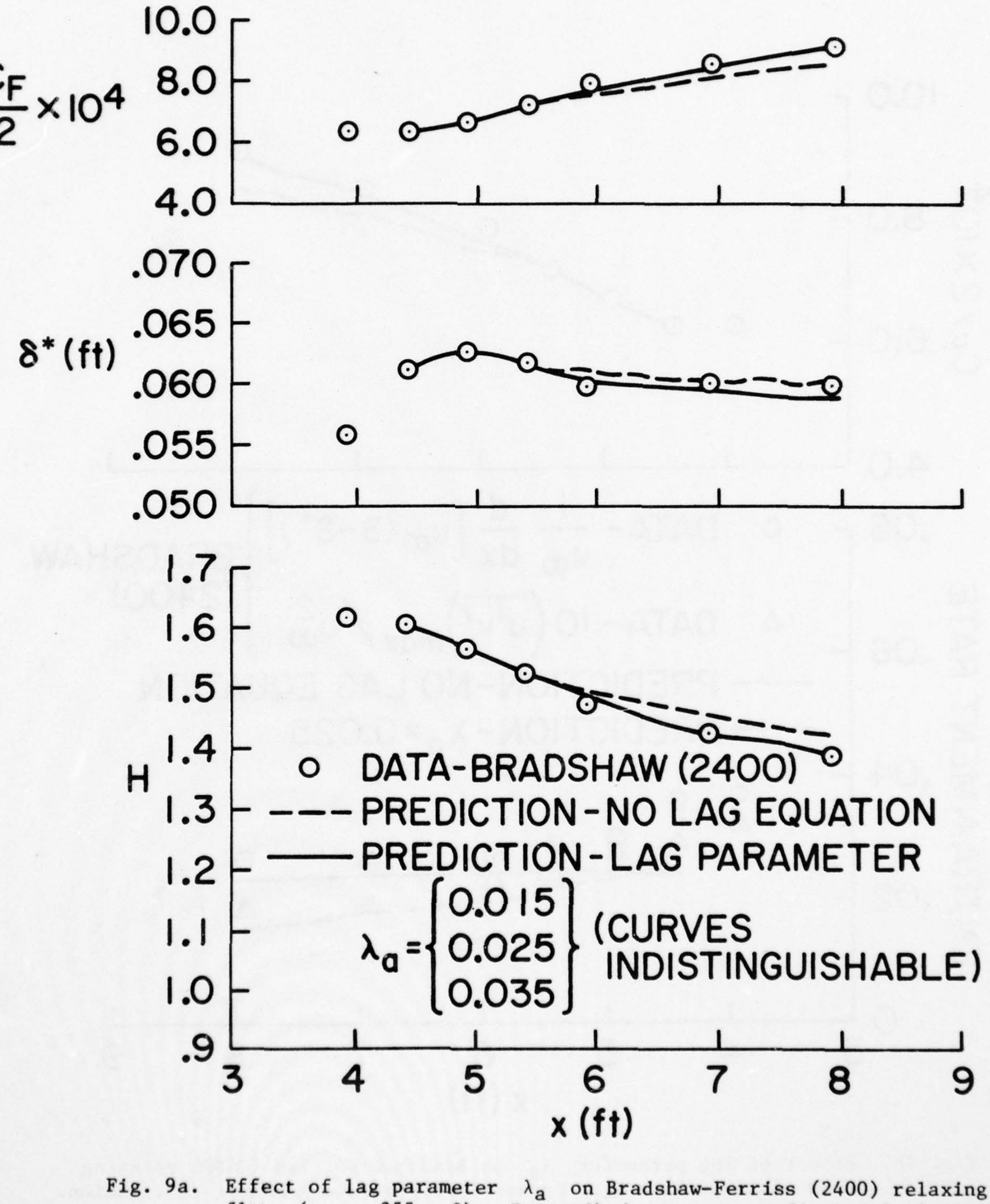


Fig. 9a. Effect of lag parameter λ_a on Bradshaw-Ferriss (2400) relaxing flow ($a = -.255 \rightarrow 0$). Prescribed pressure gradient calculation. Mean boundary layer parameters.

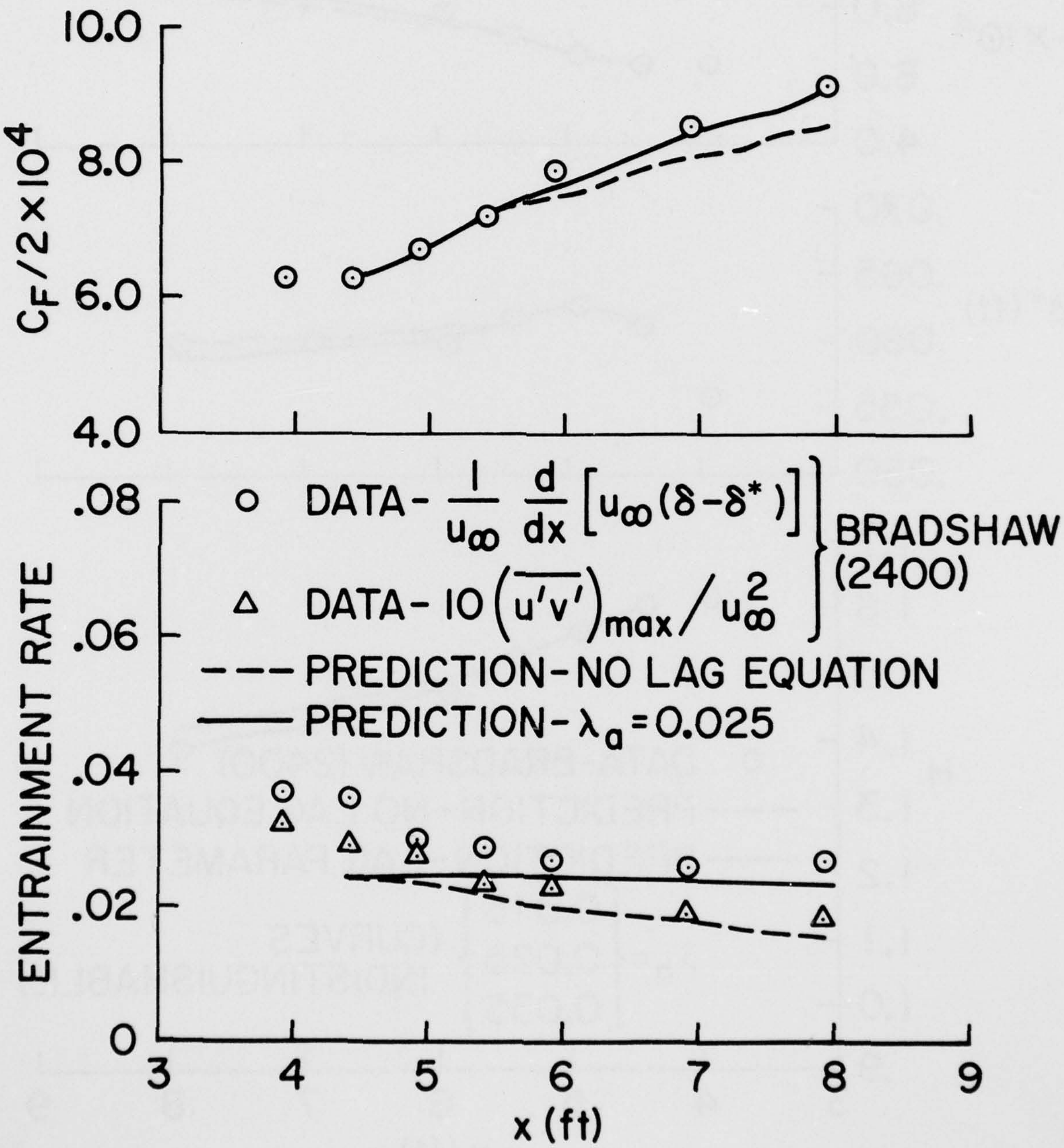


Fig. 9b. Effect of lag parameter λ_a on Bradshaw-Ferriss (2400) relaxing flow ($a = -.255 \rightarrow 0$). Presecribed pressure gradient calculation. Skin friction and entrainment.

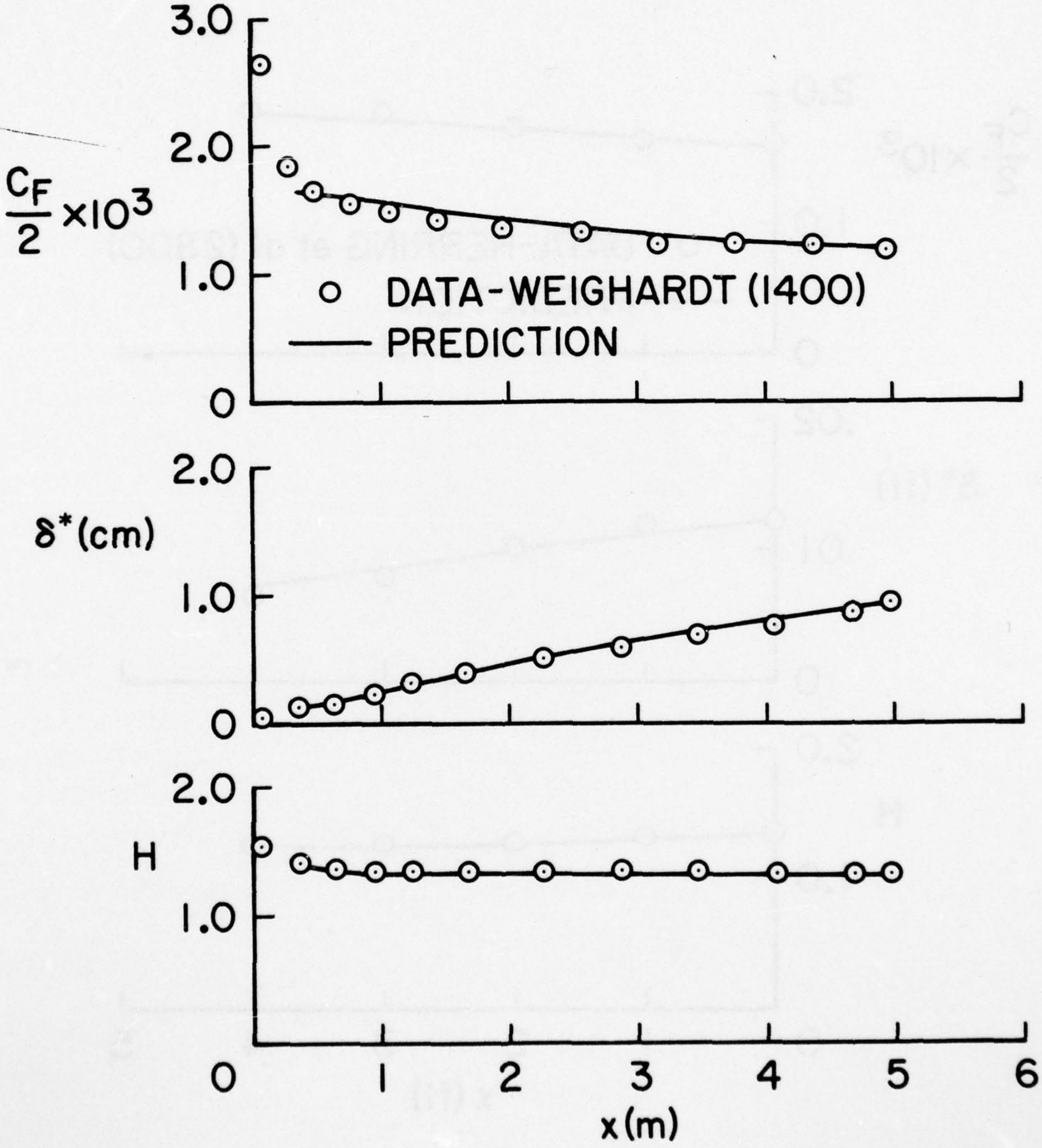


Fig. 10. Results -- Weighardt's flat plate flow (1400). Prescribed pressure gradient calculation.

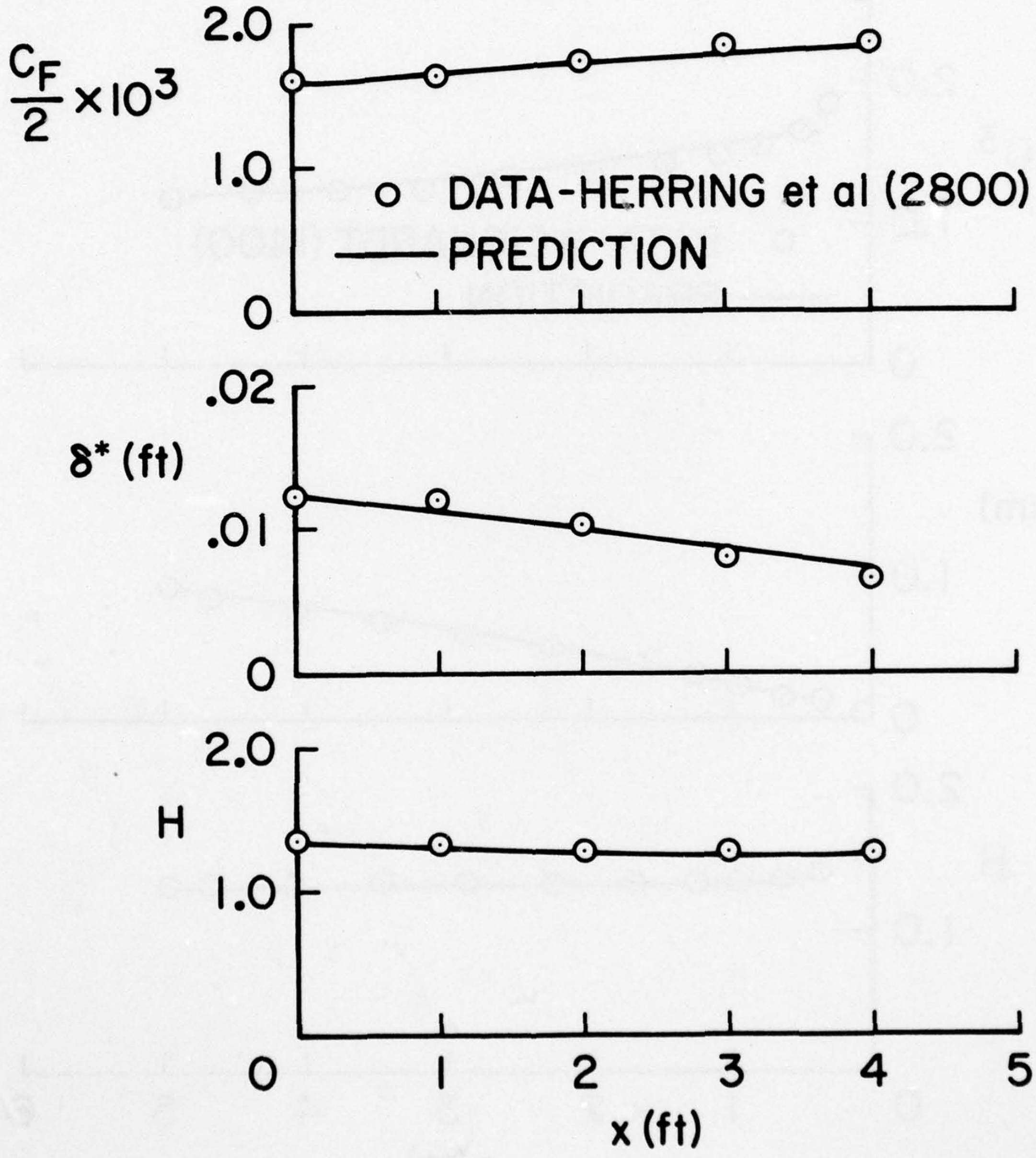


Fig. 11. Results -- Herring-Norbury (2800) equilibrium flow ($\beta = -.53$) in strong negative pressure gradient. Prescribed pressure gradient calculation.

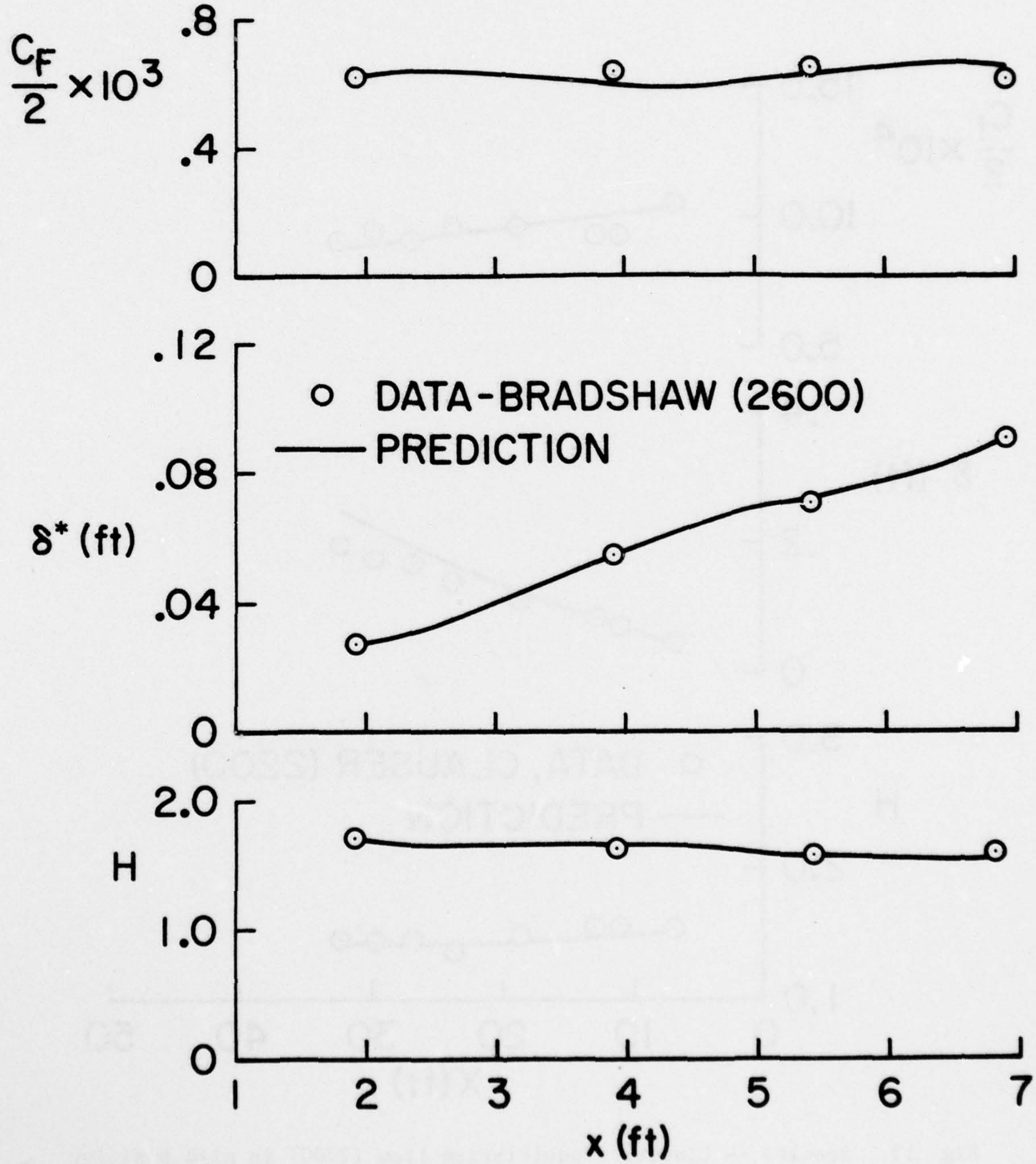


Fig. 12. Results -- Bradshaw-Ferriss (2600) equilibrium flow (a = -.255). Prescribed pressure gradient calculation.

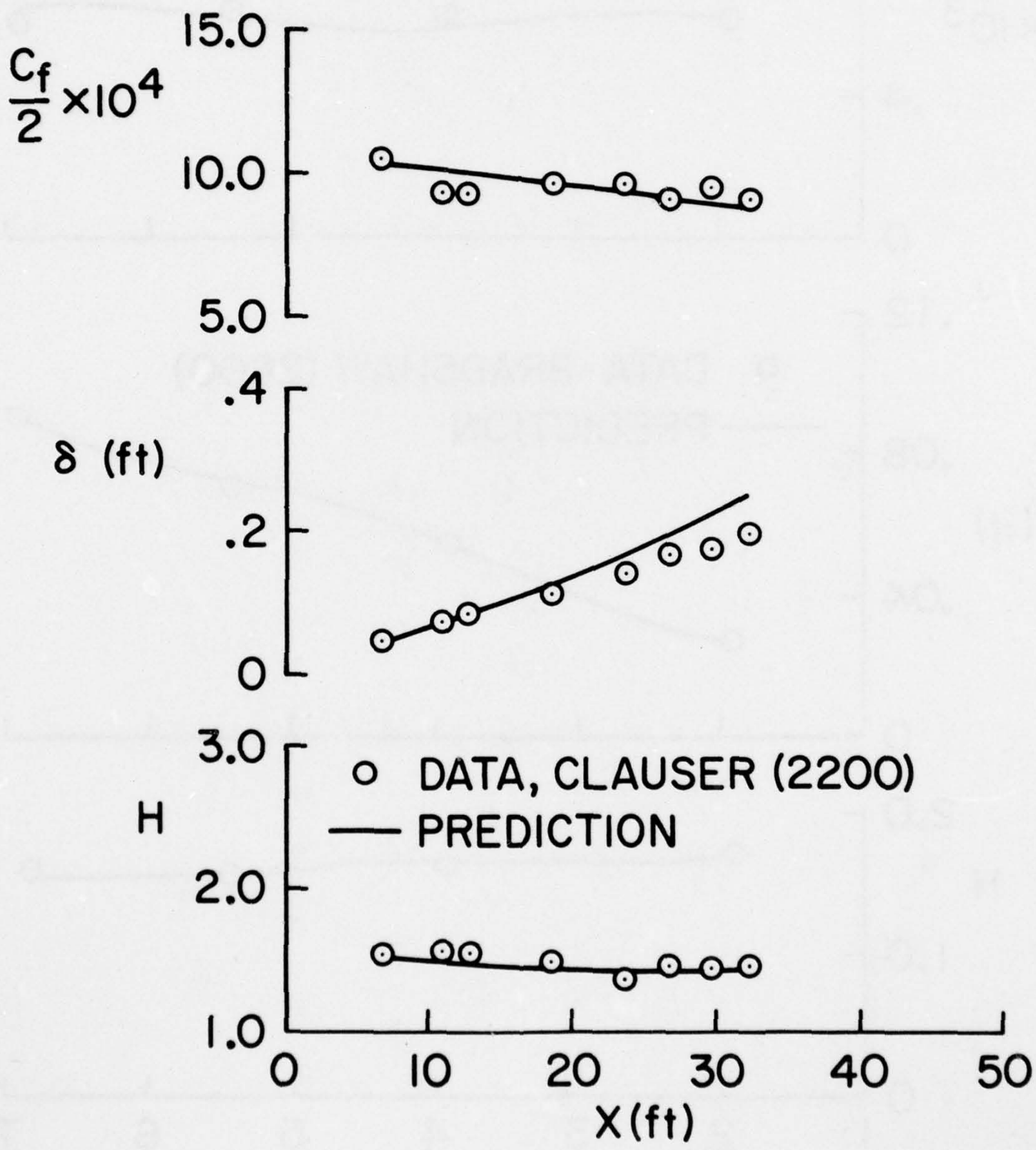


Fig. 13. Results -- Clauser's equilibrium flow (2200) in mild positive pressure gradient. Prescribed pressure gradient calculation.

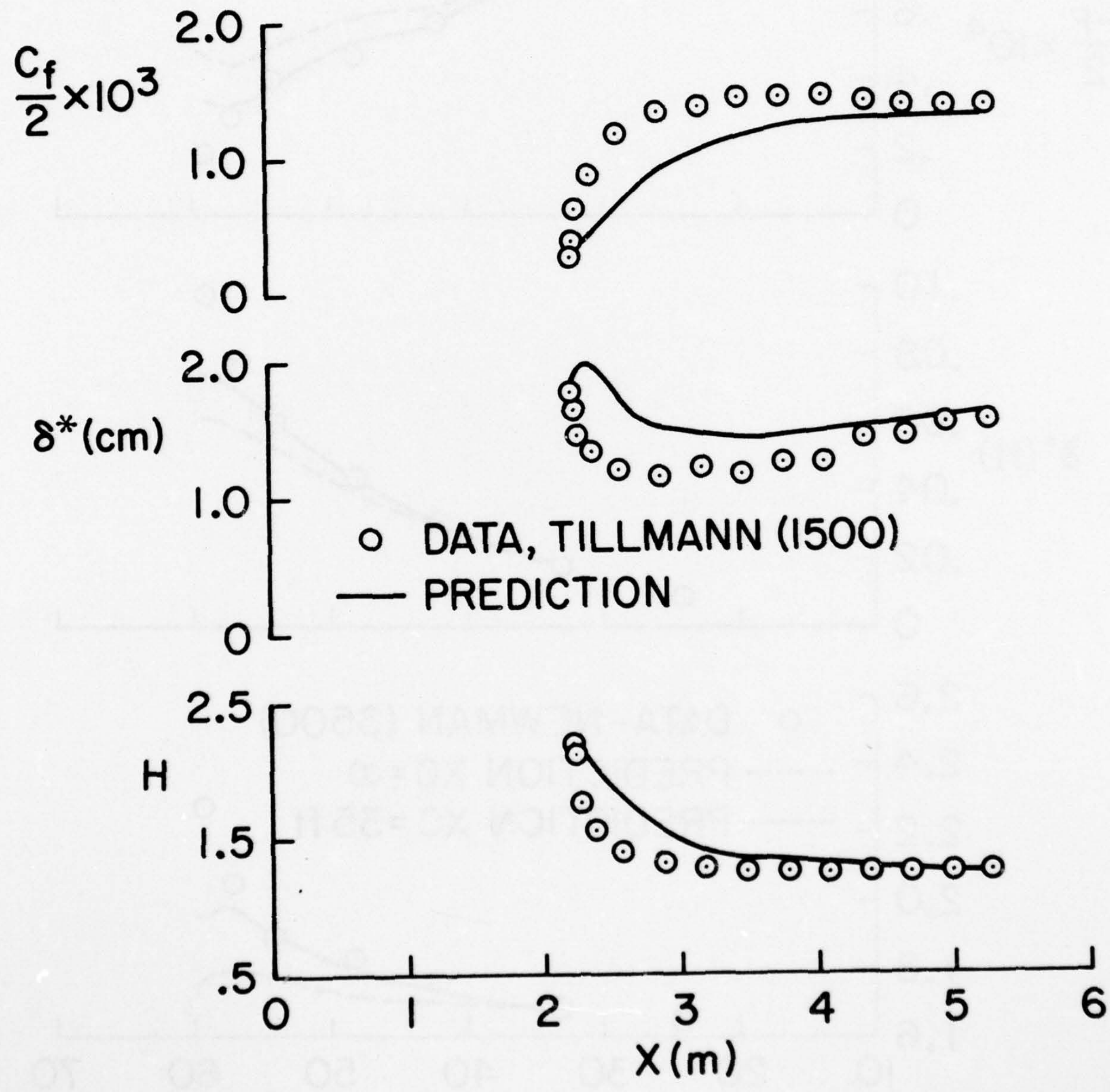


Fig. 14. Tillmann ledge flow (1500). Results for prescribed pressure gradient calculation.

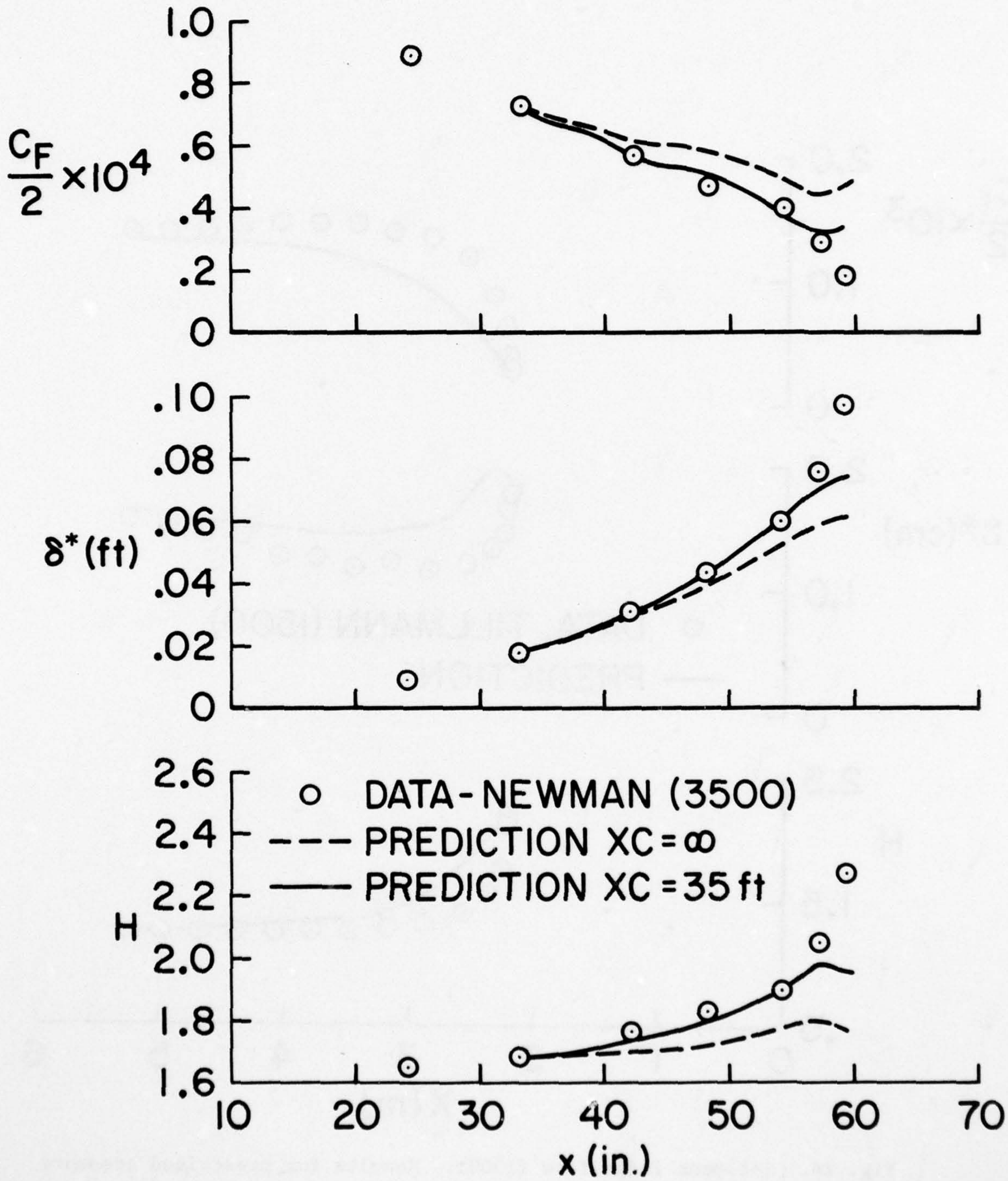


Fig. 15. Results -- Newman airfoil flow (3500). Prescribed pressure gradient calculation.

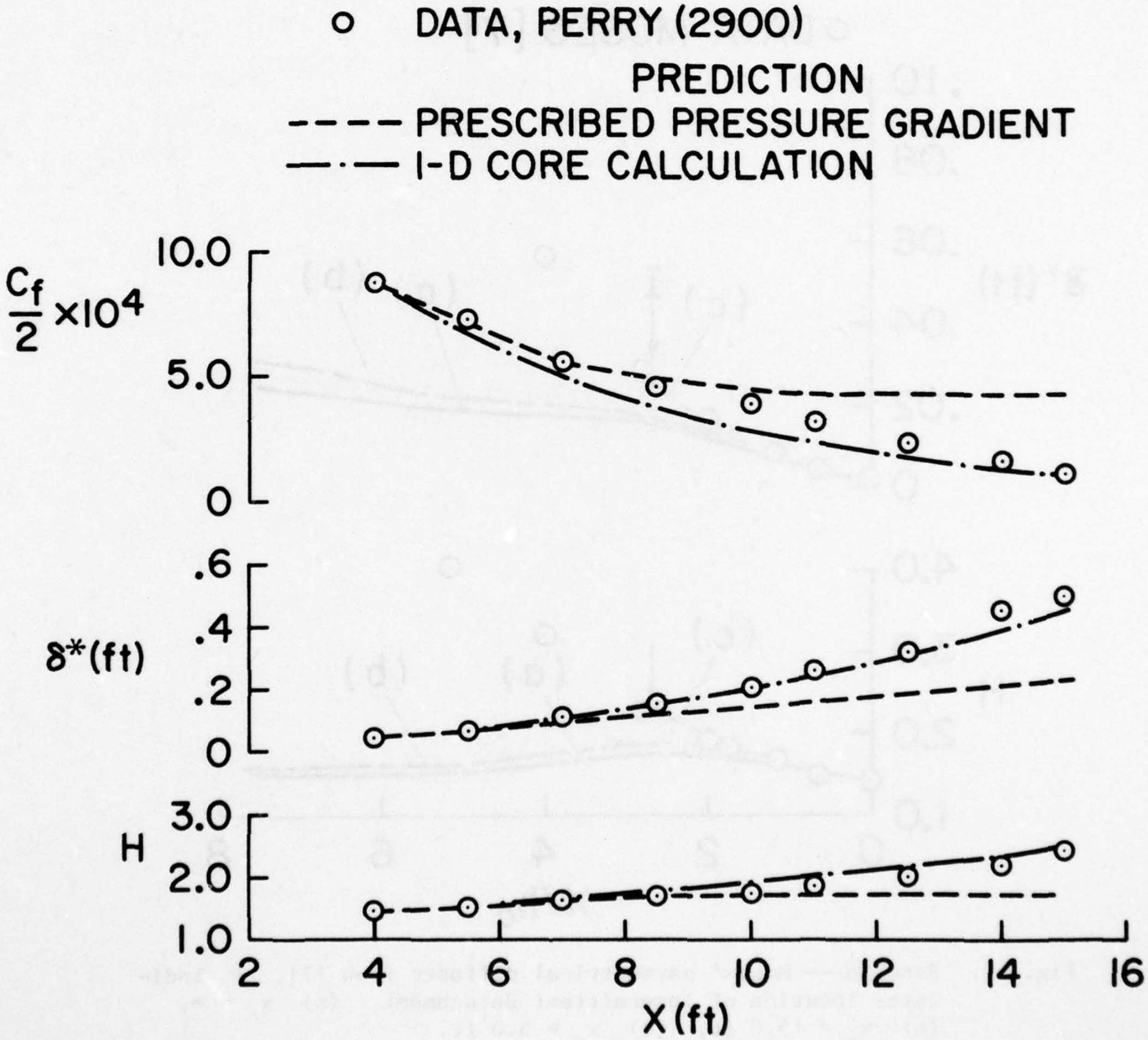


Fig. 16. Results -- Perry diffuser flow (2900) showing comparison between prescribed pressure gradient and the 1-D core diffuser calculation.

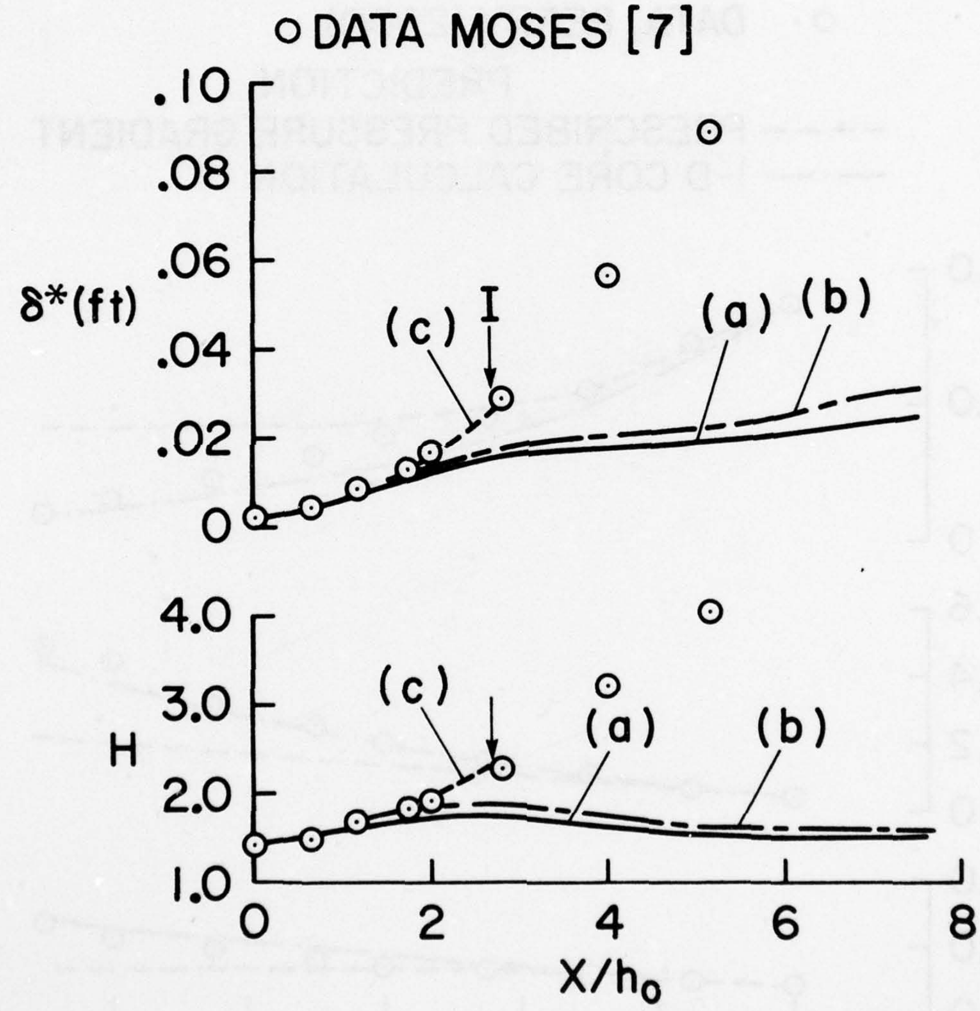


Fig. 17. Results -- Moses' asymmetrical diffuser flow [7]. I indicates location of intermittent detachment. (a) $x_c = \infty$, (b) $x_c = 15.0$ ft, (c) $x_c = 5.0$ ft.

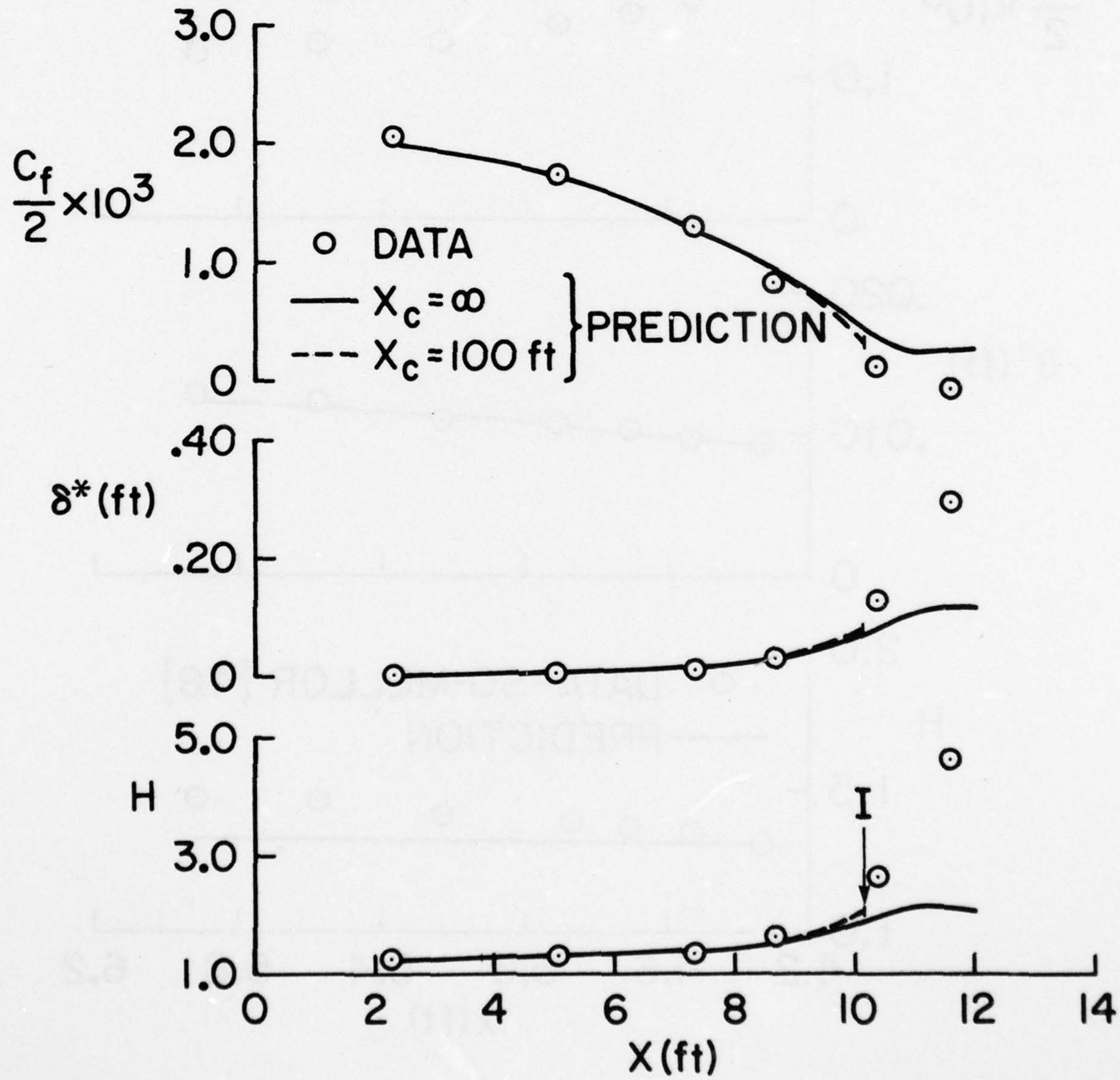


Fig. 18. Strickland-Simpson flow (lower wall) as calculated with prescribed pressure gradient. (a) $x_c = \infty$, (b) $x_c = 100$ ft. I indicates location of intermittent detachment.

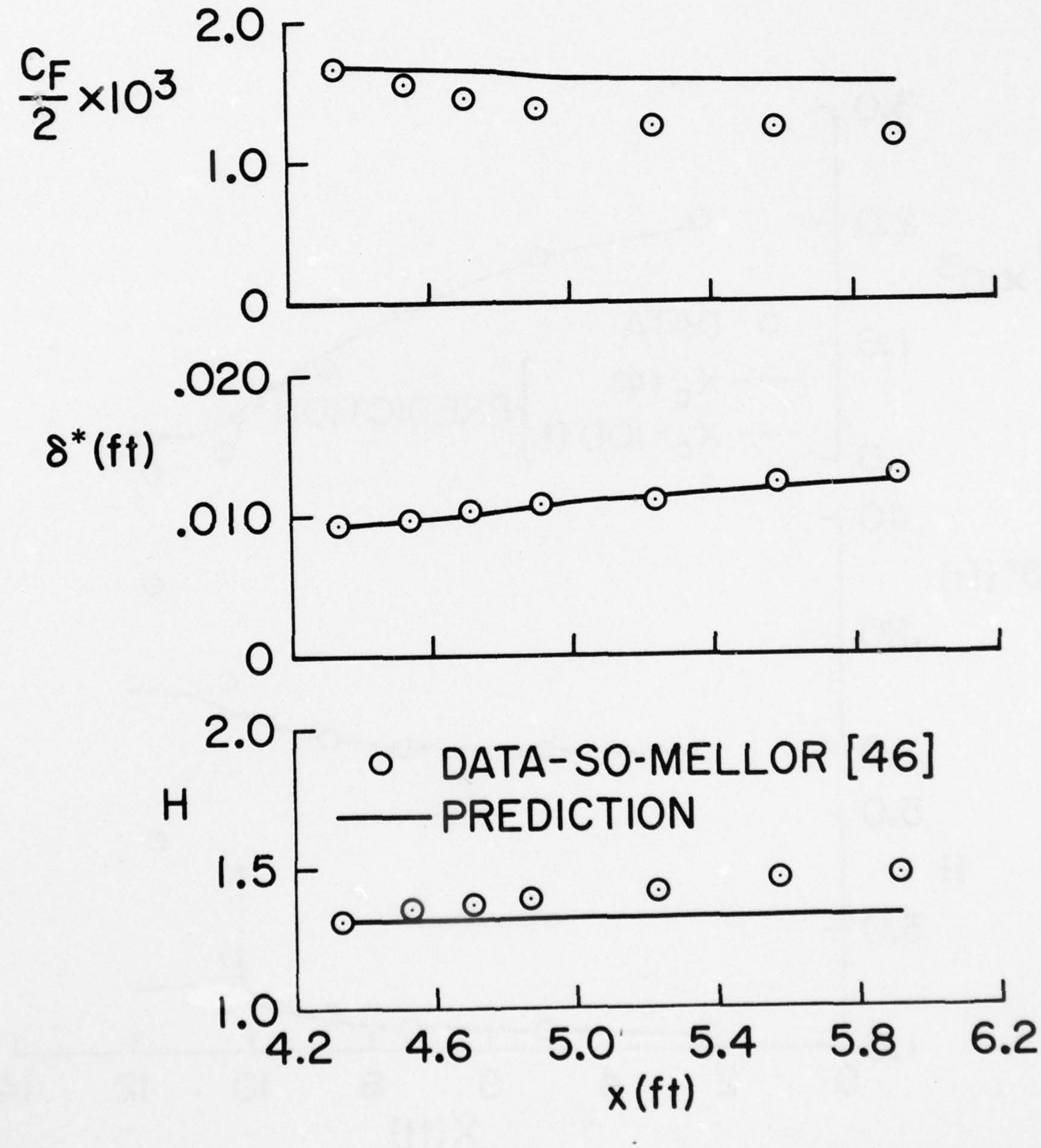
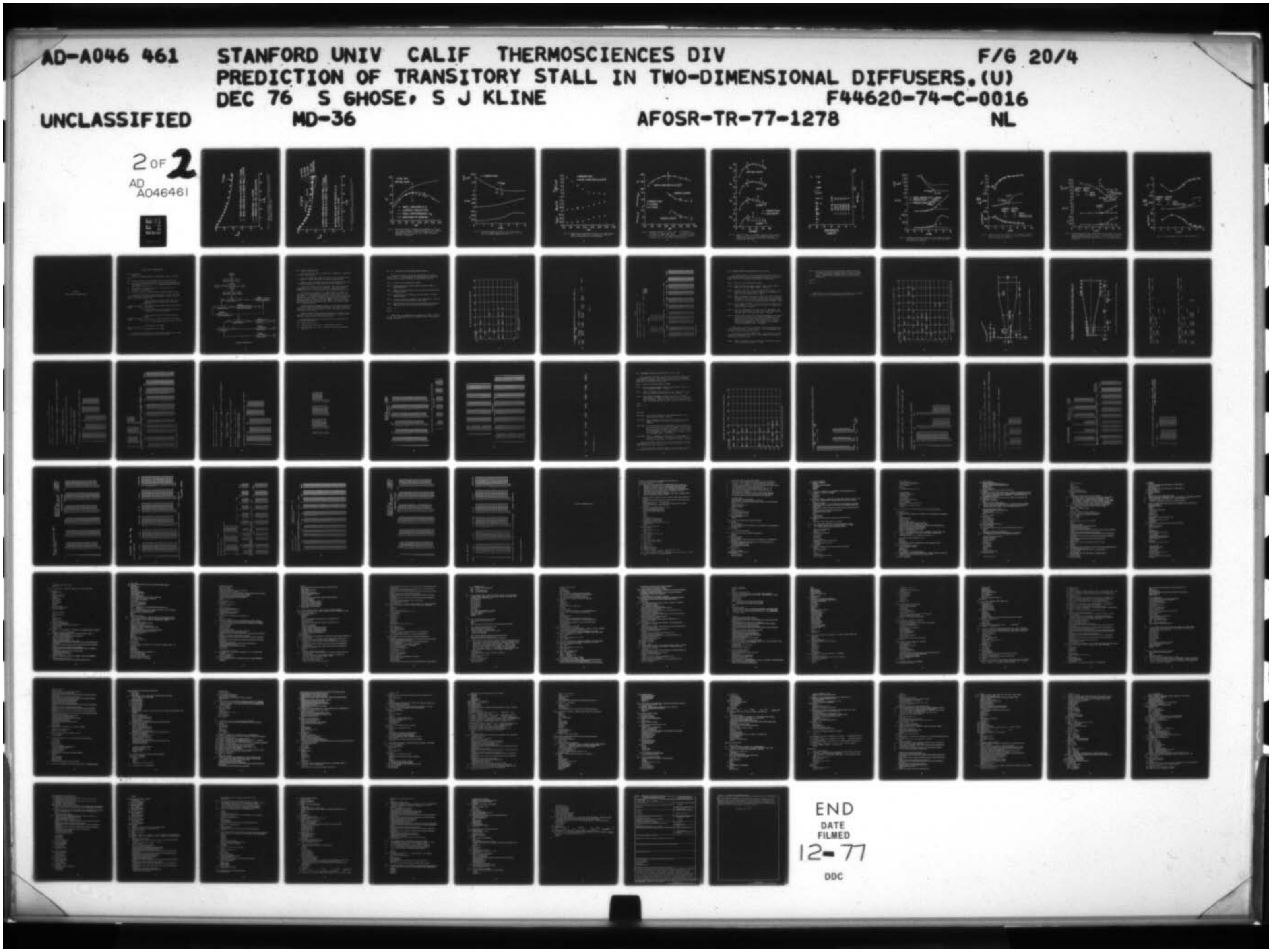
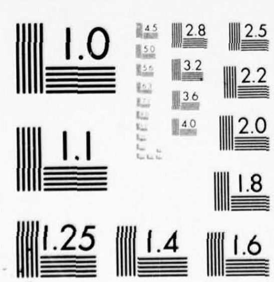


Fig. 19. Results -- So-Mellor's [46] convex wall boundary layer as calculated with prescribed pressure gradient.


AD-A046 461 STANFORD UNIV CALIF THERMOSCIENCES DIV F/G 20/4
PREDICTION OF TRANSITORY STALL IN TWO-DIMENSIONAL DIFFUSERS.(U)
DEC 76 S GHOSE, S J KLINE F44620-74-C-0016
UNCLASSIFIED MD-36 AFOSR-TR-77-1278 NL
2 OF 2
AD A046461
END
DATE FILMED
12-77
DDC

2 OF 2

AD
A046461



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963

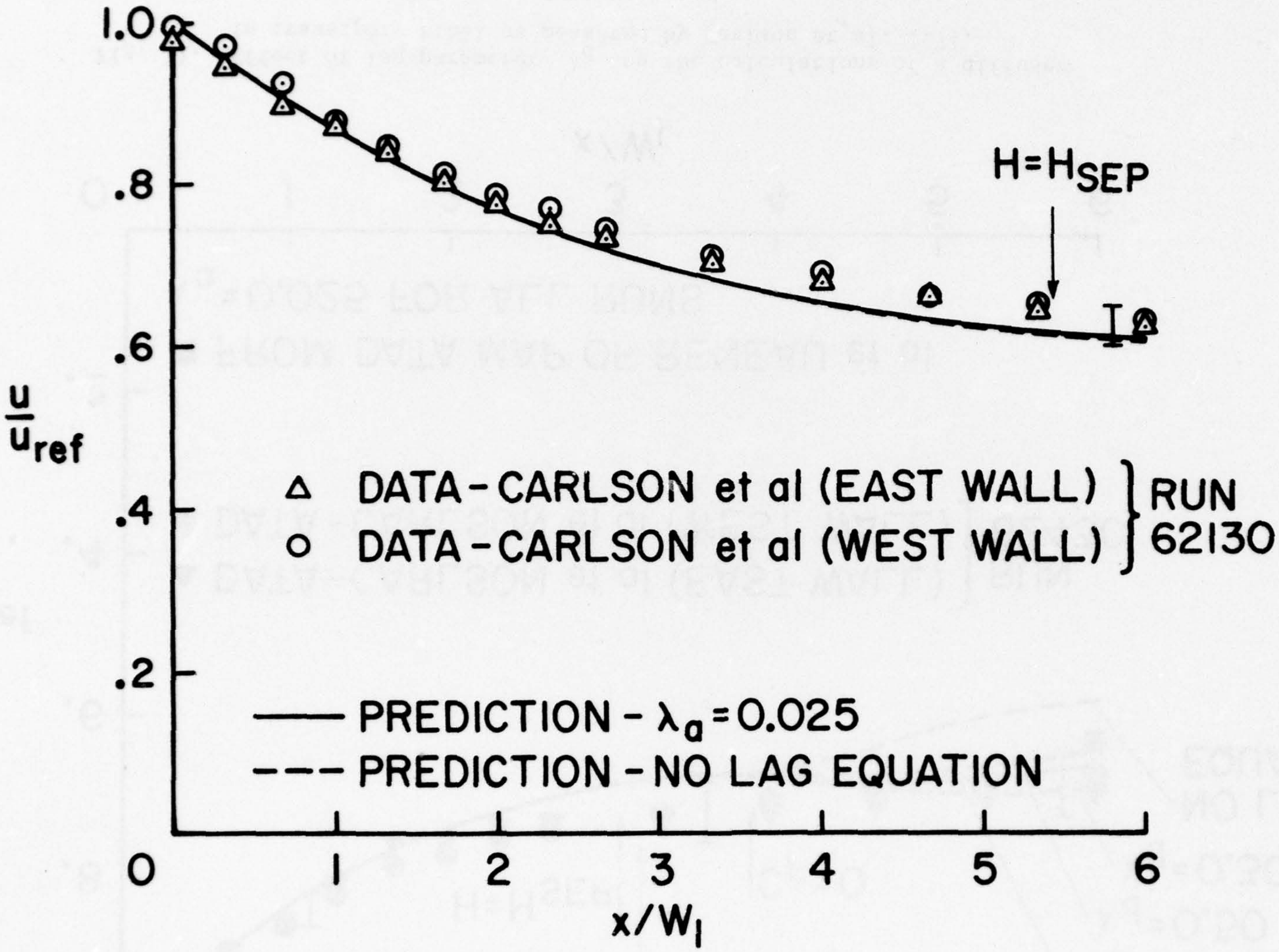


Fig. 20. Effect of lag parameter λ_a on the predictions of the unstalled diffuser flow of Carlson et al. [27]. Calculation is 1-D core with symmetric boundary layers.

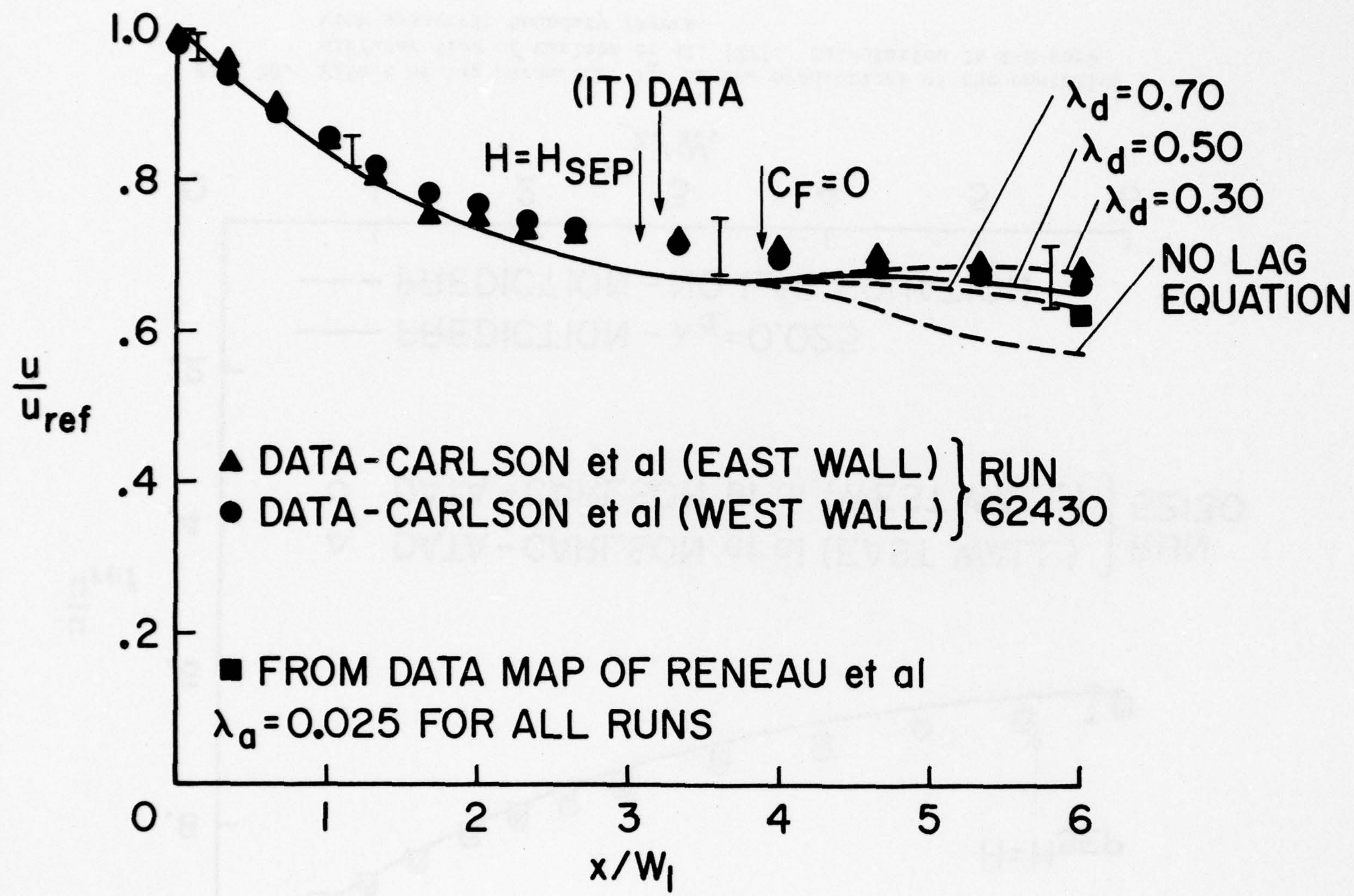


Fig. 21. Effect of lag parameter λ_d on the calculations of a diffuser in transitory stall as measured by Carlson et al. [27].

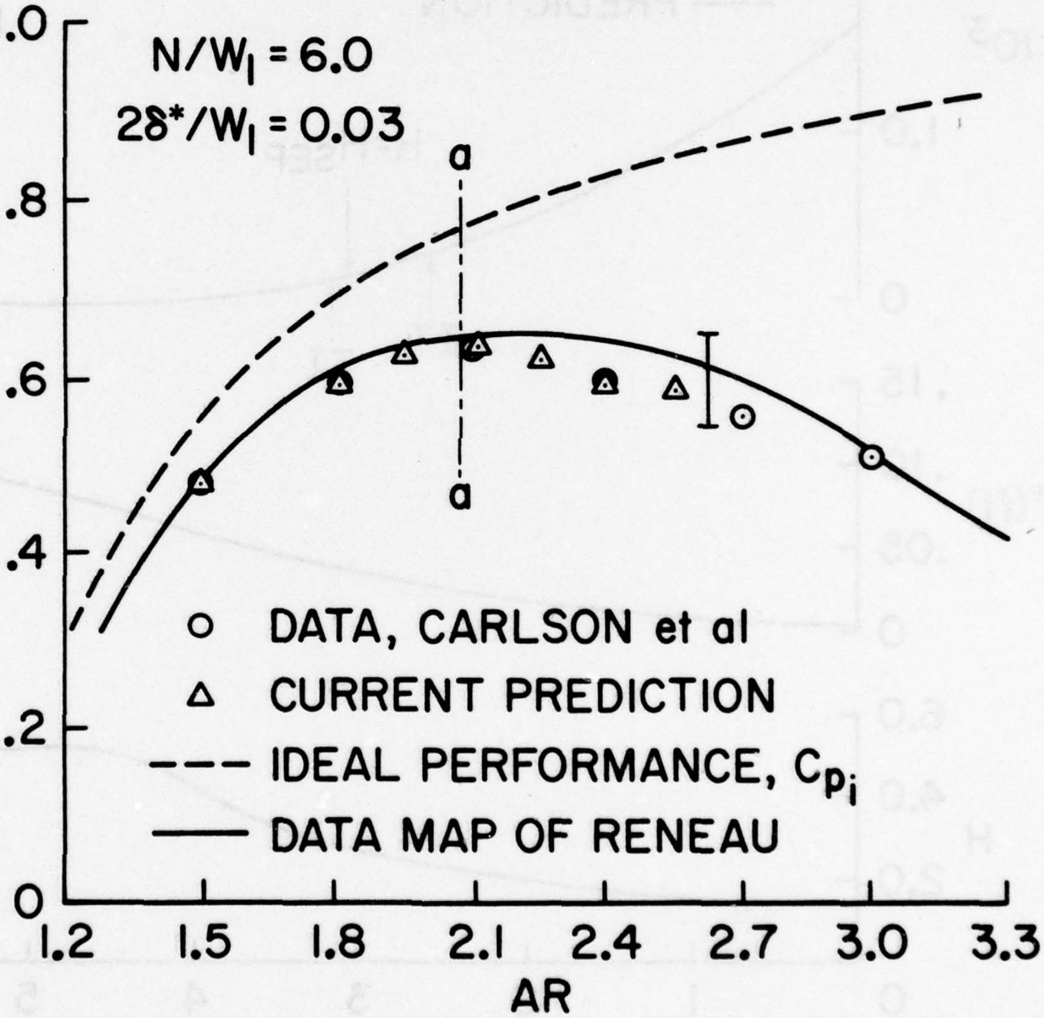


Fig. 22. Predicted variation of boundary layer quantities H, δ^* and $C_f/2$ along the walls of a diffuser operating in the transitory stall regime. Diffuser is same as from Carlson et al. [27], Run 62430, $N/W_1 = 6$, $AR = 2.4$, $B_1 = .030$. No data are available for comparison.

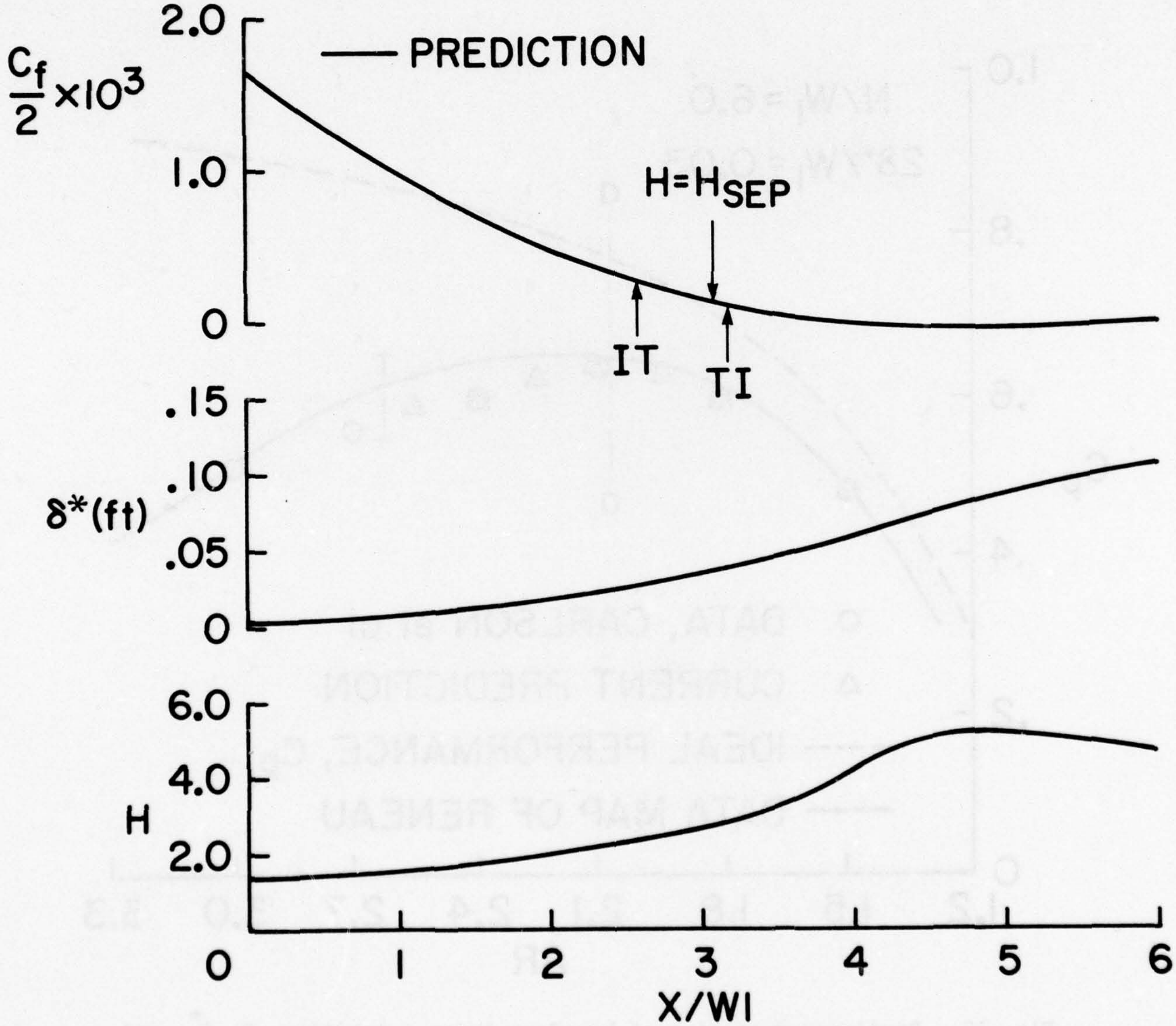


Fig. 23. Predicted performance of $N/W_1 = 6$, $B_1 = .030$ diffuser family, as compared against the data of Carlson et al. [27], and the data maps of Reneau et al. [45].

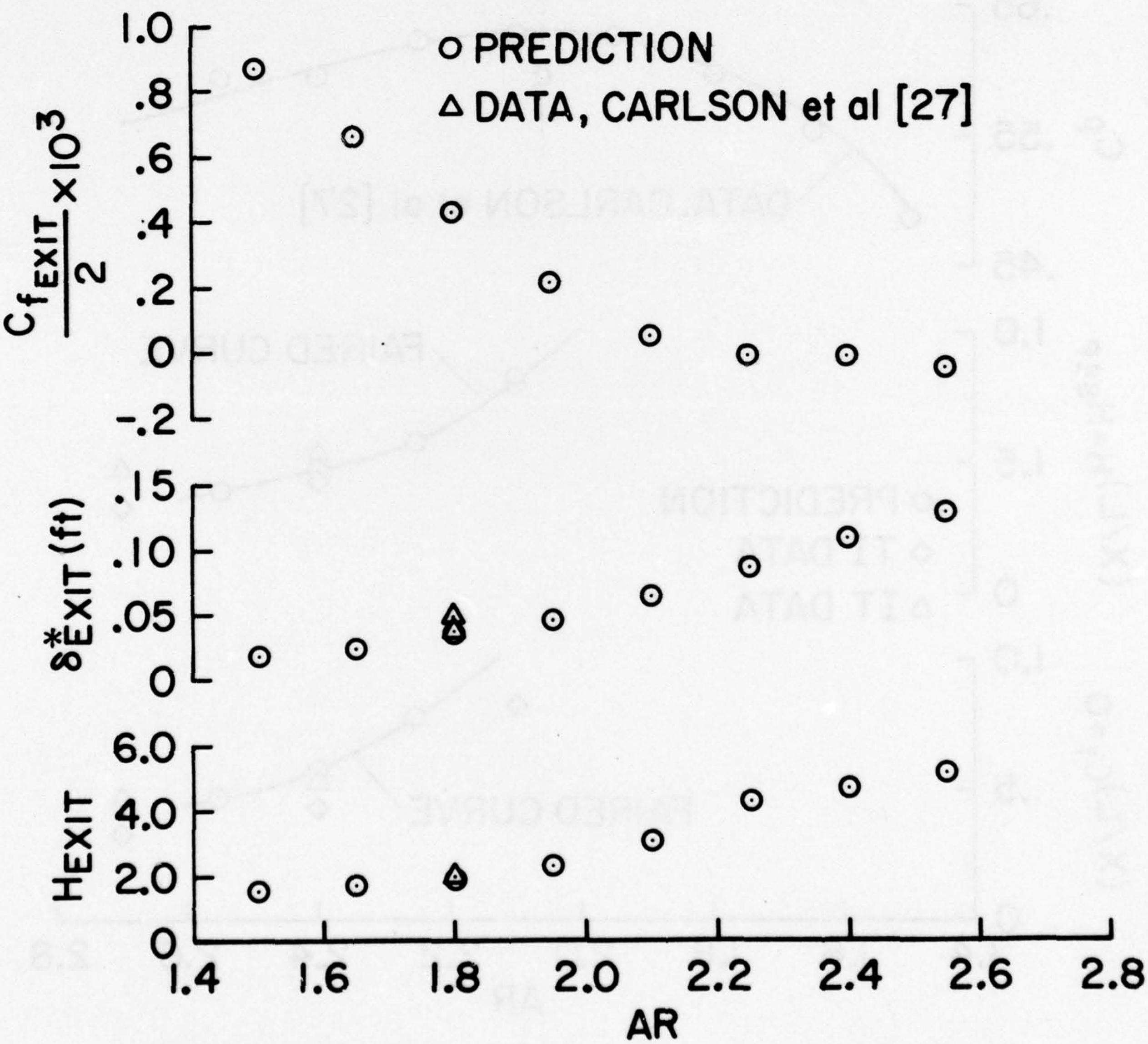


Fig. 24. Predicted exit conditions for $N/W_1 = 6$, $B_1 = .030$ diffuser family. Only one data point is available for comparison. Data are from Carlson et al. [27], Run 62430.

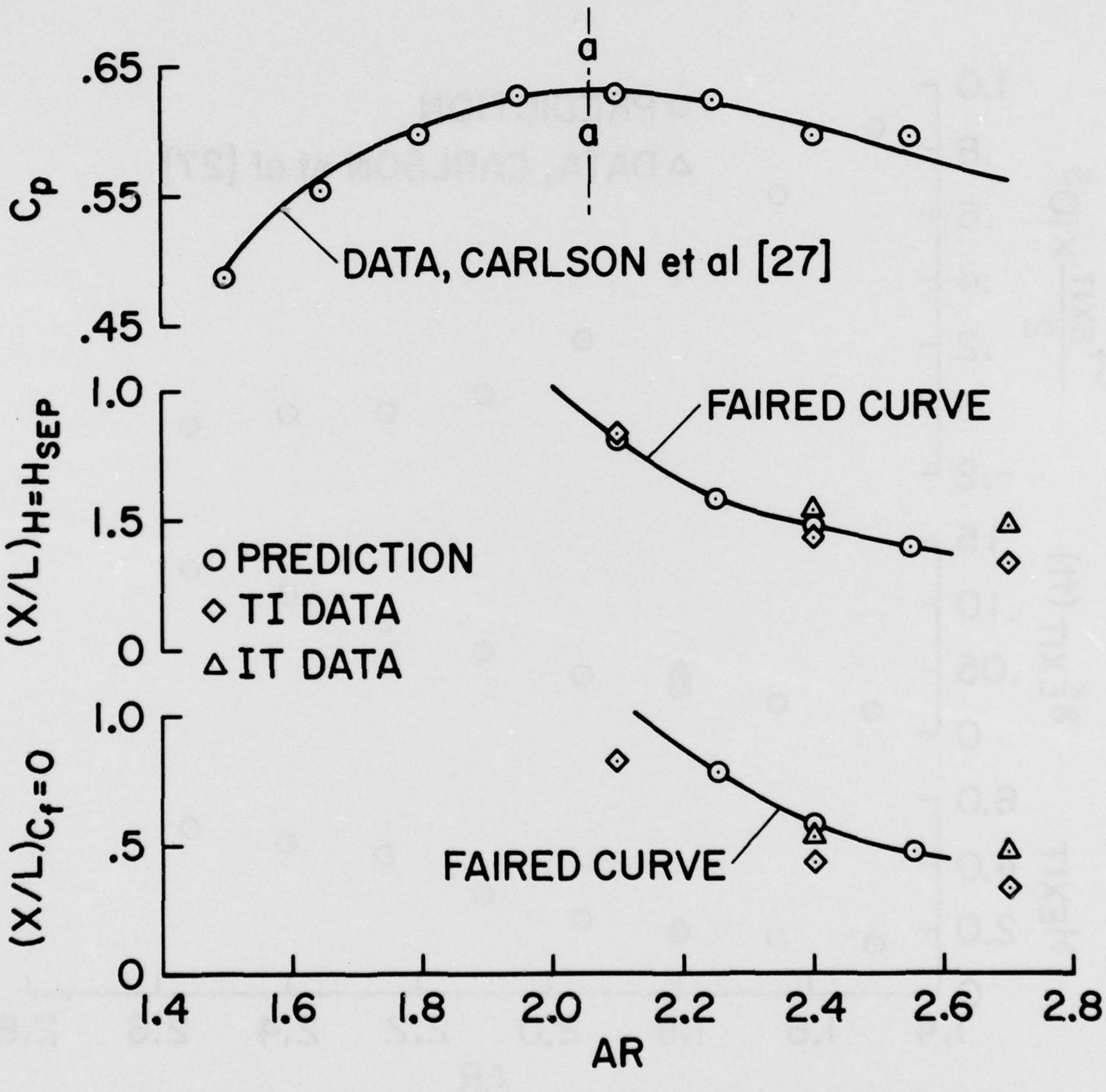


Fig. 25. Predicted variation of exit C_p, location of intermittent detachment ($H = H_{sep}$), and zero wall shear ($C_f = 0$) location as fraction of length (X/L). Data are from Carlson et al. [27] for $N/W_1 = 6$, $B_1 = .025$ family of diffusers. TI - incipient transitory stall. IT - intermittent transitory stall.

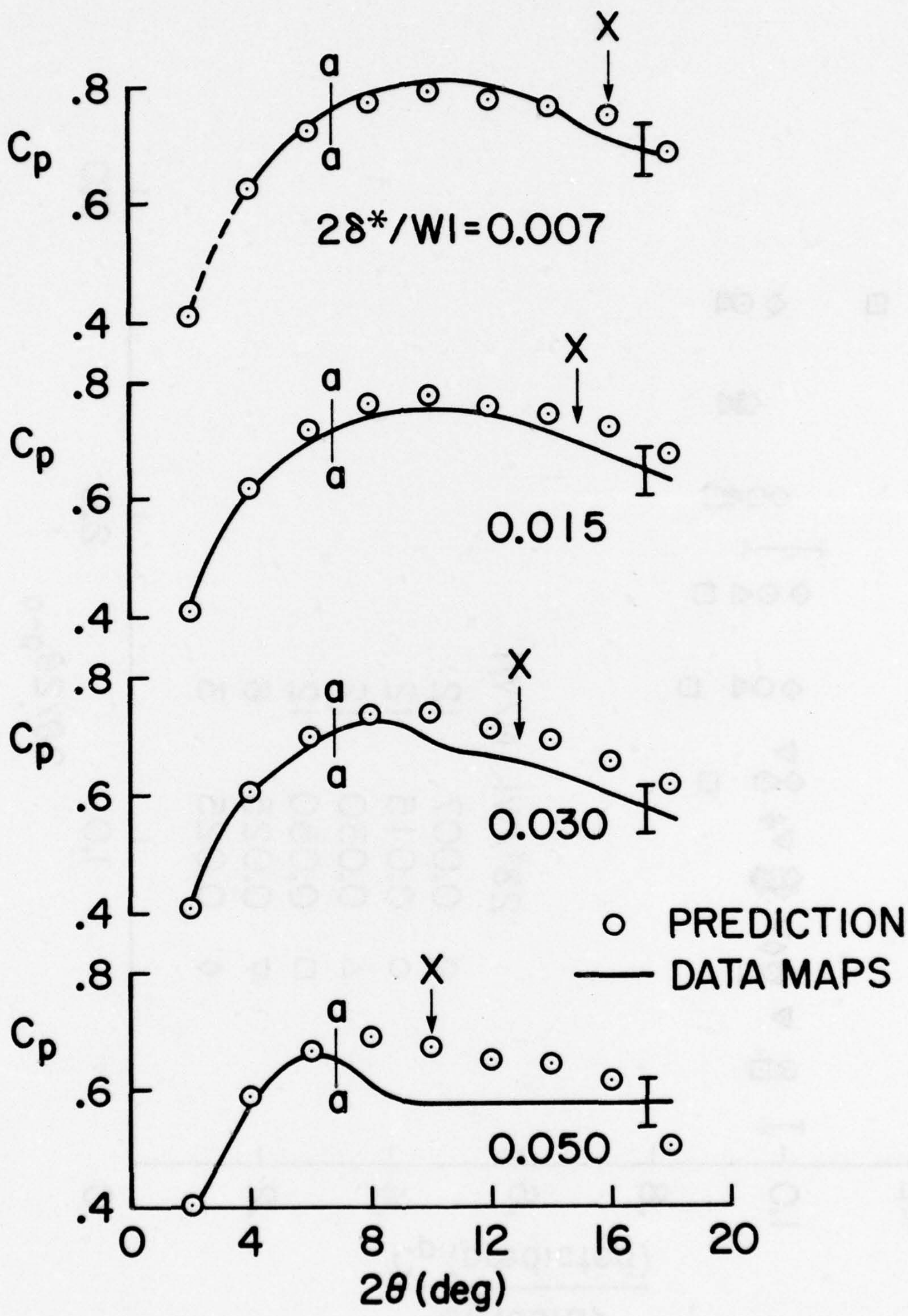


Fig. 26. Summary of $N/W_1 = 12$ diffusers, comparing the data maps of Reneau et al. [45] with 1-D core diffuser prediction. X is location where upper and lower wall shear layers begin to interfere with each other.

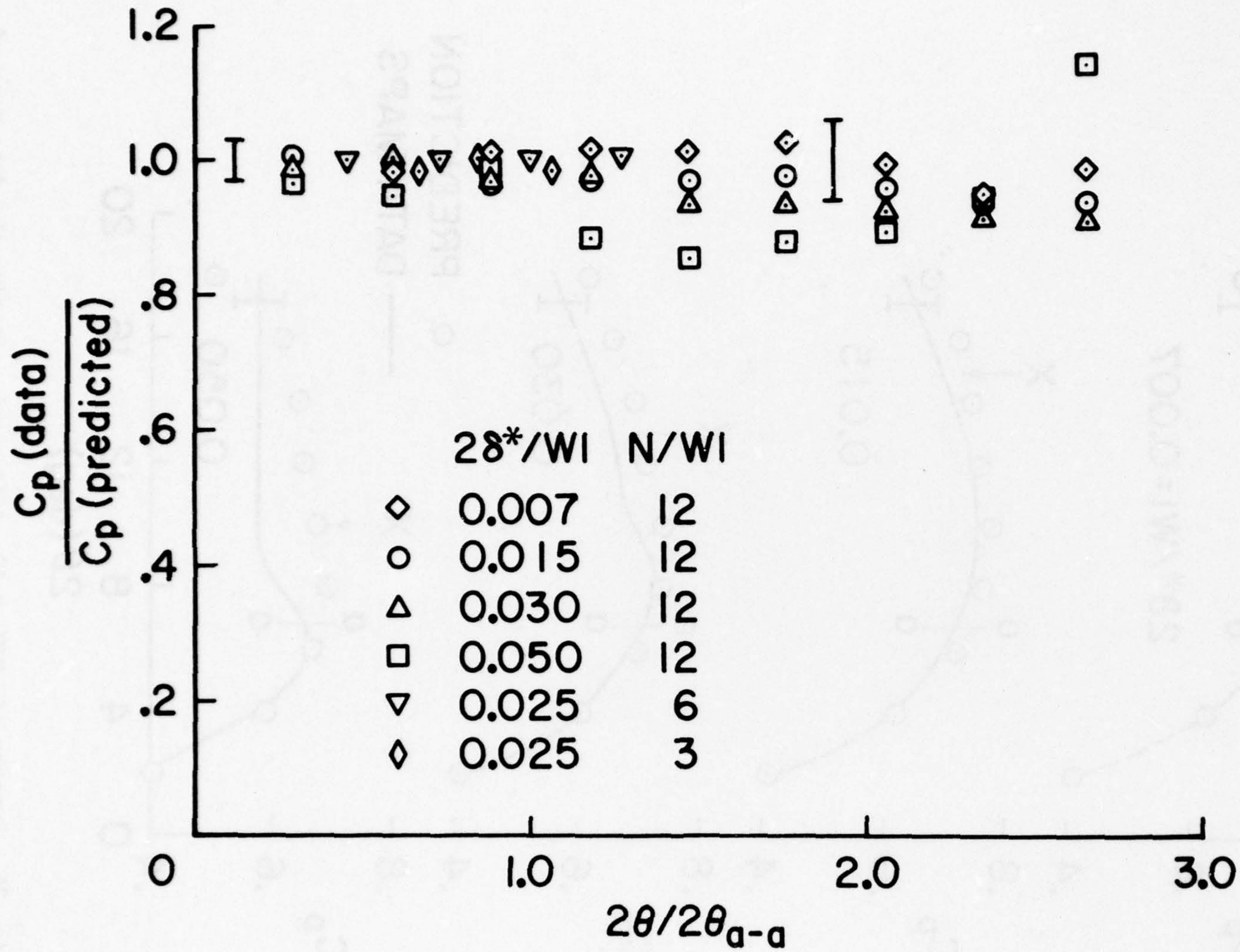


Fig. 27. Summary of performance of all tested diffusers.

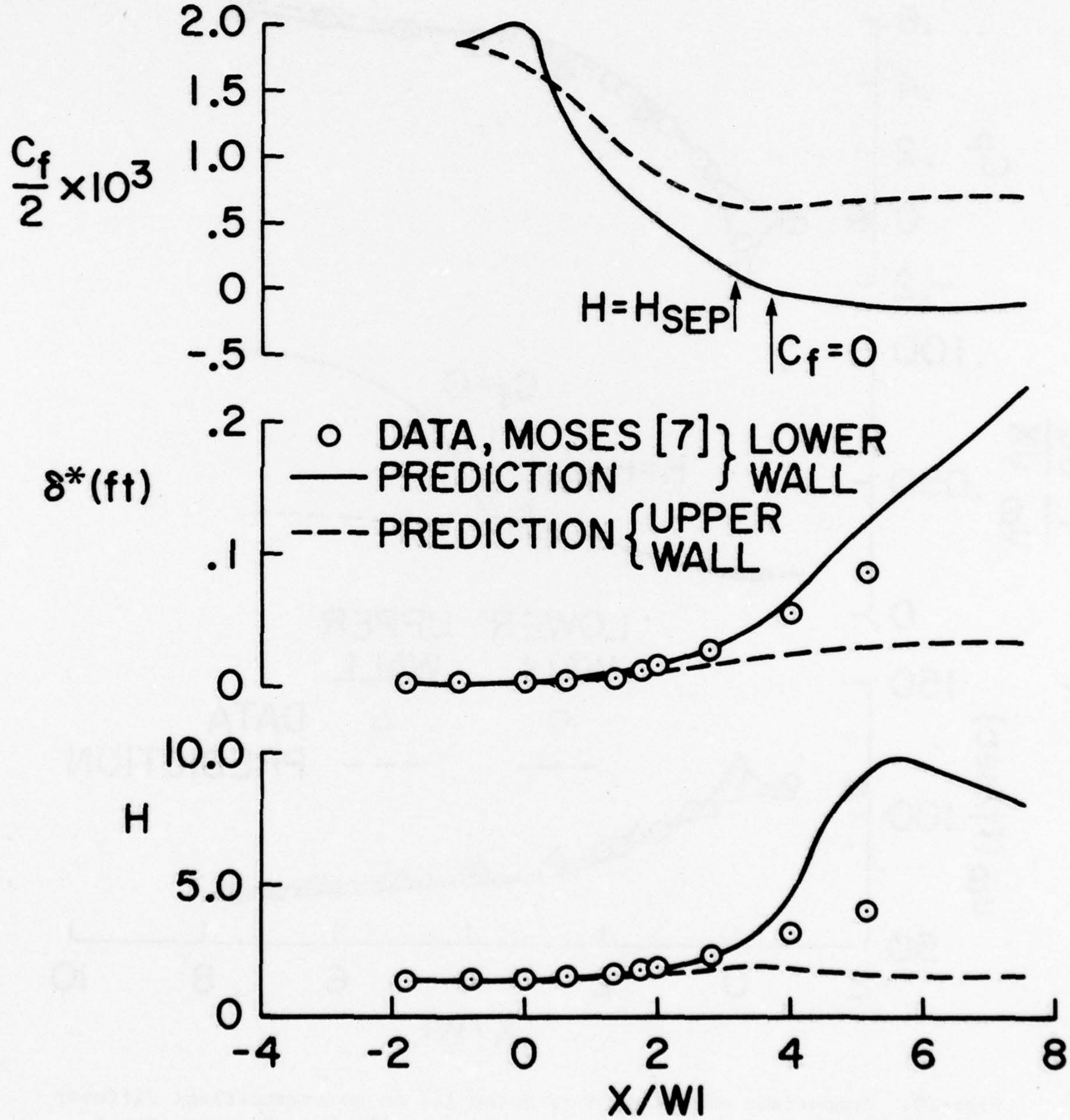


Fig. 28. Comparison of the data of Moses [7] on an asymmetrical diffuser with the 2-D core calculation. X_c = 100 ft. No $C_f/2$ data are available for comparison.

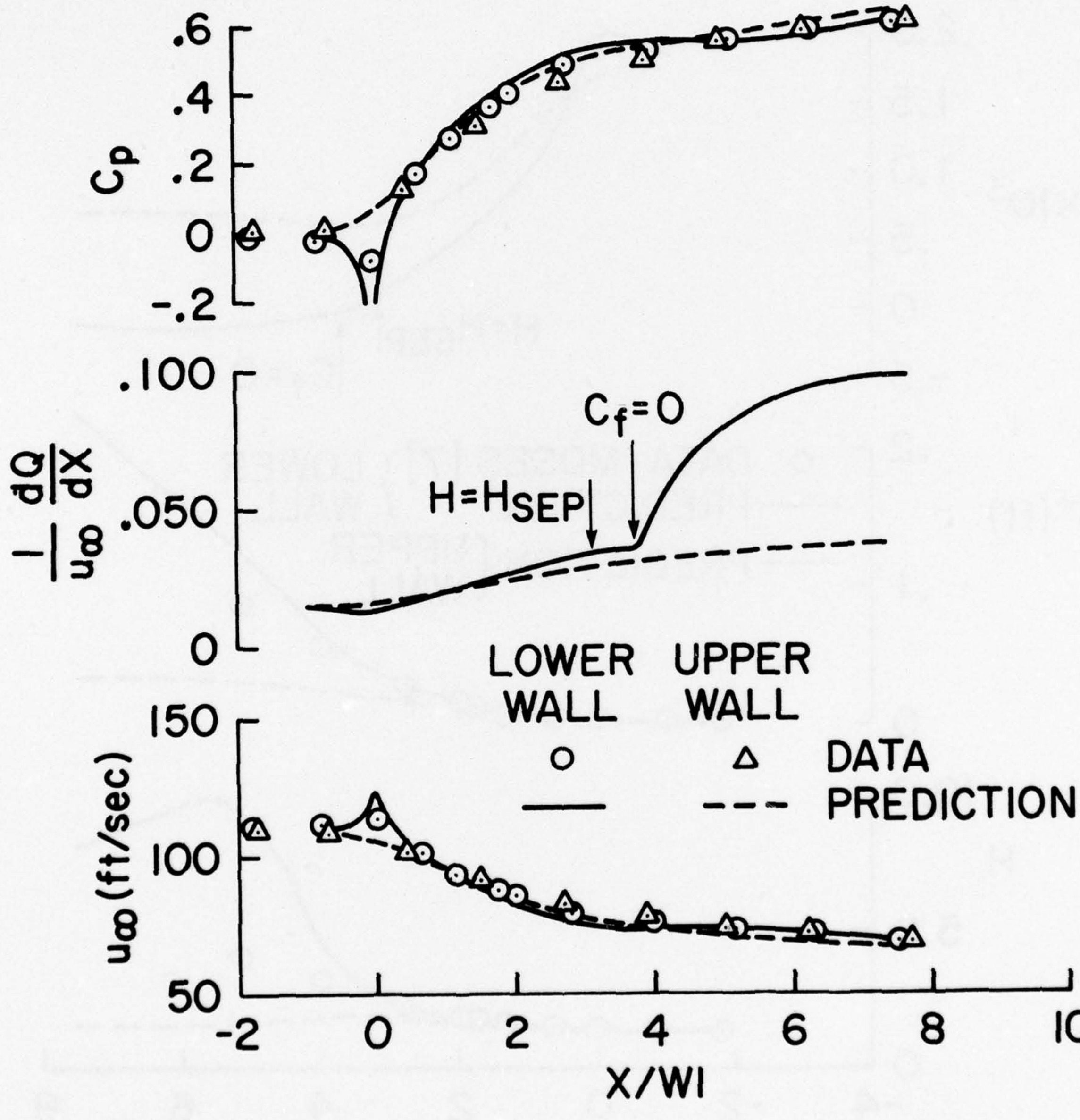


Fig. 29. Comparison of the data of Moses [7] on an asymmetrical diffuser with the 2-D core calculation. x_c = 100 ft. No entrainment data are available for comparison.

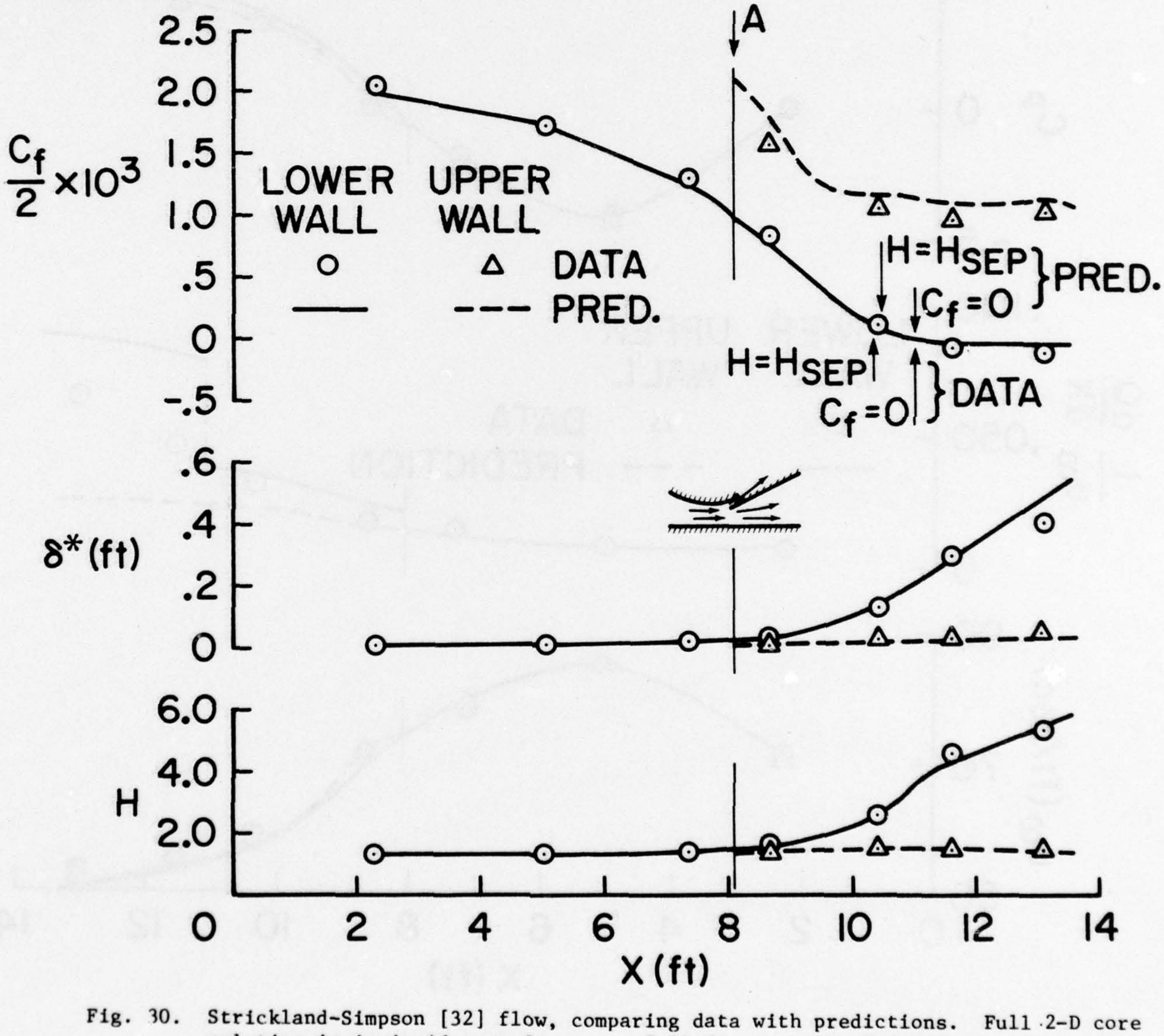


Fig. 30. Strickland-Simpson [32] flow, comparing data with predictions. Full 2-D core solution in inviscid core from x = 8.11 ft to exit of diffuser. Prescribed pressure gradient calculation from inlet to x = 8.11 ft, at which point (marked A) the upper boundary layer was removed. No lag equation in reversed flow region.

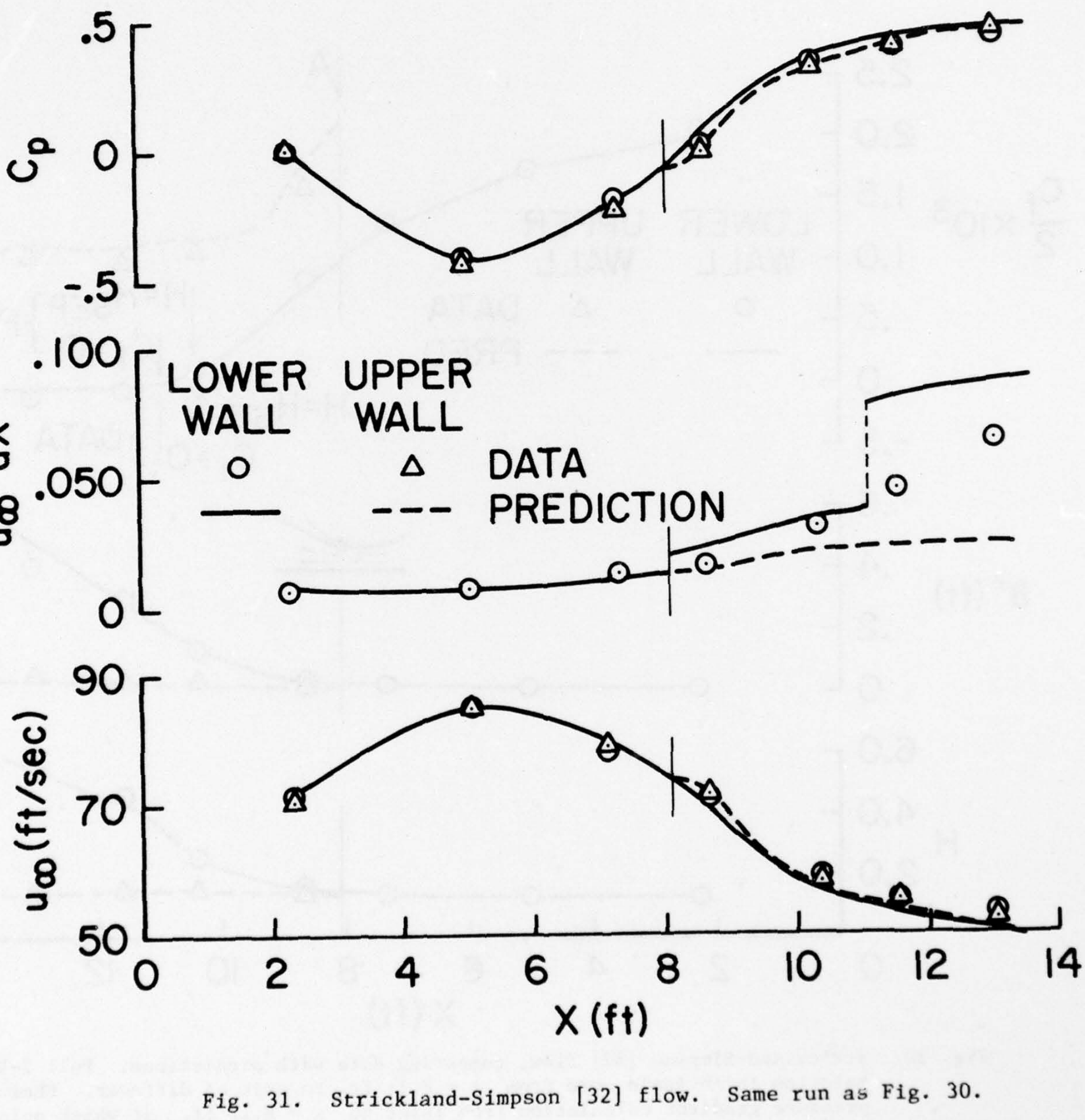


Fig. 31. Strickland-Simpson [32] flow. Same run as Fig. 30.

Appendix

USER'S GUIDE TO PROGRAM TSTALL

USER'S GUIDE TO PROGRAM TSTALL

UG1. INTRODUCTION

TSTALL is a FORTRAN program that performs four types of computations.

(a) Turbulent boundary layer development with prescribed pressure

(b) Calculation of diffusers operating in the unstalled and transitory stall regimes with $2\theta/2\theta_{a-a} = 1.2$, assuming one-dimensional core and symmetric b.l.'s.

(c) Calculation of the same diffusers as in (b) but with a two-dimensional core assumption.

(d) Solution of Laplace's equation using a boundary integral method.

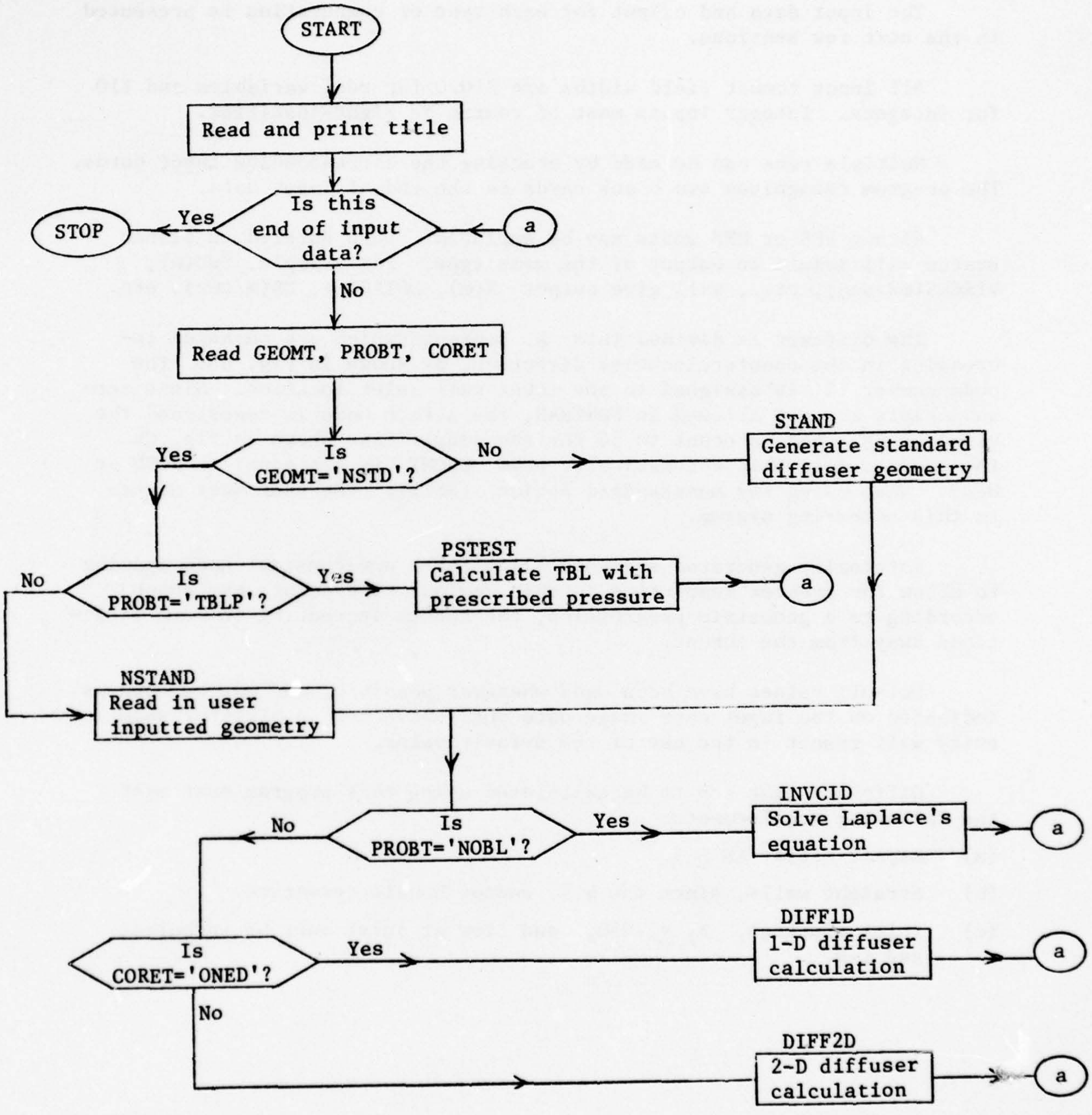
Access to the subroutines for each calculation is done in the MAIN routine according to inputted keywords for geometry, problem, and core types. The options available are:

Geometry type (GEOMT)
- Standard symmetric diffuser (STDD), where the coordinates defining the geometry are internally generated.
- Non-standard diffuser (NSTD), where the user enters the geometry.
- Standard asymmetric diffuser (HALF), where again the geometry is internally generated.

Problem type (PROBT)
- Calculate the TBL, given the pressure gradient (TBLP).
- Two-dimensional inviscid calculation only (NOBL).
- One- or two-dimensional diffuser calculation (DIFF).

Core type CORET)
- One-dimensional core (ONED).
- Two-dimensional core (TWOD).

Flowchart for the MAIN routine is shown in the figure below. Names of called subroutines are marked above the relevant boxes.



Flowchart MAIN Routine

UG2. GENERAL CONSIDERATIONS

The input data and output for each type of computation is presented in the next few sections.

All input format field widths are E10.0 for real variables and I10 for integers. Integer inputs must of course be right-justified.

Multiple runs can be made by stacking the corresponding input cards. The program recognizes two blank cards as the end of input data.

Either FPS or MKS units may be employed. Data entered in either system will result in output of the same type. For example, SWU(m), VISCOS(m2/sec), etc., will give output X(m), DSTAR(m), UB(m/sec), etc.

The diffuser is divided into N segments which are numbered increasing in the counterclockwise direction, as shown in Fig. U6. The node number 0 is assigned to the lower wall inlet location. Since zero subscripts are not allowed in FORTRAN, the zeroth node is reassigned the value of N, and is equal to 36 for the sample case shown in Fig. U6. The program does this automatically when GEOMT is set equal to STDD or HALF. When using the nonstandard option (NSTD), the user must adhere to this numbering system.

Internally generated segments allow for a non-constant node spacing to allow for greater resolution in this region. The points are spaced according to a geometric progression, the spaces increasing in both directions away from the throat.

Default values have been used wherever possible, and may be used as indicated on the input card image data outlined next. A blank or zero entry will result in the use of the default value.

Diffusers that are to be calculated using this program must meet the following requirements:

(a) Aspect ratio $AS \geq 5$.

(b) Straight walls, since the b.l. cannot handle curvature.

(c) Inlet blockage, $B_1 \leq .050$, and flow at inlet must be turbulent and 1-D.

UG3. B.L. CALCULATION WITH PRESCRIBED PRESSURE GRADIENT

The input data may be conveniently entered using the template shown in Fig. U1. A description of the input parameters on each card follows, with the format information in parentheses at the end.

Card 1. Title for user identification (A(80)).

Card 2. Keywords specifying geometry (NSTD) and problem (TBLP), as shown (3(6X,A4)).

Card 3. Number of stations along the flow for which velocity data will be inputted (I10).

Card 4. Starting values of XX,DELST,H and the kinematic viscosity VISCOS (4E10.5).

Card 5. Interval between b.l. printouts, IPR (recommended=1), location of 3-D source XC (default=1E5 if left blank).

Card 6. Repeated values of XX station location SWU and the corresponding velocity VIU, until all data are exhausted (8E10.0).

Card 7. ......

Card 8. ......

Sample input for Bradshaw-Ferriss relaxing flow (2400) is shown in Fig. U2. The corresponding output is presented in Fig. U3 and plotted in Figs. 9a and 9b.

TBL CALCULATION WITH PRESCRIBED PRESSURE GRADIENT

Card \ Column →	10	20	30	40	50	60	70	80
1	TITLE							
2	GEOMT	PROBT	CORET					
3	NPTS							
4	XX	DELST	H	VISCOS				
5	IPRXC							
6	SWU(1)	VIU(1)	SWU(2)	VIU(2)	etc.			
7	SWU(5)	VIU(5)	SWU(6)	VIU(6)	etc.			
. . .								
etc.	/*							

Fig. U1. Template for turbulent boundary layer calculations with prescribed pressure gradient.

```
** BRADSHAW-FERRISS RELAXING FLOW, A=-0.255-->0.(IDENT 2400) **
       NSTD         TBLP
          7
4.417     0.0611     1.603      0.000156
          1   0.0
3.917     112.18     4.417      111.09     4.917      110.0      5.417      110.0
5.917     110.0      6.917      110.0      7.917      110.0
```

Fig. U2

FIG. U3

** BRADSHAW-FERRISS RELAXING FLOW, A=-0.255-->0.(IDENT 2400) ** 6.

GEOMETRY= NSTD PROBLEM TYPE= TBLP CORE VELOCITY PROFILE=

X,DELST,H,VISCOS,XC,IPR= 4.4170E 00 6.1100E-02 1.6030E 00 1.5600E-04 $\underbrace{1.0000\text{E } 05}_{\text{default } X_c \to \infty}$ 1

X	UI (U_∞)
0.39170E 01	0.11218E 03
0.44170E 01	0.11109E 03
0.49170E 01	0.11000E 03
0.54170E 01	0.11000E 03
0.59170E 01	0.11000E 03
0.69170E 01	0.11000E 03
0.79170E 01	0.11000E 03

START VALUES,UT= 2.79884 UB= 39.55864 DELTA= 0.25512 DELST= 0.06110 H= 1.60300

X	DSTAR(δ^*)	H	HSEP	CP	DELT(δ)	UB (U_β)	UT (U_τ)	UI (U_∞)	CF/2	DQDX/UI
4.41700	0.06110	1.603	2.315	0.0	0.25512	39.55864	2.79884	111.08998	0.000635	0.02442
4.47810	0.06133	1.597	2.313	0.003	0.25711	39.25581	2.80234	110.93831	0.000638	0.02448
4.53920	0.06157	1.594	2.312	0.006	0.25912	38.96411	2.80532	110.78357	0.000641	0.02455
4.60030	0.06182	1.591	2.310	0.008	0.26115	38.67746	2.80815	110.62947	0.000644	0.02461
4.66140	0.06205	1.587	2.309	0.011	0.26316	38.38991	2.81118	110.47975	0.000647	0.02467
4.72250	0.06226	1.584	2.307	0.013	0.26514	38.09102	2.81474	110.33841	0.000651	0.02473
4.84469	0.06258	1.576	2.303	0.018	0.26893	37.45436	2.82541	110.09929	0.000658	0.02482
4.96689	0.06262	1.567	2.299	0.020	0.27228	36.70284	2.84394	109.95586	0.000669	0.02487
5.08909	0.06238	1.556	2.293	0.021	0.27516	35.83452	2.87066	109.91504	0.000682	0.02487
5.21129	0.06200	1.544	2.287	0.021	0.27778	34.92152	2.90124	109.93317	0.000697	0.02484
5.33349	0.06215	1.535	2.284	0.020	0.28106	34.18015	2.92007	109.97440	0.000700	0.02479
5.45569	0.06073	1.519	2.273	0.019	0.28343	33.27689	2.95253	110.00809	0.000734	0.02474
5.57789	0.06179	1.516	2.275	0.019	0.28620	32.49922	2.97748	110.01930	0.000720	0.02470
5.70009	0.06071	1.502	2.266	0.019	0.28899	31.75302	3.00101	110.01637	0.000748	0.02465
5.82229	0.06079	1.495	2.263	0.019	0.29184	31.05159	3.02277	110.00720	0.000755	0.02461
6.06669	0.06041	1.479	2.255	0.020	0.29753	29.71867	3.06389	109.99347	0.000776	0.02449
6.31108	0.06010	1.465	2.247	0.020	0.30319	28.47223	3.10234	109.99164	0.000796	0.02438
6.55548	0.05983	1.452	2.240	0.020	0.30887	27.31047	3.13794	109.99452	0.000814	0.02425
6.79988	0.05962	1.439	2.234	0.020	0.31455	26.22922	3.17072	109.99849	0.000831	0.02411
7.04428	0.05904	1.425	2.226	0.020	0.32024	25.05376	3.20656	110.00092	0.000853	0.02397
7.16648	0.05917	1.421	2.224	0.020	0.32310	24.58075	3.22051	110.00131	0.000857	0.02390
7.28868	0.05911	1.416	2.222	0.020	0.32596	24.12379	3.23389	110.00140	0.000864	0.02382
7.41088	0.05907	1.412	2.219	0.020	0.32881	23.68222	3.24673	110.00128	0.000871	0.02374
7.53308	0.05903	1.407	2.216	0.020	0.33167	23.25522	3.25906	110.00105	0.000878	0.02366
7.65528	0.05892	1.402	2.214	0.020	0.33453	22.81071	3.27201	110.00073	0.000885	0.02357
7.89968	0.05891	1.394	2.209	0.020	0.34023	22.02457	3.29433	110.00003	0.000897	0.02341
7.91700	0.05882	1.386	2.205	0.020	0.34591	21.28683	3.31502	110.00000	0.000908	0.02324

*** END ***

UG4. STANDARD DIFFUSER CALCULATION WITH 1-D OR 2-D CORE

The template shown in Fig. U4 details input data for diffuser calculations with either a one- or two-dimensional inviscid core. Refer to Figs. U5 and U6 for standard diffuser geometry and nomenclature.

Card 1. Title for user identification (A(80)).

Card 2. Keywords specifying geometry (HALF or STDD), problem (NOBL or DIFF), and core type (ONED or TWOD). (3(6X,A4)).

Card 3. X1, RC1, N, RC2, X2, W1 (Fig. U5), TWOTHD (2θ in degrees -- for asymmetric unit, enter twice the angle of the diverging wall), aspect ratio AS (default AS-8). (8E10.5).

Card 4. Number of segments in inlet (N1), throat curve (NC1), diffusing section (N2), exit curve (NC2), tailpipe (N3). ND1 and ND2 are fractions of the inlet and diffusing sections where node nearest the throat is to be located (default ND1=5*N1, ND2=5*N2). (7I10).

Card 5. Inlet blockage $B1=2\delta^*/W1$, inlet velocity U_∞(UI), kinematic viscosity VISCOS, location of source or sink for 3-D correction XC (default XC=1E5). (4E10.5).

Card 6. Inlet b.l. parameters, lower wall H and δ^* (HS,DELSTS), and upper wall (HU, DELSTU). Leaving HU and DELSTU blank implicitly sets them equal to the lower wall values. (4E10.0).

Card 7. Boundary layer print interval IPR (recommended=2), type of printout for interior points in the inviscid core NORMPR (for NOBL option only, =-1,+1,0 for normalized, dimensional or both), ITMAX the maximum allowable iterations for 2-D core diffuser calculation (recommended 8 to 10), CPEROR is the convergence criterion (recommended .025). (3I10,E10.0).

Sample input for the 1-D core diffuser of Carlson and Johnston [27] AR=2.4, N/W1=6, B1=.025, is shown in Fig. U7. The corresponding output is shown in Fig. U8 and plotted in Fig. 21.

For inviscid calculations (PROBT='NOBL'), additional input data are needed for determination of the values of the complex function and its derivatives at interior points.

Card 8. LINES is the number of lines along which interior point values are to be computed (default 0). (I10).

Card 9. Enter one card for each line along which interior point valu[es] are desired. X1,Y1 (coordinates of start of line), X2,Y2 (en[d] of line), NSEGS (number of segments that each line is divided into). Do not place any interior point on a node location. (4E10.0,I10).

Card 10. ......

Card 11. ......

.

.

Sample input for an inviscid solution of the Carlson et al. diffuser is shown in Fig. U9 and the corresponding output in Fig. U10.

STANDARD DIFFUSER CALCULATION WITH 1-D OR 2-D POTENTIAL CORE

Card \ Column →	10	20	30	40	50	60	70	80
1	TITLE							
2	GEOMT	PROBT	CORET					
3	X1	RC1	N	RC2	X2	W1	TWOTHD	AS
4	N1	NC1	N2	NC2	N3	ND1	ND2	
5	B1	UI	VISCOS	XC				
6	HS	DELSTS	HU	DELSTU				
7	IPR	NORMPR	ITMAX	CPEROR				
8	LINES							
9	X1	Y1	X2	Y2	NSEGS			
10	X1	Y1	X2	Y2	NSEGS			
11	etc.	etc.						

Fig. U4. Template for standard diffuser calculations. Cards 8 through the end are for interior point computations only and are not required for diffuser calculations.

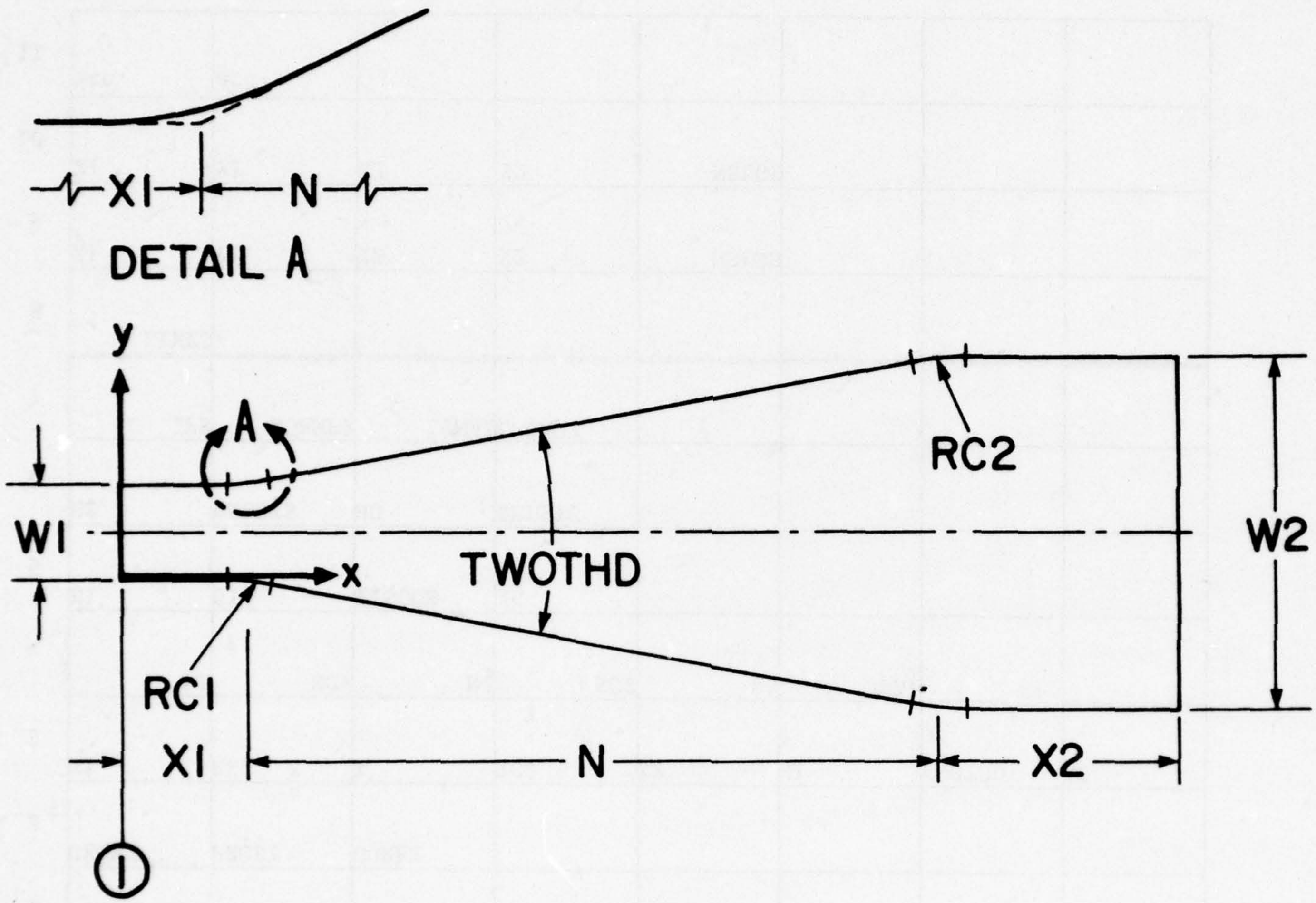


Fig. U5. Geometry of standard two-dimensional diffusers.

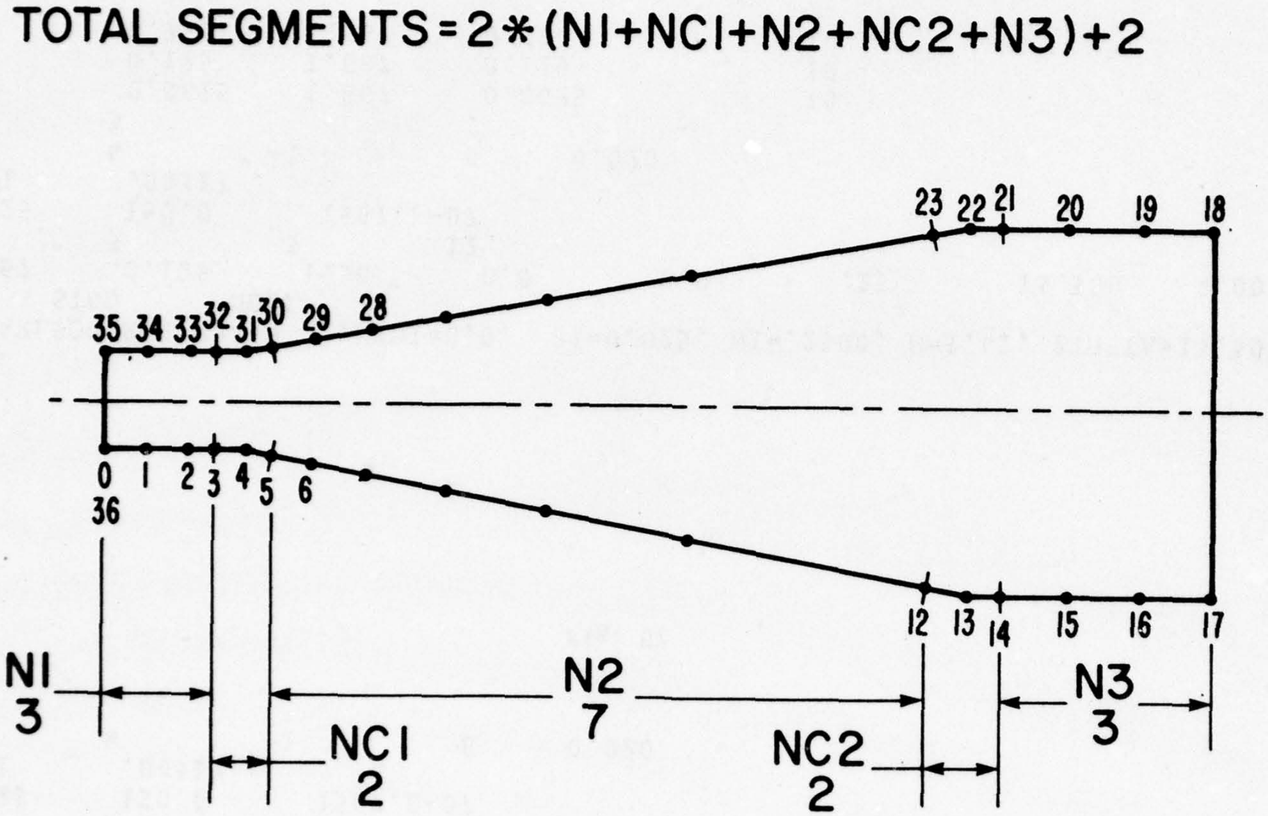


Fig. U6. Segment locations for a typical 36-node standard diffuser.

```
**CARLSON RUN 62430    ,N/W1=6.0,   B1=0.025, W1=.2500, H=1.41, 2THETA=13.309
      STDD         DIFF        ONED
0.167      0.125      1.50      0.0      0.0      .25      13.309    4.00
           3           3          12
0.025      150.0       1567.E-07
1.41       .00317
           4          -1          6      0.020
```

Fig. U7

```
**CARLSON RUN 62430    ,N/W1=6.0,   B1=0.025, W1=.2500, H=1.41, 2THETA=13.309
      STDD         HOBL
0.167      0.125      1.50      0.0      0.0      .25      13.309    4.00
           3           3          12
0.025      150.0       1567.E-07
1.41       .00317
           4          -1          6      0.020
           3
0.0        0.0625     1.667     0.0625          10
0.0        0.125      1.667     0.125           10
0.0        0.750      1.667     0.750           10
```

Fig. U9

Fig. U8

```
**CARLSON RUN 62430    ,N/W1=6.0,  B1=0.025, W1=.2500, H=1.41, 2THETA=13.309

GEOMETRY= STRD     PROBLEM TYPE= DIFF     CORE VELOCITY PROFILE= CNED

DIFFUSER GEOMETRY-INLET (X1-FT), THROAT RAD(RC1-FT), DIFFUSING LENGTH(N-FT), EXIT RADIUS(RC2-FT), TAILPIPE(X2-FT)
                  0.16700        0.12500                 1.50000                 0.0                 0.0

            WIDTH(W1-FT), TWOTHD(DEGREES), ASPECT-RATIO
                  0.25000        13.30900                4.00000

SEGMENT DISTRIBUTION - INLET,THROAT CURVE, DIFFUSING SECTTION, EXIT CURVE, TAILPIPE
                         3      3             12                  0            0

 B1,UI,VISCOS,XC=       0.02500    150.00000    0.0001567 0.10000E 06

INLET BL VALUES:LOWER WALL-H= 1.41, DELSTS= 0.31700E-02(FT)
            UPPER WALL-H= 1.41, DELSTU= 0.31700E-02(FT)

BL PRINT INTERVAL(IPR)= 4,  NORMPR=-1, MAX # ITERATIONS= 6, MAX ALLOWABLE CP ERROR= 2.00000E-02
```

1	9.58397E-02	0.0	0.0	2.50000E-01	0.0
2	1.49084E-01	0.0	0.0	2.50000E-01	9.58397E-02
3	1.59733E-01	0.0	0.0	2.50000E-01	1.49084E-01
4	1.64561E-01	-9.32813E-05	-3.86358E-02	2.50000E-01	1.59733E-01
5	1.69390E-01	-3.73542E-04	-7.73304E-02	2.50187E-01	1.64562E-01
6	1.74218E-01	-8.42094E-04	-1.16141E-01	2.50747E-01	1.69399E-01
7	1.99098E-01	-3.74476E-03	-1.16143E-01	2.51684E-01	1.74250E-01
8	2.42072E-01	-8.75844E-03	-1.16143E-01	2.57490E-01	1.99299E-01
9	3.03140E-01	-1.58832E-02	-1.16143E-01	2.67517E-01	2.42564E-01
10	3.82303E-01	-2.51189E-02	-1.16143E-01	2.81766E-01	3.04047E-01
11	4.79559E-01	-3.64656E-02	-1.16143E-01	3.00238E-01	3.83746E-01
12	5.94911E-01	-4.99234E-02	-1.16143E-01	3.22931E-01	4.81663E-01
13	7.28356E-01	-6.54922E-02	-1.16143E-01	3.49847E-01	5.97796E-01
14	8.79896E-01	-8.31720E-02	-1.16143E-01	3.80984E-01	7.32147E-01
15	1.04953E 00	-1.02963E-01	-1.16143E-01	4.16344E-01	8.84715E-01
16	1.23726E 00	-1.24865E-01	-1.16143E-01	4.55926E-01	1.05550E 00
17	1.44308E 00	-1.48877E-01	-1.16143E-01	4.99729E-01	1.24450E 00
18	1.66700E 00	-1.75001E-01	-1.16143E-01	5.47755E-01	1.45172E 00
19	1.66700E 00	4.25001E-01	1.16143E-01	6.00003E-01	1.67715E 00
20	1.44308E 00	3.98877E-01	1.16143E-01		
21	1.23726E 00	3.74865E-01	1.16143E-01		
22	1.04953E 00	3.52963E-01	1.16143E-01		
23	8.79896E-01	3.33172E-01	1.16143E-01		
24	7.28356E-01	3.15492E-01	1.16143E-01		
25	5.94911E-01	2.99923E-01	1.16143E-01		
26	4.79559E-01	2.86466E-01	1.16143E-01		

27	3.82303E-01	2.75119E-01	1.16143E-01
28	3.03140E-01	2.65883E-01	1.16143E-01
29	2.42072E-01	2.58758E-01	1.16143E-01
30	1.99098E-01	2.53745E-01	1.16143E-01
31	1.74218E-01	2.50842E-01	1.16141E-01
32	1.69390E-01	2.50374E-01	7.73304E-02
33	1.64561E-01	2.50093E-01	3.86358E-02
34	1.59733E-01	2.50000E-01	0.0
35	1.49084E-01	2.50000E-01	0.0
36	9.58397E-02	2.50000E-01	0.0
37	0.0	2.50000E-01	0.0
38	0.0	0.0	0.0

REFERENCE QUANTITIES,VELOCITY= 150.00000VOLUME FLOWRATE= 36.54900CP MULTIPLIER= 1.05272
START VALUES,UT= 6.12627 UB= 20.77888 DELTA= 0.01877 DELST= 0.00317 H= 1.41000

X	DSTAR	H	HSEP	CP	DELT	UB	UT	UI	CF/2	DQDX/UI
0.0	0.00317	1.410	2.203	0.0	0.01877	20.77888	6.12627	150.00000	0.001668	0.01564
0.01268	0.00319	1.407	2.202	-0.000	0.01899	20.55560	6.12988	150.03418	0.001669	0.01563
0.03487	0.00323	1.403	2.200	-0.001	0.01937	20.18271	6.13558	150.09303	0.001671	0.01561
0.07291	0.00330	1.398	2.198	-0.002	0.02002	19.59961	6.14298	150.18333	0.001673	0.01557
0.13631	0.00343	1.391	2.194	-0.004	0.02112	18.77022	6.14891	150.29617	0.001674	0.01549
0.22507	0.00438	1.433	2.219	0.086	0.02439	24.54646	5.53115	143.37146	0.001488	0.01687
0.27579	0.00527	1.477	2.243	0.154	0.02693	29.36885	5.04756	137.93402	0.001337	0.01815
0.37723	0.00743	1.573	2.294	0.265	0.03271	37.93306	4.20999	128.61624	0.001071	0.02086
0.47967	0.01018	1.685	2.349	0.352	0.03931	45.62292	3.48823	120.78925	0.000840	0.02376
0.65619	0.01690	1.947	2.461	0.456	0.05353	57.87091	2.46043	110.64963	0.000494	0.02859
0.79299	0.02363	2.207	2.559	0.505	0.06593	66.78337	1.82695	105.57513	0.000299	0.03157
0.93515	0.03458	2.679	2.707	0.541	0.08347	78.75491	1.10770	101.57660	0.000119	0.03431
1.08731	0.04994	3.528	2.911	0.552	0.10478	93.97580	0.34833	100.36282	0.000012	0.03545
1.16338	0.05919	4.215	3.036	0.551	0.11631	102.94664	-0.14244	100.46410	-0.000002	0.04140
1.20142	0.06390	4.553	3.088	0.550	0.12262	106.32613	-0.29383	100.64836	-0.000009	0.04840
1.22360	0.06654	4.730	3.113	0.549	0.12633	107.83481	-0.35942	100.69676	-0.000013	0.05191
1.23628	0.06801	4.820	3.125	0.549	0.12846	108.51588	-0.38914	100.69736	-0.000015	0.05390
1.26163	0.07081	4.963	3.144	0.550	0.13272	109.50212	-0.43391	100.63441	-0.000019	0.05781
1.29699	0.07350	5.070	3.157	0.551	0.13699	110.12737	-0.46525	100.50607	-0.000021	0.06127
1.33771	0.07852	5.183	3.171	0.555	0.14557	110.32304	-0.49037	100.04665	-0.000024	0.06717
1.36940	0.08154	5.212	3.175	0.558	0.15096	110.11322	-0.49462	99.69470	-0.000025	0.07017
1.40744	0.08499	5.192	3.172	0.563	0.15749	109.40553	-0.48412	99.18315	-0.000024	0.07326
1.45316	0.08943	5.114	3.163	0.569	0.16631	108.16011	-0.45980	98.44739	-0.000021	0.07658
1.53423	0.09608	4.991	3.147	0.579	0.17983	106.00177	-0.41275	97.31294	-0.000018	0.08021
1.66103	0.10723	4.782	3.120	0.594	0.20300	102.59027	-0.33765	95.54597	-0.000012	0.08365
1.67715	0.10866	4.721	3.116	0.596	0.20600	101.47459	-0.31457	95.33231	-0.000011	0.08369

***********END OF ROUTINE DIFF1D********

```
**CARLSON RUN 62430    ,N/W1=6.0,  B1=0.025, W1=.2500, H=1.41, 2THETA=13.309

GEOMETRY= STDD    PROBLEM TYPE= NOBL    CORE VELOCITY PROFILE=


DIFFUSER GEOMETRY-INLET (X1-FT), THROAT RAD(RC1-FT), DIFFUSING LENGTH(N-FT), EXIT RADIUS(RC2-FT), TAILPIPE(X2-FT)
                     0.16700         0.12500            1.50000                  0.0                  0.0


                WIDTH(W1-FT), TWOTHD(DEGREES), ASPECT-RATIO
                     0.25000         13.30900           4.00000

SEGMENT DISTRIBUTION - INLET,THROAT CURVE, DIFFUSING SECTTION, EXIT CURVE, TAILPIPE
                         3      3           12                  0          0


 B1,UI,VISCOS,XC=      0.02500    150.00000    0.0001567 0.10000E 06

INLET BL VALUES:LOWER WALL-H= 1.41, DELSTS= 0.31700E-02(FT)
                UPPER WALL-H= 1.41, DELSTU= 0.31700E-02(FT)


BL PRINT INTERVAL(IPR)= 4,  NORMPR=-1, MAX # ITERATIONS= 6, MAX ALLOWABLE CP ERROR= 2.00000E-02
```

1	9.58397E-02	0.0	0.0	2.50000E-01	0.0
2	1.49084E-01	0.0	0.0	2.50000E-01	9.58397E-02
3	1.59733E-01	0.0	0.0	2.50000E-01	1.49084E-01
4	1.64561E-01	-9.32813E-05	-3.86358E-02	2.50000E-01	1.59733E-01
5	1.69390E-01	-3.73542E-04	-7.73304E-02	2.50187E-01	1.64562E-01
6	1.74218E-01	-8.42094E-04	-1.16141E-01	2.50747E-01	1.69399E-01
7	1.99098E-01	-3.74476E-03	-1.16143E-01	2.51684E-01	1.74250E-01
8	2.42072E-01	-8.75844E-03	-1.16143E-01	2.57490E-01	1.99299E-01
9	3.03140E-01	-1.58832E-02	-1.16143E-01	2.67517E-01	2.42564E-01
10	3.82303E-01	-2.51189E-02	-1.16143E-01	2.81766E-01	3.04047E-01
11	4.79559E-01	-3.64656E-02	-1.16143E-01	3.00238E-01	3.83746E-01
12	5.94911E-01	-4.99234E-02	-1.16143E-01	3.22931E-01	4.81663E-01
13	7.28356E-01	-6.54922E-02	-1.16143E-01	3.49847E-01	5.97796E-01
14	8.79896E-01	-8.31720E-02	-1.16143E-01	3.80984E-01	7.32147E-01
15	1.04953E 00	-1.02963E-01	-1.16143E-01	4.16344E-01	8.84715E-01
16	1.23726E 00	-1.24865E-01	-1.16143E-01	4.55926E-01	1.05550E 00
17	1.44308E 00	-1.48877E-01	-1.16143E-01	4.99729E-01	1.24450E 00
18	1.66700E 00	-1.75001E-01	-1.16143E-01	5.47755E-01	1.45172E 00
19	1.66700E 00	4.25001E-01	1.16143E-01	6.00003E-01	1.67715E 00
20	1.44308E 00	3.98877E-01	1.16143E-01		
21	1.23726E 00	3.74865E-01	1.16143E-01		
22	1.04953E 00	3.52963E-01	1.16143E-01		
23	8.79896E-01	3.33172E-01	1.16143E-01		
24	7.28356E-01	3.15492E-01	1.16143E-01		
25	5.94911E-01	2.99923E-01	1.16143E-01		
26	4.79559E-01	2.86466E-01	1.16143E-01		

27	3.82303E-01	2.75119E-01	1.16143E-01
28	3.03140E-01	2.65883E-01	1.16143E-01
29	2.42072E-01	2.58758E-01	1.16143E-01
30	1.99098E-01	2.53745E-01	1.16143E-01
31	1.74218E-01	2.50842E-01	1.16141E-01
32	1.69390E-01	2.50374E-01	7.73304E-02
33	1.64561E-01	2.50093E-01	3.86358E-02
34	1.59733E-01	2.50000E-01	0.0
35	1.49084E-01	2.50000E-01	0.0
36	9.58397E-02	2.50000E-01	0.0
37	0.0	2.50000E-01	0.0
38	0.0	0.0	0.0

LENGTH SCALE = 2.50000E-01
VELOCITY SC ALE = 1.50000E 02
NORMALIZED INLET VELOCITY= 1.00000E 00
NORMALIZED EXIT VELOCITY = 4.12813E-01

NORMALIZED SOLUTION

#	XC	YC	ALPHA	LN(VEL)	VELOCITY	CP
1	0.383359	0.0	0.0	0.003805	1.003812	-0.007638
2	0.596336	0.0	0.0	0.033849	1.034428	-0.070040
3	0.638931	0.0	0.0	0.064250	1.066360	-0.137122
4	0.658245	-0.000373	-0.038636	0.092676	1.097106	-0.203641
5	0.677558	-0.001494	-0.077330	0.090297	1.094499	-0.197927
6	0.696872	-0.003368	-0.116141	0.037925	1.038653	-0.078800
7	0.796391	-0.014979	-0.116143	-0.013212	0.986875	0.026077
8	0.968287	-0.035034	-0.116143	-0.067253	0.934959	0.125852
9	1.212560	-0.063533	-0.116143	-0.124378	0.883046	0.220230
10	1.529210	-0.100475	-0.116143	-0.189467	0.827400	0.315410
11	1.918238	-0.145863	-0.116143	-0.262982	0.768755	0.409015
12	2.379642	-0.199694	-0.116143	-0.343523	0.709267	0.496940
13	2.913424	-0.261969	-0.116143	-0.429266	0.650987	0.576216
14	3.519585	-0.332688	-0.116143	-0.518524	0.595398	0.645501
15	4.198120	-0.411851	-0.116143	-0.609886	0.543413	0.704702
16	4.949032	-0.499458	-0.116143	-0.702211	0.495488	0.754491
17.	5.772320	-0.595510	-0.116143	-0.794523	0.451797	0.795880
18	6.667988	-0.700005	-0.116143	-0.884761	0.412813	0.829586
19	6.667988	1.700005	0.116143	-0.884761	0.412813	0.829586
20	5.772320	1.595510	0.116143	-0.794523	0.451797	0.795880
21	4.949032	1.499458	0.116143	-0.702211	0.495488	0.754491
22	4.198120	1.411851	0.116143	-0.609886	0.543413	0.704702
23	3.519585	1.332687	0.116143	-0.518524	0.595398	0.645501
24	2.913424	1.261969	0.116143	-0.429266	0.650987	0.576216
25	2.379642	1.199694	0.116143	-0.343523	0.709267	0.496940
26	1.918238	1.145863	0.116143	-0.262982	0.768755	0.409015
27	1.529210	1.100475	0.116143	-0.189467	0.827400	0.315410
28	1.212560	1.063532	0.116143	-0.124378	0.883046	0.220230
29	0.968287	1.035033	0.116143	-0.067253	0.934959	0.125852
30	0.796391	1.014978	0.116143	-0.013212	0.986875	0.026077
31	0.696872	1.003368	0.116141	0.037926	1.038654	-0.078802
32	0.677558	1.001493	0.077330	0.090296	1.094498	-0.197925
33	0.658245	1.000373	0.038636	0.092676	1.097106	-0.203641
34	0.638931	1.000000	0.0	0.064250	1.066360	-0.137122
35	0.596336	1.000000	0.0	0.033849	1.034428	-0.070040
36	0.383359	1.000000	0.0	0.003805	1.003812	-0.007638
37	0.0	1.000000	0.0	0.0	1.000000	0.0
38	0.0	0.0	0.0	0.0	1.000000	0.0

VALUE OF ANALYTIC FUNCTION AND ITS DERIVATIVES AT 33 BOUNDARY AND/OR INTERIOR POINTS.

NORMALIZED VALUES

#	XO	YO	U	V	VEL-MAG	ALPHA	#
1	0.0	0.250000	0.998905	-0.000718	0.998905	-0.000719	1
2	0.666800	0.250000	0.985273	-0.027342	0.985652	-0.027744	2
3	1.333599	0.250000	0.864390	-0.043206	0.865469	-0.049943	3
4	2.000399	0.250000	0.761775	-0.034063	0.762536	-0.044685	4
5	2.667199	0.250000	0.680821	-0.027232	0.681365	-0.039978	5
6	3.333999	0.250000	0.615328	-0.022271	0.615731	-0.036178	6
7	4.000797	0.250000	0.561234	-0.018561	0.561541	-0.033059	7

8	4.667595	0.250000	0.515773	-0.015721	0.516012	-0.030471	8
9	5.334396	0.250000	0.476968	-0.013525	0.477159	-0.028348	9
10	6.001198	0.250000	0.443285	-0.011873	0.443444	-0.026778	10
11	6.667995	0.250000	0.003222	-0.000842	0.003330	-0.255764	11
12	0.0	0.500000	0.998355	0.000000	0.998355	0.000000	12
13	0.666800	0.500000	0.974213	0.000000	0.974213	0.000000	13
14	1.333599	0.500000	0.865991	0.000000	0.865991	0.000000	14
15	2.000399	0.500000	0.763283	0.000000	0.763283	0.000000	15
16	2.667199	0.500000	0.681903	0.000000	0.681903	0.000000	16
17	3.333999	0.500000	0.616126	0.000000	0.616126	0.000000	17
18	4.000797	0.500000	0.561839	0.000000	0.561839	0.000000	18
19	4.667595	0.500000	0.516239	0.000000	0.516239	0.000000	19
20	5.334396	0.500000	0.477327	0.000000	0.477327	0.000000	20
21	6.001198	0.500000	0.443542	0.000000	0.443542	0.000000	21
22	6.667995	0.500000	0.003363	0.000000	0.003363	0.000001	22
23	0.0	3.000000	-0.000339	0.000343	0.000483	2.349736	23
24	0.666800	3.000000	-0.000302	0.000392	0.000495	2.226433	24
25	1.333599	3.000000	-0.000259	0.000420	0.000493	2.123013	25
26	2.000399	3.000000	-0.000226	0.000433	0.000489	2.051423	26
27	2.667199	3.000000	-0.000207	0.000447	0.000492	2.004574	27
28	3.333999	3.000000	-0.000196	0.000469	0.000509	1.966317	28
29	4.000797	3.000000	-0.000185	0.000508	0.000540	1.921081	29
30	4.667595	3.000000	-0.000165	0.000565	0.000588	1.855151	30
31	5.334396	3.000000	-0.000118	0.000642	0.000653	1.753128	31
32	6.001198	3.000000	-0.000018	0.000724	0.000724	1.595953	32
33	6.667995	3.000000	0.000150	0.000757	0.000772	1.375878	33

NORMALIZED VALUES

#	XO	YO	(DP/DX)/RHO	(DP/DY)/RHO	(DP/DS)/RHO	(DP/DN)/RHO	CURVATURE	CP	#
1	0.0	0.250000	0.000723	0.003521	0.000720	0.003522	-0.003529	0.002189	1
2	0.666800	0.250000	0.109602	0.102784	0.106708	0.105784	-0.108886	0.028490	2
3	1.333599	0.250000	0.151264	-0.004313	0.151291	0.003243	-0.004330	0.250963	3
4	2.000399	0.250000	0.103817	-0.004566	0.103917	0.000076	-0.000131	0.418539	4
5	2.667199	0.250000	0.074162	-0.002925	0.074219	0.000041	-0.000088	0.535742	5
6	3.333999	0.250000	0.054819	-0.001946	0.054853	0.000038	-0.000100	0.620876	6
7	4.000797	0.250000	0.041669	-0.001338	0.041690	0.000040	-0.000127	0.684672	7
8	4.667595	0.250000	0.032432	-0.000937	0.032446	0.000052	-0.000194	0.733732	8
9	5.334396	0.250000	0.025794	-0.000641	0.025802	0.000090	-0.000396	0.772319	9
10	6.001198	0.250000	0.021028	-0.000351	0.021030	0.000212	-0.001080	0.803357	10
11	6.667995	0.250000	0.000011	-0.000001	0.000011	0.000002	-0.182773	0.999989	11
12	0.0	0.500000	0.004061	-0.000000	0.004061	-0.000000	0.000000	0.003287	12
13	0.666800	0.500000	0.104552	-0.000000	0.104552	-0.000000	0.000000	0.050910	13
14	1.333599	0.500000	0.148902	0.000000	0.148902	0.000000	-0.000000	0.250060	14
15	2.000399	0.500000	0.104188	0.000000	0.104188	0.000000	-0.000000	0.417399	15
16	2.667199	0.500000	0.074397	-0.000000	0.074397	-0.000000	0.000000	0.535008	16
17	3.333999	0.500000	0.054957	-0.000000	0.054957	-0.000000	0.000000	0.620386	17
18	4.000797	0.500000	0.041757	0.000000	0.041757	0.000000	-0.000000	0.684337	18
19	4.667595	0.500000	0.032494	0.000000	0.032494	0.000000	-0.000000	0.733497	19
20	5.334396	0.500000	0.025847	0.000000	0.025847	0.000000	-0.000000	0.772159	20
21	6.001198	0.500000	0.021090	0.000000	0.021090	0.000000	-0.000000	0.803270	21
22	6.667995	0.500000	0.000012	-0.000000	0.000012	-0.000000	0.000002	0.999989	22
23	0.0	3.000000	-0.000000	0.000000	0.000000	-0.000000	0.086408	1.000000	23
24	0.666800	3.000000	-0.000000	0.000000	0.000000	-0.000000	0.097039	1.000000	24
25	1.333599	3.000000	0.000000	0.000000	0.000000	-0.000000	0.082277	1.000000	25
26	2.000399	3.000000	0.000000	0.000000	0.000000	-0.000000	0.044690	1.000000	26
27	2.667199	3.000000	-0.000000	0.000000	0.000000	0.000000	-0.001007	1.000000	27
28	3.333999	3.000000	-0.000000	0.000000	0.000000	0.000000	-0.042379	1.000000	28
29	4.000797	3.000000	-0.000000	0.000000	0.000000	0.000000	-0.076113	1.000000	29
30	4.667595	3.000000	-0.000000	0.000000	0.000000	0.000000	-0.103822	1.000000	30
31	5.334396	3.000000	-0.000000	0.000000	0.000000	0.000000	-0.125602	1.000000	31

32	6.001198	3.000000	-0.000000	0.000000	0.000000	0.000000	-0.132884	1.000000	32
33	6.667995	3.000000	-0.000000	0.000000	0.000000	0.000000	-0.106838	0.999999	33

*** END ***

UG5. NONSTANDARD DIFFUSER CALCULATION WITH 1-D OR 2-D CORE

For nonstandard diffusers, node points describing the geometry have to be entered by the user. The location of the axes and the numbering system must be the same as that for the standard case, Fig. U6. The template, Fig. U11, describes the input data.

Card 1. Title for user identification (A(80)).

Card 2. Keyword specifying geometry (NSTD), problem (NOBL or DIFF), and core type (ONED or TWOD). (3(6X,A4)).

Card 3. Number of segments, total N, lower wall NR, upper wall NL. For a 2-D core calculation, NR should be = NL. N=NR+NL+2. (3I10).

Card 4. Enter (XW,YW) coordinates of segment end points, and ALW the angle between the segment and the positive X direction (CCW positive CW negative). Enter one card for each node, beginning with node 1 and ending with node N (which is really node zero). (3E10.0).

Card 5. ......

Card 6. ......

.

.

Card (4+N).

Card (5+N). Inlet width W1, divergence angle TWOTHD (degrees), aspect ratio AS (default AS=8). (3E10.0).

Card (6+N). Inlet blockage B1, inlet velocity UI, kinematic viscosity VISCOS, location of 3-D source or sink XC (default 1E5). (4E10.0).

Card (7+N). B.1. print interval IPR (recommended=2), type of printout for interior points in the inviscid core NORMPR (for NOBL option only, =-1,+1,0 for normalized, dimensional, or both), ITMAX the maximum number of iterations allowed for 2-D core calculation (recommended 8 to 10)., CPEROR is the convergence criterion (recommended=.025). (3I10,E10.0).

Card (8+N). Inlet b.l. parameters, lower wall H and δ^* (HS,DELSTS), and upper wall (HU, DELSTU). Leaving HU and DELSTU blank implicitly sets them equal to the lower wall values. (4E10.0).

Fig. U12 is sample input data for a 2-D core calculation of Strickland-Simpson's flow. Fig. U13 is the first and last few pages of output, which are plotted in Figs. 30 and 31.

If only an inviscid solution is desired, cards 8 through the end of the standard diffuser input should be added to the end of this deck.

NONSTANDARD DIFFUSER CALCULATION WITH 1-D OR 2-D POTENTIAL CORE

Card \ Column →	10	20	30	40	50	60	70	80
1	TITLE							
2	GEOMT	PROBT	CORET					
3	N	NR	NL					
4	XW(1)	YW(1)	ALW(1)					
5	XW(2)	YW(2)	ALW(2)					
6	---	---	---					
.	---	---	---					
.								
.	-----							
4+N	XW(N)	YW(N)	ALW(N)					
5+N	W1	TWOTHD	AS					
6+N	B1	UI	VISCOS	XC				
7+N	IPR	NORMPR	ITMAX	CPEROR				
8+N	HS	DELSTS	HU	DELSTU				

Fig. U11. Template for nonstandard diffuser calculations.

```
C--****STRICKLAND-SIMPSON'S SEPARATING FLOW IN SIMULTANEOUS ITERATION****
      NSTD      DIFF      TWOD
        24        11        11
0.392      0.0       0.0
0.892
1.392
1.892
2.392
2.892
3.392
3.892
4.392
4.892
5.392
5.392     1.891     0.194
4.892     1.792     0.194
4.392     1.693     0.194
3.892     1.594     0.194
3.392     1.490     0.194
2.892     1.396     0.194
2.392     1.297     0.188
1.892     1.203     0.181
1.392     1.115     0.181
0.892     1.021     0.181
0.392     0.928     0.157
0.00      0.870     0.067
0.0       0.0       0.0
0.87      21.8      6.00
0.0380    73.8       167.67E-6  0.0
        3         -1           8  0.020
1.53      0.027     2.0        .006
```

C--****STRICKLAND-SIMPSON'S SEPARATING FLOW IN SIMULTANEOUS ITERATICN****

GEOMETRY= NSTD   PROBLEM TYPE= DIFF   CORE VELOCITY PROFILE= TWOD

****NON-STANDARD DUCT, USER INPUTTED WALL COORDINATES

NODE#	XW	YW	ALW
1	3.92000E-01	0.0	0.0
2	8.92000E-01	0.0	0.0
3	1.39200E 00	0.0	0.0
4	1.89200E 00	0.0	0.0
5	2.39200E 00	0.0	0.0
6	2.89200E 00	0.0	0.0
7	3.39200E 00	0.0	0.0
8	3.89200E 00	0.0	0.0
9	4.39200E 00	0.0	0.0
10	4.89200E 00	0.0	0.0
11	5.39200E 00	0.0	0.0
12	5.39200E 00	1.89100E 00	1.94000E-01
13	4.89200E 00	1.79200E 00	1.94000E-01
14	4.39200E 00	1.69300E 00	1.94000E-01
15	3.89200E 00	1.59400E 00	1.94000E-01
16	3.39200E 00	1.49000E 00	1.94000E-01
17	2.89200E 00	1.39600E 00	1.94000E-01
18	2.39200E 00	1.29700E 00	1.88000E-01
19	1.89200E 00	1.20300E 00	1.81000E-01
20	1.39200E 00	1.11500E 00	1.81000E-01
21	8.92000E-01	1.02100E 00	1.81000E-01
22	3.92000E-01	9.28000E-01	1.57000E-01
23	0.0	8.70000E-01	6.70000E-02
24	0.0	0.0	0.0

```
SEGMENT COUNT, TOTAL=24    LOWER WALL= 11      UPPER WALL= 11

INLET WIDTH=  8.70000E-01(FT),   TWOTHETA=  2.18000E 01(DEG),    ASPECT RATIO=  6.00000E 00    KC= 1.00000E 05

INLET BLOCKAGE=  3.80000E-02,    INLET  CORE VELOCITY=  7.38000E 01(FT/SEC),   KINEMATIC VISCOSITY= 1.67670E-04(FT2/SEC)

INLET BL VALUES:LOWER WALL-H= 1.53, DELSTS= 0.27000E-01(FT)
                UPPER WALL-H= 2.00, DELSTU=  0.60000E-02(FT)

B.L.PRINT INTERVAL= 3,   PRINT TYPE(NORMPR) =  -1,    MAX CP ERROR=0.02000,    MAX # ITERATIONS= 8

BOUNDARY WIDTH FOR THE FIRST ITERATION--
```

#	SWU(J)	SWT(J)	WI(J)
1	0.0	0.0	8.70000E-01
2	3.96267E-01	3.89312E-01	9.28004E-01
3	9.04843E-01	8.88961E-01	1.02100E 00
4	1.41360E 00	1.38879E 00	1.11500E 00
5	1.92129E 00	1.88756E 00	1.20301E 00
6	2.43004E 00	2.38739E 00	1.29701E 00
7	2.93975E 00	2.88815E 00	1.39600E 00
8	3.44851E 00	3.38798E 00	1.49000E 00
9	3.95221E 00	3.88972E 00	1.59400E 00
10	4.46892E 00	4.39048E 00	1.69300E 00
11	4.97862E 00	4.89124E 00	1.79200E 00
12	5.48833E 00	5.39200E 00	1.89100E 00

OUTPUT FROM SOLUTION
OF 2-D LAPLACE'S EQN.

LENGTH SCALE = 8.70000E-01
VELOCITY SCALE = 7.38000E 01
NORMALIZED INLET VELOCITY= 1.00000E 00
NORMALIZED EXIT VELOCITY = 4.70468E-01

NORMALIZED SOLUTION

#	XC	YC	ALPHA	LN(VEL)	VELOCITY	CP
1	0.450575	0.0	0.0	-0.066235	0.935911	0.124070
2	1.025287	0.0	0.0	-0.140551	0.868879	0.245049
3	1.599998	0.0	0.0	-0.220418	0.802184	0.356502
4	2.174711	0.0	0.0	-0.298655	0.741815	0.449710
5	2.749424	0.0	0.0	-0.373293	0.688464	0.526018
6	3.324137	0.0	0.0	-0.444251	0.641305	0.588728
7	3.898849	0.0	0.0	-0.511529	0.599578	0.640506
8	4.473562	0.0	0.0	-0.575080	0.562660	0.683414
9	5.048275	0.0	0.0	-0.634980	0.529946	0.719157
10	5.622988	0.0	0.0	-0.691316	0.500916	0.749083
11	6.197701	0.0	0.0	-0.743442	0.475475	0.773924
12	6.197701	2.173563	0.194000	-0.764726	0.465461	0.783347
13	5.622988	2.059770	0.194000	-0.711211	0.491049	0.758871
14	5.048275	1.945976	0.194000	-0.654300	0.519806	0.729802
15	4.473562	1.832184	0.194000	-0.594051	0.552086	0.695201
16	3.898849	1.712643	0.194000	-0.529214	0.589068	0.652999
17	3.324137	1.604597	0.194000	-0.458492	0.632236	0.600278
18	2.749424	1.490804	0.188000	-0.384162	0.681021	0.536210
19	2.174711	1.382757	0.181000	-0.310104	0.733371	0.462167
20	1.599998	1.281609	0.181000	-0.231583	0.793277	0.370711
21	1.025287	1.173563	0.181000	-0.136993	0.871977	0.239657
22	0.450575	1.066667	0.157000	-0.024942	0.975366	0.048661
23	0.0	1.000000	0.067000	0.0	1.000000	0.0
24	0.0	0.0	0.0	0.0	1.000000	0.0

REFERENCE QUANTITIES,VELOCITY= 73.79999VOLUME FLOWRATE= 60.22078CP MULTIPLIER= 1.13673
START VALUES,UT= 2.29609 UB= 20.61395 DELTA= 0.12526 DELST= 0.02700 H= 1.53000

ITERATION 0,
LOWER WALL B.C.
CALCULATION.
↓

X	DSTAR	H	HSEP	CP	DELT	UB	UT	UI	CF/2	DQDX/UI
0.0	0.02700	1.530	2.275	0.0	0.12526	20.61395	2.29609	73.79999	0.000968	0.02218
0.08100	0.02948	1.556	2.289	0.036	0.13141	21.84360	2.18519	72.45232	0.000910	0.02297
0.18900	0.03302	1.596	2.309	0.081	0.13989	23.43257	2.04417	70.75967	0.000835	0.02405
0.35100	0.03858	1.655	2.337	0.136	0.15292	25.52373	1.86071	68.58080	0.000736	0.02556
0.51300	0.04403	1.712	2.364	0.180	0.16505	26.98830	1.72210	66.84709	0.000653	0.02687
0.72900	0.04919	1.742	2.377	0.219	0.17962	27.87650	1.61063	65.22430	0.000610	0.02821
1.05300	0.05861	1.805	2.405	0.272	0.20333	29.20454	1.45255	62.96014	0.000532	0.03022
1.59300	0.07902	1.946	2.463	0.351	0.24957	31.87627	1.18573	59.45618	0.000398	0.03375
2.13300	0.10456	2.112	2.527	0.411	0.30306	34.42004	0.95793	56.65666	0.000286	0.03699
2.45700	0.12444	2.250	2.576	0.441	0.34064	36.34485	0.81581	55.19545	0.000218	0.03887
2.99699	0.16145	2.504	2.657	0.480	0.40704	39.24753	0.61192	53.23277	0.000132	0.04163
3.53699	0.20258	2.775	2.735	0.507	0.47916	41.71574	0.44951	51.82753	0.000075	0.04379
3.86099	0.23688	3.058	2.807	0.519	0.53026	44.18808	0.32028	51.20825	0.000039	0.04481
4.07699	0.25975	3.255	2.853	0.525	0.56423	45.62479	0.24417	50.84634	0.000023	0.04543
4.23899	0.27707	3.404	2.886	0.529	0.58993	46.61180	0.19285	50.64226	0.000014	0.04578
4.40099	0.29449	3.550	2.916	0.532	0.61601	47.57707	0.14308	50.47954	0.000008	0.04607
4.72499	0.33262	3.912	2.984	0.536	0.67052	49.70094	0.03286	50.25757	0.000000	0.04648
4.94099	0.34763	4.334	3.054	0.565	0.67759	50.08832	-0.03253	48.66046	-0.000000	0.05596
5.07599	0.36375	4.478	3.075	0.567	0.70216	50.53050	-0.04398	48.56383	-0.000001	0.05939
5.15698	0.37330	4.555	3.086	0.568	0.71707	50.73871	-0.04972	48.49809	-0.000001	0.06124
5.23798	0.38286	4.631	3.097	0.569	0.73203	50.91206	-0.05480	48.42729	-0.000001	0.06297
5.37298	0.39815	4.724	3.109	0.572	0.75707	51.09520	-0.06131	48.29327	-0.000002	0.06563
5.39200	0.40024	4.757	3.111	0.572	0.76061	51.13803	-0.06337	48.27277	-0.000002	0.06652

DIFFUSER WIDTH FOR THE FIRST ITERATION (EFFECTIVE WIDTH FOR LINEAR CORE PROFILE METHOD)

K	SW(K)	WI(K)
1	0.0	8.70000E-01
2	3.92000E-01	9.28004E-01
3	8.92000E-01	1.02100E 00
4	1.39200E 00	1.11500E 00
5	1.89200E 00	1.20301E 00
6	2.39200E 00	1.29701E 00
7	2.89200E 00	1.39600E 00
8	3.39200E 00	1.49000E 00
9	3.89200E 00	1.59400E 00
10	4.39200E 00	1.69300E 00
11	4.89200E 00	1.79200E 00
12	5.39200E 00	1.89100E 00

SOLVE 2-D LAPLACE'S EQN.
IN NEW E.F.C.

LENGTH SCALE = 8.70000E-01
VELOCITY SC ALE = 7.38000E 01
NORMALIZED INLET VELOCITY= 1.00000E 00
NORMALIZED EXIT VELOCITY = 5.85125E-01

NORMALIZED SOLUTION

#	XC	YC	ALPHA	LN(VEL)	VELOCITY	CP
1	0.450575	0.046006	0.029704	-0.050225	0.951016	0.095569
2	1.025287	0.061647	0.029782	-0.115680	0.890760	0.206546
3	1.599998	0.081674	0.038870	-0.186246	0.830070	0.310984
4	2.174711	0.106445	0.048567	-0.251459	0.777665	0.395237
5	2.749424	0.138208	0.062455	-0.307929	0.734968	0.459822
6	3.324137	0.177118	0.068711	-0.360674	0.697206	0.513904
7	3.898849	0.218960	0.085339	-0.409729	0.663830	0.559330
8	4.473562	0.276094	0.105965	-0.441125	0.643312	0.586149
9	5.048275	0.337302	0.102038	-0.469455	0.625343	0.608947
10	5.622988	0.394964	0.105478	-0.505317	0.603314	0.636012
11	6.197701	0.460051	0.115532	-0.534274	0.586094	0.656493
12	6.197701	2.173563	0.194000	-0.537586	0.584156	0.658761
13	5.622988	2.059770	0.194000	-0.509764	0.600637	0.639235
14	5.048275	1.945976	0.194000	-0.478526	0.619696	0.615977
15	4.473562	1.832184	0.194000	-0.443975	0.641481	0.588502
16	3.898849	1.712643	0.194000	-0.404601	0.667243	0.554787
17	3.324137	1.604597	0.194000	-0.357627	0.699334	0.510932
18	2.749424	1.490804	0.188000	-0.304701	0.737344	0.456324
19	2.174711	1.382757	0.181000	-0.249714	0.779024	0.393122
20	1.599998	1.281609	0.181000	-0.187751	0.828821	0.313055
21	1.025287	1.173563	0.181000	-0.107953	0.897669	0.194190
22	0.450575	1.066667	0.157000	-0.010838	0.989221	0.021442
23	0.0	1.000000	0.067000	0.0	1.000000	0.0
24	0.0	0.031034	0.034681	0.0	1.000000	0.0

COMPARE $C_{P_{1D}}$ AND $C_{P_{2D}}$

VELOCITY COMPARISON

NODE#	1-D VEL	2-D VEL	(1D-2D VEL)	CP 1-D	CP 2-D	(1D-2D CP)
1	6.80979E 01	7.01849E 01	-2.08699E 00	1.48557E-01	9.55695E-02	5.29878E-02
2	6.40809E 01	6.57381E 01	-1.65717E 00	2.46046E-01	2.06547E-01	3.94996E-02
3	6.07154E 01	6.12591E 01	-5.43716E-01	3.23161E-01	3.10985E-01	1.21766E-02
4	5.78639E 01	5.73917E 01	4.72183E-01	3.85245E-01	3.95237E-01	-9.99218E-03
5	5.54711E 01	5.42406E 01	1.23055E 00	4.35035E-01	4.59823E-01	-2.47878E-02
6	5.35814E 01	5.14538E 01	2.12758E 00	4.72874E-01	5.13904E-01	-4.10306E-02
7	5.21724E 01	4.89906E 01	3.18172E 00	5.00232E-01	5.59330E-01	-5.90977E-02
8	5.11527E 01	4.74764E 01	3.67630E 00	5.19575E-01	5.86149E-01	-6.65739E-02
9	5.04823E 01	4.61503E 01	4.33203E 00	5.32086E-01	6.08947E-01	-7.68604E-02
10	4.89420E 01	4.45246E 01	4.41739E 00	5.60205E-01	6.36012E-01	-7.58070E-02
11	4.82728E 01	4.32538E 01	5.01901E 00	5.72150E-01	6.56493E-01	-8.43437E-02
12	3.43510E 01	4.31107E 01	-8.75977E 00	7.83347E-01	6.58761E-01	1.24585E-01
13	3.62394E 01	4.43270E 01	-8.08762E 00	7.58871E-01	6.39235E-01	1.19636E-01
14	3.83616E 01	4.57336E 01	-7.37192E 00	7.29802E-01	6.15977E-01	1.13825E-01
15	4.07439E 01	4.73413E 01	-6.59737E 00	6.95201E-01	5.88502E-01	1.06699E-01
16	4.34732E 01	4.92425E 01	-5.76932E 00	6.53000E-01	5.54787E-01	9.82121E-02
17	4.66590E 01	5.16108E 01	-4.95183E 00	6.00278E-01	5.10932E-01	8.93456E-02
18	5.02593E 01	5.44160E 01	-4.15662E 00	5.36211E-01	4.56324E-01	7.98862E-02
19	5.41228E 01	5.74919E 01	-3.36919E 00	4.62167E-01	3.93122E-01	6.90452E-02
20	5.85438E 01	6.11670E 01	-2.62315E 00	3.70712E-01	3.13056E-01	5.76560E-02
21	6.43519E 01	6.62480E 01	-1.89613E 00	2.39657E-01	1.94190E-01	4.54673E-02
22	7.19820E 01	7.30045E 01	-1.02248E 00	4.86613E-02	2.14426E-02	2.72187E-02
23	7.38000E 01	7.38000E 01	0.0	0.0	0.0	0.0
24	7.38000E 01	7.38000E 01	0.0	0.0	0.0	0.0

LARGEST ABSOLUTE ERRORS, VELOCITY= 8.75977E 00(FT/SEC)   CPERR= 1.24585E-01

LARGEST ERROR

******ITERATION NUMBER 1 ******

******LOWER WALL VALUES******

BOUNDARY LAYER CALCULATION-PRESCRIBED PRESSURE GRADIENT

1	0.0	8.64000E-01	7.38000E 01
2	3.92000E-01	9.20419E-01	7.20045E 01
3	8.92000E-01	1.01138E 00	6.62480E 01
4	1.39200E 00	1.10335E 00	6.11670E 01
5	1.89200E 00	1.18932E 00	5.74919E 01
6	2.39200E 00	1.28129E 00	5.44160E 01
7	2.89200E 00	1.37825E 00	5.16108E 01
8	3.39200E 00	1.47021E 00	4.92425E 01
9	3.89200E 00	1.57216E 00	4.73413E 01
10	4.39200E 00	1.66912E 00	4.57336E 01
11	4.89200E 00	1.76608E 00	4.43270E 01
12	5.39200E 00	1.86305E 00	4.31107E 01

START VALUES,UT= 2.29609 UB= 20.61395 DELTA= 0.12526 DELST= 0.02700 H= 1.53000

X	DSTAR	H	HSEP	CP	DELT	UB	UT	UI	CF/2	DQDX/UI
0.0	0.02700	1.530	2.275	0.0	0.12526	20.61395	2.29609	73.79999	0.000968	0.01991
0.08100	0.02833	1.539	2.281	0.020	0.12924	21.08925	2.24138	73.04701	0.000942	0.02032
0.18300	0.03022	1.556	2.289	0.047	0.13475	21.73604	2.16821	72.04544	0.000906	0.02089
0.35100	0.03335	1.583	2.303	0.086	0.14358	22.73886	2.05835	70.55685	0.000851	0.02179
0.62100	0.03937	1.634	2.327	0.147	0.15959	24.46271	1.87703	68.14426	0.000759	0.02339
0.94500	0.04815	1.711	2.361	0.218	0.18153	26.81345	1.65077	65.25996	0.000643	0.02554
1.26900	0.06247	1.843	2.421	0.287	0.21081	30.23279	1.38239	62.32993	0.000494	0.02803

****H>.9*HSEP OR H<HO AT X= 1.91700E 00

CONTINUE B.L. CALCULATION WITH 1-D STREAMTUBE ITERATION

REFERENCE QUANTITIES,VELOCITY= 73.79999VOLUME FLOWRATE= 62.21338CP MULTIPLIER= 1.06508

X	DSTAR	H	HSEP	CP	DELT	UB	UT	UI	CF/2	DQDX/UI
1.26900	0.06247	1.843	2.421	0.287	0.21081	30.23279	1.38239	62.32993	0.000492	0.02803
1.45640	0.07109	1.920	2.453	0.318	0.22813	31.87463	1.25523	60.96558	0.000424	0.02933
1.70626	0.08342	2.026	2.495	0.353	0.25209	33.91716	1.10083	59.35297	0.000344	0.03101
2.08105	0.10654	2.234	2.570	0.397	0.29346	37.35674	0.87101	57.30792	0.000231	0.03330
2.45585	0.13483	2.524	2.652	0.432	0.34155	41.50935	0.62809	55.64018	0.000131	0.03537
2.83064	0.17587	2.996	2.791	0.452	0.39816	46.45537	0.37792	54.62926	0.000048	0.03680
3.45530	0.25268	4.130	3.073	0.479	0.48809	54.15311	-0.01550	53.27026	-0.000000	0.05188
4.07996	0.33983	5.873	3.242	0.498	0.61355	58.59552	-0.13649	52.29384	-0.000007	0.07299
4.45475	0.39129	6.334	3.284	0.506	0.69592	59.08641	-0.15409	51.87451	-0.000009	0.07930
4.64215	0.41744	6.518	3.300	0.510	0.73850	59.17387	-0.15991	51.65234	-0.000010	0.08163
4.89201	0.45269	6.732	3.318	0.516	0.79622	59.20540	-0.16595	51.35445	-0.000010	0.08420
5.26680	0.50654	6.985	3.337	0.524	0.88527	59.10839	-0.17185	50.91940	-0.000011	0.08715
5.39200	0.52479	7.054	3.343	0.527	0.91567	58.97451	-0.17240	50.77721	-0.000012	0.08819

******ITERATION NUMBER 8 ******

(ITERATIONS 2 THROUGH 7
ARE NOT SHOWN)

******UPPER WALL VALUES******

BOUNDARY LAYER CALCULATION-PRESCRIBED PRESSURE GRADIENT
START VALUES,UT= 1.97868 UB= 37.87976 DELTA= 0.01863 DELST= 0.00600 H= 2.00000

X	DSTAR	H	HSEP	CP	DELT	UB	UT	UI	CF/2	DQDX/UI
0.0	0.00600	2.000	2.475	0.0	0.01863	37.87976	1.97868	73.79899	0.000719	0.03762
0.01800	0.00566	1.873	2.425	0.000	0.01897	33.32161	2.19403	73.79919	0.000884	0.03756
0.03300	0.00543	1.794	2.391	0.000	0.01931	30.07425	2.34480	73.79826	0.001010	0.03746
0.04200	0.00534	1.759	2.376	0.000	0.01955	28.52841	2.41440	73.79750	0.001070	0.03738
0.05700	0.00521	1.708	2.353	0.000	0.01999	26.22449	2.51652	73.79564	0.001163	0.03722
0.07500	0.00508	1.657	2.329	0.000	0.02054	23.80580	2.62155	73.79221	0.001267	0.03700
0.09300	0.00503	1.622	2.312	0.000	0.02113	21.85382	2.70322	73.78711	0.001338	0.03674
0.12300	0.00493	1.570	2.286	0.001	0.02214	19.12593	2.81353	73.77364	0.001454	0.03627
0.15900	0.00483	1.520	2.261	0.001	0.02338	16.38484	2.92007	73.74667	0.001576	0.03568
0.17700	0.00485	1.505	2.253	0.002	0.02401	15.32236	2.95853	73.72762	0.001610	0.03538
0.20100	0.00484	1.484	2.242	0.003	0.02487	14.05459	3.00273	73.69518	0.001660	0.03499
0.23700	0.00487	1.459	2.229	0.005	0.02616	12.50180	3.05243	73.62862	0.001719	0.03442
0.26100	0.00488	1.445	2.221	0.006	0.02702	11.56008	3.08106	73.57001	0.001754	0.03407
0.27900	0.00491	1.436	2.215	0.008	0.02768	10.97247	3.09681	73.51730	0.001775	0.03382
0.31500	0.00498	1.421	2.207	0.011	0.02900	9.99039	3.11871	73.38603	0.001806	0.03336
0.35100	0.00508	1.410	2.201	0.016	0.03034	9.16459	3.13248	73.21368	0.001824	0.03296
0.38700	0.00517	1.399	2.195	0.022	0.03171	8.53012	3.13575	72.99164	0.001846	0.03264
0.45900	0.00546	1.386	2.188	0.038	0.03456	7.66866	3.11792	72.37621	0.001856	0.03221
0.51900	0.00574	1.378	2.183	0.055	0.03703	7.15212	3.09102	71.72420	0.001857	0.03203
0.55500	0.00594	1.375	2.182	0.067	0.03857	6.98907	3.06629	71.29031	0.001850	0.03198
0.61500	0.00629	1.371	2.180	0.087	0.04121	6.80902	3.02025	70.51845	0.001834	0.03197
0.68700	0.00675	1.368	2.179	0.112	0.04447	6.69789	2.96008	69.54543	0.001813	0.03207
0.75900	0.00727	1.368	2.179	0.137	0.04785	6.69473	2.89667	68.56128	0.001784	0.03225
0.87900	0.00814	1.366	2.179	0.176	0.05360	6.70984	2.79807	67.00873	0.001744	0.03259
1.02300	0.00913	1.362	2.178	0.212	0.06043	6.61754	2.70035	65.50165	0.001693	0.03284
1.09499	0.00958	1.360	2.177	0.227	0.06376	6.49677	2.66514	64.87587	0.001688	0.03289
1.19099	0.01016	1.357	2.175	0.245	0.06815	6.32381	2.62311	64.12692	0.001673	0.03292
1.33499	0.01099	1.352	2.173	0.269	0.07467	6.05514	2.57030	63.14095	0.001657	0.03291
1.57499	0.01226	1.342	2.168	0.298	0.08526	5.57627	2.50175	61.83488	0.001637	0.03272
1.86299	0.01359	1.330	2.162	0.324	0.09758	5.00108	2.44032	60.66780	0.001610	0.03231
2.43899	0.01547	1.309	2.149	0.349	0.11936	3.73375	2.39950	59.54825	0.001624	0.03070
2.91898	0.01646	1.290	2.138	0.353	0.13579	2.72876	2.39041	59.34123	0.001623	0.02899
3.20698	0.01684	1.281	2.132	0.351	0.14410	2.17483	2.40298	59.46315	0.001633	0.02786
3.68698	0.01729	1.269	2.124	0.342	0.15674	1.36362	2.42887	59.87518	0.001646	0.02599
4.16698	0.01784	1.260	2.118	0.334	0.16851	0.81738	2.44647	60.22838	0.001650	0.02438
4.45498	0.01832	1.257	2.116	0.332	0.17571	0.63767	2.44662	60.30255	0.001646	0.02362
4.59898	0.01860	1.255	2.116	0.332	0.17939	0.58490	2.44377	60.29683	0.001643	0.02329
4.71897	0.01886	1.255	2.115	0.333	0.18247	0.55125	2.44058	60.27385	0.001640	0.02304
4.79097	0.01902	1.254	2.115	0.333	0.18434	0.53915	2.43804	60.25414	0.001637	0.02290
4.91097	0.01930	1.254	2.115	0.334	0.18748	0.52631	2.43325	60.21127	0.001633	0.02268
5.05497	0.01965	1.253	2.114	0.336	0.19128	0.52329	2.42645	60.13890	0.001628	0.02243
5.12696	0.01985	1.253	2.114	0.337	0.19321	0.52967	2.42237	60.09453	0.001625	0.02232
5.22296	0.02011	1.253	2.114	0.338	0.19579	0.54140	2.41669	60.03030	0.001621	0.02218
5.36696	0.02051	1.253	2.114	0.341	0.19967	0.56529	2.40775	59.92775	0.001614	0.02198
5.49833	0.02086	1.252	2.114	0.343	0.20355	0.59166	2.39874	59.83882	0.001607	0.02178

```
LENGTH SCALE                    =    8.70000E-01
VELOCITY SC ALE                 =    7.38000E 01
NORMALIZED INLET VELOCITY=           1.00000E 00
NORMALIZED EXIT VELOCITY =           8.11395E-01
```

NORMALIZED SOLUTION

#	XC	YC	ALPHA	LN(VEL)	VELCCITY	CP
1	0.450575	0.040786	0.022096	-0.056629	0.944945	0.107080
2	1.025287	0.055737	0.035139	-0.123965	0.883410	0.219586
3	1.599998	0.083266	0.061258	-0.189809	0.827117	0.315878
4	2.174711	0.127712	0.096557	-0.245820	0.782063	0.388378
5	2.749424	0.199040	0.159152	-0.271795	0.762010	0.419340
6	3.324137	0.309360	0.203296	-0.261927	0.769567	0.407767
7	3.898849	0.433420	0.231307	-0.231878	0.793043	0.371083
8	4.473562	0.574321	0.224615	-0.195849	0.822136	0.324092
9	5.048275	0.691334	0.189394	-0.186947	0.829487	0.311951
10	5.622988	0.799218	0.181765	-0.197864	0.820481	0.326811
11	6.197701	0.903222	0.177993	-0.205148	0.814526	0.336547
12	6.202322	2.150040	0.191179	-0.212867	0.808264	0.346709
13	5.627298	2.037821	0.191480	-0.207302	0.812774	0.339398
14	5.052339	1.925288	0.192176	-0.205269	0.814428	0.336707
15	4.477460	1.812342	0.192806	-0.208136	0.812097	0.340499
16	3.902632	1.693392	0.193032	-0.215470	0.806162	0.350103
17	3.327791	1.585991	0.192487	-0.219934	0.802572	0.355878
18	2.752742	1.473358	0.185396	-0.215569	0.806083	0.350231
19	2.177568	1.367146	0.177037	-0.199459	0.819174	0.328954
20	1.602361	1.268698	0.175651	-0.163627	0.849059	0.279099
21	1.027010	1.164148	0.173948	-0.101051	0.903887	0.182988
22	0.451510	1.060760	0.155094	-0.011904	0.988167	0.023526
23	0.000462	0.993119	0.070652	0.0	1.000000	0.0
24	0.0	0.031034	0.021450	0.0	1.000000	0.0

VELOCITY COMPARISON

NODE#	1-D VEL	2-D VEL	(1D-2D VEL)	CP 1-D	CP 2-D	(1D-2D CP)
1	6.97048E 01	6.97369E 01	-3.20587E-02	1.07901E-01	1.07080E-01	8.20875E-04
2	6.53235E 01	6.51957E 01	1.27808E-01	2.16523E-01	2.19586E-01	-3.06284E-03
3	6.11120E 01	6.10412E 01	7.08160E-02	3.14290E-01	3.15878E-01	-1.58828E-03
4	5.76931E 01	5.77162E 01	-2.31476E-02	3.88869E-01	3.88378E-01	4.90487E-04
5	5.61394E 01	5.62363E 01	-9.69238E-02	4.21340E-01	4.19341E-01	1.99980E-03
6	5.64779E 01	5.67940E 01	-3.16101E-01	4.14341E-01	4.07767E-01	6.57409E-03
7	5.83882E 01	5.85266E 01	-1.38397E-01	3.74054E-01	3.71083E-01	2.97093E-03
8	6.05881E 01	6.06736E 01	-8.55560E-02	3.25997E-01	3.24092E-01	1.90490E-03
9	6.11321E 01	6.12162E 01	-8.40759E-02	3.13839E-01	3.11951E-01	1.88863E-03
10	6.09038E 01	6.05515E 01	3.52325E-01	3.18954E-01	3.26811E-01	-7.85691E-03
11	6.04070E 01	6.01120E 01	2.94998E-01	3.30019E-01	3.36547E-01	-6.52772E-03
12	5.98388E 01	5.96499E 01	1.88965E-01	3.42564E-01	3.46710E-01	-4.14568E-03
13	6.01796E 01	5.99827E 01	1.96869E-01	3.35055E-01	3.39398E-01	-4.34345E-03
14	6.03034E 01	6.01048E 01	1.98608E-01	3.32316E-01	3.36707E-01	-4.39084E-03
15	6.00840E 01	5.99327E 01	1.51260E-01	3.37166E-01	3.40499E-01	-3.33321E-03
16	5.96668E 01	5.94948E 01	1.72058E-01	3.46338E-01	3.50103E-01	-3.76439E-03
17	5.93435E 01	5.92298E 01	1.13693E-01	3.53404E-01	3.55879E-01	-2.47526E-03
18	5.95591E 01	5.94889E 01	7.02362E-02	3.48696E-01	3.50231E-01	-1.53524E-03
19	6.05180E 01	6.04550E 01	6.30188E-02	3.27555E-01	3.28954E-01	-1.39982E-03
20	6.26853E 01	6.26605E 01	2.47498E-02	2.78530E-01	2.79100E-01	-5.69642E-04
21	6.67122E 01	6.67068E 01	5.32532E-03	1.82858E-01	1.82989E-01	-1.30534E-04
22	7.29244E 01	7.29267E 01	-2.31934E-03	2.35886E-02	2.35265E-02	6.21080E-05
23	7.38000E 01	7.38000E 01	0.0	0.0	0.0	0.0
24	7.38000E 01	7.38000E 01	0.0	0.0	0.0	0.0

LARGEST ABSOLUTE ERRORS, VELOCITY= 3.52325E-01(FT/SEC) CPERR= 7.85691E-03

*******END OF PROGRAM*******

LISTING OF PROGRAM TSTALL

```
//STAND JOB 'J15$D1,531','S.GHOSE',CLASS=E,TIME=(,30)
//  EXEC  FORTCL,PARM.FORT='OPT=2'
//FORT.SYSIN  DD  *
C------MAIN ROUTINE TO CALCULATE THE PERFORMANCE OF 1-D AND 2-D
C       DIFFUSERS, OPERATING IN THE UNSTALLED AND TRANSITORY STALL
C       REGIMES. THE SCHEME USES A NEW TURBULENT BOUNDARY LAYER
C       PREDICTION METHOD, TOGETHER WITH SIMULTANEOUS ITERATION
C       BETWEEN THE CORE AND THE BL. AN ADDITIONAL INVISCID MATCHING
C       TECHNIQUE IS USED TO MATCH THE 2-D FLOWFIELD WITH THE
C       THE STREAMTUBE ENVELOPING THE BL.
C       WRITTEN BY SANJOY GHOSE, MECHANICAL ENGG DEPT, STANFORD UNIV.
C       LAST REVISION MADE ON NOV 1,1976.
C
      INTEGER HEAD(20),GEOM(4),CORE(4),PROB(4),GEOMT,CORET,PROBT,ID1,
     $ID2,ID3
      COMMON/ODE1S/JSTRTS,JENDS,NDIM,SW(90),WI(90),DWI(90),DDWI(90),
     $DS(90)
      COMMON/NSTD/ID1,ID2,ID3,NST,SWT(90)
      DATA GEOM/'STDD','    ','NSTD','HALF'/,CORE/'ONED','TWOD',
     $'XXXX'/,PROB/'TBLP','NOBL','DIFF','    '/
      NDIM=90
  100 READ(5,900,END=800) (HEAD(J),J=1,20)
      WRITE(6,910) (HEAD(J),J=1,20)
      READ(5,920)GEOMT,PROBT,CORET
      WRITE(6,930)GEOMT,PROBT,CORET
      ID1=0
      ID2=0
      ID3=0
      NST=0
      DO 105 J=1,4
      IF(GEOM(J).EQ.GEOMT)ID1=J
      IF(PROB(J).EQ.PROBT)ID2=J
  105 IF(CORE(J).EQ.CORET)ID3=J
      IF(MAX0(ID1,ID2,ID3).GT.0)GO TO 110
      WRITE(6,940)
      STOP
  110 IF(ID1.EQ.3)GO TO 112
  111 CALL STAND
      GO TO 120
  112 IF(ID2.EQ.1)GO TO 120
      NST=1
      CALL NSTAND
  120 GO TO(121,122,123,123),ID2
  121 CALL PSTEST
      GO TO 100
  122 CALL INVCID
      GO TO 100
  123 IF(ID3.EQ.1)CALL DIFF1D
      IF(ID3.NE.1)CALL DIFF2D
      GO TO 100
  800 WRITE(6,950)
      STOP
  900 FORMAT(20A4)
  910 FORMAT('1',20A4//)
  920 FORMAT(3(6X,A4))
  930 FORMAT(' GEOMETRY= ',A4,'     PROBLEM TYPE= ',A4,
     $'     CORE VELOCITY PROFILE= ',A4//)
  940 FORMAT(' ****CANNOT RECOGNIZE PROBLEM TYPE--CHECK CARD # 2 ***'//)
  950 FORMAT('1******END OF PROGRAM*******')
      END
```

```
      SUBROUTINE ADAMS(DX,NEQ,DNAME,IRUNGE)
C------USES 4TH ORDER ADAMS-MOULTON PREDICTOR-CORRECTOR METHOD
C       SO SOLVE A SET OF FIRST ORDER ODE'S EXPRESSED IN THE FORM
C       Y'=F(X,Y,....). USES A 4TH ORDER RUNGE-KUTTA METHOD FOR
C       STARTING (AND RESTARTING).  CALL TO THIS ROUTINE REURNS VALUES
C       OF THE FUNCTION AT X+DX, GIVEN VALUES AT X.
C       RATE MATRIX CONTAINS DERIVATIVES FOR THE LAST 4 STEPS.
C       THE ROW RATE(1,J),J=1,NEQ HAS VALUES FOR STEP N,
C       ROW RATE(2,J) FOR STEP (N-1),ETC. VALS(J),J=1,NEQ ARE
C       VALUES OF THE VARIABLES.
C       SET IRUNGE=1 IN THE CALLING ROUTINE TO PROVIDE STARTING
C       VALUES VIA CALL TO RKS4. IRUNGE=5 CAUSES ADAMS-MOULTON
C       4TH ORDER PREDICTOR-CORRECTOR METHOD TO BE INVOKED.
C
      EXTERNAL DNAME
      REAL VALP(8),VALC(8),RATEP(8)
      COMMON/ADAM1/X ,VALS(4),RATES(4),RATE(4,8)
      COMMON/ODE1S/JSTRTS,JENDS,NDIM,SW(90),WI(90),DWI(90),DDWI(90),
     *DS(90)
      DATA STEPER/1.E-3/
      IF(NEQ.LE.8)GO TO 100
      WRITE(6,900)NEQ
      STOP
  100 CONTINUE
      ERRLOW=STEPER/5.0
  110 IF(IRUNGE.EQ.5)GO TO 200
      CALL RKS4(DX,NEQ,DNAME)
      DO 140 J=1,NEQ
  140 RATE(IRUNGE,J)=RATES(J)
      IRUNGE=IRUNGE+1
      RETURN
C
C------START ADAMS-BASHFORTH PREDICTOR ROUTINE
  200 X=X+DX
      DH=DX/24.0
      DO 230 J=1,NEQ
      VALP(J)=VALS(J)+DH*(55.0*RATE(1,J)-59.0*RATE(2,J)+37.0*RATE(3,J)
     $-9.0*RATE(4,J))
  230 CONTINUE
C
      CALL DNAME(X,VALP,RATEP)
C
C------BEGIN ADAMS-MOULTON CORRECTOR ITERATION.
      NITER=0
      DERR=0.0
  240 DO 250 J=1,NEQ
      NITER=NITER+1
      VALC(J)=VALS(J)+DH*(9.0*RATEP(J)+19.0*RATE(1,J)-5.0*RATE(2,J)
     $+RATE(3,J))
  250 DERR=AMAX1(DERR,ABS(VALP(J)-VALC(J))/(14.0*ABS(VALC(J))))
      IF(DERR.LE.STEPER)GO TO 270
      IF(NITER.GE.2)GO TO 310
      CALL DNAME(X,VALC,RATEP)
      GO TO 240
  270 DO 300 I=1,3
      IR=5-I
      DO 290 J=1,NEQ
  290 RATE(IR,J)=RATE(IR-1,J)
  300 CONTINUE
      DO 305 J=1,NEQ
      VALS(J)=VALC(J)
```

```
      RATES(J)=RATEP(J)
  305 RATE(1,J)=RATEP(J)
      IRUNGE=5
      IF(DERR.GT.ERRLOW) RETURN
      DX=2.0*DX
      IRUNGE=1
      RETURN
C
C------UNABLE TO CONVERGE IN 2 ITERATIONS OF THE CORRECTOR. CUT
C       STEPSIZE IN HALF AND RESTART.
  310 X=X-DX
      DX=0.5*DX
      IRUNGE=1
      GO TO 110
  900 FORMAT(' NUMBER OF EQUATIONS EXCEEDS ARRAY BOUNDS, AS NEQ= ',I5,
     $' INCREASE DIMENSIONS OF SCRATCH ARRAYS IN ADAMS AND RKS4')
      END
      SUBROUTINE DIFF1D
C------ROUTINE TO TEST 1-D CORE DIFFUSERS IN SIMULTANEOUS ITERATION.
      COMMON/DER2/DELT,UB,UT,UI,VT,VB,UDUI,TAUM,H,THETA,DELST,CFD2,
     $VISCOS,NBL
      COMMON/ODE1S/JSTRTS,JENDS,NDIM,SW(90),WI(90),DWI(90),DDWI(90),
     $DS(90)
      COMMON/SPLYN/XX,WT,DWT,DDWT,ISETUP,KMID
      COMMON/TEMP1/XCMX,IWALLV
      XX=0.0
      NBL=2
      CALL TBLSI(0)
      WRITE(6,930)
  930 FORMAT('0**********END OF ROUTINE DIFF1D********')
      RETURN
      END
      SUBROUTINE FACTOR(A,W,IPIVOT,D,N,IFLAG)
C
C------THIS SUBROUTINE PERFORMS A L-U DECOMPOSITION ON THE
C       GIVEN MATRIX A(N,N), AND RETURNS THE MATRIX W. IPIVOT
C      . IS A VECTOR CONTAINING THE PIVOTING ORDER.
C
      REAL A(N,N),W(N,N),D(N)
      INTEGER IPIVOT(N)
      IFLAG=1
C       INITIALIZE W,IPIVOT,D
      DO 10 I=1,N
      IPIVOT(I)=I
      ROWMAX=0.0
      DO 9 J=1,N
      W(I,J)=A(I,J)
      ROWMAX=AMAX1(ROWMAX,ABS(W(I,J)))
    9 CONTINUE
      IF(ROWMAX.EQ.0.0)GO TO 999
      D(I)=ROWMAX
   10 CONTINUE
C------GAUSS ELIMINATION WITH SCALED PARTIAL PIVOTING
      NM1=N-1
      IF(NM1.EQ.0)RETURN
      DO 20 K=1,NM1
      J=K
      KP1=K+1
      IP=IPIVOT(K)
      COLMAX=ABS(W(IP,K))/D(IP)
      DO 11 I=KP1,N
```

```
      IP=IPIVOT(I)
      AWIKOV=ABS(W(IP,K))/D(IP)
      IF(AWIKOV.LE.COLMAX)GO TO 11
      COLMAX=AWIKOV
      J=I
   11 CONTINUE
      IF(COLMAX.EQ.0.0)GO TO 999
C
C
      IPK=IPIVOT(J)
      IPIVOT(J)=IPIVOT(K)
      IPIVOT(K)=IPK
      DO 20 I=KP1,N
      IP=IPIVOT(I)
      W(IP,K)=W(IP,K)/W(IPK,K)
      RATIO=-W(IP,K)
      DO 20 J=KP1,N
      W(IP,J)=RATIO*W(IPK,J)+W(IP,J)
   20 CONTINUE
      IF(W(IP,N).EQ.0.0)GO TO 999
      RETURN
C
C------SET IFLAG=2 TO INDICATE INABILITY TO FACTORIZE MATRIX.
C
  999 IFLAG=2
      WRITE(6,9999)
 9999 FORMAT( 1H0,'****UNABLE TO COMPLETE L-U DECOMPOSITION OF MATRIX')
      STOP
      END
      SUBROUTINE PFSL
      REAL*8 A(86,87)
      COMPLEX C(91)
      COMPLEX CMPLX,ICMPLX
      REAL LNV(90),VEL(90),X0(120),Y0(120)
      COMMON/EFCVAL/XC(90),YC(90),AL(90),LNV
      COMMON/GEOM1/XD,XL,TH,WIDTH,X1,B1,SINTH,COSTH,TWOTHR,TWOTH1,
     $SINTH1,COSTH1,AS,WH,XSE,X2,XMAX,XDE
      COMMON/GEOM2/N,NR,NL,NU,NM1,NLC,NRC
      COMMON/INITAL/DELST1,H1,UI1,IPR1
      COMMON/PFSL1/C,A,X0,Y0
C------N=NUMBER OF SEGMENTS(FROM STAND OR NSTAND). XC(J),YC(J), AND
C       AL(J), ARE VALUES AT THE EDGE OF THE EFC AS PASSED VIA
C       COMMON/EFCVAL/.
C       NAROW=NO OF ROWS OF A, NACOL=NO OF COLS OF A.
C
      DATA NAROW,NACOL,VINORM/86,87,1.0/
      NU=N-2
      NUP1=NU+1
      VSCALE=UI1
C------COMPUTE THE BOUNDARY COORDINATES IN THE COMPLEX PLANE.
      DO 200 J=1,N
  200 C(J)=CMPLX(XC(J),YC(J))/XL
C       ASSIGN STARTING VALUES
      LNV(N)=0.0
      LNV(NM1)=0.0
      CALL PERSEG
      CALL GJR(A,NU,1.E-10,NAROW,NACOL)
C------A(NU*NUP1) IS MATRIX OF COEFFS WITH B VECTOR STORED IN LAST COL.
C       ANSWER LNV VECTOR RETURNED IN LAST COL, NUP1.
      DO 725 J=1,NU
  725 LNV(J)=A(J,NUP1)
```

```
      DO 810 J=1,N
  810 VEL(J)=EXP(LNV(J))
      VENORM=(VEL(NLC)+VEL(NRC))/2.0
      WRITE(6,940)XL,VSCALE,VINORM,VENORM
      WRITE(6,942)
      DO 883 J=1,N
      CP=1.0-VEL(J)**2
  883 WRITE(6,943)J,C(J),AL(J),LNV(J),VEL(J),CP
      RETURN
  940 FORMAT(1H1,46X,'LENGTH SCALE',14X,'=',1PE14.5/1H,46X,'VELOCITY SC
     &ALE',12X,'=',1PE14.5,/1H,46X,'NORMALIZED INLET VELOCITY','=',1PE
     &14.5/1H,46X,'NORMALIZED EXIT VELOCITY',' =',1PE14.5/1H0,53X,
     &'NORMALIZED SOLUTION')
  942 FORMAT(1H0,'#',9X,'XC',12X,'YC',12X,'ALPHA',7X,'LN(VEL)',6X,
     &'VELOCITY',5X,'CP')
  943 FORMAT(1H ,I3,6F14.6)
      END
      SUBROUTINE RKS4(H,NEQ,DNAME)
C------SOLVE A SET OF FIRST ORDER ODE'S,  Y'=F(X,Y(1),Y(2),..Y(NEQ))
C       USING FOURTH ORDER RUNGE-KUTTA SCHEME WITH FIXED STEPSIZE H.
C       RETURNS VALUES OF VECTOR Y AT X+DX GIVEN VALUES AT X.
C
      REAL KV(8,4),SPAN(4),YO(4)
      COMMON/ADAM1/X,Y(4),F(4),RATE(4,8)
      EXTERNAL DNAME
      DATA SPAN/0.5,0.5,1.0,1.0/
      XO=X
      DO 200 J=1,NEQ
  200 YO(J)=Y(J)
      DO 400 I=1,4
      CALL DNAME(X,Y,F)
      DO 300 J=1,NEQ
  300 KV(I,J)=H*F(J)
      DO 350 J=1,NEQ
  350 Y(J)=YO(J)+SPAN(I)*KV(I,J)
  400 X=XO+H*SPAN(I)
      DO 500 I=1,NEQ
  500 Y(I)=YO(I)+(KV(1,I)+KV(4,I)+2.0*(KV(2,I)+KV(3,I)))/6.0
      RETURN
      END
      SUBROUTINE SUBST(W,B,X,IPIVOT,N)
C------PERFORM BACK AND FORWARD SUBSTITUTION TO CALCULATE
C       THE UNKNOWN VECTOR X, AS THE SOLUTION OF A*X=B.
C
      REAL W(N,N),B(N),X(N),SUM
      INTEGER IPIVOT(N)
      IF(N.GT.1)GO TO 30
      X(1)=B(1)/W(1,1)
      RETURN
   30 IP=IPIVOT(1)
      X(1)=B(IP)
      DO 50 K=2,N
      IP=IPIVOT(K)
      KM1=K-1
      SUM=0.0
      DO 40 J=1,KM1
   40 SUM=W(IP,J)*X(J)+SUM
   50 X(K)=B(IP)-SUM
C
C
      X(N)=X(N)/W(IP,N)
```

```
      K=N
      DO 70 NP1MK=2,N
      KP1=K
      K=K-1
      IP=IPIVOT(K)
      SUM=0.0
      DO 60 J=KP1,N
   60 SUM=W(IP,J)*X(J)+SUM
   70 X(K)=(X(K)-SUM)/W(IP,K)
      RETURN
      END
      SUBROUTINE  TRIDAG(IF,L,A,B,C,D,V,M,NDIM)
C--------SUBROUTINE FOR SOLVING A SYSTEM OF LINEAR SIMULTANEOUS
C          EQUATIONS HAVING A TRIDIAGONAL COEFFICIENT MATRIX.
C          THE EQUATIONS ARE NUMBERED FROM IF THROUGH L, AND THEIR
C          SUB-DIAGONAL, DIAGONAL, AND SUPER-DIAGONAL COEFFICIENTS
C          ARE STORED IN THE ARRAYS A,B, AND C.  THE COMPUTED
C          SOLUTION VECTOR V(IF)...V(L) IS STORED IN THE ARRAY V.
      REAL A(M),B(M),C(M),D(M),V(NDIM),BETA(101),GAMMA(101)
C--------...COMPUTE INTERMEDIATE ARRAYS BETA AND GAMMA...
      BETA(IF) = B(IF)
      GAMMA(IF) = D(IF)/BETA(IF)
      IFP1 = IF+1
      DO 1 I=IFP1,L
      BETA(I) = B(I)-A(I)*C(I-1)/BETA(I-1)
    1 GAMMA(I) = (D(I)-A(I)*GAMMA(I-1))/BETA(I)
C--------...COMPUTE FINAL SOLUTION VECTOR V...
      V(L) = GAMMA(L)
      LAST = L-IF
      DO 2 K=1,LAST
      I = L-K
    2 V(I) = GAMMA(I)-C(I)*V(I+1)/BETA(I)
      RETURN
      END
      SUBROUTINE CHANGE
      COMMON/GEOM2/N,NR,NL,NU,NM1,NLC,NRC
      COMMON/TEMP1/XCMX,IWALLV
      COMMON/CON/SVAL(90),YVAL(90)
      COMMON/ODE1S/JSTRTS,JENDS,NDIM,SW(90),WI(90),DWI(90),DDWI(90),
     $DS(90)
      COMMON/ODE1U/JSTRTU,JENDU,SWU(90),WIU(90)
      COMMON/ODE2U/JTBLU,STBLU(90),DSTARU(90),UI1DU(90),DELTU(90),
     $UI2DU(90)
      COMMON/ODE2S/JTBLS,STBLS(90),DSTARS(90),UI1DS(90),DELTS(90),
     $UI2DS(90)
      COMMON/SPLYN/XINT,FINT,FPINT,FPPINT,ISETUP,KMID
      REAL HELP(90)
      NMNLC=N-NLC
C------SET UP THE SPLINE COEFFICIENTS FOR SVAL, YVAL.
      XINT=0.0
      ISETUP=0
      KMID=2
C------INTERPOLATE FOR THE VALUES OF YVAL AT THE WALL LOCATIONS SVAL.
      IF(IWALLV.EQ.1)GO TO 500
      CALL SPLINE(SVAL,YVAL,DWI,DDWI,DS,1,JTBLS,NDIM,1)
      DO 100 J=1,NRC
      XINT=SW(J+1)
      CALL SPLINE(SVAL,YVAL,DWI,DDWI,DS,1,JTBLS,NDIM,1)
  100 HELP(J)=FINT
      DO 200 J=1,NRC
  200 YVAL(J)=HELP(J)
```

```
      RETURN
  500 CONTINUE
      CALL SPLINE(SVAL,YVAL,DWI,DDWI,DS,1,JTBLU,NDIM,1)
      DO 600 J=1,NNNLC
      XINT=SWU(J)
      CALL SPLINE(SVAL,YVAL,DWI,DDWI,DS,1,JTBLU,NDIM,1)
  600 HELP(J)=FINT
      DO 800 J=NLC,NM1
  800 YVAL(J)=HELP(N-J)
      RETURN
      END
      SUBROUTINE GJR(A,N,EPS,NAROW,NACOL)
C------DOUBLE PRECISION SOLUTION OF A*X=B. THE VECTOR B IS AUGMENTED ONTO THE
C       LAST COLUMN OF A. THE ANSWER, X, IS ALSO RETURNED IN
C       THE LAST COL OF A.
C       IPIVOT IS A VECTOR CONTAINING THE PIVOTING ORDER.
C
      REAL*8 A(NAROW,NACOL),D(90),B(90),X(90),ROWMAX,COLMAX,AWIKOV,RATIO
     $,SUM,DABS,DMAX1
      INTEGER IPIVOT(90)
      IFLAG=1
      NP1=N+1
C       INITIALIZE IPIVOT,D,B
      DO 10 I=1,N
      IPIVOT(I)=I
      ROWMAX=0.0
      B(I)=A(I,NP1)
      DO 9 J=1,N
      ROWMAX=DMAX1(ROWMAX,DABS(A(I,J)))
    9 CONTINUE
      IF(ROWMAX.EQ.0.0)GO TO 999
      D(I)=ROWMAX
   10 CONTINUE
C------GAUSS ELIMINATION WITH SCALED PARTIAL PIVOTING
      NM1=N-1
      IF(NM1.EQ.0)RETURN
      DO 20 K=1,NM1
      J=K
      KP1=K+1
      IP=IPIVOT(K)
      COLMAX=DABS(A(IP,K))/D(IP)
      DO 11 I=KP1,N
      IP=IPIVOT(I)
      AWIKOV=DABS(A(IP,K))/D(IP)
      IF(AWIKOV.LE.COLMAX)GO TO 11
      COLMAX=AWIKOV
      J=I
   11 CONTINUE
      IF(COLMAX.EQ.0.0)GO TO 999
C
C
      IPK=IPIVOT(J)
      IPIVOT(J)=IPIVOT(K)
      IPIVOT(K)=IPK
      DO 20 I=KP1,N
      IP=IPIVOT(I)
      A(IP,K)=A(IP,K)/A(IPK,K)
      RATIO=-A(IP,K)
      DO 20 J=KP1,N
      A(IP,J)=RATIO*A(IPK,J)+A(IP,J)
   20 CONTINUE
```

```
      IF(A(IP,N).EQ.0.0)GO TO 999
      GO TO 25
C
C------SET IFLAG=2 TO INDICATE INABILITY TO FACTORIZE MATRIX.
C
  999 IFLAG=2
      WRITE(6,9999)
      STOP
   25 IF(N.GT.1)GO TO 30
      X(1)=B(1)/A(1,1)
      RETURN
   30 IP=IPIVOT(1)
      X(1)=B(IP)
      DO 50 K=2,N
      IP=IPIVOT(K)
      KM1=K-1
      SUM=0.0
      DO 40 J=1,KM1
   40 SUM=A(IP,J)*X(J)+SUM
   50 X(K)=B(IP)-SUM
C
C
      X(N)=X(N)/A(IP,N)
      K=N
      DO 70 NP1MK=2,N
      KP1=K
      K=K-1
      IP=IPIVOT(K)
      SUM=0.0
      DO 60 J=KP1,N
   60 SUM=A(IP,J)*X(J)+SUM
   70 X(K)=(X(K)-SUM)/A(IP,K)
C------PLACE ANSWER VECTOR IN LAST COL OF A.
      DO 80 J=1,N
   80 A(J,NP1)=X(J)
      RETURN
 9999 FORMAT( 1H0,'****UNABLE TO COMPLETE L-U DECOMPOSITION OF MATRIX')
      END
      SUBROUTINE DERPS(X,VALS,RATES)
C------RETURNS DDELDX,DUBDX,DUTDX TO CALLING ROUTINE. THIS IS
C       STORED IN VECTOR RATES.
C       TBL COMPUTATION WITH PRESSURE SPECIFIED.
      REAL KAP,VALS(3),RATES(3)
      REAL A(3,3),B(3),W(3,3),D(3)
      INTEGER IPIVOT(3),IFLAG
      COMMON/DER1/DDELDX,DUBDX,DUTDX,DUIDX1
      COMMON/DER2/DELT,UB,UT,UI1,VT,VB,UDUI,TAUM,H,THETA,DELST,CFD2,
     $VISCOS,NBL
      COMMON/ODE1S/JSTRTS,JENDS,NDIM,SW(90),VI(90),DVI(90),DDVI(90),
     $DS(90)
      COMMON/ODE1U/JSTRTU,JENDU,SWU(90),VIU(90)
      COMMON/ODE2U/JTBLU,STBLU(90),DSTARU(90),UI1DU(90),DELTU(90),
     $UI2DU(90)
      COMMON/SPLYN/XX,UI,DUIDX,DDUI,ISETUP,KMID
      COMMON/TEMP1/XC,IWALLV
      COMMON/TEMP2/IEXIT,VEL(90)
      DATA KAP/.41/
C------SET UP COEFFICIENTS FOR THE A MATRIX, ANS SOLVE A*RATES=B
      XX=X
      IF(IWALLV.EQ.1)GO TO 100
      CALL SPLINE(SW,VEL,DVI,DDVI,DS,JSTRTS,JENDS,NDIM,4)
```

```
      GO TO 150
  100 CALL SPLINE(SWU,VIU,DVI,DDVI,DS,JSTRTU,JENDU,NDIM,4)
  150 CONTINUE
      DELT=VALS(1)
      UB=VALS(2)
      UT=VALS(3)
      CALL BLVALU
      DKU=DELT/(KAP*UI)
      CALL TAUMAX(TAUMEQ)
      TAUM=TAUMEQ
      A(1,1)=THETA/DELT
      A(1,2)=(DELT/UI)*(0.5-0.75*VB-1.58949*VT)
      A(1,3)=DKU*(1.0-4.0*VT-1.58949*VB)
      A(2,1)=UT**2/(KAP*DELT)
      A(2,2)=UT
      A(2,3)=UT/KAP+UI-UB
      A(3,1)=1.0-DELST/DELT
      A(3,2)=-0.5*DELT/UI
      A(3,3)=-DKU
      B(1)   =(KAP*VT)**2-2.0*DELST*DUIDX/UI+THETA/(XC-X)
      B(2)   =UT*DUIDX
      B(3)   =10.0*TAUM/UI**2-(DELT-DELST+DELT*(VT+0.5*VB))*DUIDX/UI
      CALL FACTOR(A,W,IPIVOT,D,3,IFLAG)
      CALL SUBST(W,B,RATES,IPIVOT,3)
      RETURN
      END
      SUBROUTINE FERSEG
C------SET UP THE A MATRIX OF COEFFICIENTS TO SOLVE LAPLACE'S EQN
C       IN 2-D USING PLEMELJ'S FORM OF THE CAUCHY INTEGRAL FORMULA.
C       LINEAR APPROX FOR THE FUNCTION BETWEEN NODE POINTS.
      COMPLEX C(91)
      REAL XO(120),YO(120)
      REAL*8 A(86,87),BB,DIMAG,DREAL
      COMPLEX ZERO,ICMPLX
      COMPLEX*16 CS(180),LAMDAO(180)
      COMPLEX*16 ZO,ERP,LERP,DMP1,DMM1,TEMP
      COMPLEX*16 CDLOG
      COMMON/EFCVAL/XC(90),YC(90),AL(90),ALNV(90)
      COMMON/GEOM2/N,NR,NL,NU,NM1,NLC,NRC
      COMMON/PFSL1/C,A,XO,YO
      DATA PI/3.141593/
      NUP1=NU+1
      NM2=N-2
      ZERO=(0.0,0.0)
      ICMPLX=(0.0,1.0)
C     +++ EXTEND C ARRAY ++++
      DO 30 J=1,N
      CS(J)=C(J)
   30 CS(J+N)=C(J)
C     ++++ EACH PASS CORRESPONDS TO ONE UNKN ZO BOUNDARY POINT ++++
      DO 500 M=1,NU
      ZO=CS(M)
      JSTART=M+1
      JEND=NM1+M
      MJEND=JEND-1
C     ++++ FORM GEOMETRY COEFFICENTS ++++
      LAMDAO(JSTART)=ZERO
      DO 50 J=JSTART,MJEND
      ERP=(CS(J+1)-ZO)/(CS(J)-ZO)
      LERP=CDLOG(ERP)/(ERP-1.0)
      LAMDAO(J)=LAMDAO(J)+ERP*LERP
```

```
 50 LAMDAO(J+1)=-LERP
    DMP1=CS(JSTART)-ZO
    DMM1=ZO-CS(JEND)
    TEMP=(DMP1-DMM1)/2.0
    LAMDAO(M)=CDLOG(DMP1/DMM1)-(ICMPLX*PI)
   &-(TEMP*(1.0/DMP1+1.0/DMM1))
    IF(M.EQ.NRC.OR.M.EQ.NLC) LAMDAO(M)=CDLOG((ICMPLX*DMP1)/(-DMM1))
   &-ICMPLX*(PI/2.0)-(TEMP*(1.0/DMP1+1.0/DMM1))
    LAMDAO(JSTART)=LAMDAO(JSTART)+(TEMP/DMP1)
    LAMDAO(JEND)=LAMDAO(JEND)+(TEMP/DMM1)
    IF(M.EQ.1) GO TO 70
    M1=M-1
    DO 60 J=1,M1
 60 LAMDAO(J)=LAMDAO(N+J)
 70 CONTINUE
C     ++++ FORM A MATRIX ++++
    BB=0.0
    DO 250 J=1,NU
    A(M,J)=DIMAG(LAMDAO(J))
250 BB=BB+AL(J)*DREAL(LAMDAO(J))
    A(M,NUP1)=BB
   &+AL(NM1)*DREAL(LAMDAO(NM1))
   &+AL(N)*DREAL(LAMDAO(N))
500 CONTINUE
C      ++++ A MATRIX FORMULATION COMPLETE ++++
    RETURN
    END
    SUBROUTINE EFGEOM
C------GIVEN THE WALL LOCATION COORDINATES,XW(I),YW(I),ALW(I),
C      AND DISPLACEMENTS DSTARS(I),DSTARU(I),LOCATE THE BOUNDARY
C      OF THE EFC. NOTE STBLS AND STBLU ARE DISPLACED ONE ELEMENT
C      AHEAD OF THE REST.
    REAL DSHIFT(90)
    REAL*8 A(86,87)
    COMPLEX C(91)
    REAL XO(120),YO(120)
    COMMON/EFCVAL/X(90),Y(90),AL(90),ALNV(90)
    COMMON/GEOM2/N,NR,NL,NU,NM1,NLC,NRC
    COMMON/ODE1S/JSTRTS,JENDS,NDIM,SW(90),WI(90),DWI(90),DDWI(90),
   $DS(90)
    COMMON/ODE1U/JSTRTU,JENDU,SWU(90),WIU(90)
    COMMON/ODE2S/JTBLS,STBLS(90),DSTARS(90),UI1DS(90),DELTS(90),
   $UI2DS(90)
    COMMON/ODE2U/JTBLU,STBLU(90),DSTARU(90),UI1DU(90),DEITU(90),
   $UI2DU(90)
    COMMON/PFSL1/C,A,XO,YO
    COMMON/SPLYN/XINT,FINT ,DDSTDX,FPPINT,ISETUP,KMID
    COMMON/TEMP1/XCMX,IWALLV
    COMMON/WALVAL/XW(90),YW(90),ALW(90)
    NRCP1=NRC+1
    NMNLC=N-NLC
C
    IF(IWALLV.EQ.1)GO TO 105
C------ON LOWER WALL, DSHIFT(J+1)=DSTARS(J), I.E. DISPLACED ONE
C      ELEMENT.
    DSHIFT(1)=DSTARS(N)
    DO 50 J=2,NRCP1
 50 DSHIFT(J)=DSTARS(J-1)
C------SET UP DSTARS AND ITS DERIVATIVE DDSTS ON LOWER BOUNDARY.
    XINT=SW(1)
    ISETUP=0
```

```
      KMID=2
      CALL SPLINE(SW,DSHIFT,DWI,DDWI,DS,1,NBCP1,NDIM,2)
      X(N)=XW(N)
      Y(N)=YW(N)+DSTARS(N)
      X0(N)=XW(N)
      Y0(N)=YW(N)+DELTS(N)
      AL(N)=ALW(N)+ATAN(DDSTDX)
      DO 100 J=1,NRC
      XINT=SW(J+1)
      CALL SPLINE(SW,DSHIFT,DWI,DDWI,DS,1,NBCP1,NDIM,2)
      SINALJ=SIN(ALW(J))
      COSALJ=COS(ALW(J))
      X(J)=XW(J)-DSTARS(J)*SINALJ
      Y(J)=YW(J)+DSTARS(J)*COSALJ
      X0(J)=XW(J)-DELTS(J)*SINALJ
      Y0(J)=YW(J)+DELTS(J)*COSALJ
  100 AL(J)=ALW(J)+ATAN(DDSTDX)
      GO TO 160
C
C------SET UP DSTARU AND ITS  DERIV DDSTS ON UPPER BOUNDARY.
C------FLIP INDICES TO MAKE DSTARU INCREASE IN SAME DIRECTION AS SWU.
  105 DO 110 J=NLC,NM1
  110 DSHIFT(N-J)=DSTARU(J)
      XINT=SWU(1)
      ISETUP=0
      KMID=2
      CALL SPLINE(SWU,DSHIFT,DWI,DDWI,DS,1,NMNLC,NDIM,2)
      DO 150 J=1,NMNLC
      XINT=SWU(J)
      CALL SPLINE(SWU,DSHIFT,DWI,DDWI,DS,1,NMNLC,NDIM,2)
      NMJ=N-J
      SINALJ=SIN(ALW(NMJ))
      COSALJ=COS(ALW(NMJ))
      X(NMJ)=XW(NMJ)+DSTARU(NMJ)*SINALJ
      Y(NMJ)=YW(NMJ)-DSTARU(NMJ)*COSALJ
      X0(NMJ)=XW(NMJ)+DELTU(NMJ)*SINALJ
      Y0(NMJ)=YW(NMJ)-DELTU(NMJ)*COSALJ
  150 AL(NMJ)=ALW(NMJ)-ATAN(DDSTDX)
  160 CONTINUE
C   160 WRITE(6,897)
C   897 FORMAT('1WALL COORDINATES AND EFFECTIVE FLOW CHANNEL LOCATION'//)
C       WRITE(6,898)
C   898 FORMAT('0',T4,'J',T10,'XW(J)',T25,'YW(J)',T40,'ALW(J)',T55,
C      $'X(J)',T70,'Y(J)',T85,'AL(J)',T100,'DELSTAR'/)
C       WRITE(6,900) (J,XW(J),YW(J),ALW(J),X(J),Y(J),AL(J),DSTARS(J),
C      $J=1,NRC)
C       WRITE(6,900) (J,XW(J),YW(J),ALW(J),X(J),Y(J),AL(J),DSTARU(J),
C      $J=NRCP1,NM1)
C       WRITE(6,900)N,XW(N),YW(N),ALW(N),X(N),Y(N),AL(N),DSTARS(N)
C   900 FORMAT(I5,1P7E15.5)
      RETURN
      END
      SUBROUTINE DERSI(X,VALS,RATES)
C------RETURNS DDELDX,DUBDX,DUTDX,DUIDX TO CALLING ROUTINE, VIA THE
C       VECTOR RATES.
C       TBL COMPUTATION WITH SIMULTANEOUS ITERATION BETWEEN THE
C       BOUNDARY LAYER AND ONE-DIMENSIONAL CORE.
      REAL KAP,VALS(4),RATES(4)
      REAL A(4,4),B(4),W(4,4),D(4)
      INTEGER IPIVOT(4),IFLAG
      COMMON/DER1/DDELDX,DUBDX,DUTDX,DUIDX
```

```
      COMMON/DER2/DELT,UB,UT,UI ,VT,VB,UDUI,TAUM,H,THETA,DELST,CPD2,
     $VISCOS,NBL
      COMMON/GEOM1/XD,W1,TH,WDTH,X1,B1,SINTH,COSTH,TWOTHR,TWOTH1,
     $SINTH1,COSTH1,AS,WH,XC,X2,XMAX,XDE
      COMMON/ODE1S/JSTRTS,JENDS,NDIM,SW(90),WI(90),DWI(90),DDWI(90),
     $DS(90)
      COMMON/ODE1U/JSTRTU,JENDU,SWU(90),WIU(90)
      COMMON/ODE2S/JTBLS,STBLS(90),DSTARS(90),UI1DS(90),DELTS(90),
     $UI2DS(90)
      COMMON/ODE2U/JTBLU,STBLU(90),DSTARU(90),UI1DU(90),DELTU(90),
     $UI2DU(90)
      COMMON/SPLYN/XX,WIDTH,DWDX,DDWDX,ISETUP,KMID
      COMMON/TEMP1/XCMX,IWALLV
      DATA KAP/0.41/
C------WIDTH=WIDTH OF 1-D CORE SECTION(CHANNEL SIZE MINUS BLOCKAGE)
C------SET COEFFICIENTS FOR THE A MATRIX, AND SOLVE A*RATES=B
      XX=X
      TWOTHR=0.0
      COSTH=1.0
      IF(NBL.EQ.1)GO TO 100
      IF(XX.LE.X1.OR.XX.GE.XDE)GO TO 100
      COSTH=COSTH1
      TWOTHR=TWOTH1
  100 CONTINUE
      DELT=VALS(1)
      UB=VALS(2)
      UT=VALS(3)
      UI=VALS(4)
      WIDTH=UI
      DWDX=DUIDX
      IF((UB-UI)*UT.LE.0.0)GO TO 110
      UT=-UT
  110 CALL BLVALU
      DKU=DELT/(KAP*UI)
      CALL TAUMAX(TAUMEQ)
      TAUM=TAUMEQ
C------IWALLV=0 IS THE LOWER WALL.IWALLV=1 IS UPPER WALL.
      IF(IWALLV.EQ.1)GO TO 130
      CALL SPLINE(SW,WI,DWI,DDWI,DS,JSTRTS,JENDS,NDIM,4)
      GO TO 140
  130 CALL SPLINE(SWU,WIU,DWI,DDWI,DS,JSTRTU,JENDU,NDIM,4)
  140 CONTINUE
      DDSTDX=DELST*DDELDX/DELT-DELT*(VT+.5*VB)*DUIDX/UI+DELT*DUTDX/
     $(KAP*UI)+DELT*DUBDX/(2.0*UI)
      IF(ABS(DDSTDX).GT.1.E-4)XCMX=(0.5*WH-DELST)/DDSTDX
      A(1,1)=THETA/DELT
      A(1,2)=(DELT/UI)*(.5-0.75*VB-1.58949*VT)
      A(1,3)=DKU*(1.0-4.0*VT-1.58949*VB)
      A(1,4)=2.0*DELST/UI
      A(2,1)=UT**2/(KAP*DELT)
      A(2,2)=UT
      A(2,3)=UT/KAP+UI-UB
      A(2,4)=-UT
      A(3,1)=1.0-DELST/DELT
      A(3,2)=-0.5*DELT/UI
      A(3,3)=-DKU
      A(3,4)=(DELT-DELST+DELT*(VT+0.5*VB))/UI
      A(4,1)=A(3,1)-1.0
      A(4,2)=A(3,2)
      A(4,3)=-DKU
      A(4,4)=(COSTH/(NBL*UI))*(WIDTH-NBL*DELST/COSTH)+(DELT/UI**2)*
```

```
     $       (UT/KAP+0.5*UB)
      B(1)    = (KAP*VT)**2+THETA/XCMX
      B(2)    =0.0
      B(3)    = 10.0*TAUM/UI**2
      B(4)    =-DWDX*COSTH/NBL
C
C
C------IF UT BECOMES SMALL, THEN THE MATRIX BECOMES ILL-CONDITIONED.
C      FREEZE DUTDX, AND REMOVE THE DSF EQN. SOLVE THE REDUCED SET.
      IF(ABS(UT).GT.0.025) GO TO 200
      A(1,3)=A(1,4)
      A(2,1)=A(3,1)
      A(2,2)=A(3,2)
      A(2,3)=A(3,4)
      A(3,1)=A(4,1)
      A(3,2)=A(4,2)
      A(3,3)=A(4,4)
      B(2)=B(3)
      B(3)=B(4)
      CALL FACTOR(A,W,IPIVOT,D,3,IFLAG)
      CALL SUBST(W,B,RATES,IPIVOT,3)
      RATES(4)=RATES(3)
      RATES(3)=DUTDX
      RETURN
C
C
  200 CALL FACTOR(A,W,IPIVOT,D,4,IFLAG)
      CALL SUBST(W,B,RATES,IPIVOT,4)
      RETURN
      END
      SUBROUTINE SPLINE(X,F,FP,FPP,DX,JSTART,JEND,NDIM,INTERP)
C
C     ++++ SPLINE IN TENSION FIT OF F(X)
C     FIRST DERIVATIVE AT POINT=FP
C     SECOND DERIVATIVE AT POINT=FPP
C      START AND END OF INTERVAL=JSTART,JEND
C      TENSION FACTOR=SIGMA
C      ROUTINE BY RINEHART
C
      REAL X(NDIM),F(NDIM),FP(NDIM),FPP(NDIM),DX(NDIM)
      REAL A(101),B(101),C(101),D(101)
      COMMON/SPLYN/XINT,FINT,FPINT,FPPINT,ISETUP,KMID
C
C-------SPLINE FIT OF F(X) USED FOR FINDING FIRST & 2ND DERIVATIVES AT
C       THE POINTS, & ALSO FOR INTERPOLATION.  ISETUP= COUNTER OF # OF
C       TIMES ROUTINE CALLED USING SAME FIT.  WHEN ISETUP=0 FPP TABLE
C       DETERMINED.  CUBIC RUNOUT END CONDITION.  KMID IS GUESS
C       INDEX OF INTERVAL WHERE X(KMID) < XINT < X(KMID+1).  WHEN
C       (INTERP=1, FIND FINT),(=2, FIND FPINT),(=3, FIND FPPINT),
C       (=4, FIND FINT & FPINT).  FOR INTERP>4 NO INTERPOLATION, FIND
C       ONLY DERIVATIVES AT POINTS.  FOR INTERP=0 SETUP ONLY.
      SIGMA=2.5
      SS=SIGMA*SIGMA
      SIGMA=SIGMA*(JEND-JSTART)/(X(JEND)-X(JSTART))
      SISO=1.0
      IF(ISETUP.NE.0)GO TO 180
      JENDM1=JEND-1
      DO 120 J=JSTART,JENDM1
  120 DX(J)=X(J+1)-X(J)
      DX(JEND)=DX(JEND-1)
      JSTP1=JSTART+1
```

```
      DO 140 J=JSTP1,JENDM1
      H1=DX(J)
      HM1=DX(J-1)
      SH1=SIGMA*H1
      SHM1=SIGMA*HM1
      A(J)=((1.0/HM1)-(SIGMA/SINH(SHM1)))*SISQ
      BP1=SIGMA*((1.0/TANH(SHM1))+(1.0/TANH(SH1)))
      B(J)=(BP1-(1.0/H1)-(1.0/HM1))*SISQ
      C(J)=((1.0/H1)-(SIGMA/SINH(SH1)))*SISQ
      D(J)=((F(J+1)-F(J))/H1)-((F(J)-F(J-1))/HM1)
  140 CONTINUE
      JOPPS = 0
      NDIAG=101
      A(JSTART)=0.0
      C(JEND)=0.0
C-------CUBIC RUNOUT END CONDITIONS
      J=JSTART
      HI=DX(J)
      SHI=SIGMA*DX(J)
      B(J)=((SIGMA*2.0/TANH(SHI))-(2.0/HI))*SISQ+B(J+1)
      C(J)=(1.0/HI-SIGMA/SINH(SHI))*SISQ+C(J+1)
      J=JEND
      HI=DX(J)
      SHI=SIGMA*DX(J)
      HMI=DX(J-1)
      A(J)=(1.0/HM1-SIGMA/SINH(HMI))*SISQ+A(J-1)
      B(J)=((SIGMA*2.0/TANH(SHI))-(2.0/HI))*SISQ+B(J-1)
      C(JEND)=0.0
      CALL TRIDAG(JSTP1,JENDM1,A,B,C,D,FPP,NDIAG,NDIM)
      FPP(JSTART)=0.0
      FPP(JEND)=0.0
      IF(INTERP.LE.0) GO TO 1000
  180 IF(INTERP.GT.4) GO TO 700
C--------FIND INTERVAL OF INTERPOLATION.  X(KMID) < XINT < X(KMID+1)
      IF(X(JSTART).GT.X(JEND)) GO TO 250
      IF(XINT.LE.X(JSTART).OR.XINT.GE.X(JEND)) GO TO 2000
C-------X IS A MONOTONICALLY INCREASING FUNCTION WITH THE INDEX.
  200 IF(XINT.GE.X(KMID)) GO TO 220
      KMID = KMID-1
      GO TO 200
  220 IF(XINT.LE.X(KMID+1)) GO TO 300
      KMID = KMID+1
      GO TO 220
C-------X IS A MONOTONICALLY DECREASING FUNCTION WITH THE INDEX.
  250 IF(XINT.GE.X(JSTART).OR.XINT.LE.X(JEND)) GO TO 2000
  260 IF(XINT.LE.X(KMID)) GO TO 270
      KMID = KMID-1
      GO TO 260
  270 IF(XINT.GE.X(KMID+1)) GO TO 300
      KMID = KMID+1
      GO TO 270
C-------PERFORM INTERPOLATION.
  300 DELX = XINT-X(KMID)
      KMIDP1 = KMID+1
      DELXP = X(KMIDP1)-XINT
      DXKMID = DX(KMID)
      GO TO (400,500,600,400), INTERP
C-------INTERPOLATE FOR F(XINT) = FINT
  400 FINT=FPP(KMID)*SISQ*SINH(SIGMA*DELXP)/SINH(SIGMA*DXKMID)
     &+(F(KMID)-FPP(KMID)*SISQ)*(DELXP/DXKMID)
     &+(FPP(KMIDP1)*SISQ)*(SINH(SIGMA*DELX))/SINH(SIGMA*DXKMID)
```

```
     &+(F(KMIDP1)-FPP(KMIDP1)*SISQ)*DELX/DXKMID
      GO TO (1000,500,600,500), INTERP
C-------INTERPOLATE FOR FP(XINT) = FPINT
  500 FPINT=-SIGMA*FPP(KMID)*COSH(SIGMA*DELXP)/SINH(SIGMA*DXKMID)
     $-(F(KMID)-FPP(KMID))/DXKMID
     $+SIGMA*FPP(KMIDP1)*COSH(SIGMA*DELX)/SINH(SIGMA*DXKMID)
     $+(F(KMIDP1)-FPP(KMIDP1))/DXKMID
      GO TO 1000
C-------INTERPOLATE FOR FPP(XINT) = FPPINT
  600 FPPINT=SS*(FPP(KMID)*SINH(SIGMA*DELXP)/SINH(SIGMA*DXKMID)
     $+FPP(KMIDP1)*SINH(SIGMA*DELX)/SINH(SIGMA*DXKMID))
      GO TO 1000
  700 CONTINUE
C     ++++ INTERPOLATION FOR FP AT POINTS, AVERAGE OF FORWARD
C        AND BACKWARD FORMULAS ++++
      DO 750 J=JSTP1,JENDM1
C     ++++ BACKWARDS DIFFERENCE ++++
      FPB=FPP(J-1)*(-SIGMA/SINH(SIGMA*DX(J-1)))
     &-(F(J-1)-FPP(J-1))/DX(J-1)
     &+FPP(J)*SIGMA*COSH(SIGMA*DX(J-1))/SINH(SIGMA*DX(J-1))
     &+(F(J)-FPP(J))/DX(J-1)
C     ++++ FORWARD DIFFERENCE ++++
      FPF=FPP(J)*((-SIGMA)*COSH(SIGMA*DX(J))/SINH(SIGMA*DX(J)))
     &-(F(J)-FPP(J))/DX(J)
     &+FPP(J+1)*SIGMA/(SINH(SIGMA*DX(J)))
     &+(F(J+1)-FPP(J+1))/DX(J)
C     ++++ AVERAGE FORWARD AND BACKWARDS DIFFERENCES ++++
      FP(J)=0.5*(FPF+FPB)
  750 CONTINUE
C------USE FORWARD DIFF FOR START SEGMENT, AND BACKWARD DIFF
C       FOR THE LAST SEGMENT.
      J=JSTART
      FP(J)=FPP(J)*((-SIGMA)*COSH(SIGMA*DX(J))/SINH(SIGMA*DX(J)))
     &-(F(J)-FPP(J))/DX(J)
     &+FPP(J+1)*SIGMA/(SINH(SIGMA*DX(J)))
     &+(F(J+1)-FPP(J+1))/DX(J)
      J=JEND
      FP(J)=FPP(J-1)*(-SIGMA/SINH(SIGMA*DX(J-1)))
     &-(F(J-1)-FPP(J-1))/DX(J-1)
     &+FPP(J)*SIGMA*COSH(SIGMA*DX(J-1))/SINH(SIGMA*DX(J-1))
     &+(F(J)-FPP(J))/DX(J-1)
      IF(INTERP.LT.5)GO TO 1000
      DO 800 J=JSTART,JEND
  800 FPP(J)=FPP(J)*SS
 1000 ISETUP = ISETUP+1
      RETURN
 2000 JOPPS = JEND
      IF(ABS(XINT-X(JSTART)).LT.ABS(XINT-X(JEND))) JOPPS = JSTART
C------USE FORWARD DIFF FOR START SEGMENT, AND BACKWARD DIFF
C       FOR THE LAST SEGMENT.
      J=JSTART
      FP(J)=FPP(J)*((-SIGMA)*COSH(SIGMA*DX(J))/SINH(SIGMA*DX(J)))
     &-(F(J)-FPP(J))/DX(J)
     &+FPP(J+1)*SIGMA/(SINH(SIGMA*DX(J)))
     &+(F(J+1)-FPP(J+1))/DX(J)
      J=JEND
      FP(J)=FPP(J-1)*(-SIGMA/SINH(SIGMA*DX(J-1)))
     &-(F(J-1)-FPP(J-1))/DX(J-1)
     &+FPP(J)*SIGMA*COSH(SIGMA*DX(J-1))/SINH(SIGMA*DX(J-1))
     &+(F(J)-FPP(J))/DX(J-1)
      FINT = F(JOPPS)
```

```
      FPINT = FP(JOPPS)
      FPPINT = FPP(JOPPS)
      RETURN
      END
      FUNCTION YINT(X,Y,XINT)
C------GIVEN 3 COORDINATES (X,Y), FIT SECOND ORDER LAGRANGE
C       POLYNOMIAL AND RETURN THE VALUE YINT, CORRESPONDING TO XINT.
      REAL X(3),Y(3),XINT
      D1=X(2)-X(1)
      D2=X(3)-X(2)
      D3=X(1)-X(3)
      YINT=-Y(1)*(XINT-X(2))*(XINT-X(3))/(D1*D3)
     $      -Y(2)*(XINT-X(1))*(XINT-X(3))/(D1*D2)
     $      -Y(3)*(XINT-X(1))*(XINT-X(2))/(D3*D2)
      RETURN
      END
      SUBROUTINE STAND
C       GENERATE NODE POINTS FOR STANDARD DIFFUSERS. STRAIGHT WALLED
C       UNITS WITH BOTH WALLS DIVERGING(GEOMT='STDD') OR ASSYMMETRIC
C       UNITS WITH ONE DIVERGING WALL(GEOMT='HALF'). FOR NOMENCLATURE
C       SEE USERS GUIDE.
C
C
      REAL N,L
      COMMON/BLIV/HS,DELSTS,HU,DELSTU
      COMMON/DER1/DDELDX,DUBDX,DUTDX,DUIDX
      COMMON/DER2/DELT,UB,UT,UI ,VT,VB,UDUI,TAUM,H,THETA,DELST,CPD2,
     $VISCOS,NBL
      COMMON/EFCVAL/X(90),Y(90),AL(90),ALNV(90)
      COMMON/GEOM1/N,W1,TH,WIDTH,X1,B1,SINTH,COSTH,TWOTHR,TWOTH1,
     $SINTH1,COSTH1,AS,WH,XC1,X2,XMAX,XDE
      COMMON/GEOM2/NS,NR,NL,NU,NSM1,NLC,NRC
      COMMON/ODE1S/JSTRTS,JENDS,NDIM,SW(90),WI(90),DWI(90),DDWI(90),
     $DS(90)
      COMMON/ODE1U/JSTRTU,JENDU,SWU(90),WIU(90)
      COMMON/INITAL/DELST1,H1,UI1,IPR1
      COMMON/NSTD/ID1,ID2,ID3,NST,SWT(90)
      COMMON/PRINT/IPR,NORMPR,CPEROR,ITMAX
      COMMON/SPLYN/XX,WT,DWT,DDWT,ISETUP,KMID
      COMMON/TEMP1/XC,IWALLV
      COMMON/WALVAL/XW(90),YW(90),ALW(90)
      DATA PI/3.141593/
      READ(5,902)X1,RC1,N,RC2,X2,W1,TWOTHD,AS
C------IF BOTH THE INLET AND DIFFUSING SECTIONS ARE OF ZERO LENGTH, QUIT.
      IF(N.EQ.0.0.AND.X1.EQ.0.0)RETURN
      WRITE(6,903)
      WRITE(6,904)X1,RC1,N,RC2,X2
      WRITE(6,910)
      WRITE(6,915)W1,TWOTHD,AS
C
C------STORE STARTING VALUES.
      IF(AS.LE.0.0)AS=8.0
      READ(5,901)N1,NC1,N2,NC2,N3,ND1,ND2
      WRITE(6,934)
      WRITE(6,935)N1,NC1,N2,NC2,N3
      READ(5,900) B1,UI,VISCOS,XC
      IF(XC.EQ.0.0)XC=1.E5
      WRITE(6,920) B1,UI,VISCOS,XC
C------N,W1,DELST(FT), TWOTH(DEGREES), B1(N-D), UI(FT/SEC), VISCOS(FT2/SEC)
      READ(5,905)HS,DELSTS,HU,DELSTU
      READ(5,906)IPR,NORMPR,ITMAX,CPEROR
```

```
      H=HS
      H1=HS
      DELST=DELSTS
      DELST1=DELSTS
      THETA=DELST1/H1
      IF(HU.NE.0.0)GO TO 60
      HU=H1
      DELSTU=DELST1
   60 CONTINUE
      WRITE(6,907)HS,DELSTS,HU,DELSTU
      WRITE(6,908)IPR,NORMPR,ITMAX,CPEROR
      NRC=N1+NC1+N2+NC2+N3
      NLC=NRC+1
      NS=2+2*NRC
      NRCP1=NRC+1
      NSM1=NS-1
      NSM2=NS-2
      NEND=0
      TWOTHR=TWOTHD*PI/180.0
      THR=TWOTHR/2.0
      TAND2=TAN(THR/2.0)
      TANTH1=TAN(THR)
      SINTH1=SIN(THR)
      COSTH1=COS(THR)
      TWOTH1=TWOTHR
      SINTH=SINTH1
      COSTH=COSTH1
      RC1MT=RC1*TAND2
      RC2MT=RC2*TAND2
      C1=X1-RC1MT
      C2=X1+RC1MT*COSTH1
      C3=X1+N-RC2MT*COSTH1
      C4=X1+N+RC2MT
      UI1=UI
      H1=H
      DELST1=DELST
      IPR1=IPR
C
C------W1,W2 ARE INLET,EXIT WIDTHS(FT),L IS SLANT LENGTH ALONG WALL.
      L=N/COSTH1
      XDE=X1+L
      XMAX=XDE+X2
      W2=W1+2.0*L*SINTH1
C------STARTING VALUE AT NODE ZERO(=NS).
      X(NS)=0.0
      Y(NS)=0.0
      AL(NS)=0.0
      X(NSM1)=0.0
      Y(NSM1)=W1
      WI(1)=W1
      SW(1)=0.0
      SWU(1)=0.0
      AL(NSM1)=0.0
C------COORDINATES FOR INLET SECTION, N1 SEGMENTS.
      XO=X(NS)
      IF(N1.EQ.0)GO TO 120
C------ARITHMETIC PROGRESSION FROM INLET TO THROAT.
      IF(ND1.EQ.0)ND1=5*N1
      FL=X1-RC1MT
      A=FL/N1
      D=0.0
```

```
      IF(N1.EQ.1)GO TO 100
      A=EL/ND1
      D=2.0*(EL-A*N1)/(N1*(N1-1))
      NSTART=0
      NEND=NSTART+N1
  100 DO 110 J=1,N1
      DX=A+(N1-J)*D
      X(J)=XO+DX
      Y(J)=0.0
      AL(J)=0.0
      SW(J+1)=X(J)
  110 XO=X(J)
C
C------THROAT CURVE, NC1 SEGMENTS.
  120 IF(NC1.EQ.0)GO TO 140
      XCL=RC1MT*(1.0+COSTH1)
      DX=XCL/NC1
      NSTART=N1+1
      NEND=NSTART+NC1-1
      DO 130 J=NSTART,NEND
      X(J)=XO+DX
      Y(J)=-RC1+SQRT(RC1**2-(X(J)-(X1-RC1MT))**2)
      AL(J)=-ARSIN((X(J)-C1)/RC1)
      SW(J+1)=SW(NSTART)+RC1*ABS(AL(J))
  130 XO=X(J)
C
C------DIFFUSING SECTION, N2 SEGMENTS.
  140 IF(N2.EQ.0)GO TO 160
C------ARITHMETIC PROGRESSION FROM THROAT TO TAILPIPE.
      IF(ND2.EQ.0)ND2=5*N2
      EL=N-(RC1MT+RC2MT)*COSTH1
      A=EL/ND2
      D=2.0*(EL-A*N2)/(N2*(N2-1))
      NSTART=NEND+1
      NEND=NSTART+N2-1
      DO 150 J=NSTART,NEND
      DX=A+(J-NSTART)*D
      X(J)=XO+DX
      Y(J)=(X1-X(J))*TANTH1
      AL(J)=-THR
      SW(J+1)=SW(NSTART)+(X(J)-C2)/COSTH1
  150 XO=X(J)
C
C------TAILPIPE INLET CURVE, NC2 SEGMENTS.
  160 IF(NC2.EQ.0)GO TO 180
      YTEMP=Y(NEND)
      XCL=RC2MT*(1.0+COSTH1)
      DX=XCL/NC2
      NSTART=NEND+1
      NEND=NSTART+NC2-1
      DO 170 J=NSTART,NEND
      X(J)=XO+DX
      Y(J)=RC2-N*TANTH1-SQRT(RC2**2-(X(J)-(X1+N+RC2MT))**2)
      DZ=SQRT((X(J)-C3)**2+(Y(J)-YTEMP)**2)
      BETA=2.0*ARSIN(DZ/(2.0*RC2))
      SW(J+1)=SW(NSTART)+RC2*BETA
      AL(J)=BETA-THR
  170 XO=X(J)
C
C------TAILPIPE SECTION, N3 SEGMENTS.
  180 IF(N3.EQ.0)GO TO 200
```

```
      XCL=X2-RC2MT
      DX=XCL/N3
      NSTART=NEND+1
      NEND=NSTART+N3-1
      DO 190 J=NSTART,NEND
      X(J)=XO+DX
      Y(J)=-N*TANTH1
      AL(J)=0.0
      SW(J+1)=SW(NSTART)+X(J)-C4
  190 XO=X(J)
C
C------MAP UPPER BOUNDARY FROM LOWER WALL.
  200 DO 250 J=1,NRC
      NSM1MJ=NSM1-J
      X(NSM1MJ)=X(J)
      Y(NSM1MJ)=W1-Y(J)
      AL(NSM1MJ)=-AL(J)
      SWU(J+1)=SW(J+1)
  250 WI(J+1)=W1-2.0*Y(J)
      JSTRTS=1
      JSTRTU=1
      JENDS=NRCP1
      JENDU=NRCP1
      JENDP1=JENDS+1
      WRITE(6,1212) (J,X(J),Y(J),AL(J),WI(J),SW(J),J=1,JENDS)
 1212 FORMAT(' ',I5,1P5E15.5)
      DO 300 J=1,NS
      XW(J)=X(J)
      YW(J)=Y(J)
  300 ALW(J)=AL(J)
      WRITE(6,1313) (J,X(J),Y(J),AL(J),       J=JENDP1,NS)
 1313 FORMAT(' ',I5,1P3E15.5)
C
C------IF ID1=4 THEN SET UP STANDARD DIFFUSER WITH ONLY 1 DIVERGING
C       WALL, AT AN ANGLE -THETA=-(TWOTHD/2). NOTE TWOTHD ENTERED MUST
C       BE DOUBLE THIS VALUE. MODIFIES OUTPUT FROM STAND BY CHOPPING
C       OFF TOP WALL.
      IF(ID1.NE.4)RETURN
      NSTART=N1+2
      SST=SW(N1+1)
      DO 350 J=NSTART,NRCP1
      SWU(J)=X(J-1)
      WI(J)=WI(J)-(WI(J)-W1)/2.0
      NMJ=NS-J
      XW(NMJ)=XW(J-1)
      YW(NMJ)=W1
  350 ALW(NMJ)=0.0
      DO 400 J=1,NS
      X(J)=XW(J)
      Y(J)=YW(J)
  400 AL(J)=ALW(J)
      THD=TWOTHD/2.0
      WRITE(6,940)THD
      WRITE(6,1515) (J,X(J),Y(J),AL(J),SW(J),J=1,JENDS)
 1515 FORMAT(' ',I5,1P4E15.5)
      WRITE(6,1414) (J,X(J),Y(J),AL(J),J=JENDP1,NS)
 1414 FORMAT(' ',I5,1P3E15.5)
      RETURN
  940 FORMAT('1***STANDARD DIFFUSER WITH 1 DIVERGING WALL AT AN ANGLE'
     $,F7.3,'(DEGREES)***'//' WALL COORDINATES-NODE#',T30,'X-COORD',
     $T45,'Y-COORD',T60,'ALPHA(RAD)',T75,'WIDTH'/)
```

```
  900 FORMAT(4F10.5)
  901 FORMAT(7I10)
  902 FORMAT(8F10.5)
  903 FORMAT('-DIFFUSER GEOMETRY-INLET (X1-FT), THROAT RAD(RC1-FT), DIFF
     $USING LENGTH(N-FT), EXIT RADIUS(RC2-FT), TAILPIPE(X2-FT)')
  904 FORMAT(' ',T18,F10.5,T34,F10.5,T54,F10.5,T78,F10.5,T99,F10.5//)
  905 FORMAT(4E10.0)
  906 FORMAT(3I10,F10.0)
  907 FORMAT('0INLET BL VALUES:LOWER WALL-H=',F5.2,', DELSTS=',E12.5,
     $'(FT)'/T17,'UPPER WALL-H=',F5.2,', DELSTU=',E12.5,'(FT)'//)
  908 FORMAT('-BL PRINT INTERVAL(IPP)=',I2,',  NORMPR=',
     $I2,', MAX # ITERATIONS=',I2,', MAX ALLOWABLE CP ERROR=',
     $1PE12.5//)
  910 FORMAT('-',T17,'WIDTH(W1-FT), TWOTHD(DEGREES), ASPECT-RATIO')
  915 FORMAT(' ',T18,F10.5,T34,F10.5,T54,F10.5//)
  920 FORMAT(1H0,' B1,UI,VISCOS,XC=',2F12.5,F12.7,E12.5/)
  930 FORMAT(' SEGMENT DISTRIBUTION - INLET,THROAT CURVE, DIFFUSING SECT
     $TION, EXIT CURVE, TAILPIPE')
  935 FORMAT(' ',T24,I2,T31,I2,T45,I2,T64,I2,T76,I2//)
      END
      SUBROUTINE DERSIL(X,VALS,RATES)
C------RETURNS DDELDX,DUBDX,DUTDX,DUIDX TO CALLING ROUTINE, VIA THE
C       VECTOR RATES.
C       TBL COMPUTATION WITH SIMULTANEOUS ITERATION BETWEEN THE
C       BOUNDARY LAYER AND CORE, ASSUMING LINEAR VELOCITY VARIATION
C       BETWEEN THE UPPER AND LOWER DELTASTAR LINES.
      REAL KAP,VALS(4),RATES(4)
      REAL A(4,4),B(4),W(4,4),D(4)
      INTEGER IPIVOT(4),IFLAG
      COMMON/DER1/DDELDX,DUBDX,DUTDX,DUIDX
      COMMON/DER2/DELT,UB,UT,UI ,VT,VB,UDUI,TAUM,H,THETA,DELST,CFD2,
     $VISCOS,NBL
      COMMON/GEOM1/XD,W1,TH,WDTH,X1,B1,SINTH,COSTH,TWOTHR,TWOTH1,
     $SINTH1,COSTH1,AS,WH,XC1,X2,XMAX,XDE
      COMMON/LINEAR/WDIF(90),DU2D(90),DDU2D(90),WMD,DWEDX,UEFF,DUEDX
      COMMON/ODE1S/JSTRTS,JENDS,NDIM,SW(90),WI(90),DWI(90),DDWI(90),
     $DS(90)
      COMMON/ODE1U/JSTRTU,JENDU,SWU(90),WIU(90)
      COMMON/ODE2S/JTBLS,STBLS(90),DSTARS(90),UI1DS(90),DELTS(90),
     $UI2DS(90)
      COMMON/ODE2U/JTBLU,STBLU(90),DSTARU(90),UI1DU(90),DELTU(90),
     $UI2DU(90)
      COMMON/SPLYN/XX,Y,DYDX,DDYDX,ISETUP,KMID
      COMMON/TEMP1/XC,IWALLV
      DATA KAP/0.41/
C------WI CONTAINS (WDIF-DSTARU). WIU CONTAINS UI2DU(WITH PROPER INDICES).
C------SET COEFFICIENTS FOR THE A MATRIX, AND SOLVE A*RATES=B
      XX=X
      DELT=VALS(1)
      UB=VALS(2)
      UT=VALS(3)
      UI=VALS(4)
      Y=UI
      DYDX=RATES(4)
      IF((UB-UT)*UT.LE.0.0)GO TO 110
      UT=-UT
  110 CALL BLVALU
      DKU=DELT/(KAP*UI)
      CALL TAUMAX(TAUMEQ)
      TAUM=TAUMEQ
C------IWALLV=0 IS THE LOWER WALL.IWALLV=1 IS UPPER WALL.
```

```
      CALL SPLINE(SW,WI,DWI,DDWI,DS,JSTRTS,JENDS,NDIM,4)
      WMD=Y
      DWEDX=DYDX
      WEFF=WMD-DELST
      CALL SPLINE(SW,WIU,DU2D,DDU2D,DS,JSTRTS,JENDS,NDIM,4)
      UEFF=0.5*(Y+UI)
      DUEDX=DYDX
C
      A(1,1)=THETA/DELT
      A(1,2)=(DELT/UI)*(.5-0.75*VB-1.58949*VT)
      A(1,3)=DKU*(1.0-4.0*VT-1.58949*VB)
      A(1,4)=2.0*DELST/UI
      A(2,1)=UT**2/(KAP*DELT)
      A(2,2)=UT
      A(2,3)=UT/KAP+UI-UB
      A(2,4)=-UT
      A(3,1)=1.0-DELST/DELT
      A(3,2)=-0.5*DELT/UI
      A(3,3)=-DKU
      A(3,4)=(DELT-DELST+DELT*(VT+0.5*VB))/UI
      A(4,1)=A(3,1)-1.0
      A(4,2)=A(3,2)
      A(4,3)=-DKU
      A(4,4)=DELST/UI+0.5*WEFF/UEFF
      CORR3D=THETA/(XC-X)
      IF(UT.LE.0.0) CORR3D=0.0
      B(1)   =(KAP*VT)**2+CORR3D
      B(2)   =0.0
      B(3)   = 10.0*TAUM/UI**2
      B(4)=-DWEDX-0.5*WEFF*DUEDX/UEFF
C
C
C------IF UT BECOMES SMALL, THEN THE MATRIX BECOMES ILL-CONDITIONED.
C      FREEZE DUTDX, AND REMOVE THE DSF EQN. SOLVE THE REDUCED SET.
      IF(ABS(UT).GT.0.025)GO TO 200
      A(1,3)=A(1,4)
      A(2,1)=A(3,1)
      A(2,2)=A(3,2)
      A(2,3)=A(3,4)
      A(3,1)=A(4,1)
      A(3,2)=A(4,2)
      A(3,3)=A(4,4)
      B(2)=B(3)
      B(3)=B(4)
      CALL FACTOR(A,W,IPIVOT,D,3,IFLAG)
      CALL SUBST(W,B,RATES,IPIVOT,3)
      RATES(4)=RATES(3)
      RATES(3)=DUTDX
      RETURN
C
C
  200 CALL FACTOR(A,W,IPIVOT,D,4,IFLAG)
      CALL SUBST(W,B,RATES,IPIVOT,4)
      RETURN
      END
      SUBROUTINE DIFF2D
C------CALCULATION OF DIFFUSERS WITH 2-D CORE.
C      NST=0 IS STANDARD DIFFUSER,NST=1 IS NONSTANDARD. SW(J),J=1,NRCP1
C      IS LOWER WALL VALUES FOR SW. SWU(J),J=1,NMNLC IS UPPER WALL VALUES
C      FOR SWU. SWT(J) CONTAINS VALUES FOR LOWER WALL, AND J=1,NRCP1 OR
C      1,NMNLC ,WHICHEVER IS LARGER.
```

```
      REAL*8 A(86,87)
      COMPLEX C(91)
      REAL XO(120),YO(120),SWTEMP(90),LNV(90)
      COMMON/BLIV/HS,DELSTS,HU,DELSTU
      COMMON/DER2/DELT,UB,UT,UI,VT,VB,UDUI,TAUM,H,THETA,DELST,CFD2,
     $VISCOS,NBL
      COMMON/EFCVAL/XC(90),YC(90),AL(90),LNV
      COMMON/GEOM1/XD,W1,TH,WIDTH,X1,B1,SINTH,COSTH,TWOTHR,TWOTH1,
     $SINTH1,COSTH1,AS,WH,XC1,X2,XMAX,XDE
      COMMON/GEOM2/N,NR,NL,NU,NM1,NLC,NRC
      COMMON/INITAL/DELST1,H1,UI1,IPR1
      COMMON/NSTD/ID1,ID2,ID3,NST,SWT(90)
      COMMON/ODE1S/JSTRTS,JENDS,NDIM,SW(90),WI(90),DWI(90),DDWI(90),
     $DS(90)
      COMMON/ODE1U/JSTRTU,JENDU,SWU(90),WIU(90)
      COMMON/ODE2S/JTBLS,STBLS(90),DSTARS(90),UI1DS(90),DELTS(90),
     $UI2DS(90)
      COMMON/ODE2U/JTBLU,STBLU(90),DSTARU(90),UI1DU(90),DELTU(90),
     $UI2DU(90)
      COMMON/LINEAR/WDIF(90),DU2D(90),DDU2D(90),WMD,DWEDX,UEFF,DUEDX
      COMMON/PFSL1/C,A,XO,YO
      COMMON/PRINT/IPR,NORMPR,CPEROR,ITMAX
      COMMON/SIAD/IT,ER,CPERR,OMEGA
      COMMON/SPLYN/XX,WT,DWT,DDWT,ISETUP,KMID
      COMMON/TEMP1/XC2,IWALLV
      COMMON/TEMP2/IEXIT,VEL(90)
      COMMON/WALVAL/XW(90),YW(90),ALW(90)
      NLCP1=NLC+1
      NRCP1=NRC+1
      NMNLC=N-NLC
C------SIMULTANEOUS ITERATION ON THE ENTIRE CHANNEL.
      XX=0.0
      IF(CPEROR.LE.0)CPEROR=.02
      IF(ITMAX.LE.0)ITMAX=1
      IT=0
C------STORE WIDTH OF ENTIRE DIFFUSER IN WDIF.
      DO 5 J=1,NRCP1
    5 WDIF(J)=WI(J)
C
C------CALCULATE POTENTIAL FLOW IN DIFFUSER WITH BARE WALLS.
      DO 10 J=1,N
      XC(J)=XW(J)
      YC(J)=YW(J)
   10 AL(J)=ALW(J)
      CALL PFSL
      DO 50 J=NLC,NM1
      DSTARU(J)=DELSTU+0.004*SWU(N-J)
      UI2DU(J)=UI1*EXP(LNV(J))
      UI1DU(J)=UI2DU(J)
      WIU(N-J)=UI2DU(J)
   50 DELTU(J)=0.0
C------SIMULTANEOUS ITERATION ON ENTIRE CHANNEL.
      IWALLV=0
      NBL=2
      CALL TBLSIL(0)
      CALL CONVRT(0)
      WRITE(6,1212)
      WRITE(6,1313)
      WRITE(6,1111) (K,SW(K),WI(K),K=1,NRCP1)
C
C------SET UP THE BOUNDARIES OF THE EFC AND SOLVE 2-D POTENTIAL FLOW.
```

```
C************MAIN LOOP BEGINS HERE********
  100 IT=IT+1
      CALL EFGEOM
      CALL PFSL
C------THE VECTORS XO YO ALONG WHICH THE UI2DS AND UI2DU ARE
C        TO BE FOUND ARE SET UP BELOW.
      DO 150 J=1,N
      VMAG=EXP(LNV(J))
  150 XO(J)=VMAG*UI1
      DO 250 J=1,NRC
  250 UI2DS(J)=XO(J)
      DO 260 J=NLC,NM1
  260 UI2DU(J)=XO(J)
      UI2DS(N)=XO(N)
      WRITE(6,2222)
      WRITE(6,2223)
C------COMPARE THE 1-D AND 2-D VELOCITIES AND CP'S ALONG THE Y=DELSTAR LINE.
      CPERR=0.0
      ER=0.0
      DO 280 J=1,NRC
      ERR=UI1DS(J)-UI2DS(J)
      CP1D=1.0-(UI1DS(J)/UI1)**2
      CP2D=1.0-(UI2DS(J)/UI1)**2
      DCP=CP1D-CP2D
      CPERR=AMAX1(CPERR,ABS(DCP))
      WRITE(6,3333) J,UI1DS(J),UI2DS(J),ERR,CP1D,CP2D,DCP
  280 ER=AMAX1(ER,ABS(ERR))
      DO 285 J=NLC,NM1
      ERR=UI1DU(J)-UI2DU(J)
      CP1D=1.0-(UI1DU(J)/UI1)**2
      CP2D=1.0-(UI2DU(J)/UI1)**2
      DCP=CP1D-CP2D
      CPERR=AMAX1(CPERR,ABS(DCP))
      WRITE(6,3333) J,UI1DU(J),UI2DU(J),ERR,CP1D,CP2D,DCP
  285 ER=AMAX1(ER,ABS(ERR))
      ERR=UI1DS(N)-UI2DS(N)
      CP1D=1.0-(UI1DS(N)/UI1)**2
      CP2D=1.0-(UI2DS(N)/UI1)**2
      DCP=CP1D-CP2D
      CPERR=AMAX1(CPERR,ABS(DCP))
      WRITE(6,3333) N,UI1DS(N),UI2DS(N),ERR,CP1D,CP2D,DCP
      ER=AMAX1(ER,ABS(ERR))
      WRITE(6,950) ER,CPERR
C
C
      IF(CPERR.LE.CPEROR) RETURN
      WRITE(6,1414) IT
      IF(IT.LE.ITMAX) GO TO 290
      WRITE(6,940) IT
      RETURN
C
C------STORE UI2DU IN WIU AFTER FLIPPING INDICES.
  290 DO 300 J=1,NMNLC
  300 WIU(J)=UI2DU(N-J)
      UI=UI1
      IPR=IPR1
      NBL=1
      XX=0.0
      IF(MOD(IT,2).EQ.0)GO TO 450
C
C------LOWER WALL B.L. CALCULATION.
```

```
      WRITE(6,970)
      WRITE(6,975)
      DO 320 J=2,NRCP1
  320 WI(J)=WDIF(J)-DSTARU(N-J)
      WI(1)=WDIF(1)-DSTARU(NM1)
      WRITE(6,5555) (J,SW(J),WI(J),WIU(J),J=1,NRCP1)
      DELST=DELSTS
      H=HS
C------CALCULATE BL WITH PRESCRIBED PRESSURE GRADIENT.IF RETURNED
C       VALUE OF IEXIT=0, THEN BL HAS REACHED POINT OF INTERMITTENT
C       SEPARATION(H>=HSEP), AND THE REST HAS TO BE CALCULATED WITH
C       STREAMTUBE ITER.
      DO 420 J=1,NRC
  420 VEL(J+1)=UI2DS(J)
      VEL(1)=UI2DS(N)
      IWALLV=0
      CALL TBLPS
      IF(IEXIT.NE.0)GO TO 430
      WRITE(6,965)
      CALL TBLSIL(0)
  430 CALL CONVRT(0)
      GO TO 100
C
C
C------UPPER WALL . USE BL CALCULATION WITH SPECIFIED
C       PRESSURE GRADIENT(FROM UI2DU OBTAINED IN LAST ITERATION).
  450 IWALLV=1
      WRITE(6,960)
      WRITE(6,975)
      DELST=DELSTU
      H=HU
      JENDU=NMNLC
      CALL TBLPS
      CALL CONVRT(1)
      GO TO 100
  940 FORMAT(' ***UNABLE TO CONVERGE IN',I2,' ITERATIONS'//)
  950 FORMAT('0LARGEST ABSOLUTE ERRORS, VELOCITY=',1PE12.5,'(FT/SEC)',
     $5X,'CPERR=',1PE12.5//)
  960 FORMAT(' ******UPPER WALL VALUES******'//)
  965 FORMAT(' CONTINUE B.L. CALCULATION WITH LINEAR V.P. METHOD')
  970 FORMAT(' ******LOWER WALL VALUES******'//)
  975 FORMAT(' BOUNDARY LAYER CALCULATION-PRESCRIBED PRESSURE GRADIENT')
 1111 FORMAT(' ',I5,1P2E15.5)
 1212 FORMAT('1DIFFUSER WIDTH FOR THE FIRST ITERATION')
 1313 FORMAT('-',T4,'K',T10,'SW(K)',T25,'WI(K)'/)
 1414 FORMAT('1******ITERATION NUMBER ',I4,'  ******'//)
 2222 FORMAT('1VELOCITY COMPARISON'//)
 2223 FORMAT('ONODE#',T12,'1-D VEL',T27,'2-D VEL',T42,'(1D-2D VEL)',
     $T57,'CP 1-D',T72,'CP 2-D',T87,'(1D-2D CP)'/)
 3333 FORMAT(' ',I5,1P6E15.5)
 5555 FORMAT(I5,1P3E15.5)
      END
      SUBROUTINE NSTAND
C------READS IN AND PROCESSES GEOMETRY FOR A NON-STANDARD DUCT.
C       ALSO SUPPLIES AN ESTIMATE FOR DUCT WIDTH TO BE USED FOR
C       SIMULTANEOUS ITERATION IN THE FIRST LOOP.
      COMMON/BLIV/HS,DELSTS,HU,DELSTU
      COMMON/DER1/DDELDX,DUBDX,DUTDX,DUIDX
      COMMON/DER2/DELT,UB,UT,UI ,VT,VB,UDUI,TAUM,H,THETA,DELST,CFD2,
     $VISCOS,NBL
      COMMON/EFCVAL/X(90),Y(90),AL(90),ALNV(90)
```

```
      COMMON/GEOM1/XD,W1,TH,WIDTH,X1,B1,SINTH,COSTH,TWOTHR,TWOTH1,
     $SINTH1,COSTH1,AS,WH,XC1,X2,XMAX,XDE
      COMMON/GEOM2/N ,NR,NL,NU,NM1 ,NLC,NRC
      COMMON/NSTD/ID1,ID2,ID3,NST,SWT(90)
      COMMON/ODE1S/JSTRTS,JENDS,NDIM,SW(90),WI(90),DWI(90),DDWI(90),
     $DS(90)
      COMMON/ODE1U/JSTRTU,JENDU,SWU(90),WIU(90)
      COMMON/INITAL/DELST1,H1,UI1,IPR1
      COMMON/PRINT/IPR,NORMPR,CPEROR,ITMAX
      COMMON/SPLYN/XX,WT,DWT,DDWT,ISETUP,KMID
      COMMON/TEMP1/XC,IWALLV
      COMMON/WALVAL/XW(90),YW(90),ALW(90)
      WRITE(6,897)
  897 FORMAT('1****NON-STANDARD DUCT, USER INPUTTED WALL COORDINATES'//
     $' NODE#',T10,'XW',T25,'YW',T40,'ALW'/)
      READ(5,900)N,NR,NL
      READ(5,910) (XW(J),YW(J),ALW(J),J=1,N)
      WRITE(6,899) (J,XW(J),YW(J),ALW(J),J=1,N)
      IF(XW(N).EQ.0.0.AND.YW(N).EQ.0.0)GO TO 50
      WRITE(6,915) XW(N),YW(N)
      STOP
   50 READ(5,910)W1,TWOTHD,AS
      READ(5,911)B1,UI,VISCOS,XC
      IF(XC.EQ.0.0) XC=1.E5
      READ(5,920)IPR,NORMPR,ITMAX,CPEROR
      READ(5,925)HS,DELSTS,HU,DELSTU
      H=HS
      H1=HS
      DELST=DELSTS
      DELST1=DELSTS
      IF(HU.NE.0.0)GO TO 60
      HU=H1
      DELSTU=DELST1
   60 CONTINUE
      WRITE(6,930)N,NR,NL
      WRITE(6,935)W1,TWOTHD,AS,XC
      WRITE(6,940)B1,UI,VISCOS
      WRITE(6,945)HS,DELSTS,HU,DELSTU
      WRITE(6,950)IPR,NORMPR,CPEROR,ITMAX
C------IF THIS IS AN INVISCID CALCULATION(NOBL),RETURN TO MAIN,AFTER SETTING
C       EFCVALUES=WALLVALUES.
      IF(ID2.NE.2)GO TO 70
      DO 65 J=1,N
      X(J)=XW(J)
      Y(J)=YW(J)
   65 AL(J)=ALW(J)
      RETURN
   70 THETA=DELST/H
      NM1=N-1
      NRC=NR
      NLC=NRC+1
      NRCP1=NRC+1
      NMNLC=N-NLC
      H1=H
      UI1=UI
      DELST1=DELST
      IPR1=IPR
C------FIND ARC LENGTHS BETWEEN INPUTTED WALL COORDINATES USING A
C       STRAIGHT LINE APPROXIMATION.
      SW(1)=0.0
      SW(2)=SQRT((XW(1)-XW(N))**2+(YW(1)-YW(N))**2)
```

```
      DO 100 J=2,NRC
      JM1=J-1
  100 SW(J+1)=SW(J)+SQRT((XW(J)-XW(JM1))**2+(YW(J)-YW(JM1))**2)
      SWU(1)=0.0
      DO 110 J=2,NMNLC
      JM1=J-1
      NMJ=N-J
      NP=NMJ+1
  110 SWU(J)=SWU(JM1)+SQRT((XW(NMJ)-XW(NP))**2+(YW(NMJ)-YW(NP))**2)
      SCALE=SW(NRCP1)/SWU(NMNLC)
      IF(NRCP1.GT.NMNLC)GO TO 300
C------GREATER NO OF SEGS ON UPPER WALL(NRCP1.LE.NMNLC). SO WILL
C       MAP UPPER WALL COORDINATES ONTO THE LOWER ONE.
  200 DO 210 J=1,NMNLC
  210 SWT(J)=SWU(J)*SCALE
      WI(1)=W1
      K=3
      DO 250 J=2,NMNLC
      JM1=J-1
      NMJ=N-J
      IF(SWT(J).GT.SW(2))GO TO 230
C------SWT(J) IS BETWEEN NODES N AND 1.
      RATIO=SWT(J)/SW(2)
      XT=RATIO*XW(1)
      YT=RATIO*YW(1)
      GO TO 250
C------SWT(J) LIES BETWEEN NODE 1 AND NRC.
  220 K=K+1
  230 IF(SWT(J).GT.SW(K))GO TO 220
      KP1=K+1
      KM1=K-1
      KM2=K-2
      RATIO=(SWT(J)-SW(KM1))/(SW(K)-SW(KM1))
      XT=XW(KM2)+RATIO*(XW(KM1)-XW(KM2))
      YT=YW(KM2)+RATIO*(YW(KM1)-YW(KM2))
  250 WI(J)=SQRT((XW(NMJ)-XT)**2+(YW(NMJ)-YT)**2)
      WRITE(6,955)
      WRITE(6,800) (J,SWU(J),SWT(J),WI(J),J=1,NMNLC)
      GO TO 700
C
C------GREATER NO OF SEGS ON LOWER WALL(NRCP1.GT.NMNLC). MAP LOWER
C       WALL ONTO UPPER ONE.
  300 DO 310 J=1,NRCP1
  310 SWT(J)=SW(J)/SCALE
      WI(1)=W1
      K=1
      DO 350 J=2,NRCP1
      GO TO 330
  320 K=K+1
  330 IF(SWT(J).GT.SWU(K))GO TO 320
      KM1=K-1
      RATIO=(SWT(J)-SWU(KM1))/(SWU(K)-SWU(KM1))
      NMK=N-K
      JM1=J-1
      NM=NMK+1
      XT=XW(NM)+RATIO*(XW(NMK)-XW(NM))
      YT=YW(NM)+RATIO*(YW(NMK)-YW(NM))
  350 WI(J)=SQRT((XW(JM1)-XT)**2+(YW(JM1)-YT)**2)
      DO 360 J=1,NRCP1
  360 SWT(J)=SW(J)
      WRITE(6,960)
```

```
      WRITE(6,800) (J,SW(J),SWT(J),WI(J),J=1,NRCP1)
  700 CONTINUE
      JSTRTS=1
      JSTRTU=1
      JENDS=NRCP1
      JENDU=NMNLC
  800 FORMAT(' ',I5,1P3E15.5)
  899 FORMAT(' ',I5,1P3E15.5)
  900 FORMAT(3I10)
  910 FORMAT(3E10.0)
  911 FORMAT(4E10.0)
  915 FORMAT('0***ORIGIN IMPROPERLY LOCATED,XW(N)=',1PE12.5,'YW(N)=',
     $1PE12.5///)
  920 FORMAT(3I10,E10.0)
  925 FORMAT(4E10.0)
  930 FORMAT('1SEGMENT COUNT, TOTAL=',I2,'    LOWER WALL= ',I2,
     $'        UPPER WALL= ',I2//)
  935 FORMAT('0INLET WIDTH= ',1PE12.5,'(FT),  TWOTHETA= ',1PE12.5,
     $'(DEG),    ASPECT RATIO= ',1PE12.5,'     XC=',1PE12.5//)
  940 FORMAT('0INLET BLOCKAGE= ',1PE12.5,',    INLET  CORE VELOCITY= ',
     $1PE12.5,'(FT/SEC),   KINEMATIC VISCOSITY=',1PE12.5,'(FT2/SEC)'//)
  945 FORMAT('0INLET BL VALUES:LOWER WALL-H=',F5.2,', DELSTS=',
     $E12.5,'(FT)'/T17,'UPPER WALL-H=',F5.2,', DELSTU= ',
     $E12.5,'(FT)'//)
  950 FORMAT('0B.L.PRINT INTERVAL=',I2,',    PRINT TYPE(NORMPR)=',I4,
     $',    MAX CP ERROR=',F7.5,',     MAX # ITERATIONS=',I2//)
  955 FORMAT('-BOUNDARY WIDTH FOR THE FIRST ITERATION--'//
     $' #',T10,'SWU(J)',T25,'SWT(J)',T40,'WI(J)'/)
  960 FORMAT('-BOUNDARY WIDTH FOR THE FIRST ITERATION--'//
     $' #',T10,'SW(J)',T25,'SWT(J)',T40,'WI(J)'/)
      RETURN
      END
      SUBROUTINE TBLPS
C------CALCULATE TURBULENT BOUNDARY LAYER PARAMETERS WITH SPECIFIED
C       PRESSURE GRADIENT.
      EXTERNAL DERPS
      REAL KAP,VALSM1(3),RATEM1(3)
      REAL XP(3),YP(3),ZP(3)
      INTEGER KMID,JSTART,JEND,JTBL,IRUNGE,IEXIT
      COMMON/ADAM1/X ,VALS(4),RATES(4),RATE(4,8)
      COMMON/BLIV/HS,DELSTS,HU,DELSTU
      COMMON/DER1/DDELDX,DUBDX,DUTDX,DUIDX1
      COMMON/DER2/DELT,UB,UT,UI1,VT,VB,UDUI,TAUM,H,THETA,DELST,CFD2,
     $VISCOS,NBL
      COMMON/LAG/XO,TAUML,TAUMEQ
      COMMON/ODE1S/JSTRTS,JENDS,NDIM,SW(90),VI(90),DVI(90),DDVI(90),
     $DS(90)
      COMMON/ODE1U/JSTRTU,JENDU,SWU(90),VIU(90)
      COMMON/ODE2S/JTBLS,STBLS(90),DSTARS(90),UI1DS(90),DELTS(90),
     $UI2DS(90)
      COMMON/ODE2U/JTBLU,STBLU(90),DSTARU(90),UI1DU(90),DELTU(90),
     $UI2DU(90)
      COMMON/PRINT/IPR,NORMPR,UERR,ITMAX
      COMMON/SPLYN/XX,UI,DUIDX,DDUI,ISETUP,KMID
      COMMON/TEMP1/XCMX,IWALLV
      COMMON/TEMP2/IEXIT,VEL(90)
      EQUIVALENCE (RATEM1(1),DDELDX),(VALSM1(1),DELT)
C------IMPOSED PRESSURE GRAD IS OBTAINED VIA CALL TO SPLINE.
C      THE EQUIVALENCING IMPLICITLY SETS DELT,UB,UT EQUAL TO VALS.
C------SET UP THE EXTERNALLY IMPOSED PRESSURE FIELD
      IF(IWALLV.EQ.1)GO TO 20
```

```
      IF(XX.LT.0.0)XX=SW(1)
      X=XX
      XMAX=SW(JENDS)
      ISETUP=0
      KMID=JSTRTS+1
      CALL SPLINE(SW,VEL,DVI,DDVI,DS,JSTRTS,JENDS,NDIM,4)
      GO TO 30
   20 IF(XX.LT.0.0)XX=SWU(1)
      X=XX
      XMAX=SWU(JENDU)
      ISETUP=0
      KMID=JSTRTU+1
      CALL SPLINE(SWU,VIU,DVI,DDVI,DS,JSTRTU,JENDU,NDIM,4)
C
C------INITIALIZE COUNTERS, COMPUTE START VALUES OF UT,UB,DELT.
   30 JTBL=0
      NLOOP=0
      UIREF=UI
      CALL START
      CALL TAUMAX(TAUM)
      XO=X
      HO=H
      TAUML=TAUMEQ
      TAUM=TAUMEQ
      WRITE(6,900)
      IRUNGE=1
      DX=DELST
      IEXIT=0
      DO 50 J=1,3
      VALS(J)=VALSM1(J)
      RATEM1(J)=0.0
   50 RATES(J)=0.0
      HSEP=1.+1./(1.-DELST/DELT)
C
C*************************************************************
C------BEGIN MAIN LOOP.
C------PRINT INITIAL VALUES.
      GO TO 105
  100 NLOOP=NLOOP+1
      HSEP=1.+1./(1.-DELST/DELT)
C------IF IWALLV=0(LOWER WALL) AND H>=0.9*HSEP OR H<HO,SWITCH OVER TO
C       LINEAR V.P. ITERATION. RETURN TO CALLING ROUTINE AFTER
C       SETTING B.L. VALUES TO THAT AT THE LAST PRINTOUT.
      IF(IWALLV.EQ.1)GO TO 106
      CP=1.0-(UI/UIREF)**2
      IF (CP.LE.0.2)GO TO 107
      IF(H.LT.0.9*HSEP.AND.H.GT.HO)GO TO 107
      WRITE(6,920)X
  920 FORMAT(' ****H>.9*HSEP OR H<HO AT X=',1PE12.5)
      H=HO
      UB=UBO
      UT=UTO
      DUIDX1=DUIDXO
      TAUM=TAUMO
      IF(IWALLV.EQ.1)GO TO 104
      XX=STBLS(JTBL)
      DELST=DSTARS(JTBL)
      DELT=DELTS(JTBL)
      UI1=UI1DS(JTBL)
      JTBLS=JTBL
      RETURN
```

```
C
  104 XX=STBLU(JTBL)
      DELST=DSTARU(JTBL)
      DELT=DELTU(JTBL)
      UI1=UI1DU(JTBL)
      JTBLU=JTBL
      RETURN
C
C------IF H>=HSEP, AND UPPER WALL, THEN SET DELST=CONST TO EXIT.
  106 IF(H.LT.0.9*HSEP)GO TO 107
      WRITE(6,930)X
  930 FORMAT(' INTERMITTENT SEPARATION AT X=',1PE12.5,
     $'FT, SET DELST=CONST TO EXIT'/)
      JTBLU=JTBL
      RETURN
C
C------STORE CURRENT VALUES.
  107 DO 110 J=1,3
      VALSM1(J)=VALS(J)
  110 RATEM1(J)=RATES(J)
C------STORE THE LAG PARAMETERS.
      XO=X
      TAUML=TAUM
C------PRINT CURRENT VALUES.
      XX=X
      IF(MOD(NLOOP,IPR).NE.0)GO TO 150
  105 JTBL=JTBL+1
      HSEP=1.+1./(1.-DELST/DELT)
      DQDX=10.0*TAUM/UI**2
      CP=1.0-(UI/UIREF)**2
      WRITE(6,910)XX,DELST,H,HSEP,CP,DELT,UB,UT,UI,CFD2,DQDX
      IF(IWALLV.EQ.1)GO TO 120
      STBLS(JTBL)=XX
      DSTARS(JTBL)=DELST
      DELTS(JTBL)=DELT
      UI1DS(JTBL)=UI
      GO TO 130
  120 STBLU(JTBL)=XX
      DSTARU(JTBL)=DELST
      DELTU(JTBL)=DELT
      UI1DU(JTBL)=UI
  130 HO=H
      UBO=UB
      UTO=UT
      DUIDXO=DUIDX
      TAUMO=TAUM
C
      IF(IEXIT.EQ.1)GO TO 260
  150 CALL ADAMS(DX,3,DERPS,IRUNGE)
      IF(X.LT.XMAX)GO TO 100
C
C------EXIT VALUE COMPUTATIONS.
C       OBTAIN VALUES AT X=XMAX BY EXTRAPOLATION.
      IEXIT=1
      XX=XMAX
      IF(IWALLV.EQ.1)GO TO 165
      DO 160 J=1,3
      JJ=JTBL-3+J
      XP(J)=STBLS(JJ)
      YP(J)=DSTARS(JJ)
  160 ZP(J)=UI1DS(JJ)
```

```
      GO TO 170
  165 DO 168 J=1,3
      JJ=JTBL-3+J
      XP(J)=STBLU(JJ)
      YP(J)=DSTARU(JJ)
  168 ZP(J)=UI1DU(JJ)
  170 DELST=YINT(XP,YP,XMAX)
      UI=UI2DU(JENDU)
      JTBLS=JTBL+1
      JTBLU=JTBL+1
      GO TO 105
  260 CONTINUE
  900 FORMAT(1H0,'  X                  DSTAR            H       HSEP            CP      DE
     $LT        UB          UT              UI             CF/2               DQDX/UI'/)
  910 FORMAT(2F10.5,4X,2F7.3,2X,F6.3,4F12.5,F12.6,F12.5)
      RETURN
      END
      SUBROUTINE BLVALU
C------GIVEN UT,UI,UB,DELT, COMPUTE B.L. THICKNESSES DSTAR,D2STAR,
C       AND SHAPE FACTORS H AND HBAR. ALSO CF/2=CFD2.
      COMMON/BLOLD/ETOLD,UMINO,UZOLD
      COMMON/DER1/DDELDX,DUBDX,DUTDX,DUIDX1
      COMMON/DER2/DELT,UB,UT,UI1,VT,VB,UDUI,TAUM,H,THETA,DSTAR,CFD2,
     $VISCOS,NBL
      COMMON/LAG/XO,TAUMO,TAUMEQ
      COMMON/SPLYN/X,UI,DUI,DDUI,ISETUP,KMID
      REAL PI,PID2
      DATA PI,PID2/3.141593,1.57079/
      VT=UT/(0.41*UI)
      VB=UB/UI
      DSDDEL=VT+0.5*VB
      DSTAR=DSDDEL*DELT
      THDDEL=DSDDEL-2.0*VT**2-0.375*VB**2-1.58949*VT*VB
      THETA=THDDEL*DELT
      H=DSTAR/THETA
      CFD2=(UT/UI)**2
      IF(UT.LT.0.0) CFD2=-CFD2
      RETURN
C
C**********************
C
      ENTRY START
C------GIVEN H,DSTAR,UI, FIND UT, UB. ZI=DSTAR/DELT.
C       OBTAINED BY FITTING COLES' WALL/WAKE PROFILE TO THE INPUT .
C------INITIALIZE LAG EQUATION PARAMETERS, AND TAUM.
      XO=X
      IF(TAUM.LE.0.0)TAUM=10.0
      TAUMO=TAUM
      REDST=UI*DSTAR/VISCOS
C------USE FLAT PLATE VALUES FOR FIRST GUESS.
      VT=0.01685/REDST**0.166667
      ZI=0.125
      VB=(ZI-VT)*2.0
      HDHM1=H/(H-1.0)
      NOUT=0
   99 NL2=0
      NLOOP=0
      ADST=DSTAR*0.41*UI/VISCOS
  100 VTO=VT
      ZIO=ZI
      NLOOP=NLOOP+1
```

```
      IF(10.GE.NLOOP)GO TO 150
      WRITE(6,900)H,DSTAR,UI,UT,UB,VT
      STOP
  150 FVT=VT*(2.05+ALOG(ADST*ABS(VT)/ZI))+2.0*(ZI-VT)-1.0
      FPVT=1.05+ALOG(ADST*ABS(VT)/ZI)
      VT=VT-FVT/FPVT
      IF(0.0001.LT.ABS(1.0-VTO/VT))GO TO 100
C------BEGIN LOOP FOR VB ITERATION.
  170 VBO=VB
      NL2=NL2+1
      IF(10.GE.NL2)GO TO 180
      WRITE(6,915)H,DSTAR,UI,UT,UB,ZI
      STOP
  180 FVB=HDHM1*(2.0*VT*VT+.375*VB*VB+1.598949*VT*VB)-VT-0.5*VB
      FPVB=HDHM1*(0.75*VB+1.598949*VT)-0.5
      VB=VB-FVB/FPVB
      IF(0.0001.LT.ABS(1.0-VBO/VB))GO TO 170
      ZI=HDHM1*(2.0*VT*VT+0.375*VB*VB+1.598949*VT*VB)
      IF(0.0001.GT.ABS(1.0-ZI/ZIO))GO TO 200
      NOUT=NOUT+1
      IF(NOUT.LE.10)GO TO 99
      WRITE(6,910)H,DSTAR,UI,UT,UB,ZI
      STOP
  200 UT=VT*.41*UI
      UB=VB*UI
      DELT=DSTAR/ZI
      CFD2=(UT/UI)**2
      IF(UT.LE.0.0)CFD2=-CFD2
      WRITE(6,920)UT,UB,DELT,DSTAR,H
C
C------INITIALIZE STARTING GUESSES FOR TAUMAX.
C
      ETOLD=0.25
C
  900 FORMAT(' VT FAILED TO CONVERGE IN 10 ITERS     ,H,DSTAR,UI,UT,UB,VT
     $ = ',6E12.5)
  910 FORMAT('  ZI FAILED TO CONVERGE IN 10 ITERS     ,H,DSTAR,UI,UT,UB,Z
     $I = ',6E12.5//)
  915 FORMAT(' VB FAILED TO CONVERGE IN 10 ITERS,H,DSTAR,UI,UT,UB,VB=',
     $ 6E12.5//)
  920 FORMAT(' START VALUES,UT= ',F10.5,'  UB= ',F10.5,' DELTA= ',
     $F10.5,' DELST= ',F10.5,'    H= ',F10.5/)
      RETURN
C
C**********
C
      ENTRY TAUMAX(TAULAG)
C------GIVEN UDUI AT WHICH TAU IS MAX,FIND THE CORRESPONDING ETA=Y/DELT
C      ,MAX DUDY, AND THE MAX SHEAR TAUM/RHO.
C      UDUI IS SET=.76 FOR ATTACHED FLOWS AND =0.6 FOR DETACHED FLOWS.
      UDUI=0.76
      IF(UT.LE.0.0)UDUI=0.60
      UDM1=1.0-UDUI
      ET=ETOLD
      IF(DUI.LT.0.0)GO TO 290
      ET=0.25
      GO TO 320
  290 NL=0
  300 ETO=ET
      IF(ET.GT.0.0)GO TO 310
      WRITE(6,960)ET
```

```
      ET=0.25
      GO TO 320
  310 NL=NL+1
      IF(NL.GT.10)GO TO 500
      FET=VT*ALOG(ET)-VB*(COS(PID2*ET))**2+UDM1
      FPET=VT/ET+PID2*VB*SIN(PI*ET)
      ET=ET-FET/FPET
      IF(0.0001.LT.ABS(1.0-ET/ETO))GO TO 300
      IF(ET.LT.0.25)ET=0.25
      ETOLD=ET
  320 DUDY=(UI/DELT)*(VT/ET+VB*PID2*SIN(PI*ET))
C------USING KUHN-NIELSEN'S EDDY VISCOSITY, WHICH INCLUDES EFFECTS
C       OF INTERMITTENCY AND PRESSURE GRADIENT PARAMETER,
C       BETA IS THE CLAUSER PRESSURE GRADIENT PARAMETER. EPS IS EDDY
C       VISCOSITY, GAMMA IS INTERMITTENCY.
C       SET EPS=0.013 THE FREE SHEAR LIMIT FOR FLOWS THAT
C       ARE NEAR AND BEYOND DETACHMENT.
      E2=0.0
      IF(UT.LE.0.5)GO TO 340
      E2=0.0038
      BETA=-DSTAR*UI*DUI/(15.0*UT*UT)
      IF(DUI.GE.0.0.OR.ABS(BETA).GE.174.0)GO TO 340
      E2=0.0038*EXP(-BETA)
  340 EPS=0.013+E2
      GAMMA=1.0/(1.0+9.0*ET**6)
      TAUMEQ=EPS*GAMMA*DUDY*UI*DSTAR
C------INCLUDE SHEAR STRESS HISTORY BY LAGGING THE EQUIL STRESS
      HLAM=0.025
      IF(UT.LE.0.0)HLAM=0.70
      DTMDX=HLAM*(TAUMEQ-TAUMO)/DELT
      TAULAG=TAUMO+DTMDX*(X-XO)
      RETURN
  500 WRITE(6,950)NL,ET,UT,UI,UB
      STOP
  950 FORMAT(1H0,' ETA FAILED TO CONVERGE IN',I4,'ITERATIONS,ET,UT,UI,UB
     $= ',4E15.5)
  960 FORMAT(' ET SET=0.25, OLD VALUE WAS ',F10.5)
      END
      SUBROUTINE PSTEST
C------DRIVER ROUTINE TO TEST TURBULENT BOUNDARY LAYER CALCULATION
C       WITH SPECIFIED PRESSURE GRADIENT.
C       SUBROUTINES REQUIRED: ADAMS,BLVALU,DERPS,FACTOR,PSTEST,RKS4,
C       SPLINE,SUBST,TRIDIAG.
C------****ROUTINE TO TEST NEW BOUNDARY LAYER PREDICTION METHOD
C       USING TAUMAX-ENTRAINMENT CORRELATION.
C
C
      COMMON/BLIV/HS,DELSTS,HU,DELSTU
      COMMON/DER1/DDELDX,DUBDX,DUTDX,DUIDX1
      COMMON/DER2/DELT,UB,UT,UI1,VT,VB,UDUI,TAUM,H,THETA,DELST,CFD2,
     $VISCOS,NBL
      COMMON/ODE1S/JSTRTS,JENDS,NDIM,SW(90),VI(90),DVI(90),DDVI(90),
     $DS(90)
      COMMON/ODE1U/JSTRTU,JENDU,SWU(90),VIU(90)
      COMMON/ODE2U/JTBLU,STBLU(90),DSTARU(90),UI1DU(90),DELTU(90),
     $UI2DU(90)
      COMMON/PRINT/IPR,NORMPR,CPEROR,ITMAX
      COMMON/SPLYN/XX,UI,DUI,DDUI,ISETUP,KMID
      COMMON/TEMP1/XC,IWALLV
      NDIM=90
C------ENTER THE IMPOSED VELOCITY DISTRIBUTION
```

```
      X=0.0
C------READ IN THE B.L.DATA. END LAST CASE WITH 2 BLANK CARDS.
C------DELST AND THETA ARE IN FT.
C       TAUM IS THE STARTING VALUE OF THE MAX SHEAR STRESS.
      READ(5,902)NPTS
      IF(NPTS.LE.0) GO TO 800
      READ(5,904)XX,DELST,H,VISCOS
      READ(5,905)IPR,XC
      IF(XC.EQ.0.0)XC=1.E5
      IF(IPR.LE.0)IPR=2
      WRITE(6,906)XX,DELST,H,VISCOS,XC,IPR
      READ(5,910) (SWU(I),VIU(I),I=1,NPTS)
      WRITE(6,915)
          WRITE(6,920) (SWU(I),VIU(I),I=1,NPTS)
      THETA=DELST/H
      JSTRTU=1
      JENDU=NPTS
      DELSTU=DELST
      HU=H
      IWALLV=1
      UI2DU(JENDU)=VIU(JENDU)
      CALL TBLPS
      WRITE(6,930)
  800 CONTINUE
  902 FORMAT(I10)
  904 FORMAT(4E10.5)
  905 FORMAT(I10,E10.5)
  906 FORMAT(1H0,'X,DELST,H,VISCOS,XC,IPR=   ',1P5E12.4,I5///)
  910 FORMAT(8E10.0)
  915 FORMAT(1H0,'                  X                    UI'/)
  920 FORMAT(1H0,2E20.5)
  930 FORMAT(1H0,'***IN MAIN ROUTINE***')
      RETURN
      END
      SUBROUTINE INVCID
      CALL PFSL
      CALL OUTINT(IPOINT)
      RETURN
      END
      SUBROUTINE OUTINT(IPOINT)
C------COMPUTE FUNCTION VALUES AND DERIVATIVES AT INTERIOR POINTS.
      REAL*8 A(86,87)
      REAL X0(120),Y0(120),LNV(90),TX0(120),TY0(120)
      COMPLEX FPZ(90),ETA(120),C(91),FPZ0(120),FPPZ0(120),Z0(120)
      COMPLEX ZO,ZERO,ITWOPI,CMPLX,CEXP,CONJG,FPZO,PGRAD,PGRADS
      COMPLEX*16 GAMMA1(120),GAMMA2(120)
      COMPLEX*16 ETAJ,ETAJP,ETAJM,ERP,LERP,CDLOG,FDZO,FDDZO
      COMMON/EFCVAL/XC(90),YC(90),AL(90),LNV
      COMMON/GEOM1/XD,XL,TH,WIDTH,X1,B1,SINTH,COSTH,TWOTHR,TWOTH1,
     $SINTH1,COSTH1,AS,WH,XCE,X2,XMAX,XDE
      COMMON/GEOM2/N,NR,NL,NU,NM1,NLC,NRC
      COMMON/INITAL/DELST1,H1,VIN,IPR1
      COMMON/PFSL1/C,A,X0,Y0
      COMMON/PRINT/IPR,NORMPR,UERR,ITMAX
      COMMON/WALVAL/XW(90),YW(90),ALW(90)
      ZERO=(0.0,0.0)
      ITWOPI=(0.0,6.283185)
      READ(5,901)LINES
C------LINES=#LINES ALONG WHICH ANALYTIC FUNCTION AND DERIVS ARE CALCULATED.
      IF(LINES.EQ.0) RETURN
      IF(LINES.LT.0)GO TO 715
```

```
      IPOINT=0
      DO 714 L=1,LINES
C------ENTER 1 CARD FOR EACH LINE ALONG WHICH THE INTERIOR POINTS
C       ARE TO BE CALCULATED. (X1,Y1),(X2,Y2), ARE START AND END
C       POINTS OF LINE, AND NSEGS IS NO. OF SEGMENTS ALONG LINE.
      READ(5,951)X1,Y1,X2,Y2,NSEGS
      IF(NSEGS.GT.0)GO TO 712
      IPOINT=IPOINT+1
      XO(IPOINT)=X1
      YO(IPOINT)=Y1
      GO TO 714
  712 NPOINT=NSEGS+1
      DX=(X2-X1)/NSEGS
      DY=(Y2-Y1)/NSEGS
      DO 713 J=1,NPOINT
      JM1=J-1
      IPOINT=IPOINT+1
      XO(IPOINT)=X1+DX*JM1
  713 YO(IPOINT)=Y1+DY*JM1
  714 CONTINUE
  715 NP1 = N+1
      VSCALE = VIN
      DO 717 J=1,N
  717 FPZ(J) = CEXP(CMPLX(LNV(J),-AL(J)))
      WRITE(6,950) IPOINT
      DO 760 K=1,IPOINT
      ZO = CMPLX(XO(K),YO(K))/XL
      ZO(K) = ZO
      C(NP1)=C(1)
      DO 720 J=1,NP1
  720 ETA(J) = C(J)-ZO
      DO 730 J=1,N
      ETAJ = ETA(J)
      ETAJP = ETA(J+1)
      ERP = ETAJP/ETAJ
      LERP = CDLOG(ERP)
      GAMMA1(J) = LERP/(ETAJP-ETAJ)
      GAMMA2(J) = 1.0/(ETAJ*ETAJP)
  730 CONTINUE
      FDZO = ZERO
      FDDZO = ZERO
      DO 740 J=1,N
      JP1 = J+1
      JM1 = J-1
      IF(JM1.EQ.0) JM1 = N
      ETAJ = ETA(J)
      ETAJP = ETA(JP1)
      ETAJM = ETA(JM1)
      FDZO = FDZO+FPZ(J)*(ETAJP*GAMMA1(J)-ETAJM*GAMMA1(JM1))
  740 FDDZO = FDDZO+FPZ(J)*(GAMMA1(JM1)-GAMMA1(J)+ETAJP*GAMMA2(J)-
     &  ETAJM*GAMMA2(JM1))
      FPZO(K) = FDZO/ITWOPI
  760 FPPZO(K) = FDDZO/ITWOPI
  850 IF(NORMPR) 860,860,870
C------NORMALIZED NEUMANN PRINTOUT.
  860 WRITE(6,952)
      WRITE(6,960)
      DO 865 K=1,IPCINT
      FPZO = FPZO(K)
      VMAG = CABS(FPZO)
      UT = REAL(FPZO)
```

```
      VI = -AIMAG(FPZO)
      ALPHA = ATAN2(VI,UI)
C------STORE VELOCITY COMPONENTS LOCALLY PARALLEL TO THE WALLS
C       IN TXO AND TYO ARRAYS.
      AMA=ALW(K)-ALPHA
      TXO(K)=VMAG*COS(AMA)
      TYO(K)=VMAG*SIN(AMA)
  865 WRITE(6,955) K,ZO(K),UI,VI,VMAG,ALPHA,K
      IF(NORMPR) 880,870,870
C-------DIMENSIONAL NEUMANN PRINTOUT.
  870 WRITE(6,954)
      WRITE(6,961)
      DO 875 K=1,IPOINT
      FPZO = FPZO(K)*VSCALE
      VMAG = CABS(FPZO)
      UI = REAL(FPZO)
      VI = -AIMAG(FPZO)
      ALPHA = ATAN2(VI,UI)
  875 WRITE(6,956) K,XO(K),YO(K),UI,VI,VMAG,ALPHA,K
C-------PRESSURE AND PRESSURE GRADIENT CALCULATIONS.
  880 IF(NORMPR) 882,882,885
C-------NORMALIZED PRESSURE DATA.
  882 WRITE(6,952)
      WRITE(6,970)
      DO 884 K=1,IPOINT
      FPZO = FPZO(K)
      VMAG = CABS(FPZO)
      VMAGSQ = VMAG*VMAG
C-------CP = (P-PIN)/QIN = 1-(V/VIN)**2
      CP = 1.0-VMAGSQ
      PGRAD = -FPZO*CONJG(FPPZO(K))
      PGRADS = PGRAD*FPZO/VMAG
      CURVE = -AIMAG(PGRADS)/VMAGSQ
  884 WRITE(6,965) K,ZO(K),PGRAD,PGRADS,CURVE,CP,K
      IF(NORMPR) 1000,885,885
C-------DIMENSIONAL PRESSURE DATA.
  885 WRITE(6,954)
      WRITE(6,975)
      VINSQ = VIN*VIN
      VSCXL = VSCALE/XL
      DO 890 K=1,IPOINT
      FPZO = FPZO(K)*VSCALE
      VMAG = CABS(FPZO)
      VMAGSQ = VMAG*VMAG
      PDIFF = (VINSQ-VMAGSQ)/2
      PGRAD = -FPZO*CONJG(FPPZO(K))*VSCXL
      PGRADS = PGRAD*FPZO/VMAG
      CURVE = -AIMAG(PGRADS)/VMAGSQ
  890 WRITE(6,966) K,XO(K),YO(K),PGRAD,PGRADS,CURVE,PDIFF,K
C------RETURN THE VELOCITY COMPONENTS LOCALLY PARALLEL TO THE WALL
C       IN XO AND YO.
 1000 DO 900 J=1,N
      XO(J)=TXO(J)*VSCALE
  900 YO(J)=TYO(J)*VSCALE
      RETURN
  901 FORMAT(I10)
  950 FORMAT(1H0/1H0,24X,'VALUE OF ANALYTIC FUNCTION AND ITS DERIVATIVES
     & AT',I4,' BOUNDARY AND/OR INTERIOR POINTS.')
  951 FORMAT(4F10.0,I10)
  952 FORMAT(1H0,67X,'NORMALIZED VALUES')
  954 FORMAT(1H0,67X,'DIMENSIONAL VALUES')
```

```
955 FORMAT(1H ,27X,I4,6F14.6,I3)
956 FORMAT(1H ,27X,I4,1P6E14.5,I3)
960 FORMAT(1H0,30X,'#',8X,'X0',12X,'Y0',12X,'U',13X,'V',12X,
   &'VEL-MAG',8X,'ALPHA',3X,'#')
961 FORMAT(1H0,30X,'#',7X,'X0',12X,'Y0',12X,'U',13X,'V',12X,
   &'VEL-MAG',7X,'ALPHA',5X,'#')
965 FORMAT(1H ,13X,I4,8F14.6,I3)
966 FORMAT(1H ,13X,I4,1P8E14.5,I3)
970 FORMAT(1H0,16X,'#',8X,'X0',12X,'Y0',7X,'(DP/DX)/RHO',3X,'(DP/DY)/R
   &HO',3X,'(DP/DS)/RHO',3X,'(DP/DN)/RHO',5X,'CURVATURE',8X,'CP',6X,
   &'#')
975 FORMAT(1H0,16X,'#',7X,'X0',12X,'Y0', 8X,'(DP/DX)/RHO',3X,'(DP/DY)/
   &RHO',3X,'(DP/DS)/RHO',3X,'(DP/DN)/RHO',5X,'CURVATURE',3X,
   &'(P-PIN)/RHO',2X,'#')
    END
    SUBROUTINE CONVRT(IWALL)
C------INTERPOLATE AND CHANGE INDICES FROM JTBLS OR JTBLU TO N.
C      DONE BY CALLING SUBROUTINE CHANGE.
    COMMON/CON/SVAL(90),YVAL(90)
    COMMON/GEOM2/N,NR,NL,NU,NM1,NLC,NRC
    COMMON/ODE1S/JSTRTS,JENDS,NDIM,SW(90),WI(90),DWI(90),DDWI(90),
   $DS(90)
    COMMON/ODE1U/JSTRTU,JENDU,SWU(90),WIU(90)
    COMMON/ODE2U/JTBLU,STBLU(90),DSTARU(90),UI1DU(90),DELTU(90),
   $UI2DU(90)
    COMMON/ODE2S/JTBLS,STBLS(90),DSTARS(90),UI1DS(90),DELTS(90),
   $UI2DS(90)
    COMMON/SPLYN/XINT,FINT,FPINT,FPPINT,ISETUP,KMID
    COMMON/TEMP1/XCMX,IWALLV
    IF(IWALL.EQ.1)GO TO 600
    IF(JTBLS.GT.90)GO TO 1500
C
    T1=DSTARS(1)
    T2=DELTS(1)
    T3=UI1DS(1)
    DO 500 K=1,3
    IF(K-2) 100,200,300
100 DO 150 J=1,JTBLS
    SVAL(J)=STBLS(J)
150 YVAL(J)=DSTARS(J)
    CALL CHANGE
    DO 160 J=1,NRC
160 DSTARS(J)=YVAL(J)
    GO TO 500
C
200 DO 250 J=1,JTBLS
250 YVAL(J)=DELTS(J)
    CALL CHANGE
    DO 260 J=1,NRC
260 DELTS(J)=YVAL(J)
    GO TO 500
300 DO 350 J=1,JTBLS
350 YVAL(J)=UI1DS(J)
    CALL CHANGE
    DO 360 J=1,NRC
360 UI1DS(J)=YVAL(J)
500 CONTINUE
C------STARTING VALUES AT NODE N.
    DSTARS(N)=T1
    DELTS(N)=T2
    UI1DS(N)=T3
```

```
      RETURN
C
  600 IF(JTBLU.GT.90)GO TO 1600
      DO 1000 K=1,3
      IF(K-2)700,800,900
  700 DO 750 J=1,JTBLU
      SVAL(J)=STBLU(J)
  750 YVAL(J)=DSTARU(J)
      CALL CHANGE
      DO 760 J=NLC,NM1
  760 DSTARU(J)=YVAL(J)
      GO TO 1000
  800 DO 850 J=1,JTBLU
  850 YVAL(J)=DELTU(J)
      CALL CHANGE
      DO 860 J=NLC,NM1
  860 DELTU(J)=YVAL(J)
      GO TO 1000
  900 DO 950 J=1,JTBLU
  950 YVAL(J)=UI1DU(J)
      CALL CHANGE
      DO 960 J=NLC,NM1
  960 UI1DU(J)=YVAL(J)
 1000 CONTINUE
      RETURN
C------JTBLU OR JTBLS IS GT 90. PRINT ERROR MESSAGE
C       AND STOP. CORRECT BY INCREASING IPR.
 1500 WRITE(6,920)JTBLS
      STOP
 1600 WRITE(6,910)JTBLU
      STOP
  920 FORMAT('-JTBLS=',I2,',WHICH IS .GT.90, INCREASE IPR AND RERUN'//)
  910 FORMAT('-JTBLU=',I2,',WHICH IS .GT.90, INCREASE IPR AND RERUN'//)
      END
      SUBROUTINE TBLSI(IWALL)
C------CALCULATE DSTARS,UI1DS FOR A TBL AND 1-D CORE IN SIMULTANEOUS
C       ITERATION.
C       IWALL=0 IS LOWER WALL. IWALL=1 IS UPPER.
      EXTERNAL DERSI
      REAL   KAP,VALSM1(4),RATEM1(4)
      REAL XP(3),YP(3),ZP(3),UP(3)
      INTEGER KMID,JSTART,JEND,JTBL,IRUNGE,IEXIT
      COMMON/ADAM1/X ,VALS(4),RATES(4),RATE(4,8)
      COMMON/DER1/DDELDX,DUBDX,DUTDX,DUIDX
      COMMON/DER2/DELT,UB,UT,UI,VT,VB,UDUI,TAUM,H,THETA,DELST,CFD2,
     $VISCOS,NBL
      COMMON/GEOM1/XD,W1,TH,WIDTH,X1,B1,SINTH,COSTH,TWOTHR,TWOTH1,
     $SINTH1,COSTH1,AS,WH,XC,X2,XMAX,XDE
      COMMON/GEOM2/N,NR,NL,NU,NM1,NLC,NRC
      COMMON/INITAL/DELST1,H1,UI1,IPR1
      COMMON/LAG/XO,TAUMO,TAUMEQ
      COMMON/ODE1S/JSTRTS,JENDS,NDIM,SW(90),WI(90),DWI(90),DDWI(90),
     $DS(90)
      COMMON/ODE1U/JSTRTU,JENDU,SWU(90),WIU(90)
      COMMON/ODE2S/JTBLS,STBLS(90),DSTARS(90),UI1DS(90),DELTS(90),
     $UI2DS(90)
      COMMON/ODE2U/JTBLU,STBLU(90),DSTARU(90),UI1DU(90),DELTU(90),
     $UI2DU(90)
      COMMON/PRINT/IPR,NORMPR,UERR,ITMAX
      COMMON/SPLYN/XX,WT,DWT,DDWT,ISETUP,KMID
      COMMON/TEMP1/XCMX,IWALLV
```

```
      EQUIVALENCE (RATEM1(1),DDELDX),(VALSM1(1),DELT)
      IWALLV=IWALL
C
C------THE CHANNEL WIDTH IS OBTAINED VIA CALL TO SPLINE.
C       EQUIVALENCING IMPLICITLY SETS DELT,UB,UT,UI EQUAL TO VALS.
C       WH=DIFFUSER HEIGHT, AND, IF NOT SPECIFIED IS SET
C       TO AN AVERAGE ASPECT RATIO OF 8. XC IS THE LOCATION
C       OF THE FICTITIOUS SOURCE TO CORRECT FOR 3-D EFFECTS.
C       XCMX=XC-X, THE DISTANCE OF THE SOURCE FROM PRESENT X.
C------SET UP THE CHANNEL WIDTH AS A FUNCTION OF XX.
C
      JTBL=0
      ISETUP=0
      IF(IWALL.EQ.1)GO TO 25
C------IF(XX.GT.0), THEN THIS IS A CONTINUATION OF BL ROUTINE
C       FOR WHICH H>HSEP.
      IF(XX.GT.0.0) JTBL=JTBLS-1
      X=XX
      XMAX=SW(NRC+1)
      KMID=JSTRTS+1
      CALL SPLINE(SW,WI,DWI,DDWI,DS,JSTRTS,JENDS,NDIM,4)
      GO TO 30
   25 IF(XX.GT.0.0) JTBL=JTBLU-1
      X=XX
      XMAX=SWU(N-NLC)
      CALL SPLINE(SWU,WIU,DWI,DDWI,DS,JSTRTU,JENDU,NDIM,4)
   30 CONTINUE
C
C------INITIALIZE COUNTERS AND COMPUTE THE START VALUES OF DELT,UB,UT.
C       START VALUE FOR UI WAS READ IN AND PASSED THRU COMMON/DER2/
      NLOOP=0
      IRUNGE=1
      DX=DELST
      IEXIT=0
      UIREF=UI1
      Q=UIREF*(W1-NBL*DELST1)
      CPCOEF=1.0/(1.0-NBL*DELST1/W1)**2
      WRITE(6,940)UIREF,Q,CPCOEF
      WH=AS*W1
      XCMX=1.E4
C------IF THIS IS NEW RUN, CALL START.
      WT=UI
      IF(XX.GT.0.0) GO TO 40
      DWT=0.0
      CALL START
   40 DO 50 J=1,4
      VALS(J)=VALSM1(J)
      IF(XX.EQ.0.0) RATEM1(J)=0.0
   50 RATES(J)=RATEM1(J)
C------INITIALIZE LAG PARAMS WITH EQUILIBRIUM VALUES.
      IF(XX.GT.0.0)GO TO 60
      CALL TAUMAX(DUMMY)
      XO=X
      TAUMO=TAUMEQ
      TAUM=TAUMEQ
   60 WRITE(6,900)
C------PRINT INITIAL VALUES.
      GO TO 105
C
C------BEGIN MAIN LOOP ****************
  100 NLOOP=NLOOP+1
```

```
C------STORE CURRENT VALUES.
      DO 110 J=1,4
      VALSM1(J) =VALS(J)
  110 RATEM1(J) =RATES(J)
      XX=X
C------STORE THE LAG PAREMETERS.
      XO=X
      TAUMO=TAUM
      IF((UB-UI)*UT.LE.0.0)GO TO 103
C       CHANGE THE SIGN OF UT TO REMOVE THE DOUBLE-VALUEDNESS OF UT.
      UT=-UT
      VALS(3)=UT
      WT=UI
      CALL BLVALU
  103 CONTINUE
C------PRINT CURRENT VALUES.
      IF(MOD(NLOOP,IPR).NE.0)GO TO 150
  105 JTBL=JTBL+1
      HSEP=1.+1./(1.-DELST/DELT)
      DQDX=10.0*TAUM/UI**2
      CP=1.0-(UI/UIREF)**2
      WRITE(6,910)XX,DELST,H,HSEP,CP,DELT,UB,UT,UI,CFD2,DQDX
      IF(IWALL.EQ.1)GO TO 130
      STBLS(JTBL)=XX
      DSTARS(JTBL)=DELST
      UI1DS(JTBL)=UI
      DELTS(JTBL)=DELT
      GO TO 140
  130 STBLU(JTBL)=XX
      DSTARU(JTBL)=DELST
      UI1DU(JTBL)=UI
      DELTU(JTBL)=DELT
  140 CONTINUE
      IF(IEXIT.EQ.1)GO TO 260
  150 CALL ADAMS(DX,4,DERSI,IRUNGE)
      IF(X.LT.XMAX)GO TO 100
C
C------EXIT VALUE CALCULATIONS.
C       OBTAIN VALUES AT X=XMAX BY EXTRAPOLATION.
      IEXIT=1
      XX=XMAX
      DO 160 J=1,3
      JJ=JTBL-3+J
      XP(J)=STBLS(JJ)
      YP(J)=DSTARS(JJ)
      UP(J)=DELTS(JJ)
  160 ZP(J)=UI1DS(JJ)
      DELST=YINT(XP,YP,XMAX)
      UI=YINT(XP,ZP,XMAX)
      DELT=YINT(XP,UP,XMAX)
C------IF LAST TWO VALUES OF X ARE VERY CLOSE TOGETHER, THEN CAN HAVE
C       PROBLEMS WITH SPLINE-ELIMINATE LAST BUT ONE POINT.
      IF(XMAX-STBLS(JTBL-1).LT.0.005*(STBLS(JTBL-1)-STBLS(JTBL-2)))
     $JTBL=JTBL-1
      JTBLS=JTBL+1
      GO TO 105
  260 CONTINUE
  900 FORMAT(1H0,'  X              DSTAR          H       HSEP       CP     DE
     *LT         UB          UT          UI          CF/2           DQDX/UI'/)
  910 FORMAT(2F10.5,4X,2F7.3,2X,F6.3,4F12.5,F12.6,F12.5)
  940 FORMAT(1H ,' REFERENCE QUANTITIES,VELOCITY=',F10.5,'VOLUME FLOWRAT
```

```
     $E=',F10.5,'CP MULTIPLIER=',F10.5)
      RETURN
      END
      SUBROUTINE TBLSIL(IWALL)
C------CALCULATE DSTARS,UI1DS FOR A TBL AND 1-D CORE IN SIMULTANEOUS
C       ITERATION, ASSUMING LINEAR VEL PROFILE ACROSS CHANNEL.
C       IWALL=0 IS LOWER WALL. IWALL=1 IS UPPER.
      EXTERNAL DERSIL
      REAL   KAP,VALSM1(4),RATEM1(4)
      REAL XP(3),YP(3),ZP(3),UP(3)
      INTEGER KMID,JSTART,JEND,JTBL,IRUNGE,IEXIT
      COMMON/ADAM1/X ,VALS(4),RATES(4),RATE(4,8)
      COMMON/DER1/DDELDX,DUBDX,DUTDX,DUIDX
      COMMON/DER2/DELT,UB,UT,UI,VT,VB,UDUI,TAUM,H,THETA,DELST,CFD2,
     $VISCOS,NBL
      COMMON/GEOM1/XD,W1,TH,WIDTH,X1,B1,SINTH,COSTH,TWOTHR,TWOTH1,
     $SINTH1,COSTH1,AS,WH,XC1,X2,XMAX,XDE
      COMMON/GEOM2/N,NR,NL,NU,NM1,NLC,NRC
      COMMON/INITAL/DELST1,H1,UI1,IPR1
      COMMON/LAG/XO,TAUMO,TAUMEQ
      COMMON/LINEAR/WDIF(90),DU2D(90),DDU2D(90),WMD,DWEDX,UEFF,DUEDX
      COMMON/ODE1S/JSTRTS,JENDS,NDIM,SW(90),WI(90),DWI(90),DDWI(90),
     $DS(90)
      COMMON/ODE1U/JSTRTU,JENDU,SWU(90),WIU(90)
      COMMON/ODE2S/JTBLS,STBLS(90),DSTARS(90),UI1DS(90),DELTS(90),
     $UI2DS(90)
      COMMON/ODE2U/JTBLU,STBLU(90),DSTARU(90),UI1DU(90),DELTU(90),
     $UI2DU(90)
      COMMON/PRINT/IPR,NORMPR,CPEROR,ITMAX
      COMMON/SPLYN/XX,WT,DWT,DDWT,ISETUP,KMID
      COMMON/TEMP1/XC,IWALLV
      EQUIVALENCE (RATEM1(1),DDELDX),(VALSM1(1),DELT)
      IWALLV=IWALL
C
C------THE CHANNEL WIDTH IS OBTAINED VIA CALL TO SPLINE.
C       EQUIVALENCING IMPLICITLY SETS DELT,UB,UT,UI EQUAL TO VALS.
C       WH=DIFFUSER HEIGHT, AND, IF NOT SPECIFIED IS SET
C       TO AN AVERAGE ASPECT RATIO OF 8. XC IS THE LOCATION
C       OF THE FICTITIOUS SOURCE TO CORRECT FOR 3-D EFFECTS.
C       XC IS  THE DISTANCE OF THE SOURCE FROM THE ORIGIN.
C       WIU CONTAINS UI2DU, WI CONTAINS (WDIF-DSTARU)
C
      JTBL=0
      ISETUP=0
C------IF(XX.GT.0), THEN THIS IS A CONTINUATION OF BL ROUTINE
C       FOR WHICH CP>0.3.
      IF(XX.GT.0.0)JTBL=JTBLS-1
      X=XX
      XMAX=SW(NRC+1)
      KMID=JSTRTS+1
      CALL SPLINE(SW,WI,DWI,DDWI,DS,JSTRTS,JENDS,NDIM,4)
      ISETUP=0
      CALL SPLINE(SW,WIU,DU2D,DDU2D,DS,JSTRTS,JENDS,NDIM,4)
C
C------INITIALIZE COUNTERS AND COMPUTE THE START VALUES OF DELT,UB,UT.
C       START VALUE FOR UI WAS READ IN AND PASSED THRU COMMON/DER2/
      NLOOP=0
      IRUNGE=1
      DX=DELST
      IEXIT=0
      UIREF=UI1
```

```
      Q=UIREF*(W1-NBL*DELST1)
      CPCOEF=1.0/(1.0-NBL*DELST1/W1)**2
      WRITE(6,940) UIREF,Q,CPCOEF
      WH=AS*W1
C------IF THIS IS NEW RUN, CALL START.
      WT=UI
      IF(XX.GT.0.0) GO TO 40
      DWT=0.0
      CALL START
   40 DO 50 J=1,4
      VALS(J)=VALSM1(J)
      IF(XX.EQ.0.0) RATEM1(J)=0.0
   50 RATES(J)=RATEM1(J)
C------INITIALIZE LAG PARAMS WITH EQUILIBRIUM VALUES.
      IF(XX.GT.0.0)GO TO 60
      CALL TAUMAX(DUMMY)
      XO=X
      TAUMO=TAUMEQ
      TAUM=TAUMEQ
   60 WRITE(6,900)
      CFD2=(UT/UI)**2
C------PRINT INITIAL VALUES.
      GO TO 105
C
C------BEGIN MAIN LOOP ****************
  100 NLOOP=NLOOP+1
C------STORE CURRENT VALUES.
      DO 110 J=1,4
      VALSM1(J)=VALS(J)
  110 RATEM1(J)=RATES(J)
      XX=X
C------STORE THE LAG PAREMETERS.
      XO=X
      TAUMO=TAUM
      IF((UB-UI)*UT.LE.0.0)GO TO 103
C       CHANGE THE SIGN OF UT TO REMOVE THE DOUBLE-VALUEDNESS OF UT.
      UT=-UT
      VALS(3)=UT
      WT=UI
      CALL BLVALU
  103 CONTINUE
C------PRINT CURRENT VALUES.
      IF(MOD(NLOOP,IPR).NE.0)GO TO 150
  105 JTBL=JTBL+1
      HSEP=1.+1./(1.-DELST/DELT)
      DQDX=10.0*TAUM/UI**2
      CP=1.0-(UI/UIREF)**2
      WRITE(6,910)XX,DELST,H,HSEP,CP,DELT,UB,UT,UI,CFD2,DQDX
      STBLS(JTBL)=XX
      DSTARS(JTBL)=DELST
      UI1DS(JTBL)=UI
      DELTS(JTBL)=DELT
      IF(IEXIT.EQ.1)GO TO 260
  150 CALL ADAMS(DX,4,DERSIL,IRUNGE)
      IF(X.LT.XMAX) GO TO 100
C
C------EXIT VALUE CALCULATIONS.
C       OBTAIN VALUES AT X=XMAX BY EXTRAPOLATION.
      IEXIT=1
      XX=XMAX
      DO 160 J=1,3
```

```
      JJ=JTBL-3+J
      XP(J)=STBLS(JJ)
      YP(J)=DSTARS(JJ)
      UP(J)=DELTS(JJ)
  160 ZP(J)=UI1DS(JJ)
      DELST=YINT(XP,YP,XMAX)
      UI=YINT(XP,ZP,XMAX)
      DELT=YINT(XP,UP,XMAX)
C------IF LAST TWO VALUES OF X ARE VEPY CLOSE TOGETHER, THEN CAN HAVE
C       PROBLEMS WITH SPLINE-ELIMINATE LAST BUT ONE POINT.
      IF(XMAX-STBLS(JTBL-1).LT.0.005*(STBLS(JTBL-1)-STBLS(JTBL-2)))
     $JTBL=JTBL-1
      JTBLS=JTBL+1
      GO TO 105
  260 CONTINUE
  900 FORMAT(1H0,'  X            DSTAR           H     HSEP          CP    DE
     $LT        UB          UT          UI          CF/2         DQDX/UI'/)
  910 FORMAT(2F10.5,4X,2F7.3,2X,F6.3,4F12.5,F12.6,F12.5)
  940 FORMAT(1H ,' REFERENCE QUANTITIES,VELOCITY=',F10.5,'VOLUME FLOWRAT
     $E=',F10.5,'CP MULTIPLIER=',F10.5)
      RETURN
      END
/*
```

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER AFOSR-TR- 77 - 1278	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) PREDICTION OF TRANSITORY STALL IN TWO-DIMENSIONAL DIFFUSERS		5. TYPE OF REPORT & PERIOD COVERED INTERIM
		6. PERFORMING ORG. REPORT NUMBER Report MD-36
7. AUTHOR(s) S GHOSE S J KLINE		8. CONTRACT OR GRANT NUMBER(s) F44620-74-C-0016
9. PERFORMING ORGANIZATION NAME AND ADDRESS STANFORD UNIVERSITY DEPARTMENT OF MECHANICSAL ENGINEERING STANFORD CALIFORNIA 94305		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS 2307A4 61102F
11. CONTROLLING OFFICE NAME AND ADDRESS AIR FORCE OFFICE OF SCIENTIFIC RESEARCH/NA BLDG 410 BOLLING AIR FORCE BASE, D C 20332		12. REPORT DATE Dec 76
		13. NUMBER OF PAGES 184
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) UNCLASSIFIED
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

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

18. SUPPLEMENTARY NOTES

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

FLUID MECHANICS
FLOW SEPARATION
DIFFUSERS
TRANSITORY STALL

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

A computer program has been developed that predicts the performance of diffusers operating in the transitory stall region, including the region of optimum recovery at fixed length. Results agree with data to the uncertainty in existing data. The important changes in method that underlie this advance are two: (i) simultaneous calculation of the outer flow and the separating layer which eliminates singular behavior in the zone of flow detachment; (ii) an improved boundary layer method that adequately approximates backflows near the wall and also represents both attached and free shear layers with good accuracy;

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

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered) the method employs Bradshaw's entrainment-shear miximum correlation, which is shown to hold over an extended range of flow conditions. Comparisons of the boundary layer scheme with the data of the 1968 Conference on Computation of Turbulent Boundary Layers and with the separating flow data of Strickland and Simpson both show good agreement in H, δ^*, $C_f/2$.

H, delta (*), C sub f/2