

LOUGHBOROUGH
UNIVERSITY OF TECHNOLOGY
LIBRARY

AUTHOR/FILING TITLE

PAPADAKIS, M

ACCESSION/COPY NO.

106548/02

VOL. NO.

CLASS MARK

~~30.~~

LOAN COPY

DATE DUE:

-7 MAY 1999

010 6548 02



COMPUTER ANALYSIS OF WING DESIGN

FOR GENERAL AVIATION AIRCRAFT

BY

MICHAEL PAPADAKIS

A Thesis

submitted in partial fulfilment of
the requirements for the award of
the degree of Master of Science
of the Loughborough University of Technology
January, 1981.

Supervisors: F.G. MACCABEE, M.Sc., D.C.Ae., C. Eng., A.F.R.Ae.S.
Department of Transport Technology
K.S. PEAT, B.Sc., Ph.D.
Department of Engineering Mathematics

Loughborough University of Technology Library	
Date	May 81
Class	
Accession No.	106548/02

SUMMARY

The calculation of the two dimensional viscous incompressible flow about single and multielement aerofoil sections is considered.

A panel method, based on vorticity and source distributions is used for the calculation of the potential flow. Once the velocity distribution is known, integral boundary layer methods are employed to predict the viscous effects. A wake model has also been developed for the calculation of the wake behind the aerofoil system.

The solution is iterative. At the end of each iteration the velocities on the aerofoil are corrected for viscosity and wake effects; the wake position is also relaxed, before the next iteration starts.

The mathematical model of the flow, together with the computer program written to test the model are described here in detail. The numerical results obtained using the computer program are found to be in good agreement with both experimental data and exact solutions.

ACKNOWLEDGEMENTS

The author wishes to acknowledge the co-operation and assistance he has received from many quarters; in particular he is indebted to his supervisors, Dr. Peat and Mr. Maccabee for their help and advice during the course of this research.

The author also wishes to thank Mr. Tony Cross and Mr. David Butter, of British Aerospace, for their technical advice and Dr. Shanehchi who helped with the computer program.

Finally for the work of typing thanks are due to Miss Judy Briers.

CONTENTS

	<u>PAGE</u>
SUMMARY	1
ACKNOWLEDGEMENTS	2
CONTENTS	3
LIST OF FIGURES	8
LIST OF SYMBOLS	11
INTRODUCTION	15
<u>Section 1:</u> MODELLING OF THE AIR FLOW - A GENERAL DISCUSSION	
1.1 Introduction	19
1.2 Models to Describe the Air and Some of its Properties	19
1.3 Potential Flow	23
1.4 Methods for Solving for the Flow Around an Aerofoil	26
1.4.1 Conformal Mapping	26
1.4.2 Singularities	26
<u>Section 2:</u> THE HISTORY AND DEVELOPMENT OF TWO-DIMENSIONAL POTENTIAL FLOW METHODS	
2.1 Introduction	28
2.2 The Problem of Calculating Wing Lift	28
2.3 The Early Stages	29
2.4 Development of the First Methods	30
2.5 Modern Methods	33
2.6 Concluding Remarks	38
<u>Section 3:</u> SURFACE SINGULARITY MODEL FOR THE CALCULATION OF THE POTENTIAL FLOW ABOUT THICK AEROFOIL SECTIONS	
3.1 Introduction	41
3.2 Linear Vorticity Panel Method	42
3.2.1 Model Description	42
3.2.2 Equations	42
3.3 Symmetrical Linear Vortex and Constant Source Surface Singularity Model	
3.3.1 Basic Considerations	43
3.3.2 Theoretical Basis	45
3.3.3 Numerical Solution Procedure	46
3.3.4 Velocity Equations	46

	<u>PAGE</u>
<u>Section 4:</u> VISCIOUS FLOW - A GENERAL DISCUSSION	
4.1 Introduction	50
4.2 Genesis of the Boundary Layer Concept	50
4.3 Viscous Flow Model	52
4.4 Viscous Flow Representation	54
4.5 Aerofoil Surface Velocity Equations	56
<u>Section 5:</u> LAMINAR BOUNDARY LAYER	
5.1 Introduction	59
5.2 Thwaites Method	59
5.3 Curle's Method	61
5.3.1 Equations	62
5.3.2 Calculation Procedure	62
5.3.3 Laminar Separation	63
<u>Section 6:</u> BOUNDARY LAYER TRANSITION	
6.1 Introduction	67
6.2 Crabtree Criterion	67
6.3 Michel's Method	68
6.4 Granville's Method	68
6.4.1 Instability Prediction	69
6.4.2 Transition Prediction	69
<u>Section 7:</u> SHORT BUBBLE ANALYSIS	
7.1 Introduction	72
7.2 Short Bubble Model of Shear Layer	73
7.2.1 Laminar Part of Shear Layer	74
7.2.2 Turbulent Part of Shear Layer	75
7.2.3 Bubble Length at Bursting	76
7.3 Method of Solution	77
<u>Section 8:</u> TURBULENT BOUNDARY LAYER	
8.1 Introduction	80
8.2 Head's Entrainment Method	80
8.2.1 Equations	81
8.2.2 Method of Calculation	81
8.3 The Method of Head and Patel	82
8.3.1 Equations	83
8.3.2 Method of Calculation	83
8.4 Initial Values	84
8.5 Turbulent Boundary Layer Separation	
8.5.1 Head's Criterion	85
8.5.2 Epplers Method	85
8.6 Concluding Remarks	86

Section 9: WAKE ANALYSIS

9.1	Introduction	92
9.2	Potential Flow Wake Model	92
9.3	Wake Singularity Sheet	93
9.4	Velocity Equations	94
9.5	Wake Viscous Flow Analysis	95
9.5.1	Equations	95
9.5.2	Initial Conditions	97
9.5.3	Calculation Procedure	97
9.6	Calculation of Wake Singularities	97
9.7	Wake Relaxation	98
9.8	End of Potential Core	100
9.9	Aerofoil Velocities	101

Section 10: CONFLUENT BOUNDARY LAYER

10.1	Introduction	107
10.2	General Considerations	107
10.2.1	Main Assumptions	108
10.3	Model Description	108
10.4	Equations for the Calculation of Main Region 1 and 2	110
10.4.1	Coles Velocity Profile	110
10.4.2	Main Region 1	111
10.4.3	Main Region 2	115
10.5	Initial Conditions	117
10.5.1	Main Region 1 is Preceded by a Core Region	117
10.5.2	Main Region 2 Not Preceded by Core	118
10.5.3	Main Region 2	119
10.6	Calculation Procedure	119
10.7	Confluent Boundary Layer Parameters	
10.7.1	Thickness δ_4	119
10.7.2	Displacement Thickness δ^*	120
10.7.3	Momentum Thickness θ	120
10.7.4	Skin Friction	120
10.7.5	Separation	121
10.8	Further Comments	121

Section 11: SEPARATED FLOW MODEL

11.1	Introduction	125
11.2	Flow Separation	125
11.3	Separation Flow Model	127
11.3.1	Basic Considerations and Assumptions	127
11.3.2	Approximations for the Free Shear Layer	129
11.3.3	Theoretical Basis	130
11.3.4	Method of Solution	130
11.4	Concluding Remarks	132

	<u>PAGE</u>
<u>Section 12:</u> OVERALL CALCULATION PROCEDURE	
12.1 Introductory Comment	136
12.1.1 First Iteration	136
12.1.2 All Other Iterations	137
12.1.3 Final Iteration	137
12.2 Aerodynamic Coefficients	138
12.2.1 Lift and Moment Coefficients	138
12.2.2 Drag Coefficient	141
 <u>Section 13:</u> METHOD VERIFICATION	 142
 <u>Section 14:</u> CONCLUSIONS AND FURTHER WORK	
14.1 Conclusions	167
14.2 Further Work	
 REFERENCES	 170
 <u>Appendix 1:</u> THE COMPUTER PROGRAM	 185
 <u>Appendix 2:</u> RUNNING THE PROGRAM	
A2.1 Program Input	239
A2.1.1 Geometry Definition	239
A2.1.2 Other Geometry Inputs	240
A2.1.3 Input of Flow Parameters	241
A2.2 Other Instructions	
A2.2.1 Potential Flow Analysis	241
A2.2.2 Viscous Flow Analysis	241
A2.2.3 Wake Analysis	242
A2.3 Further Comments	243
 <u>Appendix 3:</u> COMPUTER PROGRAM SUBROUTINES	 251
 <u>Appendix 4:</u> DATA SMOOTHING - MATRIX INVERSION	
A4.1 Data Smoothing	254
A4.2 Matrix Inversion	255
 <u>Appendix 5:</u> METHODS FOR SOLVING A SET OF LINEAR EQUATIONS	
A5.1 Successive Orthogonalisation Method	268
A5.2 Gauss Elimination with Partial Pivoting	271

<u>Appendix 6:</u>	VELOCITY INDUCED BY PARABOLIC VORTICITY DISTRIBUTION ON LINEAR PANEL	
	A6.1 Equation of Induced Velocity	274
	A6.2 Evaluation of A_1, A_2, A_3	275
	A6.3 Limiting Values	277
	A6.3.1 Case 1: Limiting Value for T	277
	A6.3.2 Case 2: Limiting Value for L	278
<u>Appendix 7:</u>	FORMULATION OF THE INFLUENCE COEFFICIENTS	
	A7.1 Introduction	280
	A7.2 Source-Vortex Relation	280
	A7.2.1 Influence Coefficients Due to Linear Vorticity	281
	A7.2.2 Linear Source Influence Coefficients	283
	A7.2.3 Influence Coefficients Due to Uniform Source	284
<u>Appendix 8:</u>	LINEAR VORTICITY METHOD APPLIED TO THICK AEROFOIL SECTIONS	286
	A8.1	
<u>Appendix 9:</u>	LINEAR VORTICITY MODEL - A GENERAL DISCUSSION	
	A9.1 Introduction	298
	A9.2 Aerofoil Surface Representation	298
	A9.3 Aerofoil Trailing Edge	299
	A9.3.1 Sharp Trailing Edge	299
	A9.3.2 Blunt Trainilg Edge	300
	A9.4 Inaccurate Surface Corner Points	301
	A9.5 Thin Aerofoils	302
<u>Appendix 10:</u>	VISCOUS FLOW BOUNDARY CONDITIONS	
	A10.1 Normal Velocity Boundary Conditions	307
	A10.1.1 Surface Transpiration Method	308
	A10.1.2 Equivalent Sources	309
	A10.2 Tangential Velocity Boundary Condition	310
<u>Appendix 11:</u>	WAKE GEOMETRY	
	All.1 Wake of Main Aerofoil	315
	All.1.1 Calculation of X_N, Z_N	315
	All.1.2 Calculation of wake N coordinates	317
	All.2 Wake of Flap	318
	All.3 Single Element Aerofoil	318
<u>Appendix 12:</u>	COMPUTER PROGRAM TO CALCULATE THE CONFLUENT BOUNDARY LAYER	320

LIST OF FIGURES

- Figure 1.1 Mathematical models to describe the air
- Figure 2.1 Potential flow methods
- Figure 3.1 Aerofoil surface representation
- Figure 4.1 Viscous flow model decomposition
- Figure 4.2 Attached flow models
- Figure 6.1 Boundary layer Reynolds number at transition against pressure gradient parameter
- Figure 6.2 $R\delta_2$ at transition plotted against R_s
- Figure 6.3 Stability curve
- Figure 6.4 Transition curve
- Figure 7.1 Short and long bubble pressure distributions
- Figure 7.2 Short bubble model
- Figure 8.1 Equilibrium flow measured by Herring and Norbury
- Figure 8.2 Measurements by Schubauer and Spangenberg
- Figure 8.3 Measurements of Schubauer and Klebanoff
- Figure 8.4 Measurements by Ludwig and Tillmann
- Figure 9.1 Wake flow
- Figure 9.2 Wake singularity sheet model
- Figure 9.3 Wake relaxation
- Figure 9.4 Potential core parameters
- Figure 10.1 Slot flow and confluent boundary layer model
- Figure 10.2 Layers of the flow model in main region 1
- Figure 10.3 Layers of the flow in main region 2
- Figure 11.1 Initial position of wake singularity sheets

- Figure 11.2 Separated flow model
- Figure 11.3 Aerodynamic characteristics
- Figure 13.1 Comparison of calculated and exact pressure distribution on an aerofoil section with slotted flap deflected 30°
- Figure 13.2 GA(W)-1 aerofoil coordinates
- Figure 13.3 Comparison of calculated and experimental pressure distributions on a GA(W)-1 aerofoil section
- Figure 13.4 Geometry definition of a GA(W)-1 aerofoil section with 29% Fowler flap
- Figure 13.5 Comparison of calculated and experimental pressure distributions on a GA(W)-1 aerofoil with 29% Fowler flap for various incidences and flap deflections
- Figure 13.6 Experimental and calculated lift curves for a GA(W)-1 section
- Figure 13.7 Experimental and calculated lift curves for a NACA4412 aerofoil section
- Figure 13.8 Wake relaxation
- Figure 13.9 Boundary layer calculation
-
- Figure A1.1 Flow model decomposition
- Figure A1.2 Subroutine decomposition of computer program
- Figure A4.1 Cubic spline fit on momentum thickness results
- Figure A4.2 C_p versus x/c plot by fitting cubic splines to the experimental data
- Figure A4.3 Aerofoil surface plot using cubic splines
- Figure A8.1 Pressure distribution on three aerofoil sections, calculated using the linear vorticity model

- Figure A8.2 Pressure distribution on GA(W)-1 aerofoil section with 29%
Fowler flap from the linear vorticity model
- Figure A8.3 High lift aerofoil pressure distribution using the linear
vorticity model
- Figure A8.4 Pressure distribution on the FX-67-K-170/17 aerofoil section
using the linear vorticity model
- Figure A8.5 Comparison of inviscid lift curve with the experimental one
for a NACA4412 aerofoil section
- Figure A9.1 Comparison of cosine and curvature methods for distribution
of segment corner points
- Figure A9.2 Variation of CL with number of panels (Linear vorticity model)
- Figure A9.3 Effect of thick trailing edge on CL calculation when using
the linear vorticity method.
- Figure A9.4 Effect of incorrect aerofoil surface coordinates on pressure
distribution
- Figure A10.1
- Figure A10.2 Wake geometry

LIST OF SYMBOLS

- A_{JK} = Influence coefficient on induced velocity on control point J to singularity on segment K.
- C = Aerofoil chord
- C_d = Drag coefficient
- C_E = Entrainment coefficient
- C_F = Skin friction coefficient
- C_l = Lift coefficient
- C_m = Pitching moment coefficient
- C_p = Pressure coefficient
- C_τ = Shear stress coefficient
- D = Drag force
- $F(r_1)$ = Empirical function defined in Eq.(8.15)
- F_0, F_1 = Universal functions
- G = Empirical function of H (Eq. 8.13)
- G_N = Normal component of influence coefficient
- G_T = Tangential component of influence coefficient
- \underline{G}_S = Influence coefficient due to a source singularity
- \underline{G}_V = Influence coefficient due to a vortex singularity
- G_{AA}^N = Normal influence coefficient of induced velocity on aerofoil due to aerofoil singularities
- G_{AV}^N = Normal influence coefficient of induced velocity on aerofoil due to wake vortex singularities
- G_{AS}^N = Normal influence coefficient of induced velocity on aerofoil due to wake source singularities
- G_{WA}^N = Normal influence coefficient of induced velocity on wake due to aerofoil singularities

- G_{WV}^{GN} = Normal influence coefficient of induced velocity on wake due to wake vortices
 G_{WS}^{GN} = Normal influence coefficient of induced velocity on wake due to wake sources
 G_0, G_1 = Universal functions
 H = $\frac{\delta^*}{\theta}$ shape parameter
 \bar{H}, H_ϵ = Shape parameter based on energy thickness $(\frac{\epsilon}{\theta}, \frac{\delta^{**}}{\theta})$
 L = Lift force
 M = Aerodynamic pitching moment
 Q = Boundary layer entrainment
 R = Universal gas constant
 Re_x = Reynolds number based on length x
 U, U_e = Velocity at the edge of the boundary layer
 U' = dU/dx
 U'' = d^2U/dx^2
 ∞, V_∞ = Free stream velocity
 V_{BL} = Velocity component on aerofoil surface due to boundary layer
 V_I = Inviscid velocity on aerofoil surface
 V_N = Normal velocity component
 V_T = Tangential velocity component
 b_j = Thickness of jet layer
 b_w = Thickness of wake layer
 $\underline{i}, \underline{j}, \underline{k}$ = Unit vectors along the x, y and z axes respectively
 ℓ = $\frac{\theta}{U} (\frac{\partial u}{\partial y})_{y=0}$, parameter defined in Eq. (5.6)
 ℓ_1 = Short bubble laminar length
 ℓ_2 = Short bubble turbulent length
 $\frac{\ell_1}{s}$ = ℓ_1 / θ_s
 $\frac{\ell_2}{s}$ = ℓ_2 / θ_s

- $m = -\frac{\theta^2}{v} \frac{dU}{dx}$, pressure parameter
 \underline{n} = normal unit vector
 \underline{r} = Position vector
 s = Length along aerofoil surface
 \underline{t} = Tangential unit vector
 t_{TE} = Trailing edge thickness
 u, v, w = Perturbation velocities
 x, y, z = Cartesian coordinates
 Δx = Increment of x
 $\nabla = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2}$
 α = Incidence
 γ = Vortex strength
 γ_A = Aerofoil singularities
 γ_S = Source strength (wake)
 γ_V = Vortex strength (wake)
 $\gamma(s)$ = Circulation per unit length
 $\underline{\gamma}$ = Vorticity vector
 δ = Boundary layer thickness
 δ^* = Displacement thickness $\int_0^{\infty} (1 - \frac{u}{U}) dy$
 δ^{**} = Energy thickness
 ϵ = Energy thickness, wake segment angle with aerofoil chord
 θ = Momentum thickness $\int_0^{\infty} \frac{u}{U} (1 - \frac{u}{U}) dy$
 $\lambda = \frac{dU}{dx} \frac{\theta^2}{v}$
 $\mu = \lambda^2 U U'' / (U')^2$
 ν = Kinematic viscosity of air
 π = Function of H
 ρ = Air density
 σ = Source strength

τ = shear stress
 τ_w = Wall shear stress
 ϕ = Potential flow function
 ψ = Stream function

Subscripts

EQ = Equilibrium values
inst = Instability
inv = Inviscid
j = Jet layer
l = Lower
m = Mean value
R = Indicates conditions at reattachment
S = Indicates conditions at separation
tran = Transition
u = Upper
w = Wake, wake layer
 x_0 = Conditions at slot exit
 δ = Conditions at the edge of the boundary layer

INTRODUCTION

During recent years the field of computational fluid dynamics has developed sufficiently to initiate some changes in traditional methods of aerodynamic design. Both computer power and the efficiency of numerical algorithms are improving with time, while the energy cost for driving large wind tunnels is becoming progressively higher. Partly for these reasons it has been advocated that the impact of computational aerodynamics on future methods of aircraft design will be profound (14).

There are three compelling reasons for developing computational aerodynamics. One is to investigate important new technological capabilities that cannot easily be investigated experimentally. A second concerns energy conservation. Large wind tunnels require large amounts of energy; whereas computers require comparatively negligible amounts. In coming years, energy considerations are likely to impose significant restrictions on testing time in such tunnels (14). The development of computational aerodynamics and advanced computers can be expected to lessen considerably the impact of such restrictions. Finally the third reason for developing computational aerodynamics relates to economics, since computer speed has increased at a much greater rate than computer cost.

The net cost of conducting a given numerical simulation with a fixed algorithm has decreased rapidly (Diagram 1). Equally remarkable has been the improvement in the computational efficiency of numerical algorithms for a given computer. This is illustrated in Diagram 2, which compares the trend in computational cost attributable to computer improvements alone, with the corresponding trend attributable to algorithm improvements alone.

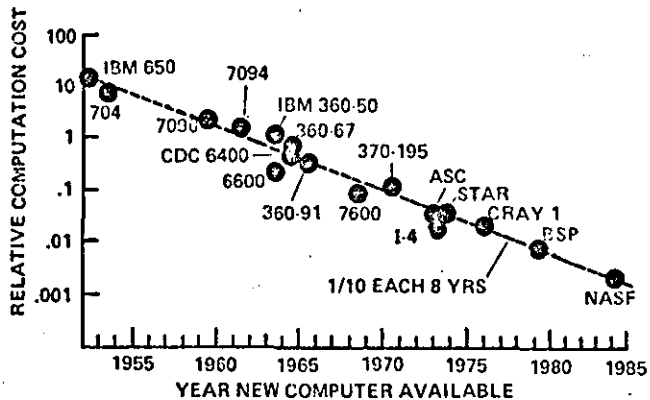


DIAGRAM 1: Trend of relative computation cost for numerical flow simulation on large computers; given flow and algorithm (14)

The two trends have compounded to bring about an altogether extraordinary cost reduction in computational aerodynamics. The cost of experiments, by contrast, has been increasing over the same period.

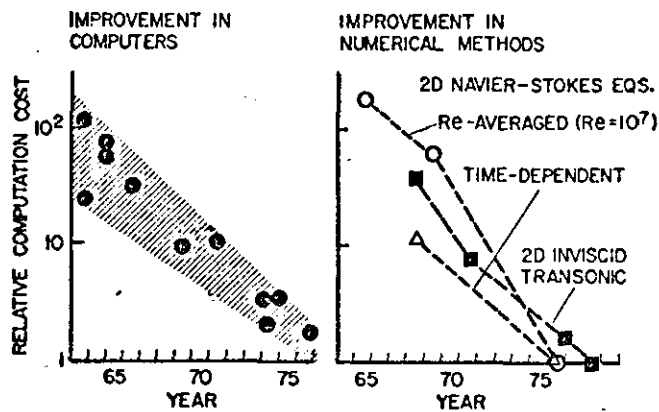


DIAGRAM 2: Cost effectiveness improvements in computer hardware and in numerical methods

Historical progress in computational aerodynamics can be characterised by a series of steps, each representing a more refined approximation to the full Navier-Stokes equations. Four major stages of approximation stand out. In order of their evolution and complexity they are:

1. linearized inviscid
2. nonlinear inviscid

3. Reynolds averaged Navier-Stokes

4. full Navier-Stokes

The present work is concerned with the first stage. Numerical computation methods using this stage of approximation are termed 'panel methods' because complex aircraft geometries are modelled by a large number of contiguous surface panels. Solution of the full Navier-Stokes equations requires conservation of mass, momentum and energy which contain altogether 60 partial derivative terms when written out in three Cartesian coordinates; whereas the linearised inviscid approximation truncates this to the well known Laplace equation containing only three terms.

The application of panel methods together with boundary layer theory to aerofoil design and analysis was accomplished many years ago. Since then a number of methods have been, and are being, developed (mainly in Europe and the U.S.A.), which deal with the two-dimensional viscous flow about single and multielement aerofoils. In Britain, however, most of the work has been concerned with high subsonic and transonic flow and very little has been done in the low subsonic area. Thus the design of a method for the analysis of general aviation aerofoils appeared necessary in order to fill the gap between existing methods.

It is the aim of this work to develop a complete mathematical model for the analysis of attached incompressible viscous flow about single and two element aerofoils. A number of available potential flow and boundary layer methods, some of which are of recent development, have been combined to create the flow model. The model consists of:

1. a potential flow analysis
2. a number of integral boundary layer methods
3. a wake flow analysis.

For the sake of completeness separated flow regions have also been considered and some of them, such as short bubble separation, are included in the computer program developed to test the flow model. A model to analyse trailing edge separation is also included in the present work although it has not yet been tested.

In the following sections a full description of the flow model is given. Section 1 considers mathematical models to describe the airflow. A literature survey of up-to-date potential flow methods is carried out in Section 2. Section 3 starts with a brief description of the linear vorticity model; this is followed with a detailed description of the new potential flow model. Viscosity and methods to model and account for it are discussed in Section 4. The boundary layer methods used in the present model are described in Sections 5 to 8. The wake analysis is considered in Section 9. With two element aerofoil sections, interaction between the wake flow of the upstream aerofoil and the boundary layer of the flap can take place in certain cases; a confluent boundary analysis is then necessary. This is described in Section 10. Trailing edge separation is considered in Section 11. The overall calculation procedure is discussed in section 12. Section 13 presents some results obtained from a number of runs of the computer program. The results are found to be in good agreement with both theory and experiment. Finally in Section 14 suggestions for further work are given.

MODELLING OF THE AIR FLOW - A GENERAL DISCUSSION

1.1 INTRODUCTION

The basic problem when dealing with aerofoil analysis is to calculate the aerodynamic forces and moments and the pressure distributions which result from the motion of the aerofoil through the air.

A general mathematical model of the flow about aerofoils of arbitrary geometry is thus required. Any such model must have built into it all the individual characteristic features which distinguish the particular flow from the others.

The basic theory and assumptions that lead to the modelling of the real flow about two dimensional aerofoil sections is briefly described in this section.

The aim is to list the important equations and the assumptions that lead to simpler flow models. No attempt has been made to derive any of the equations. These matters may be found in textbooks. (e.g. Refs. 100,29,54).

1.2 MODELS TO DESCRIBE THE AIR AND SOME OF ITS PROPERTIES

We are concerned with air and hence strictly, with the motion of air molecules. Thus we should start with the kinetic theory of gases as developed by Boltzmann and Maxwell which itself already represents a highly-ingenious model of whatever may happen in reality. The main assumptions and the derivation of the basic equations are fully described in (ref. 121, p.144).

It is possible to derive first Boltzmann's equation (ref. 65) which

describes the gas in terms of the motion of its constituent particles, and then from this to derive the Navier-Stokes equations, the continuity equation and the energy equation for a fluid.

In deriving the Navier-Stokes equations in this way a number of additional assumptions are implied: the gas must not be too dense but, on the other hand, there must be a sufficient number of collisions to preserve macroscopic equilibrium (Ref. 65). It is fortunate, however, that despite these assumptions the Navier-Stokes equations, as it happens, give an extremely close approximation to the behaviour of a gas over a much wider range of conditions than are to be expected from the analytical derivations.

It should be noted that the set of equations is not closed in that there are more unknowns than there are equations. The unknown properties are density, pressure, temperature, and the three components of velocity. The extra equations required is the equation of state, which may also be deduced from kinetic theory.

Thus the full set of equations is

$$\left. \begin{aligned} \rho \frac{Du}{Dt} &= X - \frac{\partial p}{\partial x} + \frac{\partial}{\partial x} \left[\mu \left(2 \frac{\partial u}{\partial x} - \frac{2}{3} \operatorname{div} u \right) \right] + \frac{\partial}{\partial y} \left[\mu \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) \right] + \frac{\partial}{\partial z} \left[\mu \left(\frac{\partial w}{\partial x} + \frac{\partial u}{\partial z} \right) \right] \\ \rho \frac{Dv}{Dt} &= Y - \frac{\partial p}{\partial y} + \frac{\partial}{\partial y} \left[\mu \left(2 \frac{\partial v}{\partial y} - \frac{2}{3} \operatorname{div} u \right) \right] + \frac{\partial}{\partial z} \left[\mu \left(\frac{\partial v}{\partial z} + \frac{\partial w}{\partial y} \right) \right] + \frac{\partial}{\partial x} \left[\mu \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) \right] \\ \rho \frac{Dw}{Dt} &= Z - \frac{\partial p}{\partial z} + \frac{\partial}{\partial z} \left[\mu \left(2 \frac{\partial w}{\partial z} - \frac{2}{3} \operatorname{div} u \right) \right] + \frac{\partial}{\partial x} \left[\mu \left(\frac{\partial w}{\partial x} + \frac{\partial u}{\partial z} \right) \right] + \frac{\partial}{\partial y} \left[\mu \left(\frac{\partial v}{\partial z} + \frac{\partial w}{\partial y} \right) \right] \end{aligned} \right\} \quad (1.1)$$

$$\frac{\partial \rho}{\partial t} + \frac{\partial(\rho u)}{\partial x} + \frac{\partial(\rho v)}{\partial y} + \frac{\partial(\rho w)}{\partial z} = 0$$

$$p - \rho R T = 0,$$

Energy equation

(1.2)

The possibility of solving the above system of equations for the case of arbitrary geometry is still quite remote despite the advances that have been made in computer technology. Thus the above equations have to be simplified even further.

By assuming that the flow is isothermal and incompressible the equation of state and the energy equation become superfluous as far as the calculation of the flow field is concerned. The flow field can now be considered independently from the equations of thermodynamics.

The new set of equations is therefore,

$$\begin{aligned}
 \rho \left(\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} + w \frac{\partial u}{\partial z} \right) &= X - \frac{\partial p}{\partial x} + \mu \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2} \right) \\
 \rho \left(\frac{\partial v}{\partial t} + u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y} + w \frac{\partial v}{\partial z} \right) &= Y - \frac{\partial p}{\partial y} + \mu \left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} + \frac{\partial^2 v}{\partial z^2} \right) \\
 \rho \left(\frac{\partial w}{\partial t} + u \frac{\partial w}{\partial x} + v \frac{\partial w}{\partial y} + w \frac{\partial w}{\partial z} \right) &= Z - \frac{\partial p}{\partial z} + \mu \left(\frac{\partial^2 w}{\partial x^2} + \frac{\partial^2 w}{\partial y^2} + \frac{\partial^2 w}{\partial z^2} \right) \\
 \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} &= 0.
 \end{aligned} \tag{1.3}$$

At this point it should be mentioned that in writing the equations of motion the Eulerian instead of the Lagrangian approach has been used.

The Lagrangian description of the flow field provides a history of the fluid as in the case of the motion of a particle in ordinary dynamics.

When using the Eulerian method only the fields of the various physical quantities are considered (107). There is no need to trace directly the paths of individual particles of the fluid. Instead the conditions at a general point in the fluid are observed. For example, if \underline{v} is the velocity field of the fluid then \underline{v} is a function of the space variables x, y, z and also of the time if the flow is unsteady. This field describes the velocity of a particle of fluid which is located at (x, y, z) at time t ; thus the actual path of any individual particle is not exhibited directly in this description.

Another concept used to simplify matters is that of the air particle which may be thought of as a 'fluid element' or 'body of fluid'. Bulk properties are actually thought of as interactions between such particles. The concept of fluid particles is useful in that it allows the physics of

fluid flows to be distinguished from that of solid bodies and of plasticity. Fluid particles can easily be moved relative to one another; there is no special initial arrangement of the fluid particles; and small forces are sufficient and little work needs to be done, to bring about a different arrangement of the particles and let them flow if the changes are slow enough. But this is also the reason why it is so difficult to describe and to understand fluid motions.

Going back to the equations (1.3) further simplifications are required before useful solutions can be obtained.

One drastic but nevertheless often useful simplification is to ignore the viscosity of the air altogether and, moreover, to assume the flow to be irrotational. In these potential flows, only the condition of zero normal velocity can be fulfilled and tangential slip must be allowed to occur along a solid wall. An even more useful assumption for high Reynolds' number flow is that of Prandtl's (88) boundary layer. According to this all viscous effects are confined to a thin layer along the surface, of the body. Outside the boundary layer, the flow is taken to be inviscid. Within the boundary layer it is assumed that the velocity along the body is much less than that normal to it, and that streamwise variations are much smaller than cross-stream variations. In that flow model, the condition of zero tangential velocity can be fulfilled and account must be taken of the fact that the slowed-down flow near the surface takes up more room and displaces the streamlines in the external flow outwards, compared with where they would have been had there been no boundary layer. The existence of such a displacement thickness means that the flow outside the boundary layer and hence the pressure along the surface of a given body, is the same as the irrotational flow about a hypothetical body with zero normal velocity which lies wholly outside the given body (Ref. 65).

Thus even the boundary conditions to be applied depend on the simplified model of the flow we choose to adopt. In this flow model of Prandtl, work must be done by the body on the boundary layer, as it moves through the air, and momentum is exchanged. Also, the boundary layer air is left behind the body in the form of a wake and the reduced momentum in the wake corresponds to a drag force on the body.

1.3 POTENTIAL FLOW

Figure 1.1 shows the analytical steps that lead to potential flow theory. For a steady, inviscid, incompressible flow, Euler's equations of fluid motion reduce to two relatively simple relationships that govern the velocity vector (\underline{v})

$$\text{div } \underline{v} = \underline{\nabla} \cdot \underline{v} = 0 \quad (1.4a)$$

$$\text{curl } \underline{v} = \underline{\nabla} \times \underline{v} = 0 \quad (1.4b)$$

The first equation satisfies conservation of mass; the second one assures that the dynamics of the flow is treated correctly (123).

In addition to satisfying equation (1.4) one must assure that any mathematical description of the flow field around a given body shape satisfies the boundary condition that there be no velocity normal to the body at all points on its surface. If \underline{n} is the unit vector normal to the surface, the following must hold

$$\underline{v} \cdot \underline{n} = 0 \quad (1.5)$$

To assist in the solution of equation (1.4) two functions are introduced. The first of these is known as the velocity potential, ϕ , and is defined such that

$$u = \frac{\partial \phi}{\partial x} \quad v = \frac{\partial \phi}{\partial y} \quad w = \frac{\partial \phi}{\partial z} \quad (1.6)$$

or generally,

$$\underline{v} = \underline{\nabla} \phi \quad (1.7)$$

Equation (1.7) satisfies identically equation (1.4b). However, in order to satisfy equation (1.4a), it follows that ϕ must be a harmonic function (1.23), that is

$$\nabla^2 \phi = 0 \quad (1.8)$$

The operator, ∇^2 , known as the Laplacian is defined as

$$\nabla^2 = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2}$$

A flow for which equation (1.4) is satisfied, and hence ϕ can be defined, is known as a potential flow. The resulting fluid motion is described as being irrotational. This follows since, in the limit at a point, the curl of the velocity vector, which is zero, is equal to twice the rotational or angular velocity.

The stream function, ψ , is related to the velocity components by

$$u = \frac{\partial \psi}{\partial y} \quad v = - \frac{\partial \psi}{\partial x} \quad (1.9)$$

ψ can only be defined for two-dimensional, or axisymmetric, flow. To obtain a particular component, the partial derivative of ψ is taken in the direction normal to the velocity and to the left as one looks in the direction of the velocity.

For an irrotational flow in order to satisfy equation (1.4b) the stream function ψ must also be harmonic

$$\nabla^2 \psi = 0 \quad (1.10)$$

The change in the potential function ϕ between two points A and B (Diagram 1.1) can be expressed in vector notation as

$$\phi(B) - \phi(A) = \int_A^B \underline{v} \cdot d\underline{R} \quad (1.11)$$

where \underline{R} is the radius vector to the curve along which the integration is being performed.

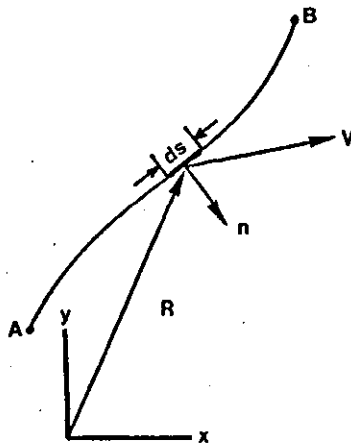


DIAGRAM 1.1: Two dimensional flow through a line element

If now ϕ_1 and ϕ_2 are functions satisfying equation (1.11), then, because this equation is linear, their sum will also satisfy (1.11). In general both the velocity potential and stream function can be constructed by summing less complicated functions,

$$\begin{aligned}\phi(x,y) &= \sum_{i=1}^n \phi_i(x,y) \\ \psi(x,y) &= \sum_{i=1}^n \psi_i(x,y)\end{aligned}\tag{1.12}$$

Equation (1.12) represents the real benefit to be gained in describing a flow in terms of ϕ and ψ .

When summing the flow functions, the velocities will add vectorially. This is obvious from equation (1.12) since

$$\text{grad } \phi = \text{grad } \phi_1 + \text{grad } \phi_2$$

or

$$\underline{V} = \underline{V}_1 + \underline{V}_2 + \dots$$

The simple flows from which more complicated patterns can be developed are referred to as elementary flow functions. There are three of them:

1. uniform rectilinear flow
2. vortex flow
3. source flow

Vortices and sources produce singularities in the potential flow field. An element may consist of singularities of constant strength, or the strength may be considered to vary in some simple manner along the element.

1.4 METHODS FOR SOLVING FOR THE FLOW ROUND AN AEROFOIL

Profile theory can be treated in two different ways: first by the method of conformal mapping and second, by the so called method of singularities.

For practical purposes the method of singularities is considerably simpler than conformal mapping. In general, the method of singularities produces only approximate solutions, whereas conformal mapping leads to exact solutions, although these often require considerable effort.

1.4.1 Conformal Mapping

This method is limited to two dimensional problems. The flow about a given body is obtained by using conformal mapping to transform it into a known flow about another body (usually a circular cylinder), (see Ref.101).

1.4.2 Singularities

In the method of singularities the body in the flow field is substituted by sources, sinks and vortices. Through superposition of their flow fields with a uniform flow a suitable body contour (profile) is produced. The flow field within the contour has no physical meaning. For the creation of a symmetric profile in a symmetric incident flow field, only sources and sinks are required, whereas for the creation of camber, vortices must be added to the profile.

The method of singularities can also be applied to three dimensional flows, such as wings of finite span and fuselages.

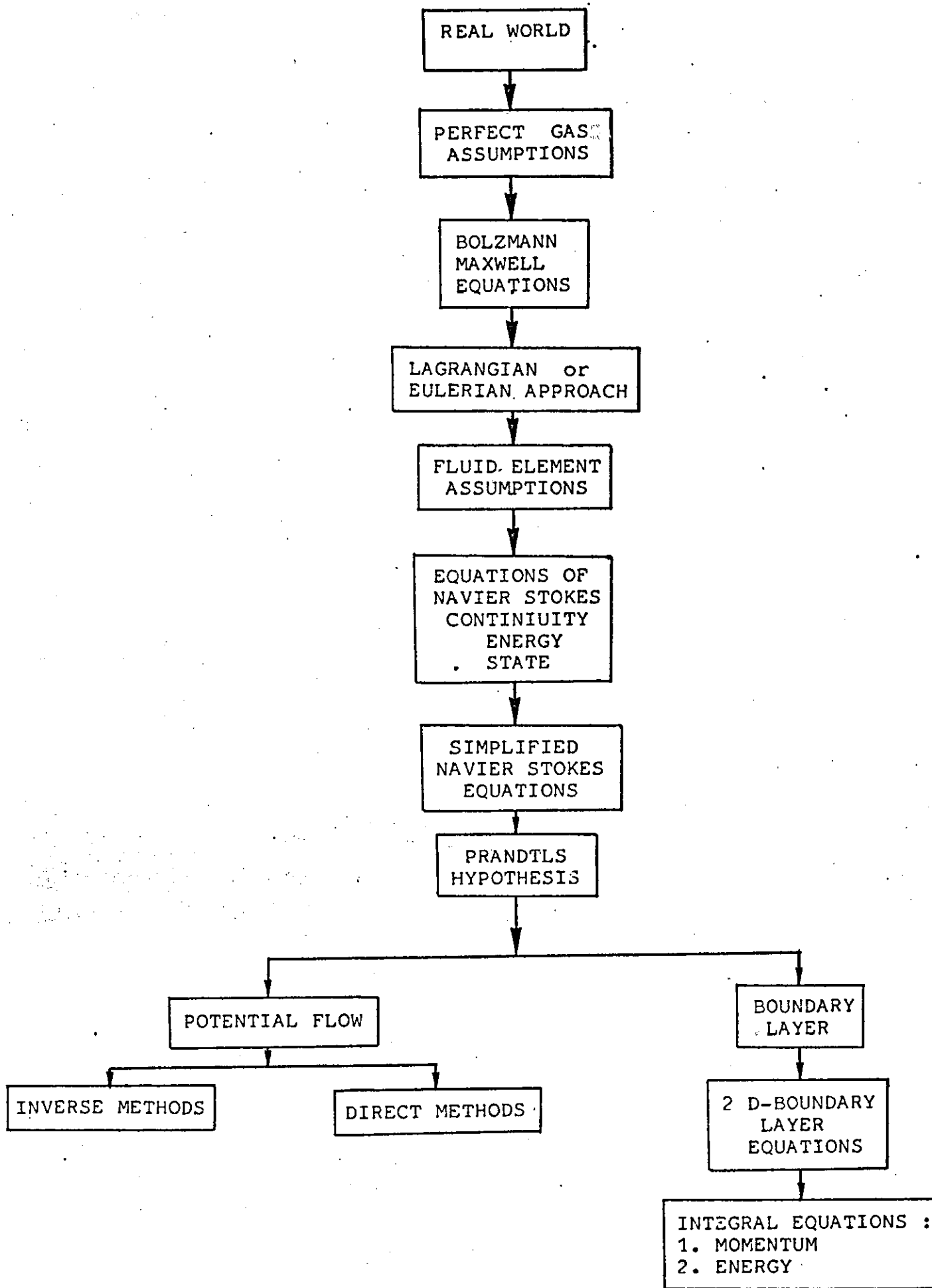


FIGURE 1.1 Mathematical Models to Describe the Air

THE HISTORY AND DEVELOPMENT OF TWO DIMENSIONAL

POTENTIAL FLOW METHODS

2.1 INTRODUCTION

In this section various methods for calculating the potential flow about single and multielement aerofoil sections are presented, (see Figure 2.1). The survey that follows has been based mainly on reference 105.

The development of the early methods has evolved the present day sophisticated analysis techniques and this progression is described here. This includes both conformal transformation and singularity methods.

2.2 THE PROBLEM OF CALCULATING WING LIFT

A systematic study of the problem of lift developed by a given wing would seem to begin with consideration of the lift developed by a slice or section out of the wing. Modelling the problem in this fashion has the advantage that one need consider only flow in two dimensions rather than in three, a greater mathematical simplification. Further it would seem reasonable to assume that the fluid is inviscid, if for no other reason than to take advantage of the extensive analytical studies (Refs. 125,129) that had been carried out for this case during the nineteenth century. These studies had been successful at explaining several experimental facts and present far less mathematical difficulty than one would encounter working the more general equations for the flow of a viscous fluid formulated by Navier and by Stokes about 1840.

The immensity of the problem facing engineers in 1900 trying to devise a rational means of calculating wing lift can be better appreciated when one realizes that in the contemporary view lift was the force reacting to the change in the momentum of the airstream striking the inclined lower surface of the wing. Such a force would be proportional to $\sin^2 \alpha$ where α is the angle by which the lower surface is inclined to the airflow. If one were to assume that a wing is flying at fifty miles an hour, with $\alpha=6^\circ$, then it could develop about 3.1N of lift per square meter of surface according to this theory. Since it was then impossible to build a wing lighter than this weight many scientists confidently predicted that man would never fly. More perceptive individuals noted however that the flight of gliders could not be explained by such small values of lift and therefore something must be wrong with the theory.

2.3 THE EARLY STAGES

Lord Rayleigh had shown in 1878 that the swerving flight of a 'cut' tennis ball could be explained at least in general terms by comparing it to the case of a cylinder placed in an inviscid uniform stream. By superposing a circulatory flow upon the cylinder, the cylinder developed a force normal to the direction of the uniform stream, directly proportional to the strength of the circulatory flow. This result along with the earlier work of Helmholtz and Kirchhoff was known to the German mathematician M.W. Kutta who was investigating the lift produced by cambered aerofoil at $\alpha=0$. In a paper of his published in 1902 he examined the theoretical flow round a thin aerofoil formed by a circular arc. He concluded that the flow over the upper surface was equal to that over the lower surface at the trailing edge. The flow would therefore leave the surface smoothly at finite

velocity. In order to obtain an approximate solution for the lift he was ready to accept the idea of an infinite velocity at the sharp leading edge, a situation studied by Helmholtz.

2.4 DEVELOPMENT OF THE FIRST METHODS

Joukowski, working independently along somewhat simpler lines, was able to obtain exact solutions for a certain class of aerofoils in inviscid flow. He first showed that when a cylindrical body of arbitrary cross-section moves with velocity, V , in a fluid whose density is ρ and there is a circulation of magnitude, Γ , around the body, a force is produced equal to the product $\rho V \Gamma$ per unit length of the cylinder. The direction of the force is normal both to the velocity, V , and the axis of the cylinder. Joukowski also assumed the flow to leave the aerofoil smoothly at the trailing edge. By means of this hypothesis the whole problem of lift becomes purely mathematical: one has only to determine the amount of circulation so that for zero vertex angle at the trailing edge the velocity of the flow leaving the upper surface is equal to the flow leaving the lower surface. If the tangents to the upper and lower surfaces form a finite angle, the trailing edge is a stagnation point.

Joukowski then found a transformation $\zeta = z + c^2/z$ by which a circle in the z -plane becomes an aerofoil in the ζ -plane. Since the transformation is conformal, the fluid velocity and pressure which exist at any point on the surface of the cylinder can be related quantitatively to those which exist at the corresponding point on the aerofoil. Integration of these pressures in the direction normal to the free stream velocity then gives the aerofoil lift (which is also the same as the lift produced by the generating cylinder).

The Joukowski transform technique was a great step forward in analyzing the lift of aerofoils. It gives the correct variation of lift with angle of attack and predicts lift values which are very close to measured values at the same angles of attack. However the Joukowski transform technique also has a number of disadvantages:

1. It is an inverse technique, that is one does not know beforehand precisely what the aerofoil will look like. As a result it is difficult to use the technique to estimate the characteristics of a given aerofoil.
2. It leads always to an aerofoil with a cusp, at the trailing edge. This is impractical structurally.
3. It generates aerofoils which have their minimum pressure point very far forward. Consequently, they have thick boundary layers, and therefore higher drag and lower maximum lift values than aerofoils with the minimum pressure point further back.
4. Being an inviscid theory, it cannot be used to estimate either lift characteristics near stall or drag values.
5. It is tedious to determine the ordinates of the aerofoil accurately.

These deficiencies were soon recognized and many investigators tried to devise more general transforms which could be used to represent a great variety of aerofoils, in particular those with finite trailing edge angles. Karman and Trefftz (Ref. 128), von Mises (Ref. 133), Muller (Ref. 134) and Theodorsen (Ref. 115) were among the leaders in this effort, which by 1932 had reached the point where one could determine the lift characteristics of a great variety of aerofoils. The great effort required to complete a calculation, however, discouraged thoughts of a further generalization in the transform technique.

Analyses of the lift characteristics of various Joukowski aerofoils in the meantime revealed that the aerofoil thickness contributed little to the lift. It therefore seemed to some that if aerofoils for which one had difficulty finding conformal transforms could be characterised by their mean camber lines only, then perhaps one could have a relatively simple, yet direct method of evaluating the lift and pressure distribution of arbitrary aerofoils. Such an approach is obviously most appropriate when the actual aerofoils are thin. These ideas were developed in the early 1920's by Munk (Ref. 81), Birnbaum (Ref. 123), and Glauert (Ref. 124).

Glauert replaced the aerofoil by its mean camber line which he assumed that it lies near the chord line. On this basis he made the approximation that the velocities over the aerofoil could be represented by a continuous distribution of vortices lying along the chord line. The variation in vorticity with chord location is not known initially. The velocity induced at point X_1 on the chord of the aerofoil due to the vortex sheet is given by

$$v(X_1) = \int_0^C \frac{\gamma dx}{2\pi(X-X_1)} \quad (2.1)$$

where γ is the vortex strength per unit length. This induced velocity is actually calculated for a point on the chord but according to Glauert's approximation may be taken to be the same as the induced velocity at the corresponding point of the aerofoil itself. Since the resultant of the free stream velocity and the induced velocity adjacent to the aerofoil must be parallel to the surface at each point of the aerofoil and since the flow angularities are small one may write this statement as

$$\alpha + \frac{v}{V} = \frac{dy}{dx} \quad (2.2)$$

where $\frac{dy}{dx}$ is the slope of the mean camber line at X_1 . It will be seen that these two equations are sufficient to provide a complete solution of the problem in terms of the shape of the curved line which represents the aerofoil. The solution is obtained as $\gamma(x)$. Then according to

Joukowski's theorem

$$\left. \begin{aligned} L &= \int_0^C \rho V \gamma dx \\ M &= \int_0^C \rho V \gamma x dx \end{aligned} \right\} \quad (2.3)$$

Values of C_L and C_{M_0} computed by this method were found to be in close agreement with experimental determinations of these quantities. Glauert's method can be seen to be considerably simpler to use than the transform technique. During the 1930's when designers tried to find ways of reducing wing drag by eliminating external bracing, they were forced by structural considerations to abandon the very thin aerofoils they had been using until that time. They found that in order to predict the lift and moment characteristics of the newer and thicker wing sections more elaborate analytical methods or extensive wind tunnel testing, were necessary. Some of the new analytical methods were based on the following. Since the sum of solutions to the Laplace equation which describes the inviscid incompressible flow, is also a solution, one can describe a thick cambered aerofoil at angle of attack by superimposing solutions for a curved line, a flat plate at angle of attack and a thick symmetrical aerofoil at $\alpha=0$. Karamcheti (Ref. 127) provides such a solution.

2.5 MODERN METHODS FOR THE CALCULATION OF THE POTENTIAL FLOW

It has only been within the last 20 years or so that interest in improved analytical methods has been taken up again. This is mainly due to:

1. Recent sharp increase in the cost of making models and carrying out test.
2. The desire to optimize certain aspects of aerofoil behaviour and to investigate the characteristics of unconventional aerofoils.

3. The appearance of large digital computers which made it possible to consider the use of what had previously been rather laborious methods on a routine basis.

Weber's method was one of the first and perhaps most widely used methods of the current revival. By transforming a symmetrical two-dimensional aerofoil at angle of attack into a slit, she was able to show that the source distribution which she used to represent the thickness at zero lift can be placed along the chord line rather than on the surface with little error, provided the aerofoil is no thicker than about 10% of the chord. Using this assumption and by superimposing a vortex distribution on a flat plate at angle of attack, Weber obtained the equation

$$V(x,z) = \frac{V_0}{\sqrt{1 + \left(\frac{dz}{dx}\right)^2}} \left\{ \cos \alpha \left[1 + \frac{1}{\pi} \int_0^1 \frac{dx'}{x-x'} \frac{dz}{dx} \right] \pm \sin \alpha \sqrt{\frac{1-x}{x}} \right. \\ \left. \left[1 + \frac{1}{\pi} \int_0^1 \left(\frac{dz}{dx} - \frac{2z(x')}{1-(1-2x')^2} \cdot \frac{dx'}{x-x'} \right) \right] \right\} \quad (2.4)$$

The positive sign holds for the upper surface, the negative sign for the lower surface. $V(x,z)$ is the velocity along the aerofoil surface. The pressure coefficients along the surface are given by

$$C_p = 1 - \left(\frac{V(x,z)}{V_\infty} \right)^2$$

Weber extended her approach to treat camber aerofoils (Ref. 139)

Comparisons between Weber's results and exact theory for Jourkowski aerofoils indicate that her method predicts pressures which are low by about 1%. Maximum camber must be less than about 4% of chord and thickness less than 10% of chord to obtain results of this accuracy.

The success of Weber's approach and its adaptability to computer solution seems to have served as a spur to the development of more exact aerofoil representation schemes which are practical only if carried out by

digital computer. The method of Hess and Smith (Ref. 126) is among the best known of these developments. In this method the non-lifting aerofoil surface is replaced by a source sheet with strength $\sigma(s)$ where s is the distance measured along the aerofoil surface. The sum of the velocity induced by the source sheet and the free stream velocity is forced to satisfy the condition that its component normal to the aerofoil surface at each value of s is zero. This statement is written mathematically as a Fredholm integral equation of the second kind

$$2\pi\sigma(s) + \oint \sigma(s') \ln r(s,s') ds' = F(s) \quad (2.5)$$

where $r(s,s')$ is the distance between the point of interest, s , and any other point on the surface, s' . The function $\sigma(s')$ represents the source strength at points other than s ; $\sigma(s)$ is the source strength at s ; and $F(s)$ represents the component of the free stream velocity normal to the surface at s . The left side of the equation represents the component of the velocity induced by the source sheet which is normal to the surface. For a given aerofoil geometry and freestream velocity the quantity to be found is $\sigma(s')$ which occurs under the integral sign.

In order to solve the integral equation Hess and Smith made the following approximations:

- a. The contour of the aerofoil can be modelled by N straight line segments.
- b. $\sigma(s')$ is constant over each segment.
- c. The integral is evaluated at only one point - generally the midpoint - of each segment.

By solving the system of the N simultaneous equations σ can be found on each segment and therefore the tangential component of velocity and the surface pressure.

Lifting aerofoils can be treated with this method by superimposing a vortex sheet of suitable strength so that the total flow satisfied the local tangency condition as well as the Kutta conditions at the trailing edge.

A different approach to the problem was taken by Martensen (Ref. 131). By requiring that the strength of the vortex sheet be identical to the velocity distribution on the surface of the aerofoil he was able to show that in the interior of a closed vortex sheet the velocity is everywhere zero. Thus on the inner side of the vortex sheet the net tangential velocity which is the sum of the free stream and that induced by the vortex sheet is zero. This can be written as

$$\frac{\gamma(s)}{2} - \frac{1}{2\pi} \frac{\partial}{\partial n} \oint \gamma(s') \ln r(s, s') ds' = V_{\infty} \left(\frac{dx}{ds} \cos \alpha + \frac{dy}{ds} \sin \alpha \right) \quad (2.6)$$

By discretising the above integral Martensen ended up solving a system of simultaneous equations. In order to obtain a closed solution of the problem an extra equation is required and is obtained from the Kutta condition. When dealing with very thin aerofoils Martensen's method gives rather poor results. This is because when the upper and lower surface control points are very close together the vortices located there induce strong tangential velocities on each other. While this induced velocity actually decays very rapidly for points in the neighbourhood of the control point, the method of approximating the integral which Martensen used assumes it to be constant.

By using a different limiting approximation Jacob improved the results however at the cost of restricting the way in which the control points can be distributed on the aerofoil surface.

Oeller formulated the problem in terms of the stream function as the dependent variable. The stream function ψ is governed by Laplace's equation

$$\nabla^2 \psi = 0$$

which is linear, so the solution of the flow field can be obtained by superposition. The aerofoil is represented by a distribution of vorticity along the surface of strength $\gamma(s)$. Adding the freestream function to the stream function of this vortex sheet results in the stream function of the whole flow field

$$\psi = V_{\infty} \cos \alpha z - V_{\infty} \sin \alpha x + \frac{1}{2\pi} \int \gamma(s') \ln r(s, s') ds' \quad (2.7)$$

r = the distance between a point on the aerofoil surface and a field point (x, z) .

The stream function of the vortex sheet is found by integration of the stream functions of elementary vortices from the lower surface trailing edge point ($s=0$) to the upper surface trailing edge point ($s=s_{TE}$). The value of the stream function ψ is constant along a streamline. Hence ψ is also constant along the aerofoil contour which is part of the stagnation streamline.

To solve the integral equation the aerofoil geometry and the vortex distribution are discretized as follows. The aerofoil surface is divided into N segments. The $(N+1)$ corner points of these segments are placed on the aerofoil's surface and are then connected by straight lines. The vorticity along each segment is constant. By choosing N control points the integral equation reduces to a set of linear algebraic equations,

$$\psi_c - \sum_{j=1}^N k_{ij}^v \gamma_j = V_{\infty} \cos \alpha z_j - V_{\infty} \sin \alpha x_j, \quad j=1, 2, \dots, N. \quad (2.8)$$

ψ_c = stream function at the contour of the aerofoil.

k_{ij}^v 's = aerodynamic influence coefficient.

Since the system of N linear equations has $N+1$ unknowns an extra equation is required and this is obtained from the Kutta condition.

Several improvements in the transform approach to predicting aerofoil characteristics have also appeared in recent years. Lighthill (Ref. 136) chose to specify the desired velocity distribution about the aerofoil in

closed form. Sato (Ref. 135) extended this approach to permit a velocity distribution of any kind to be specified. As evaluated by Sato, the velocity distribution is assumed in such a way that front and rear stagnation points can be treated separately. A well-behaved function $g(\theta)$ takes up the velocity distribution everywhere with the exception of the stagnation points and three constants which are imbedded. The constants are determined by $g(\theta)$, the fact that the aerofoil is a closed curve, and the fact that the flow field at infinity is uniform. A set of initial values must be given to the three constants in order to obtain $g(\theta)$ from the specified velocity distribution. This $g(\theta)$ is then used to obtain a new set of values for the constants which will give a closed curve as the aerofoil geometry. The process is repeated iteratively until the initial and final constant values match. In this way Sato's method always guarantees an aerofoil geometry giving the desired velocity distribution. Because of the repetitive nature of many of its steps and the need for piecewise integration, it is best done on a digital computer.

2.6 CONCLUDING REMARKS

The history and development of potential flow methods for the analysis of single and multielement aerofoils has been presented.

Most of the methods described are of the singularity type. There exist a large number of panel methods today, and it is not possible to mention all of them. However the reader is referred to the work of Argyris (3), Callaghan and Beaty (10), Eppler and Somers (34), Kennedy and Marsden (61), Mavriplis (73), Morgan (79), Morino and Kuo (80), Seeborn and Newman (103), Smith (106).

A number of currently used methods have been extensively tested by Freuler and Gregorek (40) and Sytsma (112). The results indicate quite a

variation in the performance of such methods, thus the user has to select the method that best fits his requirements.

A typical example of transonic flow analysis is described in reference (11), however, this kind of flow is beyond the scope of the present work.

A rather detailed introduction into the theory of the methods described here can be found in reference 17.

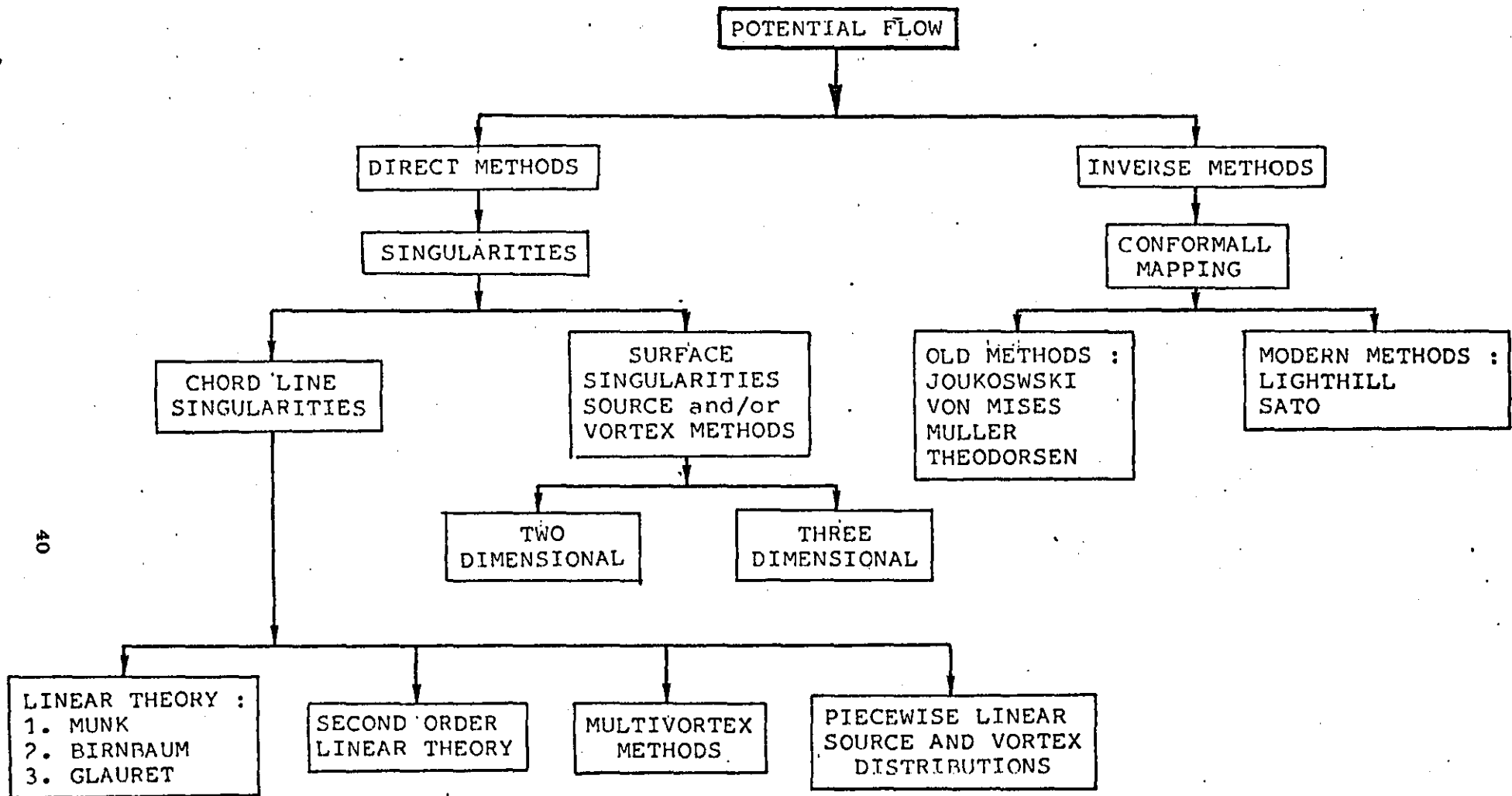


FIGURE 2.1 Potential Flow Methods

SURFACE SINGULARITY MODEL FOR THE CALCULATION OF THE POTENTIAL FLOW ABOUT THICK AEROFOIL SECTIONS

3.1 INTRODUCTION

The surface panel method philosophy for solving arbitrary incompressible potential flow problems involves the mating of classical potential theory with contemporary numerical techniques. Classical theory is used to reduce an arbitrary flow problem to a surface integral equation relating boundary conditions to an unknown singularity distribution (Ref. 66). The numerical techniques are then used to calculate an approximate solution to the integral equation.

All properly formulated surface panel methods are exact in the sense that the difference between the approximate numerical solution and the exact solution to the integral equation can be made arbitrarily small at the expense of increasing the number of computations, assuming that computer errors are not a problem. This does not imply that all panel methods are equally successful. Indeed, vast differences exist with respect to prediction accuracy versus computational effort, reliability, simplicity and applicability to a given analysis problem.

In this section a brief description of the surface singularity method used for the potential flow calculation, during the early stages of the research, is given. The inability, of the linear vorticity model to deal with wing sections having a blunt trailing edge, enforced the use of a rather more sophisticated surface singularity method. This method is outlined in some detail here.

3.2 LINEAR VORTICITY PANEL METHOD

Maskew (70) applied the linear vorticity model to the thick aerofoil section problem, and showed that a linear vorticity distribution combined with low order panels can predict the potential flow pressure distribution around multielement sections with good accuracy.

A basic limitation of the model, however, is that it can only deal with sections having a sharp trailing edge, thus it is not suitable for viscous flow analysis.

3.2.1 Model Description

In order to allow for both finite thickness and circulation, the aerofoil contour is approximated by a closed polygon (Diagram 3.1). A continuous distribution of vortices is then placed on each side of the polygon, with the vortex strength per unit length, $\gamma(s)$, varying linearly from one corner to the next and continuous along the corner. Control points are chosen midway between the corners. The vorticity values are solved at the corner points using a boundary condition that requires the total normal velocity at each control point to be zero.

3.2.2 Equations

The velocity at any point P (Diagram 3.1) is given by (70):

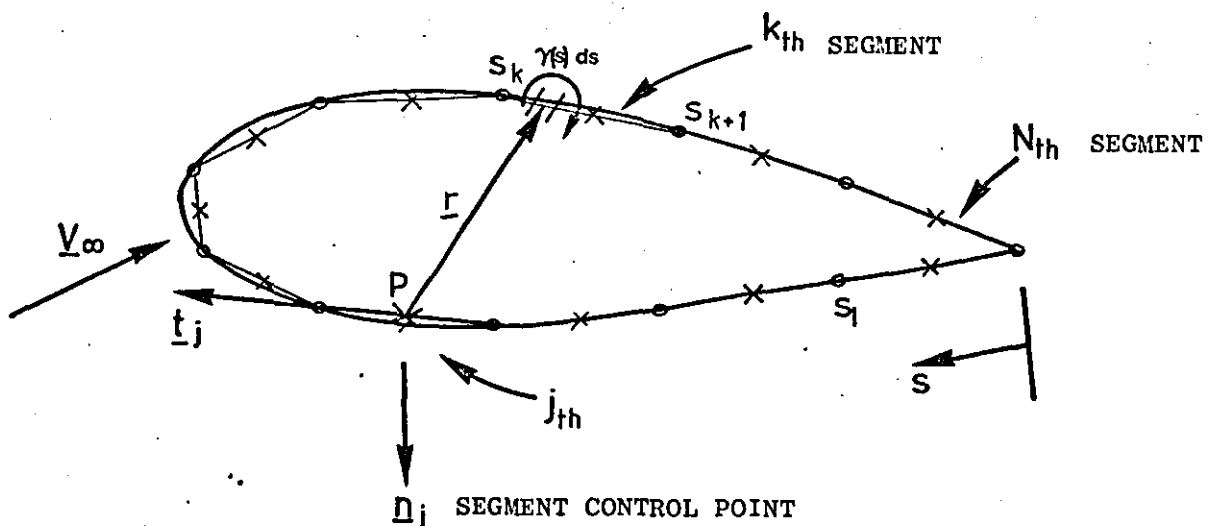


DIAGRAM 3.1

$$\frac{v}{p} = \frac{1}{2\pi} \oint \frac{\gamma(s) \underline{r}^j}{r^2} ds \pm \frac{\gamma(s)}{2} \frac{t_p}{p} + v_\infty \quad (3.1)$$

For zero normal velocity the product $\frac{v}{p} \cdot \underline{n}_p$ must be zero. Therefore

$$\frac{1}{2\pi} \oint \frac{\gamma(s) (\underline{r}^j) \cdot \underline{n}_p}{r^2} ds + v_\infty \cdot \frac{n_p}{p} = 0 \quad (3.2)$$

Using the panel method described above the integral equation is modified to a set of N linear equations given by the following formulae:

$$\frac{1}{2\pi} \sum_{k=1}^N (\gamma_k A_{jk}) + v_\infty \cdot \underline{n}_j = 0, \quad j=1,2,\dots,N \quad (3.3)$$

where

γ_k is the value of the singularity at corner point k.

A_{jk} is the influence coefficient of the singularity γ_k

associated with the control point j.

\underline{n}_j is the normal unit vector of panel j.

Note that if there are N corner points and hence N+1 unknown γ values at the corners, the N control points provide one less equation than unknowns. This situation is remedied by applying the Kutta condition at the trailing edge. This requires that $\gamma_{N+1} = -\gamma_1$, assuring that the velocities induced at the trailing edge are finite.

Having determined the vortex strengths, the velocity field and, hence, the pressure distribution around the aerofoil can be calculated.

3.3 SYMMETRICAL LINEAR VORTEX AND CONSTANT SOURCE SURFACE SINGULARITY MODEL

3.3.1 Basic Considerations

Bristow (7), Maskew and Woodward (72) have shown that if a combination of sources and vortices is used for the calculation of the potential flow about thick aerofoil sections, then the singularity strength values are minimized (Diagram 3.2); thus violently opposing singularity strengths on upper and lower surfaces are eliminated and any flow leakage effects are considerably reduced.

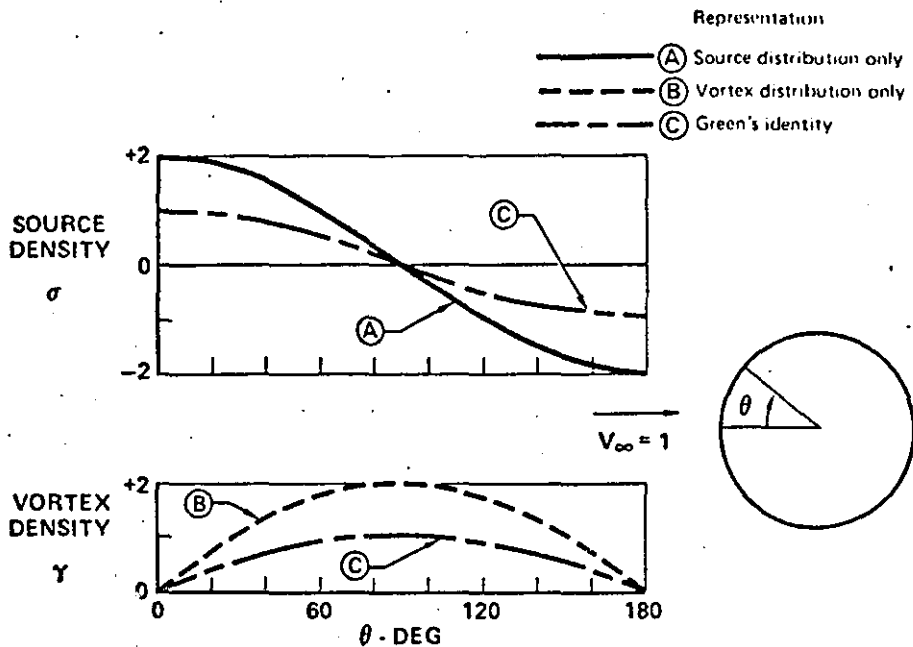


DIAGRAM 3.2: Equivalent singularity representation for a circular cylinder

Slotted aerofoil applications are particularly sensitive to flow leakage since it is possible for flow distortion on one surface to affect the control points on a neighbouring one, and also, because of the way the Kutta condition is applied, the overall circulation can be affected.

The symmetry of the model arises from the constraint that the mean line of the aerofoil is a streamline of the internal flow (72). Such a constraint would clearly require that the magnitudes of both the source and vortex distributions be equal at corresponding points on the upper and lower surfaces. By enforcing this simple symmetry condition, "doublet" effects between the upper and lower surfaces would be eliminated and the source and vortex singularities would again work in harmony each providing its own characteristic to the flow (72). In such a model, the correct singularity would become dominant in limiting cases where either incidence or thickness goes to zero.

The model can be extended to three dimensional potential flow calculations (85).

3.3.2 Theoretical Basis

The problem is to find a potential function ϕ having prescribed values over a certain boundary surface S and which satisfies Laplace's equation $\nabla^2\phi=0$ within the region enclosed by S . This is known as Dirichlet's problem (See integral equations by Moiseiwitsch). Neumann showed in 1870 that this problem is equivalent to the solution of an integral equation.

From reference 78 any solution to Laplace's equation, can be expressed in the form

$$\phi_P = \frac{1}{4\pi} \oint \left(\phi \frac{\partial}{\partial n} \left(\frac{1}{r} \right) - \frac{1}{r} \frac{\partial \phi}{\partial n} \right) dS \quad (3.1)$$

where S is a closed surface at every point of whose interior $\nabla^2\phi=0$ and $\nabla^2\psi=0$. P is a point interior to S (diagram 3.1), r is the distance of P from the element of area dS , \underline{n} is the normal vector to the body surface at dS , positive outwards and ϕ_P is the value of ϕ at P .

If the fluid extends to infinity then the integral needs to be taken only over the finite boundary (the aerofoil surface in this case) provided ϕ_P tends to zero at infinity (Ref. 70). The value of ϕ_P on the surface S is

$$\phi_P = \frac{1}{2\pi} \oint \left(\phi \frac{\partial}{\partial n} \left(\frac{1}{r} \right) - \frac{1}{r} \frac{\partial \phi}{\partial n} \right) dS \quad (3.2)$$

The first term in the integral equation (3.2) is the velocity potential for a surface distribution of doublets with axes normal to S and of density ϕ per unit area. The second term is the velocity potential for a surface distribution of sources with density $\sigma = -\frac{\partial \phi}{\partial n}$ per unit area.

For two dimensional flow, it is generally more convenient to use vortex singularities in place of doublets. It is possible (Ref. 7) to replace a surface doublet distribution of density μ by an equivalent vortex distribution where the vortex density vector $\underline{\gamma}$ satisfies the following equation at each surface point

$$\underline{\gamma} = \underline{n} \times \nabla \mu \quad (3.3)$$

The problem now becomes to determine the strength of the source and vortex singularities so that the required boundary conditions are satisfied everywhere.

3.3.3 Numerical Solution Procedure

There are many ways of obtaining a numerical solution to the integral equation. For example the singularities could be represented by a continuous series in which case the problem would be reduced to that of finding the coefficients of the singularity series. However, such an approach could pose severe difficulties when considering flap edges or rapid changes in surface conditions (70). Instead the integral equation is solved using a panel method.

The panel method approach has proved to be very powerful because it only requires specification of the boundaries. In contrast field methods such as finite difference or finite element methods (Refs. 2,18) need to consider the entire domain of perturbed flow.

The aerofoil surface is represented by an inscribed polygon with an even number of sides or panels. Each panel has a constant source distribution. The midpoint of each panel is selected as the boundary condition control point (Figure 3.1). The even number of panels allows corresponding upper and lower surface panels to be defined such that they have equal singularity strengths. The vorticity unknowns are located at the panel edges.

Assuming that there are N_m panels on the m^{th} aerofoil element then the number of unknown source strengths and vortex strengths on the element are N_m and $N_m + 1$ respectively. However, since there are only N_m control points and, therefore, N_m equations to solve, the extra unknown has to be eliminated. This is done, once again, by using the Kutta condition which in this case requires that

$$\gamma_1 = \gamma_{N+1} = 0$$

The condition is simultaneously applied to each trailing edge when considering a multielement aerofoil.

3.3.4 Velocity Equations

The velocity vector at a general point J in the plane of the aerofoil

is given by (25)

$$\underline{v}_J = \sum_{m=1}^{N_T} \left(\frac{1}{2\pi} \sum_{k=2}^{N_m} \bar{G}_V(J,K) \gamma_V(K) + \frac{1}{2\pi} \sum_{k=1}^{N_m} \bar{G}_S(J,K) \gamma_S(K) \right) + v_\infty \quad (3.4)$$

where N_T is the total number of aerofoil elements, N_m is the number of panels on the m^{th} element, $\gamma_V(K)$ is the vortex strength at the K^{th} corner point, $\gamma_S(K)$ is the source strength on the K^{th} panel, $G_V(J,K)$ and $G_S(J,K)$ are the influence coefficients for the velocity induced at the J^{th} control point by the vorticity and source on the K^{th} element respectively. (see Appendix 7).

Making use of the symmetry and Kutta conditions we can define new influence coefficients and singularity strengths such that,

$$\underline{v}_J = \sum_{m=1}^{N_T} \left(\frac{1}{2\pi} \sum_{k=1}^{N_m} \bar{G}(J,K) \gamma(K) \right) + v_\infty \quad (3.5)$$

where

$$\gamma(K) = \gamma_V(K+1) = \gamma_V(N_m - K + 1) \quad \left. \vphantom{\gamma(K)} \right\} \quad (3.6)$$

$$\bar{G}(J,K) = \bar{G}_V(J,K+1) + \bar{G}_V(J,N_m - K + 1), \quad K=1 \text{ to } N_m/2$$

$$\gamma(K) = \gamma_S(K - N_m/2) = \gamma_S\left(\frac{3N_m}{2} - K + 1\right) \quad \left. \vphantom{\gamma(K)} \right\} \quad (3.7)$$

$$\bar{G}(J,K) = \bar{G}_S\left(J, K - \frac{N_m}{2}\right) + \bar{G}_S\left(J, \frac{3N_m}{2} - K + 1\right), \quad K = \frac{N_m}{2} + 1 \text{ to } N_m$$

If J is now considered as a mid segment point on the aerofoil section surface we can resolve the velocity into components normal and tangential to the J^{th} segment to give

$$\underline{v}_{JN} = \sum_{m=1}^{N_T} \left(\frac{1}{2\pi} \sum_{k=1}^{N_m} GN(J,K) \gamma(K) \right) - v_\infty \sin(\theta_J - \alpha) \quad (3.8)$$

$$\underline{v}_{JT} = \sum_{m=1}^{N_T} \left(\frac{1}{2\pi} \sum_{k=1}^{N_m} GT(J,K) \gamma(K) \right) + v_\infty \cos(\theta_J - \alpha) \quad (3.9)$$

where $GN(J,K)$ and $GT(J,K)$ are the normal and tangential components of the influence coefficients, θ_J is the angle with which the J^{th} segment subtends the aerofoil chord and α is the incidence of the aerofoil chord to the free stream direction.

Since there are N panels and N unknown singularity strengths we have two N^{th} order systems of simultaneous equations which can for convenience be expressed in matrix form

$$[V_N] = \frac{1}{2\pi} [GN] [Y] - [V_\infty \sin(\theta-\alpha)] \quad (3.10)$$

$$[V_T] = \frac{1}{2\pi} [GT] [Y] - [V_\infty \cos(\theta-\alpha)] \quad (3.11)$$

$$N = \sum_{m=1}^{N_T} N_m$$

Eliminating γ between equations (3.10) and (3.11) the following result is obtained

$$[V_T] = [GT] [GN]^{-1} ([V_N] + [V_\infty \sin(\theta-\alpha)]) + [V_\infty \cos(\theta-\alpha)] \quad (3.12)$$

For the inviscid case the normal velocity on the aerofoil surface is zero and the tangential velocity therefore becomes the inviscid solution to the flow about the aerofoil. Hence the inviscid velocity is

$$\left. \begin{aligned} [V_I] &= [A] [V_\infty \sin(\theta-\alpha)] + [V_\infty \cos(\theta-\alpha)] \\ [A] &= [GT] [GN]^{-1} \end{aligned} \right\} \quad (3.13)$$

Note that the velocities are calculated at the panel midpoints.

The pressure coefficient C_p is calculated from

$$C_p = 1 - \left(\frac{V_I}{V_\infty} \right)^2 \quad (3.14)$$

This result is then used to calculate the boundary layer development over the aerofoil as shown in the following sections.

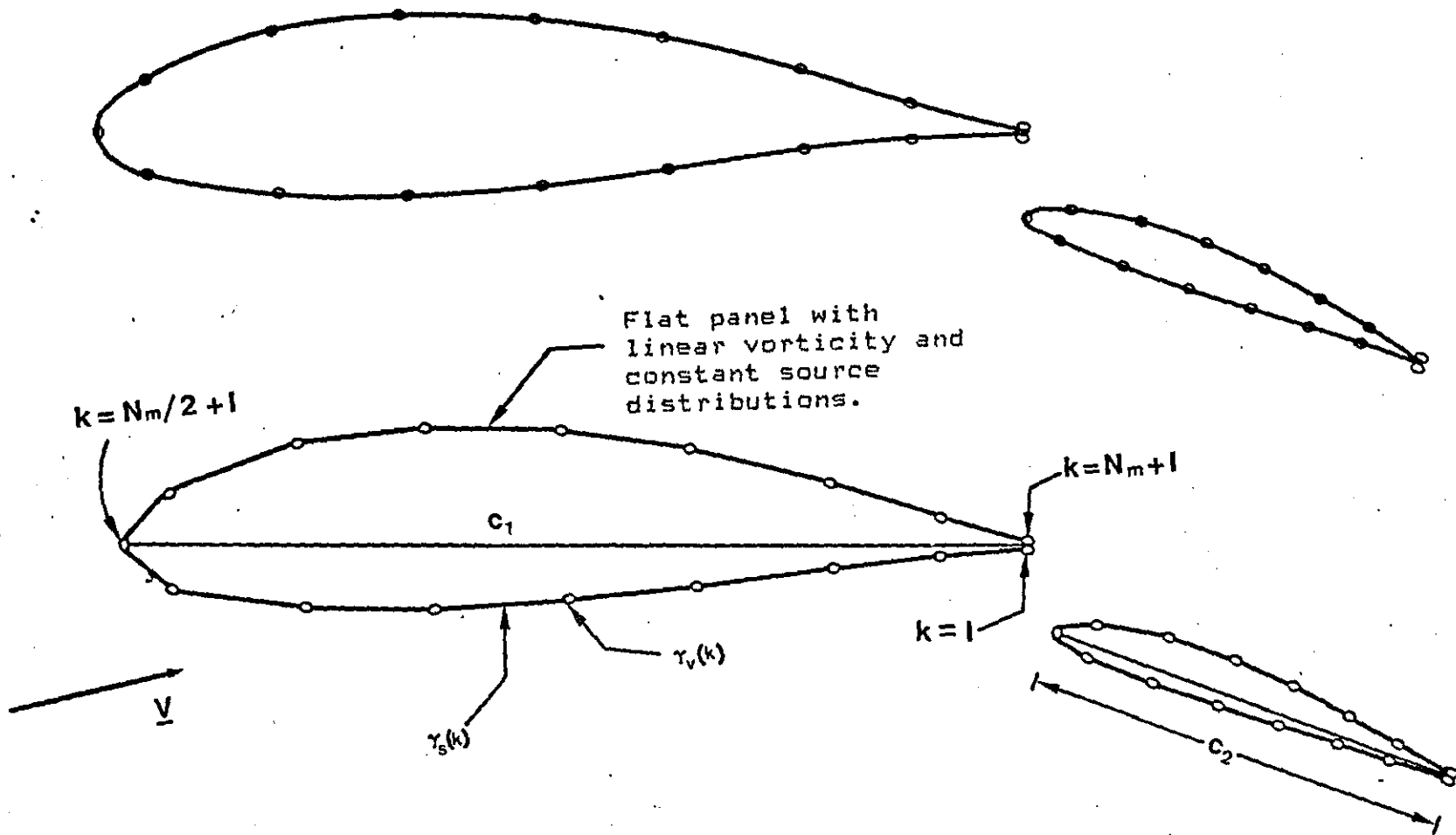


FIGURE 3.1 Aerofoil Surface Representation

VISCOUS FLOW - A GENERAL DISCUSSION

4.1 INTRODUCTION

Once the potential flow calculation has been completed the ideal velocity and therefore pressure distributions about the aerofoil are known.

To make use of the above calculation certain assumptions have to be made before the viscosity effects can be considered.

Starting with a short introduction into the boundary layer concept, this section gives a brief description of all viscous regions encountered in the present analysis, and outlines the general calculation procedure.

Having obtained the viscous flow parameters ways must be found to include these parameters in the flow model. This is known as the viscous flow representation. Ways for representing viscosity are described at the end of the section.

4.2 GENESIS OF THE BOUNDARY LAYER CONCEPT

The equations of motion of a viscous fluid were established in the first half of the last century by Navier (1832), Poisson (1831), Saint Venant (1843) and Stokes (1845) having attained the form that is now called the Navier-Stokes equations.

Attempts to obtain exact solutions to these equations, however have failed apart from a small number of special cases where the non-linear terms were either negligibly small or identically vanishing. This having not been the case in the majority of the problems met in practice, it was necessary to introduce some approximations for solutions. The simplest was of course to neglect the viscosity of the fluid, but this brought about

nothing but the d'Alembert paradox, according to which a solid body of any shape placed in a uniform stream experiences no resistance.

As a second approximation the non-linear terms of the equations were ignored. This assumption is only true for slow motion, however it was hoped that for faster motions the solutions might give a better representation of the flow than those obtained by neglecting viscosity. Indeed drag was predicted this time but its magnitude was rather too small.

In the paper of 1905, Prandtl started from the clear recognition that the most important consideration for fluids of small viscosity is the behaviour of the fluid at the wall of the solid boundary. He suggested that the variation of velocity from the value corresponding to irrotational motion to the zero velocity demanded by the condition of no slip at the wall takes place within a thin layer adjacent to the wall. This layer was called by Prandtl transition layer or boundary layer. The smaller the viscosity the thinner is the transition layer. In spite however of the small viscosity, the steep velocity gradient, produces marked effects which are comparable in magnitude to those due to inertia force.

The small thickness of the boundary layer allows the following approximations to be made:

1. The variation of pressure normal to the wall is negligibly small, and the variation of velocity along the wall is much smaller than its variation normal to it.
2. For two dimensional flows the effect of moderate curvature of the wall is negligibly small so that x and y may be taken as the distances along and normal to the wall, respectively and u and v as the corresponding velocity components.

The x -component of the Navier-Stokes equations is then simplified to the form

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} + \frac{1}{\rho} \frac{\partial p}{\partial x} = \nu \frac{\partial^2 u}{\partial y^2} \quad (4.1)$$

where

t = time ρ = density
 p = pressure ν = kinematic viscosity

The pressure p is regarded as a function of x and t and prescribed by the irrotational motion (potential flow) outside the boundary layer. The equation above is parabolic, although the original Navier-Stokes equations are elliptic. Thus it can be integrated step by step in the direction of x when u is known at a fixed value of x for all values of y and t , the upstream influence being suppressed to the order of approximation.

From a number of investigations carried out Prandtl showed that in certain cases, the flow separates from the surface at a point entirely determined by external conditions, it was explained that this was due to the increase in pressure in the streamwise direction.

Prandtl's model of the viscous flow has proved to be a very useful one. A greater number of boundary layer methods have evolved from Prandtl's concept (see Ref. 114). Most of these methods are concerned with the calculation of laminar and turbulent boundary layers.

4.3 VISCOUS FLOW MODEL

The viscous flow model used here is divided into the following main regions:

1. Laminar boundary layer
2. Boundary layer transition
3. Short bubble flow
4. Turbulent boundary layer
5. Wake viscous flow
6. Confluent boundary layer.

Figure 4.1 shows all viscous flow regions involved in the model together with the methods used for their analysis.

A laminar boundary layer is assumed to exist on the upper and lower surface of each aerofoil component. At some stage the laminar boundary layer becomes unstable and undergoes a transition phase before the flow becomes turbulent. Transition may take place over a short bubble whose behaviour plays a vital part in the overall flow behaviour. The method used for the calculation of the above viscous regions are based on the following equations

$$u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} = U \frac{dU}{dx} + \frac{\partial}{\partial y} \left(\frac{\tau}{\rho} \right) \quad (4.2)$$

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

and on a number of experimental results.

The viscous flow in the wake behind the aerofoil is calculated using Green's lag entrainment method.

The effects of viscosity are more complex and stronger on multiple aerofoils, in particular, when the wake of the main aerofoil mixes in with the boundary layer along the flap. In such cases methods for calculating the development of ordinary boundary layers cannot be applied directly and must be extended to cover these more complex flows (65). Consequently, the available methods are even more tentative. Of these, only some that have proved useful are mentioned: Foster (38), Foster and Irwin (39), Coradia (109).

Boundary layer and wake surveys indicate that, in the configuration shown in Figure 4.2b, there is a stream of clean air flowing through the slot with full total head (core region), and the boundary layer on the flap only just merges with the wake from the main aerofoil. If the gap is wider, the lift drops, following the trend in inviscid flow, which is reinforced by viscous losses on the flap itself (65). If the gap is smaller,

the viscous layers merge and this reduces the effectiveness of the slot and leads to a substantial loss of lift.

More details about the individual viscous flow regions could be found in the appropriate sections of this work, where methods for the calculation of the particular region are also described.

4.4 VISCOUS FLOW REPRESENTATION

The boundary layer causes the irrotational flow outside it to be that about not the solid surface itself, but a surface displaced into the fluid through a distance δ^* , the 'displacement thickness' of the layer.

Let x be the distance along the surface in consideration (e.g. aerofoil) and y the distance measured normal to the surface, let u and v be the corresponding velocities and U the value of u just outside the boundary layer. The difference $(U-u)$ represents the reduction in flow velocity due to the presence of rotational flow in the boundary layer. The total reduction in volume flow per unit span is:

$$\int_0^{\infty} (U-u) dy$$

Now between the surface and any stream line just outside the boundary layer, there must be a constant volume flow per unit span. This will be so if the flow reduction inside the layer is compensated for by an outward displacement of such a stream line through a distance δ^* (which produces a flow increase $U\delta^*$, since the velocity is U in the region of stream line displacement), provided that,

$$\delta^* = \frac{1}{U} \int_0^{\infty} (U-u) dy \quad (4.3)$$

This displacement of the irrotational flow streamlines implies that they can be regarded as streamlines of the irrotational flow around a surface displaced into the fluid through a distance δ^* .

A similar argument may be used for the wake behind the aerofoil. In this case:

$$\delta^* = \frac{1}{U} \int_{-\infty}^{\infty} (U-u) dy$$

Computationally however it is not very efficient to simply add δ^* to the y-coordinates of the aerofoil or wake. This is because the influence coefficients of the aerofoil and wake will have to be recalculated. Instead other ways for taking into account the displacement thickness are used. (Ref. 67,68).

The most commonly used method represents the effect of the boundary layer by means of an equivalent surface distribution of sources whose strength is defined by

$$\sigma = \frac{d}{dx} \int_0^{\infty} (U-u) dy = \frac{d}{dx}(U\delta^*) \quad (4.4)$$

(see Appendix 10).

A method similar to the one just described, but which approaches more directly the problem of finding a surface $y=\delta^*(x)$, the irrotational flow about which is the same as the flow outside the given boundary layer, is the 'velocity comparison' method. This method is used in the present mathematical model.

As shown in Appendix 10, there is at the edge of the boundary layer a normal velocity condition imposed on the outer inviscid flow of:

$$v = \frac{1}{\rho} \frac{d(\rho U \delta^*)}{dx}$$

which for incompressible flow is

$$v = \frac{d(U\delta^*)}{dx} \quad (4.5)$$

Lighthill (67) describes another method where the problem is that of finding the flow induced in the presence of the body by a given distribution of vorticity. This is the mean velocity method.

4.5 AEROFOIL SURFACE VELOCITY EQUATIONS

Having calculated the normal velocity $v=V_N$, equation (3.12) is solved with this prescribed normal velocity. Finally consistent with integral boundary layer theory the normal velocity is considered small in relation to the tangential velocity and so the latter is taken as the complete solution (25).

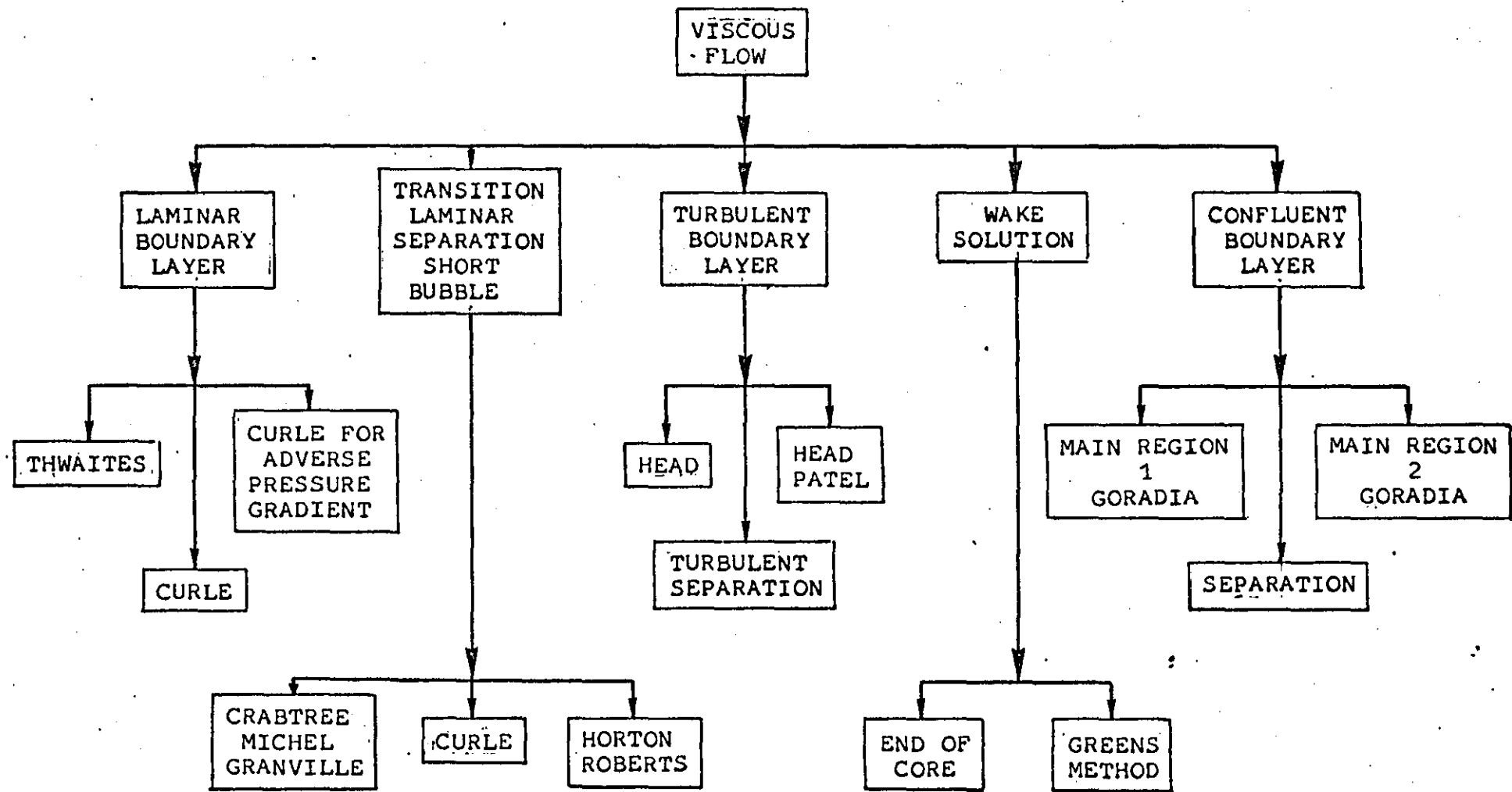
$$[V_T] = [V_I] + [V_{BL}] \quad (4.6)$$

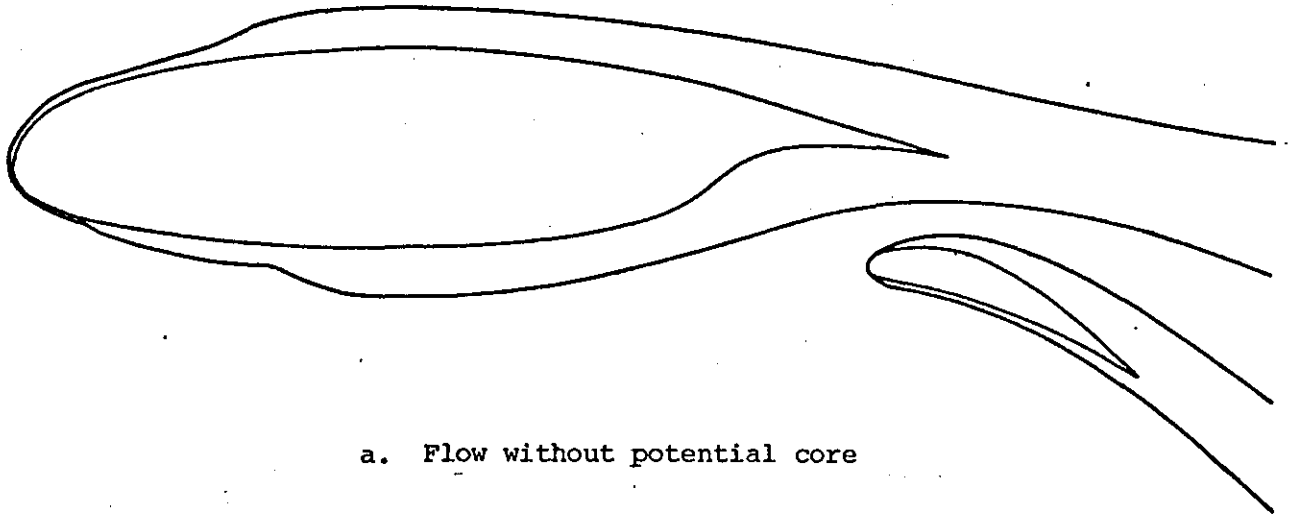
where

$$[V_I] = [A] [V_\infty \sin(\theta-\alpha)] + [V_\infty \cos(\theta-\alpha)]$$

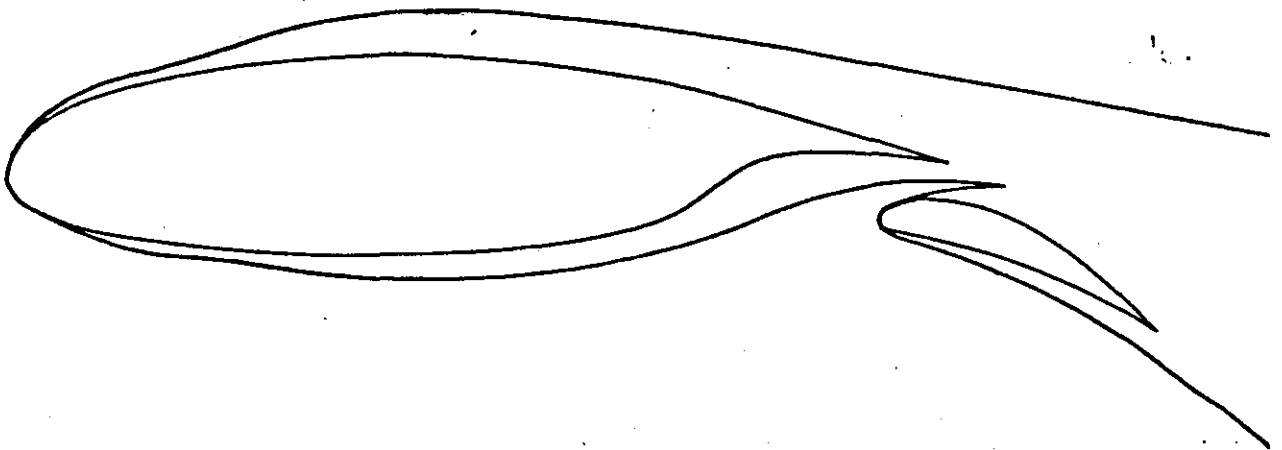
$$[V_{BL}] = [A] \left[d \frac{(U\delta^*)}{ds} \right]$$

$$[A] = [GT] [GN]^{-1}$$





a. Flow without potential core



b. Flow with potential core

FIGURE 4.2 Attached Flow Models

LAMINAR BOUNDARY LAYER

5.1 INTRODUCTION

Two methods for predicting the incompressible laminar boundary layer parameters will be outlined briefly. Both methods are of the integral type.

5.2 THWAITES METHOD

Thwaites method (Ref. 117) is universally accepted as one of the better of the existing calculation methods.

Starting with Prandtl's equation of motion of steady two dimensional boundary layer flow

$$u \frac{\partial u}{\partial x} + v \frac{\partial v}{\partial x} = U \frac{\partial U}{\partial x} + v \frac{\partial^2 u}{\partial y^2} \quad (5.1)$$

and integrating between $y=0$ and $y=\infty$ using

$$\begin{aligned} \delta^* &= \int_0^{\infty} \left(1 - \frac{u}{U}\right) dy \\ \theta &= \int_0^{\infty} \frac{u}{U} \left(1 - \frac{u}{U}\right) dy \\ H &= \delta^* / \theta \end{aligned}$$

and the continuity equation

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0 \quad (5.2)$$

the integral momentum equation is obtained

$$\frac{d\theta}{dx} = -(H+2) \frac{dU}{dx} \frac{\theta}{U} + \frac{v}{U^2} \left(\frac{\partial u}{\partial y}\right)_{y=0} \quad (5.3)$$

Setting $y=0$ in equation 5.1 gives

$$\frac{dU}{dx} U + v \left(\frac{\partial^2 u}{\partial y^2}\right)_{y=0} = 0 \quad (5.4)$$

The essential features of equations (5.3) and (5.4) are the two terms:

$$\left(\frac{\partial u}{\partial y}\right)_{y=0} \quad \text{and} \quad \left(\frac{\partial^2 u}{\partial y^2}\right)_{y=0}$$

From equation (5.4)

$$-U \frac{dU}{dx} = v \left(\frac{\partial^2 u}{\partial y^2}\right)_{y=0} \quad (5.5)$$

Therefore in order to integrate (5.3) a relation between $(\partial u / \partial y)_{y=0}$ and $(\partial^2 u / \partial y^2)_{y=0}$ is required. For if some such relation is assumed (e.g. by the use of a family of velocity distributions), then $(\partial u / \partial y)_{y=0}$ is known as a function of x since the value of $(\partial^2 u / \partial y^2)_{y=0}$ is known from (5.5).

The two derivatives of u with respect to y indicate the behaviour of the velocity distribution at the boundary. Its general 'shape' elsewhere is indicated by the value of the shape parameter u , which also occurs in the momentum equation.

Equation (5.3), therefore, is not only concerned with boundary values, but relates these with the velocity distribution as a whole. For this reason the momentum equation is capable of giving good results. Thus a relation between H and $(\partial^2 u / \partial y^2)_{y=0}$ is also required.

Defining two parameters m and l as:

$$\left. \begin{aligned} m &= \frac{\theta^2}{U} \left(\frac{\partial^2 u}{\partial y^2}\right)_{y=0} = -\frac{\theta^2}{v} \frac{dU}{dx} \\ l &= \frac{\theta}{U} \left(\frac{\partial u}{\partial y}\right)_{y=0} \end{aligned} \right\} \quad (5.6)$$

equations (5.3) and (5.4) can be written as follows:

$$\left. \begin{aligned} \frac{d\theta}{dx} &= -(H+2) \frac{du}{dx} \frac{\theta}{U} + \frac{v l}{U \theta} \\ 0 &= U \frac{dU}{dx} + \frac{v U m}{\theta^2} \end{aligned} \right\} \quad (5.7)$$

Taking θ as the principal dependent variable and assuming that H and l are only functions of m Thwaites obtained the following equations:

$$\left. \begin{aligned} \frac{U}{v} \frac{d\theta^2}{dx} &= L(m) \\ L(m) &= 2((H(m)+2)m+\ell m) \end{aligned} \right\} \quad (5.8)$$

The assumptions, giving 5.8, are based on a critical examination of known solutions to the laminar boundary layer problem.

Thwaites has simplified the above equations even further by taking

$$L(m) = 0.45 + 6m$$

without seriously affecting the accuracy of the solution.

Thus the final set of equations required for the analysis of the laminar boundary layer is

$$\left. \begin{aligned} \theta^2 &= 0.45 U^{-6} v \int_0^x U^5 dx \\ m &= \frac{dU}{dx} \frac{\theta^2}{v} \\ \left(\frac{\partial u}{\partial y}\right)_{y=0} &= \frac{U}{\theta} \ell(m) \\ \delta^* &= \theta H(m) \end{aligned} \right\} \quad (5.9)$$

with $\ell(m)$, $H(m)$ and $L(m)$ given in Table 5.1 (ref. 117).

At any point of the system under consideration U , dU/dx and x are available so that θ can be found from (5.9). Once θ is known m , $\left(\frac{\partial u}{\partial y}\right)_{y=0}$ and δ^* follow from (5.9) and Table 5.1.

The parameter m is known as the pressure gradient parameter and if negative the pressure is decreasing (i.e. favourable pressure gradient). Positive values of m refer to adverse pressure gradient (i.e. increasing pressure with x). If the value of m becomes greater or equal to 0.082 laminar separation takes place.

5.3 CURLE'S METHOD

In this method the non-dimensional parameters H , ℓ and L are assumed to depend on two parameters namely:

$$\lambda = \frac{dU}{dx} \frac{\theta^2}{v} \quad (5.10)$$

and

$$\mu = \frac{\lambda^2 \theta \theta''}{(U')^2}$$

Thus

$$\begin{aligned} L &= F_0(\lambda) - \mu G_0(\lambda) \\ L^2 &= F_1(\lambda) - \mu G_1(\lambda) \end{aligned} \quad (5.11)$$

Careful examination of the available range of exact solutions of the boundary layer equations has enabled the four functions F_0, G_0, F_1, G_1 to be tabulated and the above functional forms agree with the exact solutions to a remarkable accuracy (Ref.28).

5.3.1 Equations

The full set of equations necessary for the calculation of the laminar boundary layer is:

$$\theta^2 = 0.45 \nu U^{-6} \int_0^x (1+2.22g) U^5 dx \quad (5.12)$$

$$\lambda = \frac{dU}{dx} \frac{\theta^2}{\nu}, \quad \mu = \frac{\lambda^2 \theta \theta''}{(U')^2} \quad (5.13)$$

$$g = F_0(\lambda) - 0.45 + 6\lambda - \mu G_0(\lambda) \quad (5.14)$$

$$L = F_0(\lambda) - \mu G_0(\lambda) = 0.45 - 6\lambda + g \quad (5.15)$$

$$L^2 = F_1(\lambda) - \mu G_1(\lambda) \quad (5.16)$$

$$H = -\left(2 + \frac{L-2\theta}{2\lambda}\right) \quad (5.17)$$

$$\delta^* = H\theta \quad (5.18)$$

$$\tau_w = \frac{\mu U}{\theta} \quad (5.19)$$

5.3.2 Calculation Procedure

The method of completing a calculation of the usual boundary layer characteristics is as follows.

Firstly the value of θ at station x is found from (5.12) with $g=0$. Once again it is assumed that U and its derivatives are known at x .

From the value of θ, U, U' and U'' the values of λ and μ can be obtained. Table 5.2 is used at this stage to find $F_0(\lambda)$ and $G_0(\lambda)$. Equation (5.14) gives a new estimate for g and the calculation is repeated until θ has converged to the required decimal figure. When the final value of θ has been found together with the final values of λ and μ at station x , F_0, F_1, G_0 and G_1 are found from Table 5.2. The calculation proceeds by finding L, ℓ, H, δ^* and τ_w from the equations above.

The above procedure is repeated at each station x until laminar separation or transition is predicted.

5.3.3 Laminar Separation

At separation

$$F_1(\lambda) - \mu G_1(\lambda) = 0 \quad (5.20)$$

which yields a relationship between λ and μ at separation which supercedes Thwaites separation criterion ($\lambda = -0.082$).

Generally speaking Curle's method is a refinement of that due to Thwaites. Reference 28 indicates that the errors of this method are only 5% of those given by Thwaites method (see Tables 5.3 and 5.4).

<i>m</i>	<i>l</i> (m)	<i>H</i> (m)	<i>L</i> (m)
+0.082	0	3.70	0.938
+0.0818	0.011	3.69	0.953
+0.0816	0.016	3.66	0.956
+0.0812	0.024	3.63	0.962
+0.0808	0.030	3.61	0.967
+0.0804	0.035	3.59	0.969
+0.080	0.039	3.58	0.971
+0.079	0.049	3.52	0.970
+0.078	0.055	3.47	0.963
+0.076	0.067	3.38	0.952
+0.074	0.076	3.30	0.936
+0.072	0.083	3.23	0.919
+0.070	0.089	3.17	0.902
+0.068	0.094	3.13	0.886
+0.064	0.104	3.05	0.854
+0.060	0.113	2.99	0.825
+0.056	0.122	2.94	0.797
+0.052	0.130	2.90	0.770
+0.048	0.138	2.87	0.744
+0.040	0.153	2.81	0.691
+0.032	0.168	2.75	0.640
+0.024	0.182	2.71	0.590
+0.016	0.195	2.67	0.539
+0.008	0.208	2.64	0.490
0	0.220	2.61	0.440
-0.016	0.244	2.55	0.342
-0.032	0.268	2.49	0.249
-0.048	0.291	2.44	0.156
-0.064	0.313	2.39	0.064
-0.080	0.333	2.34	-0.028
-0.10	0.359	2.28	-0.138
-0.12	0.382	2.23	-0.251
-0.14	0.404	2.18	-0.362
-0.20	0.463	2.07	-0.702
-0.25	0.500	2.00	-1.000

TABLE 5.1

TABLE 5.2
The Universal Functions

λ	F_0	G_0	$F_0 - 0.45 + 6\lambda$	F_1	G_1	F_1/G_1
0.0855	0	0.9165	0.0630	0.1296	0.2626	
0.08	0.0258	0.90	0.0558	0.1236	0.2535	
0.07	0.0736	0.87	0.0436	0.1128	0.2378	
0.06	0.1225	0.84	0.0325	0.1025	0.2228	
0.05	0.1724	0.81	0.0224	0.0925	0.2087	
0.04	0.2236	0.78	0.0136	0.0830	0.1953	
0.03	0.2761	0.75	0.0061	0.0738	0.1827	
0.02	0.3299	0.72	-0.0001	0.0651	0.1710	
0.01	0.3848	0.69	-0.0052	0.0567	0.1600	
0	0.4410	0.66	-0.0090	0.0487	0.1498	
-0.01	0.4987	0.63	-0.0113	0.0411	0.1404	
-0.02	0.5572	0.60	-0.0128	0.0338	0.1318	
-0.03	0.6167	0.57	-0.0133	0.0270	0.1240	
-0.04	0.6777	0.54	-0.0123	0.0207	0.1160	
-0.05	0.7404	0.51	-0.0096	0.0149	0.1075	0.139
-0.06	0.8053	0.48	-0.0047	0.0095	0.0971	0.098
-0.07	0.8729	0.45	+0.0029	0.0047	0.0852	0.055
-0.08	0.9434	0.42	0.0134	0.0010	0.0728	0.014
-0.09	1.0166	0.39	0.0256	-0.0019	0.0601	-0.032
-0.10	1.0928	0.36	0.0428	-0.0039	0.0470	-0.083
-0.11	1.1724	0.33	0.0624	-0.0051	0.0335	-0.152
-0.12	1.2540	0.30	0.0840	-0.0055	0.0197	-0.279
-0.13	1.3373	0.27	0.1073	-0.0051	0.0054	-0.944
-0.133 (7)	1.3686	0.2589	0.1164	-0.0047	0	$-\infty$

TABLE 5.3
Calculated Values of L

u_1/u_0	λ	μ	$L(\text{exact})$	$L(\text{Thwaites})$	$L(\text{eqn 15})$	u_1/u_0	λ	μ	$P(\text{exact})$	$P(\text{Thwaites})$	$P(\text{eqn 16})$
ξ^n	0.0855	0	0	-0.063	0	ξ^n	0.0855	0	0.1296	0.1156	0.1296
	0.0611	-0.0075	0.122	0.083	0.123		0.0611	-0.0075	0.1050	0.0955	0.1053
	0.0333	-0.0089	0.267	0.250	0.266		0.0333	-0.0089	0.0784	0.0729	0.0786
	0	0	0.441	0.450	0.441		0	0	0.0486	0.0484	0.0487
	-0.0266	0.0156	0.584	0.610	0.587		-0.0266	+0.0156	0.0269	0.0313	0.0273
	-0.0488	0.0345	0.708	0.743	0.715		-0.0488	0.0345	0.0110	0.0186	0.0119
-0.0682	0.0561	0.824	0.859	0.835	-0.0682	0.0561	0	0.0689	0.0007		
$1-\xi$	-0.0121	0	0.512	0.523	0.511	$1-\xi$	-0.0121	0	0.0400	0.0404	0.0396
	-0.0346	0	0.647	0.658	0.645		-0.0346	0	0.0246	0.0266	0.0241
	-0.0645	0	0.836	0.837	0.836		-0.0645	0	0.0077	0.0106	0.0073
	-0.0841	0	0.980	0.955	0.973		-0.0841	0	0	0.0015	-0.0002
$\sin \xi$	0.0685	-0.0114	0.091	0.039	0.091	$\sin \xi$	0.0685	-0.0114	0.1140	0.1015	0.1139
	0.0165	-0.0540	0.388	0.351	0.388		0.0165	-0.0540	0.0713	0.0600	0.0712
	-0.0243	-0.0944	0.639	0.596	0.641		-0.0243	-0.0944	0.0434	0.0328	0.0429
	-0.0582	-0.1322	0.860	0.799	0.858		-0.0582	-0.1322	0.0238	0.0137	0.0236
	-0.0865	-0.1661	1.056	0.969	1.057		-0.0865	-0.1661	0.0103	0.0006	0.0098
	-0.1170	-0.2056	1.292	1.152	1.293		-0.1170	-0.2056	0	---	-0.0006
$1-\xi^2$	0.028	+0.011	0.277	0.282	0.279	$1-\xi^2$	0.028	+0.011	0.0702	0.0686	0.0701
	-0.039	-0.020	0.684	0.684	0.682		-0.039	-0.020	0.0237	0.0240	0.0236
	-0.074	-0.041	0.922	0.894	0.919		-0.074	-0.041	0.0062	0.0062	0.0065
	-0.092	-0.053	1.056	1.002	1.052		-0.092	-0.053	0	---	0.0001

TABLE 5.4
Calculated Values of P

BOUNDARY LAYER TRANSITION

6.1 INTRODUCTION

The problem of transition from laminar to turbulent flow in the boundary layer is unparalleled in having attracted the interest of investigators for so many years. Despite the enormous amount of research effort devoted to it, our understanding of the problem is far from complete. Tani (Ref. 113) and Reshotko (Ref. 92,93) give a detailed description of the phenomenon based on our present knowledge of it.

Predicting the onset of transition is a very difficult task since a sound knowledge of the individual effects of all factors affecting transition is required. Such important factors are: Pressure gradient, surface curvature, free stream turbulence and surface roughness (Ref. 113).

It was early hypothesized by Reynolds that transition is a consequence of instability of the laminar flow. This hypothesis was further developed by Rayleigh (Ref. 100) and to this day remains most highly regarded. For the purpose of predicting the onset of transition a number of simple semiempirical methods have been developed (e.g. Ref. 22,55).

Three such methods have been chosen, on the basis of simplicity, accuracy and computational efficiency, for the present mathematical model. These methods are briefly outlined in this section.

6.2 CRABTREE CRITERION

An empirical criterion for transition is obtained by plotting the boundary layer Reynolds number based on momentum thickness (R_{θ}) against the pressure gradient parameter (m).

By calculating R_{θ} and m at the observed transition position on various aerofoil sections Crabtree (Ref. 22) obtained a universal curve (Figure 6.1).

To use this curve for predicting the transition position (more strictly the limit of existence of the laminar layer) it is only necessary to calculate the boundary layer development and to plot R_{θ} and m on (Figure 6.1). The intersection with the universal curve gives the estimated transition conditions.

From (Figure 6.1) it can be seen that the curve is not very well established in the region where $m < 0.04$ due to the scarcity of experimental points in regions of low pressure gradients. In any case the accuracy is expected to be highest where $m > 0.04$ for then due to the steep adverse pressure gradients the value of m changes quite rapidly with distance along the surface.

6.3 MICHEL'S METHOD

A method proposed by Michel (Ref. 77) depends on a universal curve of R_{θ} at transition against R_s at transition where s is distance measured along the aerofoil surface. The calculated curve of R_{θ} versus R_s for a given case then intersects the universal curve at the predicted transition point. Michel's method gives good results in two dimensional cases provided the velocity distribution is similar to one or more of those used in constructing the universal curve (Figure 6.2).

6.4 GRANVILLE'S METHOD

In Granville's method it is first necessary to determine the limit of stability from the universal curve of R_{θ} versus m . A process of iteration using a curve of average R_{θ} against average m over the unstable region then yields the transition position.

6.4.1 Instability Prediction

As mentioned above the first step in the calculation is to locate the point of laminar instability. Schlichting (Ref. 100) has solved the Orr-Sommerfeld equation assuming Polhausen laminar velocity profiles. The results of this linearized stability are presented in (Figure 6.3).

Using the following equation to approximate the curve the value of $(R_{e_\theta})_{crit}$ can be calculated for each value of the pressure parameter. Instability is predicted if the local value of R_{e_θ} is greater than the corresponding critical value.

$$(R_{e_\theta})_{crit} = \exp(5.46963 + 43.37458k + 218.28k^2 - 1934.6k^3 - 23980k^4) \quad \text{for } -0.1567 \leq k \leq 0.0767 \quad (6.1)$$

6.4.2 Transition Prediction

Once the instability point has been located the search for the transition point begins. Granville (Ref. 109) correlated

$$\Delta(R_{e_\theta})_{tran} = (R_{e_\theta})_{tran} - (R_{e_\theta})_{inst} \quad (6.2)$$

against the average pressure gradient parameter \bar{k} which is equal to

$$\bar{k} = \frac{1}{s - s_{inst}} \int_{s_{inst}}^s k \, ds \quad (6.3)$$

The symbols in (Figure 6.4) represent the correlation of experimental data by Granville. Goradia (Ref. 8) has approximated this curve by the following equation

$$\Delta(R_{e_\theta})_{tran} = 825.45 + 28183.5\bar{k} + 721988\bar{k}^2 + 6317380\bar{k}^3 \quad -0.05 \leq \bar{k} \leq 0.0767 \quad (6.4)$$

Transition is said to have occurred if

$$\Delta R_{e_\theta} = (R_{e_\theta})_{loc} - (R_{e_\theta})_{inst} \geq \Delta(R_{e_\theta})_{tran} \quad (6.5)$$

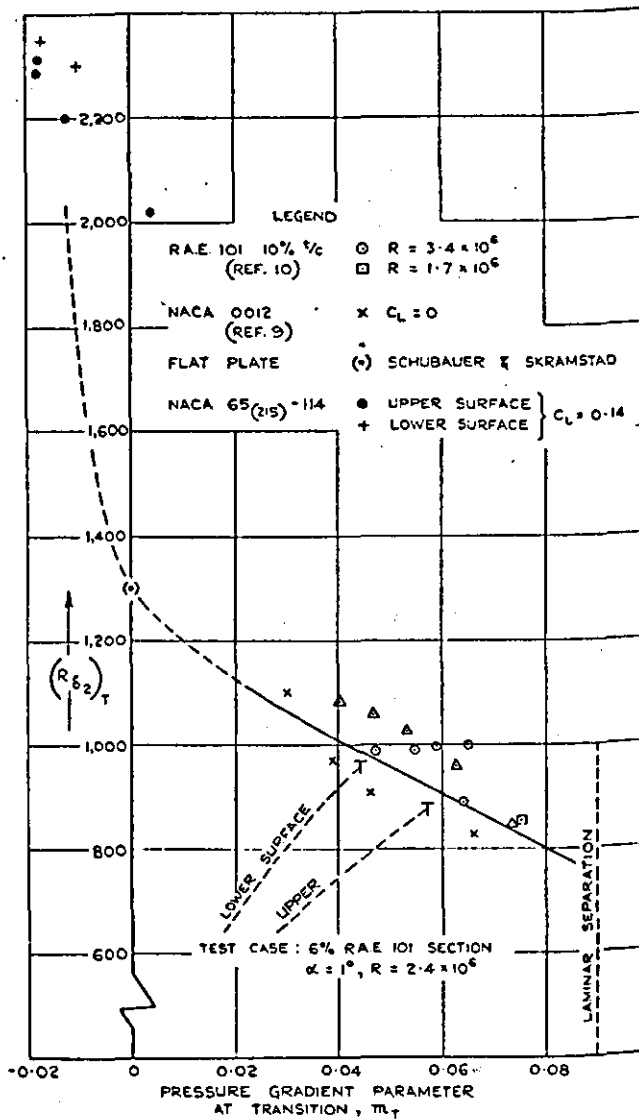


FIGURE 6.1: Boundary layer Reynolds number at transition against pressure gradient parameter

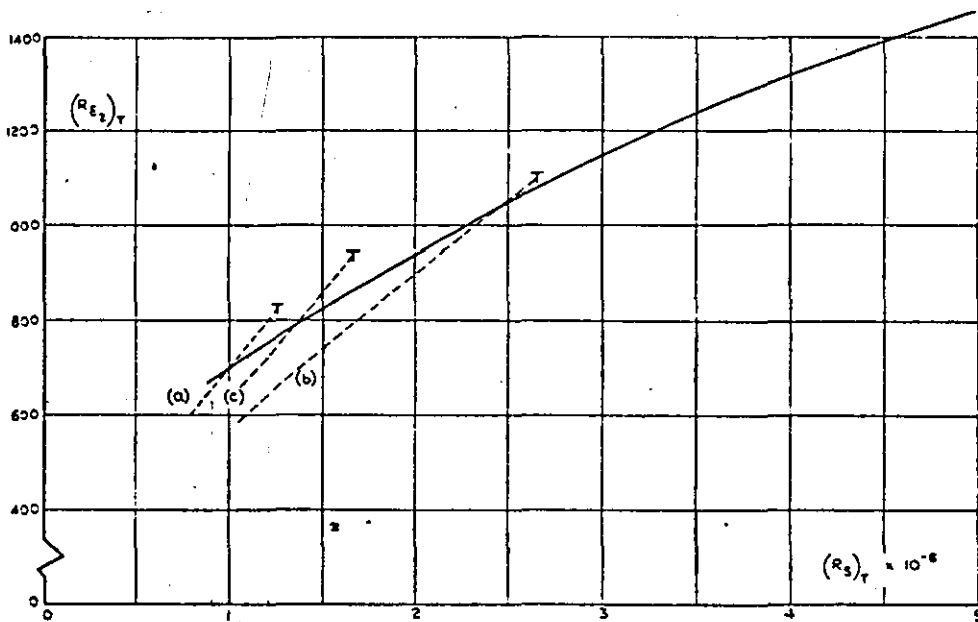


FIGURE 6.2: R_{δ_2} at transition plotted against R_s

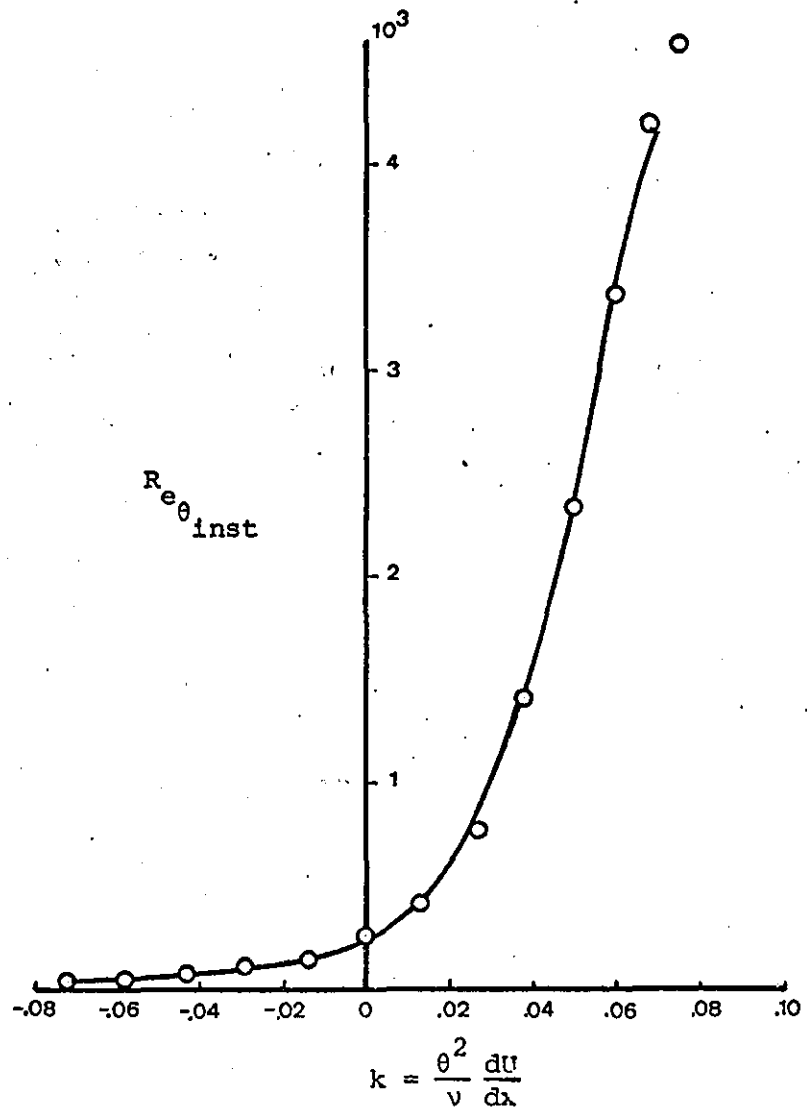


FIGURE 6.3: Stability Curve

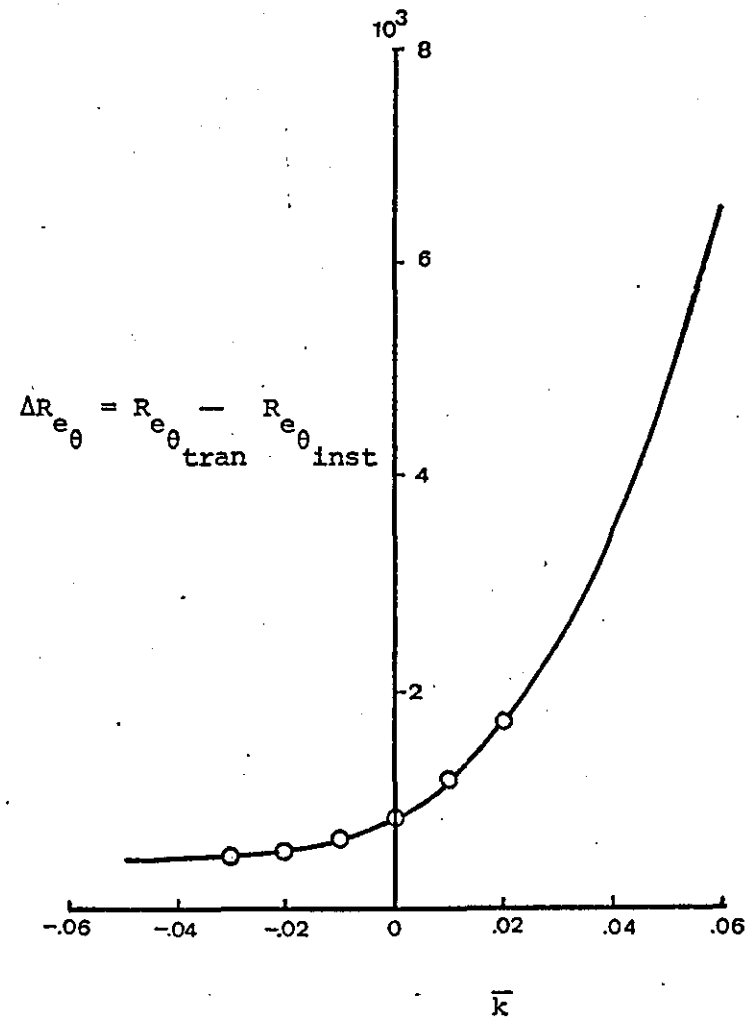


FIGURE 6.4: Transition Curve

SHORT BUBBLE ANALYSIS

7.1 INTRODUCTION

The laminar boundary layer over the nose of a thin aerofoil at high incidence fails to remain attached to the upper surface in the region of high adverse pressure gradient that occurs just downstream of the suction peak. The separated shear layer which is formed may curve back to the aerofoil surface to form a shallow region of reverse flow known as a separation bubble. The fluid is static in the forward region of the bubble and a constant pressure region results. At high Reynolds numbers the extent of such a bubble is very small of the order of 1 per cent chord (Ref. 41), and the slight step in the pressure distribution produced by the dead air region has a negligible effect on the forces acting on the aerofoil. However with a change in incidence or speed the shear layer may fail to reattach and the 'short bubble' may 'burst' to form either a 'long bubble', or an unattached, free shear layer. This change in mode of reattachment can occur gradually or quite sharply, depending on the type of aerofoil. The pressure distribution associated with a long bubble is quite different from that of inviscid flow (Figure 7.1) and the forces acting on the aerofoil are therefore modified, sometimes quite drastically, by the change in mode of reattachment. In particular, bubble bursting creates an increase in drag and an undesirable change in pitching moment. If a very large bubble is formed on bursting, or if the shear layer fails to reattach, there is also an appreciable fall in lift. (thin aerofoil stall). Thus short bubble fixes the maximum angle of attack possible for an aerofoil section for any given condition of Reynolds number and free-stream turbulence. It would be of obvious benefit to be able to predict the

development and bursting conditions of these separation bubbles, especially for the new series of aerofoils becoming available for general aviation application.

7.2 SHORT BUBBLE MODEL

The flow in the bubble region resembles a boundary layer in the sense that viscous effects are confined to a thin layer of fluid adjacent to a surface in motion relative to a free stream which is regarded as inviscid. However the viscous layer is not a boundary layer in the usual sense because, in the problem considered, the boundary-layer equations do not properly describe the separated portion of the viscous layer. (Ref. 6).

From a number of experimental investigations made by Gaster (Ref. 41) and others into the structure and behaviour of laminar separation bubbles Horton (Ref. 53) has developed a simple model (Figure 7.2) for the prediction of bubble growth and bursting. This model will be described briefly in this section.

The essential features of a laminar separation bubble are illustrated in Diagram 7.1.

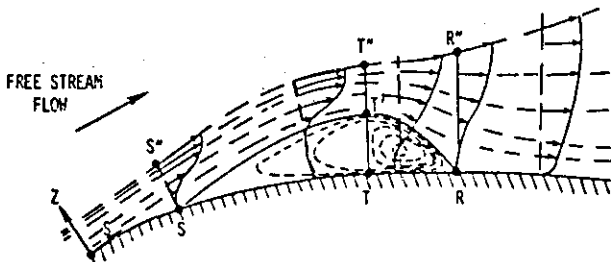


DIAGRAM 7.1: Section view of a two-dimensional short laminar separation bubble, z scale greatly exaggerated

The laminar boundary layer separates from the surface at S to reattach downstream at the point R. Between the points S and R the flow may be divided into two main regions:

1. the free shear layer contained between the outer edge S"TR" of the viscous region and the mean dividing streamline ST'R, and
2. the recirculation bubble contained between the mean dividing streamline and the aerofoil surface STR.

These two regions may then be further subdivided into parts upstream and downstream of the transition point T. Upstream of T, the free shear layer is laminar and is incapable of doing any significant diffusion, because weak viscous shear stresses operate in this region. Figure 7.1 a,b and c indicate that the surface velocity is practically constant between separation and transition. This constant pressure 'plateau' is a general feature of the laminar part of the separated flow.

Having described the main features of the bubble the equations for the calculation of its parameters are now listed below. The parameters involved are:

1. Bubble laminar length
2. Bubble turbulent length
3. Momentum thickness at reattachment
4. Shape factor at reattachment
5. Reattachment velocity
6. Bubble bursting length.

7.2.1 Laminar Part of Shear Layer

In this part of the bubble skin friction is exceedingly small and since the velocity is constant $d\theta/dx$ is zero (Ref. 53). Thus from the boundary layer momentum equation $\theta_T = \theta_S$. The laminar length (l_1) of the bubble is given by

$$\ell_1 = \theta_S \frac{4 \times 10^4}{Re_{\theta_S}}$$

$$Re_{\theta_S} = \frac{\theta_S U}{\nu}$$
(7.1)

Roberts (96) suggests that a better estimation of ℓ_1 is given by

$$\ell_1 = \theta_S (2.5 \times 10^4 \log_{10} (\coth(TF \times 10))) / Re_{\theta_S}$$
(7.2)

where TF is Taylor's turbulence factor defined by

$$TF = Tu' (C/L_S)^{1/5}$$

and θ_S is the momentum thickness at the point of separation of the laminar boundary layer.

7.2.2 Turbulent Part of Shear Layer

The turbulent length (ℓ_2) of the bubble and the velocity (U_R) at the point of reattachment are found from

$$\bar{U}_R^4 = (C_{d_m} / 4H_{\epsilon_m} + (1 - \bar{U}_R / \bar{\ell}_2)) / (C_{d_m} / 4H_{\epsilon_m} - \Lambda_r)$$
(7.3)

where

$$\bar{U}_R = \frac{U_R}{U_T} = \frac{U_R}{U_S}$$
(7.4)

$$\bar{\ell}_2 = \frac{\ell_2}{\theta_S}$$
(7.5)

$$\Lambda_r = \left(\frac{\theta}{U} \frac{dU}{dx} \right)_R = -0.0082$$
(7.6)

$$C_{d_m} = 0.0182$$
(7.7)

$$H_{\epsilon_m} = \frac{\epsilon}{\theta} = 1.5$$
(7.8)

$$H = \delta^* / \theta$$
(7.9)

$$\theta_R = \theta_S \left(\frac{1}{\bar{U}_R^3} + \frac{C_{d_m}}{4H_{\epsilon_m}} \frac{\bar{\ell}_2 (1 - \bar{U}_R^4)}{\bar{U}_R^3 (1 - \bar{U}_R)} \right)$$
(7.10)

A full derivation of the above equations can be found in Reference 53.

A graphical solution of equation 7.3 is shown in Diagram 7.2.

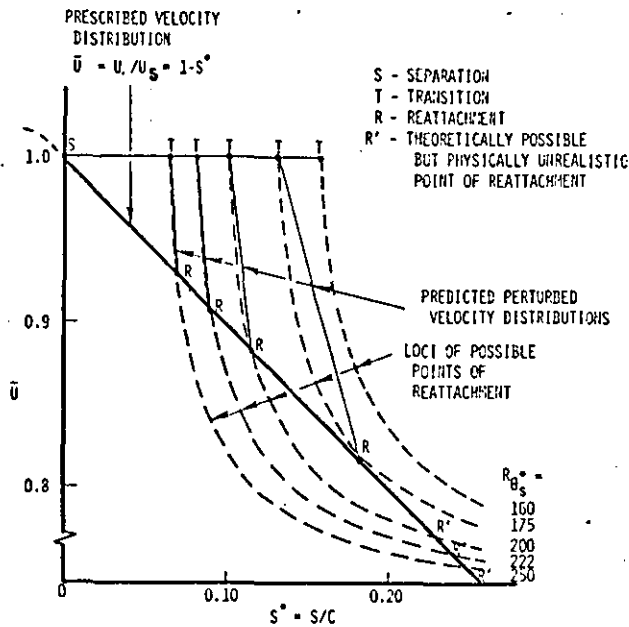


DIAGRAM 7.2: Example of graphical prediction of bubble growth and bursting.

7.2.3 Bubble Length at Bursting

The total length of the bubble (Figure 7.2) is given by

$$l = l_1 + l_2 \quad (7.11)$$

Using the non-dimensional length (\bar{l}) defined as l/θ_s equation (7.11) can be written as

$$\bar{l} = \bar{l}_1 + \bar{l}_2 \quad (7.12)$$

At bursting the non-dimensional length of the bubble may be approximated (for linear inviscid velocity distributions Ref. 53) by the curve,

$$l = 6 \times 10^4 / R e_{\theta_s} \quad (7.13)$$

7.3 METHOD OF SOLUTION

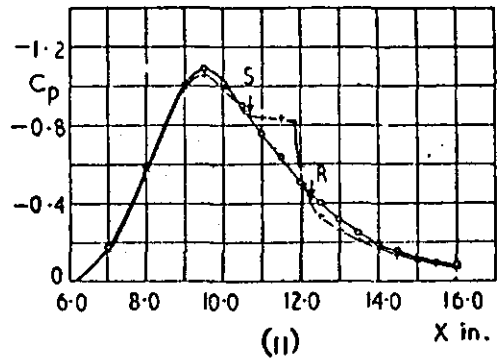
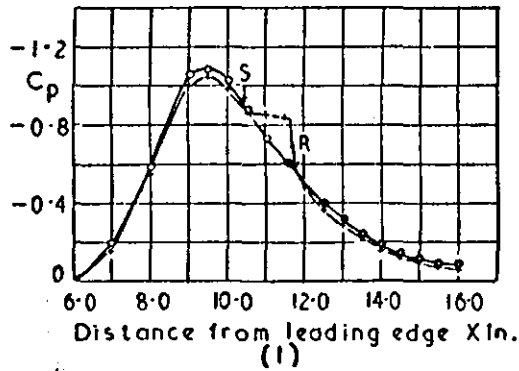
Using the value of momentum thickness (θ_s) at separation the laminar length of the bubble is calculated first from equation 7.12 taking $TF=0.01$ (Ref. 96). Equation 7.3 is then solved iteratively to find l_2 and \bar{U}_R . If no realistic solution to equation 7.3 is found then the bubble is said to have burst and its length at bursting is found from equation (7.13).

If a solution exists then the momentum thickness θ_R at reattachment is calculated next from equation (7.10) and is used as the initial value for the turbulent boundary layer analysis.

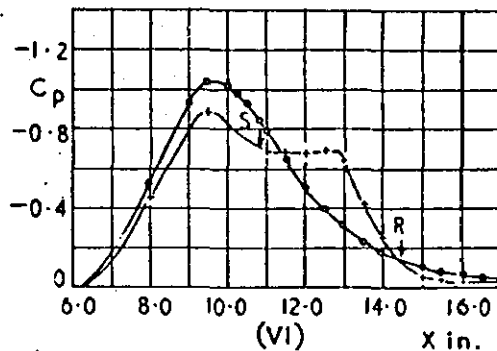
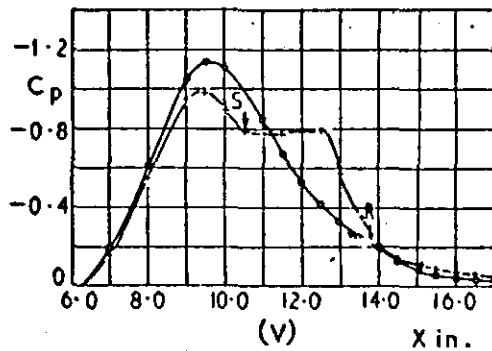
The method just described is based on Horton's short bubble model. The model is simple and easy to apply, however, since it is based on limited experimental results it has its limitations. Attempts to solve the bubble problem using the Navier-Stokes equations have been made (e.g. Ref. 6) but in such cases the bubble model is much more complex and becomes difficult to apply.

It can be argued that a bubble calculation may not be necessary since for short bubbles it is only required to know whether the bubble has burst. Thus only a criterion for short bubble bursting is required. If such an approach is used, however, it becomes difficult to estimate the extra thickening of the boundary layer due to the presence of the bubble. Thus a full bubble analysis is used in the present mathematical model of the viscous flow round an aerofoil.

Long bubble analysis is beyond the scope of the present work and the reader is referred to (64)



a) Short Bubble



b) Long Bubble

Figure 7.1 Pressure distributions.

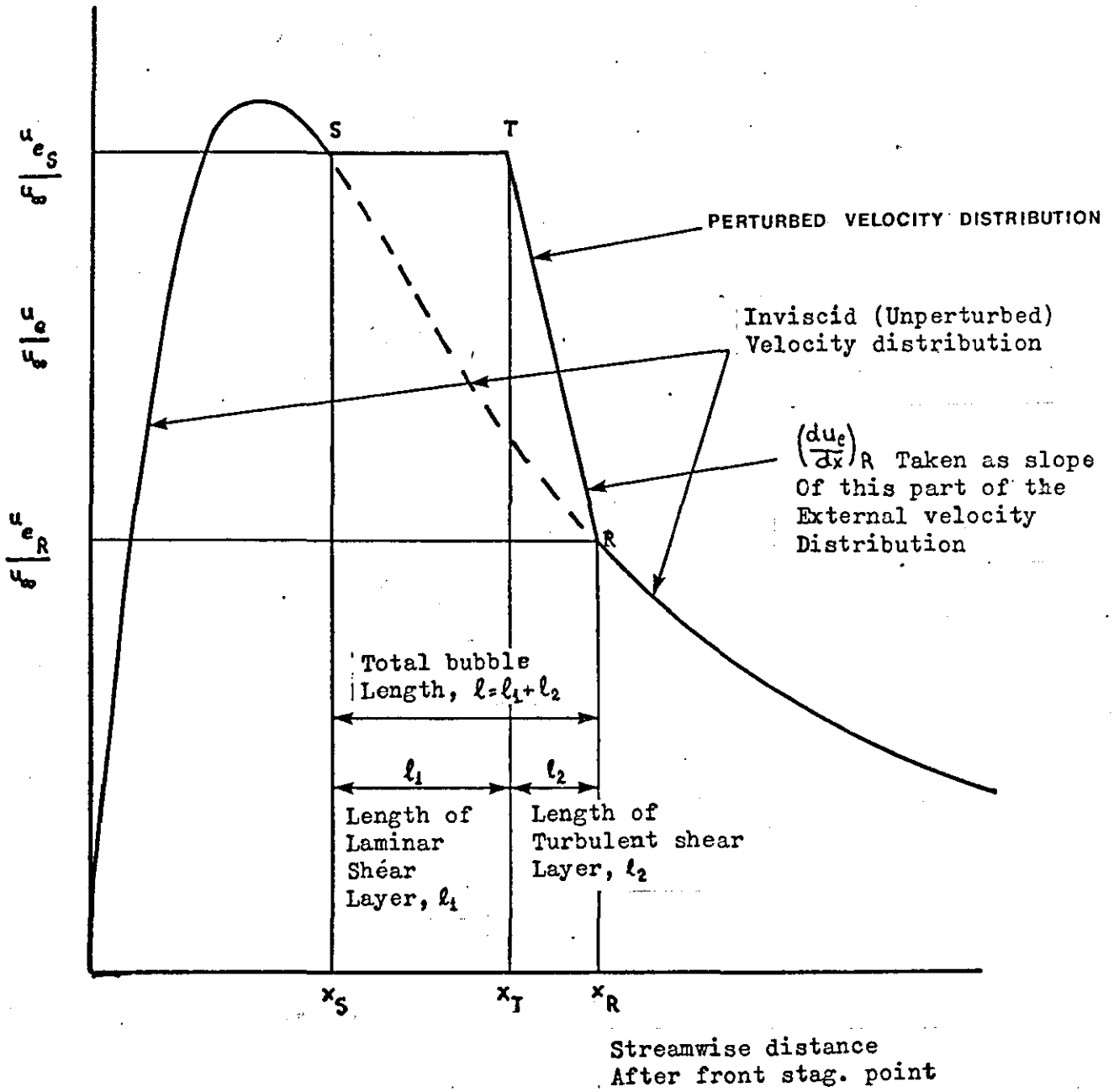


FIGURE 7.2: Simplified model of short laminar separation bubble

TURBULENT BOUNDARY LAYER

8.1 INTRODUCTION

The analysis of the turbulent boundary layer is based on integral methods, meaning that a system of ordinary differential equations derived from integral equations is employed.

The methods applied here make use of the momentum integral equation, which is usually regarded as the ordinary differential equation for θ , together with an entrainment function.

A brief description of the methods is given in this section, and ways for predicting turbulent separation are discussed.

8.2 HEAD'S ENTRAINMENT METHOD

Head's turbulent boundary layer method (Ref. 44,48) is an incompressible integral method based on the following two equations

$$\frac{d\theta}{dx} = \frac{C_F}{2} - \frac{\theta}{U} \frac{dU}{dx} \quad (H+2) \quad (8.1)$$

$$C_E = \frac{1}{U} \frac{dQ}{dx} = \frac{1}{U} \frac{d}{dx} (U(\delta - \delta^*)) = F(H) \quad (8.2)$$

Equation (8.1) is the momentum integral equation while equation (8.2) is the non-dimensional entrainment rate. The rate of entrainment $\frac{dQ}{dx}$ of fluid into the boundary layer is given by

$$\frac{dQ}{dx} = \frac{d}{dx} \int_0^{\delta} u \, dy \quad (8.3)$$

It is assumed that the turbulent boundary layer grows by a process of 'entrainment' of non-turbulent fluid at the outer edge of the layer into the turbulent region.

8.2.1 Equations

The following equations are required together with the momentum integral equation to calculate the values of H, C_F and θ .

$$\left. \begin{aligned} \frac{dH}{dx} &= -\frac{1}{\theta} \left[H(H^2 - 1) \frac{\theta}{U} \frac{dU}{dx} + \frac{H-1}{2} [(H-1)F - HC_F] \right] \\ F &= F(H) = 0.025H - 0.022 \\ C_F &= 0.246 \exp(-1.561H) R_\theta^{-0.268} \\ R_{e_\theta} &= \frac{U\theta}{\nu} \end{aligned} \right\} \quad (8.4)$$

8.2.2 Method of Calculation

Starting with the values of θ and H at transition or at the point of reattachment, in the case of a short bubble, a step by step approach is used as described below:

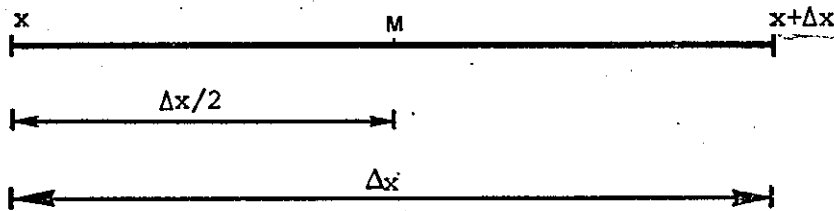


DIAGRAM 8.1

Assuming that θ and H are known at station x the values of θ and H will be calculated at station $x+\Delta x$. (Diagram 8.1). It is also assumed that U and $\frac{dU}{dx}$ are known at x and $x+\Delta x$.

Firstly R_{e_θ} is calculated, then the skin friction coefficient and the entrainment F are found at station x . Thus $\frac{dH}{dx}$ and $\frac{d\theta}{dx}$ are obtained at station x .

The value of θ and H at point M can now be found from

$$\left. \begin{aligned} \theta_{x+\Delta x/2} &= \theta_x + \left(\frac{d\theta}{dx}\right)_x \cdot \frac{\Delta x}{2} \\ H_{x+\Delta x/2} &= H_x + \left(\frac{dH}{dx}\right)_x \cdot \frac{\Delta x}{2} \end{aligned} \right\} \quad (8.5)$$

The values of R_{e_θ} , C_F , and F , $\frac{d\theta}{dx}$ and $\frac{dH}{dx}$ are calculated at $x + \frac{\Delta x}{2}$ as above. Finally using the new values of $\frac{d\theta}{dx}$ and $\frac{dH}{dx}$ the values of H and θ at $x + \Delta x$ are obtained from

$$\left. \begin{aligned} \theta_{x+\Delta x} &= \theta_x + \left(\frac{d\theta}{dx}\right)_{x+\frac{\Delta x}{2}} \cdot \Delta x \\ H_{x+\Delta x} &= H_x + \left(\frac{dH}{dx}\right)_{x+\frac{\Delta x}{2}} \cdot \Delta x \end{aligned} \right\} \quad (8.6)$$

Head's assumption that the entrainment is uniquely related to the boundary-layer form parameter H is rather a simplified one (Ref. 49) and it is perhaps surprising that it gives satisfactory results in quite a wide range of cases. The method however tends to fall down in cases of

1. Equilibrium flows, and
2. Flows where a strong pressure gradient is followed by a region of zero pressure gradient.

One of the main advantages of the method is its simplicity, thus it was used initially for the calculation of the turbulent boundary layer.

8.3 THE METHOD OF HEAD AND PATEL

This method (Ref. 49) still makes use of the entrainment idea, and its development has been based on the previous method.

Established relationships have been considered in order to determine the value of the entrainment coefficient for equilibrium layers, and the entrainment coefficient for non-equilibrium conditions is obtained by multiplying the corresponding equilibrium value ($C_{E_{eq}}$) by a suitable empirical function, thus

$$C_E = C_{E_{eq}} \times F(r_1) \quad (8.7)$$

This increases the entrainment when the rate of growth of the layer is less than that of the corresponding equilibrium layer, and decreases it, when the rate of growth is greater. This variation of entrainment is in accordance with observation from experiments.

8.3.1 Equations

The equations required for the calculation of the turbulent boundary layer are:

$$\frac{1}{U} \frac{d(U\theta)}{dx} = \frac{C_F}{2} - (H+1) \frac{\theta}{U} \frac{dU}{dx} \quad (8.8)$$

$$C_F = \exp(aH+b) \quad (8.9)$$

where

$$a = 0.019521 - 0.386786C + 0.028345C^2 - 0.0007010C^3$$

$$b = 0.191511 - 0.834891C + 0.062588C^2 - 0.001953C^3$$

$$C = \log_e R_{e_\theta}$$

$$\frac{dH^*}{dx} = \frac{H^*}{\theta} \left[\frac{1}{U} \frac{d(U\theta)}{dx} \right]_{eq} (F(r_1) - r_1) \quad (8.10)$$

$$\left[\frac{1}{U} \frac{d(U\theta)}{dx} \right]_{eq} = \frac{C_F}{2} \left[1 + \frac{H+1}{H} \pi \right] \quad (8.11)$$

$$\pi = \left[\frac{G+1.7}{6.1} \right]^2 - 1.81 \quad (8.12)$$

$$G = \frac{H-1}{H} \left(\frac{2}{C_F} \right)^{1/2} \quad (8.13)$$

$$r_1 = \frac{1}{U} \frac{d(U\theta)}{dx} / \left[\frac{1}{U} \frac{d(U\theta)}{dx} \right]_{eq} \quad (8.14)$$

$$F(r_1) = \frac{1}{2r_1 - 1} \quad \text{for } r_1 \geq 1 \quad (8.15)$$

$$F(r_1) = \frac{5-4r_1}{3-2r_1} \quad \text{for } r_1 < 1 \quad (8.16)$$

8.3.2 Method of Calculation

It is assumed that U and $\frac{dU}{dx}$ are known at all points on the aerofoil where θ and H are required. Then using the initial values of θ and H the calculation proceeds as follows.

The value of R_{e_θ} and therefore $\ln R_{e_\theta}$ is found at x (Diagram 8.2); using Table 8.1., and the values of H and $\log_e R_{e_\theta}$ a value of H^* is obtained at x .

The skin friction coefficient C_F is found next and the value of $\left[\frac{1}{U} \frac{d(U\theta)}{dx} \right]_{eq}$ is calculated from equation (8.11) using equations (8.12) and (8.13).

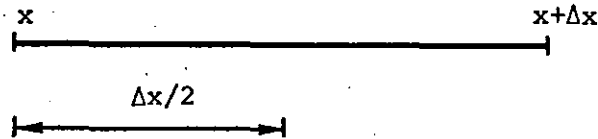


DIAGRAM 8.2

The value of $\frac{1}{U} \frac{d(U\theta)}{dx}$ at x is obtained from equation (8.8) thus r_1 and therefore $F(r_1)$ can be found. Finally from equation (8.10) $\frac{dH^*}{dx}$ can be calculated.

Applying equation (8.5) and (8.6)

$$H^*_{x+\frac{\Delta x}{2}} = H_x + \left(\frac{dH^*}{dx}\right)_x \cdot \frac{\Delta x}{2}$$

Similarly for $(U\theta)_{x+\frac{\Delta x}{2}}$. Then using Table 8.1 once more the value of H at $x+\frac{\Delta x}{2}$ is found. The calculation is repeated to find the values of H , and C_F at $x+\Delta x$.

8.4 INITIAL VALUES

The turbulent analysis is started assuming the value of the momentum thickness at the last laminar point is equal to the momentum thickness at the transition point. Thus a plot of θ will be smooth through transition while H, C_F , and δ^* will show a certain discontinuity as they go through transition.

The initial value of the turbulent shape factor is calculated using the expression (Ref. 8):

$$H = \frac{1.4754}{\log_{10} R_{e_\theta}} + 0.9698 \quad (8.17)$$

If a short bubble analysis has preceded the turbulent boundary layer analysis, the values of θ and H at the point of reattachment are used.

8.5 TURBULENT BOUNDARY LAYER SEPARATION

There exists a large number of methods of varying complexity and accuracy for predicting turbulent separation, (Ref. 12,97,111).

The aim of this work is to find a criterion which is simple to use and which predicts the point of separation with the reliability and accuracy needed for aerodynamic design purposes.

8.5.1 Head's Criterion

This method (Ref.12), like most integral methods, uses the shape factor H as the criterion for separation. Although it is not possible to give an exact value of H corresponding to separation, when H is between 1.8 and 2.4 separation is assumed to exist. The difference between the lower and upper limits of H makes very little difference in locating the separation point, since close to separation the shape factor increases quickly.

8.5.2 Epplers Method

In this case the shape factor $\tilde{H} = \frac{\delta^{**}}{\theta}$ is employed. The method (Ref.34) predicts separation if \tilde{H} goes below 1.46. However if \tilde{H} remains above 1.58 no separation takes place.

A simple relation (Ref.109) is used to relate \tilde{H} to the normal shape factor H .

$$\tilde{H} = \frac{\delta^{**}}{\theta} = \frac{1.269H}{H-0.379} \quad (8.18)$$

The above expression is deduced from experimental results.

Reference 8 indicates that for attached flow the shape factor \tilde{H} should be constrained to the range

$$1.55 \leq \tilde{H} \leq 1.85.$$

8.6 CONCLUDING REMARKS

The turbulent boundary layer methods used are simple and easy to programme and test. In view of the limited time available and the complexity of the overall model no attempt was made to programme any of the more complex methods available such as those of: Nash and Hicks (82), Nash and McDonald (83), McDonald and Stoddart (75), etc. However if required these methods can easily be put into the program.

Figures 8.1 to 8.4 give an idea of the accuracy to be expected from the method of Head and Patel.

TABLE 8.1

$$H(H^*, R_0)$$

$$2.5 \leq \log_{10} R_0 \leq 3.2$$

$\log_{10} R_0$	2.500	2.600	2.700	2.800	2.900	3.000	3.100	3.200
H^*								
3.600	0	0	0	0	0	0	0	0
3.650	0	0	0	0	0	0	0	0
3.700	0	0	0	0	0	0	0	0
3.750	0	0	0	0	0	0	0	0
3.800	0	0	0	0	0	0	0	0
3.850	0	0	0	0	0	0	2.795	2.635
3.900	0	0	0	0	2.800	2.640	2.545	2.460
3.950	0	0	2.800	2.620	2.515	2.450	2.380	2.330
4.000	0	2.655	2.540	2.450	2.375	2.320	2.270	2.235
4.100	2.500	2.400	2.330	2.280	2.230	2.180	2.115	2.085
4.200	2.340	2.275	2.220	2.170	2.125	2.090	2.055	2.025
4.300	2.260	2.195	2.135	2.090	2.050	2.020	1.985	1.955
4.400	2.185	2.130	2.085	2.040	2.005	1.965	1.940	1.915
4.600	2.085	2.030	1.985	1.950	1.915	1.880	1.855	1.830
4.800	2.010	1.960	1.910	1.880	1.845	1.810	1.780	1.750
5.000	1.960	1.905	1.865	1.825	1.795	1.765	1.725	1.700
5.500	1.885	1.795	1.750	1.715	1.675	1.650	1.625	1.600
6.000	1.840	1.735	1.695	1.655	1.625	1.590	1.565	1.540
6.500	1.750	1.700	1.655	1.620	1.585	1.555	1.525	1.495
7.000	1.725	1.670	1.625	1.590	1.555	1.525	1.490	1.470
8.000	1.675	1.625	1.580	1.540	1.505	1.480	1.455	1.435
9.000	1.635	1.580	1.535	1.495	1.460	1.435	1.405	1.385
1.000*1	1.590	1.535	1.485	1.450	1.420	1.395	1.365	1.345
1.100*1	1.545	1.485	1.445	1.415	1.380	1.355	1.335	1.320
1.200*1	1.505	1.455	1.415	1.385	1.355	1.330	1.305	1.285
1.300*1	1.470	1.425	1.390	1.355	1.325	1.295	1.270	1.250
1.400*1	1.430	1.390	1.350	1.320	1.290	1.270	1.245	1.230

TABLE 8.1 - continued

$$H(H^*, R_\theta)$$

$$3.3 \leq \log_{10} R_\theta \leq 4.0$$

$\log_{10} R_\theta$	3.300	3.400	3.500	3.600	3.700	3.800	3.900	4.000
3.600	0	0	0	0	0	0	0	0
3.650	0	0	0	0	0	0	0	0
3.700	0	0	0	0	0	0	2.850	2.605
3.750	0	0	2.835	2.680	2.590	2.510	2.450	2.405
3.800	2.795	2.665	2.565	2.480	2.425	2.370	2.330	2.290
3.850	2.555	2.480	2.415	2.360	2.315	2.265	2.235	2.200
3.900	2.395	2.340	2.290	2.250	2.215	2.185	2.150	2.120
3.950	2.290	2.250	2.210	2.180	2.145	2.115	2.090	2.070
4.000	2.195	2.165	2.135	2.105	2.085	2.060	2.045	2.025
4.100	2.060	2.035	2.015	1.995	1.975	1.955	1.940	1.910
4.200	2.000	1.975	1.955	1.935	1.915	1.900	1.875	1.870
4.300	1.935	1.915	1.890	1.875	1.860	1.845	1.835	1.820
4.400	1.890	1.865	1.850	1.830	1.815	1.800	1.790	1.775
4.600	1.805	1.785	1.765	1.745	1.735	1.720	1.705	1.700
4.800	1.725	1.700	1.670	1.660	1.650	1.635	1.630	1.620
5.000	1.675	1.655	1.635	1.620	1.610	1.600	1.585	1.580
5.500	1.575	1.560	1.540	1.525	1.510	1.500	1.485	1.480
6.000	1.520	1.500	1.480	1.465	1.455	1.440	1.435	1.420
6.500	1.475	1.455	1.440	1.425	1.415	1.405	1.395	1.385
7.000	1.445	1.430	1.415	1.400	1.385	1.375	1.370	1.360
8.000	1.415	1.395	1.380	1.365	1.350	1.340	1.330	1.325
9.000	1.365	1.350	1.340	1.325	1.315	1.305	1.295	1.290
1.000*1	1.330	1.320	1.310	1.295	1.285	1.280	1.270	1.265
1.000*1	1.305	1.290	1.280	1.270	1.260	1.250	1.245	1.240
1.200*1	1.270	1.255	1.245	1.235	1.230	1.225	1.220	1.215
1.300*1	1.235	1.220	1.210	1.205	1.195	1.190	1.190	1.185
1.400*1	1.215	1.205	1.190	1.185	1.180	1.175	1.170	1.170

TABLE 8.1 - continued

$$H(H^*, R_0)$$

$$4.2 \leq \log_{10} R_0 \leq 5.4$$

$\log_{10} R_0$	4.200	4.400	4.600	4.800	5.000	5.200	5.400
H^*							
3.600	0	0	0	0	0	2.764	2.640
3.650	0	2.665	2.535	2.455	2.405	2.365	2.330
3.700	2.490	2.415	2.350	2.305	2.270	2.245	2.230
3.750	2.330	2.270	2.225	2.19	2.170	2.150	2.140
3.800	2.230	2.180	2.145	2.120	2.095	2.080	2.070
3.850	2.145	2.110	2.075	2.050	2.035	2.025	2.015
3.900	2.075	2.040	2.015	1.995	1.980	1.975	1.970
3.950	2.025	1.995	1.975	1.955	1.845	1.935	1.930
4.000	1.985	1.960	1.935	1.920	1.905	1.895	1.890
4.100	1.885	1.865	1.850	1.840	1.835	1.835	1.830
4.200	1.850	1.830	1.815	1.805	1.795	1.790	1.785
4.300	1.800	1.785	1.775	1.765	1.760	1.750	1.745
4.400	1.760	1.745	1.730	1.720	1.715	1.710	1.705
4.600	1.680	1.670	1.660	1.655	1.650	1.650	1.645
4.800	1.610	1.605	1.600	1.595	1.595	1.590	1.590
5.000	1.565	1.555	1.550	1.545	1.540	1.540	1.540
5.500	1.465	1.460	1.450	1.450	1.450	1.445	1.445
6.000	1.410	1.400	1.400	1.400	1.395	1.395	1.395
6.500	1.370	1.365	1.360	1.355	1.355	1.355	1.355
7.000	1.345	1.335	1.330	1.325	1.325	1.320	1.320
8.000	1.310	1.305	1.300	1.295	1.290	1.285	1.285
9.000	1.280	1.270	1.260	1.260	1.260	1.255	1.255
1.000*1	1.250	1.245	1.240	1.235	1.230	1.230	1.230
1.100*1	1.230	1.225	1.220	1.215	1.215	1.215	1.215
1.200*1	1.210	1.205	1.205	1.200	1.195	1.195	1.195
1.300*1	1.185	1.185	1.185	1.185	1.185	1.180	1.180
1.400*1	1.170	1.170	1.170	1.170	1.170	1.170	1.170

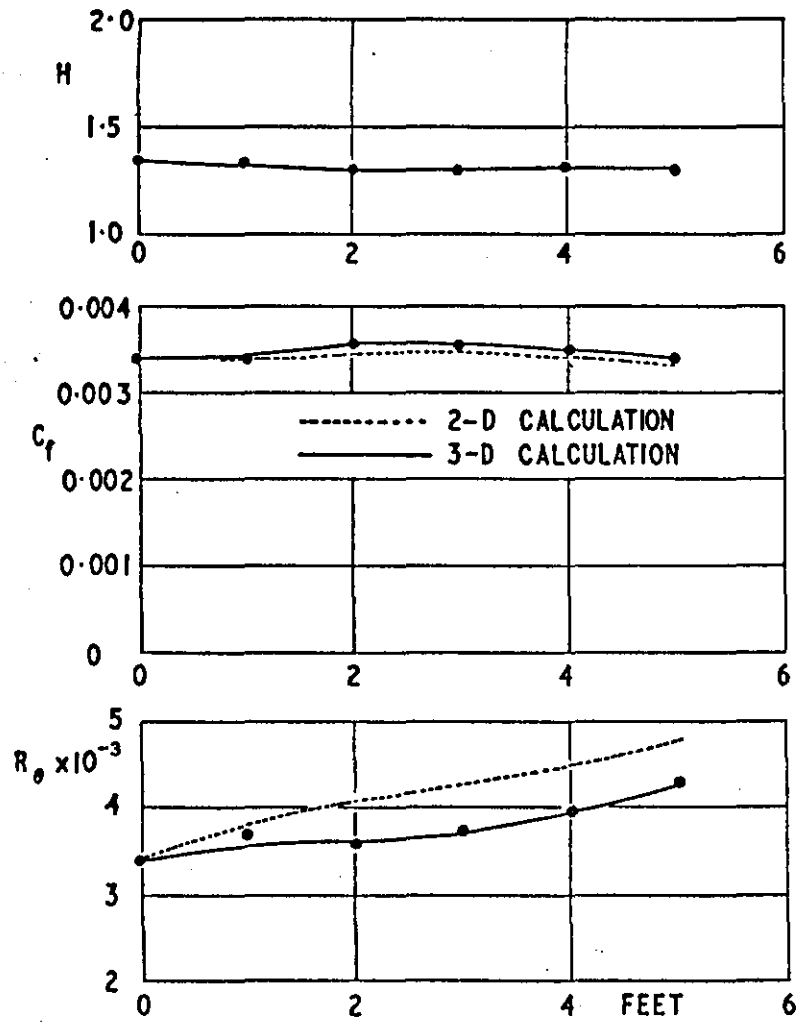


FIGURE 8.1: Equilibrium flow measured by Herring and Norbury ($\beta = -0.35$).

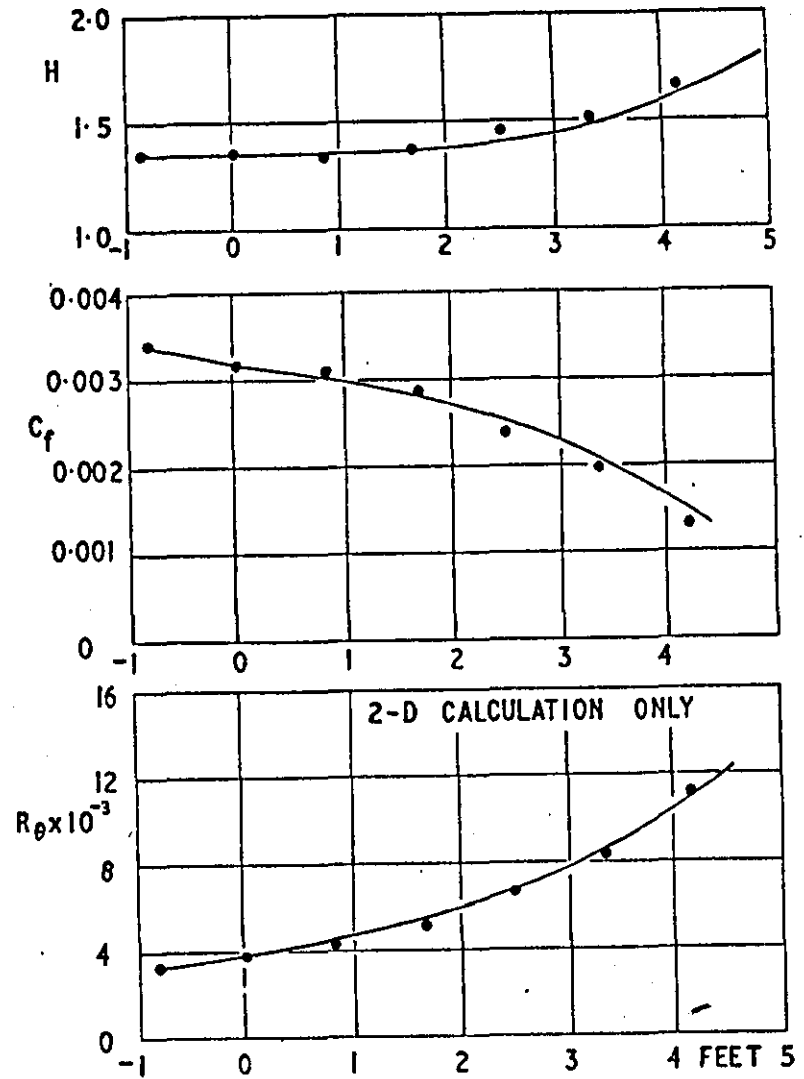


FIGURE 8.2: Measurements by Schubauer and Spangenberg (flow A).

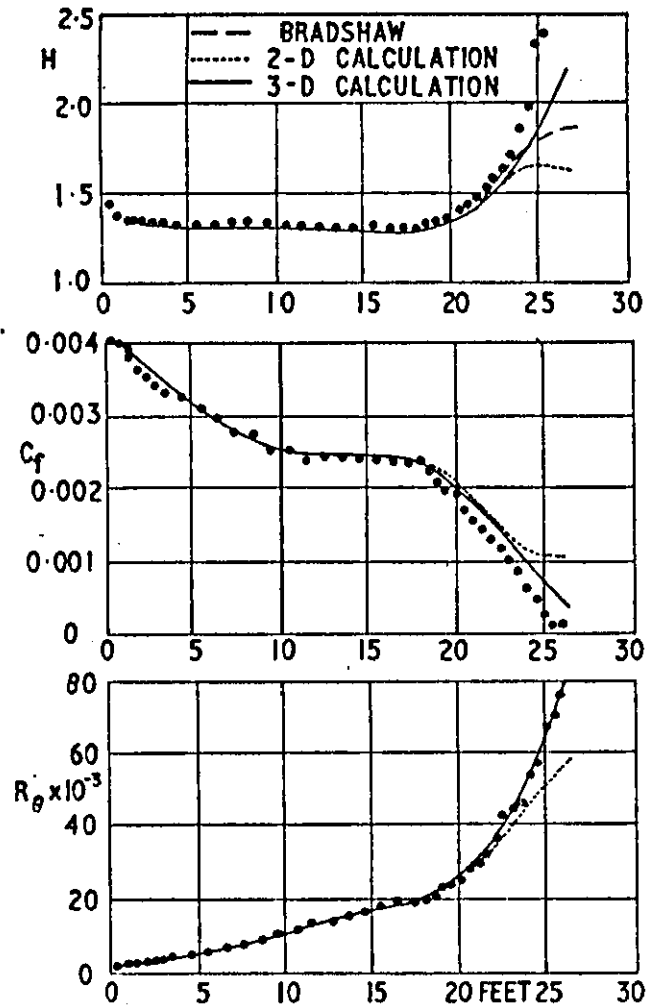


FIGURE 8.3: Measurements of Schubauer and Klebanoff.

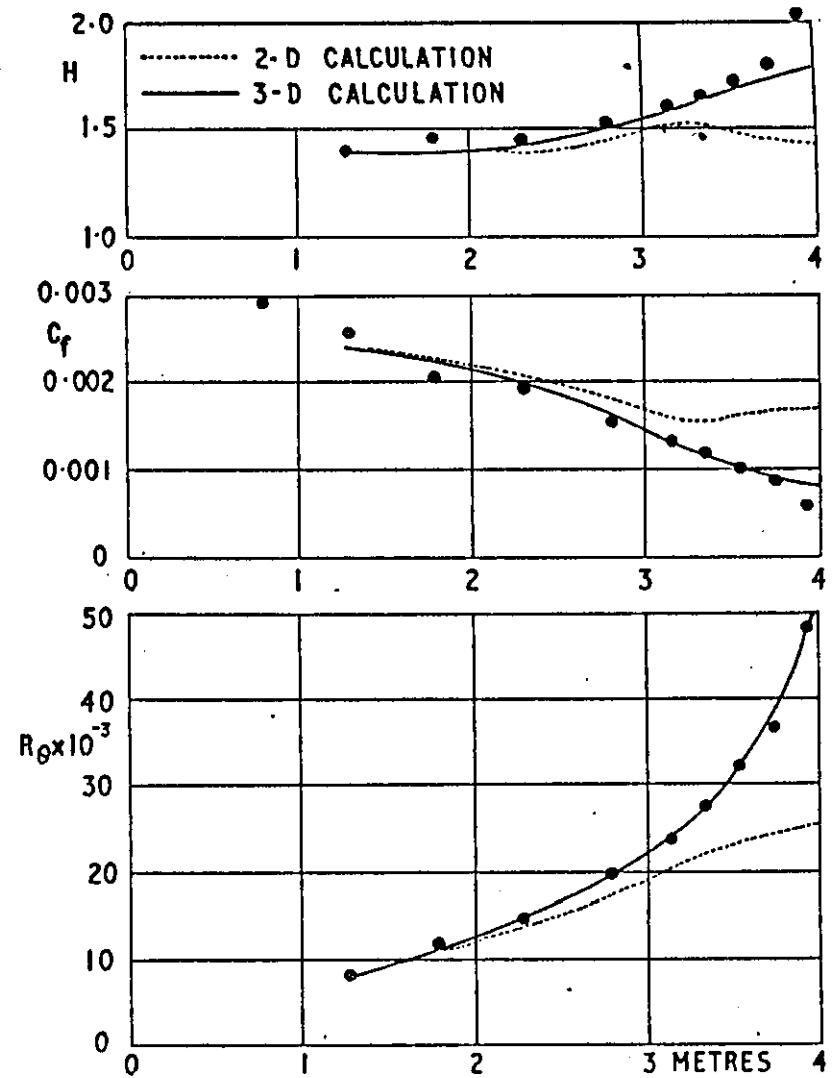


FIGURE 8.4: Measurements by Ludwig and Tillmann. (Strong adverse pressure gradient).

WAKE ANALYSIS

9.1 INTRODUCTION

On a lifting wing the streamlines in the region of the trailing edge must be inclined to the direction of the mainstream and thus the wake will be curved as it turns back into the mainstream direction. This curvature implies that there is a circulation in the viscous wake, i.e., a vorticity component across the mainstream, in addition to the streamwise vorticity component. This circulation must be taken into account in the calculation of lift. Since it is usually negative the actual lift is less when the wake is included than when it is ignored (65).

The boundary layer air that is left behind the body in the form of a wake, and the reduced momentum in the wake corresponds to a drag force on the body. Thus inclusion of the wake effects in the calculation of the lift and drag forces is necessary.

The wake analysis is divided into three main parts. The calculation begins by defining the initial position of the wake singularity sheet. A potential flow calculation follows. The aim of the analysis is to estimate the effect of the wake on the aerofoil potential flow and vice versa. A viscous flow calculation of the wake region is carried out next. Finally the wake is relaxed and the values of the wake singularities are calculated for the next iteration.

9.2 POTENTIAL FLOW WAKE MODEL

The wake singularity sheet is that potential flow which is attached to the average trailing edge point of an aerofoil component. The singularity

model of the aerofoil component is in fact extended onto the wake which can be considered as two layers; one being the development of the upper surface boundary layer and the other the development of the lower surface boundary layer. (Figure 9.1).

As with the aerofoil the wake singularity sheet is divided into panels. A continuous distribution of vortices and sources is placed on each panel; both varying linearly between corner points and being continuous along the corner. Control points are chosen midway between the panel ends. In theory the last segment of the wake must be semi-infinite to carry the wake into the far field.

9.3 WAKE SINGULARITY SHEET

Since the position of the wake singularity sheet is not known at the start of the analysis, an initial approximate position has to be assumed. This position is updated during each iteration through a wake relaxation. The geometry of the wake singularity sheets for a two element aerofoil is shown in Figure 9.2. The wake of the main component is assumed to be parallel to the flap upper surface. The distance is

$$h_{\text{slot}} + \frac{1}{2} t_{\text{TE}}$$

where the symbols h_{slot} and t_{TE} stand for the slot height at the upstream slot exit and the trailing edge thickness of the upstream aerofoil, respectively. The corner points (end points) of the wake panels are chosen by shifting the corresponding flap surface points along the surface normal. The procedure gives a wake centerline which extends from the slot exit to the trailing edge of the flap.

The initial position and the total length of the flap wake singularity sheet are selected by extending the chord length of the flap by 100%. All segments representing this wake singularity sheet are of equal length.

The calculation of the initial wake geometry is fully described in Appendix 11.

9.4 VELOCITY EQUATIONS

The normal and tangential velocity components induced at the control point of each wake segment, due to the contribution of

1. Aerofoil singularities
2. Wake singularities
3. Free stream velocity

are calculated from the following set of equations (25). The equations are given in matrix form

$$[V_N] = \frac{1}{2\pi} ([GN_{wA}] [\gamma_A] + \sum_{i=1}^{N_w} [GN_{wv}]_i [\gamma_v]_i + [GN_{ws}]_i [\gamma_s]_i) - [V_\infty \sin(\epsilon - \alpha)] \quad (9.1)$$

$$[V_T] = \frac{1}{2\pi} ([GT_{wA}] [\gamma_A] + \sum_{i=1}^{N_w} [GT_{wv}]_i [\gamma_v]_i + [GT_{ws}]_i [\gamma_s]_i) - [V_\infty \cos(\epsilon - \alpha)] \quad (9.2)$$

N_w = number of wakes

$$[\gamma_A] = 2\pi [GN_{AA}]^{-1} \left(\left[\frac{dU\delta^*}{ds} \right]_A + [V_\infty \sin(\theta_A - \alpha)] \right) - \sum_{i=1}^{N_w} ([A_1]_i [\gamma_v]_i + [A_2]_i [\gamma_s]_i) \quad (9.3)$$

$$\left. \begin{aligned} [A_1]_i &= [GN_{AA}]^{-1} [GN_{AV}]_i \\ [A_2]_i &= [GN_{AA}]^{-1} [GT_{AV}]_i \end{aligned} \right\} \quad (9.4)$$

The calculation of γ_v and γ_s is described in a later paragraph.

The mean wake velocity magnitude is

$$V_m^2 = V_N^2 + V_T^2 \quad (9.5)$$

The upper and lower wake velocities are given by (25):

$$\left. \begin{aligned} V_u^2 &= \left(V_m + \frac{\gamma_v}{2} \right)^2 + \left(\frac{\gamma_s}{2} \right)^2 \\ V_l^2 &= \left(V_m - \frac{\gamma_v}{2} \right)^2 + \left(\frac{\gamma_s}{2} \right)^2 \end{aligned} \right\} \quad (9.6)$$

or within the assumption of the normal velocity contribution being small in relation to the tangential velocity

$$\left. \begin{aligned} V_u &= V_m + \frac{\gamma_v}{2} \\ V_l &= V_m + \frac{\gamma_v}{2} \end{aligned} \right\} \quad (9.7)$$

9.5 WAKE VISCOUS FLOW ANALYSIS

The viscous flow in the wake behind each aerofoil component is calculated using Green's lag entrainment method (Ref. 8,45).

The method is an integral method formulated in terms of the momentum thickness θ , the shape factor H and the entrainment coefficient C_E where

$$\theta = \int_0^{\delta} \frac{u}{U_w} (1 - \frac{u}{U_w}) dy \quad (9.8)$$

$$H = \frac{\delta^*}{\theta}, \quad \delta^* = \int_0^{\delta} (1 - \frac{u}{U_w}) dy \quad (9.9)$$

$$C_E = \frac{1}{U_w} \frac{d}{dx} (\int_0^{\delta} u dy) \quad (9.10)$$

δ is the distance from the wake singularity sheet to the outer edge of the wake, y is measured perpendicular to the wake singularity sheet and U_w is the velocity at the wake outer edge (Figure 9.1).

Note that the method has to be applied twice at each point to give the total thickness of the wake.

9.5.1 Equations

The following equations are used for the calculation of the wake viscous parameters on one side of the wake singularity sheet.

1. Momentum integral equation

$$\frac{d\theta}{dx} = -(H+2) \frac{\theta}{U_w} \frac{dU_w}{dx} \quad (9.11)$$

2. Entrainment equation

$$\frac{dH}{dx} = \frac{1}{\theta} \frac{dH}{dH_1} (C_E + H_1 (H+1) \frac{\theta}{U_w} \frac{dU_w}{dx}) \quad (9.12)$$

3. Lag equation

$$\frac{dC_E}{dx} = \frac{F}{\theta} \left[\frac{2.8}{H+H_1} ((C_{\tau})_{EQ_0}^{\frac{1}{2}} - \tau (C_{\tau})^{\frac{1}{2}}) + \left(\frac{\theta}{U_w} \frac{dU_w}{dx} \right)_{EQ} - \frac{\theta}{U_w} \frac{dU_w}{dx} \right] \quad (9.13)$$

The lag equation is an empirical equation which takes into account the influence of the upstream flow history on the turbulent stresses.

The various terms in the above three equations are given by the following empirical formulae (8):

$$H_1 = \frac{\delta - \delta^*}{\theta} = 3.15 + \frac{1.72}{H-1} - 0.01(H-1)^2 \quad (9.14)$$

$$\frac{dH}{dH_1} = - \frac{(H-1)^2}{1.72 + 0.02(H-1)^3} \quad (9.15)$$

$$\left. \begin{aligned} C_\tau &= 0.024C_E + 1.2C_E^2 \\ F &= \frac{C_\tau}{0.012 + 1.2C_E} \\ \lambda &= 0.5 \end{aligned} \right\} \quad (9.16)$$

$$\left(\frac{\theta}{U_w} \frac{dU_w}{dx} \right)_{EQ_0} = - \frac{1.25}{H} \left(\frac{H-1}{6.432H} \right)^2 \quad (9.17)$$

$$(C_E)_{EQ_0} = -H_1(H+1) \left(\frac{\theta}{U_w} \frac{dU_w}{dx} \right)_{EQ_0} \quad (9.18)$$

$$(C_\tau)_{EQ_0} = 0.024(C_E)_{EQ_0} + 1.2(C_E)_{EQ_0}^2 \quad (9.19)$$

$$(C_E)_{EQ} = \left(\frac{(C_\tau)_{EQ_0}}{1-2\lambda^2} + 0.0001 \right)^{1/2} - 0.01 \quad (9.20)$$

$$\left(\frac{\theta}{U_w} \frac{dU_w}{dx} \right)_{EQ} = - \frac{(C_E)_{EQ}}{H_1(H+1)} \quad (9.21)$$

For the upper outer edge (25)

$$U_w = U_u$$

and for the lower outer edge (25)

$$U_w = U_l$$

(see Section 9.4).

The total displacement thickness of the wake is given by

$$\delta_w^* = \delta_u^* + \delta_l^* \quad (9.22)$$

In addition the distance from the wake singularity sheet to the lower edge of the wake denoted by the symbol δ_l is needed to predict the end of the core region

$$\delta_l = \theta_l(H_1)_l + \delta_l^* \quad (9.23)$$

9.5.2 Initial Conditions

The initial values of θ and H for the upper and lower parts of the wake are provided by the corresponding parts of the boundary layer, at the trailing edge of the aerofoil. An initial value of the entrainment coefficient C_E is assumed to be given by its equilibrium value $(C_E)_{EQ}$ which can be calculated using equations (9.14), (9.17), (9.18), (9.19) and (9.20) together with the initial values of θ , H and U_w .

9.5.3 Calculation Procedure

The wake flow is calculated on both sides of the wake singularity sheet solving the equation for $\frac{d\theta}{dx}$, $\frac{dH}{dx}$ and $\frac{dC_E}{dx}$ in marching fashion beginning at the trailing edge of the upstream aerofoil. At each point of the wake centerline, the wake parameters of both sides are calculated before the integration procedure advances to the next point. The values of $\frac{d\theta}{dx}$, $\frac{dH}{dx}$ and $\frac{dC_E}{dx}$ at station x are used to calculate θ, H and C_E at station $x+\Delta x$, using

$$\left. \begin{aligned} \theta_{x+\frac{\Delta x}{2}} &= \theta_x + \left(\frac{d\theta}{dx}\right)_x \frac{\Delta x}{2}, \text{ and} \\ \theta_{x+\Delta x} &= \theta_x + \left(\frac{d\theta}{dx}\right)_{x+\frac{\Delta x}{2}} \Delta x \end{aligned} \right\} \quad (9.24)$$

and similarly for H and C_E .

9.6 CALCULATION OF WAKE SINGULARITIES

It is shown in Appendix 10 that the incremental velocities imposed on the outer inviscid flow at the upper and lower edge of the wake are:

$$\Delta V_u = -V_u (\delta_u^{*+} + \theta_u) \left(\frac{d\epsilon}{ds} + \frac{d^2 \delta_u^*}{ds^2} \right) + \frac{1}{\rho} \frac{d(\rho V_u \delta_u^*)}{ds} \quad (9.25)$$

$$\Delta V_l = V_l (\delta_l^{*+} + \theta_l) \left(\frac{d\epsilon}{ds} - \frac{d^2 \delta_l^*}{ds^2} \right) - \frac{1}{\rho} \frac{d(\rho V_l \delta_l^*)}{ds} \quad (9.26)$$

where $\frac{d\epsilon}{ds}$ is the local curvature of the wake singularity sheet.

Consider the velocity vector at the mid point of the wake segments in terms of two contributions; one being due to the singularity strengths γ_v, γ_s of the particular segment considered and the other the contribution of all other singularities both in the wakes and on the aerofoil elements and also including the free stream. If \underline{v}_m is the average induced velocity vector across the singularity sheet, and $\frac{\gamma_v}{2} \underline{t}$ and $\frac{\gamma_s}{2} \underline{t}$ are the velocity vectors due to the local singularities, then the velocities on the upper and lower sides of the sheet respectively are (25):

$$\underline{v}_u = \underline{v}_m + \frac{\gamma_v}{2} \underline{t} + i \frac{\gamma_s}{2} \underline{t} \quad (9.27)$$

$$\underline{v}_l = \underline{v}_m - \frac{\gamma_v}{2} \underline{t} - i \frac{\gamma_s}{2} \underline{t} \quad (9.28)$$

Note that the sense of the local singularity contribution depends on which side the singularity sheet is approached.

From reference 68 the velocity jump across the wake is equal to $\underline{v}_u - \underline{v}_l$ which is also equal to $\Delta \underline{v}_u - \Delta \underline{v}_l$. Thus from equations (9.25) to (9.28), the values of the wake singularities, at the panel midpoint, are given in terms of the wake viscous parameters by

$$\gamma_v = -v_u (\delta_u^* + \theta_u) \left(\frac{d\epsilon}{ds} + \frac{d^2 \delta_u^*}{ds^2} \right) - v_l (\delta_l^* + \theta_l) \left(\frac{d\epsilon}{ds} - \frac{d^2 \delta_l^*}{ds^2} \right) \quad (9.29)$$

and

$$\gamma_s = \frac{d(v_u \delta_u^*)}{ds} + \frac{d(v_l \delta_l^*)}{ds} \quad (9.30)$$

9.7 WAKE RELAXATION

Since the final position of the wake singularity sheet is not known at the start of the analysis its shape has to be determined iteratively using a relaxation method.

By definition the direction of the wake singularity sheet is in the local streamline direction and hence

$$\begin{aligned} \underline{v}_{\text{mean}} &= \underline{v}_T \\ \underline{v}_N &= 0 \end{aligned}$$

If the calculation indicates a non-zero normal velocity (V_N) the wake singularity sheet needs to be repositioned. This is done by rotating each segment about its upstream end point to realign it with the local flow while the downstream segments are translated to maintain a continuous sheet.

The coordinates of the repositioned wake segment at its downstream end are

$$x'_N = x_N + \sum_{k=1}^N \Delta S_k \left[\left(\frac{V_{T_k}}{A} - 1 \right) \cos \epsilon_k - \frac{RV_{N_k}}{A} \sin \epsilon_k \right] \quad (9.31)$$

and

$$z'_N = z_N + \sum_{k=1}^N \Delta S_k \left[\left(\frac{V_{T_k}}{A} - 1 \right) \sin \epsilon_k + \frac{RV_{N_k}}{A} \cos \epsilon_k \right] \quad (9.32)$$

where

$$A = \sqrt{V_{T_k}^2 + R^2 V_{N_k}^2}$$

where N indicates the wake segment which is relaxed with the first segment being closest to the aerofoil; x_N and z_N are the coordinates of the segment downstream end point before relaxation, x'_N, z'_N are the coordinates after relaxation, ΔS_k is the length of the kth wake segment, $\tan \epsilon_k$ is the slope of the unrelaxed kth segment and V_{N_k}, V_{T_k} the normal and tangential velocities at the midpoint of the kth segment (Figure 9.3). R is a relaxation factor (25) where $0 < R < 1$. ($R=1$ for the present analysis).

Each time the wake is relaxed it will be necessary to recalculate the influence coefficients involving the wake. This however does not greatly increase the computational time of the method which is dominated by the inversion of the matrix $[G_{AA}^N]$ which is still computed once only.

The wake iteration is terminated once the values of V_N are nearly zero. However to avoid excessive computation time, resulting from either slow convergence or from a non-converging wake, the maximum number of iterations is fixed within the program. If this number is exceeded the program stops and prints the current values of V_N so that the user may decide whether to continue the iteration or proceed with the other stages of the analysis.

9.8 END OF POTENTIAL CORE

The method of Reference 8 is used to predict whether interaction between the wake flow of the main aerofoil and the boundary layer on the upper surface of the flap has taken place. In the case of a flow interaction the method calculates where the resulting potential core ends.

The physical boundaries of the potential core region are shown in Figure 9.4a. The end of the core is defined by the normal to the upper surface of the flap which passes through the point of intersection of wake and boundary layer edges. This definition is consistent with the aerodynamic model of the confluent boundary layer for which initial values must be provided along the same surface normal.

The notation used in determining the end of the core is illustrated in Figure 9.4b. It is assumed that properties of the boundary layer on the flap upper surface are known. The wake flow calculation proceeds in marching fashion along the wake singularity sheet. At each step of the calculation it is checked whether or not the end of the core region has been reached. Knowing the wake properties at station S_i , which include the half width of the wake δ_{li} , the following calculations are performed.

The distance $\overline{S_i S_F}$ along the normal to the flap surface is first determined. (Appendix 11). Then the distance d along the surface normal measured from the point S_i to the edge of the wake is obtained from

$$d = \frac{\Delta S \delta_{li}}{\Delta S \cos \gamma + (\delta_{li} - \delta_{li-1}) \sin \gamma} \quad (9.33)$$

where ΔS is the arc length between the points S_i and S_{i-1} on the wake singularity sheet, γ is the angle formed by the normal to the wake singularity sheet at point S_i and the surface normal of the flap. (Figure 9.4b).

The end of the core region has been reached if

$$d + \delta_{BL} \geq |\overline{S_i S_F}| \quad (9.34)$$

where δ_{BL} is the thickness of the boundary layer at the point S_F .

9.9 AEROFOIL VELOCITIES

The normal and tangential velocities on the aerofoil surface due to aerofoil singularities, wake singularities and free stream are:

$$[V_N] = \frac{1}{2\pi} ([GN_{AA}] [\gamma_A] + \sum_{i=1}^{N_w} ([GN_{AV}]_i [\gamma_V]_i + [GN_{AS}]_i [\gamma_S]_i)) - [V_\infty \sin(\theta_A - \alpha)] \quad (9.35)$$

$$[V_T] = \frac{1}{2\pi} ([GT_{AA}] [\gamma_A] + \sum_{i=1}^{N_w} ([GT_{AV}]_i [\gamma_V]_i + [GT_{AS}]_i [\gamma_S]_i)) + [V_\infty \cos(\theta_A - \alpha)] \quad (9.36)$$

Equations (9.35) and (9.36) can now be solved for a normal velocity on the aerofoil surface due to the boundary layer (Equation 4.5) while making use of equations (A7.24), defined in Appendix 7, to eliminate the wake singularities and source influence coefficients. Taking the tangential velocity as the viscid solution about the aerofoil, the aerofoil surface velocity is obtained as a summation of the inviscid velocity V_I , a contribution due to the boundary layer V_{BL} , and a contribution due to the wake(s) V_w . Thus upon eliminating γ_A between equations (9.35) and (9.36)

$$[V] = [V_I] + [V_{BL}] + [V_{w_1}] + [V_{w_2}] \quad (9.37)$$

where

$$[V_I] = [A] [V_\infty \sin(\theta_A - \alpha)] + [V_\infty \cos(\theta_A - \alpha)]$$

$$[V_{BL}] = [A] \left[\frac{dU\delta^*}{ds} \right]$$

$$[V_{w_1}] = \frac{1}{2\pi} ([A_{11}] [\gamma_{V_1}] - [A_{12}] [\gamma_{S_1}])$$

$$[V_{w_2}] = \frac{1}{2\pi} ([A_{21}] [\gamma_{V_2}] - [A_{22}] [\gamma_{S_2}])$$

and

$$[A] = [GT_{AA}] [GN_{AA}]^{-1}$$

$$[A_{11}] = [GT_{AV_1}] - [A] [GN_{AV_1}]$$

$$[A_{12}] = [GN_{AV_1}] - [A] [GT_{AV_1}]$$

$$[A_{21}] = [GT_{AV_2}] - [A] [GN_{AV_2}]$$

$$[A_{22}] = [GN_{AV_2}] - [A] [GT_{AV_2}]$$

$\gamma_{v_1}, \gamma_{v_2}, \gamma_{s_1}, \gamma_{s_2}$ are given by equations (9.29) and (9.30).

The pressure coefficient C_{P_i} at the i th control point is given by

$$C_{P_i} = 1 - \frac{V_i}{V_\infty}$$

where

$$V_i = V_{I_i} + V_{BL_i} + V_{w_{1i}} + V_{w_{2i}}$$

(see equation (9.37)).

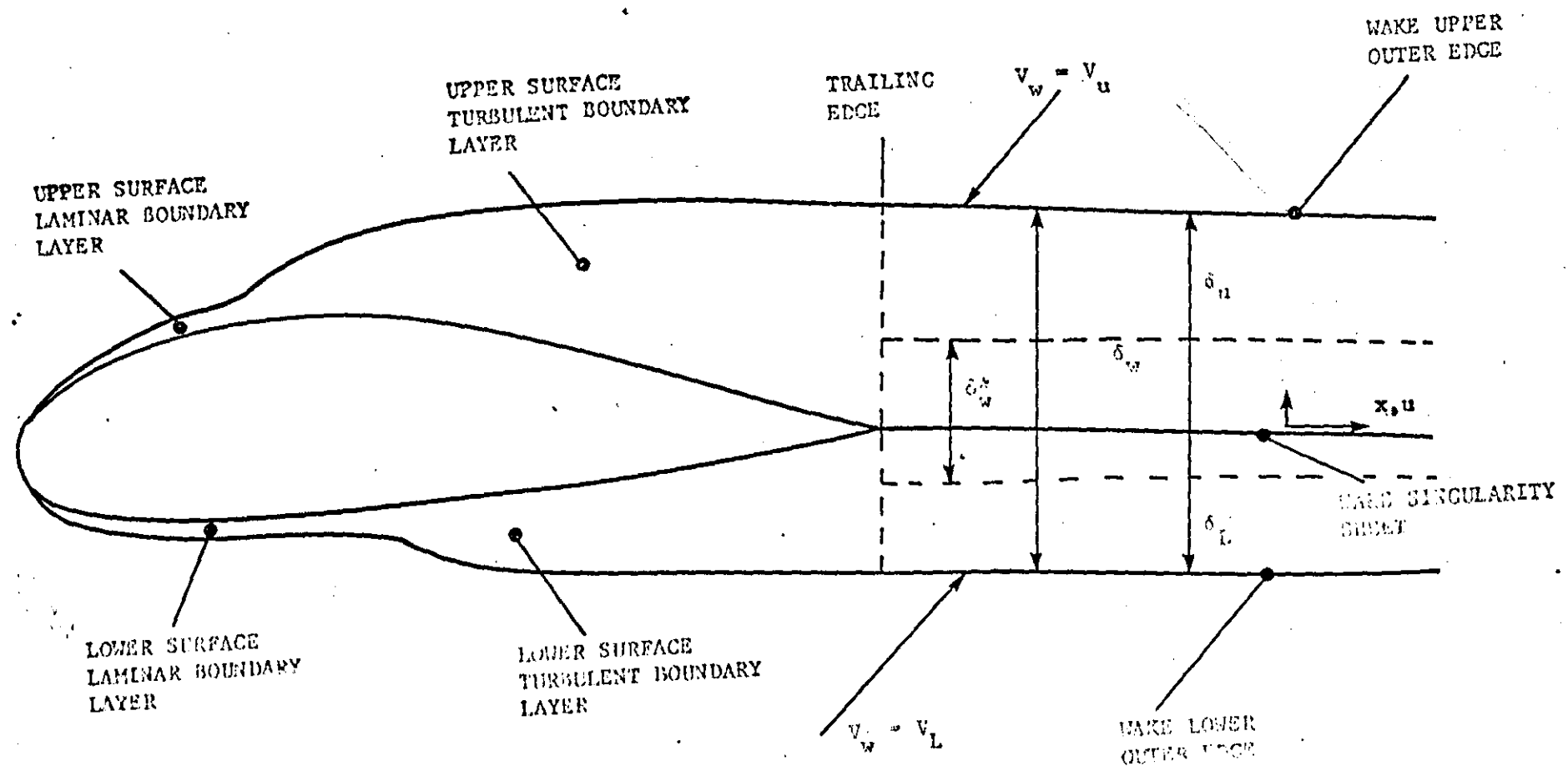


FIGURE 9.1: Wake Viscous Flow Parameters

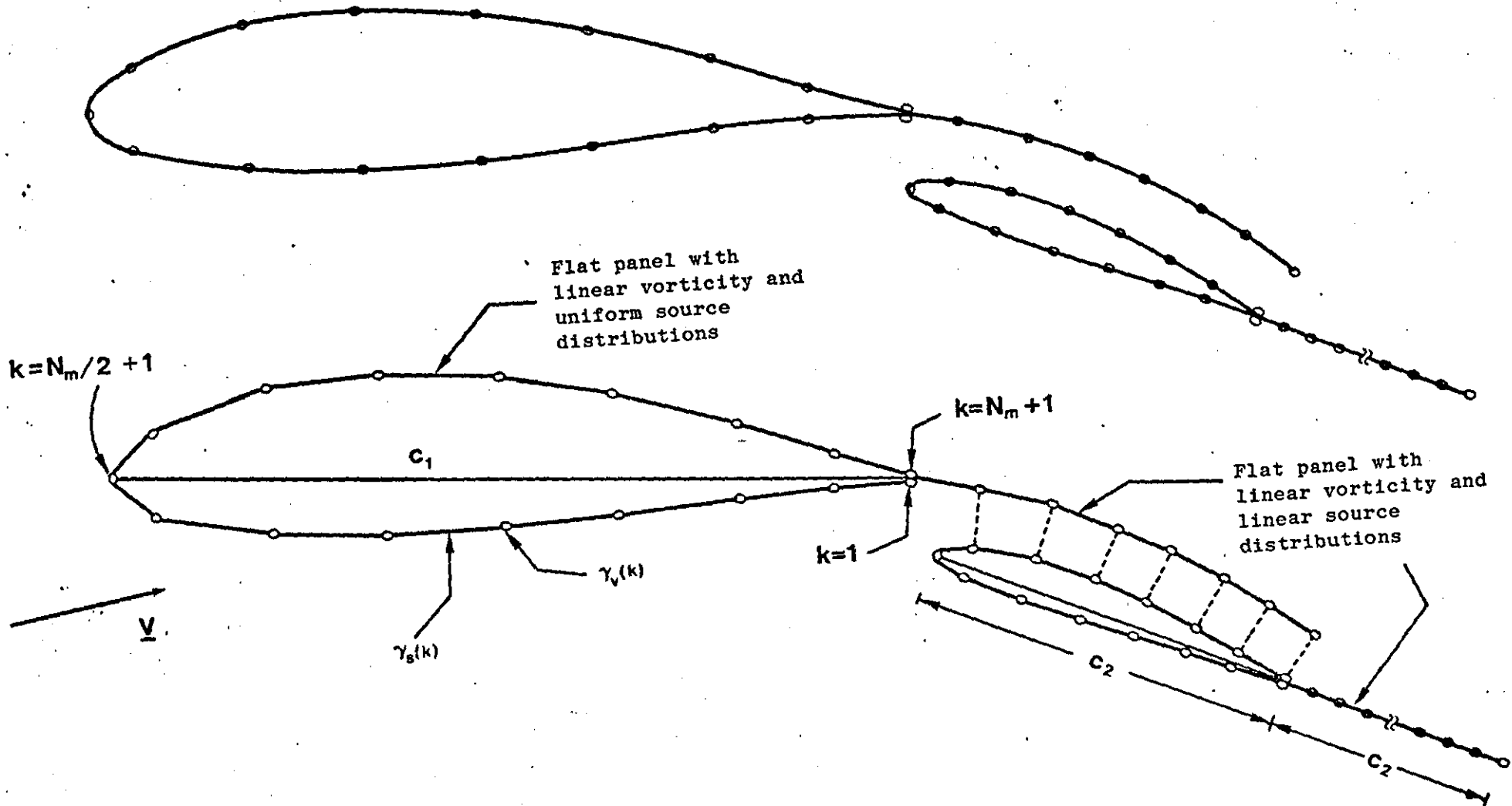


FIGURE 9.2 Initial Wake Geometry

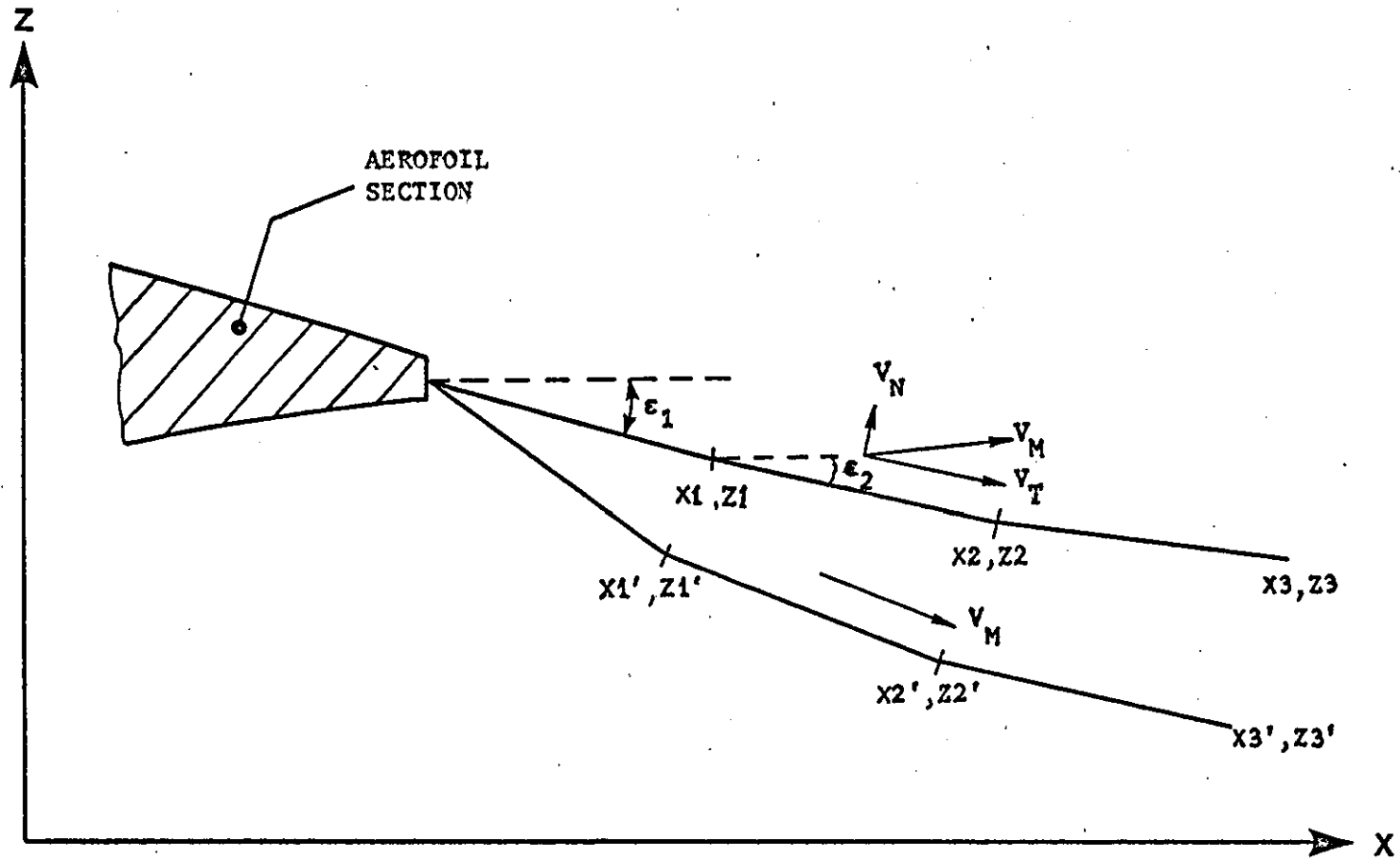


FIGURE 9.3: Wake Relaxation

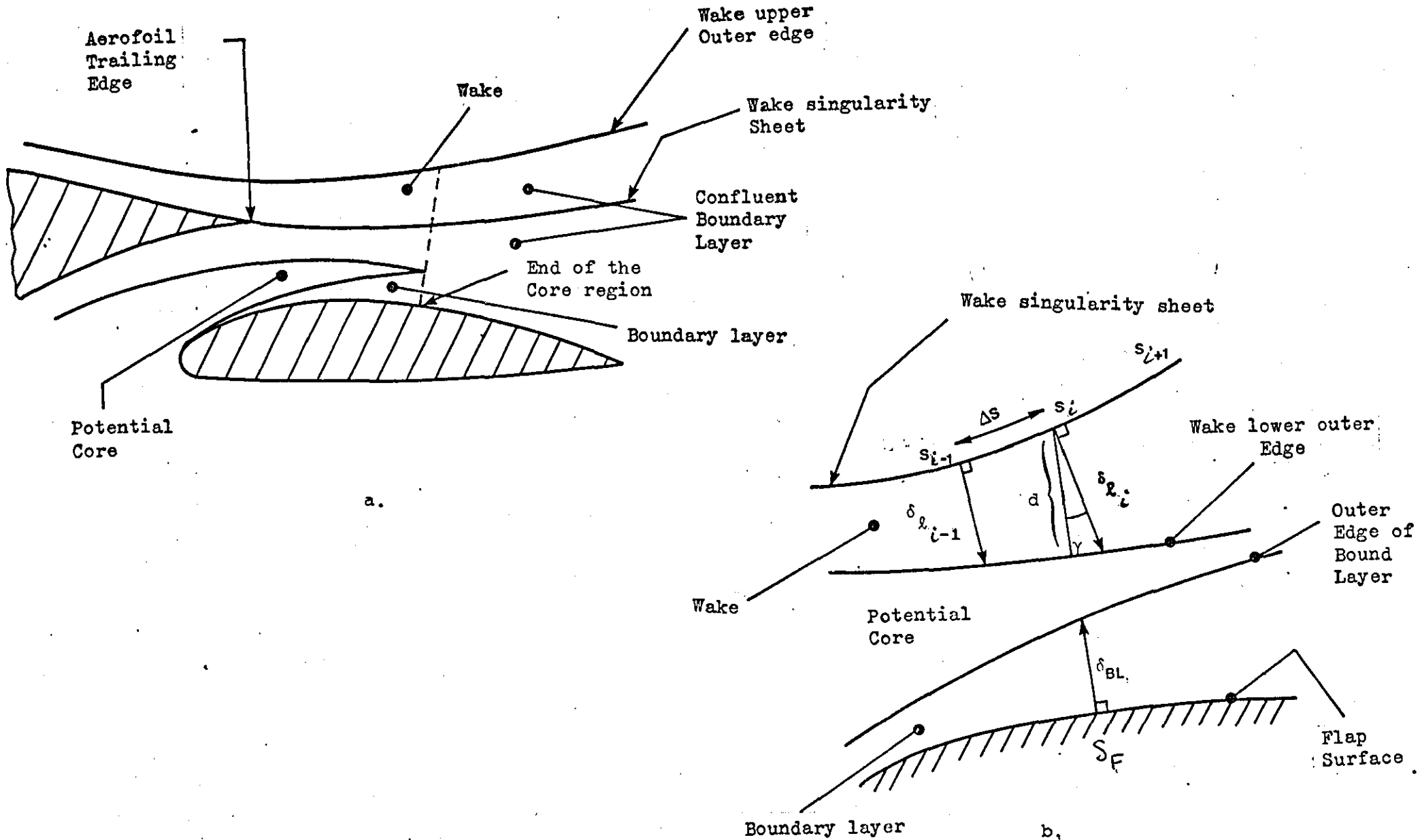


FIGURE 9.4: Core Region Calculation

CONFLUENT BOUNDARY LAYER

10.1 INTRODUCTION

The incompressible mathematical model of the confluent boundary layer due to Goradia (Ref. 109) is used to calculate the boundary layer growth downstream of the slot. The model is based on a limited experimental data. Thus the results should be treated with care. The present section gives a brief description of the flow model together with the necessary set of equations for the calculation of the flow parameters.

10.2 GENERAL CONSIDERATIONS

As shown in (Figure 10.1) the flow over the flap consists of the following regions:

1. Core region
2. Confluent boundary layer
3. Ordinary turbulent boundary layer.

The existence of this kind of flow region depends on a number of parameters such as:

1. Aerofoil geometry
2. Flight conditions.

For interaction between the flap boundary layer and the wake flow behind the main aerofoil the gap between the aerofoil and the flap should be small. Other parameters such as flap deflection and aerofoil angle of attack are also important. Finally for all flow regions to exist the flap chord length must be relatively large. There are many cases (at optimum conditions) when the potential core extends beyond the flap trailing edge

thus there is no interaction between the flap boundary layer and the main aerofoil wake.

10.2.1 Main Assumptions

The prediction of the confluent boundary layer flow is based on the following assumptions:

1. The flow is two dimensional and incompressible.
2. Surface curvatures are assumed small so that their effect can be ignored.
3. The viscous layers comprising the confluent boundary layer regions are governed by the two dimensional incompressible boundary layer equations

$$u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} = U_e \frac{dU_e}{dx} + \frac{\partial}{\partial y} \left(\frac{\tau}{\rho} \right) \quad (10.1)$$

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0 \quad (10.2)$$

4. The static pressure in the direction normal to the surface along which the confluent boundary layer develops is constant.
5. The flow is attached.
6. The velocity profiles of the individual viscous layers are self-similar.

10.3 MODEL DESCRIPTION

Goradia divides the confluent boundary layer into two flow regions, each of which consists of a number of viscous layers.

The first region is called main region 1. and starts immediately downstream of the potential core. As shown in Figure 10.2 the region is subdivided into three viscous layers termed:

wall layer

jet layer

wake layer

The wall layer is the continuation of the upstream boundary layer on the flap surface. The jet layer and wake layer represent the remainder of the inner and outer part of the viscous wake of the main aerofoil respectively.

The velocity at the outer edge of the wall layer is denoted by U_m . The symbol U_w denotes the velocity at the common boundary of jet layer and wake layer. The outer edges of the three layers are denoted by the variables $\delta_5, \delta_3, \delta_4$ (Figure 10.2).

The velocities U_m, U_w at the edge of the wall layer and jet layer respectively must be determined from a viscous solution, in contrast to U_e which is known from the wake potential flow solution.

The wake layer from $y=\delta_3$ to $y=\delta_4$ is shown to terminate at station C. At this point main region 2 starts. This region consists of two layers, namely the wall layer and the jet layer. At the end of this region the jet layer disappears and the flow is that of the ordinary turbulent boundary layer type (Figure 10.3).

The aim of the model is to predict the confluent boundary layer displacement thickness, the overall skin friction coefficient and find whether separation of the flow has taken place.

The governing equations of the viscous layers are the turbulent boundary layer equations mentioned earlier. The equations of each layer are solved utilizing an integral method and the assumption of self-similar profile. Turbulence is modelled by empirical relations (Ref. 109) for the following quantities:

1. Growth functions for the widths of the jet layer and wake layer.
2. Shear stress.
3. Velocity profiles and their integrals.

Goradia's original method uses the assumption of a one parameter velocity profile for the wall layer

$$\frac{u}{U_m} = \left(\frac{y}{\delta_5}\right)^{1/n} \quad (10.3)$$

Reference 8 assumes Coles' two parameter velocity profile in order to predict the point of separation. This method has been adopted here.

10.4 EQUATIONS FOR THE CALCULATION OF MAIN REGIONS 1 AND 2

The necessary equations for the calculation of all relevant confluent boundary layer parameters are listed here. The equations are taken directly from Ref.8, where details for their derivation could be found.

10.4.1 Coles Velocity Profile

Coles (17) law of the wake in the turbulent boundary layer defines the following two parameter velocity profile at the wall

$$u = \frac{u_\tau}{k} \left[\ln \frac{yu_\tau}{\nu} + c \right] + \frac{u_\beta}{2} \left[1 - \cos \frac{\pi y}{\delta_5} \right] \quad (10.4)$$

where u_τ is the friction velocity which is defined in terms of the wall shear τ_ω and density ρ and is given by

$$u_\tau = \sqrt{\frac{\tau_\omega}{\rho}} \quad (10.5)$$

u_β is an unknown parameter with dimensions of velocity, k and c are constants and have the following values

$$k = 0.41 \quad c = 2.05 \quad (10.6)$$

ν is the kinematic viscosity of air and δ_5 is the thickness of the wall layer.

Using equation (10.4) and the equations defining δ_5^* , δ_5 and δ_5^{**} the following equations are obtained:

$$U_m \delta_5^* = \left(\frac{u_\tau}{k} + \frac{u_\beta}{2} \right) \delta_5 \quad (10.7)$$

$$U_m^2 \theta_5 = \left[U_m \frac{u_\tau}{k} + U_m \frac{u_\beta}{2} - k_1 \frac{u_\tau}{k} u_\beta - 2 \left(\frac{u_\tau}{k} \right)^2 - \frac{3}{8} u_\beta^2 \right] \delta_5 \quad (10.8)$$

$$U_m^3 \delta_5^{**} = \left[\frac{5}{16} u_\beta^3 - \frac{9}{8} U_m u_\beta^2 + U_m^2 u_\beta + 2 U_m^2 \frac{u_\tau}{k} - 3 k_1 U_m \frac{u_\tau}{k} u_\beta - 6 U_m \left(\frac{u_\tau}{k} \right)^2 + 3 k_2 \frac{u_\tau}{k} u_\beta^2 + 3 k_3 \left(\frac{u_\tau}{k} \right)^2 u_\beta + 6 \left(\frac{u_\tau}{k} \right)^3 \right] \delta_5 \quad (10.9)$$

$$\text{where } k_1 = 1.58949, \quad k_2 = 0.697958, \quad k_3 = 1.846111 \quad (10.10)$$

10.4.2 Main Region 1

A set of eight first order ordinary differential equations employed for the analysis of this region. These equations are the:

Momentum integral equation of the wall layer

$$A_{11} \frac{dU_m}{dx} + A_{12} \frac{d\theta_5}{dx} = B_1 \quad (10.11)$$

Energy integral equation of the wall layer

$$A_{21} \frac{dU_m}{dx} + A_{23} \frac{d\delta_5^{**}}{dx} = B_2 \quad (10.12)$$

Momentum integral equation of the jet layer

$$A_{31} \frac{dU_m}{dx} + A_{34} \frac{d\delta_5}{dx} + A_{35} \frac{d\delta_5^*}{dx} + A_{36} \frac{dU_w}{dx} = B_3 \quad (10.13)$$

Momentum integral equation of the wake layer

$$A_{41} \frac{dU_m}{dx} + A_{44} \frac{d\delta_5}{dx} + A_{45} \frac{d\delta_5^*}{dx} + A_{46} \frac{dU_w}{dx} = B_4 \quad (10.14)$$

Equation 10.7 differentiated with respect to x

$$A_{51} \frac{dU_m}{dx} + A_{54} \frac{d\delta_5}{dx} + A_{55} \frac{d\delta_5^*}{dx} + A_{57} \frac{du_\tau}{dx} + A_{58} \frac{du_\beta}{dx} = 0 \quad (10.15)$$

Equation 10.8 differentiated with respect to x

$$A_{61} \frac{dU_m}{dx} + A_{62} \frac{d\theta_5}{dx} + A_{64} \frac{d\delta_5}{dx} + A_{67} \frac{du_\tau}{dx} + A_{68} \frac{du_\beta}{dx} = 0 \quad (10.16)$$

Equation 10.9 differentiated with respect to x

$$A_{71} \frac{dU_m}{dx} + A_{73} \frac{d\delta_5^{**}}{dx} + A_{74} \frac{d\delta_5}{dx} + A_{77} \frac{du_\tau}{dx} + A_{78} \frac{du_\beta}{dx} = 0 \quad (10.17)$$

Setting $u=U_m$ and $y=\delta_5$ in equation 10.4 gives

$$U_m = \frac{u_\tau}{k} \left(\ln \frac{\delta_5 u_\tau}{\nu} + c \right) + u_\beta$$

Differentiating the above with respect to x the final equation of the set is obtained.

Differentiated skin friction law equation

$$A_{81} \frac{dU_m}{dx} + A_{84} \frac{d\delta_5}{dx} + A_{87} \frac{du_\tau}{dx} + A_{88} \frac{du_\beta}{dx} = 0 \quad (10.18)$$

The above set of equations can be written in matrix form as:

$$[A] \left[\frac{d\phi}{dx} \right] = [B] \quad (10.19)$$

where

$$A = \begin{bmatrix} A_{11} & A_{12} & 0 & 0 & 0 & 0 & 0 & 0 \\ A_{21} & 0 & A_{23} & 0 & 0 & 0 & 0 & 0 \\ A_{31} & 0 & 0 & A_{34} & A_{35} & A_{36} & 0 & 0 \\ A_{41} & 0 & 0 & A_{44} & A_{45} & A_{46} & 0 & 0 \\ A_{51} & 0 & 0 & A_{54} & A_{55} & 0 & A_{57} & A_{58} \\ A_{61} & A_{62} & 0 & A_{64} & 0 & 0 & A_{67} & A_{68} \\ A_{71} & 0 & A_{74} & 0 & 0 & 0 & A_{77} & A_{78} \\ A_{81} & 0 & 0 & A_{84} & 0 & 0 & A_{87} & A_{88} \end{bmatrix} \quad B = \begin{bmatrix} B_1 \\ B_2 \\ B_3 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad (10.20)$$

$$\left[\frac{d\phi}{dx} \right] = \left[\frac{dU_m}{dx} \quad \frac{d\delta_5}{dx} \quad \frac{d\delta_5^{**}}{dx} \quad \frac{d\delta_5}{-dx} \quad \frac{d\delta_5^*}{dx} \quad \frac{dU_w}{dx} \quad \frac{du_\tau}{dx} \quad \frac{du_\beta}{dx} \right]$$

The elements of matrix A are determined from the following equations:

$$A_{11} = (2\theta_5 - \delta_5 + \delta_5^*) / U_m \quad (10.21)$$

$$A_{12} = 1.0 \quad (10.22)$$

$$A_{21} = (3\delta_5^{**} - 2\delta_5 + 2\delta_5^*) / U_m \quad (10.23)$$

$$A_{23} = 1.0 \quad (10.24)$$

$$A_{31} = b_j \left[2U_m - U_w - 4U_m S_{m3} + 3U_w S_{m3} + 2(U_m - U_w) S_{m5} \right] + (U_m - U_w) (\delta_5 - \delta_5^*) \quad (10.25)$$

$$A_{34} = U_m (U_m - U_w) \quad (10.26)$$

$$A_{35} = -U_m (U_m - U_w) \quad (10.27)$$

$$A_{36} = b_j \left[2U_m S_{m3} - U_w S_{m3} - 2(U_m - U_w) S_{m5} \right] \quad (10.28)$$

$$S_{m3} = 0.5644, \quad S_{m5} = 0.4331 \quad (10.29)$$

$$A_{41} = (U_w - U_e) (b_j - b_j S_{m_3} + \delta_5 - \delta_5^*) \quad (10.30)$$

$$A_{44} = U_m (U_w - U_e) \quad (10.31)$$

$$A_{45} = -U_m (U_w - U_e) \quad (10.32)$$

$$A_{46} = b_w U_e S_{m_1} + 2b_w (U_w - U_e) S_{m_2} + b_j (U_w - U_e) S_{m_3} \quad (10.33)$$

$$S_{m_1} = 1.178, \quad S_{m_2} = 0.786 \quad (10.34)$$

$$A_{51} = \delta_5^* \quad (10.35)$$

$$A_{54} = -\left(\frac{u_\tau}{k} + \frac{u_\beta}{2}\right) \quad (10.36)$$

$$A_{55} = U_m \quad (10.37)$$

$$A_{57} = -\delta_5/k \quad (10.38)$$

$$A_{58} = -\delta_5/2 \quad (10.39)$$

$$A_{61} = 2U_m \theta_5 - \delta_5 \left(\frac{u_\tau}{k} + \frac{u_\beta}{2}\right) \quad (10.40)$$

$$A_{62} = U_m^2 \quad (10.41)$$

$$A_{64} = -U_m^2 \theta_5 / \delta_5 \quad (10.42)$$

$$A_{67} = -\frac{\delta_5}{k} (U_m - k_1 u_\beta - 4 \frac{u_\tau}{k}) \quad (10.43)$$

$$A_{68} = -\delta_5 \left(\frac{U_m}{2} - k_1 \frac{u_\tau}{k} - \frac{3}{4} u_\beta\right) \quad (10.44)$$

$$A_{71} = 3U_m^2 \delta_5^{**} + \frac{9}{8} \delta_5 u_\beta^2 - 2\delta_5 U_m u_\beta - 4\delta_5 U_m \frac{u_\tau}{k} + 3\delta_5 k_1 \frac{u_\tau}{k} u_\beta + 6\delta_5 \left(\frac{u_\tau}{k}\right)^2 \quad (10.45)$$

$$A_{73} = U_m^3 \quad (10.46)$$

$$A_{74} = -U_m^3 \delta_5^{**} / \delta_5 \quad (10.47)$$

$$A_{77} = -\frac{\delta_5}{k} (2U_m^2 - 3k_1 U_m u_\beta - 12U_m \frac{u_\tau}{k} + 3k_2 u_\beta^2 + 6k_3 \frac{u_\tau}{k} u_\beta + 18\left(\frac{u_\tau}{k}\right)^2) \quad (10.48)$$

$$A_{78} = -\delta_5 \left(\frac{15}{16} u_\beta^2 - \frac{9}{4} U_m u_\beta + u_m^2 - 3k_1 U_m \frac{u_\tau}{k} + 6k_2 \frac{u_\tau}{k} u_\beta + 3k_3 \left(\frac{u_\tau}{k}\right)^2\right) \quad (10.49)$$

$$A_{81} = 1 \quad (10.50)$$

$$A_{84} = -\frac{u}{k\delta_5} \quad (10.51)$$

$$A_{87} = -\left(\frac{U}{u} \beta + \frac{1}{k}\right) \quad (10.52)$$

$$A_{88} = -1.0 \quad (10.53)$$

The following equations are used for the calculations of the coefficients of matrix B.

$$B_1 = -U_e \frac{dU_e}{dx} \frac{\delta_5}{U_m^2} + \frac{\tau_\omega - \tau(\delta_5)}{\rho U_m^2} \quad (10.54)$$

$$\frac{\tau_\omega - \tau(\delta_5)}{\rho U_m^2} = 1.385 Y^{-45.79} e^{-0.918H_5} + 17.21Y - 0.743Y^2 \quad (\text{empirical}) \quad (10.55)$$

$$Y = \ln \frac{U_m \theta_5}{v} \quad H_5 = \delta_5^*/\theta_5 \quad (10.56)$$

$$B_2 = -2U_e \frac{dU_e}{dx} \frac{\delta_5 - \delta_5^*}{U_m^2} - \frac{2\tau(\delta_5)}{\rho U_m^2} + 2 \int_0^{\delta_5} \frac{\tau}{\rho U_m^2} \frac{\partial}{\partial Y} \left(\frac{u}{U_m}\right) dy \quad (10.57)$$

$$\int_0^{\delta_5} \frac{\tau}{\rho U_m^2} \frac{\partial}{\partial Y} \left(\frac{u}{U_m}\right) dy = 0.889 10^{23} Y^{-158.7} e^{-0.636H_5} + 48.55Y - 1.82Y^2 \quad (\text{empirical}) \quad (10.58)$$

$$\frac{\tau(\delta_5)}{\rho U_m^2} = \frac{\tau_\omega}{\rho U_m^2} - \frac{\tau_\omega - \tau(\delta_5)}{\rho U_m^2} \quad \begin{array}{l} \text{shear stress at the} \\ \text{outer edge of the wall layer} \end{array} \quad (10.59)$$

$$\frac{\tau_\omega}{\rho U_m^2} = 0.943 10^{16} Y^{-114.6} e^{-1.819H_5} + 35.68Y - 1.365Y^2 \quad \text{wall shear} \quad (10.60)$$

The term $\frac{\tau_\omega - \tau(\delta_5)}{\rho U_m^2}$ in (10.59) can be found from equation (10.55).

$$B_3 = U_e \frac{dU_e}{dx} b_j + U_m^2 \frac{\tau(\delta_3)}{\rho U_m^2} - U_m^2 \frac{\tau(\delta_5)}{\rho U_m^2} - \quad (10.61)$$

$$(U_m - U_w) \left[U_m - 2U_m S_{m3} + U_m S_{m3} + (U_m - U_w) S_{m5} \right] \frac{db_j}{dx} \quad (10.62)$$

$$S_{m3} = 0.5644, \quad S_{m5} = 0.4331 \quad (10.63)$$

$$\tau(\delta_3) = 0.3\tau_\omega \quad \begin{array}{l} \text{shear stress at the outer edge of} \\ \text{the jet layer (assumption)} \end{array} \quad (10.64)$$

τ_w is obtained from equation above

$\tau(\delta_5)$ is also obtained from equations above

$$\frac{db_j}{dx} = 0.17 \frac{U_m - U_w}{U_m + U_w} \quad \text{growth of jet layer} \quad (10.65)$$

$$B_4 = \left[U_e S_{m_1} - 2(U_w - U_e) S_{m_1} + 2(U_w - U_e) S_{m_2} \right] b_w \frac{dU_e}{dx} - U_m^2 \frac{\tau(\delta_3)}{\rho U_m^2} + (U_w - U_e) \left[(U_m - U_w) S_{m_3} - U_m \right] \frac{db_j}{dx} - (U_w - U_e) \left[U_e S_{m_1} + (U_w - U_e) S_{m_2} \right] \frac{db_w}{dx} \quad (10.66)$$

$$S_{m_1} = 1.178, \quad S_{m_2} = 0.786 \quad (10.67)$$

$$\frac{db_w}{dx} = 0.185 \frac{U_e - U_w}{U_e + U_w} \quad (10.68)$$

10.4.3 Main Region 2

The wake layer does not exist in this region hence

$$U_w = U_e \quad \tau(\delta_3) = 0 \quad (10.69)$$

The number of unknowns reduces to seven, so

$$\bar{\phi} = (U_m, \theta_5, \delta_5^{**}, \delta_5^*, u_\tau, u_\beta)$$

The governing equations of main region 2 are derived from those of main region 1 by eliminating the momentum integral equation of the wake layer and rewriting the momentum integral equation of the jet layer using equation (10.69). All other equations remain the same. In particular the empirical coefficients in the shear-stress terms and the growth function for the jet layer thickness are not changed. Writing the set of ordinary differential equations as

$$[\bar{A}] \left[\frac{d\bar{\phi}}{dx} \right] = [\bar{B}] \quad (10.70)$$

where

$$\bar{A} = \begin{bmatrix} \bar{A}_{11} & \bar{A}_{12} & 0 & 0 & 0 & 0 & 0 \\ \bar{A}_{21} & 0 & \bar{A}_{23} & 0 & 0 & 0 & 0 \\ \bar{A}_{31} & 0 & 0 & \bar{A}_{34} & \bar{A}_{35} & \bar{A}_{36} & 0 \\ \bar{A}_{41} & 0 & 0 & \bar{A}_{44} & \bar{A}_{45} & \bar{A}_{46} & \bar{A}_{47} \\ \bar{A}_{51} & \bar{A}_{52} & 0 & \bar{A}_{54} & 0 & \bar{A}_{56} & \bar{A}_{57} \\ \bar{A}_{61} & 0 & \bar{A}_{63} & \bar{A}_{64} & 0 & \bar{A}_{66} & \bar{A}_{67} \\ \bar{A}_{71} & 0 & 0 & \bar{A}_{74} & 0 & \bar{A}_{76} & \bar{A}_{77} \end{bmatrix} \quad \bar{B} = \begin{bmatrix} \bar{B}_1 \\ \bar{B}_2 \\ \bar{B}_3 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad (10.71)$$

The elements of matrix A and matrix B are given by:

Coefficients of matrix \bar{A}

$$\bar{A}_{11} = \bar{A}_{11} \quad (10.72)$$

$$\bar{A}_{12} = A_{12} \quad (10.73)$$

$$\bar{A}_{21} = A_{21} \quad (10.74)$$

$$\bar{A}_{23} = A_{23} \quad (10.75)$$

$$\bar{A}_{31} = b_j \left[2U_m - U_e - 4U_m S_{m3} + 3U_e S_{m3} + 2(U_m - U_e) S_{m5} \right] + (U_m - U_e) (\delta_5 - \delta_5^*) \quad (10.76)$$

$$\bar{A}_{34} = U_m (U_m - U_e) \quad (10.77)$$

$$\bar{A}_{35} = -U_m (U_m - U_e) \quad (10.78)$$

$$\bar{A}_{36} = 0 \quad (10.79)$$

$$\frac{db_j}{dx} = 0.17 \frac{U_m - U_e}{U_m + U_e} \quad (10.80)$$

$$\bar{A}_{41} = A_{51} \quad \bar{A}_{44} = A_{54} \quad \bar{A}_{45} = A_{55} \quad \bar{A}_{46} = A_{57} \quad \bar{A}_{47} = A_{58} \quad (10.81)$$

$$\bar{A}_{51} = A_{61} \quad \bar{A}_{52} = A_{62} \quad \bar{A}_{54} = A_{64} \quad \bar{A}_{56} = A_{67} \quad \bar{A}_{57} = A_{68} \quad (10.82)$$

$$\bar{A}_{61} = A_{71} \quad \bar{A}_{63} = A_{73} \quad \bar{A}_{64} = A_{74} \quad \bar{A}_{66} = A_{77} \quad \bar{A}_{67} = A_{78} \quad (10.83)$$

$$\bar{A}_{71} = A_{81} \quad \bar{A}_{74} = A_{84} \quad \bar{A}_{76} = A_{87} \quad \bar{A}_{77} = A_{88} \quad (10.84)$$

Coefficients of matrix \bar{B}

$$\bar{B}_1 = B_1 \quad (10.85)$$

$$\bar{B}_2 = B_2 \quad (10.86)$$

$$\bar{B}_3 = b_j \left[U_e - 2U_m S_{m_3} + U_e S_{m_3} + 2(U_m - U_e) S_{m_5} \right] \frac{dU_e}{dx} - U_m^2 \frac{\tau(\delta_5)}{\rho U_m^2} - (U_m - U_e) \left[U_m - 2U_m S_{m_3} + U_e S_{m_3} + (U_m - U_e) S_{m_5} \right] \frac{db_j}{dx} \quad (10.87)$$

$$\frac{db_j}{dx} = 0.17 \frac{U_m - U_e}{U_m + U_e} \quad (10.88)$$

10.5 INITIAL CONDITIONS

In specifying initial values for main region 1 two cases are distinguished depending on whether or not a potential core exists at the slot exit. A potential core exists at the slot exit if

$$\delta_{BL}(x_0) + \delta_{s_1} < h_{slot} \quad (10.89)$$

where

$\delta_{BL}(x_0)$ = thickness of the boundary layer on the upper surface of the flap at the slot exit

δ_{s_1} = boundary layer thickness at the lower surface trailing edge of the main aerofoil

h_{slot} = slot height at the slot exit.

10.5.1 Main Region 1 is Preceded by a Core Region

In this case $\delta_{BL}(x_0) + \delta_{s_1} < h_{slot}$. Denoting the location of the end of the core by x_e the initial values read,

$$\left. \begin{aligned} U_m &= V_c(x_e) & \delta_5 &= \delta_{BL}(x_e) \\ \theta_5 &= \theta_{BL}(x_e) & \delta_5^* &= \delta_{BL}^*(x_e) \\ \delta_5^{**} &= 1.73\theta_{BL}(x_e) & U_w &= 0.8 U_{wake}(x_e) \end{aligned} \right\} \quad (10.90)$$

$$u_\tau = U_m \sqrt{\frac{\tau_w}{\rho U_m^2}} \quad u_\beta = U_m - \frac{u_\tau}{k} \left(\ln \frac{\delta_5 u_\tau}{\nu} + c \right) \quad (10.91)$$

$$\frac{\tau_w}{\rho U_m^2} = 0.123 \cdot 10^{-0.678 H_5} (Re_{\theta_5})^{-0.268}, \quad H_5 = \frac{\delta_5^*}{\theta_5} \quad (10.92)$$

$$Re_{\theta_5} = \frac{U_m \theta_5}{\nu}$$

$V_c(x_e)$ is the surface velocity of the flap at the slot exit.

U_{wake} = wake centerline velocity.

The initial values of b_j and b_w are given by

$$b_j = (\delta_l)_{wake} \quad b_w = \frac{(\delta_u)_{wake}}{k_2} \quad (10.93)$$

$$k_2 = 2.5$$

where δ_l, δ_u are defined in diagram 10.1

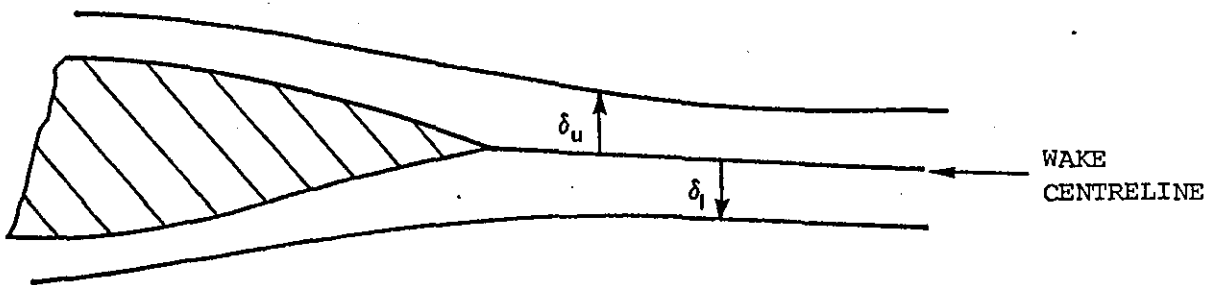


DIAGRAM 10.1

10.5.2 Main Region 1 Not Preceded by Core

For this case

$$\delta_{BL}(x_0) + \delta_{s_1} \geq h_{slot}$$

and the computation enters main region 1 directly at the slot exit. The

initial values at the slot exit x_0 are

$$\left. \begin{aligned} U_m &= V_c(x_0) & u_\tau &= U_m \sqrt{\frac{\tau_\omega}{\rho U_m^2}} \\ \theta_5 &= \theta_{BL}(x_0) = \theta_{s_2} & u_\beta &= U_m - \frac{\rho U_m}{k} \left(\ln \frac{\delta_5 u}{\nu} + c \right) \\ \delta_5^{**} &= 1.68 \theta_5 & \frac{\tau_\omega}{\rho U_m^2} &= 0.123 \cdot 10^{-0.678 H_5} (Re_{\theta_5})^{-0.268} \\ \delta_5 &= \delta_{BL}(x_0) = \delta_{s_2} & H_5 &= \frac{\delta_5^*}{\theta_5} \quad Re_{\theta_5} = \frac{U_m \theta_5}{\nu} \\ \delta_5^* &= \delta_{BL}^*(x_0) & & \\ U_w &= 0.8 V_c(x_0) & & \end{aligned} \right\} \quad (10.94)$$

The thicknesses of the jet layer and wake layer are initially

$$b_j = \delta_{s_1} \quad b_w = \frac{\delta_f}{k_2} = 0.4\delta_f \quad (10.95)$$

where δ_{s_1} and δ_f denote the boundary layer thicknesses at the lower and upper surface trailing edge of the upstream aerofoil respectively.

10.5.3 Main Region 2

It is assumed that this region is always preceded by main region 1. Therefore the initial values of main region 2 are simply the values of the variables at the end of main region 1.

10.6 CALCULATION PROCEDURE

Using the initial conditions at the beginning of main region 1, equation (10.19) is used to calculate the values of all the derivatives $\frac{d\phi}{dx}$ (equation (10.20)). The values of ϕ are found at the next step using equations (8.5) and (8.6) (Section 8). The values of $\frac{d\phi}{dx}$ at the next station x are now calculated and the values of ϕ for the next step are predicted and so on until the end of the region is reached. (i.e. when the value of b_w is found to be zero).

Equation (10.70) is then used to predict the development of main region 2 using the same method as above.

10.7 CONFLUENT BOUNDARY LAYER PARAMETERS

10.7.1 Thickness δ_4

This is obtained from

$$\delta_4 = \delta_5 + b_j + k_2 b_w, \quad k_2 = 2.5 \quad (10.95)$$

where b_j and b_w are calculated from the empirical growth functions

$$\left. \begin{aligned} \frac{db_j}{dx} &= 0.17 \frac{U_m - U_w}{U_m + U_w} \\ \frac{db_w}{dx} &= 0.185 \frac{U_e - U_w}{U_e + U_w} \end{aligned} \right\} \quad (10.96)$$

and the initial values of equations

. By definition the wake

thickness b_w is zero in main region 2.

10.7.2 Displacement Thickness δ^*

This is calculated from

$$\delta^* = \delta_5^* + \delta_j^* + \delta_w^* \quad (10.97)$$

using

$$\delta_j^* = b_j \left[1 - \frac{U_m}{U_e} + \left(\frac{U_m}{U_e} - \frac{U_w}{U_e} \right) S_{m_3} \right] \quad (10.98)$$

$$\delta_w^* = b_w \left(1 - \frac{U_w}{U_e} \right) S_{m_1} \quad (10.99)$$

$$S_{m_1} = 1.178 \quad , \quad S_{m_3} = 0.5644 \quad (10.100)$$

10.7.3 Momentum Thickness θ

The confluent boundary layer momentum thickness is given by

$$\theta = \theta_5 + \theta_j + \theta_w \quad (10.101)$$

with

$$\theta_j = b_j \left[\frac{U_m}{U_e} \left(1 - \frac{U_m}{U_e} \right) - \left(1 - \frac{U_m}{U_e} \right) \left(\frac{U_m}{U_e} - \frac{U_w}{U_e} \right) S_{m_3} \right. \\ \left. + \frac{U_m}{U_e} \left(\frac{U_m}{U_e} - \frac{U_w}{U_e} \right) S_{m_3} - \left(\frac{U_m}{U_e} - \frac{U_w}{U_e} \right)^2 S_{m_5} \right] \quad (10.102)$$

$$\theta_w = b_w \left[\left(1 - \frac{U_w}{U_e} \right) S_{m_1} - \left(1 - \frac{U_w}{U_e} \right)^2 S_{m_2} \right] \quad (10.103)$$

$$S_{m_2} = 0.786 \quad , \quad S_{m_5} = 0.4331 \quad (10.104)$$

10.7.4 Skin Friction C_f

The skin friction coefficient is given by

$$C_f = \frac{\tau_w}{\frac{1}{2} \rho U_\infty^2} \quad (10.105)$$

The wall shear τ_w is computed from the friction velocity u_τ

$$\tau_w = u_\tau^2 \rho \quad (10.106)$$

Thus for $U_\infty = 1$

$$C_f = 2u_\tau^2 \quad (10.107)$$

10.7.5 Separation

Separation is assumed to take place when

$$C_{f_s} \leq 0.001$$

where

$$C_{f_s} = \frac{\tau_w}{\frac{1}{2}\rho V_c^2} = 2u_{\tau}^2/V_c^2.$$

10.8 FURTHER COMMENTS

The confluent boundary layer model just described has been programmed for the purpose of the present analysis. However the model has not yet been tested due to the limited time available. The program listing is given in Appendix 12 . Experimental results for comparison with theory can be found in Ref. (87).

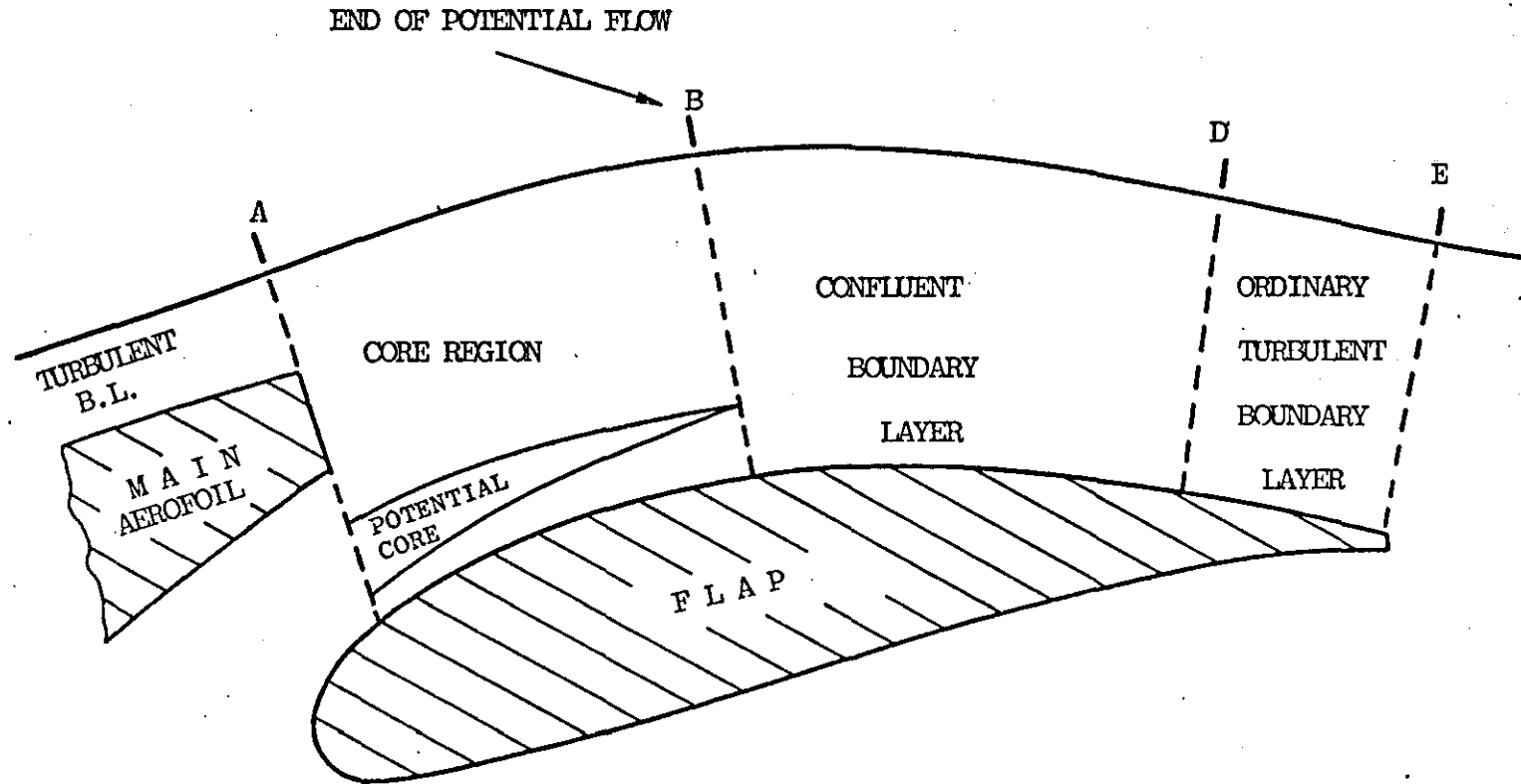


FIGURE 10.1 Slot Flow and Confluent Boundary Layer Model

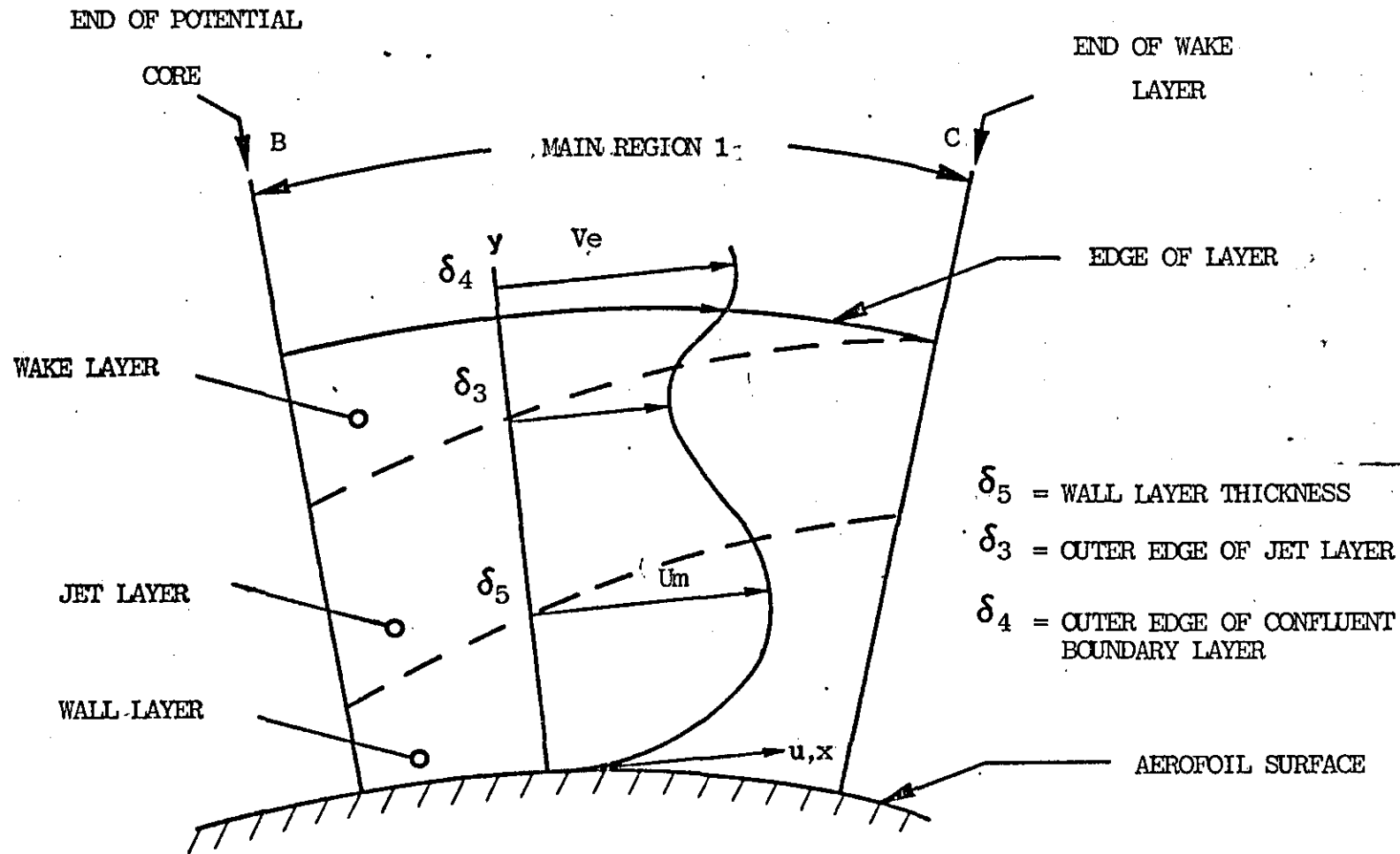


FIGURE 10.2 Layers of the Flow Model in Main Region 1

END OF WAKE LAYER
C

END OF JET LAYER

MAIN REGION 2

JET LAYER

WALL LAYER

Y
 δ_4

δ_5

U_e

U_m

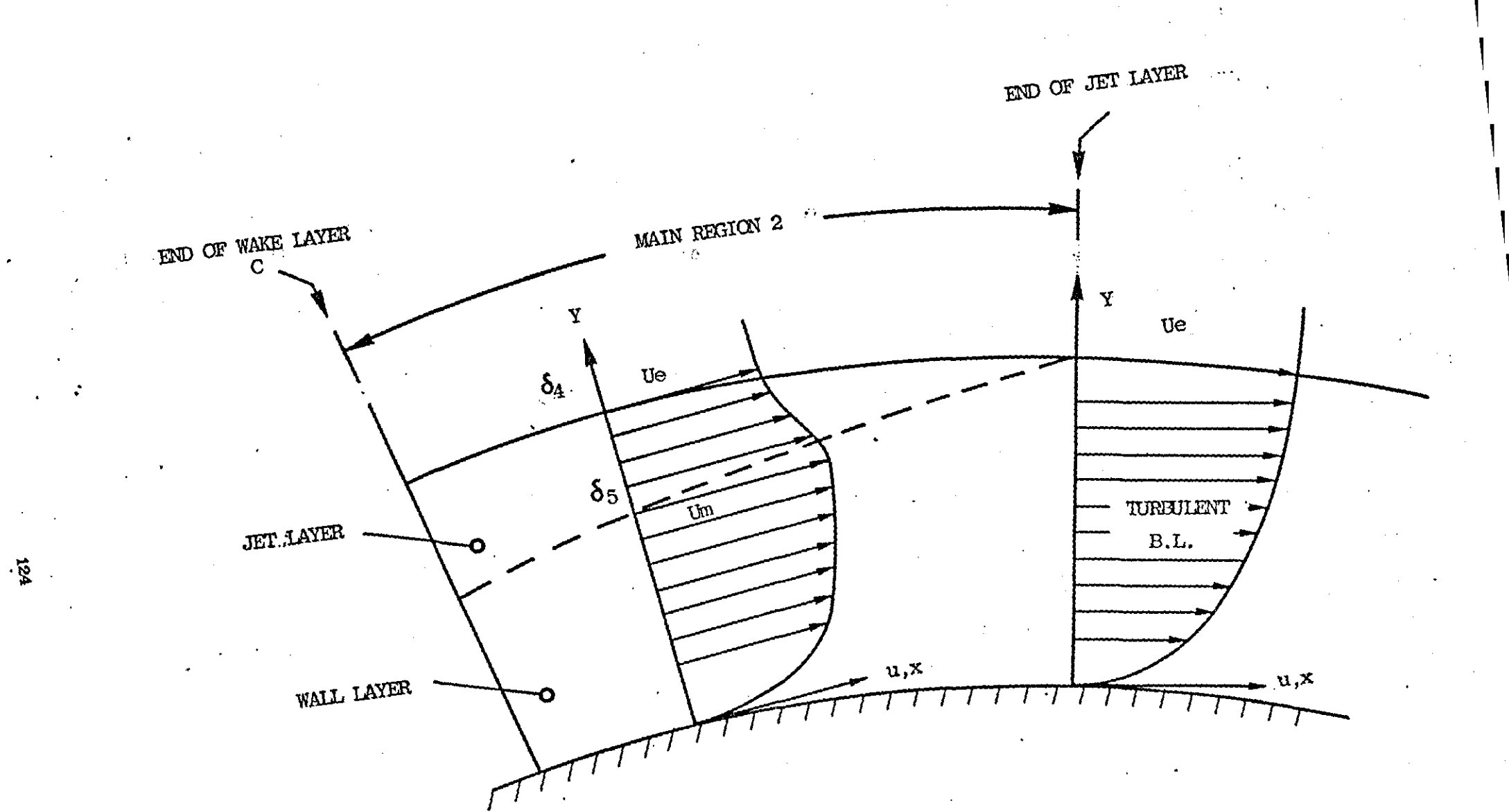
U_e

TURBULENT
B.L.

u,x

u,x

FIGURE 10.3 Layers of the Flow Model in Main Region 2



SEPARATED FLOW MODEL

11.1 INTRODUCTION

Flow separation is one of the most important of fluid flow phenomena related to aerodynamic lift and drag. Following a brief discussion on flow separation, a description of a mathematical model for turbulent boundary layer separation is given.

It is not the aim of this work to develop a separated flow model; instead a simple model is chosen which can be used to carry out preliminary numerical studies on aerofoil flow near and beyond the point of stall.

11.2 FLOW SEPARATION

Three types of boundary layer separation have been identified as contributing to aerofoil stall and are illustrated in Diagram 11.1..

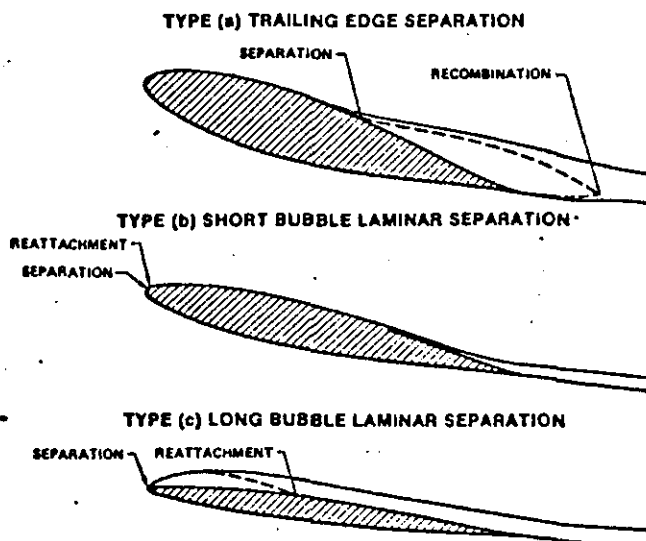


DIAGRAM 11.1: Illustration of three types of separated flow on an aerofoil

These types of separation can occur singly or in combination. The classic form of separation, described by Prandtl in his original paper proposing

the concept of a boundary layer, is shown as type (a) of Diagram 11.1. It is referred to as trailing edge separation. The separating boundary layer can be either laminar or turbulent. Separation first occurs at the aerofoil trailing edge and moves forward with increasing angle of attack. Trailing edge separation leads to aerofoil stall on relatively thick aerofoils ($t/c > 15\% - 18\%$). Analytical descriptions are available for fully developed trailing edge separation. Jacob (57) characterized the trailing edge separation bubble by distributing sources within the aerofoil near the point of boundary layer separation. The flow from the sources simulated the region of reverse flow. Their strength was determined by specifying equal pressures at the point of boundary layer separation, the aerofoil trailing edge, and a point on the equivalent aerofoil surface between the inner and outer flows. Similar but more refined models have been developed by Farn, Goldschmeid and Whirlow (36), Geller (1937) and Dvrorak (136). More advanced models have been developed by Gross (46), Chow and Spring (16), Korst (64), and Zumvalt and Sharad (122). The last model is a rather sophisticated model of the flow within the separation bubble.

Another form of boundary layer is illustrated as type (b) of diagram 11.1. Laminar separation is followed almost immediately by transition to turbulent flow in the shear layer and then reattachment. The resulting short bubble ($l/c = 1\% - 2\%$) develops as the aerofoil angle of attack is increased until reattachment no longer occurs. The bubble then is said to have "burst". On thick aerofoils short bubble laminar separation causes a thicker downstream boundary layer, but on thinner aerofoils short bubble bursting can lead directly to aerofoil stall. A detailed account of the short bubble separation phenomenon can be found in References 23 and 69. An analytical short bubble model is described in Section 7 of this work.

When the short bubble burst, a free shear layer is formed that recombines with the flow from the lower surface behind the aerofoil trailing

edge. However, under certain circumstances of aerofoil thickness and Reynolds number, the flow reattaches to the aerofoil surface and a new turbulent boundary layer is formed. This is flow (c) of diagram 11.1 and is referred to as a long laminar separation bubble. Increases of angle of attack cause the bubble length to increase until reattachment does not occur.

Aerofoil stall resulting from the types of flow separation described above is pictured in Diagram 11.2 (Ref. 15). Stall (d) of Diagram 11.2 is a combined short bubble and trailing edge type of stall.

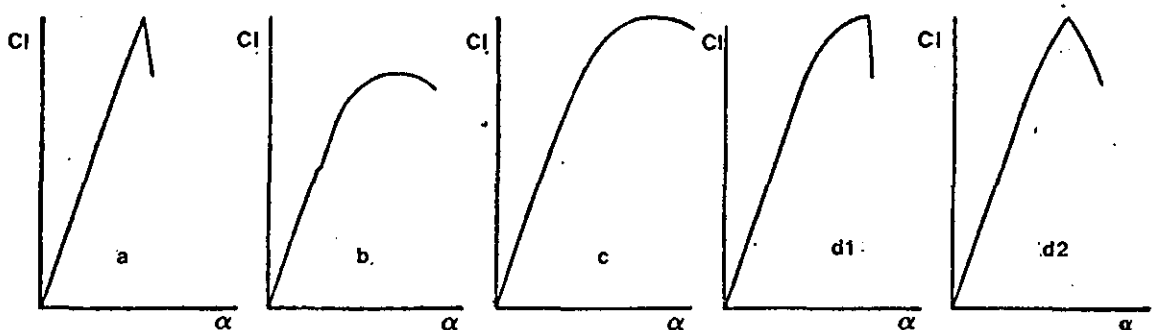


DIAGRAM 11.2: Aerofoil stall classification

11.3 SEPARATION FLOW MODEL

The separated flow model used here is that of Maskew and Dvorak (71) and is briefly discussed below.

11.3.1 Basic Considerations and Assumptions

The mathematical flow model is shown in Diagram 11.3. As can be seen from the diagram four main regions are used to model the real flow.

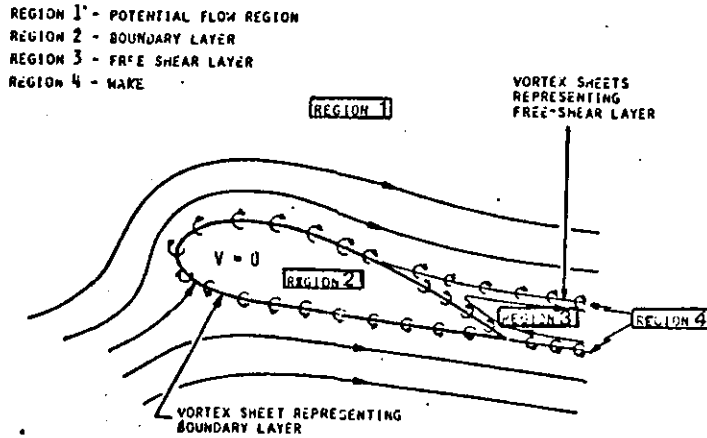


DIAGRAM 11.3: Mathematical flow model

- Region 1: This is the region exterior to the boundary layer and separated wake and is almost irrotational since the shear is everywhere so low that viscous stresses impart a negligible rotation to the fluid. Since the flow Mach number is assumed to be very low, region 1 is very nearly a potential flow region.
- Region 2: This is the region next to the aerofoil surface and is termed the boundary layer. The shear stresses are high, and hence, the viscous stresses are high. Thus significant vorticity is present in this region.
- Region 3: This is called the free shear layer. The thin flow region fed by the separating boundary layer has rotation, but only moderate shear.
- Region 4: The flow between the two shed boundary layers, the wake, is a region with low overall vorticity and significant viscous stresses.

The main assumptions used to define the model are:

1. The boundary layer and free shear layers do not have significant thickness and, hence, can be represented as slip surfaces; that is streamlines across which there exists a jump in velocity.
2. The wake does not have significant vorticity and has constant total pressure (lower than the free-stream total pressure). It is therefore taken to be a potential flow region.

Once again, the mathematical problem is to find the vorticity sheet strength such that the appropriate boundary conditions are met. The position of the vorticity sheet representing the free shear layer is not known a priori.

The boundary condition for the aerofoil surface is that of zero normal velocity ($V_N=0$). However, when allowing for viscous effects the incremental normal velocity $V_N = \frac{d}{ds}(U\delta^*)$ replaces the $V_N=0$. The boundary conditions are satisfied at the control points (panel midpoints).

11.3.2 Approximations for the Free Shear Layer

11.3.2.1 Wake shape

Initially, the streamlines are not known and so the shape of the free shear layers must be obtained iteratively starting from an initial position. Figure 11.1 shows the initial geometry suggested by reference 71. The upper and lower sheets are represented by parabolic curves passing from the separation points to a common point downstream which is positioned at a distance 0.2 to 0.3 chord lengths downstream on the mean wake line. The slope at the upstream end is the mean between the free stream direction and the local surface slope.

11.3.2.2 Pressure in the wake

If the average velocity in the upper shear layer is

$$v = \frac{1}{2} (v_{\text{outer}} + v_{\text{inner}}) \quad (11.1)$$

then
$$v_{\text{outer}} = v + \frac{\gamma_u}{2}, \quad v_{\text{inner}} = v - \frac{\gamma_u}{2} \quad (11.2)$$

where γ_u is the vorticity on the upper sheet.

The jump in total pressure across the shear layer is then

$$\Delta H = H_{\text{inner}} - H_{\text{outer}} = (P_{\text{inner}} + \frac{1}{2}\rho(v - \frac{\gamma_u}{2})^2) - (P_{\text{outer}} + \frac{1}{2}\rho(v + \frac{\gamma_u}{2})^2) \quad (11.3)$$

$\therefore \Delta H = -\rho v \gamma_u = \rho v \gamma_\ell$

where γ_ℓ is the vorticity on the lower sheet. The value of P is assumed constant across the shear layer. From assumption 2, the jump in total pressure across the free shear layer is the same everywhere.

11.3.3 Theoretical Basis

The potential flow analysis is based on the following integral equation:

$$\int_C k\gamma(s)ds + \gamma_\ell \left(\int_L kds - \int_U kds \right) + \underline{V}_\infty \cdot \underline{n} = V_N \quad (11.4)$$

where the constant value of the strength of the lower free vorticity sheet is used and where the kernel function, K , is the normal velocity component (at the boundary point for which $\underline{V} \cdot \underline{n}$ is being enforced) due to a unit point vortex at the point associated with the element, ds , of the line of integration, and where the integration paths, C, L and U are the aerofoil and the lower and upper free vortex sheet locations, respectively. The unknowns are the vorticity strengths on the aerofoil, and on the free sheets represented by $\gamma(s)$ and $\dot{\gamma}$, respectively. The former is a function of position on the aerofoil, and the latter is a constant. Two auxiliary conditions are applied; the first is related to the Kutta condition, and specifies that the vorticity values at the separation points on the upper and lower surfaces are equal but opposite and have the value of the free vorticity sheets, i.e., γ_ℓ . The second condition concerns the vorticity distribution on the aerofoil surface in the separated region; this distribution is constrained to start and finish with zero vorticity (i.e. zero velocity).

The right side of equation (11.4), V_N , is zero for the initial potential flow solution. Following a boundary layer analysis, however, the displacement effect is represented by a piecewise constant source distribution; V_N then becomes the integrated normal velocity induced by the boundary layer source distribution.

11.3.4 Method of Solution

The aerofoil contour is represented by an inscribed polygon (Figure 11.2).

The individual panels representing the polygon each have a linear variation of vorticity along them (see section 3.2). The free vortex sheets are represented by a number of panels of uniform vorticity. The value of the vorticity at the start of the i th panel is denoted by γ_i . Thus the function $\gamma(s)$, in equation (11.4) can be expressed in terms of the unknown sequence, $[\gamma_i]$ and the integral equation in the unknown function, $\gamma(s)$ becomes a set of algebraic equations in the unknowns, $[\gamma_i]$. Initially there are $N+1$ unknown γ_i values (for N panels), but the auxiliary conditions remove two unknowns:

$$\gamma_{\text{separation}} = -\gamma_1, \quad \gamma_{N+1} = 0 \quad (11.5)$$

The γ value just downstream of the separation point on the upper surface is set to zero. A square set of linear equations is obtained by introducing one unknown source strength distributed uniformly along the aerofoil surface.

Once the vorticity is found the velocity (V) at any point in the flow field is found by adding to the free stream the velocities induced by the vorticity and source distributions. The value of C_p is given from

$$C_p = 1 - \left(\frac{V}{V_\infty}\right)^2 + \frac{\Delta H}{\frac{1}{2}\rho V_\infty^2} \quad (11.6)$$

where ΔH is zero everywhere except in the wake region where $\Delta H = \rho V \gamma_x$.

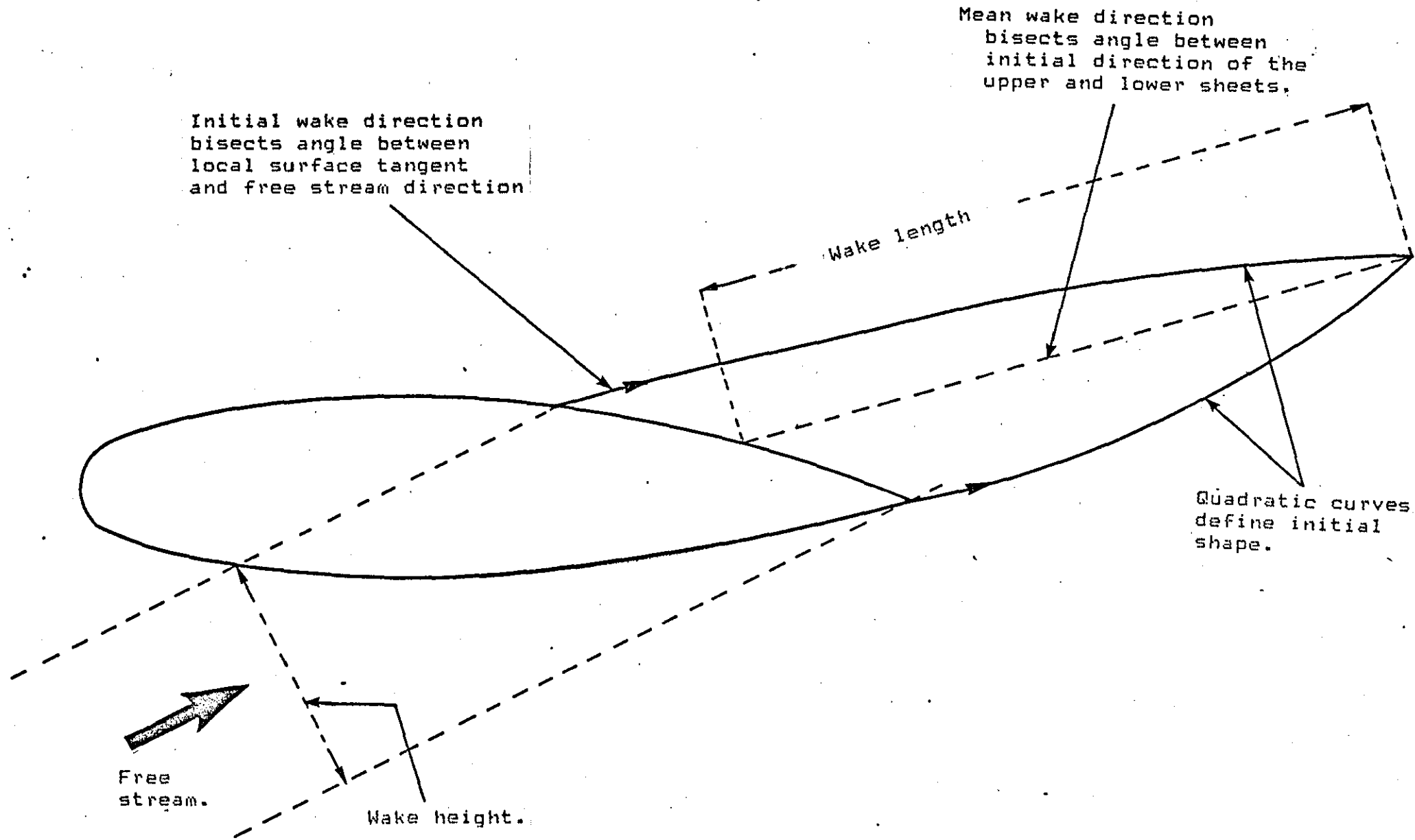
The overall solution to the problem is an iterative one. Firstly during each iteration the wake shape is calculated as follows, once the separation points have been located. Using the vorticity distribution, calculated from the previous iteration, velocities are calculated at the panel midpoints on the free vortex sheets. The new wake shape is then determined by piecewise integration, starting at the separation points. The downstream end points, which were coincident in the initial wake, are allowed to move independently in subsequent iterations. At each wake iteration, the wake influence coefficients at the surface control points are recalculated and a new potential flow solution is obtained. The number of wake iterations is fixed.

Once the wake shape is known the overall iteration continues. This involves a boundary layer analysis which gives the new separation points and the boundary layer source distribution. The sources are set to zero in the separated region.

A new wake shape is calculated using the new separation points together with information from the previous calculation of the wake. A new potential flow solution is then obtained and so on.

11.4 CONCLUDING REMARKS

Maskew's and Dvorak's separation model has been described. The model, however, has not yet been included in the computer program. However, since the program already contains a linear vorticity model for the calculation of the potential flow, the separated flow model should be relatively easy to include. Figure 11.3 indicates that good results can be expected from the model.



Initial wake direction
bisects angle between
local surface tangent
and free stream direction

Mean wake direction
bisects angle between
initial direction of the
upper and lower sheets,

Wake length

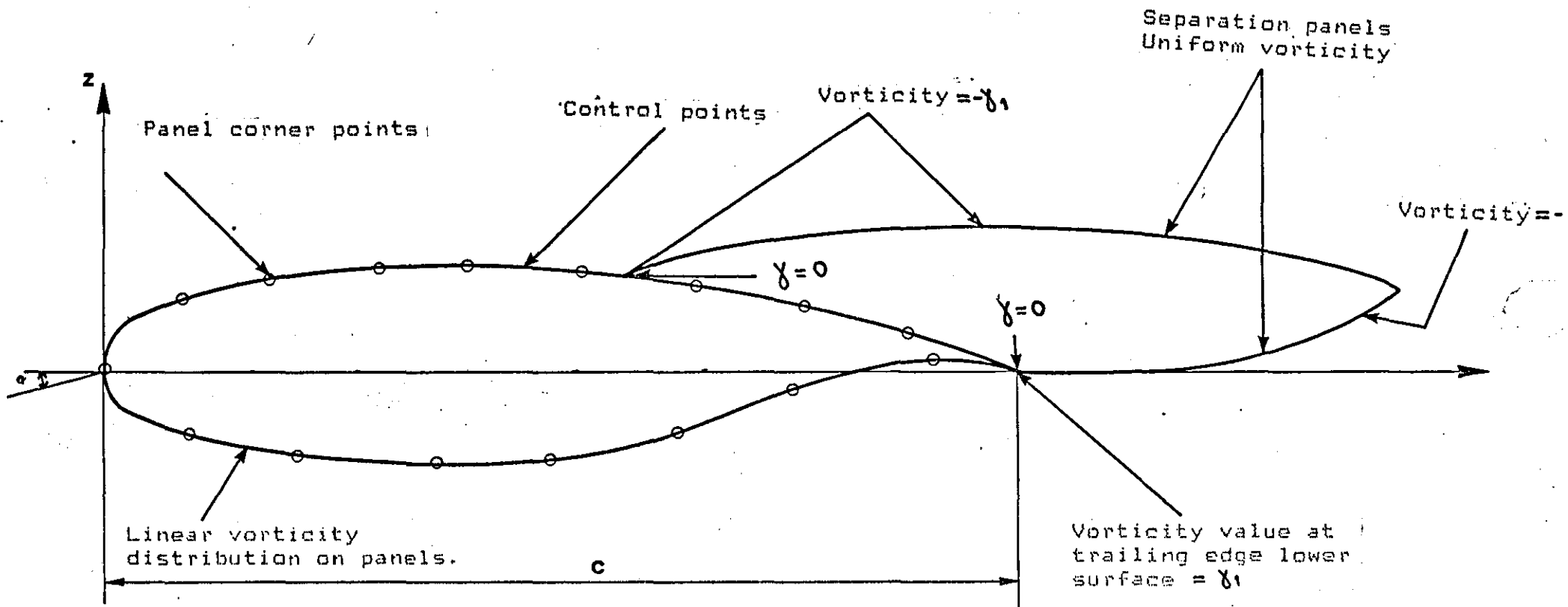
Quadratic curves
define initial
shape.

Free
stream.

Wake height.

Figure 11.1 Initial position of wake singularity sheets

Figure 11.2 Separated flow model



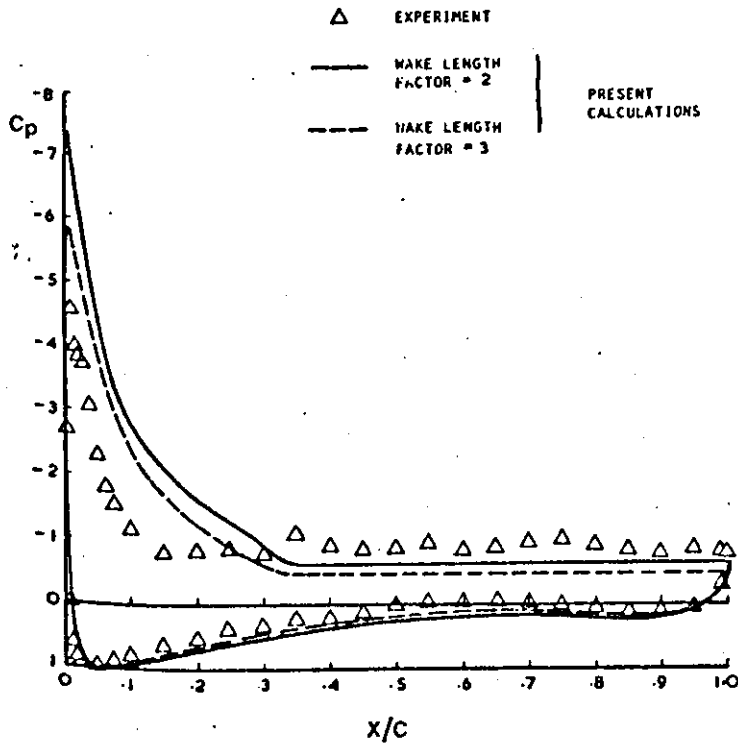


FIGURE 11.3a Pressure Distributions at Incidence
 21.14° ; GA(W)-1 Airfoil; $Re = 6.3 \times 10^6$

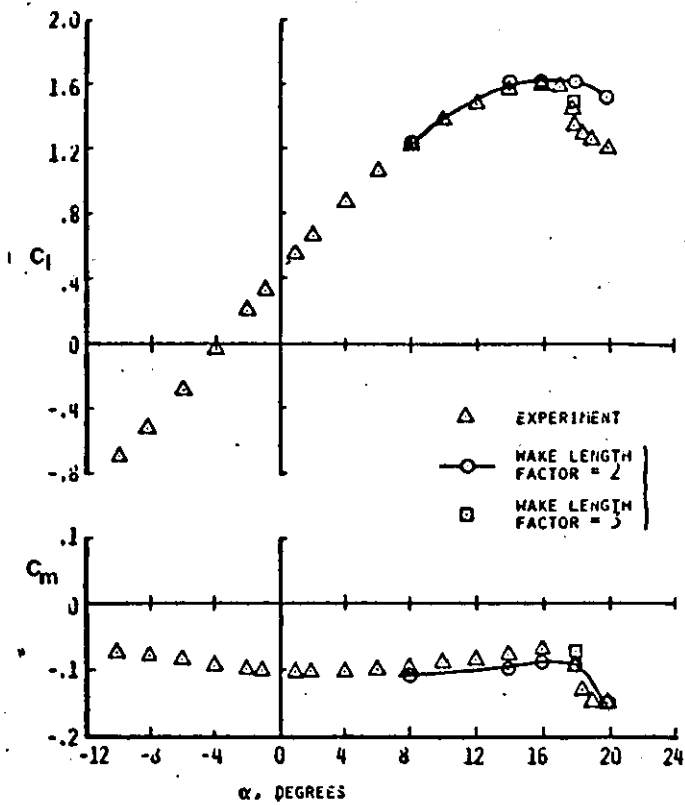


FIGURE 11.3b Lift and Pitching Moment Characteristics
at $Re = 2.1 \times 10^6$; GA(W)-1 Air-

OVERALL CALCULATION PROCEDURE

12.1 INTRODUCTORY COMMENT

The objective of the solution procedure is to find those particular distributions of surface velocity and boundary layer displacement thickness which simultaneously satisfy both the potential flow problem and the boundary layer problems. The desired solutions of surface velocity and displacement thickness, from which all other flow parameters can be computed, must be arrived at by an iterative procedure since the coupling of the flow problems is mathematically nonlinear. The computer program uses a cyclic iteration described below.

12.1.1 First Iteration

During this cycle the following computations are performed:

1. The position of the wake singularity sheet of each component is computed.
2. All wake singularities are set to zero.
3. A potential flow analysis is carried out.
4. A viscous analysis is performed based on velocities calculated in step 3. The analysis involves laminar and turbulent boundary layers.
5. The aerofoil inviscid velocities are corrected for viscous effect using $V_N = d(U\delta^*)/ds$ and $V_T = V_{INV} + V_{BL}$.
6. The tangential and normal velocity components at the middle of each wake panel, due to aerofoil singularities and free stream velocity are calculated.
7. A viscous wake analysis is then carried out.

8. The results of the wake viscous analysis are used to calculate the wake singularities at the end of each wake panel. These singularities (sources, vortices) are stored for the next iteration.
9. The wake position is relaxed and stored for the next iteration.

12.1.2 All Other Iterations

Here it is assumed that there is no interaction between the wake of the upstream aerofoil and the flap boundary layer. This is true in most cases when the aerofoil and flap are at the optimum setting. If this is not the case a final iteration is required in order to carry out a confluent boundary layer analysis.

During each of the rest of the iterations the following computations are performed.

1. A new boundary layer analysis using the corrected aerofoil surface velocities.
2. The velocities induced by the wake on the aerofoil are calculated.
3. The aerofoil surface velocities are corrected for viscosity and wake influence.
4. The tangential and normal velocities on each wake segment are calculated.
5. A wake viscous analysis is carried out.
6. The new wake singularities are calculated and stored for the next iteration.
7. The new wake position is calculated and stored for the next iteration.

Once all iterations have been completed the final set of aerofoil surface velocities is used to calculate the overall forces and pitching moment.

12.1.3 Final Iteration

This is required if a confluent boundary layer is found to exist. In this case the following procedure is adopted. The end of the potential

core is first located. A confluent boundary analysis is carried out starting from the end of the core. The velocities of the flap surface under the influence of the confluent boundary layer are corrected for viscosity using the displacement thickness of the wall layer only. The rest of the aerofoil velocities are left unchanged since the confluent boundary layer does not affect the boundary layer displacement thickness of the main aerofoil and the flap lower surface. The wake of the main aerofoil is assumed to be part of the confluent boundary layer. However a new viscous analysis is required for the wake of the flap. A new set of singularity strengths is calculated for the second wake and V_{W2} is recalculated. Finally the final set of aerofoil surface velocities is given by

$$V_T = V_{INV} + V_{VIS} + V_{W2}$$

Note V_{W1} is not considered since its effect is assumed to be taken into consideration in the confluent boundary layer analysis.

Once the final set of aerofoil velocities is known the overall forces and pitching moment are calculated.

12.2 AERODYNAMIC COEFFICIENTS

12.2.1 Lift and Moment Coefficients

The lift coefficient C_l of a multielement aerofoil is calculated by integrating the pressure and friction forces. The calculations are performed in the aerofoil axis system. The forces and the moment acting on a multielement aerofoil are (8):

- A Component of the force in direction of the x-axis, termed axial force
- N Component of the force in direction of the z-axis, termed normal force
- $M_{O,O}$ Pitching moment about the origin of the axis system, positive nose up.

Corresponding force and moment coefficients are defined by

$$C_A = \frac{A}{q_\infty C_{ref}} \quad C_N = \frac{N}{q_\infty C_{ref}} \quad C_{M_{O,O}} = \frac{M_{O,O}}{q_\infty C_{ref}^2} \quad (12.1)$$

where

$$q_\infty = \frac{1}{2} \rho_\infty U_\infty^2 \quad \text{and} \quad C_{ref} = \sum_{i=1}^{N_T} C_i$$

where N_T is the number of aerofoil elements.

Both the surface pressure P_s and the wall shear stress τ_w contribute to these forces and moment coefficients. Their contributions are calculated by discretizing the aerofoil geometry in exactly the same way as in the potential flow calculation, i.e., by replacing the actual aerofoil surface by a polygon. The corner points of the polygon (Diagram 12.1)

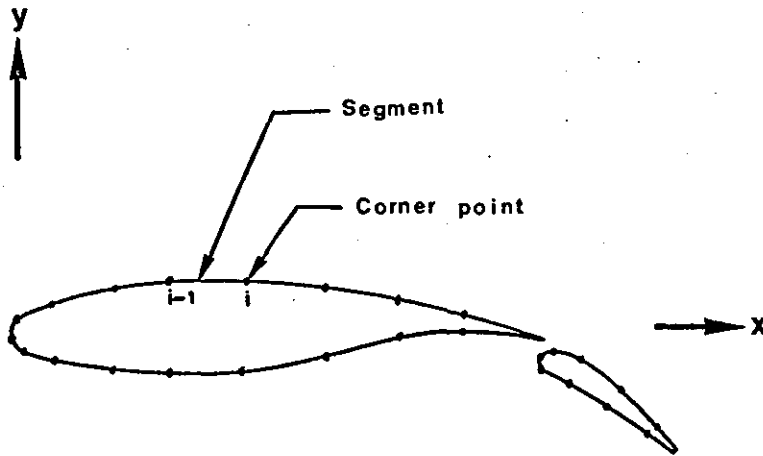


DIAGRAM 12.1: Discretization of the geometry

are positioned on the aerofoil surface and are identical with those used during the potential flow analysis. The contribution of the surface pressure to C_A, C_N and $C_{M_{O,O}}$ read

$$C_{AP} = \frac{1}{C_{ref}} \sum_{m=1}^{N_T} \sum_{i=2}^{N_m} C_P(z_i - z_{i-1}) \quad (12.2)$$

$$C_{NP} = - \frac{1}{C_{ref}} \sum_{m=1}^{N_T} \sum_{i=2}^{N_m} C_P(x_i - x_{i-1}) \quad (12.3)$$

$$(C_{M_{O,O}P}) = \frac{1}{C_{ref}^2} \sum_{m=1}^{N_T} \sum_{i=2}^{N_m} C_P [\bar{x}(x_i - x_{i-1}) + \bar{z}(z_i - z_{i-1})] \quad (12.4)$$

where

$$C_P = \frac{P - P_\infty}{\rho a_\infty^2} \quad (\text{at the panel midpoint})$$

$$\bar{x} = \frac{1}{2}(x_i + x_{i-1}), \quad \bar{z} = \frac{1}{2}(z_i + z_{i-1})$$

N_T = total number of aerofoil components

N_m = number of surface points on component m

The contribution of the wall shear to $C_A, C_N, C_{M_{O,O}}$ is

$$C_{AF} = \frac{1}{C_{ref}} \sum_{m=1}^{N_T} \sum_{i=2}^{N_m} C_f(x_i - x_{i-1}) \quad (12.5)$$

$$C_{NF} = \frac{1}{C_{ref}} \sum_{m=1}^{N_T} \sum_{i=2}^{N_m} C_f(z_i - z_{i-1}) \quad (12.6)$$

$$(C_{M_{O,O}})_F = \frac{1}{C_{ref}^2} \sum_{m=1}^{N_T} \sum_{i=2}^{N_m} C_f [\bar{z}(x_i - x_{i-1}) - \bar{x}(z_i - z_{i-1})] \quad (12.7)$$

In these equations C_f is the value of the skin friction coefficient

$$C_f = \frac{\tau_w}{\rho a_\infty^2} \quad (12.8)$$

at the midpoint of the i th panel. Note that for the purpose of computing the lift coefficient, the sign of C_f is reversed on the lower surface of each aerofoil component (8).

$$C_f = -C_f \quad (i \leq I_{stag}) \quad (12.9)$$

I_{stag} is the index of the stagnation point of the m th component.

Axial-force and pitching moment coefficients are obtained from

$$C_{AT} = C_{AP} + C_{AF} \quad C_{NT} = C_{NP} + C_{NF} \quad (C_{M_{O,O}})_T = (C_{M_{O,O}})_P + (C_{M_{O,O}})_F \quad (12.10)$$

The lift coefficient Cl follows from

$$Cl = C_{NT} \cos \alpha - C_{AT} \sin \alpha \quad (12.11)$$

where α is the angle of attack.

12.2.2 Drag Coefficient

The drag coefficient of the aerofoil is calculated using the Squire and Young formulae (Ref. 108). The drag coefficient C_{d_s} of each surface of each of the N_T aerofoil components is obtained from

$$C_{d_s} = 2 \frac{\theta}{C_{ref}} \left(\frac{V}{U_\infty} \right)^{1/2} (H+5) \quad (12.12)$$

where the boundary layer momentum thickness θ , the shape factor H , and the potential flow velocity are given by their values at the trailing edge point. In the case of a confluent boundary layer the chosen momentum thickness is that of the wall layer only since the outer wake portion of the confluent boundary layer is already represented by the upstream aerofoil (8). The total profile drag coefficient (C_d) of the high-lift aerofoil is the sum of the drag coefficients of the $2N_T$ surfaces.

METHOD VERIFICATION

Some results from applying the flow model to a number of cases are presented in this section. The method has been tested against theoretical and experimental results.

The potential flow model was first applied to Williams' two component aerofoil (Ref. 120) using a total of 95 panels. The calculated pressure distribution for aerofoil and flap are shown in Figure 13.1(a) and (b) respectively. Clearly the agreement between the exact and calculated pressures is very good. The calculated lift coefficients for the main aerofoil and flap are 2.9075 and 0.83046 respectively compared with the exact values 2.9065 and 0.8302 (Ref. 120, p.7).

To test the whole model two single element aerofoils, namely a GA(W)-1 and a NACA4412, and one two element aerofoil, a CA(W)-1 wing section with a 29%C Fowler flap were used.

Figure 13.2 shows the geometry of the GA(W)-1 aerofoil section. In Figure 13.3(a) to (c) the calculated pressures are compared with the experimental ones (Ref. 76) at three angles of attack. A total of 74 panels were used to represent the aerofoil surface. The calculated pressures were obtained after 3 iterations. The Reynolds number for both theory and experiment is 6×10^6 . Figure 13.3(a) shows the pressure distribution on the aerofoil at $\alpha = -4.11$ degrees. The agreement between theory and experiment is good. Figure 13.3(b) and 13.3(c) show the pressure distributions at $\alpha = 4.17$ and 12.01 degrees respectively. Generally speaking the theory predicts

the correct pressure distribution. However due to trailing edge separation in the case when $\alpha=12.04$ there is some disagreement between the experimental and calculated values near the upper surface trailing edge region.

Figure 13.4(a) gives the geometry of the GA(W)-1 aerofoil with a 29% Fowler flap. The flap pivot point locations are given in Figure 13.4(b) A total of 76 panels were used in this case; 48 panels were distributed on the surface of the main aerofoil and 28 on the flap. The extensive experimental results of Reference 135 were employed to test the theory. The calculated results presented here were obtained after 3 to 4 iterations.

Figure 13.5(a) compares the experimental results with the calculated ones for both aerofoil and flap. The angle of attack is 0° and the flap deflection is 10° , the Reynolds number is 2.2×10^6 . The agreement is generally good with the exception of the lower surface of the aerofoil near the trailing edge and the flap upper surface. The same trend is present in Figure 13.5(b) where the angle of attack is 5° , however, the disagreement is less in this case. Figure 13.5(c) and (d) present a more difficult test for the model. In this case the flap deflection has been increased to 30° for the same angles of attack. The agreement between experimental and calculated values is good in both cases. Finally Figure 13.5(e) presents the pressure distribution on both aerofoil and flap at an angle of attack of 5° with flap deflection of 40° . Once again good agreement between experiment and theory is demonstrated.

Figure 13.6 presents the calculated and experimental lift curves for the GA(W)-1 aerofoil section at a Reynolds number of 2.7×10^6 . Clearly the inviscid curve over-estimates CL. The CL curve corrected for viscosity and wake effects is in good agreement with the experimental one (Ref. 76) up to an angle of attack of about 7 degrees. Separation

reduces the predicted value of CL considerably at higher angles of attack.

The large camber of the NACA4412 aerofoil presents a difficult test for the model. Figure 13.7 compares the theoretical with the experimental (Ref.1) lift curves. Once again the agreement is good up to about 8 degrees of incidence when separation effects dominate and reduce the theoretical values of the lift coefficient.

A wake relaxation is presented in Figure 13.8. The results are obtained using a GA(W)-1 section at $\alpha=0^\circ$ and $R_e=6 \times 10^6$. Five iterations were used to investigate the shape of the wake. Clearly the wake position has nearly converged after the 3rd iteration. The wake, which is one aerofoil chord long and is modelled using 18 panels, is shown to curve rapidly near the aerofoil trailing edge. The wake trailing edge is shown to have turned back to the free stream direction after the third iteration. Table 13.1 shows the normal and tangential wake velocities obtained.

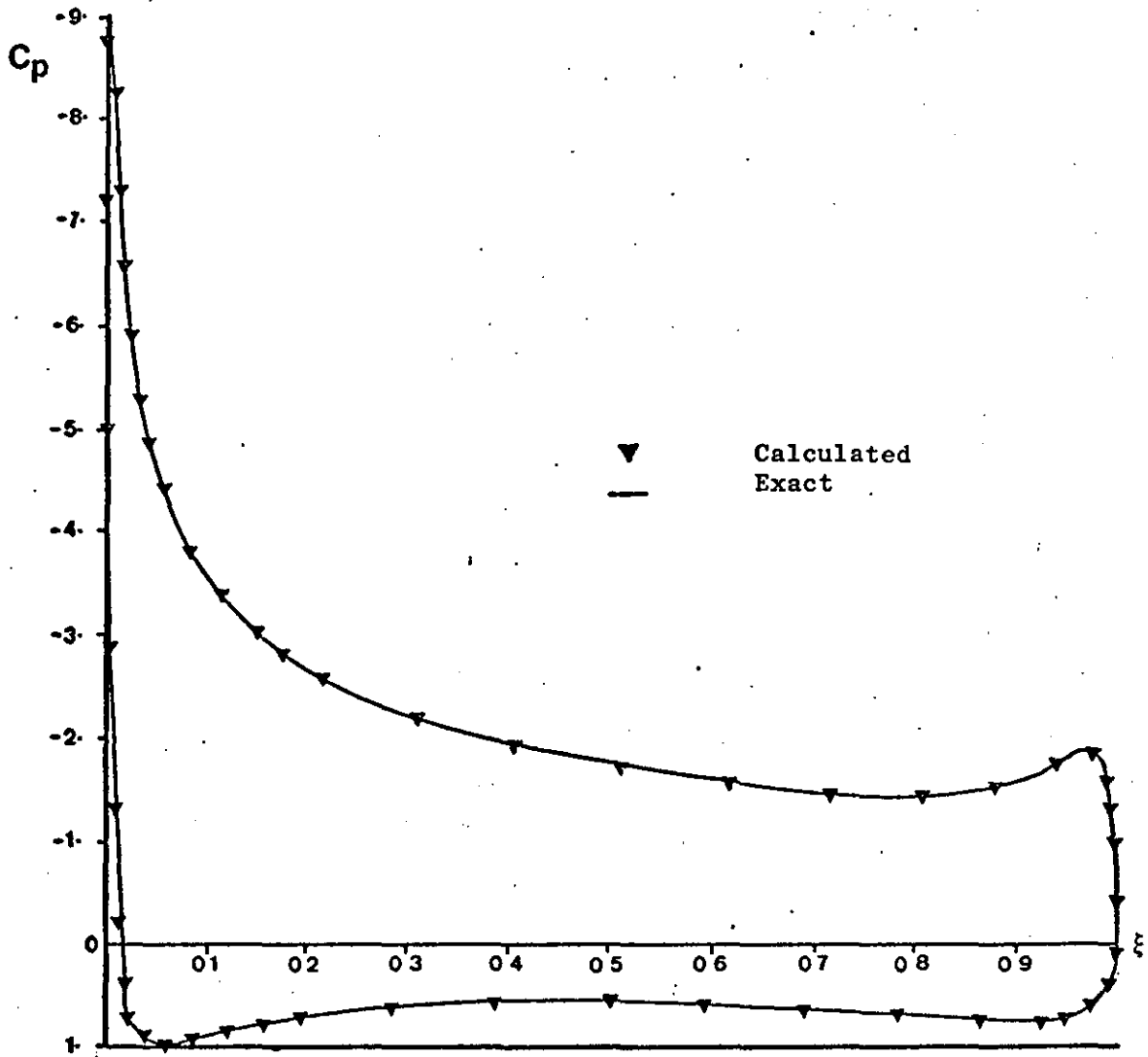
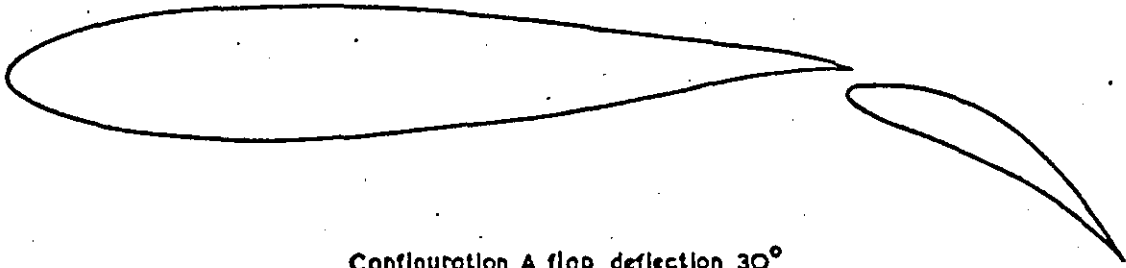
Using the same aerofoil at an incidence of 6° , the results of five iterations are shown. It can be seen that the value of V_N has become practically zero on most panels after the third iteration.

Finally some typical boundary layer analysis results are presented in Figure 13.9.

To summarise, a number of theoretical and experimental results have been used to test the flow model developed. In general the calculated results are in very good agreement with both theory and experiment. Only a small number of panels is required to model the aerofoil element(s) and the wake(s). This makes the method very economical to use. In most cases only 90 sec. of C.P.U. were required on the PRIME 400 computer to carry out 5 iterations.

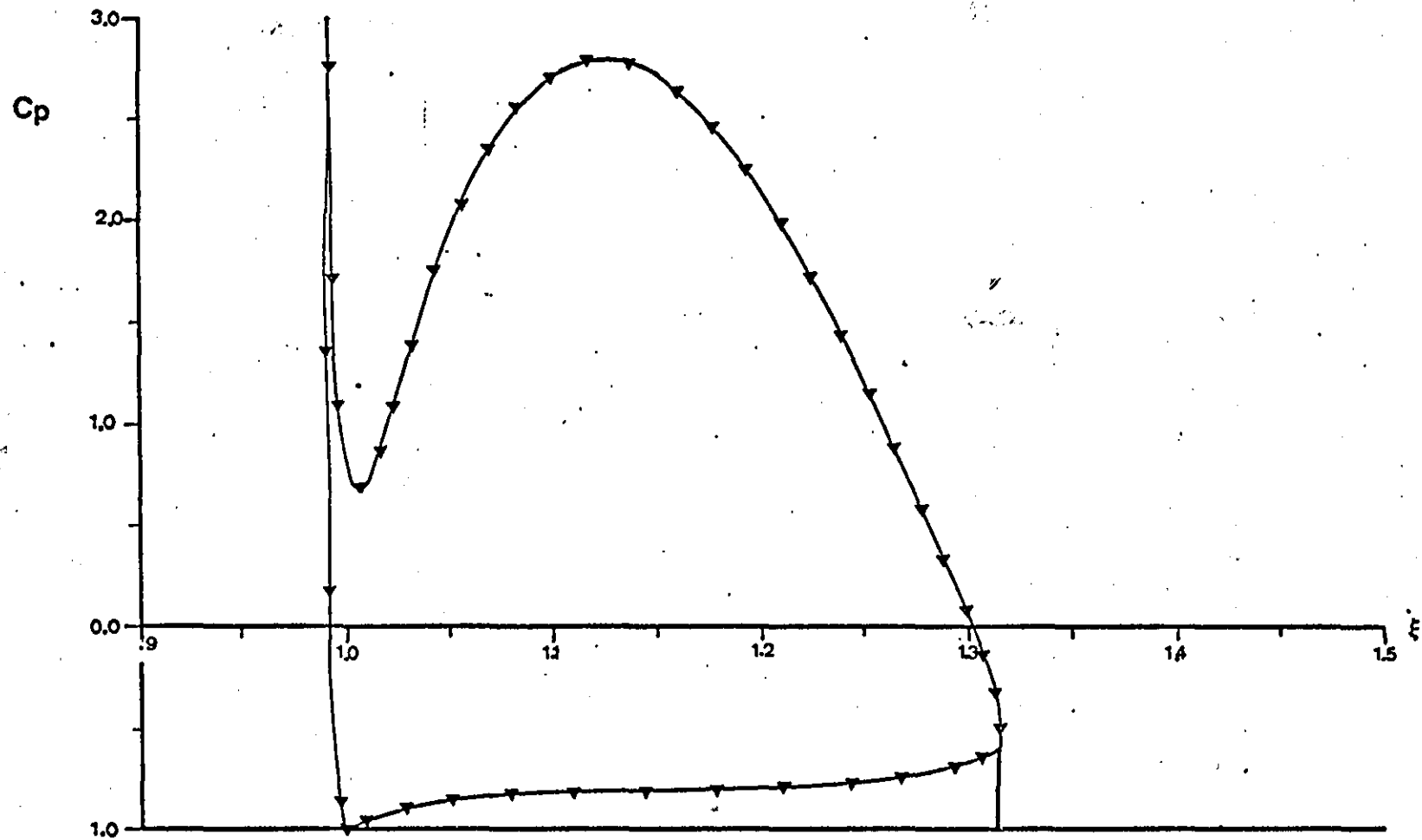
ITERATION 1		2		3		4		5	
VN1	VT1	VN1	VT1	VN1	VT1	VN1	VT1	VN1	VT1
-0.1060	0.8799	-0.0702	0.8061	0.0109	0.7832	0.0036	0.7800	0.0071	0.7838
-0.0688	0.9172	-0.1169	0.8365	-0.0010	0.8056	-0.0040	0.7860	0.0065	0.7676
-0.0471	0.9358	-0.0894	0.8977	-0.0201	0.9029	-0.0265	0.8923	-0.0179	0.8663
-0.0316	0.9476	-0.0459	0.9201	-0.0004	0.9537	-0.0173	0.9807	-0.0231	0.9899
-0.0196	0.9558	-0.0241	0.9286	0.0182	0.9512	0.0103	0.9765	0.0026	0.9952
-0.0099	0.9618	-0.0123	0.9336	0.0172	0.9495	0.0150	0.9601	0.0134	0.9659
-0.0018	0.9665	-0.0048	0.9372	0.0125	0.9509	0.0087	0.9587	0.0085	0.9577
0.0052	0.9702	0.0006	0.9400	0.0084	0.9525	0.0049	0.9609	0.0033	0.9622
0.0113	0.9731	0.0048	0.9424	0.0052	0.9539	0.0024	0.9624	0.0018	0.9658
0.0166	0.9756	0.0085	0.9444	0.0027	0.9553	0.0007	0.9633	0.0012	0.9668
0.0213	0.9776	0.0119	0.9462	0.0007	0.9565	-0.0002	0.9639	0.0006	0.9669
0.0255	0.9793	0.0153	0.9477	-0.0010	0.9575	-0.0009	0.9645	0.0001	0.9669
0.0293	0.9808	0.0188	0.9490	-0.0026	0.9582	-0.0012	0.9651	-0.0001	0.9672
0.0328	0.9820	0.0227	0.9502	-0.0042	0.9587	-0.0014	0.9654	-0.0000	0.9677
0.0359	0.9831	0.0273	0.9513	-0.0060	0.9586	-0.0011	0.9649	0.0007	0.9672
0.0388	0.9841	0.0334	0.9523	-0.0097	0.9569	-0.0016	0.9611	0.0012	0.9621
0.0414	0.9849	0.0435	0.9534	-0.0217	0.9559	-0.0055	0.9582	-0.0008	0.9574
0.0439	0.9856	0.0665	0.9755	-0.0252	0.9894	-0.0103	0.9936	-0.0027	0.9948

TABLE 13.1: Normal and Tangential wake velocities
(GA(W) -1, Incidence 6°)



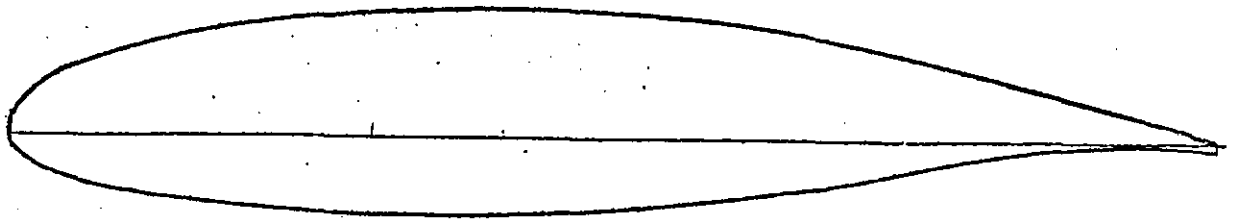
(a) Main Aerofoil

FIGURE 13.1 Comparison of exact and calculated pressure distributions for the two element aerofoil shown above (Incidence = 0 deg)



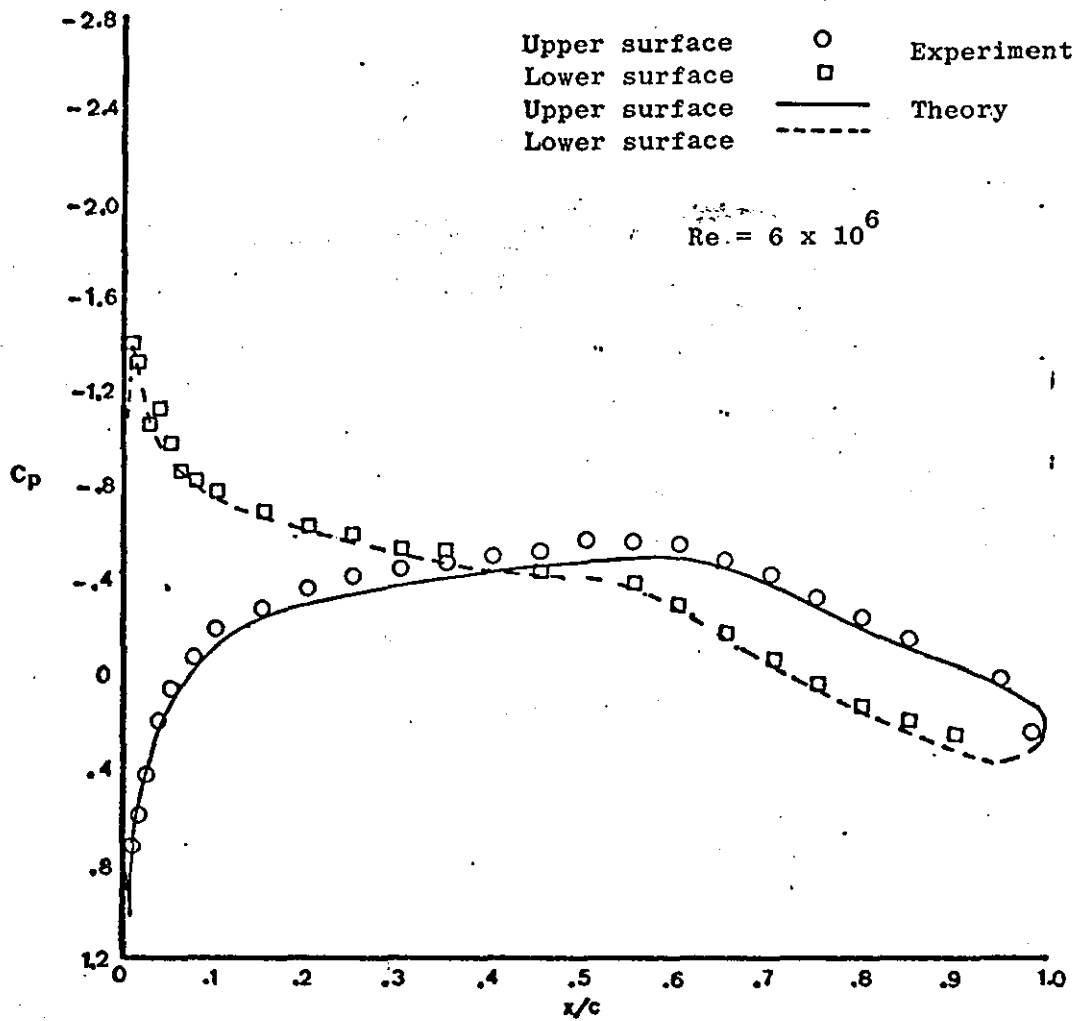
(b) Flap

FIGURE 13.1 Continued



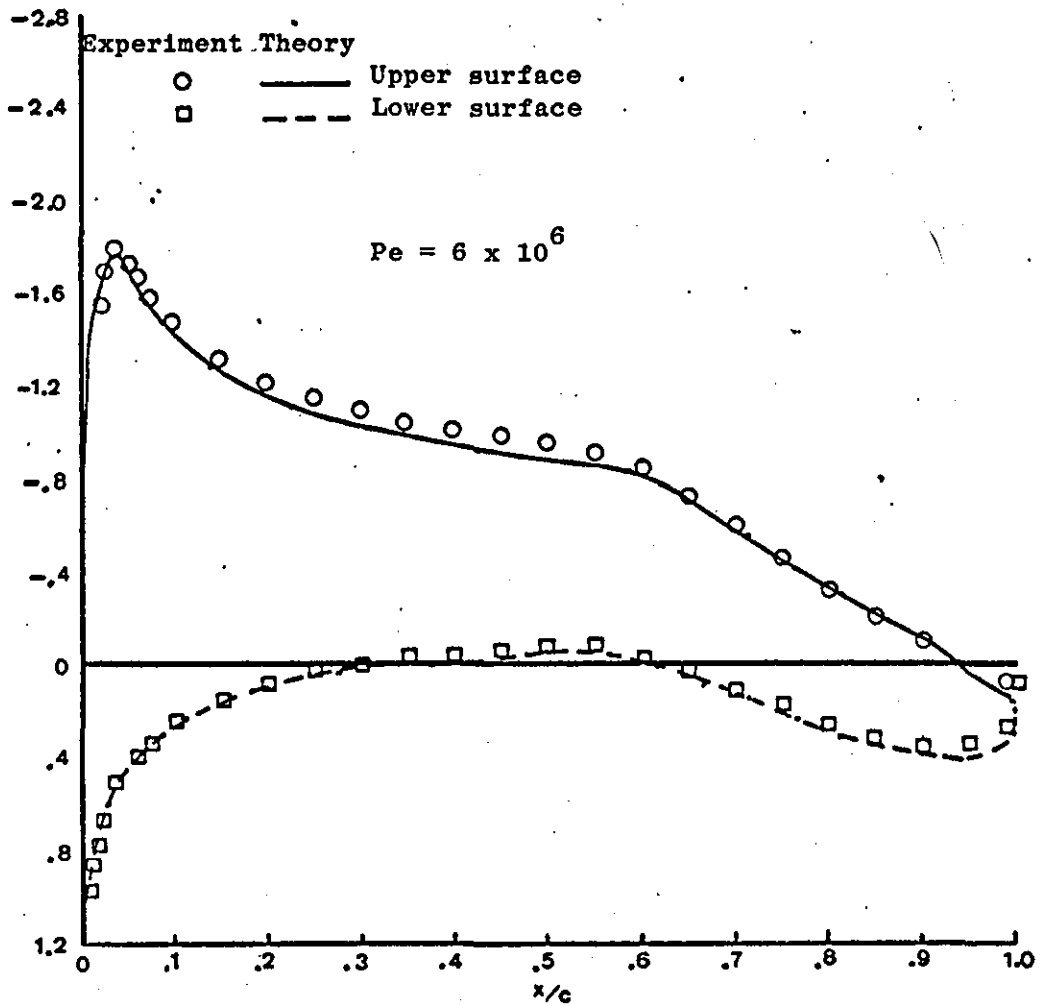
Upper Surface		Lower Surface	
X / c	Z / c	X / c	Z / c
0.00000	0.00000	0.00000	0.00000
.00200	.01300	.00200	-.00930
.00500	.02040	.00500	-.01380
.01250	.03070	.01250	-.02050
.02500	.04170	.02500	-.02690
.03750	.04965	.03750	-.03190
.05000	.05589	.05000	-.03580
.07500	.06551	.07500	-.04210
.10000	.07300	.10000	-.04700
.12500	.07900	.12500	-.05100
.15000	.08400	.15000	-.05430
.17500	.08840	.17500	-.05700
.20000	.09200	.20000	-.05930
.25000	.09770	.25000	-.06270
.30000	.10160	.30000	-.06450
.35000	.10400	.35000	-.06520
.40000	.10491	.40000	-.06490
.45000	.10445	.45000	-.06350
.50000	.10258	.50000	-.06100
.55000	.09910	.55000	-.05700
.57500	.09668	.57500	-.05400
.60000	.09371	.60000	-.05080
.62500	.09006	.62500	-.04690
.65000	.08599	.65000	-.04280
.67500	.08136	.67500	-.03840
.70000	.07634	.70000	-.03400
.72500	.07092	.72500	-.02940
.75000	.06513	.75000	-.02490
.77500	.05907	.77500	-.02040
.80000	.05286	.80000	-.01600
.82500	.04646	.82500	-.01200
.85000	.03988	.85000	-.00860
.87500	.03315	.87500	-.00580
.90000	.02639	.90000	-.00360
.92500	.01961	.92500	-.00250
.95000	.01287	.95000	-.00260
.97500	.00609	.97500	-.00400
1.00000	-.00070	1.00000	-.00800

FIGURE 13.2 GA(W)-1 Aerofoil Coordinates

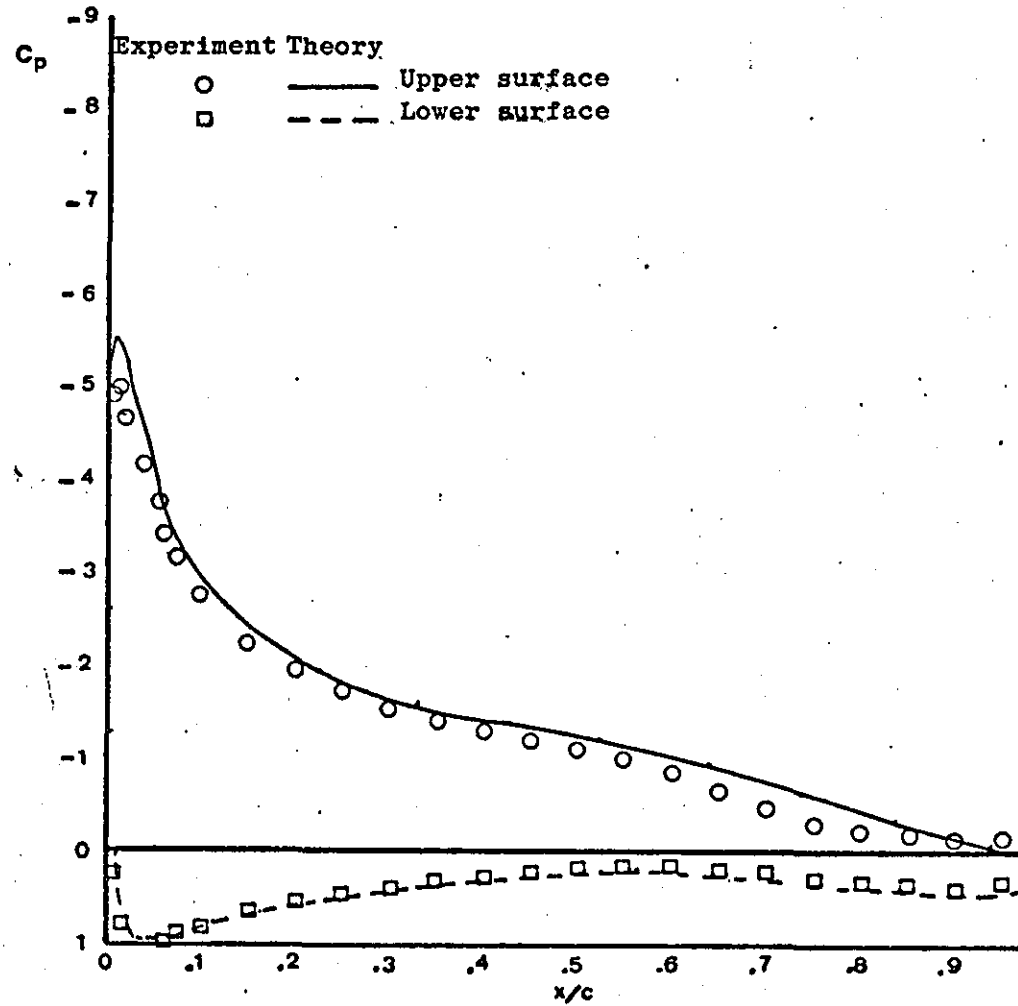


(a) Incidence = -4.11 deg

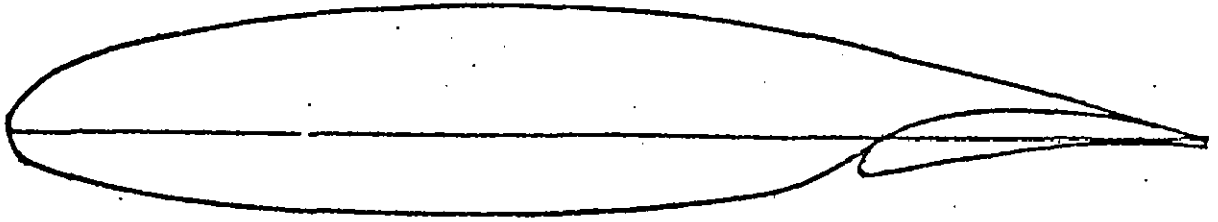
FIGURE 13.3 Comparison of experimental and calculated pressures for a GA(W) -1 aerofoil section



(b) Incidence = 4.17 deg



(c) Incidence = 12.01 deg



29%C Fowler Flap Coordinates

Upper Surface		Lower Surface	
X_f/c	Z_f/c	X_f/c	Z_f/c
0.00000	-.02350	0.00000	-.02350
.00030	-.02000	.00100	-.02700
.00200	-.01790	.00200	-.02880
.00400	-.01550	.00400	-.03000
.00800	-.01130	.00800	-.03100
.01200	-.00780	.01200	-.03040
.01800	-.00330	.02000	-.02880
.02300	.00000	.03000	-.02700
.02800	.00230	.05000	-.02350
.03800	.00700	.07000	-.01980
.04800	.01100	.09000	-.01600
.05800	.01410	.11000	-.01300
.06800	.01680	.13000	-.01000
.07800	.01900	.15000	-.00770
.08800	.02070	.17000	-.00580
.09800	.02180	.19000	-.00360
.10800	.02230	.21000	-.00270
.11800	.02280	.23000	-.00280
.12800	.02300	.25000	-.00350
.13800	.02340	.27000	-.00500
.14800	.02280	.29000	-.00800
.15800	.02230		
.16800	.02190		
.19000	.01980		
.21000	.01680		
.23000	.01380		
.25000	.00980		
.27000	.00590		
.29000	-.00070		

Nose Radius = .0075c

Nose Radius Location ($X_f/c, Z_f/c$) = (.0075, -.0235)

FIGURE 13.4a 29%c Fowler Flap Configuration

29&C Flap Pivot Point Locations - Computer Design Settings

δ_f	x/c	z/c
0° (Nested)	.730	-.040
10°	.880	-.061
15°	.900	-.055
20°	.920	-.049
25°	.930	-.046
30°	.940	-.043
40°	.950	-.040

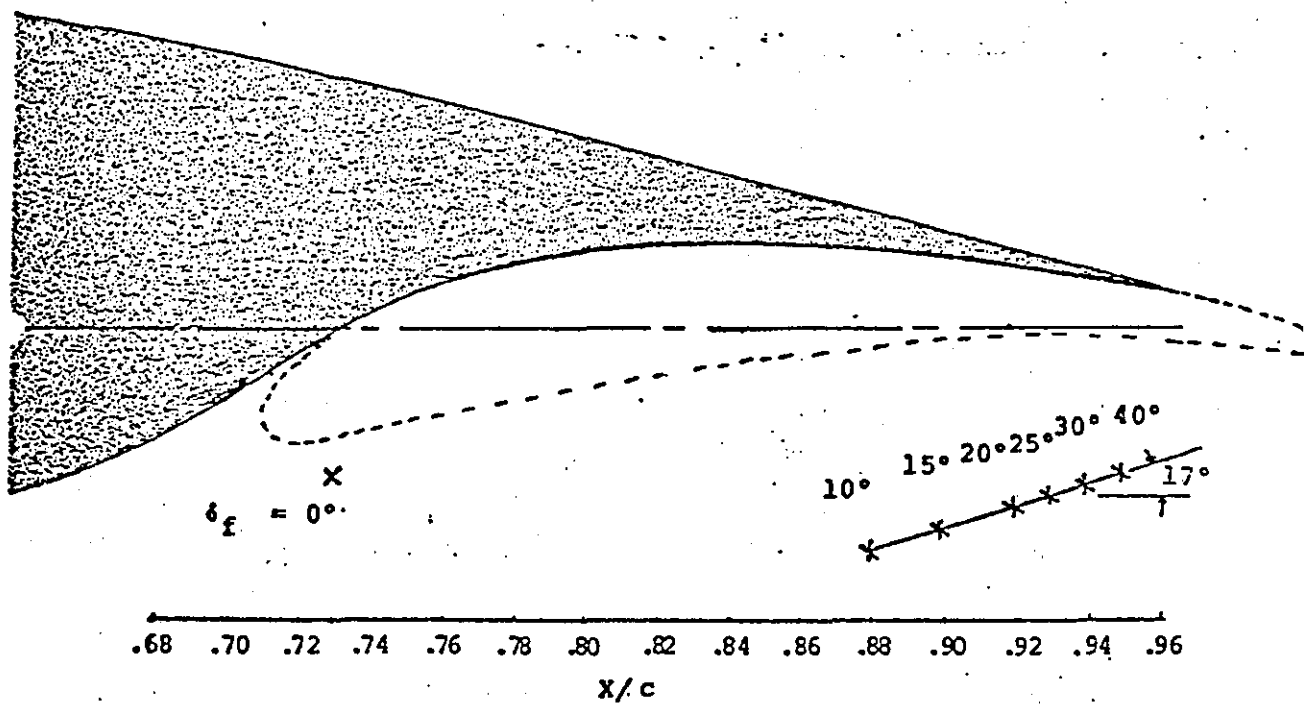
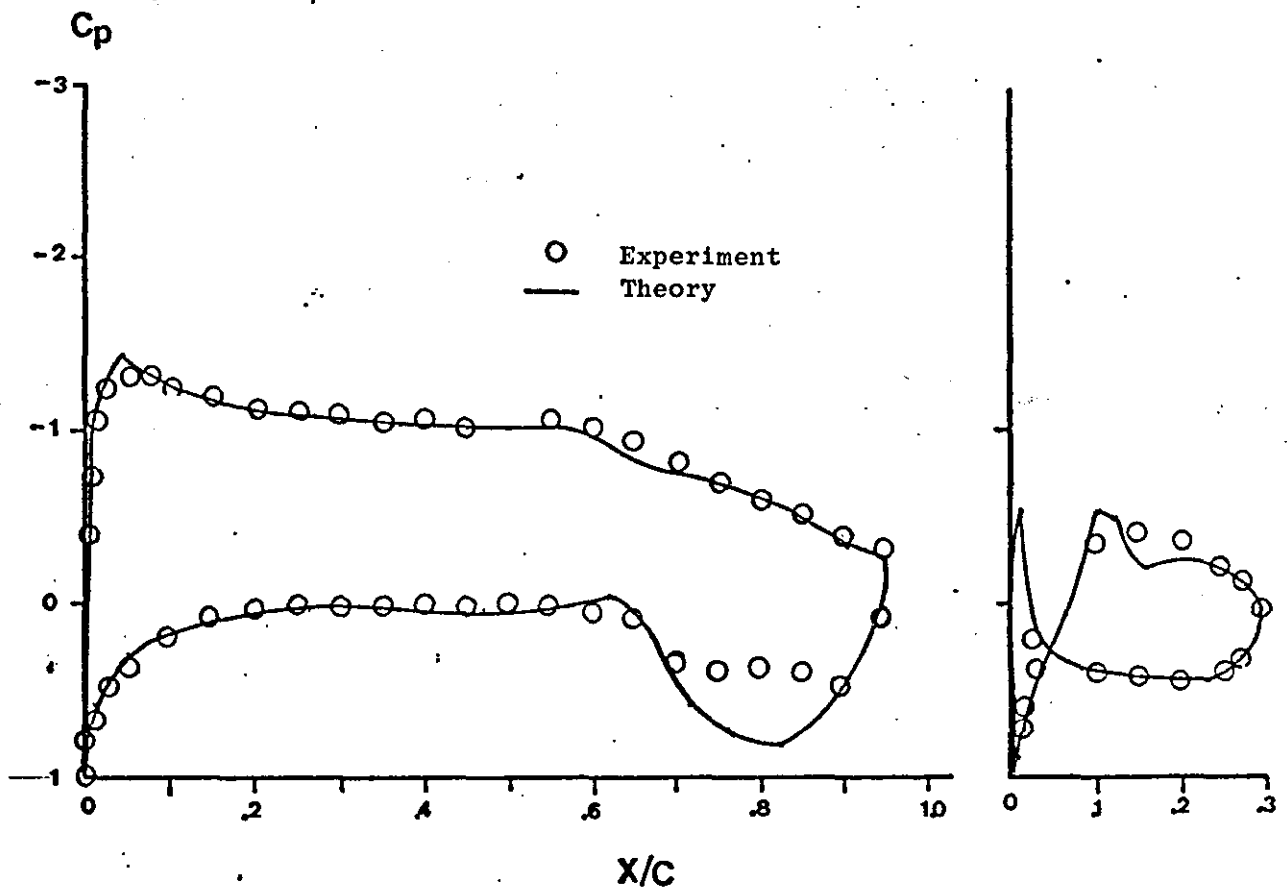
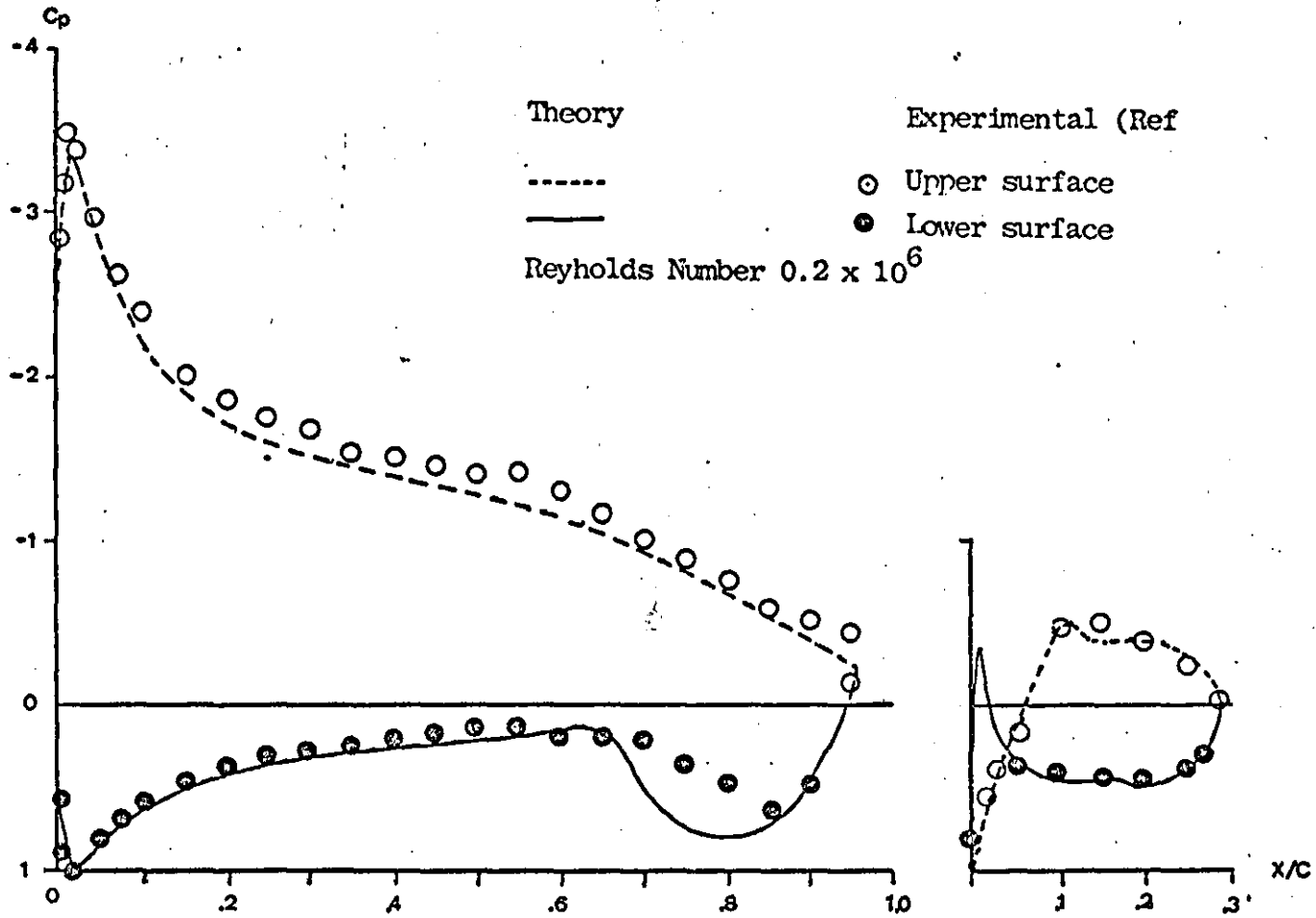


FIGURE 13.4b Flap Pivot Point Locations



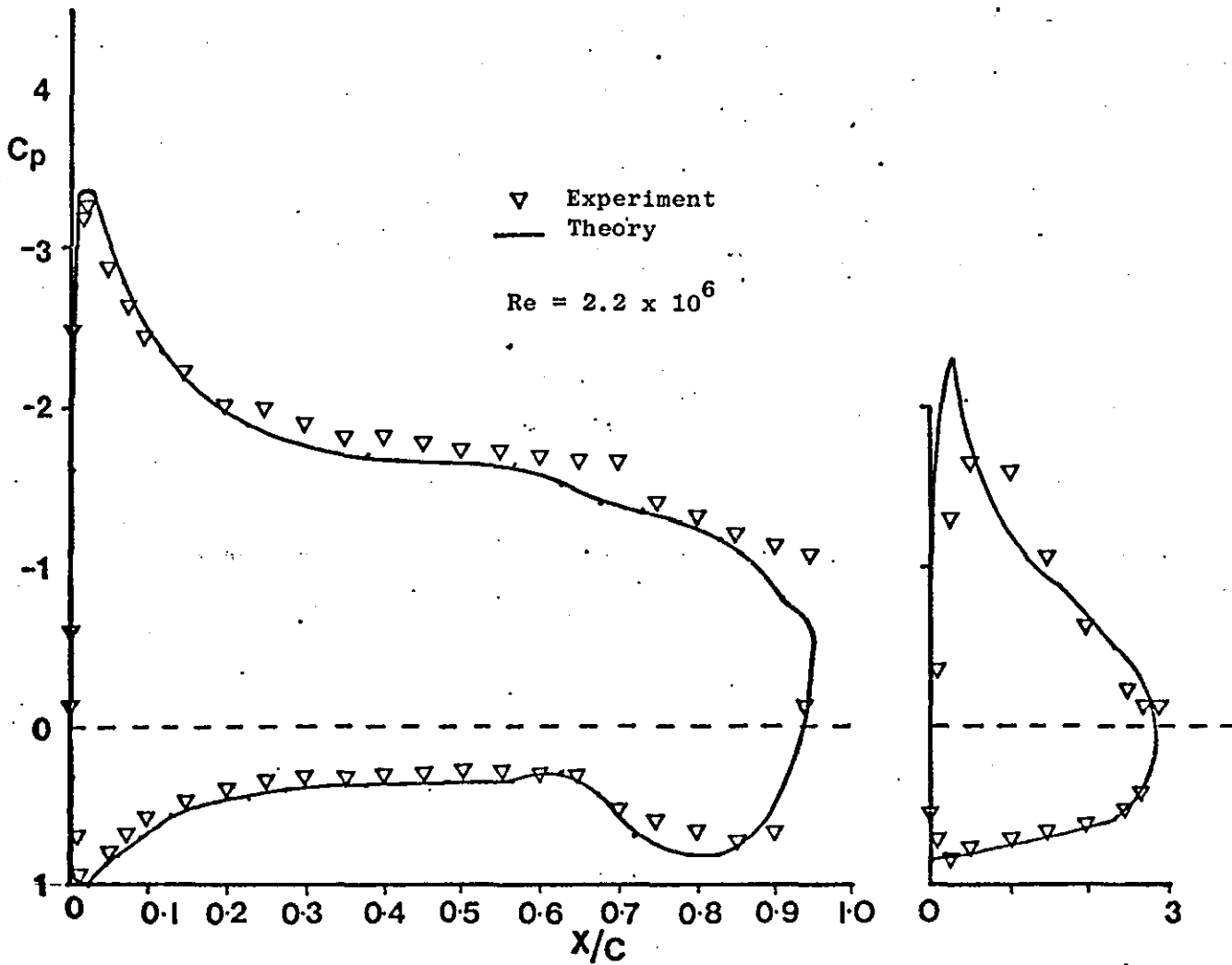
(a) Incidence = 0 deg - Flap deflection = 10 deg

FIGURE 13.5 Comparison of calculated and experimental pressures for a GA(W) -1 aerofoil with 29% c Fowler flap



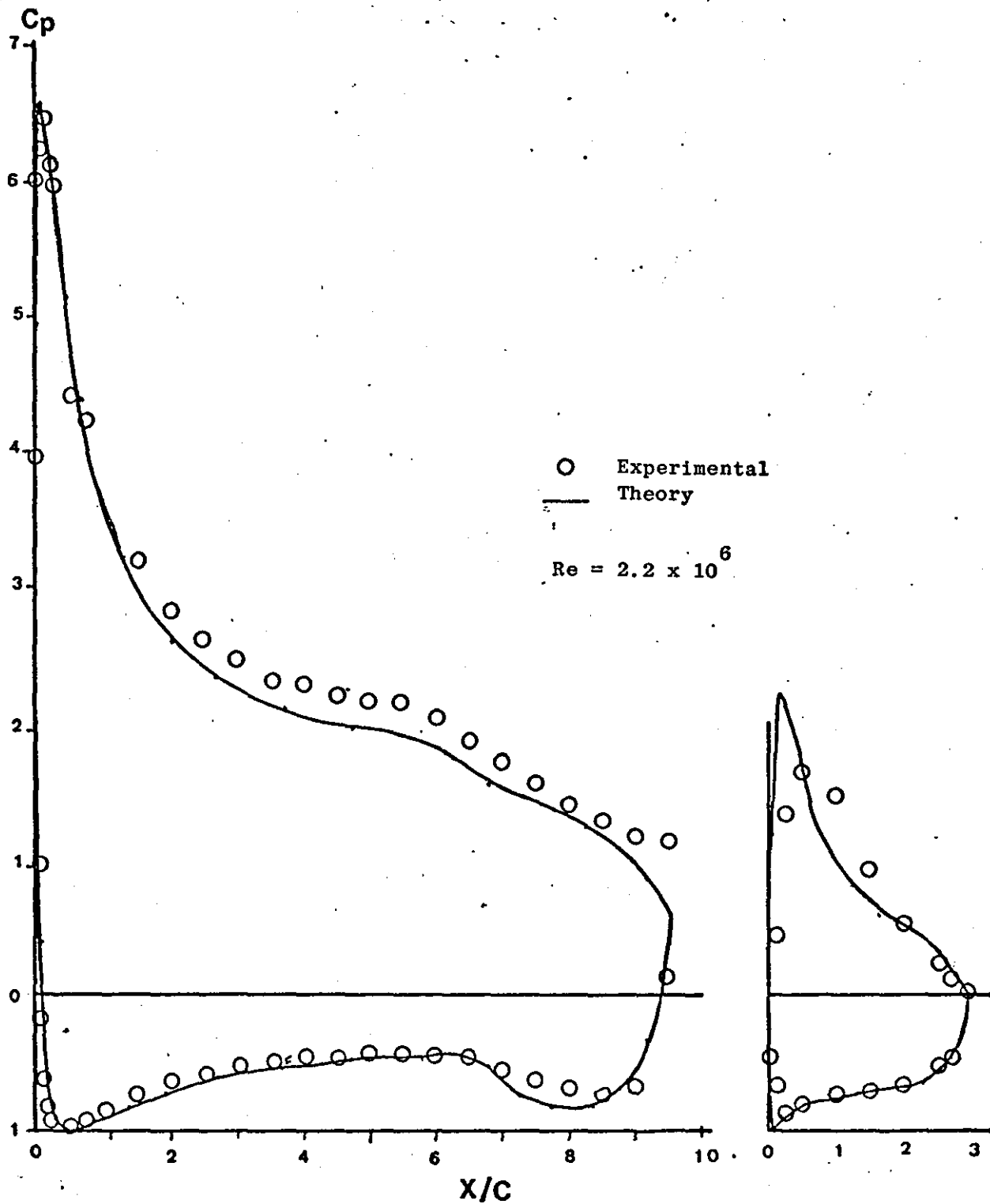
(b) Incidence = 5 deg, Flap deflection = 10 deg

FIGURE 13.5 Continued



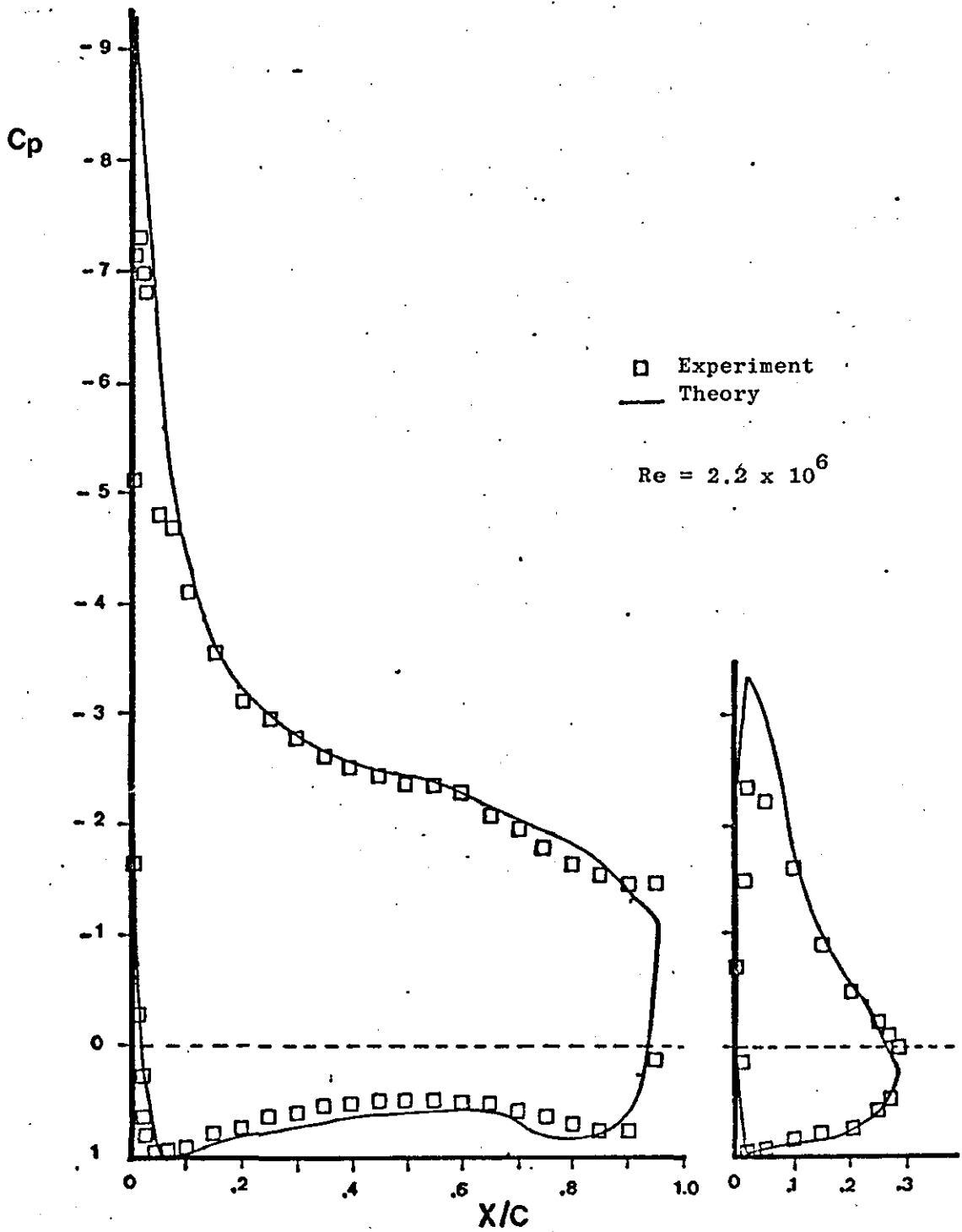
(c) Incidence = 0 deg - Flap deflection = 30 deg

FIGURE 13.5 Continued



(d) Incidence = 5 deg - Flap deflection = 30 deg

FIGURE 13.5 Continued



(e) Incidence = 5 deg - Flap deflection = 40 deg

FIGURE 13.5 Continued

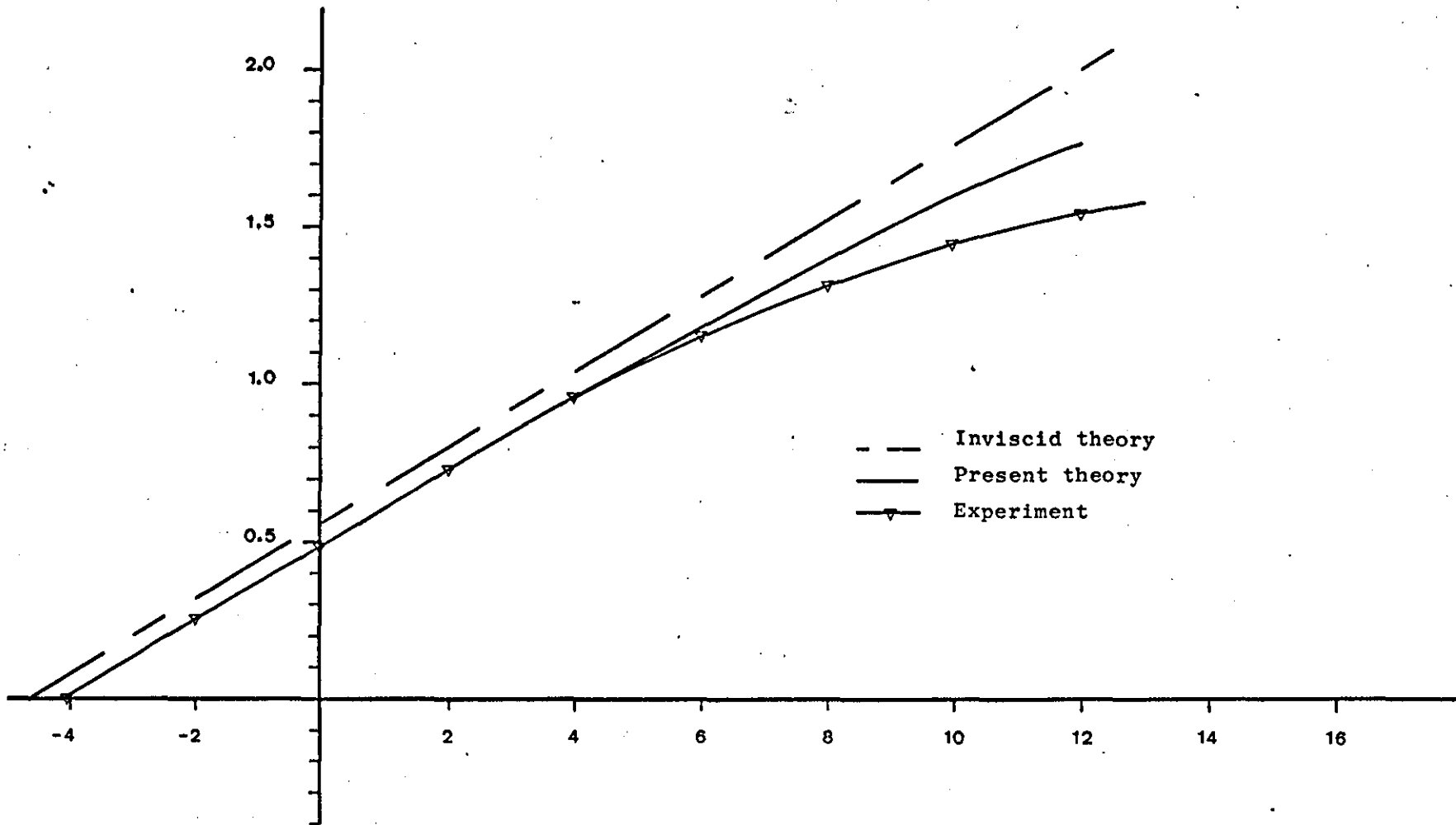


FIGURE 13,6 Lift coefficient versus angle of attach for the GA(W) -1 aerofoil section ($Re = 2.7 \times 10^6$)

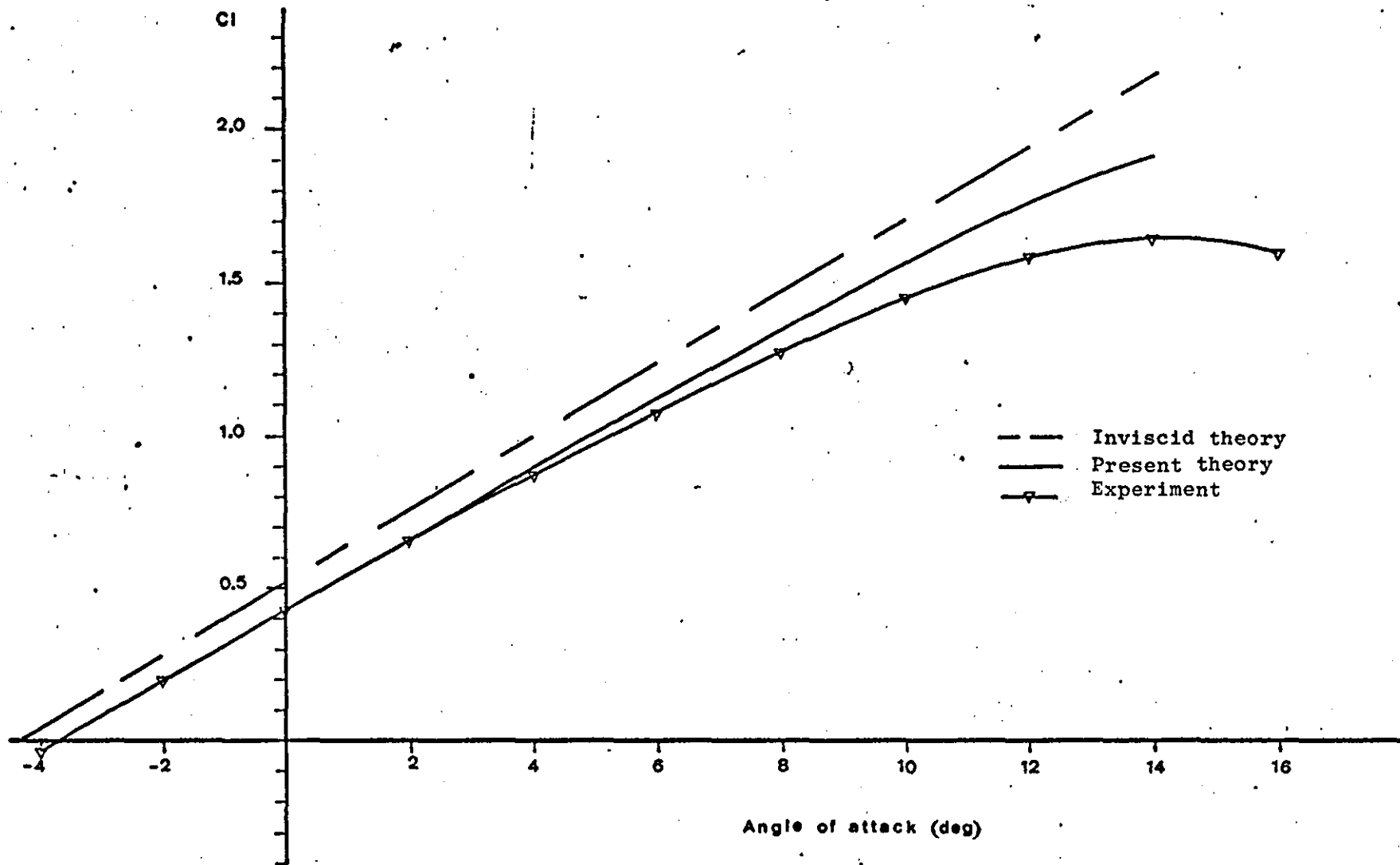


FIGURE 13.7 Lift coefficient versus angle of attack
for a NACA4412 airfoil section ($Re = 6 \times 10^6$)

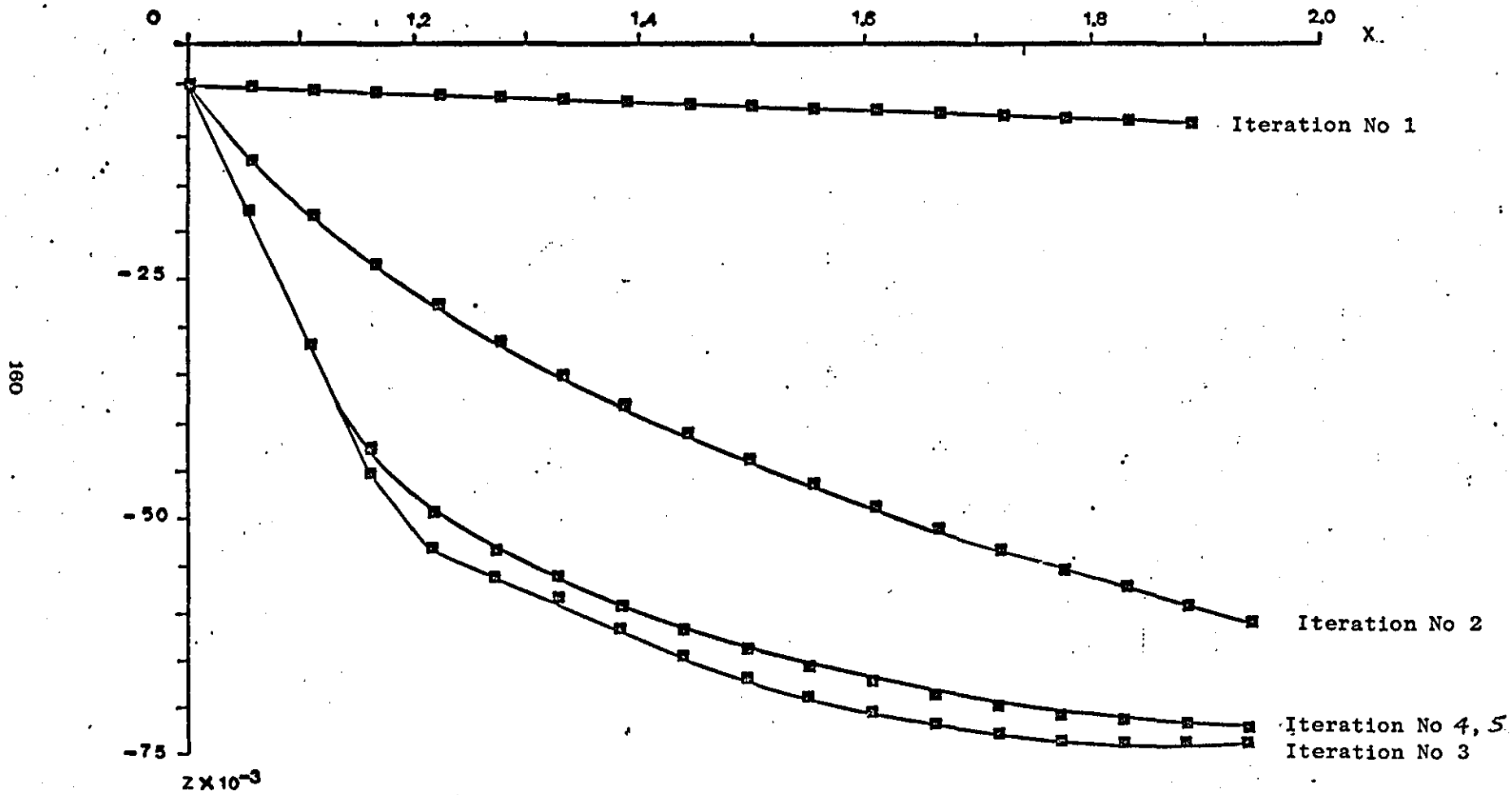
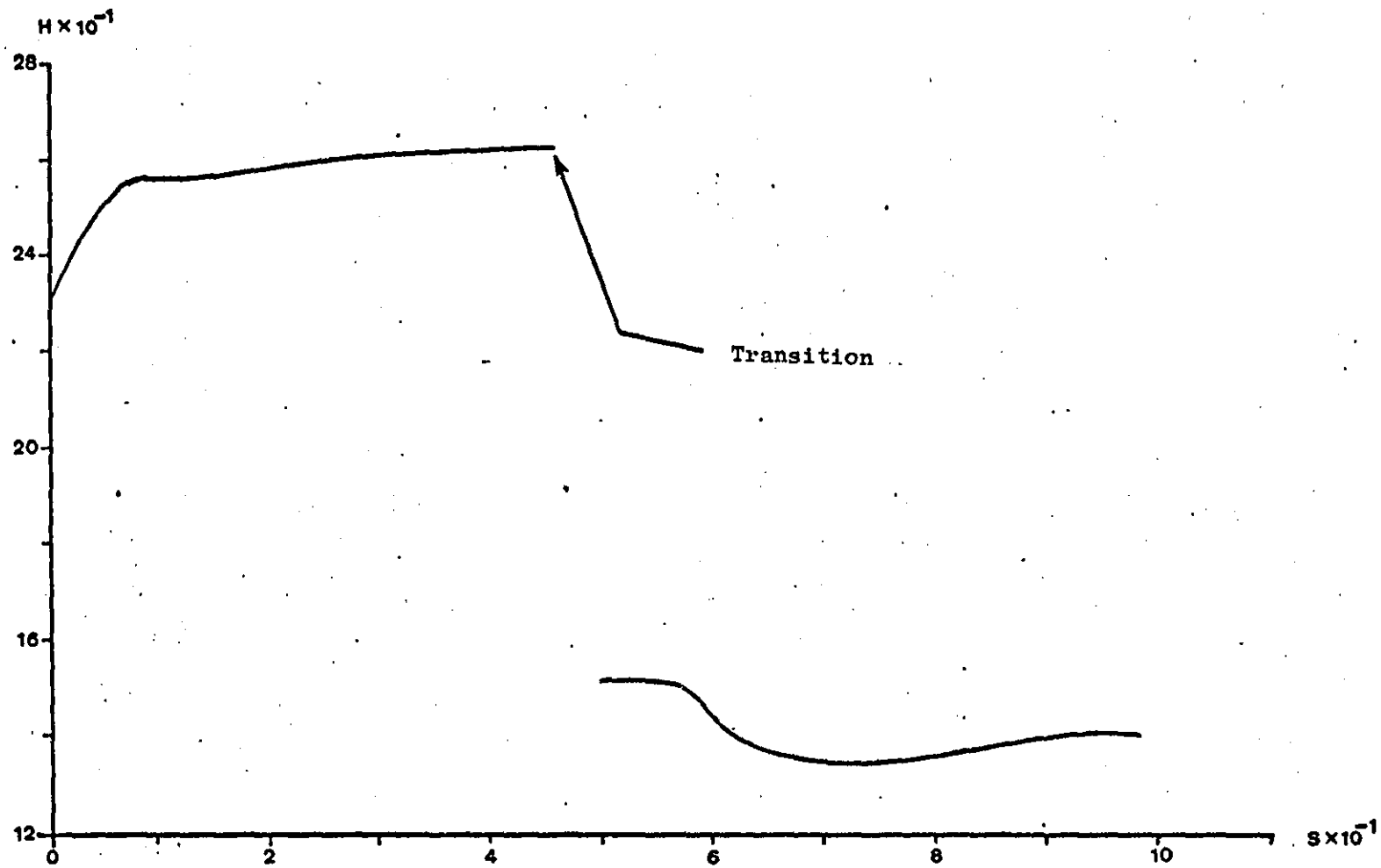
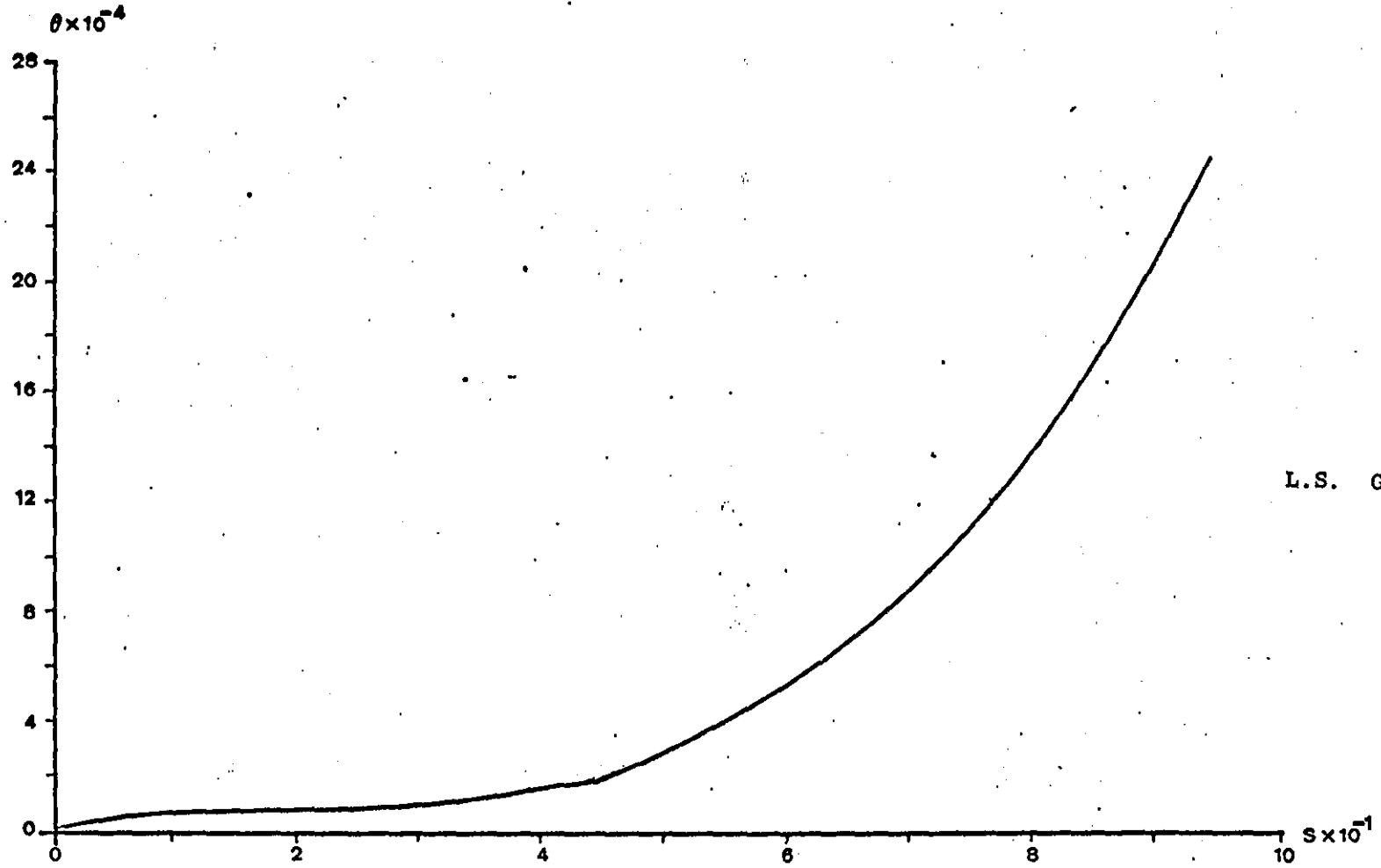


FIGURE 13.8 Wake relaxation



(a) Plot of shape parameter versus distance along aerofoil lower surface

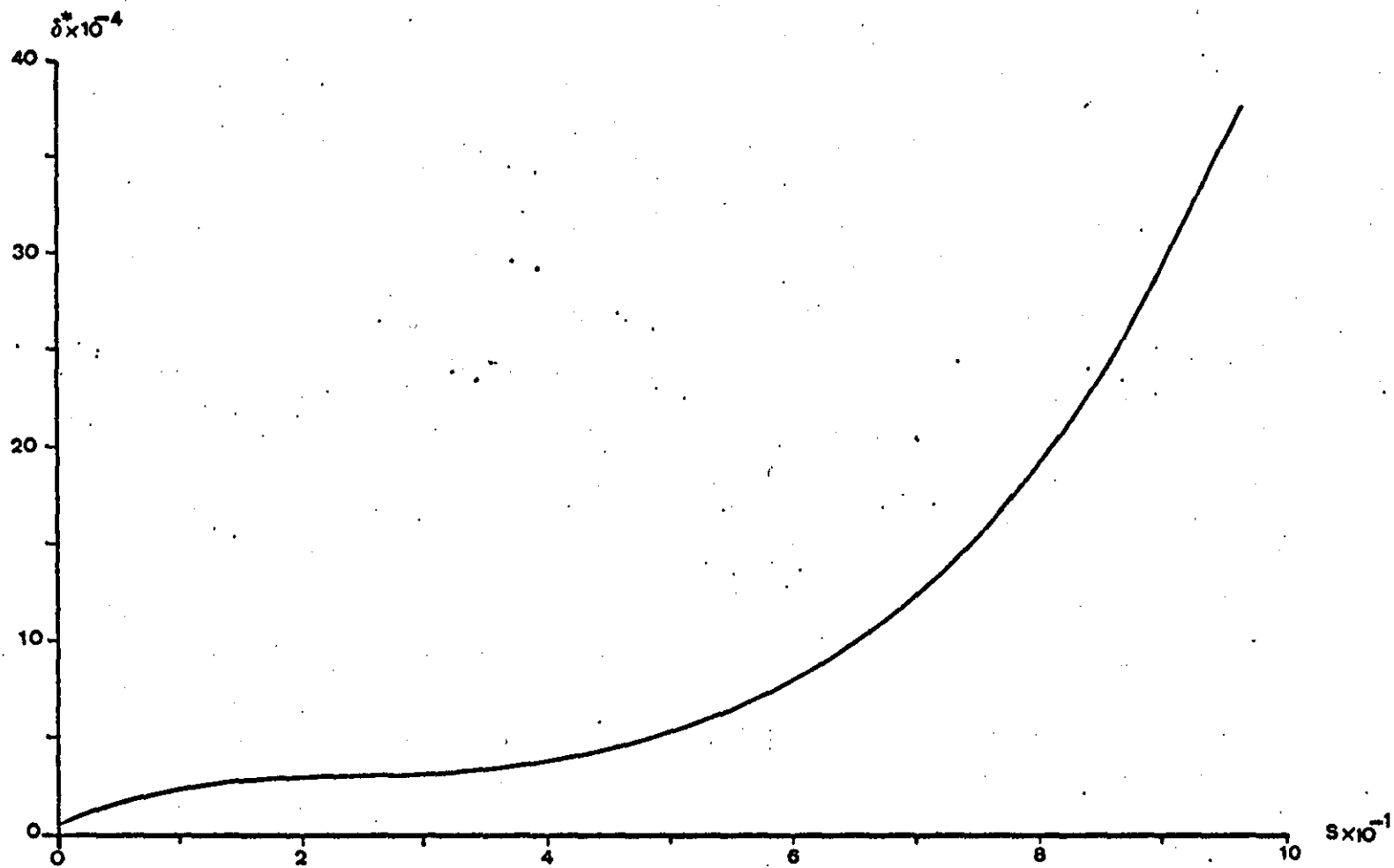
FIGURE 13.9 Boundary layer culailation (GA(W) -1 aerofoil section - Incidence 0 deg - $Re = 6 \times 10^6$)



L.S. G

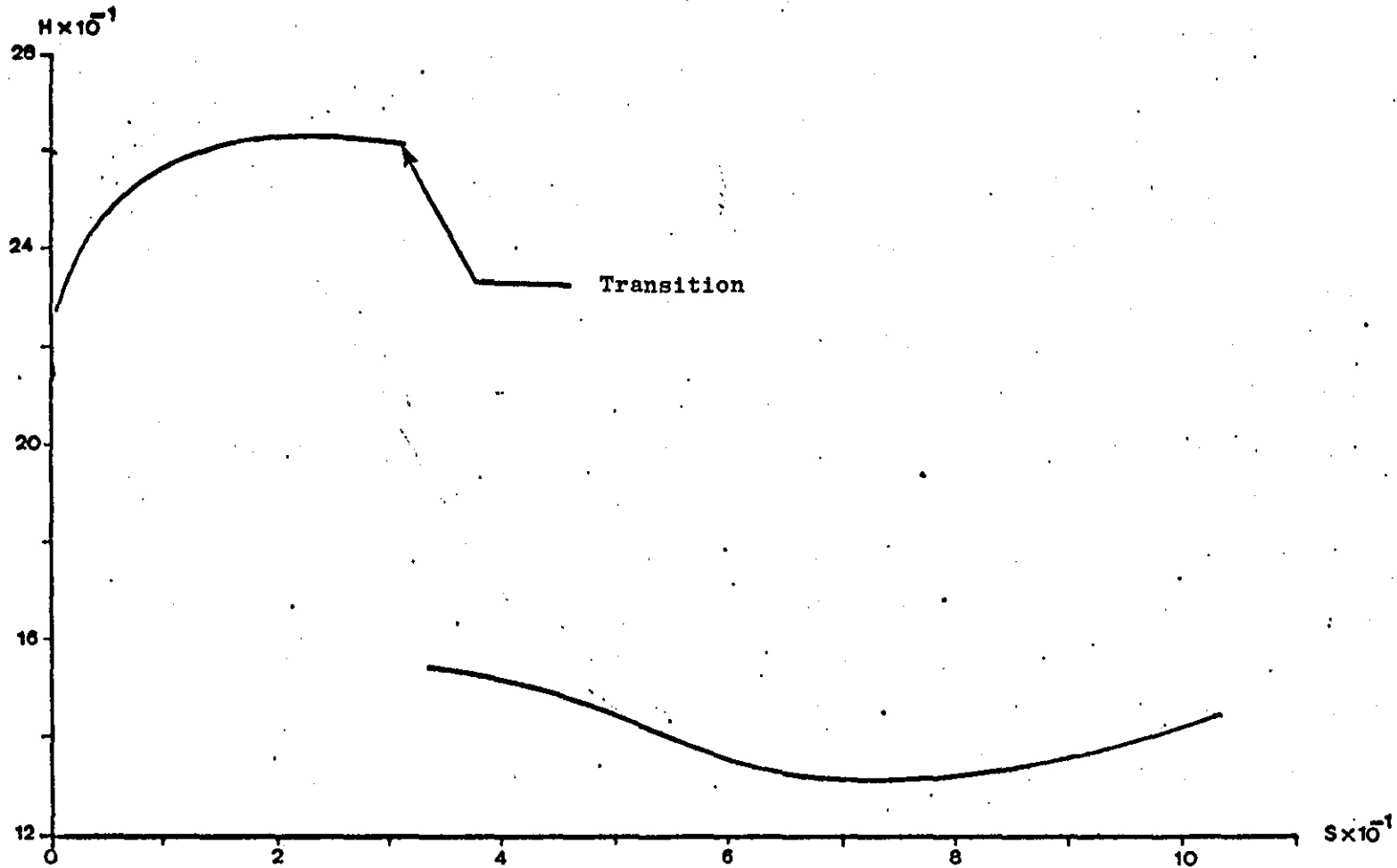
(b) Plot of momentum thickness (Lower surface)

FIGURE 13.9 Continued



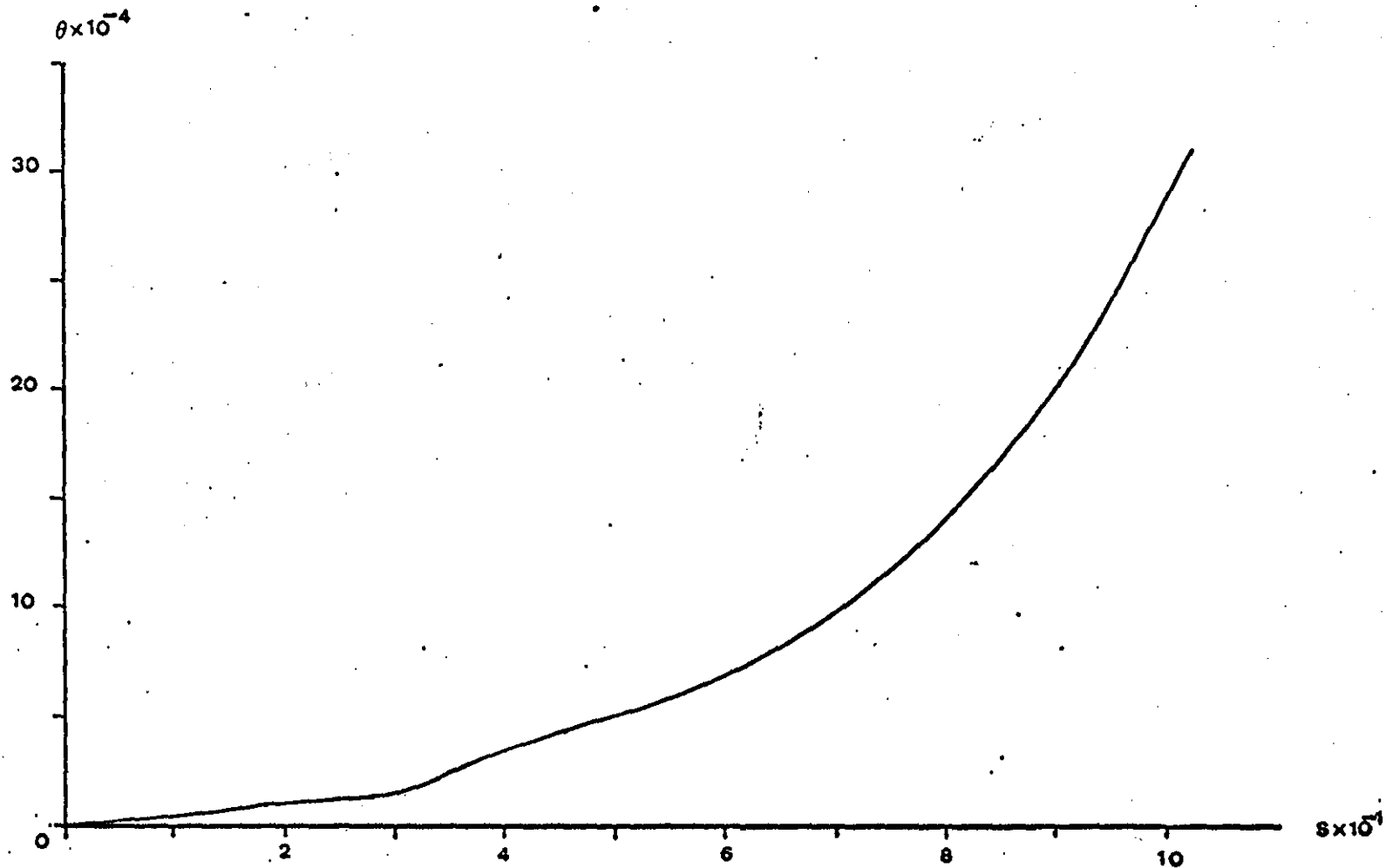
(c) Plot of displacement thickness (Lower surface)

FIGURE 13.9 Continued



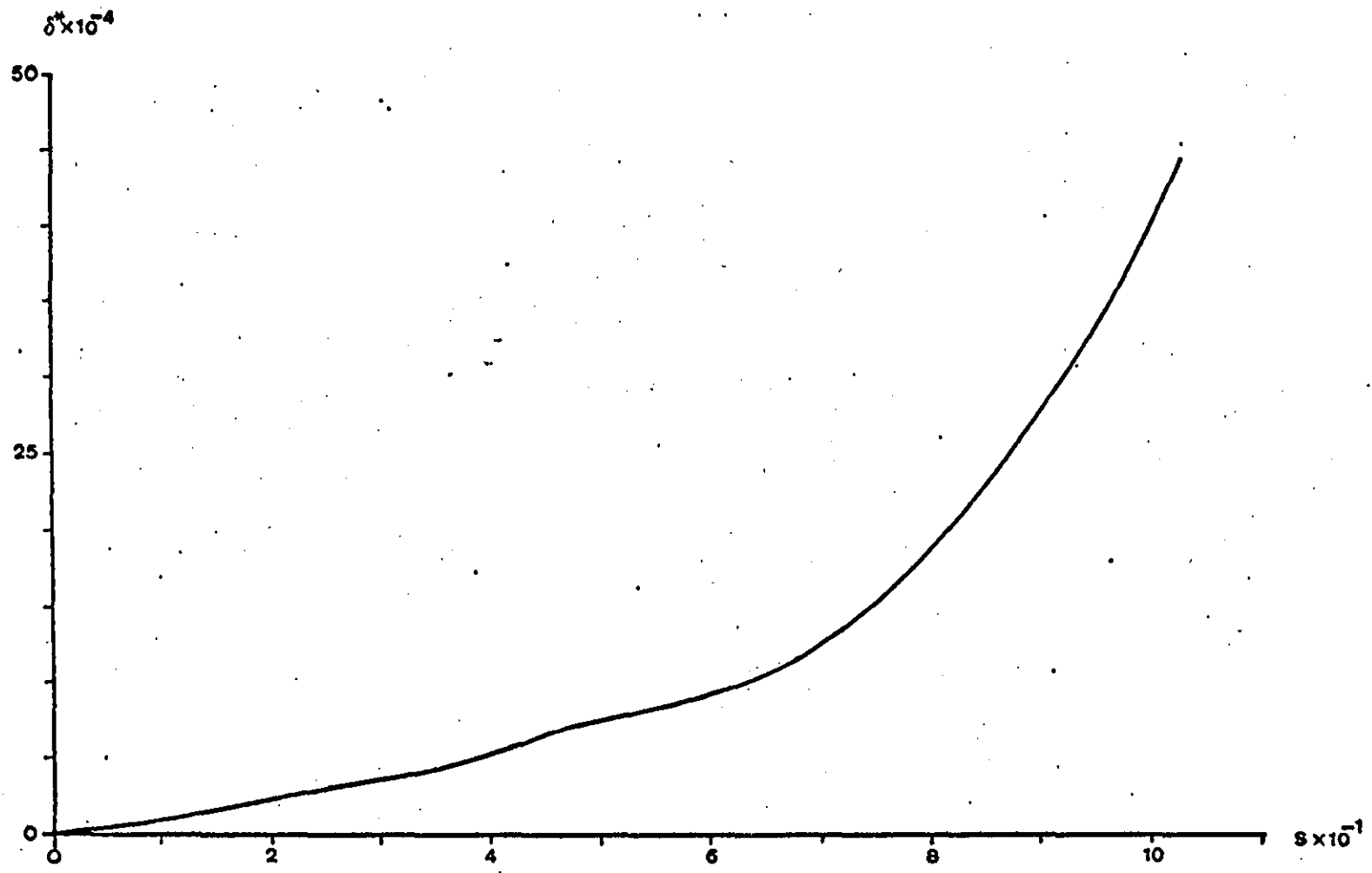
(d) Plot of shape parameter (Upper surface)

FIGURE 13.9 Continued



(e) Plot of momentum thickness (Upper surface)

FIGURE 13.9 Continued



(f) Plot of displacement thickness (Upper surface)

FIGURE 13.9 Continued

CONCLUSIONS - FURTHER WORK

14.1 CONCLUSIONS

A mathematical model has been developed for the calculation of the attached viscous incompressible flow about single and two element aerofoil sections in free air.

A potential flow method, of recent development, based on symmetrical vortex and source singularities distributed on flat panels is used for the calculation of the ideal flow. The method can be applied to thick aerofoils of arbitrary geometry. It also allows aerofoils with both sharp and blunt trailing edges to be analysed.

Since singularity distributions on opposite panels are equal, flow leakages are minimized. Also the use of both vortices and sources results in smoother singularity distributions. Results obtained with the present method are shown to be in very good agreement with exact solutions.

The potential flow model has been combined with an iterative boundary layer and wake solution in order to correct the ideal aerodynamic characteristics of the aerofoil for real flow effects. The viscous analysis is carried out using integral boundary layer methods. Viscous effects are accounted for by using a modified boundary condition for the normal velocity at the aerofoil surface. In this way the aerofoil influence coefficient need only be calculated once and the matrix inversion is also carried out once only. The wake analysis consists of a wake relaxation and a wake viscous analysis which is used to calculate the new wake singularities for the next iteration. No overall convergence criterion has been found so far. Thus the results have to be checked at the end of the analysis.

Results obtained using the flow model are found to be in very good agreement with experimental data even in cases where flap deflections are large (e.g. 40 degrees). Only a small number of panels is required to carry out the analysis (a total of 76 panels on a two element aerofoil is found to be sufficient).

The computer program written to test the model requires very little CPU time and this makes it a very economical tool for aerofoil analysis.

Finally it can be said that the aim of the present research has been accomplished and that the method developed is a powerful and yet economical tool which can be used for aerofoil design work.

14.2 FURTHER WORK

It is clear from the results obtained that, depending on aerofoil geometry and flow conditions, there exists an angle of attack above which separation effects dominate the flow. Thus models to describe such flows have to be included in the present model in order to allow near-stall and post-stall calculations to be carried out. A model for short bubble separation is included in the present program. Trailing edge separation has also been considered in an earlier section. However, the model was ^{not} tested with the rest of the flow model. Use should be made of the linear vorticity model in this case, since it is more adaptable to the separation model.

The confluent boundary layer model, for which a number of computer programs are given in Appendix 12, needs to be tested and combined with the rest of the flow model. The experimental results of Pot (87) and Preston (89) may be used to test the confluent boundary layer model. This will allow the flow model to analyse multi-element aerofoils at 'off-design' conditions i.e. in cases where the gap between aerofoil and its flap is very small.

From the pressure diagrams obtained it can be seen that in certain cases high velocities are present near the flap leading edge. In such cases

compressibility becomes important and the results have to be corrected. The equations which can be used to correct the velocities and boundary layer parameters for compressible flow are given in Refs.(8) and (109 p.26).

The viscous flow methods used by the present model could be tested more extensively against available experimental results (see Refs. 5,63,90,109) and if necessary they should be replaced with more up-to-date methods. The computer programs allow this to be carried out with ease.

Finally the computer program itself needs further work to bring it up to production standard. This work involves:

- a) the inclusion of geometry routines to allow interpolation from the input aerofoil coordinates, so that the distribution of panels is done within the program. A distribution of panels based on aerofoil curvature (see Appendix 9) should be tried and compared with the cosine method of distributing panels.
- b) an improvement of the interactive nature of the program to enable the desired variables to be obtained.
- c) a reduction of the storage space required. This may be done by a more efficient redistribution of the common blocks and by dimensioning all subroutine matrices dynamically.
- d) generalisation of the model to deal with any number of aerofoil elements.

REFERENCES

- [1] ABBOT I.H., VON DOENHOFF A.E., *"Theory of wing sections"*, Dover Publications, 1959.
- [2] ARGYRIS J.H., MARECZEK G., SCHARPF D.W., *"Two and three dimensional flow using finite elements"*, RAeS Journal, pp.961-964, Nov.1969.
- [3] ARGYRIS J.H., SCHARPF D.W., *"Two and three dimensional potential flow by the method of singularities"*, RAeS Journal, pp.959-961, Nov.1969.
- [4] BHATELEY I.C., BRADLEY R.G., *"A simplified mathematical model for the analysis of multi-element airfoils near stall"*, AGARD CP-102, Paper 12, 1972.
- [5] BREBNER G.G., BAGLEY J.A., *"Pressure and boundary-layer measurements on a two-dimensional wing at low speeds"*, ARC, R&M, No.2886, Feb.1952.
- [6] BRILEY W.R., McDONALD H., *"Numerical prediction of incompressible separation bubbles"*, Journal of Fluid Mechanics, Vol.69, pp.631-656, June 1975.
- [7] BRISTOW R., *"Improvements in surface singularity analysis and design methods"*, NASA CP-2045, pp.221-236, March 1978.
- [8] BRUNE G.W., *"An improved version of the NASA-LOCHEED multielement airfoil analysis computer program"*, NASA CR-145323, March 1978.
- [9] BRUNE G.W., MANKE J.W., *"Upgraded viscous flow analysis of multi-element airfoils"*, NASA CP 2045, pp.163-182, March 1978.

- [10] CALLAGHAN J.G., BEATY T.D., "A theoretical method for the analysis and design of multielement airfoils", *Journal of Aircraft*, Vol.9, pp.844-848, December 1972.
- [11] CARLSON L.A., "TRANDES: A Fortran program for transonic airfoil analysis or design", NASA CR-2821, June 1977.
- [12] CEBECI T., MOSINSKIS G.J., SMITH A.M.O., "Calculation of Separation Points in Incompressible turbulent flows", *Journal of Aircraft*, Vol.9, pp.618-624, 1972.
- [13] CEBECI T., MOSINSKIS G.J., SMITH A.M.O., "Calculation of viscous drag in incompressible flows", Vol.9, pp.691-692, October 1972.
- [14] CHAPMAN D.R., "Computational Aerodynamics Development and Outlook", *AIAA Journal*, Vol.17, No.12, pp.1293-1313, December, 1979.
- [15] CHAPPEL P.O., "Flow separation and stall characteristics of plane, constant section wings in subcritical flow", *R.Ae.S. Journal* Vol.72, pp.82-90, 1968.
- [16] CHOW W.L., SPRING D.J., "Viscid-Inviscid interaction of incompressible separated flows", *Journal of Applied Mechanics*, Vol.43, pp.387-395, Sept. 1976.
- [17] CHUEN Y.C., "An introduction to computational fluid mechanics", John Wiley and Sons, 1979.
- [18] CHUNG T.J., "Finite element analysis in fluid dynamics", McGraw-Hill, 1978.
- [19] COLES D., "The law of the wake in the turbulent boundary layer", *Journal of Fluid Mechanics*, Vol.1, 1956.

- [20] COLLYER M.R., LOCK R.C., *"Prediction of viscous effects in steady transonic flow past an aerofoil"*, *Aeronautical Quarterly*, pp.485-505, August 1979.
- [21] CONTE S.D., BOOR de C., *"Elementary numerical analysis"*, McGraw-Hill, Second Edition, 1972.
- [22] CRABTREE L.F., *"Prediction of transition in the boundary layer on an aerofoil"*, *R.Ae.S. Journal*, Vol.62, pp.525-528, July, 1958.
- [23] CRABTREE L.F., *"Effects of leading-edge separation on thin wings in two dimensional incompressible flow"*, *Journal of the Aeronautical Sciences*, Vol.24, pp.597-604, August 1957.
- [24] CRIMI P., REEVES B.L., *"Analysis of leading edge separation bubbles on airfoils"*, *AIAA Journal*, Vol.14, pp.1548-1555, 1976.
- [25] CROSS A.G.T., *"Calculation of viscous incompressible flow about a single component aerofoil in a closed wind tunnel"*, Note YAD 3355, October 1978, British Aerospace Aircraft Group, Kingston-Brough Division.
- [26] CROSS A.G.T., *Private Communication*.
- [27] CURLE N., *"An accurate calculation method for two dimensional incompressible laminar boundary layer including cases with regions of sharp pressure gradient"*, *Aeronautical Quarterly*, Vol. 28-29, pp.149-162, August 1977.
- [28] CURLE N., *"A two-parameter method for calculating the two-dimensional incompressible laminar boundary layer"*, *R.Ae.S. Journal*, Vol.71, pp.117-123, February 1967.

- [29] DUNCAN W.J., THOM A.S., YOUNG A.P., *"The mechanics of fluids"*, Arnold, 1960.
- [30] DUTT H.N.V., SREEKANTH A.K., *"Design of aerofoils for prescribed pressure distribution in viscous incompressible flows"*, The Aeronautical Quarterly, pp.42-55, February, 1980.
- [31] DVORAK F.A., *"Calculation of compressible turbulent boundary layers with roughness and heat transfer"*, AIAA Journal, Vol.10, pp.1447-1451, November 1972.
- [32] DVORAK F.A., *"Calculation of turbulent boundary layers on rough surfaces in pressure gradient"*, AIAA Journal, Vol.7, pp.1752-1759, September 1969.
- [33] EHLERS J., WEIDENMULLER H.A., *"Proceedings of the third international conference on numerical methods in fluid mechanics"*, Springer-Verlag, Berlin, Heidelberg, New York, 1973.
- [34] EPPLER R., SOMERS D.M., *"Low speed aerofoil design and analysis"*, NASA CP-2045 pp.73-100, March 1978.
- [35] FALKNER V.M., *"The calculation of aerodynamic loading on surfaces of any shape"*, A.R.C. Technical Report, R&M 1910, August 1943.
- [36] FARN C.L.S., GOLDSCHMIED F.R., WHIRLOW D.K., *"Pressure distribution prediction for two-dimensional hydrofoils with massive turbulent separation"*, Journal of Hydronautics, Vol.10, No.3, pp.95-101, July 1976.
- [37] FOSTER D.N., *"The low-speed stalling of wings with high lift devices"*, AGARD CP-102, paper 11, April 1972.

- [38] FOSTER D.N., *"Flow around wing sections with high lift devices"*,
Journal of Aircraft, Vol.9, pp.205-210, March 1972.
- [39] FOSTER D.N., IRWIN H.P.A., *"The two-dimensional flow around a slotted flap"*, A.R.C. R M, No.3681, 1971.
- [40] FREULER R.J., GREGOREK G.M., *"An evaluation of four single element airfoil analytic methods"*, NASA CP-2045, pp.133-162, March 1978.
- [41] GASTER M., *"The structure and behaviour of laminar separation bubbles"*,
A.R.C. R&M, No.3595, 1969.
- [42] GAULT D.E., *"An experimental investigation of regions of separated laminar flow"*, NACA TN 3505, Sept. 1955.
- [43] GORADIA S.H., LYMAN V., *"Laminar stall prediction and estimation of $C_{l,max}$ "*, Journal of Aircraft, Vol.11, pp.528-536, Sept. 1974.
- [44] GREEN J.E., *"The prediction of turbulent boundary layer development in compressible flow"*, Journal of Fluid Mechanics, Vol.31, pp.753-778, 1968.
- [45] GREEN J.E., *"Two-dimensional turbulent reattachment as a boundary layer problem"*, AGARD CP-4, pp.393-427, May 1966.
- [46] GROSS L.W., *"The prediction of two-dimensional airfoil stall progression"*, AIAA paper 78-155 16th Aerospace Sciences Meeting, January 1978.
- [47] HAYOS F.G., *"Turbulent boundary-layer effects on base pressure"*,
Journal of the Aeronautical Sciences, pp.781-782, Vol.24,
October 1957.

- [48] HEAD M.R., *"Entrainment in the turbulent boundary layer"*,
ARC R&M, No. 3152, Sept. 1958.
- [49] HEAD M.R., PATEL, V.C., *"Improved entrainment method for calculating turbulent boundary layer development"*, ARC R&M, No.3643,
March 1968.
- [50] HERRING R.N., ELY W.L., *"Improved prediction of laminar leading edge separation"*, NASA CP-2045, pp.327-334, March 1978.
- [51] HOAD D.R., *"Application of the laser velocimeter to aerofoil research"*, NASA CP-2045, pp.559-570, March 1978.
- [52] HODGE J.K., STONE A.L., MILLER T.E., *"Numerical solution for airfoils near stall in optimized boundary-fitted curvilinear coordinates"*, AIAA Journal, Vol.17, pp.458-464, May 1979.
- [53] HORTON H.P., *"A semi-empirical theory for the growth and bursting of liminar separation bubbles"*, ARC Current Papers, 1073, 1969.
- [54] HOUGHTON E.L., BOSWELL R.P., *"Further aerodynamics for engineering students"*, Arnold 1969.
- [55] INGEN J.L., *"Transition, pressure gradient, suction, separation and stability theory"*, AGARD CP-224, Paper 20.
- [56] IRWIN H.P.A.H., *"A calculation for the two-dimensional turbulent flow over a slotted flap"*, ARC CP-1267, 1974.
- [57] JACOB K., STEINBACH D., *"A method for prediction of lift for multielement airfoil systems with separation"*, AGARD CP-143,
Paper 12, April 1974.

- [58] JAMES C.W., "*Incompressible boundary-layer separation*", Annual review of Fluid Mechanics, Vol.9, pp.113-144, 1977.
- [59] JAMES R.M., "*A general class of exact airfoil solutions*", Journal of Aircraft, Vol.9, pp.574-580, August 1972.
- [60] KAO H.C., "*Some aspects of airfoil stall in low-speed flow*", Journal of Aircraft, Vol.11, pp.177-180, March 1974.
- [61] KENNEDY J.L., MARSDEN D.J., "*A potential flow design method for multicomponent airfoil sections*", Journal of Aircraft, No.1, Vol. 15, pp.47-52, January 1978.
- [62] KLINE S.J., "*Some new conceptions of the mechanism of stall in turbulent boundary layers*", Journal of the Aeronautical Sciences, Vol.24, pp.470-471, June 1957.
- [63] KLINE, MORKOVIN, SORVAN, COCKREL, "*Computation of turbulent boundary layers*", AFOSR - IFP-STANFORD CONFERENCE 1968.
- [64] KORST H.H., "*A theory for base pressure in transonic and supersonic flow*", Journal of Applied Mechanics, Vol.23, pp.593-600, 1956.
- [65] KUCHEMANN D., "*The aerodynamic design of aircraft*", Pergamon 1978.
- [66] LAMB H., "*Hydrodynamics*", Cambridge University Press, 6th Edition, 1932.
- [67] LIGHTHILL M.J., "*On displacement thickness*", Journal of Fluid Mechanics, Vol.4, pp.383-392, 1958.
- [68] LOCK R.C., "*A review of methods for predicting viscous effects on aerofoils and wings at transonic speeds*", Tech.Memo. Aero 1860, July 1980.

- [69] MABEY D.G., *"Analysis and correlation of data on pressure fluctuations in separated flow"*, Journal of Aircraft, Vol.9, pp.642-645, September 1972.
- [70] MASKEW B., *"Numerical lifting surface methods for calculating the potential flow about wings and wing bodies of arbitrary geometry"*, Loughborough University Doctoral Thesis, 1972.
- [71] MASKEW B., DVORAK A., *"Investigation of separation models for the prediction of C_{lmax} "*, NASA CP-2045, pp.347-365, March 1978.
- [72] MASKEW B., WOODWARD F., *"Symmetrical singularity model for lifting potential flow analysis"*, Journal of Aircraft, Vol.9, pp.733-734, September 1976.
- [73] MAVRIPLIS F., *"Aerodynamic research on high lift systems"*, Canadian Aeronautics and Space Journal, pp.175-183, May 1971.
- [74] McCULLOUGH G.B., *"The effect of Reynolds number on the stalling characteristics and pressure distributions of four moderately thin airfoil sections"*, NACA TN 3524, November 1955.
- [75] McDONALD H., STODDART J.A.P., *"On the development of the incompressible turbulent boundary layer"*, A.R.C. R&M, No. 3484, 1967.
- [76] MCGHEE R.J., *"Low speed aerodynamic characteristics of a 17 percent thick airfoil section design for general aviation applications"*, NASA TN-D17428, December 1978.
- [77] MICHEL R., *"Determination of Transition point and calculation of drag on wing sections in incompressible flow"*, O.N.E.R.A., Publication No.58, 1952.

- [78] MILNE L.M., THOMSON C.B., *"Theoretical Hydrodynamics"* Fifth edition 1968, Macmillan and Co., Ltd.
- [79] MORGAN H.L., *"A computer program for the analysis of multielement airfoils in two dimensional subsonic-viscous flow"*, NASA SP-347, pp.713-747, March 1975.
- [80] MORINO L., KUO C.C., *"Subsonic potential aerodynamics for complex configurations: A general theory"*, AIAA Journal, Vol.12, pp. 191-197, 1974.
- [81] MUNK M.M., *"Elements of the wing section theory and of the wing theory"*, NASA RP 1050, TR 191, December 1979.
- [82] NASH J.F., HICKS J.G., *"An integral method including the effect of upstream history on the turbulent shear stress"*, AFOSR - IFP - Stanford Conference, August 1968.
- [83] NASH J.F., McDONALD H., *"A calculation method for the incompressible turbulent boundary layer including the effect of upstream history on the turbulent shear stress"*, NPL Aero Report 1234, 1967.
- [84] PERRY A.E., FOURLIE B.D., *"A study of the turbulent boundary layer separation and reattachment"*, Journal of Fluid Mechanics, pp.657-672, June 1975.
- [85] PETRIE J.A.H., *"Development of an efficient and versatile panel method for aerodynamic problems"*, University of Leeds, Doctoral Thesis, March 1979.
- [86] PETRIE J.A.H., *"Chordwise variation of vorticity: A comparison of approximations for use in panel methods"*, Aeronautical Quarterly, pp.265-270, November 1977.

- [87] POT P.J., *"Measurements in a 2D wake and in a 2D wake merging into a boundary layer"*, NLR TR 79063, June 1979.
- [88] PRANDTL, *"Applications of modern hydrodynamics to aeronautics"*, NASA RP 1050, TR 116, Dec.1979.
- [89] PRESTON J.H., *"The effect of the boundary layer and wake on the flow past a symmetrical aerofoil at zero incidence"*, A.R.C. R&M, No.2107, July 1945.
- [90] PRESTON J.H., SWEETING N.E., COX D.K., *"The experimental determination of the boundary layer and wake characteristics of a Piercy 12/40 aerofoil with particular reference to the trailing edge region"*, A.R.C. R&M, No.2013, February 1945.
- [91] PURCELL E.W., *"The vector method for solving simultaneous linear equations"*, Journal of Math.Physics. Vol.23.180, 1953.
- [92] RESHOTKO E., *"Boundary layer stability and transition"*, Annual Review of Fluid Mechanics, Vol.8, pp.311-349, 1976.
- [93] RESHOTKO E., *"Stability theory as a guide to the evaluation of transition data"*, AIAA Journal, Vol.7, pp.1086-1091, June 1969.
- [94] REYNOLDS W.C., *"Computation of turbulent flows"*, Annual Review of Fluid Mechanics", pp.183-208, Vol.8, 1976.
- [95] RIEGELS, *"Aerofoil sections"*, Butterworths Scientific Publications, London 1961.
- [96] ROBERTS W.B., *"Calculation of laminar separation bubbles and their effect on aerofoil performance"*, AIAA Journal, Vol.18, pp.25-31, January 1980.

- [97] ROBERTSON J.M., *"Prediction of turbulent boundary layer separation"*,
Journal of the Aeronautical Sciences, Vol.24, pp.631-632,
August, 1957.
- [98] RODGERS E.J., WENTZ W.H., SEETHARAM H.C., *"Instrumentation
techniques and data reduction associated with airfoil testing
programs at Wichita State University"*, NASA CP-2045, pp.539-558,
March 1978.
- [99] SCEID F., *"Theory and problems of numerical analysis"*, Schaum's
Outline Series - McGraw-Hill, 1968.
- [100] SCHLICHTING, *"Boundary layer theory"*, McGraw-Hill 1979.
- [101] SCHLICHTING, TRUCKENBRODT, *"Aerodynamics of the Airplane"*,
McGraw-Hill, 1979.
- [102] SCHUBAUER G.B., KLEBANOFF P.S., *"Investigation of separation of the
turbulent boundary layer"*, NACA 37th Annual Report, Report 1030,
pp.689-708, 1951.
- [103] SEEBOHM T., NEWMAN B.G., *"A numerical method for calculating
viscous flow round multiple-section aerofoils"*, Aeronautical
Quarterly, pp.176-188, August 1975.
- [104] SILVERSTEIN A., BECKER V.J., *"Determination of boundary layer
transition on three symmetrical airfoils in the NACA full
scale wind tunnel"*, NACA 24th Annual Report, Report 637, 1938.
- [105] SMETANA F.O., SYMMEY D.C., SMITH N.S., CARDEN R.K., *"Light
aircraft lift drag and moment prediction - A review and
analysis"*, NASA CR-2523, May 1975.

- [106] SMITH A.M.O., *"Aerodynamics of high lift airfoil systems"*,
AGARD CP-102, paper 10, 1972.
- [107] SOWERBY L., *"Vector field theory with applications"*, Logman 1974.
- [108] SQUIRE H.B., YOUNG A.D., *"The calculation of the profile drag of
aerofoils"*, ARC R&M, No.1838, November 1937, Also RRC
Technical Report, Vol.1, pp.196-221, 1938.
- [109] STEVENS W.A., GORADIA S.H., BRADEN J.A., *"Mathematical model for
two dimensional mult-component airfoils in viscous flow"*,
NASA CR-1843, July, 1971.
- [110] STRATFORD B.S., *"Flow in the laminar boundary layer near separation"*,
ARC, R&M No.3002, November 1954.
- [111] STRATFORD B.S., *"The prediction of separation of the turbulent
boundary layer"*, Journal of Fluid Mechanics, Vol.5, pp.1-16,
1959.
- [112] SYTSMA H.S., *"A comparison of panel methods for subsonic flow
computation"*, AGARD - AG-241, February 1979.
- [113] TANI I., *"Boundary layer transition"*, Annual Review of Fluid
Mechanics, Vol.1, pp.169-198, 1969.
- [114] TANI, I., *"History of boundary layer"*, Annual Review of Fluid
Mechanics, Vol.9, pp.87-112, 1977.
- [115] THEODORSEN T., GARRICK I.G., *"General potential theory of arbitrary
wing sections"*, NASA RP 1050, TR 452 December 1979.
- [116] THOMPSON B.G.J., *"A critical review of existing methods of
calculating the turbulent boundary layer"*, A.R.C. R&M No.3447, 1967.

- [117] THWAITES B., "Approximate calculation of the laminar boundary layer", *The Aeronautical Quarterly*, Vol.1, pp.245-280, Nov. 1949.
- [118] TULINIUS J.R., MARGASON R.J., "Aircraft aerodynamic design and evaluation methods", AIAA paper No. 76-15, January 1976.
- [119] VINCENT de PAUL, M., "Prevision du décrochage d'un profil d'aile en encoulement incompressible", AGARD CP-102, Paper 5, 1972.
- [120] WILLIAMS B.R., "An exact test case for the plane potential flow about two adjacent lifting aerofoils", A.R.C. R&M No. 3717, 1973.
- [121] ZEMANSKY M.W., "Heat and thermodynamics", Fifth edition, McGraw-Hill, 1968.
- [122] ZUMVALT G.W., SHARAD N.N., "A new flow model for highly separated airfoil flows at low speeds", NASA CP-2045, pp.367-382, March 1978.
- [123] BIRNHAUM, "Die tragende wirbelfuche als Hilfsmittel zur Behandiung des ebenen Problems der Tranagliel Theorie", ZAMM, 1923.
- [124] GLAUERT, H., "Theory of Thin Aerofoils", ARC R&M, 910, 1224.
- [125] HELMHOLTZ, H., "Über discontinuirliche flussingebewegungen", *Monatesberichte der Koniglichen Academic der Wissenschaften zu Berlin*, 1868, pp.215-228.
- [126] HESS, J.L., SMITH, A.M.O., "Calculation of potential flow about arbitrary bodies", *Progress in Aeronautical Sciences*, Vol.8, pp.1-138, Pergamon Press, 1967.
- [127] KARAMCHETI, K., "Principles of ideal fluid aerodynamics", John Wiley and Sons, New York, 1966.

- [128] KARMAN, T., TREFFTZ, G., "*Potentialstromung um gegebene Tragflächenquerschnitte*", ZFM, 1918.
- [129] KIRCHHOFF, G., "*Zur Theorie freier flüssigkeitsstrahlen*", Journal für die reine und angewandte Mathematik, 1869, pp.289-298.
- [130] LIGHTHILL, M.J., "*A new method to two-dimensional aerodynamic design*", A.R.C., R&M, 2112, 1945.
- [131] MARTENSEN, E., "*Berechnung der Druckverteilung an Gitterprofilen in ebener potentialstromung mit einer fredholmscher integralgleichung*", Arch. Rational Mech.Anal., Vol.3, No.3, 1959, pp.235-276.
- [132] McCORMICK, B.W., "*Aerodynamics, aeronautics, and flight mechanics*", John Wiley & Sons, 1979.
- [133] MISES, R., "*Zur Theorie des Tragflächenauftriebes*", Zeitschrift für Flugtechnik und Motorluftschiffahrt (ZFM), Vol. 11, p.68, 1920.
- [134] MULLER, W., "*Zur konstruktion von Tragflüchenprofilen*", ZAAM, 1924.
- [135] WENTZ, W.H., SEETHARAM, C.H., "*Development of a fowler flap system for a high performance general aviation airfoil*", NASA CR-2443, Dec. 1974.
- [136] DVORAK, F.A., "*The simulation of turbulent boundary layer separation on multi-element infinite swept wings*", NASA CR-137694, March 1975.
- [137] GELLER, W., "*Incompressible flow through cascades with separation*", AGARDograph No. 164, 1972.

- [138] BINGHAM, G.J., CHEN, A.W., *"Low speed aerodynamic characteristics of an airfoil optimized for maximum lift coefficient"*, NASA TN D-7071, Dec. 1972.
- [139] WEBER, J., *"The calculation of the pressure distribution on the surface of thick cambered wings and the design of wings with given pressure distribution"*, A.R.C. R & M 3026, June 1955.

APPENDIX 1

THE COMPUTER PROGRAM

A listing of the computer program developed to test the flow model is given in the following pages. The program is written in standard FORTRAN for the PRIME 400 computer of Loughborough University of Technology. The program could be run in batch mode or interactively.

Figure A1.1 shows the main functions of the computer program. A flow chart of the program, in terms of its subroutines, is given in Figure A1.2.

Considerable effort has been put into optimizing the program so that CPU time is kept low (five iterations with a total of 60 panels require 80 secs. of CPU on the PRIME 400). Required storage is approximately 150K.

More program information is given in Appendices 2 and 3.

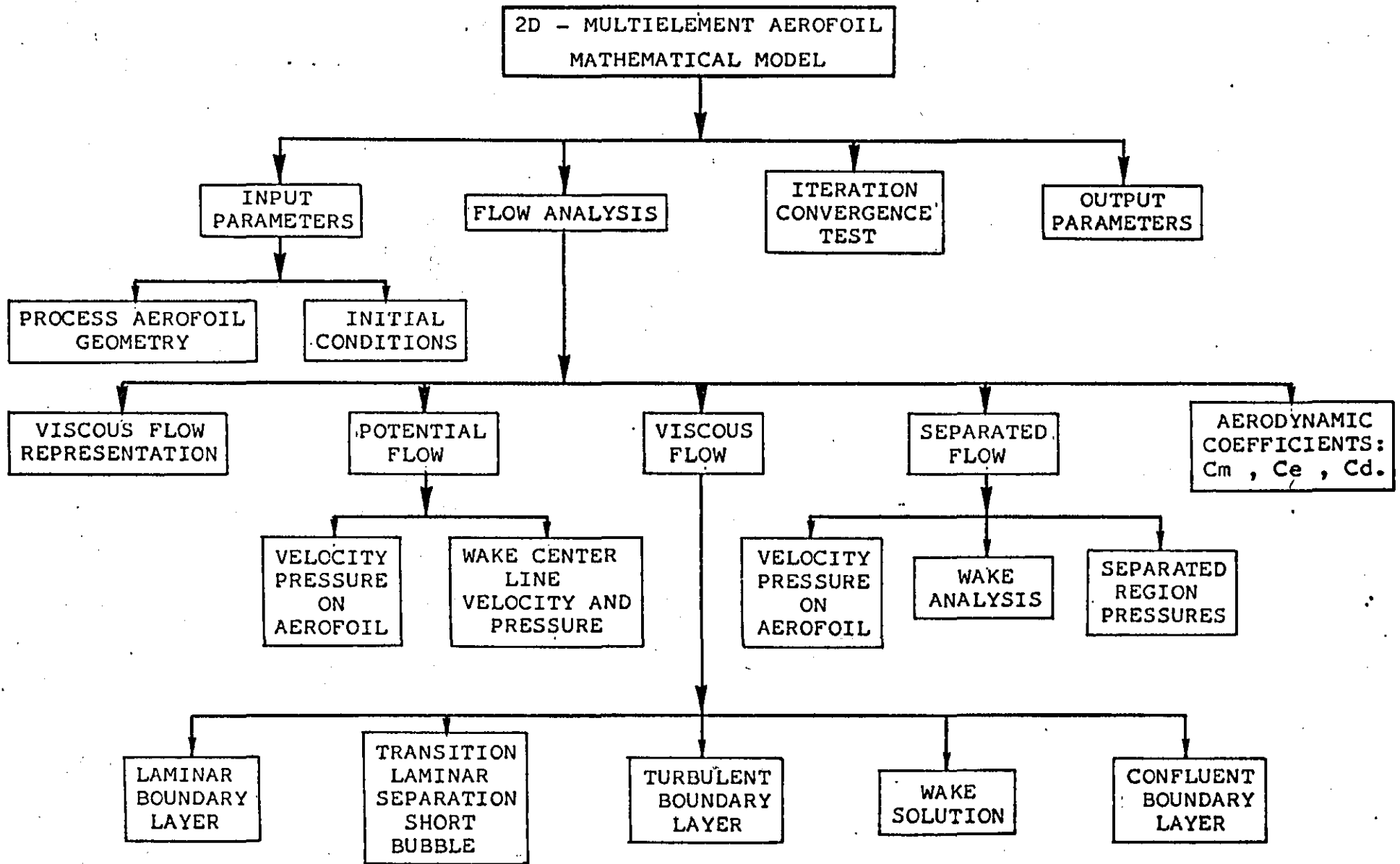


FIGURE A1.1 Flow Model Decomposition

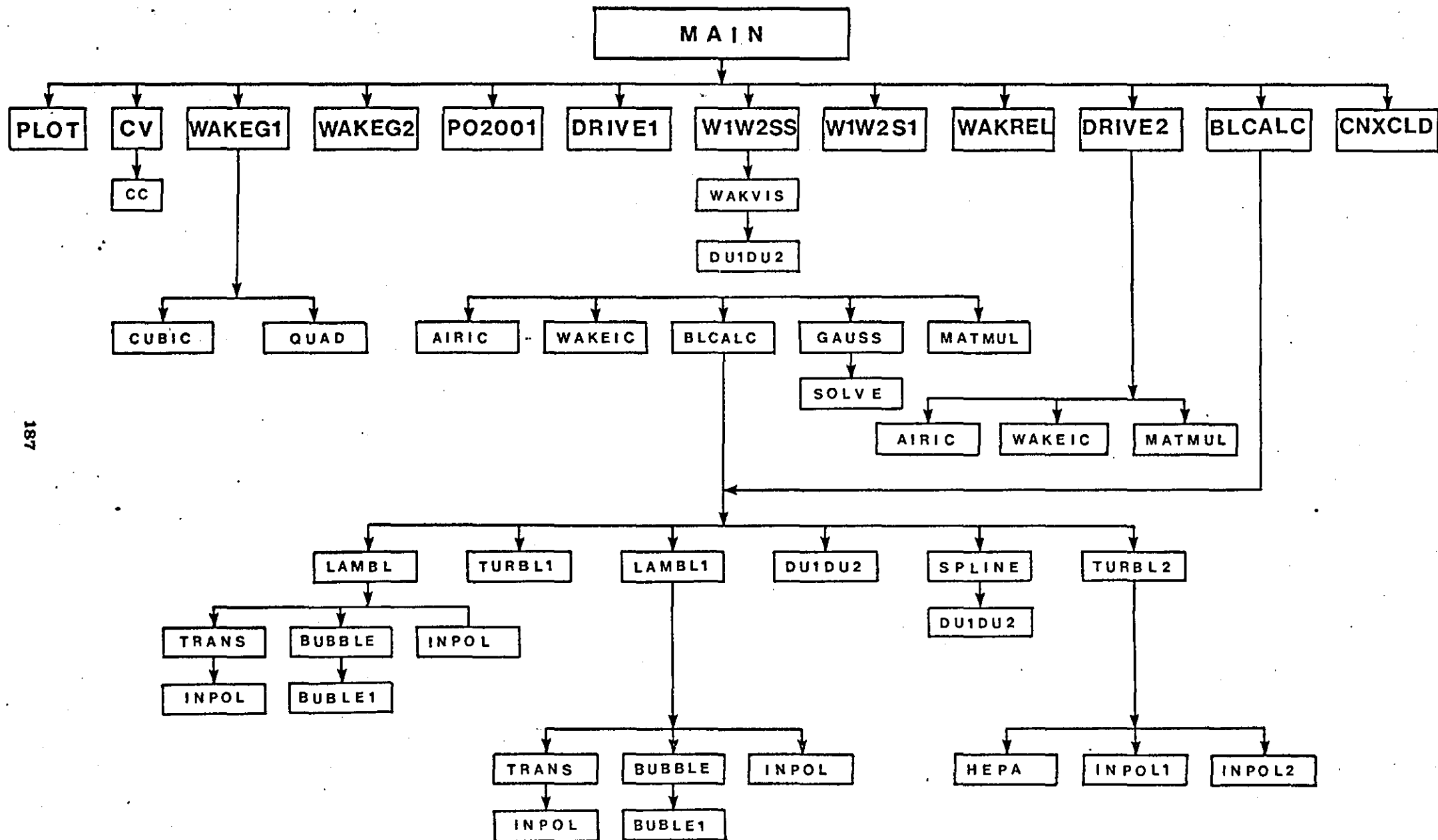


FIGURE A1-2 Subroutine decomposition of computer programme

```

(0001) C
(0002) C
(0003) C ***** M A I N P R O G R A M *****
(0004) C
(0005) C
(0006) INTEGER*2 TYPE, CODE, IP1(6)
(0007) COMMON/BLOCK1/ALPHA, CL, CBAR, XM, ZM, CHORD
(0008) COMMON/BLOCK2/CN(10), CM(10), COFP(10), CNT, CMT, COFPT
(0009) COMMON/BLOCK3/RE1, ISTAG(10)
(0010) COMMON/BL4/ALPHA2, KK, NS1, NS2, PI
(0011) COMMON/BL5/X(99), Z(99), S(99), SA(99), CN1(99), SN1(99);
(0012) +T1(99), T3(99)
(0013) COMMON/BL6/XW1(99), ZW1(99), XW2(99), ZW2(99), SW1(99), SW2(99)
(0014) COMMON/BL7/XBW1(99), ZBW1(99), XBW2(99), ZBW2(99)
(0015) COMMON/BL8/VN1(99), VN2(99), VT1(99), VT2(99)
(0016) COMMON/BL9/VUW1(99), VLW1(99), VUW2(99), VLW2(99)
(0017) COMMON/BL10/GAMV1(99), GAMV2(99), GAMS1(99), GAMS2(99)
(0018) COMMON/BL11/GAMV11(99), GAMV22(99), GAMS11(99), GAMS22(99)
(0019) COMMON/BL12/SNW1(99), SNW2(99), CNW1(99), CNW2(99)
(0020) COMMON/BL13/TW11(99), TW13(99), TW21(99), TW23(99)
(0021) COMMON/BL14/GNV1(99,99), GNV2(99,99), GTV1(99,99), GTV2(99,99)
(0022) COMMON/BL15/GAMA(99), VTOT(99)
(0023) COMMON/BL16/TIN, HIN, TIN1, HIN1, TIN2, HIN2, TIN3, HIN3
(0024) COMMON/BL17/XBAR(99), ZBAR(99)
(0025) COMMON/BL18/CP(99), CF(99)
(0026) COMMON/BL19/KN(10)
(0027) COMMON/BL20/ITRACE
(0028) COMMON/BL21/PHI(10)
(0029) COMMON/BL23/IPAR
(0030) COMMON/BL24/RES4(99), MAN1, MAN
(0031) COMMON/BL25/CDU1, CDU2, CDL1, CDL2
(0032) DIMENSION VM1(99), VM2(99);
(0033) +VI(99), VBL(99), VW1(99), VW2(99), CAP(10), CNP(10), CMP(10);
(0034) +H(99), G(99), CAF(10), CNF(10), CMF(10), D(4160), IARRAY(48);
(0035) +XP(9), ZP(9), DELX(9), DELZ(9), L(10), X100(99), Y100(99), DUDS(99)
(0036) $INSERT SYSCOM)KEYS.F
(0037) $INSERT SYSCOM)A$KEYS
(0038) WRITE(1,9968)
(0039) 9968 FORMAT(/, 'PLEASE ENTER TYPE OF ANALYSIS REQUIRED ie 1 or 0'
(0040) READ(1,*)IPAR
(0041) WRITE(1,9966)
(0042) 9966 FORMAT(/, 'TRACE ON 1 *** TRACE OFF 0',/)
(0043) READ(1,*)ITRACE
(0044) WRITE(1,202)
(0045) 202 FORMAT(/, 'PLEASE ENTER DATA FILE NAME',/)
(0046) READ(1,204)IP1

```



```

07) 204  FORMAT(6A2)
08)      CODE=0
09)      CALL SRCH$(K$READ,IP1,12,1,TYPE,CODE)
10)      CALL ERRPR$(K$NRTN,0,0,0,0)
11)      WRITE(1,9964)
12) 9964  FORMAT(/,'PLEASE ENTER OUTPUT FILE NAME',//)
13)      READ(1,204)IP1
14)      CALL SRCH$(K$WRIT,IP1,12,2,TYPE,CODE)
15)      CALL ERRPR$(K$NRTN,0,0,0,0)
16)      IF(IPAR.EQ.1)GOTO 4002
17)      WRITE(1,200)
18) 200   FORMAT(/,10X,'AEROFOIL GEOMETRY OR PRESSURE DISTRIBUTION ?'//,
19)      +9X,'TYPE 9 FOR THE FIRST OR 99 FOR THE SECOND'//)
20)      READ(1,*)IP
21)      WRITE(1,4000)
22) 4000  FORMAT(/,'PLEASE ENTER TITLE OF X-AXIS OF SECTION PLOT')
23)      READ(1,4500)IARRAY
24) 4500  FORMAT(48A1)
25) 4002  CONTINUE
26) 3500  WRITE(1,510)
27) 510   FORMAT(/,'PLEASE ENTER ALPHA - CBAR - XM - ZM - MO ')
28)      READ(1,*)ALPHA,CBAR,XM,ZM,RMO
29)      WRITE(1,512)
30) 512   FORMAT(/,'PLEASE ENTER REYNOLDS NUMBER BASED ON UNIT CHORD',//)
31)      READ(1,*)RE1
32)      CHORD=CBAR
33)      WRITE(6,220)ALPHA,RE1
34) 220   FORMAT(/,'ANGLE OF ATTACK=' ,F5.2,10X,'RE=' ,E12.7,//)
35)      RMOSQ=RMO*RMO
36)      BET=SQRT(1-RMOSQ)
37)      WRITE(1,515)
38) 515   FORMAT(/,'PLEASE ENTER TOTAL # OF COMPONENTS EG.FLAPS ETC',//)
39)      READ(1,*)NT
40)      NT=NT+1
41)      DO 9 K=2,NT
42)          READ(5,*)KN(K)
43) 9      CONTINUE
44)          L(1)=0
45)          KN(1)=0
46)          DO 5 K=2,NT
47)              KN(K)=KN(K-1)+KN(K)
48)              L(K)=KN(K)
49) 5      CONTINUE
50)          KK=KN(NT)
51)          IF(IPAR.EQ.1)GOTO 11
52)          IF(IP.EQ.99)READ(5,*)(X(K),Z(K),G(K),K=1,KK)
53)          IF(IP.EQ.9)GOTO 11
54)          GOTO 13
55) 11     CONTINUE
56)          IF(IPAR.EQ.1)KK=KK+NT-1
57)          READ(5,*)(X(K),Z(K),K=1,KK)
58)          KK=KN(NT)
59) 13     REWIND 5
60)          RAD=0.01745329252
61)          IF(NT.EQ.2)GO TO 10
62)          JJ=NT-2
63)          DO 520 K=1,JJ
64)              WRITE(1,525)
65) 525   FORMAT(/,'PLEASE ENTER XP(K),ZP(K),DELX(K),DEL(Z),PHI(K)')
66) 520   READ(1,*)XP(K),ZP(K),DELX(K),DELZ(K),PHI(K)
67)          WRITE(6,2001)K,XP(K),ZP(K),DELX(K),DELZ(K),PHI(K)
68) 2001  FORMAT(/,I3,3X,'XP=' ,F8.5,3X,'ZP=' ,F8.5,3X,'DELX=' ,F8.5,

```

```

(0109)      +3X,'DELZ=' ,F8.5,3X,'PHI=' ,F5.2,'Deg')
(0110)      NK2=2
(0111)      IF(JJ.NE.1)READ(5,*)NK2
(0112)      NK2=NK2-1
(0113)      NK1=1
(0114)  14   DO 6 II=NK1,NK2
(0115)      ALPHAD=PHI(II)*RAD
(0116)      I=KN(NK1+1)+1
(0117)      J=KN(II+2)
(0118)      IF(IPAR.EQ.1)I=KN(2)+2
(0119)      IF(IPAR.EQ.1)J=KN(3)+2
(0120)      CP10=COS(ALPHAD)
(0121)      SP=SIN(ALPHAD)
(0122)      DO 6 K=I,J
(0123)      X(K)=XP(II)-X(K)
(0124)      Z(K)=ZP(II)-Z(K)
(0125)      ALPHAD=XP(II)+DELX(II)-X(K)*CP10-Z(K)*SP
(0126)      Z(K)=ZP(II)+DELZ(II)+X(K)*SP-Z(K)*CP10
(0127)      X(K)=ALPHAD
(0128)  6   CONTINUE
(0129)      MM=KK
(0130)      IF(NT.EQ.3)MM=KK+2
(0131)      WRITE(6,9960)(I,X(I),Z(I),I=1,MM)
(0132)  9960  FORMAT(10X,I3,5X,F12.7,5X,F12.7)
(0133)      IF(NT.EQ.2)GOTO 125
(0134)      IK1=KN(2)+2
(0135)      IK2=KN(3)+2
(0136)      IVAR=IK2
(0137)  123  CONTINUE
(0138)      IF(X(IVAR).LT.X(1))GOTO 124
(0139)      IVAR=IVAR-1
(0140)      IF(IVAR.GE.IK1)GOTO 123
(0141)      IFLAG=0
(0142)      GOTO 125
(0143)  124  CONTINUE
(0144)      IFLAG=1
(0145)      ISTORE=IVAR
(0146)  125  CONTINUE
(0147)      IF(NK2.EQ.JJ)GOTO 10
(0148)      NK1=NK2+1
(0149)      NK2=JJ
(0150)      GO TO 14
(0151)  10   CONTINUE
(0152)      ALPHAD=ALPHA
(0153)      ALPHA=ALPHA*RAD
(0154)      ALPHA2=ALPHA
(0155)      IF(IPAR.EQ.1)GOTO496
(0156)      IF(IP.EQ.99)GOTO 499
(0157)      CALL CV(X,Z,S,T1,T3,XBAR,ZBAR,H,D,G,KN,KK,NT,K3)
(0158)      GOTO 498
(0159)  499  CONTINUE
(0160)      DO 34 JJ=2,NT
(0161)      K3=KN(JJ-1)+1
(0162)      K4=KN(JJ)
(0163)      DO 36 K=K3,K4
(0164)      J=K+1
(0165)      IF(K.EQ.K4)J=K3
(0166)      S1=X(J)-X(K)
(0167)      S3=Z(J)-Z(K)
(0168)      S(K)=SQRT(S1*S1+S3*S3)
(0169)      T1(K)=S1/S(K)
(0170)      T3(K)=S3/S(K)

```

```

1) 36 CONTINUE
2) RK3=0.
3) K5=K4-4
4) K6=K3+3
5) DO 40 I=K6,K5
6) IF (G(I).GT.RK3) I2=I
7) IF(G(I).GT.RK3) RK3=G(I)
8) 40 CONTINUE
9) 34 CONTINUE
0) DO 38 K=1, KK
1) D(K)=SQRT(1-G(K))
2) IF(K.GT. I2) D(K)=-D(K)
3) 38 CONTINUE
4) CALL CC(X,Z,S,T1,T3,D,KN, KK, NT)
5) 496 CONTINUE
6) IF (NT.EQ.2) GOTO 102
7) CALL WAKEG1(ISTORE, IFLAG, KN, XW1, ZW1)
8) CALL WAKEG2(KN, NT, XW2, ZW2)
9) C CALL PLOT1(100, NT, X100, Y100, 0)
0) GOTO 70
1) 102 CALL WAKEG2(KN, NT, XW1, ZW1)
2) C CALL PLOT1(100, NT, X100, Y100, 0)
3) 70 CONTINUE
4) WRITE(1, 250)
5) 250 FORMAT(10X, 'ENTER No OF ITERATIONS (NITER) REQUIRED' //,
6) +20X, '2 (= NITER (= 8' //)
7) READ(1, *) NITER
8) NITER1=NITER
9) IF(NT.GT.2) NITER1=NITER-1
0) DO 64 ITER=1, NITER1
1) WRITE(6, 9990) ITER
2) WRITE(1, 9990) ITER
3) 9990 FORMAT(//, 5X, ' ITERATION No =', I2, //)
4) CALL PD2001(ITER, NT, KN)
5) IF(ITER.GT.1) GOTO 84
6) NK=1
7) DO 84 I=2, NT
8) K3=KN(I-1)+1
9) K4=KN(I)-1
0) DO 86 K=K3, K4
1) SA1=XBAR(K+1)-XBAR(K)
2) SA2=ZBAR(K+1)-ZBAR(K)
3) SA(K)=SQRT(SA1*SA1+SA2*SA2)
4) C WRITE(1, 11123) NK, SA(NK)
5) 11123 FORMAT(I3, 3X, F12.7)
6) C NK=NK+1
7) 86 CONTINUE
8) 84 CONTINUE
9) CALL DRIVE1(ITER, NT, KN)
0) IF(MAN.LE.0) GOTO 64
1) IF(MAN1.LE.0) GOTO 64
2) C CALL PLOT1(102, NT, X100, Y100, 0)
3) NS10=NS1+1
4) WRITE(6, 9996)
5) 9996 FORMAT(///, 37X, ' GAMV1', 5X, ' GAMS1', 5X, ' GAMV11', 5X, ' GAMS11', //)
6) WRITE(6, 9995) (GAMV1(I5), GAMS1(I5), GAMV11(I5), GAMS11(I5), I5=1, NS1)
7) CALL DRIVE2(ITER, NT)
8) IFLAG=0
9) IF(NT.EQ.2) IFLAG=1
0) DO 66 I=1, NS1
1) VM1(I)=SQRT(VT1(I)*VT1(I)+VN1(I)*VN1(I))
2) VUW1(I)=VM1(I)+GAMV1(I)*.5

```

```

33) VLW1(I)=VM1(I)-GAMV1(I)*.5
34) 66 CONTINUE
35) WRITE(6,9994)
36) 9994 FORMAT(///,28X,'VM1',8X,'VN1',8X,'VT1',6X,'VUW1',6X,'VLW1',/)
37) WRITE(6,9993)(VM1(I5),VN1(I5),VT1(I5),VUW1(I5),VLW1(I5),I5=1,NS
38) 9993 FORMAT(26X,F8.4,2X,F8.4,2X,F8.4,2X,F8.4,2X,F8.4)
39) CALL W1W2SS(GAMV1,GAMS1,2,NS1,0)
10) C **** GAMV1 GAMS1 are the singularity values at the panel ends of WAI
11) C K6=KN(2)/2+1
12) CALL W1W2S1(XW1,ZW1,GAMV11,GAMS11,0.,0.,NS1,0)
13) IF(NT.EQ.2)CALL WAKREL(XW1,ZW1,VT1,VN1,NS1)
14) IF(IFLAG.EQ.1)GOTO 104
15) DO 68 I=1,NS2
16) VM2(I)=SQRT(VT2(I)*VT2(I)+VN2(I)*VN2(I))
17) VUW2(I)=VM2(I)+GAMV2(I)
18) VLW2(I)=VM2(I)-GAMV2(I)
19) 68 CONTINUE
20) NS10=NS2+1
21) WRITE(6,9989)
22) 9989 FORMAT(///,37X,'GAMV2',5X,'GAMS2',5X,'GAMV22',5X,'GAMS22',/)
23) WRITE(6,9995)(GAMV2(I5),GAMS2(I5),GAMV22(I5),GAMS22(I5),I5=1,NS
24) 9995 FORMAT(35X,F7.4,3X,F7.4,3X,F8.4,2X,F8.4)
25) WRITE(6,9987)
26) 9987 FORMAT(///,28X,'VM2',8X,'VN2',8X,'VT2',6X,'VUW2',6X,'VLW2',/)
27) WRITE(6,9993)(VM2(I5),VN2(I5),VT2(I5),VUW2(I5),VLW2(I5),I5=1,NS
28) CALL W1W2SS(GAMV2,GAMS2,2,NS2,1)
29) C **** GAMV2 GAMS2 singularities at panel ends of WAKE2
30) K6=KN(2)+(KN(3)-KN(2))/2+1
31) CALL W1W2S1(XW2,ZW2,GAMV22,GAMS22,0,RES4(K6),NS2,1)
32) CALL WAKREL(XW1,ZW1,VT1,VN1,NS1)
33) CALL WAKREL(XW2,ZW2,VT2,VN2,NS2)
34) 104 CONTINUE
35) C CALL PLOT1(100,NT,X100,Y100,0)
36) 64 CONTINUE
37) IF(NT.EQ.2)GOTO 110
38) CALL DRIVE1(ITER,NT,KN)
39) CALL DRIVE2(ITER,NT)
40) DO 72 I=1,NS1
41) VM1(I)=SQRT(VT1(I)*VT1(I)+VN1(I)*VN1(I))
42) VUW1(I)=VM1(I)+GAMV1(I)*.5
43) VLW1(I)=VM1(I)-GAMV1(I)*.5
44) 72 CONTINUE
45) 106 CONTINUE
46) DO 76 I=1,NS2
47) VM2(I)=SQRT(VT2(I)*VT2(I)+VN2(I)*VN2(I))
48) VUW2(I)=VM2(I)+GAMV2(I)
49) VLW2(I)=VM2(I)-GAMV2(I)
50) 76 CONTINUE
51) 110 CONTINUE
52) CALL CNXCLD(CAP,CNP,CMP,CAF,CNF,CMF,CAT,CNT,CMT,CL,KN,NT)
53) GOTO 300
54) 498 WRITE(1,500)
55) 500 FORMAT(// 'IS HEADING AND PARAMETERS LISTING REQUIRED ?')
56) READ(1,*)M6
57) IF(M6.LT.1)GO TO 401
58) WRITE(1,22)
59) WRITE(1,26)
60) WRITE(1,27)ALPHAD,CBAR,XM,ZM,RMO,CL
61) IF(NT.EQ.2)GOTO 12
62) WRITE(1,505)
63) 505 FORMAT(1H ,)
64) WRITE(1,92)

```

```

96) WRITE(1,30)
97) JJ=JJ+1
98) IF(IPAR.EQ.0)WRITE(1,25)(CN(K),CM(K),COFP(K),K=1,JJ)
99) 12 IF(IPAR.EQ.0)WRITE(1,32)CNT,CMT,COFPT
00) WRITE(1,506)
01) 506 FORMAT(1H ,)
02) IF(IPAR.EQ.1)GOTO 300
03) 401 WRITE(1,501)
04) 501 FORMAT(1H , 'IS DETAILED LISTING REQUIRED ?')
05) READ(1,*)M7
06) IF(M7.LT.1)GO TO 400
07) WRITE(1,23)
08) WRITE(1,37)X(1),Z(1),D(1),G(1),CP10
09) DO 7 JJ=2,NT
10) I=KN(JJ-1)+2
11) J=KN(JJ)
12) DO 15 K=I,J
13) CP10=G(K)/(BET+RMOSQ*G(K)/(2*(1+BET)))
14) WRITE(1,37)X(K),Z(K),D(K),G(K),CP10
15) 15 CONTINUE
16) WRITE(1,8)
17) 7 CONTINUE
18) 8 FORMAT(1H0)
19) 22 FORMAT(1H ,10X,'TWO DIMENSIONAL INVISCID FLOW COMPUTATION')
20) 26 FORMAT(50H
21) 23 FORMAT(1H0,6X,1HX,13X,1HZ,13X,4HV/V0,10X,3HCPI,11X,3HCPC)
22) 24 FORMAT(1H ,5F14.6)
23) 25 FORMAT(1H ,3F14.6)
24) 27 FORMAT(1H , 'ALPHA=',F7.4,3X,'CBAR=',F8.3,3X,'XM=',F5.2,3X,'ZM='
25) 1 F5.2,3X,'MO=',F5.3,3X,'CL=',F8.5)
26) 37 FORMAT(1H ,5F14.6)
27) 30 FORMAT(1H0,7X,2HCN,12X,2HCM,10X,6HC.OF.P / 1X,12HPER AEROFOIL)
28) 32 FORMAT(1H ,7X,2HCN,12X,2HCM,10X,6HC.OF.P / 1X,7HOVERALL / 1X
29) 1,3F14.6)
30) 92 FORMAT(1H0,7X,2HXP,12X,2HZP,11X,4HDELX,10X,4HDELZ,11X,3HPHI)
31) 400 WRITE(1,29)
32) 29 FORMAT(///,14X,'INCOMPRESSIBLE VISCOUS CALCULATION',//)
33) IF(IPAR.EQ.0)CALL BLCALC(G,D,S,KN,KK,NT,IPAR,DUDS)
34) WRITE(1,100)
35) 100 FORMAT(1H , 'IS PLOT REQUIRED ?')
36) READ(1,*)M5
37) IF(M5.LT.1)GO TO 300
38) ALPHA1=ALPHA*180/3.1415926536
39) WRITE(1,2999)
40) 2999 FORMAT(//,10X,'ENTER DEVICE YOU WANT TO USE FOR PLOTTING'//,
41) 123X,'VDU 1' /,
42) 223X,'TREND 2' /,
43) 323X,'TEKTRONIX 3' /,
44) 423X,'C1051N 4' ,//)
45) READ(1,*)M6
46) IF(M6.EQ.1)CALL VDU
47) IF(M6.EQ.2)CALL TREND
48) IF(M6.EQ.3)CALL T4010
49) IF(M6.EQ.4)CALL C1051N
50) CALL PENSEL(1,0,0,0)
51) CALL AXIPLO(0,120.,120.,3,2,12,12,0.,1.2,-0.6,0.6,'X',0,'Z/C',3
52) CALL GRAMOV(.2,-0.65)
53) CALL CHAA1(IARRAY,48)
54) CALL PENSEL(2,0,0,0)
55) CALL PLOT(X,Z,L,NT,KK)
56) CALL PENSEL(1,0,0,0)

```

```

0358) CALL AXIPLD(0,120.,Y65,3,2,12,K47-1,0.,1.2,Y55,-Y55,'N',0,
0359) 1' GAMMA',5)
0360) CALL GRAMOV(.3,Y55-3.5)
0361) CALL CHAHOL(38HVORTICITY DISTRIBUTION X/C*.)
0362) CALL PENSEL(2,0.0,0)
0363) CALL PLOT(X,D,L,NT,KK)
0364) CALL PENSEL(1,0.0,0)
0365) CALL MAX(G,KN,Y45,Y46,K47,Y55,Y56,Y57,Y65)
0366) CALL AXIPLD(0,120.,Y65,3,2,12,K47,0.,1.2,1.,Y55,'X/C',0,'CP',
0367) CALL GRAMOV(0.15,Y56)
0368) CALL CHAHOL(35H 2D INVISCID INCOMPRESSIBLE FLOW *.)
0369) CALL GRAMOV(.4,Y57)
0370) CALL CHAHOL(16H COMPUTATION*.)
0371) CALL GRAMOV(.3,-0.3)
0372) CALL GRAMOV(.3,2.5)
0373) CALL CHAHOL(38HPRESSURE DISTRIBUTION X/C*.)
0374) CALL GRAMOV(.2,-Y46)
0375) CALL CHAHOL(19HANGLE OF ATTACK =*.)
0376) CALL GRAMOV(.75,-Y46)
0377) CALL CHAFIX(ALPH1,5,2)
0378) IF(PHI(1).EQ.0.0)GOTO 302
0379) CALL GRAMOV(.2,-Y45)
0380) CALL CHAHOL(19HFLAP DEFLECTION =*.)
0381) CALL GRAMOV(.75,-Y45)
0382) CALL CHAFIX(PHI,4,1)
0383) 302 CALL PENSEL(2,0.0,0)
0384) CALL PLOT(X,G,L,NT,KK)
0385) CALL DEVEND
0386) 300 WRITE(1,3000)
0387) 3000 FORMAT(1H,'WOULD YOU LIKE TO RUN THE PROGRAM AGAIN ?')
0388) READ(1,*)M15
0389) IF(M15.GE.1) GOTO 3500
0390) CALL EXIT
0391) END
0392) C
0393) C
0394) C ***** S U B R O U T I N E 01 *****
0395) C
0396) C
0397) SUBROUTINE CV(X,Z,S,T1,T3,XBAR,ZBAR,H,D,G,KN,KN,NT,K3)
0398) COMMON/BLOCK1/ALPHA,CL,CBAR,XM,ZM,CHORD
0399) COMMON/BLOCK2/CN(10),CM(10),COFP(10),CNT,CMT,COFPT
0400) COMMON/BL20/ITRACE
0401) DIMENSION X(KK),Z(KK),S(KK),T1(KK),T3(KK),XBAR(KK),ZBAR(KK),
0402) 1H(KK),D(K3),G(KK),KN(NT)
0403) IF(ITRACE.EQ.1)WRITE(6,9999)
0404) 9999 FORMAT(/,'SUBROUTINE CV '/')
0405) PI=3.1415926536
0406) EPS=1.E-06
0407) DO 19 JJ=2,NT
0408) K3=KN(JJ-1)+1
0409) K4=KN(JJ)
0410) DO 1 K=K3,K4
0411) J=K+1
0412) IF(K.EQ.K4)J=K3
0413) S1=X(J)-X(K)
0414) S3=Z(J)-Z(K)
0415) S(K)=SQRT(S1*S1+S3*S3)
0416) T1(K)=S1/S(K)
0417) T3(K)=S3/S(K)
0418) XBAR(K)=(X(J)+X(K))*0.5

```

```

(0420) H(K)=S1
(0421) G(K)=S3
(0422) 1 CONTINUE
(0423) 19 CONTINUE
(0424) S1=COS(ALPHA)
(0425) S3=SIN(ALPHA)
(0426) J2=KK-1
(0427) WRITE(1,125)
(0428) 125 FORMAT(/,,' IS CHECK ON INFLUENCE COEFFICIENTS REQUIRED ?')
(0429) READ(1,*)MM1
(0430) WRITE(1,334)
(0431) 334 FORMAT(' IS LISTING OF INFLUENCE COEFFICIENTS REQUIRED ?')
(0432) READ(1,*)MM2
(0433) DO 4 J=1,KK
(0434) DO 3 JJ=2,NT
(0435) SAVE=0
(0436) K3=KN(JJ-1)+1
(0437) K4=KN(JJ)
(0438) DO 2 K=K3,K4
(0439) A1=(XBAR(J)-XBAR(K))/S(K)
(0440) A3=(ZBAR(J)-ZBAR(K))/S(K)
(0441) ASQ=A1*A1+A3*A3+0.25
(0442) F=A3*T1(K)-A1*T3(K)
(0443) FM=ABS(F)
(0444) E=A1*T1(K)+A3*T3(K)
(0445) EA=0.5-E
(0446) EB=0.5+E
(0447) IF(FM.LT.EPS)GO TO 10
(0448) U1=F/FM
(0449) THETA=ATAN(EA/FM)*U1
(0450) THETB=ATAN(EB/FM)*U1
(0451) GO TO 11
(0452) 10 U1=2*ABS(EA)
(0453) U2=2*ABS(EB)
(0454) THETA=EA*PI/U1
(0455) THETB=EB*PI/U2
(0456) 11 T=THETA+THETB
(0457) RNUML=ASQ-E
(0458) IF(RNUML.LT.EPS)GO TO 12
(0459) DENL=ASQ+E
(0460) IF(DENL.LT.EPS)GO TO 12
(0461) RL=0.5*ALOG(RNUML/DENL)
(0462) GO TO 13
(0463) 12 RL=0.0
(0464) 13 FT=F*T1(K)-E*T3(K)
(0465) ET=E*T1(K)+F*T3(K)
(0466) A=T*T1(K)
(0467) B=RL*T3(K)
(0468) C=T*T3(K)
(0469) A1=RL*T1(K)
(0470) U1=A-B
(0471) W1=C+A1
(0472) U2=T*ET+RL*FT-T3(K)
(0473) W2=RL*ET-T*FT+T1(K)
(0474) A=U1*T3(J)-W1*T1(J)
(0475) B=U2*T3(J)-W2*T1(J)
(0476) A=A*0.5
(0477) G(K)=A-B+SAVE
(0478) SAVE=A+B
(0479) IF (MM1.EQ.0.0)GOTO 2
(0480) IF (ABS(G(K)).LE.0.001) WRITE(1,120)J, JJ, K3, K4, K, G(K)

```

```

181) 120  FORMAT(1H , 'J=' , I3, 2X, 'JJ=' , I3, 2X, 'K3=' , I3, 2X, 'K4=' , I3, 2X,
182)      1 'K=' , I3, 2X, 'G(K)=' , F8.6)
183) 2    CONTINUE
184)      G(K3)=G(K3)-SAVE
185) 3    CONTINUE
186)      DENL=2*PI*(S1*T3(J)-S3*T1(J))
187)      K2=KK+1
188)      G(K2)=-DENL
189)      IF(MM2.EQ.0)GOTO 336
190)      KK2=10*INT(K2/10)
191)      WRITE(1,338)J
192) 338  FORMAT('J=' , I3)
193)      DO 50 KM=1, KK2, 5
194)      KM1=KM+1
195)      KM2=KM+2
196)      KM3=KM+3
197)      KM4=KM+4
198)      WRITE(1,333)KM, G(KM), KM1, G(KM1), KM2, G(KM2), KM3, G(KM3), KM4, G(KM
199) 333  FORMAT(5('G' , I3, '=' , F10.7, 1X))
200) 50   CONTINUE
201)      KL1=KK2+1
202)      KL2=KK2+2
203)      KL3=KK2+3
204)      KL4=KK2+4
205)      KL5=KK2+5
206)      KL6=KK2+6
207)      KL7=KK2+7
208)      KL8=KK2+8
209)      KL9=KK2+9
210)      WRITE(1,333)KL1, G(KL1), KL2, G(KL2), KL3, G(KL3), KL4,
211)      1G(KL4), KL5, G(KL5), KL6, G(KL6), KL7, G(KL7), KL8,
212)      1G(KL8), KL9, G(KL9)
213)      WRITE(1,345)
214) 345  FORMAT(1H ,)
215) 336  IF(J.NE.1)GO TO 150
216)      DO 151 K=1, KK
217)      D(K)=-G(K+1)/G(1)
218) 151  CONTINUE
219)      GO TO 4
220) 150  L=J-1
221)      KD=KK-L+1
222)      KE=KD-1
223)      DENL=G(J)
224)      DO 160 I=1, L
225) 160  DENL=DENL+G(I)*D(I*KD-KE)
226)      DO 152 K=1, KE
227)      RNUML=G(K+J)
228)      DO 161 I=1, L
229)      RNUML=RNUML+G(I)*D(I*KD-KE+K)
230) 161  CONTINUE
231)      H(K)=-RNUML/DENL
232) 152  CONTINUE
233)      DO 153 I=1, L
234)      P=D(I*KD-KE)
235)      DO 156 K=1, KE
236)      D((I-1)*KE+K)=H(K)*P+D(I*KD-KE+K)
237) 156  CONTINUE
238) 153  CONTINUE
239)      DO 154 K=1, KE
240)      D(L*KE+K)=H(K)
241) 154  CONTINUE
242) 4    CONTINUE

```



```

544)      G(K)=1-D(K)*D(K)
545) 16    CONTINUE
546)      CALL CC(X,Z,S,T1,T3,D,KN,KN,NT)
547)      RETURN
548)      END
549)  C
550)  C
551)  C ***** S U B R O U T I N E   02 *****
552)  C
553)  C
554)      SUBROUTINE CC(X,Z,S,T1,T3,D,KN,KN,NT)
555)      COMMON/BLOCK1/ALPHA,CL,CBAR,XM,ZM,CHORD
556)      COMMON/BLOCK2/CN(10),CM(10),COFP(10),CNT,CMT,COFPT
557)      COMMON/BL20/ITRACE
558)      DIMENSION X(KK),Z(KK),S(KK),T1(KK),T3(KK),D(KK),KN(NT)
559)      IF(ITRACE.EQ.1)WRITE(6,9999)
560) 9999  FORMAT(/,'SUBROUTINE   CC   '/)
561)      CMT=0.0
562)      CNT=0.0
563)      DO 111 L=2,NT
564)          I=KN(L-1)+1
565)          J=KN(L)-1
566)          DIR=X(I)-X(I+1)
567)          DO 112 K=I,J
568)              IF((X(K)-X(K+1))*DIR.LE.0.0)GOTO 113
569) 112    CONTINUE
570)          K=J
571) 113    KLE=K
572)          CX=X(I)-X(KLE)
573)          CZ=Z(I)-Z(KLE)
574)          C=SQRT(CX*CX+CZ*CZ)
575)          L1=L-1
576)          CN(L1)=0.0
577)          CM(L1)=0.0
578)          I=I+1
579)          DO 114 K=I,J
580)              K1=K+1
581)              A=D(K)*D(K)+D(K)*D(K1)+D(K1)*D(K1)
582)              CN(L1)=CN(L1)+S(K)*A*(T1(K)*CX+T3(K)*CZ)
583)              CM(L1)=CM(L1)+S(K)*(A/3.0*((XM-0.5*(X(K)+X(K1)))*T1(K)+(ZM-0.5
584) 1(Z(K)+Z(K1)))*T3(K))-(D(K1)*D(K1)-D(K)*D(K))*S(K)/12.0)
585) 114    CONTINUE
586)          CN(L1)=CN(L1)/(3*C*C)
587)          CM(L1)=CM(L1)/(CBAR*C)
588)          IF(ABS(CN(L1)).LE.0.000001)GOTO 116
589)          COFP(L1)=- (CM(L1)*CBAR/CN(L1)-(CZ*(ZM-Z(KLE))+CX*(XM-X(KLE)))/
590)          GOTO 117
591) 116    COFP(L1)=1000.0
592) 117    CONTINUE
593)          IF(L.NE.2)GOTO 115
594)          RNX=-CZ/C
595)          RNZ=CX/C
596) 115    CNT=CNT+CN(L1)*(RNZ*CX-RNX*CZ)
597)          CMT=CMT+CM(L1)*C
598) 111    CONTINUE
599)          CNT=CNT/CBAR
600)          CMT=CMT/CBAR
601)          ZO=0.0
602)          XO=0.0
603)          IF(ABS(CNT).LE.0.000001)GOTO 118
604)          COFPT=- (CMT*CBAR/CNT+RNX*(ZM-ZO)-RNZ*(XM-XO))

```

```

0605)      GOTO 119
0606) 118  COFPT=1000.0
0607) 119  CONTINUE
0608)      CL=0.0
0609)      DO 17 L=2,NT
0610)      I=KN(L-1)+3
0611)      J=KN(L)-2
0612)      DO 17 K=I,J
0613)      CL=CL+S(K)*(D(K+1)+D(K))
0614) 17   CONTINUE
0615)      CL=-CL/CBAR
0616)      RETURN
0617)      END
0618)  C
0619)  C
0620)  C ***** SUBROUTINE 03 *****
0621)  C
0622)  C
0623)      SUBROUTINE PLOT(X,Z,L,NT,KK)
0624)      COMMON/BL20/ITRACE
0625)      DIMENSION X(KK),Z(KK),L(NT)
0626)      IF(ITRACE.EQ.1)WRITE(6,9999)
0627) 9999  FORMAT(/,'SUBROUTINE PLOT ' //)
0628)      DO 1 I=2,NT
0629)      K30=L(I)
0630)      I1=I-1
0631)      K20=L(I1)+1
0632)      CALL GRAMOV (X(K20),Z(K20))
0633)      DO 2 J=K20,K30
0634)      CALL GRALIN (X(J),Z(J))
0635) 2     CONTINUE
0636)      CALL GRALIN(X(K20),Z(K20))
0637) 1     CONTINUE
0638)      CALL PICCLE
0639)      RETURN
0640)      END
0641)  C
0642)  C
0643)  C ***** SUBROUTINE 04 *****
0644)  C
0645)  C
0646)      SUBROUTINE MAX(X,KK,Y45,Y46,K47,Y55,Y56,Y57,Y65)
0647)      COMMON/BL20/ITRACE
0648)      DIMENSION X(KK)
0649)      IF(ITRACE.EQ.1)WRITE(6,9999)
0650) 9999  FORMAT(/,'SUBROUTINE MAX ' //)
0651)      DO 1 I=1,KK
0652)      IF(Y45.GT.ABS(X(I)))GOTO 1
0653)      Y45=ABS(X(I))
0654) 1     CONTINUE
0655)      Y55=-INT(Y45)-2
0656)      Y56=Y55-3
0657)      Y57=Y55-2
0658)      Y46=Y45+1
0659)      Y65=INT(Y45)*10+20
0660)      K47=Y46+2
0661)      WRITE(1,10)Y45,Y46,K47,Y55,Y56,Y57,Y65
0662) 10    FORMAT(1H ,7F14.6)
0663)      RETURN
0664)      END
0665)  C
0666)  C

```

```

7) C
3) C
7) C
)) SUBROUTINE BLCALC(G,D,S,KN,KN,NT,IPAR,DUDS)
1) COMMON/BLOCK3/RE1,ISTAG(10)
2) COMMON/BL16/TIN,HIN,TIN1,HIN1,TIN2,HIN2,TIN3,HIN3
3) COMMON/BL18/CP(99),CF(99)
4) COMMON/BL17/XBAR(99),ZBAR(99)
5) COMMON/BL20/ITRACE
5) COMMON/BL25/CDU1,CDU2,CDL1,CDL2
7) C
3) C
7) C
)) DIMENSION G(KK),D(KK),S(KK),KN(10),SU(99),SL(99),UUP(99),DUUP1(9
1) 1,DUUP2(99),UL(99),DUL1(99),DUL2(99),DEL1(99),DEL2(99),DUDS(99),
2) 1DELTL1(99),DELTL2(99),H12(99),DDSU(99),DDSL(99),DAMMY(99),CFL(99
3) +,CFU(99),UDSU(99),UDSL(99),XBU(99),XBL(99)
4) IF(ITRACE.EQ.1)WRITE(6,9999)
5) 9999 FORMAT(/,'SUBROUTINE BLCALC'//)
5) RE=RE1
7) ISTART=1
3) C SEARCH FOR LEADING EDGE STAGNATION POINT
7) DO 20 L=2,NT
0) L1=KN(L-1)+1
1) L2=KN(L)
2) RK3=0
3) N1=L1+3
4) N2=L2-3
5) DO 2 I=N1,N2
6) IF(G(I).GT.RK3)I1=I
7) IF(G(I).GT.RK3)RK3=G(I)
8) 2 CONTINUE
9) ISTAG(L)=I1
0) 100 WRITE(1,200)
1) C
2) C OUTPUT CP NEAR LE AND TE
3) C
4) WRITE(1,202)G(L1),G(I1-1),G(L1+1),G(I1),G(L1+2),G(I1+1)
5) WRITE(1,204)
6) C
7) C DECISION TO STOP OR CONTINUE THE CALCULATION
8) C
9) READ(1,*)M1
0) IF(M1.EQ.0)WRITE(1,206)
1) IF(M1.EQ.0)STOP
2) C
3) WRITE(1,216)
4) 216 FORMAT(/,'ENTER LAMINAR BOUNDARY LAYER METHOD'///)
5) WRITE(1,218)
6) 218 FORMAT(20X,'UPPER SURFACE',11X,'LOWER SURFACE'//,
7) 112X,'Curle',9X,'1',23X,'1'//,
8) 112X,'Thwaites',6X,'2',23X,'2'///)
9) READ(1,*)N4,N5
0) WRITE(1,220)
1) 220 FORMAT(/,'ENTER TURBULENT BOUNDARY LAYER METHOD'///,
2) +20X,'UPPER SURFACE',11X,'LOWER SURFACE'//,
3) +12X,'Head',10X,'1',23X,'1'//,
4) +12X,'Head-Patel',4X,'2',23X,'2'///)
5) READ(1,*)N6,N7
6) C
7) C CALCULATE DISTANCE OF LEADING EDGE STAGNATION POINT FROM
8) C THE END OF EACH SEGMENT

```

```

'29) C
'30) C
'31) C    UPPER SURFACE
'32) C
'33)     LL2=L2-IPAR
'34)     SUM=0
'35)     SU(1)=0.0
'36)     I2=1
'37)     XBU(1)=XBAR(I1)
'38)     DO 4 I=I1,LL2
'39)     I2=I2+1
'40)     SUM=SUM+S(I)
'41)     SU(I2)=SUM
'42)     XBU(I2)=XBAR(I+1)
'43) C    WRITE(1,123) I2,SU(I2)
'44) 123  FORMAT(I3,3X,F12.7)
'45) 4    CONTINUE
'46)     K1=I2
'47) C
'48) C    LOWER SURFACE
'49) C
'50)     SL(1)=0.0
'51)     SUM=0.
'52)     I2=1
'53)     I3=I1-1
'54)     XBL(1)=XBAR(I1)
'55)     DO 6 I4=L1,I3
'56)     I2=I2+1
'57)     I=I1-1+L1-I4
'58)     SUM=SUM+S(I) ?
'59)     SL(I2)=SUM
'60)     XBL(I2)=XBAR(I)
'61) 6    CONTINUE
'62)     K2=I2
'63)     I2=0
'64)     DO 16 I=I1,L2
'65)     I2=I2+1
'66)     UUP(I2)=ABS(D(I))
'67) 16   CONTINUE
'68)     IF(IPAR.EQ.1)GOTO 90
'69)     IF(ABS(UUP(1)).GT.0.1)UUP(1)=.1
'70)     I2=I2+1
'71)     UUP(I2)=.1
'72) 90   CONTINUE
'73)     I2=0
'74)     DO 18 I=L1,I1
'75)     I2=I2+1
'76)     I4=I1+L1-I
'77)     UL(I2)=ABS(D(I4))
'78)     IF(I.EQ.I1)UL(I2)=.1
'79) 18   CONTINUE
'80)     UL(1)=UUP(1)
'81)     CALL DU1DU2(UUP,SU,DUUP1,DUUP2,K1)
'82) 40   CONTINUE
'83)     IF(N4.EQ.1)CALL LAMBL(UUP,DUUP1,DUUP2,SU,DEL1,DEL2,K1,JU,RE,CFI)
'84)     IF(N4.EQ.2)CALL LAMBL1(UUP,DUUP1,SU,DEL1,DEL2,K1,JU,RE,CFU)
'85)     IF(N6.EQ.1)CALL TURBL1(SU,UUP,DEL1,DEL2,DUUP1,JU,HU,RE,K1,CFU)
'86)     IF(N6.EQ.2)CALL TURBL2(SU,UUP,DEL1,DEL2,DUUP1,JU,HU,RE,K1,CFU)
'87)     DO 8 I=1,K1
'88)     IF(DEL2(I).LE.0)DEL2(I)=DEL2(I-1)*1.2
'89)     IF(DEL1(I).LE.0)DEL1(I)=DEL1(I-1)+.0006
'90)     H12(I)=DEL1(I)/DEL2(I)

```

```

791)      UDSU(I)=UUP(I)*DEL1(I)
792)  8    CONTINUE
793)      IF(L.EQ.2)TIN=DEL2(K1)
794)      IF(L.EQ.2)HIN=DEL1(K1)/DEL2(K1)
795)      IF(L.EQ.3)TIN2=DEL2(K1)
796)      IF(L.EQ.3)HIN2=DEL1(K1)/DEL2(K1)
797)      CDC=2.*DEL2(K1)*UUP(K1)**(.5*H12(K1)+2.5)
798)      IF(L.EQ.2)CDU1=CDC
799)      IF(L.EQ.3)CDU2=CDDC/.2
300)      IF(L.EQ.2)WRITE(6,224)
301)      IF(L.EQ.3)WRITE(6,226)
302)  224  FORMAT(/,'***** FIRST ELEMENT *****',//)
303)  226  FORMAT(/,'***** SECOND ELEMENT *****',//)
304)      WRITE(6,208)
305)      WRITE(6,210)
306)      WRITE(6,212)(I,XBU(I),UUP(I),DUUP1(I),DUUP2(I),SU(I),DEL1(I),
307)  1DEL2(I),H12(I),XBU(I),I=1,K1)
308)      CALL DU1DU2(UL,SL,DUL1,DUL2,K2)
309)      IF(N5.EQ.1)CALL LAMBL(UL,DUL1,DUL2,SL,DELTL1,DELTL2,K2,JL,RE,CF)
310)      IF(N5.EQ.2)CALL LAMBL1(UL,DUL1,SL,DELTL1,DELTL2,K2,JL,RE,CFL)
311)      IF(N7.EQ.1)CALL TURBL1(SL,UL,DELTL1,DELTL2,DUL1,JL,HL,RE,K2,CF)
312)      IF(N7.EQ.2)CALL TURBL2(SL,UL,DELTL1,DELTL2,DUL1,JL,HL,RE,K2,CF)
313)      DO 10 I=1,K2
314)      IF(DELTL2(I).EQ.0)DELTL2(I)=DELTL2(I-1)*1.2
315)      IF(DELTL1(I).EQ.0)DELTL1(I)=DELTL1(I-1)+.0006
316)      H12(I)=DELTL1(I)/DELTL2(I)
317)      UDSL(I)=UL(I)*DELTL1(I)
318)  10    CONTINUE
319)      CDC=2.*DEL2(K2)*UL(K2)**(.5*H12(K2)+2.5)
320)      IF(L.EQ.2)CDL1=CDC
321)      IF(L.EQ.3)CDL2=CDC/.2
322)      IF(L.EQ.2)TIN1=DELTL2(K2)
323)      IF(L.EQ.2)HIN1=DELTL1(K2)/DELTL2(K2)
324)      IF(L.EQ.3)TIN3=DELTL2(K2)
325)      IF(L.EQ.3)HIN3=DELTL1(K2)/DELTL2(K2)
326)      WRITE(6,214)
327)      WRITE(6,210)
328)      WRITE(6,212)(I,XBL(I),UL(I),DUL1(I),DUL2(I),SL(I),DELTL1(I),
329)  1DELTL2(I),H12(I),XBL(I),I=1,K2)
330)      CALL SPLINE(SU,UDSU,DDSU,DAMMY,K1,5,1)
331)      IND=ISTART+K2-1
332)      DO 22 I=1,K1
333)      IND=IND+1
334)      DUDS(IND)=UDSU(I)
335)      CF(IND)=CFU(I)
336)  22    CONTINUE
337)      IND=K2
338)      K22=K2-1
339)      CALL SPLINE(SL,UDSL,DDSL,DAMMY,K2,5,1)
340)      DO 24 I=1,K22
341)      J=K2-I
342)      DUDS(J)=UDSL(I)
343)      CF(J)=CFL(I)
344)  24    CONTINUE
345)      ISTART=ISTART+K1+K2-1
346)  20    CONTINUE
347)  200  FORMAT(/,12X,'CP Trailing Edge'                CP Leading Edge')
348)  202  FORMAT(3(15X,F10.6,15X,F10.6//))
349)  204  FORMAT(/,5X,'If CP Values Acceptable enter 1 to continue.'//
350)  1'    'If programme termination is required enter 0')
351)  206  FORMAT('***          PROGRAMME ENDS          ***')
352)  208  FORMAT(/,32X,'U P P E R    S U R F A C E ',//)

```

```

(0853) 210  FORMAT(1X,' I' ,6X,' XBAR' ,7X,' U' ,10X,' DU1' ,8X,' DU2' ,9X,' S'
(0854)      +,10X,' DEL1' ,8X,' DEL2' ,10X,' H' ,8X,' XBAR' ,/)
(0855) 212  FORMAT((I3,2(2X,F8.5),5(2X,F10.7),2(2X,F9.5)))
(0856) 214  FORMAT(/,32X,' L O W E R   S U R F A C E ' //)
(0857)      RETURN
(0858)      END
(0859)  C
(0860)  C
(0861)  C      ***** S U B R O U T I N E   06 *****
(0862)  C
(0863)  C
(0864)      SUBROUTINE DU1DU2(U,S,DU1,DU2,K1)
(0865)      COMMON/BL20/ITRACE
(0866)  C
(0867)  C      CALCULATES DU1=DU/DS  DU2=D/DS(DU1)
(0868)  C
(0869)  C      DU1    DU2    ARE CALCULATED AT ELEMENT ENDS
(0870)  C
(0871)  C      DU1    DU2    ARE CALCULATED AT ELEMENT ENDS
(0872)  C      FOR FIRST AND LAST ELEMENT DU1 DU2 ARE CALCULATED
(0873)  C      AT ELEMENT MIDPOINT
(0874)  C
(0875)      DIMENSION U(K1),S(K1),DU1(K1),DU2(K1)
(0876)      IF(ITRACE.EQ.1)WRITE(6,9999)
(0877) 9999  FORMAT(/,'SUBROUTINE  DU1DU2' //)
(0878)  C
(0879)  C
(0880)  C      CALCULATION OF DU1
(0881)  C
(0882)      DU1(1)=(U(2)-U(1))/S(2)
(0883)      K2=K1-1
(0884)      DO 8 I=2,K2
(0885)      DU1(I)=(U(I+1)-U(I-1))/(S(I+1)-S(I-1))
(0886) 8      CONTINUE
(0887)      DU1(K1)=(U(K1)-U(K1-1))/(S(K1)-S(K1-1))
(0888)      DO 14 I=1,K1
(0889)      IF(ABS(DU1(I)).GT.150.)DU1(I)=(DU1(I)/ABS(DU1(I)))*150.
(0890) 14      CONTINUE
(0891)  C
(0892)  C      CALCULATION OF DU2
(0893)  C
(0894)      U21=(U(2)+U(1))/2
(0895)      DS=S(2)/2
(0896)      DU211=(U21-U(1))/DS
(0897)      DU221=(U(2)-U21)/DS
(0898)      DU32=(DU1(3)+DU1(2))/2
(0899)      DU2(1)=(DU221-DU211)/DS
(0900)      DU2(2)=(DU32-DU1(1))/(S(2)/2)
(0901)      K2=K1-2
(0902)      DO 10 I=3,K2
(0903)      DU2(I)=(DU1(I+1)-DU1(I-1))/(S(I+1)-S(I-1))
(0904) 10      CONTINUE
(0905)      UAV=(U(K1)+U(K1-1))/2
(0906)      DS1=(S(K1)-S(K1-1))/2
(0907)      DUK1=(UAV-U(K1-1))/DS1
(0908)      DUK2=(U(K1)-UAV)/DS1
(0909)      DUKM1=(DU1(K1-1)+DU1(K1-2))/2
(0910)      DU2(K1-1)=(DU1(K1)-DUKM1)/((S(K1)-S(K1-2))/2)
(0911)      DU2(K1)=(DUK2-DUK1)/DS1
(0912)      DO 12 I=1,K1
(0913)      IF(ABS(DU2(I)).GT.1500.)DU2(I)=(DU2(I)/ABS(DU2(I)))*1500.
(0914) 12      CONTINUE

```

```

15) 100 RETURN
16) END
17) C
18) C
19) C ***** SUBROUTINE 07 *****
20) C
21) C
22) SUBROUTINE LAMBL(U,DU1,DU2,S,DEL1,DEL2,K1,J,RE,CF9)
23) COMMON/BL20/ITRACE
24) DIMENSION U(K1),DU1(K1),DU2(K1),S(K1),DEL1(K1),CF9(K1),
25) 1DEL2(K1),GLM(80),RL(24),F(24),F11(24),G11(24)
26) REAL LAM,MEE
27) C
28) DATA RL/.0855,.08,.07,.06,.05,.04,.03,.02,.01,.0,-.01,-.02,-.03
29) 1-.04,-.05,-.06,-.07,-.08,-.09,-.1,-.11,-.12,-.13,-.133/,
30) 2F/0.,.0258,.0736,.1225,.1724,.2236,.2761,.3299,.3848,.4410,.498
31) 3.5572,.6167,.6777,.7404,.8053,.8729,.9434,1.0166,1.0928,1.1724,
32) 41.254,1.3373,1.3686/
33) DATA F11/.1296,.1236,.1128,.1025,.0925,.083,.0738,.0651,.0567,
34) 1.0487,.0411,.0338,.027,.0207,.0149,.0095,.0047,.001,-.0019,
35) 2-.0039,-.0051,-.0055,-.0051,-.0047/,
36) 3G11/.2626,.2535,.2378,.2228,.2087,.1953,.1827,.171,.16,.1498,
37) 4.1404,.1318,.1240,.116,.1075,.0971,.0852,.0728,.0601,.0470,.033
38) 5.0197,.0054,.0/
39) IF(ITRACE.EQ.1)WRITE(6,9999) ,
40) 9999 FORMAT(/,'SUBROUTINE LAMBL'//)
41) J2=0
42) J3=0
43) J4=0
44) SUM=0.0
45) DEL2(1)=SQRT(.0854/(RE*DU1(1)))
46) DEL1(1)=2.31*DEL2(1)
47) K10=0
48) SUMJ=0.0
49) DO 12 J=2,K1
50) K10=0
51) GLM(J)=0.
52) 100 SUMJ=(U(J)**5+U(J-1)**5)*.5*(S(J)-S(J-1))*(1+2.22*GLM(J))
53) DEL2SQ=.45*(SUM+SUMJ)/(U(J)**6*RE)
54) DEL2(J)=SQRT(DEL2SQ)
55) IF(K10.GT.3)GOTO 110
56) LAM=DEL2SQ*RE*DU1(J)
57) IF(LAM.EQ.0.)H1=2.61
58) IF(LAM.EQ.0)GOTO 135
59) IF(LAM.GT.0.09)LAM=0.0855
60) IF(LAM.LT.-.09)GOTO 115
61) MEE=(DEL2SQ*RE)**2*U(J)*DU2(J)
62) IF(ABS(MEE).GT.1.)GOTO 115
63) GO=.66+3*LAM
64) CALL INPOL(RL,F,24,LAM,FO)
65) GLM(J)=FO-.45+6*LAM-MEE*GO
66) K10=K10+1
67) GOTO 100
68) 110 CAPL=FO-MEE*GO
69) CALL INPOL(RL,F11,24,LAM,F1)
70) CALL INPOL(RL,G11,24,LAM,G1)
71) SMALSQ=F1-MEE*G1
72) SMAL=SQRT(ABS(F1-MEE*G1))
73) IF(F1-MEE*G1.LT.0..AND.J2.EQ.0..AND.J3.EQ.0..AND.J4.EQ.0)GOTO 125
74) H1=((2*SMAL-CAPL)/(2*LAM))-2
75) 135 DEL1(J)=H1*DEL2(J)
76) SUM=SUM+SUMJ

```

```

777) REDEL1=DEL1(J)*RE*U(J)
778) RES=S(J)*U(J)*RE
779) REDEL2=DEL2(J)*U(J)*RE
780) C WRITE(1,202)J,H1,DEL1(J),LAM
781) CALL TRANS(RES,REDEL1,REDEL2,LAM,J,J2,J3,J4)
782) C WRITE(1,204)REDEL1,REDEL2,RES
783) IF(J.GT.INT(.67*K1))GOTO 130
784) IF(J2.NE.0.AND.J3.NE.0.AND.J4.NE.0)GOTO 130
785) 115 IF(LAM.LT.-.09.AND.J2.EQ.0.AND.J3.EQ.0.AND.J4.EQ.0)GOTO 125
786) IF(LAM.LT.-.09.AND.(J2.NE.0.OR.J3.NE.0.OR.J4.NE.0))GOTO 120
787) 12 CONTINUE
788) 120 WRITE(1,206)
789) READ(1,*)J,M10
790) IF(M10.EQ.0)STOP
791) GOTO 130
792) 125 WRITE(1,208)
793) H1=-((F0-F1*G0/G1)/(2*SMAL))-2
794) DEL1(J)=H1*DEL2(J)
795) WRITE(1,202)J,H1,DEL1(J),LAM
796) J10=J-1
797) CALL BUBBLE(S,U,DEL2,DEL1,J10,K1,RE,IR)
798) J=IR
799) 202 FORMAT(//,'J=',I3,2X,'H =',F9.6,4X,'DEL1=',F9.6,4X,'LAM =',F9.
000) 204 FORMAT('REDEL1 =',F10.4,4X,'REDEL2 =',F10.4,4X,'RES =',E14.4)
001) 206 FORMAT(//,'T R A N S I T I O N   N O D E   N O T   D E F I N E
002) 1,//,1H,'ENTER TRANSITION NODE #,1 TO CONTINUE',//
003) 210X,'OR 0,0 TO STOP')
004) 208 FORMAT(//,'LAMINAR SEPARATION ',//
005) 11H,'VALUES OF H,DELTA1 AND LAMBDA AT SEPARATION,FOLLOW',//)
006) 130 RETURN
007) END
008) C
009) C
010) C ***** SUBROUTINE 08 *****
011) C
012) C
013) SUBROUTINE INPOL(X,Y,N,LAM,Y1)
014) DIMENSION X(N),Y(N)
015) REAL LAM
016) IF(X(1).GT.X(2))GOTO 8
017) DO 2 I=1,N
018) 2 IF(X(I).GE.LAM)GOTO 4
019) 8 DO 1 I=1,N
020) IF(X(I).LT.LAM)GOTO 4
021) IF(X(I).EQ.LAM)GOTO 5
022) 1 CONTINUE
023) 4 I1=I-1
024) X0=X(I)
025) X2=X(I1)
026) Y1=Y(I)
027) Y3=Y(I1)
028) A=(Y1-Y3)/(X0-X2)
029) B=(X0*Y3-X2*Y1)/(X0-X2)
030) Y1=A*LAM+B
031) 5 IF(X(I).EQ.LAM)Y1=Y(I)
032) 6 RETURN
033) END
034) C
035) C ***** SUBROUTINE 09 *****
036) C
037) SUBROUTINE TURBL1(S,U,DEL1,DEL2,DU1,J1,H1,RE,K1,CF9)
038) C

```



```

1039) C HEADS METHOD IS USED TO CALCULATE THE INCOMPRESSIBLE
1040) C TURBULENT BOUNDARY LAYER
1041) C
1042) C
1043) COMMON/BL20/ITRACE
1044) DIMENSION S(K1),U(K1),DEL1(K1),DEL2(K1),DU1(K1),CF9(K1)
1045) IF(ITRACE.EQ.1)WRITE(6,9999)
1046) 9999 FORMAT(/,'SUBROUTINE TURBL1'//)
1047) J2=K1-1
1048) RED2=RE*DEL2(J1)*U(J1)
1049) A1=ALOG10(RED2)
1050) H1=.0361065*A1**2-.3859576*A1+2.3061293
1051) DO 2 J=J1,J2
1052) D2UDU=DEL2(J)*DU1(J)/U(J)
1053) REDEL2=DEL2(J)*U(J)*RE
1054) CF=.246*EXP(-1.561*H1)/REDEL2**.268
1055) ENTR=.025*H1-.022
1056) DH=-1./DEL2(J)*(H1*(H1**2-1)*D2UDU+.5*(H1-1)*((H1**2-1)*ENTR-
1057) 1*CF))
1058) IF(ABS(DH).GT.10.)DH=DH/ABS(DH)*10.
1059) DDEL2=.5*CF-D2UDU*(H1+2)
1060) C WRITE(1,202)J,H1,DH,DDEL2
1061) SAV=(S(J+1)-S(J))*5
1062) DEL22=DEL2(J)+DDEL2*SAV
1063) H2=DH*SAV+H1
1064) UAV=(U(J+1)+U(J))*5
1065) DU1AV=(DU1(J+1)+DU1(J))*5
1066) D2UDU=DEL22*DU1AV/UAV
1067) REDEL2=DEL22*UAV*RE
1068) CF1=.246*EXP(-1.561*H2)/REDEL2**.268
1069) ENTR1=.025*H2-.022
1070) DH1=-1./DEL22*(H2*(H2**2-1)*D2UDU+.5*(H2-1)*((H2**2-1)*
1071) 1ENTR1-H2*CF1))
1072) IF(ABS(DH1).GT.10.)DH1=DH1/ABS(DH1)*10.
1073) DDEL21=.5*CF1-D2UDU*(H2+2)
1074) DEL2(J+1)=DEL2(J)+DDEL21*(S(J+1)-S(J))
1075) H1=H1+DH1*(S(J+1)-S(J))
1076) IF(H1.GT.1.9)WRITE(1,204)J
1077) IF(H1.GT.1.9)H1=1.9
1078) DEL1(J+1)=H1*DEL2(J+1)
1079) J3=J+1
1080) 2 CONTINUE
1081) WRITE(1,200)J3,H1
1082) GOTO 102
1083) 100 WRITE(1,204)J
1084) 200 FORMAT(I2,3X,'H=',F10.7)
1085) 202 FORMAT(I3,3X,'H=',F10.7,4X,'DH=',F10.7,4X,'DDELTA2=',F10.7)
1086) 204 FORMAT(/1H,'TURBULENT SEPARATION AT NODE',I3,/)
1087) 102 RETURN
1088) END
1089) C
1090) C ***** S A B R O U T I N E 10 *****
1091) C
1092) SUBROUTINE TRANS(RES,REDEL1,REDEL2,LAM,J,J2,J3,J4)
1093) C
1094) C
1095) C
1096) C
1097) REAL LAM,LAM1
1098) DIMENSION RLG(50),RLLG(50),RMT(8),RD2C(8),RD2M(10),RSM(10),
1099) 1RLP(25),RD1(25)
1100) DATA RLG/12.5,12.0,11.5,11.0,10.5,10.0,9.5,9.0,8.5,8.0,

```

```

(1) + 7.5, 7.0, 6.5, 6.0, 5.5, 5.0, 4.5, 4.0, 3.5, 3.0,
(2) + 2.5, 2.0, 1.5, 1.0, 0.5, 0.0, -0.5, -1.0, -1.5, -2.0,
(3) + -2.5, -3.0, -3.5, -4.0, -4.5, -5.0, -5.5, -6.0, -6.5, -7.0,
(4) + -7.5, -8.0, -8.5, -9.0, -9.5, -10., -10.5, -11., -11.5, -12./
(5) DATA RLLG/0.09463,0.094815,0.094630,094083,0.09317,0.091883,
(6) +0.0902338,0.0882231,0.0858547,0.0831341,0.0800679,0.0766638,
(7) +0.0729304,0.0698775,0.0645158,0.0598566,0.0549126,0.0496967,
(8) +0.0442231,0.0395063,0.0325618,0.0264054,0.0200537,0.0135238,
(9) +0.0068332,0.0000000,-.0069575,-.0140205,-.0211700,-.0283867,
(0) +-.0356510,-.0429432,-.0502436,-.0575323,-.0647893,-.0719947,
(1) +-.0791287,-.0861716,-.0931037,-.0999057,-.1065584,-.1130430,
(2) +-.1193409,-.1254340,-.1313046,-.1369354,-.1423097,-.1474112,
(3) +-.1522245,-.1567347/
(4) C
(5) C CRABTREE TRANSITION DATA
(6) C
(7) DATA RMTC/.09,.04,.02,.0,-.0046,-.0078,-.01,-.0154/
(8) DATA RD2C/752.,1012.,1125.,1291.,1400.,1600.,1800.,2400./
(9) C
(10) C MICHEL TRANSITION DATA
(11) C
(12) DATA RD2M/1457.,1390.,1322.,1238.,1152.,1052.,941.,832.,707.,573
(13) 1,RSM/5.E06,4.5E06,4.E06,3.5E06,3.E06,2.5E06,2.0E06,1.5E06,1.E06,
(14) 2.5E06/
(15) C
(16) C POHLHAUSEN TRANSITION DATA
(17) C
(18) DATA RLP/8.,7.,6.15,5.57,5.,4.48,4.08,3.57,3.3,2.64,2.,1.38,
(19) 1.81,.59,.49,.18,.0,-.18,-.41,-.81,-1.56,-1.95,-2.57,-3.74,-5.89,
(20) 2,RD1/11670.,11370.,10000.,9000.,8000.,7000.,6000.,5000.,4000.,
(21) 33000.,2000.,1500.,1000.,900.,800.,700.,645.,600.,500.,400.,300.
(22) 4,250.,200.,150.,100./
(23) C
(24) RED2=0.
(25) RED1=0.
(26) AD2=0.
(27) CAPGL=0.
(28) N=50
(29) CALL INPOL(RLLG,RLG,N,LAM,CAPGL)
(30) N1=25
(31) IF(CAPGL.GT.8.0.OR.CAPGL.LT.-5.89)GOTO 100
(32) CALL INPOL(RLP,RD1,N1,CAPGL,RED1)
(33) A1=ABS(RED1-REDEL1)
(34) IF(A1.LT.60.)WRITE(1,200)J
(35) IF(A1.LT.60.)J2=J
(36) LAM1=-LAM
(37) N2=8
(38) 100 IF(LAM1.GT.0.09.OR.LAM1.LT.-0.0154)GOTO 102
(39) CALL INPOL(RMTC,RD2C,N2,LAM1,RED2)
(40) A2=ABS(RED2-REDEL2)
(41) IF(A2.LT.110.)WRITE(1,202)J
(42) IF(A2.LT.110.)J3=J
(43) N3=10
(44) 102 IF(RES.GT.5.E06.OR.RES.LT..5E06)GOTO 104
(45) CALL INPOL(RSM,RD2M,N3,RES,AD2)
(46) A3=ABS(AD2-REDEL2)
(47) IF(A3.LT.70.)WRITE(1,204)J
(48) IF(A3.LT.70.)J4=J
(49) 104 CONTINUE
(50) 200 FORMAT(//,'POHLHAUSEN TRANSITION AT NODE',I3,/)
(51) 202 FORMAT(//,'CRABTREE TRANSITION AT NODE',I3,/)
(52) 204 FORMAT(//,'MICHEL TRANSITION AT NODE',I3,/)

```

```

(1163) 206   FORMAT('CAPGL=',F10.7,2X,'RED1=',E14.5,2X,'RED2='E14.5,
(1164)       12X,'AD2=',E14.5)
(1165)       IF(J2.NE.0.AND.J3.NE.0.AND.J4.NE.0)J=(J2+J3+J4)/3.+5
(1166) 140   RETURN
(1167)       END
(1168) C
(1169) C
(1170) C   ***** S U B R O U T I N E   11 *****
(1171) C
(1172)
(1173)       SUBROUTINE BUBBLE(S,U,DEL2,DEL1,J1,K1,RE,IR)
(1174)       DIMENSION S(K1),U(K1),DEL2(K1),DEL1(K1)
(1175)       D2S=DEL2(J1)
(1176)       SA=.5*(S(J1)+S(J1+1))
(1177)       US=.5*(U(J1)+U(J1+1))
(1178)       RED2S=RE*DEL2(J1)*US
(1179)       LBUB=6.E04/RED2S
(1180)       TF=.011
(1181)       EX=EXP(TF*10.)
(1182)       EX1=1./EX
(1183)       RL1=D2S*2.5E04*ALOG10((EX+EX1)/(EX-EX1))/RED2S
(1184)       RL1=D2S*4.E04/RED2S
(1185)       SRL1=SA+RL1
(1186)       DO 2 J=J1,K1
(1187)       IF(S(J).GT.SRL1)GOTO 100
(1188) 2     IF(S(J).EQ.SRL1)GOTO 102
(1189) 100   STEP=(S(J)-SRL1)/10.
(1190)       ST1=SRL1
(1191)       DIF=S(J)-S(J-1)
(1192)       J10=J
(1193)       DO 10 I=1,10
(1194)       ST1=ST1+STEP
(1195)       DIF1=S(J)-ST1
(1196)       R1=DIF1/DIF
(1197)       UST1=U(J)-R1*(U(J)-U(J-1))
(1198)       UR=UST1/US
(1199)       RL2=ST1-SRL1
(1200)       CALL BUBLE1(RL2,UR,D2S,SA,ST1,LBUB,I,J,IR)
(1201)       IF(IR.NE.0)GOTO 110
(1202) 10   CONTINUE
(1203) 102   K2=K1-1
(1204)       DO 12 I=J,K2
(1205)       J10=I
(1206)       ST1=S(I)
(1207)       STEP=(S(I+1)-S(I))/10.
(1208)       DIF=S(I+1)-S(I)
(1209)       DO 14 I1=1,10
(1210)       ST1=ST1+STEP
(1211)       DIF1=S(I+1)-ST1
(1212)       UST1=U(I)+R1*(U(I+1)-U(I))
(1213)       UR=UST1/US
(1214)       RL2=ST1-SRL1
(1215)       CALL BUBLE1(RL2,UR,D2S,SA,ST1,LBUB,I1,J,IR)
(1216)       IF(IR.NE.0)GOTO110
(1217) 14   CONTINUE
(1218) 12   CONTINUE
(1219)       J2=J
(1220) 110   RL2=ST1-SRL1
(1221)       RT=RL1+RL2
(1222)       IF(IR.EQ.2)GOTO 118
(1223)       WRITE(1,200)RL1,S(J1),RL2,SRL1,RT,ST1
(1224) 200   FORMAT('//,27X,'BUBBLE  CALCULATION  RESULTS'//,

```

```

(1225) +15X,'Laminar Length=',F8.4,6X,'Separation at S=',F8.4/,
(1226) +15X,'Turbulent Length=',F8.4,6X,'Transition at S=',F8.4/,
(1227) +15X,'Total Length=',F8.4,6X,'Reattachm. at S=',F8.4//)
(1228) 118 A=.0182/4./1.5
(1229) DEL2R=D2S*(1./UR**3+A*RL2*(1-UR**4)/(UR**3*(1-UR)))
(1230) IF(S(J2).GT.SRL1)GOTO 112
(1231) DO 16 J=J2,K1
(1232) IF(SRL1.LT.S(J))GOTO 114
(1233) DEL2(J)=D2S
(1234) DEL1(J)=D2S*1.5
(1235) 16 J3=J
(1236) 112 DEL2(J2)=D2S
(1237) J3=J2+1
(1238) DEL1(J2)=1.5*D2S
(1239) 114 DO 18 J=J3,K1
(1240) IF(S(J).GT.ST1)GOTO 116
(1241) DEL2(J)=(S(J)-SRL1)/RL2*(DEL2R-D2S)+D2S
(1242) DEL1(J)=1.5*DEL2(J)
(1243) 18 CONTINUE
(1244) 116 IR=J
(1245) DEL2(IR)=DEL2R
(1246) RETURN
(1247) END
(1248) C
(1249) C
(1250) C ***** S U B R O U T I N E 12 *****
(1251) C
(1252) C
(1253) SUBROUTINE BUBLE1(RL2,UR,D2S,SA,ST1,LBUB,I,J,IR)
(1254) A=.0182/4./1.5
(1255) B1=1./(A+.0082)
(1256) RL2=RL2/D2S
(1257) ERROR=UR**4-(A+(1-UR)/RL2)*B1
(1258) IF(ER*ERROR.LT.0..OR.ABS(ERROR).LT..05)IR=1
(1259) IF(I.GT.2.AND.ABS(ERROR).GT.ABS(ER))IR=1
(1260) ER=ERROR
(1261) IF(ABS((ST1-SA)/D2S-LBUB).LT.5.)GOTO 100
(1262) GOTO 1026
(1263) 100 J2=J+1
(1264) WRITE(1,200)ST1,J,J2,LBUB
(1265) 200 FORMAT(//,25X,'Bubble has Burst'//,
(1266) +25X,'AT S=',F8.5,2X,'Between node ',I3,' and node ',I3/,
(1267) +25X,'Length of Bubble ='F11.6//)
(1268) WRITE(1,202)
(1269) 202 FORMAT(//,'YOU MAY CONTINUE THE ANALYSIS OR STOP IT',/
(1270) + 'ENTER 1 TO CONT. 0 TO STOP',//)
(1271) READ(1,*)IA
(1272) IF(IA.LE.0)STOP
(1273) IR=2
(1274) 1026 RETURN
(1275) END
(1276) C
(1277) C
(1278) C ***** S U B R O U T I N E 13 *****
(1279) C
(1280) C
(1281) SUBROUTINE LAMBL1(U,DU1,S,DEL1,DEL2,K1,J,RE,CF9)
(1282) COMMON/BLOCK3/RE1,ISTAG(10)
(1283) COMMON/BL20/ITRACE
(1284) REAL M,M1,LAM
(1285) DIMENSION U(K1),DU1(K1),S(K1),DEL1(K1),CF9(K1),
(1286) 1DEL2(K1),HM(35),M(35)

```

```

) C
) DATA M/.082,.0818,.0816,.0812,.0808,.0804,.08,.079,.078,.076,.074
) +.072,.07,.068,.064,.06,.056,.052,.048,.04,.032,.024,.016,.008,0.,
) +-.016,-.032,-.048,-.064,-.08,-.1,-.12,-.14,-.2,-.25/
) DATA HM/3.7,3.69,3.66,3.63,3.61,3.59,3.58,3.52,3.47,3.38,3.3,
) +3.23,3.17,3.13,3.05,2.99,2.94,2.9,2.87,2.81,2.75,2.71,2.67,2.64
) +,2.61,2.55,2.49,2.44,2.39,2.34,2.28,2.23,2.18,2.07,2./

```

```

) C
) IF(ITRACE.EQ.1)WRITE(6,9999)
) 9999 FORMAT(/,'SUBROUTINE LAMBL1'//)
) RE=RE1
) J2=0
) J3=0
) J4=0
) SUM=0.0
) DEL2(1)=SQRT(.0854/(RE*DU1(1)))
) DEL1(1)=2.31*DEL2(1)
) SUMJ=0.0
) DO 12 J=2,K1
) 100 SUMJ=(U(J)**5+U(J-1)**5)*.5*(S(J)-S(J-1))
) DEL2SQ=.45*(SUM+SUMJ)/(U(J)**6*RE)
) DEL2(J)=SQRT(DEL2SQ)
) M1=-DEL2SQ*RE*DU1(J)
) CALL INPOL(M,HM,35,M1,H1)
) 135 DEL1(J)=H1*DEL2(J)
) SUM=SUM+SUMJ
) REDEL1=DEL1(J)*RE*U(J)
) RES=S(J)*U(J)*RE
) REDEL2=DEL2(J)*U(J)*RE
) C WRITE(1,202)J,H1,DEL1(J),M1
) LAM=-M1
) CALL TRANS(RES,REDEL1,REDEL2,LAM,J,J2,J3,J4)
) C WRITE(1,204)REDEL1,REDEL2,RES
) IF(J.GT.INT(.8*K1))GOTO 130
) IF(J2.NE.0.AND.J3.NE.0.AND.J4.NE.0)GOTO 130
) 115 IF(M1.GT.0.082.AND.J2.EQ.0.AND.J3.EQ.0.AND.J4.EQ.0)GOTO 125
) IF(M1.GT..082.AND.(J2.NE.0.OR.J3.NE.0.OR.J4.NE.0))GOTO 120
) 12 CONTINUE
) 120 WRITE(1,206)
) READ(1,*)J,M10
) IF(M10.EQ.0)STOP
) GOTO 130
) 125 WRITE(1,208)
) H1=3.7
) DEL1(J)=H1*DEL2(J)
) WRITE(1,202)J,H1,DEL1(J),M1
) J10=J-1
) CALL BUBBLE(S,U,DEL2,DEL1,J10,K1,RE,IR)
) J=IR
) 202 FORMAT(/,' J=' ,I3,2X,' H =' ,F9.6,4X,' DEL1=' ,F9.6,4X,' M1 =' ,F9.6)
) 204 FORMAT(' REDEL1 =' ,F10.4,4X,' REDEL2 =' ,F10.4,4X,' RES =' ,E14.4)
) 206 FORMAT(/,' T R A N S I T I O N   N O D E   N O T   D E F I N E D
) 1,/,1H , 'ENTER   TRANSITION NODE #,1   TO CONTINUE'/,
) 210X,'OR   0,0   TO STOP'.)
) 208 FORMAT(/,' LAMINAR SEPARATION   ',//
) 11H , 'VALUES OF H,DELTA1 AND M AT SEPARATION,FOLLOW',//)
) 130 RETURN
) END

```

```

) C
) C
) C ***** S U B R O U T I N E   14 *****
) C

```

```
) C SUBROUTINE TURBL2(S,U,DEL1,DEL2,DU1,J1,H,RE,K1,CF9)
```

```
) C The routine calculates the turbulent boundary layer  
) C parameters using the method of Head & Patel  
) C
```

```
) COMMON/BLOCK3/RE1,ISTAG(10)
```

```
) COMMON/BL20/ITRACE
```

```
) DIMENSION S(K1),U(K1),DEL1(K1),DEL2(K1),DU1(K1),HSTAR(27)
```

```
) +,RED2(23),H1(27,23),H(99),WORK(27),WORK1(27),CF9(99)  
) DATA HSTAR/3.6,3.65,3.7,3.75,3.8,3.85,3.9,3.95,4.,4.1,4.2,4.3,4.4  
) +4.6,4.8,5.,5.5,6.,6.5,7.,8.,9.,10.,11.,12.,13.,14./  
) DATA RED2/5.4,5.2,5.,4.8,4.6,4.4,4.2,4.,3.9,3.8,3.7,3.6,3.5,  
) +3.4,3.3,3.2,3.1,3.,2.9,2.8,2.7,2.6,2.5/  
) DATA H1/2.64,2.33,2.23,2.14,2.07,2.015,1.97,1.93,1.89,1.83,1.785,  
) +1.745,1.705,1.645,1.59,1.54,1.445,1.395,1.355,1.32,1.285,1.255,  
) +1.23,1.215,1.195,1.18,1.17,  
) +2.764,2.365,2.245,2.15,2.08,2.025,1.975,1.935,1.895,1.835,  
) +1.79,1.75,1.71,1.65,1.59,1.54,1.445,1.395,1.355,1.32,1.285,1.255,  
) +1.23,1.215,1.195,1.18,1.17,  
) +0,2.405,2.27,2.17,2.095,2.035,1.98,1.845,1.905,1.835,1.795,  
) +1.76,1.715,1.65,1.595,1.54,1.45,1.395,1.355,1.325,1.29,1.26,1.23,  
) +1.215,1.195,1.185,1.17,  
) +0.,2.455,2.305,2.19,2.12,2.05,1.995,1.955,1.92,1.84,1.805,  
) +1.765,1.72,1.655,1.595,1.545,1.45,1.4,1.355,1.325,1.295,1.26,1.23  
) +,1.215,1.2,1.185,1.17,  
) +0,2.535,2.35,2.225,2.145,2.075,2.015,1.975,1.935,1.85,1.815,1.775  
) +,1.73,1.66,1.6,1.55,1.45,1.4,1.36,1.33,1.3,1.26,1.24,1.22,1.205  
) +,1.195,1.17,0.,2.665,2.415,2.27,2.18,2.11,2.04,1.995,1.96,1.865  
) +,1.83,1.785,1.745,1.67,1.605,1.555,1.46,1.4,1.365,1.335,1.305,  
) +1.27,1.245,1.225,1.205,1.185,1.17,0.,0.,2.49,2.33,2.23,2.145,  
) +2.075,2.025,1.985,1.885,1.85,1.8,1.76,1.68,1.61,1.565,1.465,1.41,  
) +1.37,1.345,1.31,1.28,1.25,1.23,1.21,1.185,1.17,0.,0.,2.605,2.405,  
) +2.29,2.2,2.12,2.07,2.025,1.91,1.87,1.82,1.775,1.7,1.62,1.58,1.48,  
) +1.42,1.385,1.36,1.325,1.29,1.265,1.24,1.215,1.185,1.17,0.,0.,2.85  
) +2.45,2.33,2.235,2.15,2.09,2.045,1.94,1.875,1.835,1.79,1.705,1.63,  
) +1.585,1.485,1.435,1.395,1.37,1.33,1.295,1.27,1.245,1.22,1.19,1.17  
) +3*0.,2.51,2.37,2.265,2.185,2.115,2.06,1.955,1.9,1.845,1.8,1.72,  
) +1.635,1.6,1.5,1.44,1.405,1.375,1.34,1.305,1.28,1.25,1.225,1.19,  
) +1.175,3*0.,2.59,2.425,2.315,2.215,2.145,2.085,1.975,1.915,1.86,  
) +1.815,1.735,1.65,1.61,1.51,1.455,1.415,1.385,1.35,1.315,1.285,1.2  
) +,1.23,1.195,1.18,3*0.,2.68,2.48,2.36,2.25,2.18,2.105,1.995,1.935,  
) +1.875,1.83,1.745,1.66,1.62,1.525,1.465,1.425,1.4,1.365,1.325,1.29  
) +,1.27,1.235,1.205,1.185,3*0.,2.835,2.565,2.415,2.29,2.21,2.135,  
) +2.015,1.955,1.89,1.85,1.765,1.67,1.635,1.54,1.48,1.44,1.415,1.38,  
) +1.34,1.31,1.28,1.245,1.21,1.19,4*0.,2.665,2.48,2.34,2.25,2.165,  
) +2.035,1.975,1.915,1.865,1.785,1.7,1.655,1.56,1.5,1.455,1.43,1.395  
) +1.35,1.32,1.29,1.255,1.22,1.205,4*0.,2.795,2.555,2.395,2.29,2.195  
) +2.06,2.,1.935,1.89,1.805,1.725,1.675,1.575,1.52,1.475,1.445,1.415  
) +1.365,1.33,1.305,1.27,1.235,1.215,5*0.,2.635,2.46,2.33,2.235,2.08  
) +,2.025,1.955,1.915,1.83,1.75,1.7,1.6,1.54,1.495,1.47,1.435,1.385,  
) +1.345,1.32,1.285,1.25,1.23,5*0.,2.795,2.545,2.38,2.27,2.115,  
) +2.055,1.985,1.94,1.655,1.78,1.725,1.625,1.565,1.525,1.49,  
) +1.455,1.405,1.365,1.335,1.305,1.27,1.245,6*0.,2.64,2.45,2.32,2.18  
) +2.09,2.02,1.965,1.88,1.81,1.765,1.65,1.59,1.555,1.525,1.48,  
) +1.435,1.395,1.355,1.33,1.295,1.27,6*0.,2.8,2.515,2.375,2.23,2.125  
) +2.05,2.005,1.915,1.845,1.795,1.675,1.625,1.585,1.555,1.505,1.46,  
) +1.42,1.38,1.355,1.325,1.29,7*0.,2.62,2.45,2.28,2.17,2.09,2.04,1.9  
) +,1.88,1.825,1.715,1.655,1.62,1.59,1.54,1.495,1.45,1.415,1.385,  
) +1.355,1.32,7*0.,2.8,2.54,2.33,2.22,2.135,2.085,1.985,1.91,1.865,  
) +1.75,1.695,1.655,1.625,1.58,1.535,1.485,1.445,1.415,1.39,1.35,  
) +8*0.,2.655,2.4,2.275,2.195,2.13,2.03,1.96,1.905,1.795,1.735,1.7,
```

```

411) +1.67,1.625,1.58,1.535,1.485,1.455,1.425,1.39,9*0.,2.5,2.34,2.2
412) +2.185,2.085,2.01,1.96,1.885,1.84,1.75,1.725,1.675,
413) +1.635,1.59,1.545,1.505,1.47,1.43/
414) C
415) IF(ITRACE.EQ.1)WRITE(6,9999)
416) 9999 FORMAT(/,'SUBROUTINE TURBL2'//)
417) RD2=DEL2(J1)*U(J1)*RE
418) F1=ALOG10(RD2)
419) CALL INPOL1(RED2,F1,23,J2)
420) J3=J2-1
421) DIF=RED2(J2)-RED2(J3)
422) DIF1=F1-RED2(J3)
423) R1=DIF1/DIF
424) DO 2 I=1,27
425) WORK(I)=H1(I,J3)+R1*(H1(I,J2)-H1(I,J3))
426) IF(H1(I,J3).EQ.0..OR.H1(I,J2).EQ.0.)WORK(I)=0.
427) 2 CONTINUE
428) H2=H(J1)
429) CALL INPOL2(WORK,HSTAR,27,H2,HST)
430) HST=HST
431) M2=K1-1
432) DO 4 J=J1,M2
433) SD2=DEL2(J)
434) SU=U(J)
435) SDU=DU1(J)
436) SH=H(J)
437) DO 6 I=1,2
438) CALL HEPAS(SD2,RE,SH,SU,SDU,HST,DHST,DUD2)
439) DS=(S(J+1)-S(J))*0.5
440) IF(I.EQ.2)DS=DS*2.
441) HST=HST+DHST*DS
442) B1=U(J)*DEL2(J)+DUD2*DS
443) SU=(U(J)+U(J+1))*0.5
444) IF(I.EQ.2)SU=U(J+1)
445) SD2=B1/SU
446) RSD2=SU*RE*SD2
447) F2=ALOG10(RSD2)
448) CALL INPOL1(RED2,F2,23,J2)
449) J3=J2-1
450) DO 8 I2=1,27
451) WORK(I2)=H1(I2,J2)
452) 8 WORK1(I2)=H1(I2,J3)
453) CALL INPOL2(HSTAR,WORK,27,HST,H2)
454) CALL INPOL2(HSTAR,WORK1,27,HST,H3)
455) DIF=RED2(J2)-RED2(J3)
456) DIF1=F2-RED2(J3)
457) R1=DIF1/DIF
458) SH=H3+R1*(H2-H3)
459) SDU=(DU1(J)+DU1(J+1))*0.5
460) 6 CONTINUE
461) IF(SH.GT.1.9)WRITE(1,202)J
462) IF(SH.GT.1.9.OR.SH.LT.1.)SH=1.9
463) H(J+1)=SH
464) DEL2(J+1)=SD2
465) DEL1(J+1)=H(J+1)*DEL2(J+1)
466) 4 CONTINUE
467) 100 WRITE(1,200)(S(I),U(I),DU1(I),H(I),DEL2(I),I=1,K1)
468) 200 FORMAT(5F14.5)
469) 202 FORMAT(/,'TURBULENT SEPARATION AT NODE ',I3,/)
470) RETURN
471) END
472) C

```

```

(1473) C
(1474) C ***** S U B R O U T I N E 15 *****
(1475) C
(1476) C
(1477) SUBROUTINE HEPA(SDEL2,RE,SH,SU,SDU,HST,DHST,DUD2)
(1478) RED2=RE*SU*SDEL2
(1479) C=ALOG(RED2)
(1480) A=.019521-.386768*C+.028345*C**2-.000701*C**3
(1481) B=.191511-.834891*C+.062588*C**2-.001953*C**3
(1482) CF=EXP(A*SH+B)
(1483) D10=CF*.5-(SH+1)*SDEL2*SDU/SU
(1484) G=(SH-1)*SQRT(2./CF)/SH
(1485) PI=((G+1.7)/6.1)**2-1.81
(1486) D20=CF*.5*(1+(SH+1)/SH*PI)
(1487) R2=D10/D20
(1488) IF(R2.GE.1)FR2=1./(2.*R2-1.)
(1489) IF(R2.LT.1.)FR2=(5.-4.*R2)/(3.-2.*R2)
(1490) DHST=HST*D20*(FR2-R2)/SDEL2
(1491) DUD2=D10*SU
(1492) 2 RETURN
(1493) END
(1494) C
(1495) C
(1496) C ***** S U B R O U T I N E 16 *****
(1497) C
(1498) C
(1499) SUBROUTINE INPOL1(X,X1,N,I)
(1500) DIMENSION X(N)
(1501) DO 1 I=1,N
(1502) 1 IF (X(I).LE.X1)RETURN
(1503) END
(1504) C
(1505) C
(1506) C ***** S U B R O U T I N E 17 *****
(1507) C
(1508) C
(1509) SUBROUTINE INPOL2(X,Y,N,LAM,Y1)
(1510) DIMENSION X(N),Y(N)
(1511) REAL LAM
(1512) IF(X(1).GT.X(2).OR.X(1).EQ.0.)GOTO 8
(1513) DO 2 I=1,N
(1514) IF(X(I).GE.LAM)GOTO 4
(1515) 2 CONTINUE
(1516) 8 DO 1 I=1,N
(1517) IF(X(I).LT.LAM.AND.X(I).NE.0..AND.X(I+1).NE.0.)GOTO 4
(1518) IF (X(I).EQ.LAM)GOTO 5
(1519) 1 CONTINUE
(1520) 4 I1=I-1
(1521) X0=X(I)
(1522) X2=X(I1)
(1523) Y1=Y(I)
(1524) Y3=Y(I1)
(1525) A=(Y1-Y3)/(X0-X2)
(1526) B=(X0*Y3-X2*Y1)/(X0-X2)
(1527) Y1=A*LAM+B
(1528) 5 IF (X(I).EQ.LAM)Y1=Y(I)
(1529) 6 RETURN
(1530) END
(1531) C
(1532) C
(1533) C ***** S U B R O U T I N E 18 *****
(1534) C

```



```

(1535) C
(1536) SUBROUTINE DRIVE1(ITER,NT,KN)
(1537) C
(1538) C *****
(1539) C * The subroutine calculates the tangential velocity on *
(1540) C * the surface of each aerofoil element. This velocity *
(1541) C * is given by: *
(1542) C * *
(1543) C * Vtan= Vinv + Vvis + Vwake1 + Vwake2 *
(1544) C * *
(1545) C * For single element aerofoils Vwake2 is zero. *
(1546) C *****
(1547) C
(1548) COMMON/BL4/ALPHA2, KK, NS1, NS2, PI
(1549) COMMON/BL5/X(99), Z(99), S(99), SA(99), CN1(99), SN1(99),
(1550) +T1(99), T3(99)
(1551) COMMON/BL6/XW1(99), ZW1(99), XW2(99), ZW2(99), SW1(99), SW2(99)
(1552) COMMON/BL7/XBW1(99), ZBW1(99), XBW2(99), ZBW2(99)
(1553) COMMON/BL10/GAMV1(99), GAMV2(99), GAMS1(99), GAMS2(99)
(1554) COMMON/BL11/GAMV11(99), GAMV22(99), GAMS11(99), GAMS22(99)
(1555) COMMON/BL12/SNW1(99), SNW2(99), CNW1(99), CNW2(99)
(1556) COMMON/BL13/TW11(99), TW13(99), TW21(99), TW23(99)
(1557) COMMON/BL14/GNV1(99,99), GNV2(99,99), GTV1(99,99), GTV2(99,99)
(1558) COMMON/BL15/GAMA(99), VTOT(99)
(1559) COMMON/BL17/XBAR(99), ZBAR(99)
(1560) COMMON/BL18/CP(99), CF(99)
(1561) COMMON/BL20/ITRACE
(1562) COMMON/BL23/IPAR
(1563) COMMON/BL24/RES4(99), MAN1, MAN
(1564) DIMENSION RES(99), RES1(99), DUDS(99), GN(99,99)
(1565) +, RES2(99), RES3(99), VW1(99), VW2(99), GT(99,99)
(1566) +, KN(10), VBL(99), CAP(10), CNP(10), CMP(10), CAF(10), CNF(10)
(1567) +, CMF(10), IPIV(99)
(1568) IF(ITRACE.EQ.1)WRITE(6,9999)
(1569) 9999 FORMAT(/, 'SUBROUTINE DRIVE1' /)
(1570) PI=3.1415926536
(1571) IF(ITER.GT.1)GOTO 1000
(1572) CALL AIRIC(KK,NT,KN,GN,GT,XBAR,ZBAR,T1,T3,XBAR,ZBAR,T1,T3,
(1573) C CALL MATINV(GN,KN)
(1574) CALL GAUSS(GN,IPIV,KN)
(1575) C CALL MATMUL(GN,SN1,RES,KN,KN)
(1576) CALL SOLVE(GN,SN1,RES,IPIV,KN)
(1577) C***** RES=[GN-1]*SIN(PHIa-ALPHA) F1
(1578) DO 10 I=1,KN
(1579) SN1(I)=RES(I)
(1580) RES4(I)=2.*PI*RES(I)
(1581) 10 CONTINUE
(1582) C **** SN1=[GN-1]*SIN(PHIa-ALPHA) F2
(1583) CALL MATMUL(GT,SN1,RES,KN,KN)
(1584) C **** RES=[GT]*F2 F3
(1585) DO 20 I=1,KN
(1586) CN1(I)=CN1(I)+RES(I)
(1587) 20 CONTINUE
(1588) DO 22 I=1,KN
(1589) VTOT(I)=CN1(I)
(1590) CP(I)=1-VTOT(I)*VTOT(I)
(1591) WRITE(6,200)I,XBAR(I),CP(I),VTOT(I)
(1592) 200 FORMAT(I3,9X,'XBAR=',F12.7,9X,'CP=',F12.7,9X,'VTOT=',F12.7
(1593) 22 CONTINUE
(1594) C **** CN1=COB(PHIa-ALPHA)+F3=Vinviscid F4
(1595) C **** The above calculation need only be performed once
(1596) CALL CNXCLD(CAP,CNP,CMP,CAF,CNF,CMF,CAT,CNT,CMT,CL,KN,

```

```

) +NT)
) 1000 CONTINUE
) IF(ITER.EQ.1)WRITE(1,2000)
) 2000 FORMAT(10X,' IS VISCOUS ANALYSIS REQUIRED ? ',//)
) IF(MAN.LE.0.AND.ITER.EQ.1)READ(1,*)MAN
) IF(MAN.LE.0)GOTO 150
) DO 24 I=1,KK
) CP(I)=1-VTOT(I)*VTOT(I)
) 24 CONTINUE
) CALL BLCALC(CP,VTOT,SA,KN,KK,NT,IPAR,DUDS)
) C CALL MATMUL(GN,DUDS,RES,KK,KK)
) CALL SOLVE(GN,DUDS,RES,IPIV,KK)
) DO 30 I=1,KK
) DUDS(I)=RES(I)
) 30 CONTINUE
) C ****DUDS=[GN-1]*[D(U*DELTASTAR)/DS]aerofoil F5
) CALL MATMUL(GT,DUDS,VBL,KK,KK)
) C **** VBL=GT*F5 Viscous velocity component F6
) C **** The value of VBL is updated during each iteration
) DO 32 I=1,KK
) VTOT(I)=CN1(I)+VBL(I)
) CP(I)=1-VTOT(I)*VTOT(I)
) 32 CONTINUE
) CALL CNXCLD(CAP,CNP,CMP,CAF,CNF,CMF,CAT,CNT,CMT,CL,KN,
) +NT)
) IF(ITER.EQ.1)WRITE(1,2002)
) 2002 FORMAT(//,' IS WAKE ANALYSIS REQUIRED ? ',//)
) IF(MAN1.LE.0)READ(1,*)MAN1
) IF(MAN1.LE.0)GOTO 150
) DO 40 I=1,KK
) GAMA(I)=2.*PI*(SN1(I)+DUDS(I))
) 40 CONTINUE
) C **** GAMA=2*PI*(F2+F5) F7
) C **** This is the final value of GAMA for the first iteration
) C
) C **** During the first iteration all wake singularities are zero
) C
) IF(ITER.EQ.1)GOTO 150
) NN=KK
) NN1=1
) NN2=NS1
) CALL WAKEIC(NN,NN1,NN2,GNV1,GTV1,XBAR,ZBAR,T1,T3,XBW1,ZBW1,TW11,
) +TW13,SW1)
) C **** GNV1 GTV1 are [GNv] w1->a and [GTV] w1-->a
) NN=KK
) NN1=1
) NN2=NS2
) IF(NT.GT.2)CALL WAKEIC(NN,NN1,NN2,GNV2,GTV2,XBAR,ZBAR,T1,T3,
) +XBW2,ZBW2,TW21,TW23,SW2)
) C **** GNV2 and GTV2 are [GNv] and [GTV] w2-->a
) CALL MATMUL(GNV1,GAMV11,RES,KK,NS1)
) C **** RES=[GNv]*GAMAv wake1 F
) C CALL MATMUL(GN,RES,RES1,KK,KK)
) CALL SOLVE(GN,RES,RES1,IPIV,KK)
) C **** RES=[GN-1] * [GNv] * GAMAv wake1 F
) DO 50 I=1,KK
) GAMA(I)=GAMA(I)-RES1(I)
) 50 CONTINUE
) C **** GAMA=F7-F9
) CALL MATMUL(GTV1,GAMS11,RES,KK,NS1)
) C **** RES=[GTV]*GAMAs wake1 F
) C CALL MATMUL(GN,RES,RES2,KK,KK)

```

```

CALL SOLVE(GN,RES,RES2,IPIV,KK)
C **** RES2=[GN-1] * [GTV] * GAMAs      wake1      F13
DO 60 I=1,KK
GAMA(I)=GAMA(I)-RES2(I)
60 CONTINUE
C **** GAMA=F7-F9-F13
C
C **** CALCULATION OF VW1 WAKE1--> AEROFOIL
C
CALL MATMUL(GTV1,GAMV11,RES,KK,NS1)
C **** RES=[GTV] * GAMAv      wake1      F13
CALL MATMUL(GT,RES1,RES3,KK,KK)
C **** RES3=GT * GN-1 * GNV * GAMAv      wake1      F13
DO 70 I=1,KK
RES1(I)=RES(I)-RES3(I)
70 CONTINUE
C **** RES1=B1 * GAMAv      wake1      F13
CALL MATMUL(GNV1,GAMS11,RES,KK,NS1)
C **** RES=[GNV] * GAMAs      wake1      F13
CALL MATMUL(GT,RES2,RES3,KK,KK)
C **** RES3=GT * F13      wake1      F13
DO 80 I=1,KK
RES2(I)=RES(I)+RES3(I)
80 CONTINUE
C **** RES2=C1 * GAMAs      wake1      F20
DO 90 I=1,KK
VW1(I)=(RES1(I)-RES2(I))/2./PI
90 CONTINUE
C **** CALCULATION OF VW1 HAS BEN COMPLETED
C
IF(NT.EQ.2)GOTO 150
C For a single element aerofoil there is only one wake
C
C **** GAMA CALCULATION CONTINUED
C
CALL MATMUL(GNV2,GAMV22,RES,KK,NS2)
C CALL MATMUL(GN,RES,RES1,KK,KK)
CALL SOLVE(GN,RES,RES1,IPIV,KK)
C **** RES1=[GN-1] * [GNV] * GAMAv      wake2      F20
DO 100 I=1,KK
GAMA(I)=GAMA(I)-RES1(I)
100 CONTINUE
C **** GAMA=F7-F9-F13-F22
CALL MATMUL(GTV2,GAMS22,RES,KK,NS2)
C CALL MATMUL(GN,RES,RES2,KK,KK)
CALL SOLVE(GN,RES,RES2,IPIV,KK)
C **** RES2=[GN-1] * [GTV] * GAMAs      wake2
DO 110 I=1,KK
GAMA(I)=GAMA(I)-RES2(I)
110 CONTINUE
C
C **** GAMA calculation completed
C
C **** EVALUATION OF VW2
C
CALL MATMUL(GTV2,GAMV22,RES,KK,NS2)
CALL MATMUL(GT,RES1,RES3,KK,KK)
DO 120 I=1,KK
RES3(I)=RES(I)-RES3(I)
120 CONTINUE
C **** RES3=B2 * GAMAv      WAKE2
CALL MATMUL(GNV2,GAMS22,RES,KK,NS2)

```

```

21) CALL MATMUL(GT,RES2,RES1,KK,KK)
22) DO 130 I=1,KK
23) RES(I)=RES(I)+RES1(I)
24) 130 CONTINUE
25) C **** RES=C2 *GAMAs WAKE2
26) DO 140 I=1,KK
27) VW2(I)=(RES3(I)-RES(I))/2./PI.
28) 140 CONTINUE
29) C
30) C **** VW2 HAS BEEN EVALUATED
31) C
32) 150 CONTINUE
33) WRITE(6,9976)
34) 9976 FORMAT(///,13X,' I ',4X,' XBAR',6X,' Vinv',6X,' Vvis',7X,' VW1',7X,' V
35) +,8X,' Vtot',7X,' CP',5X,' GAMA',//)
36) DO 160 I=1,KK
37) VTOT(I)=CN1(I)+VBL(I)+VW1(I)+VW2(I)
38) CP(I)=1-VTOT(I)*VTOT(I)
39) WRITE(6,9978) I,XBAR(I),CN1(I),VBL(I),VW1(I),VW2(I),VTOT(I),CP(I
40) +GAMA(I)
41) 9978 FORMAT(12X,I2,2X,F8.5,2X,F8.5,2X,F8.5,3X,F8.5,2X,F8.5,2X,F8.5,2
42) +F8.5,2X,F8.5)
43) 160 CONTINUE
44) CALL CNXCLD(CAP,CNP,CMP,CAF,CNF,CMF,CAT,CNT,CMT,CL,KN,
45) +NT)
46) RETURN
47) END
48) C
49) C
50) C ***** S U B R O U T I N E 19 *****
51) C
52) C
53) SUBROUTINE DRIVE2(ITER,NT)
54) C
55) C *****
56) C * The routine evaluates the normal and tangential components *
57) C * VN VT of the induced velocity at the mid point of each *
58) C * wake segment. Vn and VT are then used to: *
59) C * * * * *
60) C * 1. Relax the wake *
61) C * 2. Calculate VM=SQRT(VT*VT+VN*VN) *
62) C * 3. Calculate Vu=VM+GAMAV/2 *
63) C * V1=VM-GAMAV/2 *
64) C * * * * *
65) C *****
66) C
67) COMMON/BL4/ALPHA2,KK,NS1,NS2,PI
68) COMMON/BL5/X(99),Z(99),S(99),SA(99),CN1(99),SN1(99),
69) +T1(99),T3(99)
70) COMMON/BL6/XW1(99),ZW1(99),XW2(99),ZW2(99),SW1(99),SW2(99)
71) COMMON/BL7/XBW1(99),ZBW1(99),XBW2(99),ZBW2(99)
72) COMMON/BL8/VN1(99),VN2(99),VT1(99),VT2(99)
73) COMMON/BL10/GAMV1(99),GAMV2(99),GAMS1(99),GAMS2(99)
74) COMMON/BL11/GAMV11(99),GAMV22(99),GAMS11(99),GAMS22(99)
75) COMMON/BL12/SNW1(99),SNW2(99),CNW1(99),CNW2(99)
76) COMMON/BL13/TW11(99),TW13(99),TW21(99),TW23(99)
77) COMMON/BL15/GAMA(99),VTOT(99)
78) COMMON/BL17/XBAR(99),ZBAR(99)
79) COMMON/BL19/KN(10)
80) COMMON/BL20/ITRACE
81) DIMENSION G1(99,99),G2(99,99),GAM1(99),GAM2(99),GS1(99),
82) +GS2(99),VN(99),VT(99),RES(99),RES1(99),RES2(99),RES3(99)

```

```

3) C IF(ITRACE.EQ.1)WRITE(6,9999)
1) 9999 FORMAT(/,'SUBROUTINE DRIVE2'//)
5) ALPHA=ALPHA2
5) IF(ITER.EQ.1)GOTO 112
7) DO 16 I=1,NS1
3) GAM1(I)=GAMV11(I)
7) GS1(I)=GAMS11(I)
) 16 CONTINUE
1) DO 18 I=1,NS2
2) GAM2(I)=GAMV22(I)
3) GS2(I)=GAMS22(I)
4) 18 CONTINUE
5) 112 CONTINUE
5) MFLAG=0
7) 102 CONTINUE
3) IF(MFLAG.EQ.0)CALL AIRIC(NS1,NT,KN,G1,G2,XBW1,ZBW1,TW11,TW13,XBAI
7) +,ZBAR,T1,T3,S)
) IF(MFLAG.EQ.1)CALL AIRIC(NS1,NT,KN,G1,G2,XBW2,ZBW2,TW21,TW23,XBAI
1) +,ZBAR,T1,T3,S)
2) C
3) CALL MATMUL(G1,GAMA,RES,NS1,KN)
1) CALL MATMUL(G2,GAMA,RES1,NS1,KN)
3) DO 2 I=1,NS1
5) VN(I)=RES(I)✓
7) VT(I)=RES1(I)✓
3) 2 CONTINUE
7) IF(ITER.EQ.1)GOTO 108
) IF(MFLAG.EQ.0)CALL WAKEIC(NS1,1,NS1,G1,G2,XBW1,ZBW1,TW11,TW13,XB
1) +,ZBW1,TW11,TW13,SW1)
2) IF(MFLAG.EQ.1)CALL WAKEIC(NS1,1,NS1,G1,G2,XBW2,ZBW2,TW21,TW23,XB
5) +,ZBW2,TW21,TW23,SW2)
1) CALL MATMUL(G1,GAM1,RES,NS1,NS1)-
5) CALL MATMUL(G2,GAM1,RES1,NS1,NS1)-
) CALL MATMUL(G2,GS1,RES2,NS1,NS1)-
7) CALL MATMUL(G1,GS1,RES3,NS1,NS1)
3) DO 4 I=1,NS1
) VN(I)=VN(I)+RES(I)+RES2(I)
) VT(I)=VT(I)+RES1(I)+RES3(I)
) 4 CONTINUE
) IF(NT.EQ.2)GOTO 116
) IF(MFLAG.EQ.0)CALL WAKEIC(NS1,1,NS2,G1,G2,XBW1,ZBW1,TW11,TW13,XB
1) +,ZBW2,TW21,TW23,SW2)
) IF(MFLAG.EQ.1)CALL WAKEIC(NS1,1,NS2,G1,G2,XBW2,ZBW2,TW21,TW23,XB
) +,ZBW1,TW11,TW13,SW1)
7) CALL MATMUL(G1,GAM2,RES,NS1,NS2)
) CALL MATMUL(G2,GAM2,RES1,NS1,NS2)
) CALL MATMUL(G2,GS2,RES2,NS1,NS2)
) CALL MATMUL(G1,GS1,RES3,NS1,NS2)
) 116 CONTINUE
) DO 6 I=1,NS1
) VN(I)=VN(I)+RES(I)+RES2(I)
) VT(I)=VT(I)+RES1(I)+RES3(I)
) 6 CONTINUE
) 108 CONTINUE
) IF(MFLAG.EQ.1)GOTO 106
) DO 12 I=1,NS1
) VN1(I)=VN(I)/2./PI-SNW1(I)✓
) VT1(I)=VT(I)/2./PI+CNW1(I)✓
) 12 CONTINUE
) IF(NT.EQ.2)RETURN
) MFLAG=1
) IF(ITER.EQ.1)GOTO 110

```

```

(45) DO 14 I=1,NS1
(46) GAM2(I)=GAMV11(I)
(47) GS2(I)=GAMS11(I)
(48) 14 CONTINUE
(49) DO 20 I=1,NS2
(50) GAM1(I)=GAMV22(I)
(51) GS1(I)=GAMS22(I)
(52) 20 CONTINUE
(53) 110 CONTINUE
(54) NN=NS1
(55) NS1=NS2
(56) NS2=NN
(57) GOTO 102
(58) 106 CONTINUE
(59) DO 22 I=1,NS1
(60) VN2(I)=VN(I)/2./PI-SNW2(I)
(61) VT2(I)=VT(I)/2./PI+CNW2(I)
(62) 22 CONTINUE
(63) NN=NS1
(64) NS1=NS2
(65) NS2=NN
(66) RETURN
(67) END
(68) C
(69) C
(70) C ***** S U B R O U T I N E 20 *****
(71) C
(72) C
(73) SUBROUTINE AIRIC(KK,NT,KN,GN,GT,XB,ZB,T11,T13,XBAR,ZBAR,T1,T3,S
(74) COMMON/BL20/ITRACE
(75) REAL L
(76) INTEGER SUM,SUM1
(77) C
(78) C *****
(79) C *
(80) C * This subroutine evaluates the influence coefficient of all*
(81) C * aerofoil control points due to: *
(82) C * *
(83) C * 1. Aerofoil singularities *
(84) C * 2. Wake singularities *
(85) C * *
(86) C * The normal components of the influence coefficients *
(87) C * are stored in matrix GN while the tangential components *
(88) C * are stored in matrix GT. *
(89) C * *
(90) C *****
(91) C
(92) DIMENSION KN(10),GXV(99),XB(99),T11(99),T13(99),ZB(99),
(93) +GYV(99),GXS(99),GYS(99),LN(10),GN(99,99),GT(99,99),S(99)
(94) +,XBAR(99),ZBAR(99),T1(99),T3(99)
(95) IF(ITRACE.EQ.1)WRITE(6,9999)
(96) 9999 FORMAT(//,'SUBROUTINE AIRIC'//)
(97) PI=3.1415926536
(98) EPS=.000001
(99) LN(1)=KN(2)
(100) LN(2)=KN(3)-KN(2)
(101) IA=NT-1
(102) WRITE(1,9996)(S(I),I=1,KN)
(103) 9996 FORMAT(F12.7,2X,F12.7,2X,F12.7,2X,F12.7,2X,F12.7)
(104) DO 2 J=1,KN
(105) DO 4 JJ=2,NT
(106) SAVEBY=0.

```

```

07) SAVEBX=0.
08) SAVEAY=0.
09) SAVEAX=0.
10) K3=KN(JJ-1)+1
11) K4=KN(JJ)
12) DO 6 K=K3,K4
13) IF(S(K).EQ.0)WRITE(1,9998)K,S(K)
14) 9998 FORMAT('K=',I3,3X,'DS=',F12.7).
15) A1=(XB(J)-XBAR(K))/S(K)
16) A3=(ZB(J)-ZBAR(K))/S(K)
17) ASQ=A1*A1+A3*A3+.25
18) F=A3*T1(K)-A1*T3(K)
19) E=A3*T3(K)+A1*T1(K)
20) EA=.5-E
21) EB=.5+E
22) FM=ABS(F)
23) IF(FM.LT.EPS)GOTO 114
24) U1=F/FM
25) THETA=ATAN(EA/FM)*U1
26) THETB=ATAN(EB/FM)*U1
27) GOTO 116
28) 114 U1=2.*ABS(EA)
29) U2=2.*ABS(EB)
30) THETA=EA*PI/U1
31) THETB=EB*PI/U2
32) 116 T=THETA+THETB
33) RNUM=ASQ-E
34) IF(RNUM.LT.EPS)GOTO 120
35) RDEN=ASQ+E
36) IF(RDEN.LE.EPS)GOTO 120
37) L=.5*ALOG(RNUM/RDEN)
38) GOTO 130
39) 120 L=0.0
40) 130 CONTINUE
41) F1=F*T1(K)-E*T3(K)
42) E1=E*T1(K)+F*T3(K)
43) BY=L*E1-T*F1+T1(K)
44) BX=T*E1+L*F1-T3(K)
45) AY=.5*(T*T3(K)+L*T1(K))
46) AX=.5*(T*T1(K)-L*T3(K))
47) M1=K
48) IF(JJ.EQ.3)M1=K+1
49) GXV(M1)=AX-BX+SAVEAX+SAVEBX
50) GYV(M1)=AY-BY+SAVEAY+SAVEBY
51) GX(K)=-2.*AY
52) GY(K)=2.*AX
53) SAVEBY=BY
54) SAVEBX=BX
55) SAVEAY=AY
56) SAVEAX=AX
57) C IF(J.EQ.1.OR.J.EQ.40.OR.J.EQ.80.)WRITE(1,200)J,JJ,K3,K4,K,M1,IA,
58) IF(ABS(GXV(K)).GT.10..OR.ABS(GYV(K)).GT.10.)WRITE(1,9990) J,K,FM
59) 9990 FORMAT(10X,I3,5X,I3,5X,F12.7)
60) 6 CONTINUE
61) 4 CONTINUE
62) DO 8 L1=1,IA
63) SUM=0
64) SUM1=0
65) IF(L1.EQ.1)GOTO 106
66) IB=L1-1
67) DO 10 L2=1,IB
68) SUM=SUM+LN(L2)

```

```

9)      SUM1=SUM1+2*(LN(L2)+1)
0)  10  CONTINUE
1)      K8=L1+SUM
2)      K9=LN(L1)/2+L1-1+SUM
3)      GOTO 108
4)  106  K8=1
5)      K9=LN(1)/2
6)  108  DO 12 K=K8,K9
7)      IV=K-L1+1
8)      IV1=K+1
9)      IV2=LN(L1)+1-K+SUM1
0)      IS=K-L1+1+LN(L1)/2
1)      IS1=K-L1+1
2)      IS2=LN(L1)+2-L1-K+SUM1
3)      A2=GXV(IV1)+GXV(IV2)
4)      B2=GYV(IV1)+GYV(IV2)
5)      GT(J,IV)=A2*T11(J)+B2*T13(J)
6)      GN(J,IV)=-A2*T13(J)+B2*T11(J)
7)      C1=GXS(IS1)+GXS(IS2)
8)      D1=GYS(IS1)+GYS(IS2)
9)      GT(J,IS)=C1*T11(J)+D1*T13(J)
0)      GN(J,IS)=-C1*T13(J)+D1*T11(J)
1)  C    IF(J.EQ.1.OR.J.EQ.28.OR.J.EQ.54.OR.J.EQ.55.OR.J.EQ.74.OR.J.EQ.92
2)  C    +WRITE(1,200)J,K,IV,IV1,IV2,IS,IS1,IS2
3)  200  FORMAT(8I6)
4)  12  CONTINUE
5)  8   CONTINUE
6)  2   CONTINUE
7)      DO 20 I=1,KK
8)      DO 20 J=1,KK
9)  C    IF(J.EQ.I)WRITE(6,300)I,I,GN(J,I),GT(J,I)
0)  300  FORMAT(I6,I6,5X,F12.7,5X,F12.7)
1)  20  CONTINUE
2)  40  RETURN
3)      END
4)  C
5)  C
6)  C    ***** S U B R O U T I N E    21 *****
7)  C
8)  C
9)      SUBROUTINE WAKEIC(KK,K3,K4,GN,GT,XB1,ZB1,TX1,TX3,XBAR,ZBAR,T1,T3
0)  +)
1)      COMMON/BL20/ITRACE
2)      REAL L
3)      DIMENSION GN(99,99),GT(99,99),XB1(99),ZB1(99),TX1(99),
4)  +TX3(99),XBAR(99),ZBAR(99),T1(99),T3(99),S(99)
5)  C    IF(ITRACE.EQ.1)WRITE(6,9999)
6)  9999  FORMAT('/', 'SUBROUTINE WAKEIC' //)
7)      PI=3.1415926536
8)      EPS=1E-06
9)      DO 2 J=1,KK
0)      SAVEBY=0.
1)      SAVEBX=0.
2)      SAVEAY=0.
3)      SAVEAX=0.
4)      DO 6 K=K3,K4
5)      IF(S(K).EQ.0)WRITE(1,9998)K,S(K)
6)  9998  FORMAT('K=',I3,3X,'DSW=',F12.7)
7)      A1=(XB1(J)-XBAR(K))/S(K)
8)      A3=(ZB1(J)-ZBAR(K))/S(K)
9)      F=A3*T1(K)-A1*T3(K)
0)      E=A3*T3(K)+A1*T1(K)

```



```

(2031) EA=.5-E
(2032) EB=.5+E
(2033) C FM=ABS(F)
(2034) IF(F.GT.0.0) FM=F
(2035) IF(F.LE.0.0) FM=-F
(2036) RNUM=EA*EA+F*F
(2037) RDEN=EB*EB+F*F
(2038) IF(RNUM.LE.EPS.OR.RDEN.LE.EPS)GOTO 110
(2039) L=.5*ALOG(RNUM/RDEN)
(2040) GOTO 112
(2041) 110 L=0.
(2042) 112 CONTINUE
(2043) IF(FM.LT.EPS.AND.F.GE.0.)GOTO 100
(2044) IF(FM.LT.EPS.AND.F.LE.0.)GOTO 102
(2045) T=ATAN(EA/F)+ATAN(EB/F)
(2046) GOTO 104
(2047) 100 IF(E.LT..5.OR.E.GT..5)T=PI
(2048) IF(E.EQ..5.OR.E.EQ..5)T=PI/2.
(2049) IF(E.EQ..5.OR.E.EQ..5)L=0.
(2050) GOTO 104
(2051) 102 IF(E.LT..5.OR.E.GT..5)T=-PI
(2052) IF(E.EQ..5.OR.E.EQ..5)T=-PI/2.
(2053) IF(E.EQ..5.OR.E.EQ..5)L=0.
(2054) 104 IF(FM.LE.EPS.AND.ABS(E).GT..5)T=0.
(2055) F1=F*T1(K)-E*T3(K)
(2056) E1=E*T1(K)+F*T3(K)
(2057) BY=L*E1-T*F1+T1(K)
(2058) BX=T*E1+L*F1-T3(K)
(2059) AY=.5*(T*T3(K)+L*T1(K))
(2060) AX=.5*(T*T1(K)-L*T3(K))
(2061) GXV=AX-BX+SAVEAX+SAVEBX
(2062) GYV=AY-BY+SAVEAY+SAVEBY
(2063) SAVEBY=BY
(2064) SAVEBX=BX
(2065) SAVEAY=AY
(2066) SAVEAX=AX
(2067) GT(J,K)=GXV*TX1(J)+GYV*TX3(J)
(2068) GN(J,K)=-GXV*TX3(J)+GYV*TX1(J)
(2069) 6 CONTINUE
(2070) 2 CONTINUE
(2071) RETURN
(2072) END
(2073) C
(2074) C
(2075) C ***** S U B R O U T I N E 22 *****
(2076) C
(2077) C
(2078) SUBROUTINE W1W2SS(GAMV,GAMS,K1,K2,IFLAG)
(2079) COMMON/BL4/ALPHA2, KK, NS1, NS2, PI
(2080) COMMON/BL6/XW1(99), ZW1(99), XW2(99), ZW2(99), SW1(99), SW2(99)
(2081) COMMON/BL9/VUW1(99), VLW1(99), VUW2(99), VLW2(99)
(2082) COMMON/BL15/GAMA(99), VTOT(99)
(2083) COMMON/BL16/TIN, HIN, TIN1, HIN1, TIN2, HIN2, TIN3, HIN3
(2084) COMMON/BL19/KN(10)
(2085) COMMON/BL20/ITRACE
(2086) DIMENSION DSUW1(99), TUW1(99), SLOPW1(99), DDSUW1(99),
(2087) +DSLW1(99), TLW1(99), DDSLW1(99), VDSUW1(99), VDSSLW1(99)
(2088) +, DSUW2(99), TUW2(99), SLOPW2(99), DDSUW2(99),
(2089) +DSLW2(99), TLW2(99), DDSLW2(99), VDSUW2(99), VDSSLW2(99),
(2090) +GAMV(99), GAMS(99), DUW1(99), DUW2(99), HUW1(99), HUW2(99)
(2091) +, DLW1(99), DLW2(99), HLW1(99), HLW2(99)
(2092) IF(ITRACE.EQ.1)WRITE(6,9999)

```

```

093) 9999 FORMAT(//,'SUBROUTINE W1W2SS'//)
094) IF(IFLAG.EQ.1)GOTO 100
095) K5=KN(2)
096) VINIT=VTOT(K5)
097) CALL WAKVIS(VUW1,VINIT,XW1,ZW1,TIN,HIN,TUW1,HUW1,DUW1,NS1,
098) +VDSUW1,DSUW1,DDSUW1,SLOPW1)
099) WRITE(6,9990)
100) 9990 FORMAT(///,19X,'XW1',6X,'VUW1',6X,'DUW1',5X,'DSUW1',6X,'TUW1'
101) +,6X,'HUW1',5X,'DDSUW1',4X,'VDSUW1',//)
102) WRITE(6,9997)(XW1(I5),VUW1(I5),DUW1(I5),DSUW1(I5),TUW1(I5),
103) +HUW1(I5),DDSUW1(I5),VDSUW1(I5),I5=1,NS1)
104) K6=KN(1)+1
105) VINIT=VTOT(K6)
106) CALL WAKVIS(VLW1,VINIT,XW1,ZW1,TIN1,HIN1,TLW1,HLW1,DLW1,NS1,
107) +VDSLW1,DSLW1,DDSLW1,SLOPW1)
108) WRITE(6,9998)
109) 9998 FORMAT(///,19X,'XW1',6X,'VLW1',6X,'DLW1',5X,'DSLW1',6X,'TLW1'
110) +,6X,'HLW1',5X,'DDSLW1',4X,'VDSLW1',//)
111) WRITE(6,9997)(XW1(I5),VLW1(I5),DLW1(I5),DSLW1(I5),TLW1(I5),
112) +HLW1(I5),DDSLW1(I5),VDSLW1(I5),I5=1,NS1)
113) 9997 FORMAT(17X,F7.5,2X,F8.5,2X,F8.5,2X,F7.6,3X,F8.5,2X,F8.5,
114) +2X,F8.4,2X,F8.4)
115) DO 2 K=K1,K2
116) GAMV(K)=- (VUW1(K)*(DSUW1(K)+TUW1(K))+
117) +VLW1(K)*(DSLW1(K)+TLW1(K)))
118) GAMS(K)=VDSUW1(K)+VDSLW1(K)
119) 2 CONTINUE
120) RETURN
121) 100 CONTINUE
122) K7=KN(3)
123) VIN=VTOT(K7)
124) CALL WAKVIS(VUW2,VIN,XW2,ZW2,TIN2,HIN2,TUW2,HUW2,DUW2,NS2,
125) +VDSUW2,DSUW2,DDSUW2,SLOPW2)
126) WRITE(6,9989)
127) 9989 FORMAT(///,19X,'XW2',6X,'VUW2',6X,'DUW2',5X,'DSUW2',6X,'TUW2'
128) +,6X,'HUW2',5X,'DDSUW2',4X,'VDSUW2',//)
129) WRITE(6,9997)(XW2(I5),VUW2(I5),DUW2(I5),DSUW2(I5),TUW2(I5),
130) +HUW2(I5),DDSUW2(I5),VDSUW2(I5),I5=1,NS2)
131) K8=KN(2)+1
132) VIN=VTOT(K8)
133) CALL WAKVIS(VLW2,VIN,XW2,ZW2,TIN3,HIN3,TLW2,HLW2,DLW2,NS2,
134) +VDSLW2,DSLW2,DDSLW2,SLOPW2)
135) WRITE(6,9988)
136) 9988 FORMAT(///,19X,'XW2',6X,'VLW2',6X,'DLW2',5X,'DSLW2',6X,'TLW2'
137) +,6X,'HLW2',5X,'DDSLW2',4X,'VDSLW2',//)
138) WRITE(6,9997)(XW2(I5),VLW2(I5),DLW2(I5),DSLW2(I5),TLW2(I5),
139) +HLW2(I5),DDSLW2(I5),VDSLW2(I5),I5=1,NS2)
140) DO 4 K=K1,K2
141) GAMV(K)=- (VUW2(K)*(DSUW2(K)+TUW2(K))+VLW2(K)*(DSLW2(K)+TLW2(K))
142) GAMS(K)=VDSUW2(K)+VDSLW2(K)
143) 4 CONTINUE
144) RETURN
145) END
146) C
147) C
148) C ***** SUBROUTINE 23 *****
149) C
150) C
151) SUBROUTINE MATMUL(A,B,C,N10,N20)
152) COMMON/BL20/ITRACE
153) DIMENSION A(99,99),B(99),C(99)
154) IF(ITRACE.EQ.1)WRITE(6,9999)

```

```

155) 9999 FORMAT(//,'SUBROUTINE MATMUL'//)
156) DO 2 J=1,N10
157) SUM=0.
158) DO 4 J2=1,N20
159) SUM=SUM+A(J,J2)*B(J2)
160) 4 CONTINUE
161) C(J)=SUM
162) 2 CONTINUE
163) RETURN
164) END
165) C
166) C
167) C ***** SUBROUTINE 24 *****
168) C
169) C
170) SUBROUTINE MATINV(A,N)
171) COMMON/BL20/ITRACE
172) DIMENSION A(99,99),PIVOT(99),Y(99),IP(99)
173) REAL MAXEL
174) IF(ITRACE.EQ.1)WRITE(6,9999)
175) 9999 FORMAT(//,'SUBROUTINE MATINV'//)
176) C
177) C Matrix inversion using Gauss elimination with pivoting.
178) C
179) C The subroutine evaluates the inverse of A and maps
180) C it on top of A. Thus on exit A contains the elements
181) C of its inverse.
182) C
183) C 'PIVOT' is a linear matrix of size N which on exit holds
184) C the pivotal elements obtained during the elimination.
185) C
186) C Y,IP are used as working space.
187) C
188) C N is the matrix size limited to 120.
189) C
190) C EPS:The subroutine stops the computation if the absolute
191) C value of the current pivotal element in the
192) C elimination process is less than EPS*MAXEL where
193) C EPS is a tolerance chosen by the user. MAXEL is the
194) C largest element of A in absolute value and is found internally
195) C by the routine.
196) C
197) MAXEL=0.0
198) EPS=.0000001
199) DO 5 I=1,N
200) DO 5 J=1,N
201) IF(ABS(A(I,J)).LE.MAXEL)GOTO 5
202) MAXEL=ABS(A(I,J))
203) 5 CONTINUE
204) TOLER=MAXEL*EPS
205) DO 10 I=1,N
206) IP(I)=I
207) 10 CONTINUE
208) NN=N-1
209) DO 47 K=1,NN
210) XMAX=0.0
211) IFLAG=0
212) DO 20 I=K,N
213) IF(ABS(A(K,I)).LT.ABS(XMAX))GOTO 20
214) XMAX=A(K,I)
215) MPIVOT=I
216) IFLAG=1

```

```

(2217) 20 CONTINUE
(2218) IF(IFLAG.EQ.0)GOTO 30
(2219) IX=IP(K)
(2220) IP(K)=IP(MPIVOT)
(2221) IP(MPIVOT)=IX
(2222) DO 25 J=1,N
(2223) DUMMY=A(K,J)
(2224) A(K,J)=A(MPIVOT,J)
(2225) A(MPIVOT,J)=DUMMY
(2226) 25 CONTINUE
(2227) 30 CONTINUE
(2228) PIVOT(K)=A(K,K)
(2229) ZPIVOT=A(K,K)
(2230) IF(ABS(PIVOT(K)).LT.TOLER)GOTO 120
(2231) A(K,K)=1.
(2232) KK=K+1
(2233) DO 45 I=KK,N
(2234) FACT=A(I,K)/PIVOT(K)
(2235) A(I,K)=0.0
(2236) DO 35 J=KK,N
(2237) A(I,J)=A(I,J)-FACT*A(K,J)
(2238) 35 CONTINUE
(2239) DO 40 J=1,K
(2240) A(I,J)=A(I,J)-FACT*A(K,J)
(2241) 40 CONTINUE
(2242) 45 CONTINUE
(2243) 47 CONTINUE
(2244) PIVOT(N)=A(N,N)
(2245) ZPIVOT=A(N,N)
(2246) IF(ABS(PIVOT(N)).LT.TOLER)GOTO 120
(2247) A(N,N)=1.
(2248) DO 50 J=1,N
(2249) A(N,J)=A(N,J)/PIVOT(N)
(2250) 50 CONTINUE
(2251) K=N-1
(2252) 60 CONTINUE
(2253) DO 70 J=1,N
(2254) Y(J)=0.0
(2255) IF(J.LE.K)GOTO 70
(2256) Y(J)=A(K,J)
(2257) A(K,J)=0.0
(2258) 70 CONTINUE
(2259) DO 90 J=1,N
(2260) S=0.0
(2261) KS=K+1
(2262) DO 80 JJ=KS,N
(2263) S=S+A(JJ,J)*Y(JJ)
(2264) 80 CONTINUE
(2265) A(K,J)=(A(K,J)-S)/PIVOT(K)
(2266) 90 CONTINUE
(2267) K=K-1
(2268) IF(K.GE.1)GOTO 60
(2269) DO 97 J=1,N
(2270) IF(IP(J).EQ.J)GOTO 97
(2271) JJ=IP(J)
(2272) DO 95 I=1,N
(2273)
(2274) DUMMY=A(I,J)
(2275) A(I,J)=A(I,JJ)
(2276) A(I,JJ)=DUMMY
(2277) 95 CONTINUE
(2278) IP(J)=J

```

```

(2279) IP(JJ)=JJ
(2280) 97 CONTINUE
(2281) C WRITE(6,200)
(2282) C 200 FORMAT(/,'THE INVERSE OF THE MATRIX A',/)
(2283) DO 100 I=1,N
(2284) C WRITE(6,202) I
(2285) C 202 FORMAT(/,' ROW NO.=',I4/)
(2286) C WRITE(6,204)(A(I,J),J=1,N)
(2287) 100 CONTINUE
(2288) 204 FORMAT(5(F12.7,3X))
(2289) C FORMAT(3(2X,I3),3X,2(F10.5))
(2290) 201 FORMAT(F10.4,3X,I2,I3)
(2291) RETURN
(2292) 120 CONTINUE
(2293) WRITE(1,206)ZPIVOT
(2294) 206 FORMAT(/,'THE MATRIX IS SINGULAR THE CURRENT PIVOTAL
(2295) ELEMENT=' ,F15.7)
(2296) RETURN
(2297) END
(2298) C
(2299) C
(2300) C ***** S U B R O U T I N E 25 *****
(2301) C
(2302) C
(2303) SUBROUTINE WAKREL(X,Z,VT,VN,NS)
(2304) COMMON/BL20/ITRACE
(2305) DIMENSION X(99),Z(99),DS(99),COST(99),SINT(99),VT(99),VN(
(2306) +)
(2307) IF(ITRACE.EQ.1)WRITE(6,9999)
(2308) 9999 FORMAT(/,'SUBROUTINE WAKREL'//)
(2309) N=NS+1
(2310) DO 2 I=1,NS
(2311) DX=X(I+1)-X(I)
(2312) DZ=Z(I+1)-Z(I)
(2313) DS(I)=SQRT(DX*DX+DZ*DZ)
(2314) COST(I)=DX/DS(I)
(2315) SINT(I)=DZ/DS(I)
(2316) 2 CONTINUE
(2317) DO 4 I=2,N
(2318) SUM=0
(2319) SUM1=0
(2320) I1=I-1
(2321) DO 6 J=1,I1
(2322) VM=SQRT(VT(J)*VT(J)+VN(J)*VN(J))
(2323) R1=VT(J)/VM
(2324) R2=VN(J)/VM
(2325) SUM=SUM+DS(J)*((R1-1)*COST(J)-R2*SINT(J))
(2326) SUM1=SUM1+DS(J)*((R1-1)*SINT(J)+R2*COST(J))
(2327) 6 CONTINUE
(2328) X(I)=X(I)+SUM
(2329) Z(I)=Z(I)+SUM1
(2330) 4 CONTINUE
(2331) RETURN
(2332) END
(2333) C
(2334) C
(2335) C ***** S U B R O U T I N E 26 *****
(2336) C
(2337) C
(2338) SUBROUTINE GAUSS2(A,B,X,N)
(2339) COMMON/BL20/ITRACE
(2340) DIMENSION A(10,10),B(10),X(10)

```

```

(2341) IF(ITRACE.EQ.1)WRITE(6,9999)
(2342) 9999 FORMAT(/,'SUBROUTINE GAUSS2'//)
(2343) NN=N-1
(2344) DO 10 K=1,NN
(2345) XMAX=0.0
(2346) IFLAG=0
(2347) DO 20 I=K,N
(2348) IF(ABS(A(I,K)).LT.ABS(XMAX))GOTO 20
(2349) IFLAG=1
(2350) XMAX=A(I,K)
(2351) MPIVOT=I
(2352) 20 CONTINUE
(2353) IF(IFLAG.EQ.0)GOTO 40
(2354) DO 30 J=1,N
(2355) Y=A(K,J)
(2356) A(K,J)=A(MPIVOT,J)
(2357) A(MPIVOT,J)=Y
(2358) 30 CONTINUE
(2359) Y=B(K)
(2360) B(K)=B(MPIVOT)
(2361) B(MPIVOT)=Y
(2362) 40 CONTINUE
(2363) KK=K+1
(2364) DO 50 I=KK,N
(2365) A(I,K)=A(I,K)/A(K,K)
(2366) DO 60 J=KK,N
(2367) A(I,J)=A(I,J)-A(I,K)*A(K,J)
(2368) 60 CONTINUE
(2369) B(I)=B(I)-A(I,K)*B(K)
(2370) 50 CONTINUE
(2371) 10 CONTINUE
(2372) X(N)=B(N)/A(N,N)
(2373) K=N-1
(2374) 80 KS=K+1
(2375) S=0.0
(2376) DO 70 J=KS,N
(2377) S=S+A(K,J)*X(J)
(2378) 70 CONTINUE
(2379) X(K)=(B(K)-S)/A(K,K)
(2380) K=K-1
(2381) IF(K.GE.1)GOTO 80
(2382) RETURN
(2383) END
(2384) C
(2385) C
(2386) C ***** SUBROUTINE 27 *****
(2387) C
(2388) C
(2389) SUBROUTINE W1W2S1(X,Z,GVM,GSM,V1N1,SIN1,NN,IFLAG)
(2390) C
(2391) C*****
(2392) C The subroutine calculates the wake singularity *
(2393) C at the panel ends using the singularity value *
(2394) C at the panel mid-point *
(2395) C*****
(2396) C
(2397) COMMON/BL10/GAMV1(99),GAMV2(99),GAMS1(99),GAMS2(99)
(2398) COMMON/BL20/ITRACE
(2399) DIMENSION X(99),Z(99),GVM(99),GSM(99)
(2400) IF(ITRACE.EQ.1)WRITE(6,9999)
(2401) 9999 FORMAT(/,'SUBROUTINE W1W2S1'//)
(2402) N=NN

```

```

(2403)      GVM(1)=V1N1
(2404)      GSM(1)=GAMS1(1)*1.2
(2405)      IF(IFLAG.EQ.1)GOTO 100
(2406)      DO 2 I=2,N
(2407)      GVM(I)=(GAMV1(I)+GAMV1(I-1))*0.5
(2408)      GSM(I)=(GAMS1(I)+GAMS1(I-1))*0.5
(2409)      2  CONTINUE
(2410)      GVM(NN+1)=GAMV1(NN)*0.5
(2411)      GSM(NN+1)=GAMS1(NN)*0.5
(2412)      RETURN
(2413)      100 CONTINUE
(2414)      GSM(1)=GAMS2(1)*1.2
(2415)      DO 4 I=2,N
(2416)      GVM(I)=(GAMV2(I)+GAMV2(I-1))*0.5
(2417)      GSM(I)=(GAMS2(I)+GAMS2(I-1))*0.5
(2418)      4  CONTINUE
(2419)      GVM(NN+1)=GAMV2(NN)*0.5
(2420)      GSM(NN+1)=GAMS2(NN)*0.5
(2421)      RETURN
(2422)      END
(2423)      C
(2424)      C
(2425)      C ***** S U B R O U T I N E 28 *****
(2426)      C
(2427)      C
(2428)      SUBROUTINE SPLINE(X1,Y1,DY,DDY,K10,ISTART,NOT)
(2429)      COMMON/BL20/ITRACE
(2430)      DOUBLE PRECISION X,Y,W,WORK1,WORK2,C,SS,S,K(99)
(2431)      DIMENSION Y(99),X(99),DY(99),DDY(99),W(99),
(2432)      +WORK1(99),WORK2(4,99),C(99),S(5),X1(99),Y1(99)
(2433)      IF(ITRACE.EQ.1)WRITE(6,9999)
(2434)      9999 FORMAT(/,'SUBROUTINE SPLINE'//)
(2435)      DO 1 I=1,K10
(2436)      X(I)=X1(I)
(2437)      Y(I)=Y1(I)
(2438)      1  CONTINUE
(2439)      IF(NOT.EQ.1)GOTO 100
(2440)      CALL DU1DU2(Y,X,DY,DDY,K10)
(2441)      NCAP=0
(2442)      IK=5
(2443)      DO 10 I=1,K10
(2444)      IF(ABS(DY(I)).LT..4)GOTO 10
(2445)      K(IK)=X(I)
(2446)      IK=IK+1
(2447)      IF(I.EQ.1.OR.I.EQ.K10)GOTO 10
(2448)      IF(DY(I+1)*DY(I-1).GE.0)GOTO 10
(2449)      K(IK)=X(I)
(2450)      K(IK+1)=X(I)
(2451)      IK=IK+2
(2452)      10 CONTINUE
(2453)      GOTO 102
(2454)      100 CONTINUE
(2455)      C WRITE(1,200)(X(I),Y(I),I=1,K10)
(2456)      200 FORMAT(2(F12.7,3X))
(2457)      IK=5
(2458)      K9=K10-4
(2459)      DO 12 J1=ISTART,K9
(2460)      K(IK)=X(J1)
(2461)      IK=IK+1
(2462)      12 CONTINUE
(2463)      102 CONTINUE
(2464)      NCAP=IK-4

```

```

(2465)      NCAP7=NCAP+7
(2466)      DO 20 I=1,K10
(2467)      W(I)=1.
(2468)  20   CONTINUE
(2469)      IFAIL=0
(2470)      CALL E02BAF(K10,NCAP7,X,Y,W,K,WORK1,WORK2,C,SS,IFAIL)
(2471)      DO 30 I=1,K10
(2472)      IFAIL=0
(2473)      CALL E02BCF(NCAP7,K,C,X(I),1,S,IFAIL)
(2474)      DY(I)=S(2)
(2475)      DDY(I)=S(3)
(2476)  30   CONTINUE
(2477)  40   RETURN
(2478)      END
(2479)  C
(2480)  C
(2481)  C      ***** S U B R O U T I N E      29 *****
(2482)  C
(2483)  C
(2484)      SUBROUTINE WAKVIS(UW,UWIN,XW,ZW,TIN,HIN,STOT,STOH,STOD,NS1,
(2485) +VDSUW,DSTAR,DSTAR2,DANG)
(2486)      COMMON/BL20/ITRACE
(2487)  C
(2488)  C      *****
(2489)  C      * Calculates the viscous flow on one side of the wake *
(2490)  C      * using GREENS LAG ENTRAINMENT method *
(2491)  C      *****
(2492)  C
(2493)      DIMENSION UW(99),XW(99),ZW(99),STOT(99),STOH(99),STOD(99)
(2494) +,SSW(99),SSL(99),UDAM(99),DUDAM(99),ANG(99),DANG(99),VDSUW(
(2495) +),DSTAR(99),DSTAR1(99),DSTAR2(99),RUBISH(99),SST(99),DDUDAM(
(2496) +9),STREN(99),DSTREN(99),HELP(99),HELP2(99)
(2497)      IF(ITRACE.EQ.1)WRITE(6,9999)
(2498)  9999  FORMAT(/,'SUBROUTINE WAKVIS'//)
(2499)      IF(TIN.LT.0001)TIN=0.009
(2500)      IF(HIN.LT.1.5.OR.HIN.GT.1.9)HIN=1.9
(2501)      SSL(1)=0.
(2502)      UDAM(1)=UW(1)*1.05
(2503)      H=HIN
(2504)      RLAM=.5
(2505)      H1=3.15+1.72/(H-1.)-.01*(H-1.)*(H-1.)
(2506)      Q0=-((1.25/H)*((H-1.)/(6.432*H))**2
(2507)      CEQ0=-H1*(H+1.)*Q0
(2508)      CTQ0=.024*CEQ0+1.2*CEQ0*CEQ0
(2509)      CEQ=SQRT(CTQ0/(1.2*RLAM**2)+.0001)-.01
(2510)      CE=CEQ
(2511)      M=1
(2512)      DO 4 I=1,NS1
(2513)      M=M+1
(2514)      DX=XW(I+1)-XW(I)
(2515)      DZ=ZW(I+1)-ZW(I)
(2516)      SSW(I)=SQRT(DX*DX+DZ*DZ)
(2517)      ANG(I)=ATAN(DZ/DX)
(2518)      IF(I.LT.NS1)SST(I+1)=SST(I)+SSW(I)
(2519)      SSL(M)=SSL(M-1)+SSW(I)*.5
(2520)      UDAM(M)=UW(I)
(2521)      M=M+1
(2522)      SSL(M)=SSL(M-1)+SSW(I)*.5
(2523)      IF(I.LT.NS1)UDAM(M)=(UW(I+1)+UW(I))* .5
(2524)  4     CONTINUE
(2525)      K10=2*NS1
(2526)      CALL DU1DU2(UDAM,SSL,DUDAM,DDUDAM,K10)

```



```

) M=1
) DSTAR(M)=HIN*TIN
) T=TIN
) STREN(M)=DSTAR(M)*UDAM(M)
) DO 6 J=1,NS1
) IFLAG=1
) 20 DT=-(H+2.)*T*DUDAM(M)/UDAM(M)
) H1=3.15+1.72/(H-1.)-.01*(H-1.)*(H-1.)
) DHH1=-(H-1.)*(H-1.)/(1.72+.02*(H-1.)*(H-1.))
) DH=DHH1*(CE+H1*(H+1.)*T*DUDAM(M)/UDAM(M))/T
) CT=(.024+1.2*CE)*CE
) IF(ABS(CT).LE.0.000001)CT=0
) F=CT/(.012+1.2*CE)
) Q0=-(1.25/H)*((H-1.)/(6.432*H))**2
) CEQ0=-H1*(H+1.)*Q0
) CTQ0=.024*CEQ0+1.2*CEQ0*CEQ0
) CEQ=SQRT(CTQ0/(1.2*RLAM**2)+.0001)-.01
) Q=-(CEQ)/(H1*(H+1.))
) DCE=F*((2.8/(H+H1))*(SQRT(CTQ0)-RLAM*SQRT(CT))+Q-T*DUDAM(M)/UDAM(
) +))/T
) IF(DH.GE.3.)DH=3.
) IF(DH.LE.-3.)DH=-3.
) T=T+DT*SSW(J)*.5
) H=H+DH*SSW(J)*.5
) CE=CE*DCE*SSW(J)*.5
) M=M+1
) DSTAR(M)=H*T
) STREN(M)=DSTAR(M)*UDAM(M)
) IF(IFLAG.EQ.0)GOTO 6
) IFLAG=0
) STOT(J)=T
) STOJ(J)=H
) IF(J.EQ.NS1)GOTO 6
) GOTO 20
) 6 CONTINUE
) DO 30 I=1,M
) HELP(I)=DSTAR(I)
) 30 CONTINUE
) CALL DU1DU2(DSTAR,SSL,DSTAR1,DSTAR2,M)
) CALL DU1DU2(STREN,SSL,DSTREN,RUBISH,M)
) DO 40 I=1,M
) DSTAR2(I)=HELP2(I)
) 40 CONTINUE
) NK=1
) DO 8 I=2,M,2
) DSTAR1(NK)=DSTAR1(I)
) DSTAR2(NK)=DSTAR2(I)
) UDAM(NK)=UDAM(I)
) DUDAM(NK)=DUDAM(I)
) VDSUW(NK)=DSTREN(I)
) NK=NK+1
) 8 CONTINUE
) CALL DU1DU2(ANG,SST,DANG,RUBISH,NS1)
) 600 RETURN
) END
) C
) C
) C ***** SUBROUTINE 30 *****
) C
) C
) SUBROUTINE CNXCLD(CAP,CNP,CMP,CAF,CNF,CMF,CAT,CNT,CMT,CL,KN,
) +NT)

```

```

9) COMMON/BLOCK3/RE1,ISTAG(10)
0) COMMON/BL5/X(99),Z(99),S(99),SA(99),CN1(99),SN1(99),
1) +T1(99),T3(99)
2) COMMON/BL18/CP(99),CF(99)
3) COMMON/BL20/ITRACE
4) COMMON/BL25/CDU1,CDU2,CDL1,CDL2
5) DIMENSION CAP(10),CNP(10),CMP(10),CAF(10),CNF(10),CMF(10),KN(10)
6) IF(ITRACE.EQ.1)WRITE(6,9999)
7) 9999 FORMAT(//,'SUBROUTINE CNXCLD'//)
8) KK5=0
9) DO 2 I=2,NT
0) K3=KN(I-1)+1
1) K4=KN(I)
2) IF(I.EQ.3)K3=KN(I-1)+2
3) IF(I.EQ.3)K4=KN(I)+1
4) SUM=0
5) SUM1=0.
6) SUM2=0.
7) DO 4 K=K3,K4
8) KK5=KK5+1
9) CP1=CP(KK5)
0) SUM=SUM+CP1*(Z(K+1)-Z(K))
1) SUM1=SUM1-CP1*(X(K+1)-X(K))
2) SUM2=SUM2+CP1*(.5*(X(K)+X(K+1))*(X(K+1)-X(K))+.5*(Z(K)+Z(K+1))
3) +*(Z(K+1)-Z(K)))
4) 4 CONTINUE
5) CAP(I)=SUM
6) CNP(I)=SUM1
7) CMP(I)=SUM2
8) SUM=0
9) SUM1=0.
0) SUM2=0.
1) KK6=0
2) DO 6 K=K3,K4
3) KK6=KK6+1
4) CF1=CF(KK6)
5) IF(KK6.LE.ISTAG(I))CF1=-CF(KK6)
6) SUM=SUM+CF1*(Z(K+1)-Z(K))
7) SUM1=SUM1+CF1*(X(K+1)-X(K))
8) SUM2=SUM2+CF1*(.5*(X(K)+X(K+1))*(X(K+1)-X(K))+.5*(Z(K)+Z(K+1))
9) +*(Z(K+1)-Z(K)))
0) 6 CONTINUE
1) CAF(I)=SUM
2) CNF(I)=SUM1
3) CMF(I)=SUM2
4) 2 CONTINUE
5) CAT=0.
6) CNT=0.
7) CMT=0.
8) DO 8 I=2,NT
9) CAT=CAT+CAF(I)+CAP(I)
0) CNT=CNT+CNF(I)+CNP(I)
1) CMT=CMT+CMF(I)+CMP(I)
2) 8 CONTINUE
3) CL=CNT*COS(ALPHA)-CAT*SIN(ALPHA)
4) CD=CDU1+CDU2+CDL1+CDL2
5) WRITE(6,9992)
6) 9992 FORMAT(///,28X,'CAP',8X,'CNP',8X,'CMP',8X,'CAF',8X,'CNF',8X,'CMF
7) +///)
8) WRITE(6,9991)(CAP(I5),CNP(I5),CMP(I5),CAF(I5),CNF(I5),CMF(I5),I5
9) +,NT)
0) 9991 FORMAT(28X,F8.5,2X,F8.5,2X,F8.5,2X,F8.5,2X,F8.5,2X,F8.5)

```

```

(2651) WRITE(1,9995)CAT,CNT,CMT,CL,CD
(2652) WRITE(6,9995)CAT,CNT,CMT,CL,CD
(2653) 9995 FORMAT(//,'CAT=',F8.5,3X,'CNT=',F8.5,3X,'CMT=',F8.5,3X,
(2654) +'CL=',F8.5,3X,'CD=',F8.5,//)
(2655) 40 RETURN
(2656) END
(2657) C
(2658) C
(2659) C ***** S U B R O U T I N E 31 *****
(2660) C
(2661) C
(2662) SUBROUTINE WAKEG1(ISTORE,IFLAG,KN,XW,ZW)
(2663) COMMON/BL4/ALPHA2,KN,NS1,NS2,PI
(2664) COMMON/BL5/X(99),Z(99),S(99),SA(99),CN1(99),SN1(99),
(2665) +T1(99),T3(99)
(2666) COMMON/BL20/ITRACE
(2667) DIMENSION KN(10),XW(99),ZW(99)
(2668) IF(ITRACE.EQ.1)WRITE(6,9999)
(2669) 9999 FORMAT(//,'SUBROUTINE WAKEG1'//)
(2670) KN3=KN(3)+1
(2671) IF(IFLAG.EQ.1)GOTO 100
(2672) NF=KN(3)-KN(2)
(2673) IBEG=KN(2)+1+3*NF/8
(2674) XMAX=10.
(2675) DO 20 I=IBEG,KN3
(2676) DIST=(X(1)-X(I))**2+(Z(1)-Z(I))**2
(2677) IF(DIST.GT.XMAX)GOTO 30
(2678) XMAX=DIST
(2679) 20 CONTINUE
(2680) WRITE(1,202)
(2681) 202 FORMAT(//,'ERROR IN SUB 25 AFTER FORMAT 202',//)
(2682) 30 CONTINUE
(2683) ISTORE=I-2
(2684) 100 CONTINUE
(2685) X1=X(ISTORE)
(2686) X2=X(ISTORE+1)
(2687) X3=X(ISTORE+2)
(2688) Y1=Z(ISTORE)
(2689) Y2=Z(ISTORE+1)
(2690) Y3=Z(ISTORE+2)
(2691) CALL QUAD(X1,X2,X3,Y1,Y2,Y3,AL,BL,CL)
(2692) CALL CUBIC(AL,BL,CL,ISTORE,XNOM,YNOM)
(2693) ISTART=ISTORE+2
(2694) IF(XNOM.GE.X1.AND.XNOM.LE.X2)ISTART=ISTORE+1
(2695) KN2=KN(2)+1
(2696) TET=Z(KN2)-Z(1)
(2697) HSL0T=Z(1)-YNOM+.5*TET
(2698) XW(1)=X(1)
(2699) ZW(1)=Z(1)+TET*.5
(2700) NCOUNT=2
(2701) C WRITE(1,206)ISTART,KN3
(2702) 206 FORMAT(2(I5,2X))
(2703) DO 40 I=ISTART,KN3
(2704) IF(I.EQ.KN3)GOTO 102
(2705) X1=X(I-1)
(2706) X2=X(I)
(2707) X3=X(I+1)
(2708) Y1=Z(I-1)
(2709) Y2=Z(I)
(2710) Y3=Z(I+1)
(2711) CALL QUAD(X1,X2,X3,Y1,Y2,Y3,AL,BL,CL)
(2712) 102 CONTINUE

```

```

(2713)      X2=X(I)
(2714)      Y2=Z(I)
(2715)      SLOPE=2.*AL*X(I)+BL
(2716)      SLOPEN=-1./SLOPE
(2717)      XP=X2+HSL0T/SQRT(1.+SLOPEN**2)
(2718)      ZP=SLOPEN*XP+Y2-SLOPEN*X2
(2719)      IF(ZP.GT.Y2)GOTO 104
(2720)      XP=X2-HSL0T/SQRT(1.+SLOPEN**2)
(2721)      ZP=SLOPEN*XP+Y2-SLOPEN*X2
(2722)  104  CONTINUE
(2723)      XW(NCOUNT)=XP
(2724)      ZW(NCOUNT)=ZP
(2725)      NS1=NCOUNT-1
(2726)      NCOUNT=NCOUNT+1
(2727)  40   CONTINUE
(2728)      WRITE(1,204)(XW(I),ZW(I),I=1,NCOUNT)
(2729)  204  FORMAT(F12.7,3X,F12.7)
(2730)  999  RETURN
(2731)      END
(2732)  C
(2733)  C
(2734)  C      ***** S U B R O U T I N E      32 *****
(2735)  C
(2736)  C
(2737)      SUBROUTINE QUAD(X1,X2,X3,Y1,Y2,Y3,AL,BL,CL)
(2738)      A1=Y1*X2+Y2*X3+Y3*X1-Y3*X2-X3*Y1-Y2*X1
(2739)      A2=X1*X1*Y2+X2*X2*Y3+X3*X3*Y1-X3*X3*Y2-Y3*X1*X1-Y1*X2*X2
(2740)      A3=X1*X1*X2+X2*X2*X3+X3*X3*X1-X3*X3*X2-X3*X1*X1-X2*X2*X1
(2741)      AL=A1/A3
(2742)      BL=A2/A3
(2743)      CL=Y1-BL*X1-AL*X1*X1
(2744)      RETURN
(2745)      END
(2746)  C
(2747)  C
(2748)  C      ***** S U B R O U T I N E      33 *****
(2749)  C
(2750)  C
(2751)      SUBROUTINE CUBIC(AL,BL,CL,ISTORE,XNOM,YNOM)
(2752)      COMMON/BL5/X(99),Z(99),S(99),SA(99),CN1(99),SN1(99),
(2753)      +T1(99),T3(99)
(2754)      COMMON/BL20/ITRACE
(2755)      DIMENSION XX(4)
(2756)      IF(ITRACE.EQ.1)WRITE(6,9999)
(2757)  9999  FORMAT(/,'SUBROUTINE CUBIC'//)
(2758)      EPS=.000001
(2759)      A11=2.*AL*AL
(2760)      A22=3.*AL*BL
(2761)      A33=2.*AL*CL+BL*BL-2.*AL*Z(1)+1
(2762)      A44=CL*BL-X(1)-BL*Z(1)
(2763)      XX(1)=X(ISTORE)
(2764)      XX(2)=X(ISTORE+2)
(2765)      X1=XX(1)
(2766)      X2=XX(2)
(2767)      X3=X2
(2768)      FX1=A11*XX(1)*XX(1)*XX(1)+A22*XX(1)*XX(1)+A33*XX(1)+A44
(2769)  10   CONTINUE
(2770)      FX2=A11*XX(2)*XX(2)*XX(2)+A22*XX(2)*XX(2)+A33*XX(2)+A44
(2771)      XX(3)=XX(2)-(XX(2)-XX(1))*FX2/(FX2-FX1)
(2772)      IF(ABS(XX(3)-XX(2)).LT.EPS)GOTO 20
(2773)      XX(1)=XX(2)
(2774)      XX(2)=XX(3)

```

```

775)      FX1=FX2
776)      GOTO 10
777)  20  CONTINUE
778)  C    WRITE(1,202)XX(3),X1,X3
779)  202  FORMAT(3(F10.5,3X))
780)      IF(XX(3).GE.X1.AND.XX(3).LE.X3)GOTO 30
781)      F1=A11
782)      G1=A22+XX(3)*F1
783)      H1=-A44/XX(3)
784)      R1=G1*G1-4.*F1*H1
785)      IF(R1.LT.0.)GOTO 40
786)      DELTA=SQRT(R1)
787)      XX(3)=(-G1+DELTA)/(2.*F1)
788)      IF(XX(3).GE.X1.AND.XX(3).LE.X3)GOTO 30
789)      XX(3)=(-G1-DELTA)/(2.*F1)
790)      IF(XX(3).GE.X1.AND.XX(3).LE.X3)GOTO 30
791)  40  WRITE(1,200)
792)  200  FORMAT(//,'ERROR   SUB 27 *** NO REAL ROOT *** FORMAT 200',//
793)      RETURN
794)  30  YY3=AL*XX(3)*XX(3)+BL*XX(3)+CL
795)      XNOM=XX(3)
796)      YNOM=YY3
797)      RETURN
798)      END
799)  C
800)  C
801)  C    ***** S U B R O U T I N E      34 *****
802)  C
803)  C
804)      SUBROUTINE WAKEG2(KN,NT,XW2,ZW2)
805)      COMMON/BL4/ALPHA2,KK,NS1,NS2,PI
806)      COMMON/BL5/X(99),Z(99),S(99),SA(99),CN1(99),SN1(99),
807)      +T1(99),T3(99)
808)      COMMON/BL20/ITRACE
809)      DIMENSION KN(10),XW2(99),ZW2(99)
810)      IF(ITRACE.EQ.1)WRITE(6,9999)
811)  9999  FORMAT(//,'SUBROUTINE WAKEG2'//)
812)      J=NT-1
813)      IF(NT.EQ.2)KN3=KN(NT)+1
814)      IF(NT.EQ.2)KN2=KN(1)+1
815)      IF(NT.EQ.3)KN3=KN(NT)+2
816)      IF(NT.EQ.3)KN2=KN(2)+2
817)      IVAR=KN2
818)      XMAX=10.
819)  100  CONTINUE
820)      IF(X(IVAR).GT.XMAX)GOTO 102
821)      XMAX=X(IVAR)
822)      IVAR=IVAR+1
823)      IF(IVAR.GT.KN3)GOTO 108
824)      GOTO 100
825)  102  CONTINUE
826)      ISTORE=IVAR-1
827)      XT=(X(KN2)+X(KN3))* .5
828)      ZT=(Z(KN2)+Z(KN3))* .5
829)      X1=X(ISTORE)
830)      Z1=Z(ISTORE)
831)      DX=XT-X(ISTORE)
832)      DZ=Z(ISTORE)-ZT
833)      XWT=X(ISTORE)+2.*DX
834)      ZWT=Z(ISTORE)-2.*DZ
835)      NF=(KN3-KN2+1)/2
836)      IF(NF.GT.18)NF=18

```

```

7)      XSTEP=DX/NF
3)      XW2(1)=XT
7)      ZW2(1)=ZT
0)      SLOPE=(Z1-ZWT)/(X1-XWT)
1)      B1=ZWT-SLOPE*XWT
2)      NCOUNT=2
3)      XSUM=XT
4)  104  CONTINUE
5)      XW2(NCOUNT)=XSUM+XSTEP
6)      ZW2(NCOUNT)=SLOPE*(XSUM+XSTEP)+B1
7)      XSUM=XW2(NCOUNT)
8)      NCOUNT=NCOUNT+1
9)      IF(NCOUNT.EQ.NF+1)GOTO 106
0)      GOTO 104
1)  106  CONTINUE
2)      XW2(NF+1)=XWT
3)      ZW2(NF+1)=ZWT
4)      NF1=NF+1
5)      NS2=NF1-1
6)      IF(NT.EQ.2)NS1=NF1-1
7)      IF(NT.EQ.2)NS2=0
8)  C    WRITE(6,204)(XW2(I),ZW2(I),I=1,NF1)
9)  204  FORMAT(F10.5,3X,F10.5)
0)      RETURN
1)  108  CONTINUE
2)      WRITE(1,202)
3)  202  FORMAT(//,'ERROR SUB 28 *** FORMAT 202 *** LEADING EDGE NOT FOU
4)      +',//)
5)      RETURN
6)      END
7)  C
8)  C
9)  C    ***** S U B R O U T I N E    35 *****
0)  C
1)  C
2)      SUBROUTINE PD2001(ITER,NT,KN)
3)      COMMON/BL4/ALPHA2,KK,NS1,NS2,PI
4)      COMMON/BL5/X(99),Z(99),S(99),SA(99),CN1(99),SN1(99),
5)      +T1(99),T3(99)
6)      COMMON/BL6/XW1(99),ZW1(99),XW2(99),ZW2(99),SW1(99),SW2(99)
7)      COMMON/BL7/XBW1(99),ZBW1(99),XBW2(99),ZBW2(99)
8)      COMMON/BL12/SNW1(99),SNW2(99),CNW1(99),CNW2(99)
9)      COMMON/BL13/TW11(99),TW13(99),TW21(99),TW23(99)
0)      COMMON/BL17/XBAR(99),ZBAR(99)
1)      COMMON/BL20/ITRACE
2)      DIMENSION KN(10)
3)      IF(ITRACE.EQ.1)WRITE(6,9999)
4)  9999  FORMAT(//,'SUBROUTINE PD2001'//)
5)      WRITE(6,9980)ITER
6)  9980  FORMAT(//,20X,'I T E R A T I O N    N O ',I3,//)
7)      WRITE(6,9970)
8)  9970  FORMAT(10X,'FIRST ELEMENT',/)
9)      WRITE(6,9960)
0)  9960  FORMAT(9X,'K',9X,'DS',8X,'T1',8X,'T3',7X,'SN1',7X,'CN1',/)
1)      ALPHA=ALPHA2
2)      IF(ITER.GT.1)GOTO 10
3)      KK5=0
4)      DO 2 JJ=2,NT
5)      IF(JJ.EQ.2)K3=1
6)      IF(JJ.EQ.2)K4=KN(JJ)
7)      IF(JJ.EQ.3)K3=K4+2
8)      IF(JJ.EQ.3)K4=KN(JJ)+1

```

```

:899) DO 4 K=K3,K4
:900) KK5=KK5+1
:901) XBAR(KK5)=(X(K+1)+X(K))* .5
:902) ZBAR(KK5)=(Z(K+1)+Z(K))* .5
:903) DX=X(K+1)-X(K)
:904) DZ=Z(K+1)-Z(K)
:905) S(KK5)=SQRT(DX*DX+DZ*DZ)
:906) DS=S(KK5)
:907) T1(KK5)=DX/DS
:908) T3(KK5)=DZ/DS
:909) SN1(KK5)=(DZ*COS(ALPHA)-DX*SIN(ALPHA))/DS
:910) CN1(KK5)=(DX*COS(ALPHA)+DZ*SIN(ALPHA))/DS
:911) WRITE(6,9990)KK5,S(KK5),T1(KK5),T3(KK5),SN1(KK5),CN1(KK5)
:912) 9990 FORMAT(8X,I3,5X,F8.6,2X,F8.6,2X,F8.6,2X,F8.6,2X,F8.6)
:913) 4 CONTINUE
:914) IF(NT.EQ.3.AND.JJ.EQ.2)WRITE(6,9998)
:915) 9998 FORMAT(/,10X,'SECOND ELEMENT',/)
:916) 2 CONTINUE
:917) 10 CONTINUE
:918) WRITE(6,9950)NS1
:919) 9950 FORMAT(/,10X,'WAKE OF FIRST ELEMENT *** NO OF PANELS ',I3,/)
:920) WRITE(6,9940)
:921) 9940 FORMAT(9X,'K',8X,'XW1',7X,'ZW1',7X,'DSW1',6X,'TW11',6X,'TW13',
:922) DO 6 I=1,NS1
:923) DX=XW1(I+1)-XW1(I)
:924) DZ=ZW1(I+1)-ZW1(I)
:925) SW1(I)=SQRT(DX*DX+DZ*DZ)
:926) XBW1(I)=(XW1(I+1)+XW1(I))* .5
:927) ZBW1(I)=(ZW1(I+1)+ZW1(I))* .5
:928) TW11(I)=DX/SW1(I)
:929) TW13(I)=DZ/SW1(I)
:930) SNW1(I)=(DZ*COS(ALPHA)-DX*SIN(ALPHA))/SW1(I)
:931) CNW1(I)=(DX*COS(ALPHA)+DZ*SIN(ALPHA))/SW1(I)
:932) WRITE(6,9990)I,XW1(I),ZW1(I),SW1(I),TW11(I),TW13(I)
:933) 6 CONTINUE
:934) IF(NT.EQ.2)GOTO 40
:935) WRITE(6,9930)NS2
:936) 9930 FORMAT(/,10X,'WAKE OF SECOND ELEMENT *** NO OF PANELS',I3,/)
:937) WRITE(6,9920)
:938) 9920 FORMAT(9X,'K',8X,'XW2',7X,'ZW2',7X,'DSW2',6X,'TW21',6X,'TW23',
:939) DO 8 I=1,NS2
:940) DX=XW2(I+1)-XW2(I)
:941) DZ=ZW2(I+1)-ZW2(I)
:942) SW2(I)=SQRT(DX*DX+DZ*DZ)
:943) XBW2(I)=(XW2(I+1)+XW2(I))* .5
:944) ZBW2(I)=(ZW2(I+1)+ZW2(I))* .5
:945) TW21(I)=DX/SW2(I)
:946) TW23(I)=DZ/SW2(I)
:947) SNW2(I)=(DZ*COS(ALPHA)-DX*SIN(ALPHA))/SW2(I)
:948) CNW2(I)=(DX*COS(ALPHA)+DZ*SIN(ALPHA))/SW2(I)
:949) WRITE(6,9990)I,XW2(I),ZW2(I),SW2(I),TW21(I),TW23(I)
:950) 8 CONTINUE
:951) 40 RETURN
:952) END
:953) C
:954) C
:955) C ***** SUBROUTINE 36 *****
:956) C
:957) C
:958) SUBROUTINE PLOT1(IPL0T,NT,X100,Y100,NS)
:959) COMMON/BLOCK1/ALPHA,CL,CBAR,XM,ZM,CHORD
:960) COMMON/BLOCK2/CN(10),CM(10),COFP(10),CNT,CMT,COFPT

```

```

61) COMMON/BLOCK3/RE1,ISTAG(10)
62) COMMON/BL4/ALPHA2, KK, NS1, NS2, PI
63) COMMON/BL5/X(99), Z(99), S(99), SA(99), CN1(99), SN1(99),
64) +T1(99), T3(99)
65) COMMON/BL6/XW1(99), ZW1(99), XW2(99), ZW2(99), SW1(99), SW2(99)
66) COMMON/BL7/XBW1(99), ZBW1(99), XBW2(99), ZBW2(99)
67) COMMON/BL8/VN1(99), VN2(99), VT1(99), VT2(99)
68) COMMON/BL9/VUW1(99), VLW1(99), VUW2(99), VLW2(99)
69) COMMON/BL10/GAMV1(99), GAMV2(99), GAMS1(99), GAMS2(99)
70) COMMON/BL11/GAMV11(99), GAMV22(99), GAMS11(99), GAMS22(99)
71) COMMON/BL12/SNW1(99), SNW2(99), CNW1(99), CNW2(99)
72) COMMON/BL13/TW11(99), TW13(99), TW21(99), TW23(99)
73) COMMON/BL15/GAMA(99), VTOT(99)
74) COMMON/BL16/TIN, HIN, TIN1, HIN1, TIN2, HIN2, TIN3, HIN3
75) COMMON/BL17/XBAR(99), ZBAR(99)
76) COMMON/BL18/CP(99), CF(99)
77) COMMON/BL19/KN(10)
78) COMMON/BL20/ITRACE
79) COMMON/BL21/PHI(10)
80) DIMENSION X100(99), Y100(99)
81) ALPH1=ALPHA*199/3.1415926536
82) IF(M6.EQ.0)WRITE(1,2999)
83) 2999 FORMAT(/,10X,'ENTER DEVICE YOU WANT TO USE FOR PLOTTING'//,
84) +23X,'VDU 1'/,
85) +23X,'TREND 2'/,
86) +23X,'TEKTRONIX 3'/,
87) +23X,'SIGMA 4'/,
88) +23X,'C1051N 5'//)
89) IF(M6.EQ.0)READ(1,*)M6
90) IF(M6.EQ.1)CALL VDU
91) IF(M6.EQ.2)CALL TREND
92) IF(M6.EQ.3)CALL T4010
93) IF(M6.EQ.4)CALL S5660
94) IF(M6.EQ.5)CALL C1051N
95) CALL PICCLE
96) IF(IPLOT.EQ.100)GOTO 100
97) IF(IPLOT.EQ.102)GOTO 102
98) IF(IPLOT.EQ.104)GOTO 104
99) 100 CALL PENSEL(1,0.0,0)
000) CALL AXIPLD(0,120.,120.,3,2,16,16,0.,1.6,-0.8,0.8,'X',0,'Z/C',
001) IF(M6.EQ.4)CALL AXIPLD(0,300.,300.,3,2,20,20,0.,2.,-1.,1.,
002) +'X/C',3,'Z/C',3)
003) CALL GRAMDV(.4,-0.75)
004) CALL CHAA1(IARRAY,48)
005) CALL PENSEL(2,0.0,0)
006) DO 2 I=2,NT
007) IF(I.EQ.2)K3=KN(I)+1
008) IF(I.EQ.2)K2=1
009) IF(I.EQ.3)K3=KN(I)+2
010) IF(I.EQ.3)K2=KN(I-1)+2
011) IF(I.EQ.2)KK3=K3+1
012) IF(I.EQ.3)KK4=K3+1
013) CALL GRAMDV(X(K2),Z(K2))
014) DO 4 J=K2,K3
015) CALL GRALIN(X(J),Z(J))
016) 4 CONTINUE
017) CALL GRALIN(X(K2),Z(K2))
018) 2 CONTINUE
019) IF(NS1.EQ.0)RETURN
020) XTE1=(X(KK3)+X(1))*0.5
021) ZTE1=(Z(KK3)+Z(1))*0.5
022) CALL GRAMDV(XTE1,ZTE1)

```



```

)23)      NS=NS1+1
)24)      DO 6 I=1,NS
)25)      CALL GRALIN(XW1(I),ZW1(I))
)26) 6     CONTINUE
)27)      IF(NS2.EQ.0)RETURN
)28)      XTE2=(X(KK3+1)+X(KK4))*0.5
)29)      ZTE2=(Z(KK3+1)+Z(KK4))*0.5
)30)      CALL GRAMOV(XTE2,ZTE2)
)31)      NS=NS2+1
)32)      DO 8 I=1,NS
)33)      CALL GRALIN(XW2(I),ZW2(I))
)34) 8     CONTINUE
)35)      RETURN
)36) 102   CALL PENSEL(1,0.0,0)
)37)      CALL MAX(CP, KK, Y45, Y46, K47, Y55, Y56, Y57, Y65)
)38)      CALL AXIPLD(0, 120., Y65, 3, 2, 12, K47, 0., 1.2, 1., Y55, 'X/C', 0, 'CP', 2)
)39)      CALL GRAMOV(0.15, Y56)
)40)      CALL CHAHOL(36H 2D VISCOUS INCOMPRESSIBLE FLOW *.)
)41)      CALL GRAMOV(.4, Y57)
)42)      CALL CHAHOL(16H COMPUTATION*.)
)43)      CALL GRAMOV(.3, -0.3)
)44)      CALL GRAMOV(.3, 2.5)
)45)      CALL CHAHOL(38HPRESSURE DISTRIBUTION X/C*.)
)46)      CALL GRAMOV(.2, -Y46)
)47)      CALL CHAHOL(19HANGLE OF ATTACK =*.)
)48)      CALL GRAMOV(.75, -Y46)
)49)      CALL CHAFIX(ALPH1, 5, 2)
)50)      IF(PHI(1).EQ.0.0)GOTO 302
)51)      CALL GRAMOV(.2, -Y45)
)52)      CALL CHAHOL(19HFLAP DEFLECTION =*.)
)53)      CALL GRAMOV(.75, -Y45)
)54)      CALL CHAFIX(PHI, 4, 1)
)55) 302   CALL PENSEL(2,0.0,0)
)56)      CALL PLOT(XBAR, CP, KN, NT, KK)
)57)      RETURN
)58) 104   CALL PENSEL(1,0.0,0)
)59)      CALL GRAF(X100, Y100, NS, 0)
)60)      RETURN
)61)      END
)62)  C
)63)  C
)64)  C ***** SUBROUTINE 37 *****
)65)  C
)66)  C
)67)      SUBROUTINE GAUSS(A, IP, N)
)68)      DIMENSION A(99,99), IP(99)
)69)      DO 10 I=1, N
)70)      IP(I)=I
)71) 10     CONTINUE
)72)      NN=N-1
)73)      DO 50 K=1, NN
)74)      XMAX=A(IP(K), K)
)75)      MP=K
)76)      KK=K+1
)77)      DO 20 I=KK, N
)78)      IF(ABS(A(IP(I), K)) .LE. ABS(XMAX))GOTO 20
)79)      XMAX=A(IP(I), K)
)80)      MP=I
)81) 20     CONTINUE
)82)      II=IP(MP)
)83)      IP(MP)=IP(K)
)84)      IP(K)=II

```

```

(3085)      DO 40 I=KK,N
(3086)      A(IP(I),K)=A(IP(I),K)/A(IP(K),K)
(3087)      DO 30 J=KK,N
(3088)      A(IP(I),J)=A(IP(I),J)-A(IP(I),K)*A(IP(K),J)
(3089)  30   CONTINUE
(3090)  40   CONTINUE
(3091)  50   CONTINUE
(3092)      DO 60 I=1,N
(3093)  70   FORMAT(5(F12.7,3X))
(3094)  60   CONTINUE
(3095)  80   FORMAT(5(I5,5X))
(3096)      RETURN
(3097)      END
(3098)  C
(3099)  C
(3100)  C      ***** SUBROUTINE      38 *****
(3101)  C
(3102)  C
(3103)      SUBROUTINE SOLVE(A,B,X,IP,N)
(3104)      DIMENSION A(99,99),IP(99),B(99),X(99)
(3105)      NN=N-1
(3106)      DO 10 K=1,NN
(3107)      KKK=IP(K)
(3108)      KK=K+1
(3109)      DO 20 I=KK,N
(3110)      III=IP(I)
(3111)      B(III)=B(III)-A(III,K)*B(KKK)
(3112)  20   CONTINUE
(3113)  10   CONTINUE
(3114)  C   WRITE(1,200) (B(I),I=1,N)
(3115)  200  FORMAT('B=',5(F12.7,2X))
(3116)      III=IP(N)
(3117)      X(N)=B(III)/A(III,N)
(3118)      I=N-1
(3119)  40   CONTINUE
(3120)      IS=I+1
(3121)      S=0.0
(3122)      III=IP(I)
(3123)      DO 30 J=IS,N
(3124)      JJJ=IP(J)
(3125)      S=S+A(III,J)*X(J)
(3126)  30   CONTINUE
(3127)      X(I)=(B(III)-S)/A(III,I)
(3128)      I=I-1
(3129)      IF(I.GE.1) GOTO 40
(3130)      DO 13 K=1,N
(3131)      B(K)=X(K)
(3132)  13   CONTINUE
(3133)  C   WRITE(1,200) (B(I),I=1,N)
(3134)  11   RETURN
(3135)      END

```

APPENDIX 2

RUNNING THE PROGRAM

A2.1 PROGRAM INPUT

The program input may be divided into two main parts; one involves the geometry of the aerofoil system and the other the main flow parameters.

A2.1.1 Geometry Definition

The input geometry of an aerofoil is defined by a set of surface points (x_i, z_i) . The coordinates are read into the program beginning at the lower surface trailing edge point. The reading of the data then proceeds along the lower and upper surface of the aerofoil component and ends at the upper surface trailing edge.

The aerofoil geometries may be quite general having:

- a. Arbitrary distributions of camber and thickness
- b. Blunt or sharp trailing edge shape
- c. Flap which overlaps or does not overlap with the main aerofoil.

Sharp discontinuities of the aerofoil surface must be avoided because the potential flow model is not able to handle them.

Depending on whether the linear vorticity (old model) or linear vorticity-constant source model is used, the trailing edge point of each component should be defined once or twice respectively.

The linear vorticity model can only deal with aerofoils having a sharp trailing edge. The linear vorticity-constant source model deals with all types of trailing edge.

Figure A2.1 clarifies the input geometry further. The user is advised to use a cosine distribution for the surface coordinates. The number of panels defining each component may be odd or even if the linear vorticity model is used. However when the new model is used only an even number of panels should be used and panels on the upper and lower surface should have the same x coordinates (see Figure A2.1(a)).

If the total number of points defining the aerofoil system is NMAX, for best results (Ref.7) the following formulae can be used to divide these points between the flap and the main component:

$$N_i = \text{integer of } \left[(N_{MAX} - 21N_T) \frac{C_i}{C_T} \right] + 21$$

where N_i = number of surface points defining aerofoil component i

N_T = number of aerofoil components

C_i = chord length of aerofoil component i

C_T = sum of chord lengths of all components.

A2.1.2 Other Geometry Inputs

C_{BAR} = Aerofoil chord

XM, ZM = Coordinates of the point about which the pitching moment is to be calculated (e.g. 0.25, 0.0)

N_T = Number of aerofoil components, ($N_T=1$ for a single element aerofoil and 2 for a two element aerofoil).

$KN(K)$ = Number of panels on each element.

$XP(K), ZP(K)$ = Coordinates of flap pivot point.

$DELX(K), DELZ(K)$ = $\Delta x, \Delta z$ distances relative to flap pivot representing the horizontal and vertical translation of the flap.

$PHI(K)$ = Flap rotation (i.e. deflection).

The flap coordinates define the flap in the nested position. The program

modifies this coordinate by firstly translating and then rotating then about the flap pivot point.

Examples of the aerofoil geometry definition are given in Tables A2.1 and A2.2.

A2.1.3 Input of Flow Parameters

ALPHA = Angle of attack in deg.

RE1 = Reynolds number based on aerofoil chord.

MO = Mach number.

A2.2 OTHER INSTRUCTIONS

The program allows the following calculations to be performed for one or two element aerofoils.

1. Potential flow analysis
2. Viscous flow analysis
3. Wake potential flow and viscous analysis

A2.2.1 Potential Flow Analysis

This can be carried out with both models although the old model does not allow the use of aerofoils with thick trailing edge. To choose the old model set

IPAR = 0

IP = 9

To choose the new model set

IPAR = 1

A2.2.2 Viscous Flow Analysis

Once again both models allow a viscous analysis although if the old model is used no iteration is allowed and the inviscid pressures are not corrected for viscous effects. The new model allows any number of iterations although

currently it is suggested that one uses a maximum of 8.

NITER = 2,3 etc. up to 8.

The inviscid velocity is corrected for viscosity and the new velocities are used to carry out the new boundary layer analysis.

If the pressures are available from experimental results a boundary layer analysis may be carried out by setting:

IPAR = 0 and

IP = 99

In this case the input file should contain the X,Z coordinates and the corresponding value of CP.

The user may choose between two laminar and two turbulent boundary layer methods, which may be different for the upper and lower surface. Thus if Curles' method is required to calculate both the upper and lower surface boundary layer, the input should be 1 1. If however Curles' method is required for the upper surface analysis while Thwaites method is required for the lower surface analysis then the input should be 1 2. This allows the combination of a method (Curles' method) which gives good results in cases of adverse pressure gradients (e.g. upper surface) and a method (Thwaites' method) which gives a fast solution, but which is more applicable to smoother pressure distributions (e.g. lower surface).

The same applies to the turbulent boundary layer analysis.

Natural transition is predicted by the program. The user may also fix the transition point if it is required. Similarly short bubble separation or turbulent boundary layer separation may be suppressed to allow the analysis to be continued.

A2.2.3 Wake Analysis

The number of wakes is equal to the number of aerofoil elements. Wake analysis is only available with the new model. The wake effects are included

in the pressure coefficient and aerofoil surface velocities, before the next iteration is carried out.

A 2.3 FURTHER COMMENTS

One of the main reasons for keeping the old model is that it can be combined with a flow separation model to allow near-stall and post-stall analysis to be carried out. Also in many cases where viscous effects are not important and the aerofoil has little or no camber the linear vorticity model may be used since it is simpler and requires less time to carry out the potential flow solution.

Examples on how to run the program are given at the end of this Appendix.

Finally the program output is written onto a file named by the user.

The input aerofoil geometry should also be available in a file.

TABLE A2.1: Input file for a two element aerofoil

82	
31	
.96	.01
.92	.017
.9	.0199
.878	.022
.86	.023
.84	.0233
.82	.0224
.8	.0211
.78	.0173
.772	.0157
.76	.0119
.74	.0038
.72	-.0079
.7	-.02
.675	-.033
.65	-.0423
.625	-.0469
.6	-.0508
.575	-.054
.55	-.057
.5	-.061
.45	-.0635
.4	-.0649
.35	-.0652
.3	-.0645
.25	-.0627
.175	-.057
.15	-.0543
.125	-.051
.1	-.047
.075	-.0421
.05	-.0358
.0375	-.0319
.025	-.0269
.0125	-.0205
.005	-.0138
.002	-.0093
0	0
.002	.013
.005	.0204
.0125	.0307
.025	.0417
.0375	.04965
.05	.05589
.075	.06551
.1	.073
.125	.079
.15	.084
.2	.092
.25	.0977
.3	.1016
.35	.104

. 4	. 10491
. 45	. 10445
. 5	. 10258
. 55	. 0991
. 6	. 09371
. 65	. 08599
. 7	. 07634
. 75	. 06513
. 8	. 05286
. 85	. 03988

1	- . 008
. 98	- . 005
. 96	- . 0035
. 94	- . 0028
. 92	- . 0027
. 9	- . 0036
. 88	- . 0058
. 86	- . 0077
. 84	- . 01
. 82	- . 013
. 8	- . 016
. 78	- . 0198
. 76	- . 0235
. 74	- . 027
. 722	- . 0304
. 714	- . 03
. 711	- . 027
. 71	- . 0235
. 712	- . 0179
. 718	- . 0113
. 728	- . 0033
. 738	. 0023
. 758	. 011
. 778	. 0168
. 798	. 0207
. 818	. 0223
. 838	. 023
. 858	. 0228
. 878	. 0219
. 92	. 0168
. 96	. 0098

TABLE A2.2: Input file for a single element aerofoil

74			
1.	-.008	.002	.013
.975	-.004	.005	.0204
.95	-.0026	.0125	.0307
.925	-.0025	.025	.0417
.9	-.0036	.0375	.04965
.875	-.0058	.05	.05589
.85	-.0086	.075	.06551
.825	-.012	.1	.073
.8	-.016	.125	.079
.775	-.0204	.15	.084
.75	-.0249	.175	.0884
.725	-.0294	.2	.092
.7	-.034	.25	.0977
.675	-.0384	.3	.1016
.65	-.0428	.35	.104
.625	-.0469	.4	.10491
.6	-.0508	.45	.10445
.575	-.054	.5	.10258
.55	-.057	.55	.0991
.5	-.061	.575	.09668
.45	-.0635	.6	.09371
.4	-.0649	.625	.09006
.35	-.0652	.65	.08599
.3	-.0645	.675	.08136
.25	-.0627	.7	.07634
.2	-.0593	.725	.07092
.175	-.057	.75	.06513
.15	-.0543	.775	.05907
.125	-.051	.8	.05286
.1	-.047	.825	.04646
.075	-.0421	.85	.03988
.05	-.0358	.875	.03315
.0375	-.0319	.9	.02639
.025	-.0269	.925	.01961
.0125	-.0205	.95	.01287
.005	-.0138	.975	.00609
.002	-.0093	1.	-.0007
.0	.0		

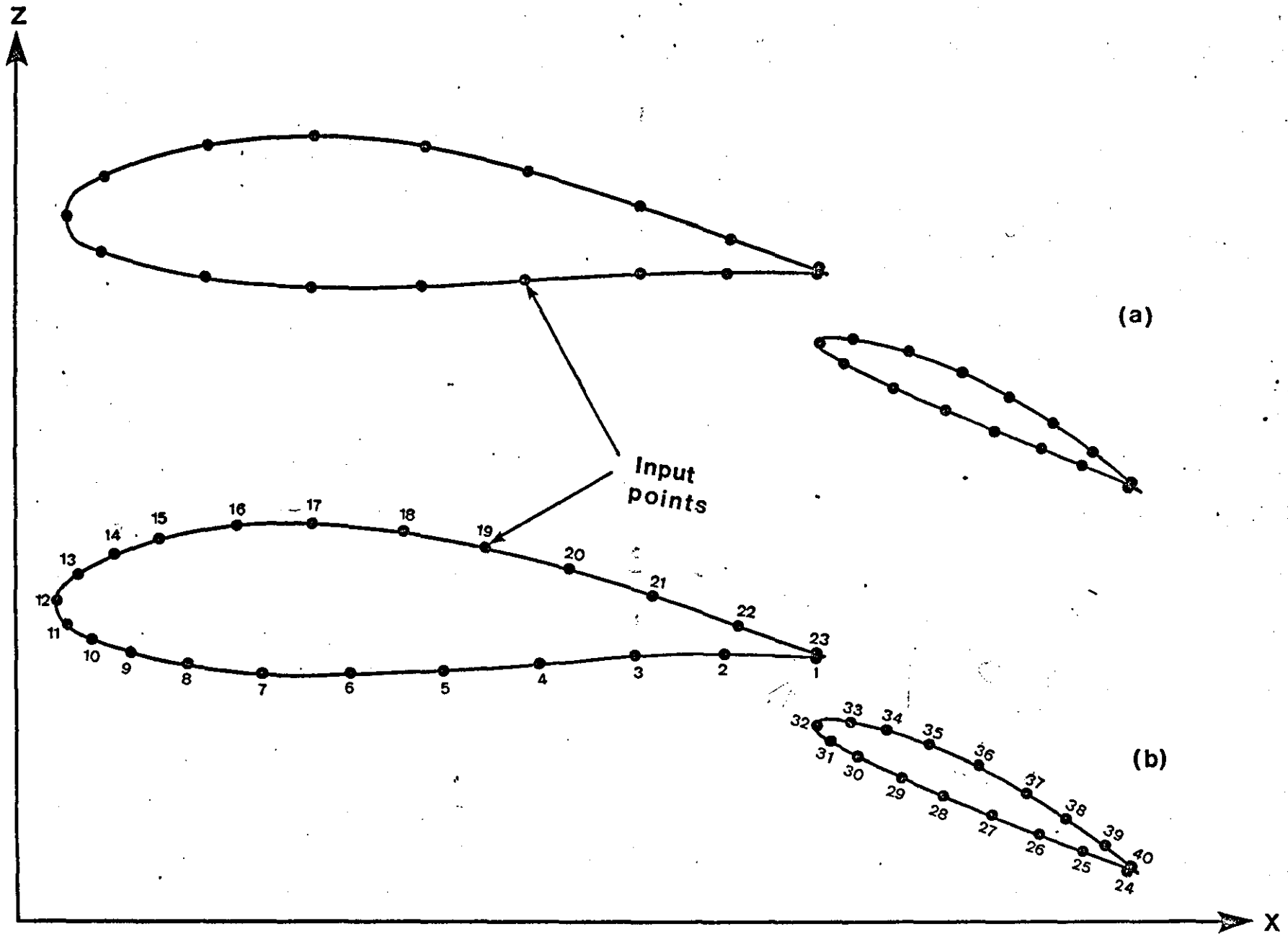


FIGURE A2-1 Example of Geometry Input For the two potential flow models

OK, SEG #MP5134BIG2

PLEASE ENTER TYPE OF ANALYSIS REQUIRED ie 1 or 0

1

TRACE ON 1 *** TRACE OFF 0

0

PLEASE ENTER DATA FILE NAME

GAWF60

PLEASE ENTER OUTPUT FILE NAME

OUT1

PLEASE ENTER ALPHA - CBAR - XM -ZM - MD

5 1. .25 .0 .02

PLEASE ENTER REYNOLDS NUMBER

2.2E06

PLEASE ENTER TOTAL # OF COMPONENTS EG.FLAPS ETC

2

PLEASE ENTER XP(K), ZP(K), DELX(K), DEL(Z), PHI(K)

.73 -.04 .15 -.021 10

ENTER No OF ITERATIONS (NITER) REQUIRED

2(<= NITER <= 8

3

ITERATION No = 1

CAT=-0.17640 CNT= 2.12798 CMT=-0.94607 CL= 2.12798 CD= 0.000

IS VISCOUS ANALYSIS REQUIRED ?

9

CP Trailing Edge

0.416859

0.833899

0.683582

CP Leading Edge

0.938502

0.990590

0.978981

If CP Values Acceptable enter 1 to continue.

If program termination is required enter 0

1

ENTER LAMINAR BOUNDARY LAYER METHOD

	UPPER SURFACE	LOWER SURFACE
Curle	1	1
Thwaites	2	2

1 1

ENTER TURBULENT BOUNDARY LAYER METHOD

	UPPER SURFACE	LOWER SURFACE
Head	1	1
Head-Patel	2	2

1 1

CRABTREE TRANSITION AT NODE 11

T R A N S I T I O N N O D E N O T D E F I N E D

ENTER TRANSITION NODE #,1 TO CONTINUE
OR 0,0 TO STOP

11 1

CRABTREE TRANSITION AT NODE 10

CRABTREE TRANSITION AT NODE 11

TURBULENT SEPARATION AT NODE 13

TURBULENT SEPARATION AT NODE 14

345
TURBULENT SEPARATION AT NODE 15

CP Trailing Edge

0.463644

0.566992

0.536138

CP Leading Edge

0.455513

0.950897

0.724685

If CP Values Acceptable enter 1 to continue.

If programme termination is required enter 0

1

ENTER LAMINAR BOUNDARY LAYER METHOD

	UPPER SURFACE	LOWER SURFACE
Curle	1	1
Thwaites	2	2

2 2

ENTER TURBULENT BOUNDARY LAYER METHOD

	UPPER SURFACE	LOWER SURFACE
Head	1	1
Head-Patel	2	2

2 2

APPENDIX 3

COMPUTER PROGRAM SUBROUTINES

The function of all program subroutines is briefly explained. The subroutines are listed in alphabetical order.

- AIRIC: The routine calculates the influence coefficients of the aerofoil source and vortex singularities on the aerofoil and wake(s).
- BLCALC: This routine provides the necessary geometry and velocities for the boundary layer analysis. It also controls all routines involved in the boundary layer calculation.
- BUBBLE-BUB LE1: The subroutines perform the short bubble analysis.
- CC: Calculates the lift coefficient and pitching moment for the linear vorticity model.
- CNXCLD: Calculates the overall forces and pitching moment for the linear vorticity and constant source model.
- CUBIC: Solves a cubic equation using Newton-Raphson's method.
- CV: Calculates the influence coefficients of the aerofoil vortices on the aerofoil control points (old model).
- DRIVE1: Controls the calculation of the inviscid, viscous and wake potential flow for the new model.
- DRIVE2: Calculates the wake normal and tangential velocities.
- DU1DU2: Finds the first and second derivatives of a set of data.
- HEPA: See TURBL2.
- INPOL-INPOL1-INPOL2: Routines used for linear interpolation.

LAMBL: Calculates the laminar boundary layer using Curles' method.

LAMBL1: Calculates the laminar boundary layer using Thwaites' method.

MAX: Finds the maximum value of a set of data stored in a linear matrix.

MATINV: Finds the inverse of matrix and stores it on top of the original matrix.

MATMUL: Multiplies a rectangular or square matrix with a vector.

QUAD: Finds the equation of a quadratic ($y=ax^2+bx+c$) from a set of three (x,y) values.

PLOT-PLOT1: Routines used for plotting data.

PO2001: Calculates a set of geometrical parameters for the aerofoil and its wake.

SPLINE: Fits cubic splines to a set of data for interpolation or differentiation.

TRANS: Estimates the point of laminar boundary layer transition.

TURBL1: Calculates the turbulent boundary layer using Head's method.

TURBL2: Calculates the turbulent boundary layer using the method of Head and Patel.

WAKEG1: Generates the geometry of the wake of the main aerofoil component.

WAKEG2: Generates the geometry of the wake of the flap/single element aerofoil.

WAKEIC: Finds the influence coefficients of the wake singularities on the wake/aerofoil control points.

WAKREL: Relaxes the wake.

WAKVIS: Performs a wake viscous analysis on one side of the wake singularities sheet, using Green's lag entrainment method.

WIW2SS: Calculates the wake singularity strength at the panel midpoint using the results of the wake viscous analysis.

W1W2S1: Calculates the wake source and vortex singularity strengths at the end of the wake panels using the values at the panel midpoints.

APPENDIX 4

DATA SMOOTHING - MATRIX INVERSION

A4.1 DATA SMOOTHING

Some of the variables have to be smoothed out during the analysis before they can be differentiated or used for any other purpose.

The smoothing of a set of data is accomplished by use of a standard computer subroutine namely NAG-EO2BAF. The routine computes a weighted least-squares approximation to an arbitrary set of data by a cubic spline with knots prescribed by the programme. The resulting function $S(x)$ can then be differentiated using the subroutine NAG-EO2BCF. This routine evaluates the cubic spline $S(x)$ and its first three derivatives at prescribed arguments x .

When choosing the knots care should be taken so that oscillations of the function $S(x)$ are avoided. Knots divide the group of data into subgroups, and the routine fits a spline to each subgroup. At the knot the function and its first two derivatives are continuous. Discontinuities may be introduced by use of multiple knots. Thus four knots at the same x_1 position make $S(x)$ discontinuous at x_1 , three knots make $\frac{dS(x)}{dx}$ discontinuous at x_1 , two knots make $\frac{d^2S(x)}{dx^2}$ discontinuous at x_1 . The routine inserts automatically four knots at the smallest and largest x arguments.

The knots should be grouped more closely in regions where the function (underlying the data) or its derivatives change more rapidly than elsewhere. Figures (1 to 3) illustrate some of the points mentioned above.

A4.2 MATRIX INVERSION

Matrix inversion is carried out using Gauss elimination with partial pivoting (Ref.21). The algorithm is programmed to map the inverse of matrix $[A]$ onto $[A]$ such that only one matrix is required for the inversion. This saves a lot of memory space especially when dealing with large matrices. A listing of the subroutine that performs the inversion is given below.


```

IFLAG=1
20 CONTINUE
IF(IFLAG.EQ.0)GOTO 30
IX=IP(K)
IP(K)=IP(MPIVOT)
IP(MPIVOT)=IX
DO 25 J=1,N
DUMMY=A(K,J)
A(K,J)=A(MPIVOT,J)
A(MPIVOT,J)=DUMMY
25 CONTINUE
30 CONTINUE
PIVOT(K)=A(K,K)
ZPIVOT=A(K,K)
IF(ABS(PIVOT(K)).LT.TOLER)GOTO 120
A(K,K)=1.
KK=K+1
DO 45 I=KK,N
FACT=A(I,K)/PIVOT(K)
A(I,K)=0.0
DO 35 J=KK,N
A(I,J)=A(I,J)-FACT*A(K,J)
35 CONTINUE
DO 40 J=1,K
A(I,J)=A(I,J)-FACT*A(K,J)
40 CONTINUE
45 CONTINUE
47 CONTINUE
PIVOT(N)=A(N,N)
ZPIVOT=A(N,N)
IF(ABS(PIVOT(N)).LT.TOLER)GOTO 120
A(N,N)=1.
DO 50 J=1,N
A(N,J)=A(N,J)/PIVOT(N)
50 CONTINUE
K=N-1
60 CONTINUE
DO 70 J=1,N
Y(J)=0.0
IF(J.LE.K)GOTO 70
Y(J)=A(K,J)
A(K,J)=0.0
70 CONTINUE
DO 90 J=1,N
S=0.0
KS=K+1
DO 80 JJ=KS,N
S=S+A(JJ,J)*Y(JJ)
80 CONTINUE
A(K,J)=(A(K,J)-S)/PIVOT(K)
90 CONTINUE
K=K-1
IF(K.GE.1)GOTO 60
DO 97 J=1,N

```

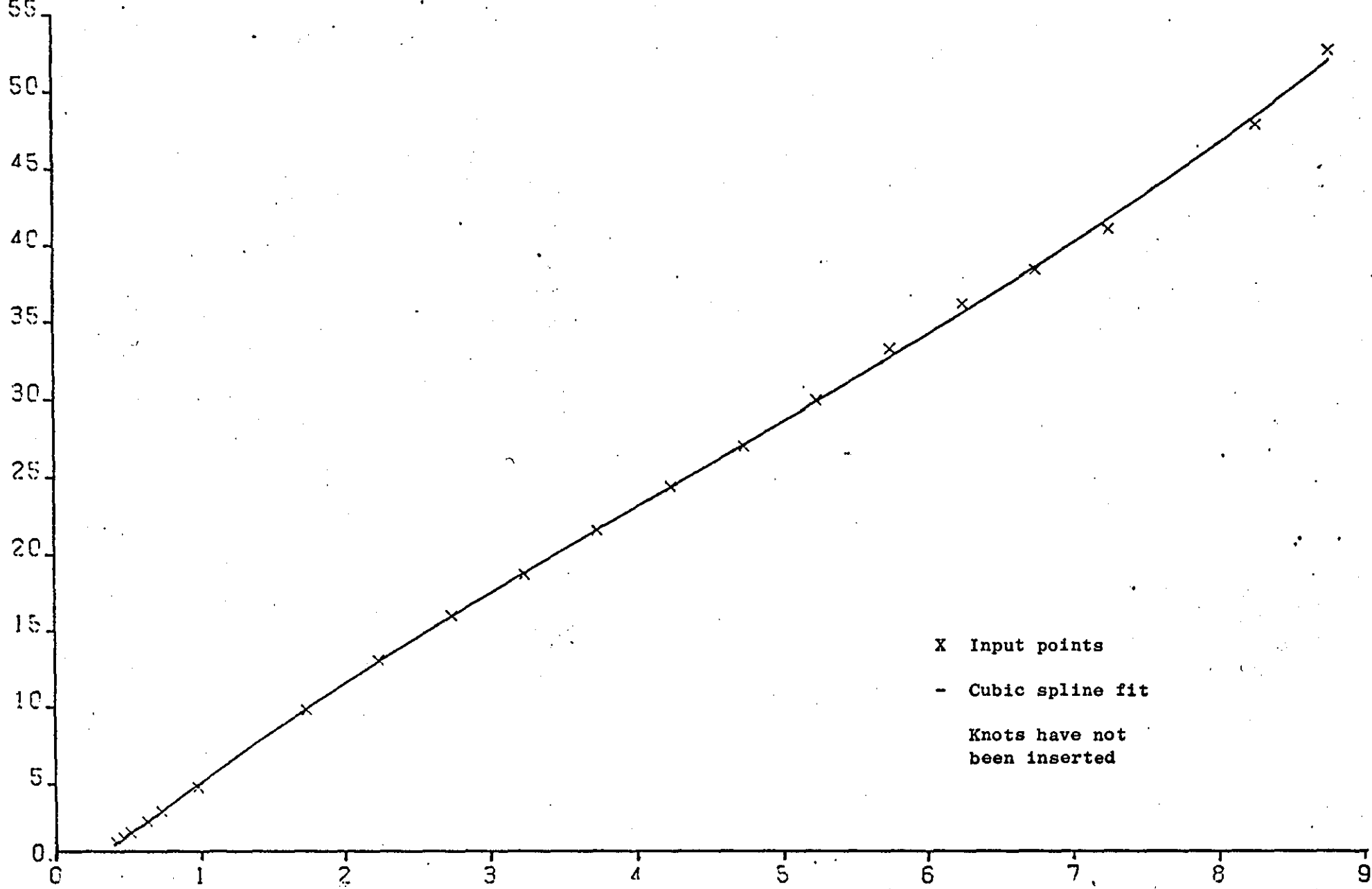
```

IF(IP(J).EQ.J)GOTO 97
JJ=IP(J)
DO 95 I=1,N

DUMMY=A(I,J)
A(I,J)=A(I,JJ)
A(I,JJ)=DUMMY
95 CONTINUE
IP(J)=J
IP(JJ)=JJ
97 CONTINUE
WRITE(1,200)
200 FORMAT(/,'THE INVERSE OF THE MATRIX A',/)
DO 100 I=1,N
WRITE(1,202)I
202 FORMAT(/,'          ROW NO. =',I4/)
WRITE(1,204)(A(I,J),J=1,N)
100 CONTINUE
204 FORMAT(5(F12.7,3X))
RETURN
120 CONTINUE
WRITE(1,206)ZPIVOT
206 FORMAT(/,'THE MATRIX IS SINGULAR THE CURRENT PIVOTAL
ELEMENT=',F15.7)
RETURN
END
BOTTOM

```

$\times 10^{-4}$



x Input points
- Cubic spline fit
Knots have not
been inserted

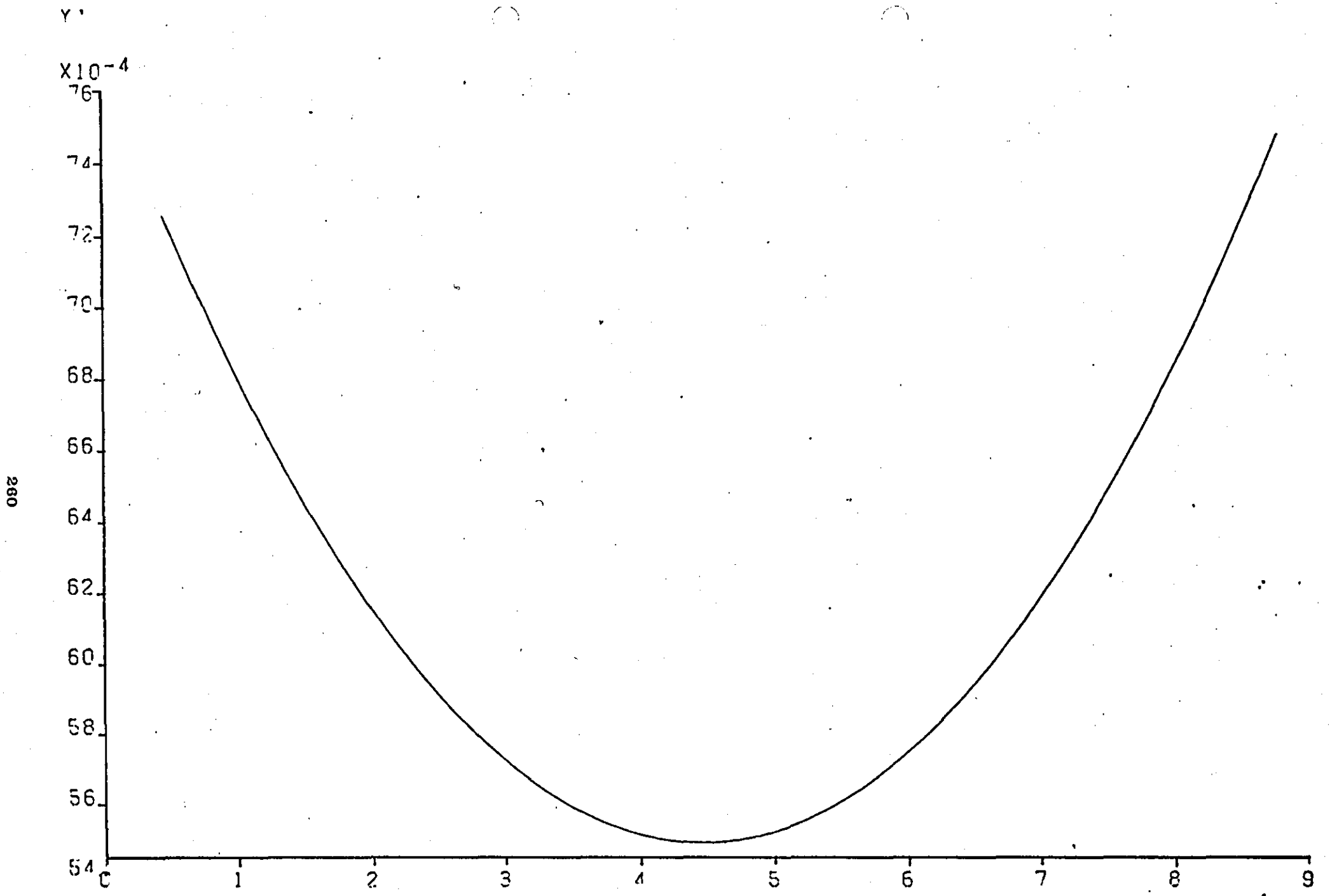


FIGURE A4-1 (b) • First derivative obtained from ... $\times 10^{-1}$

Y

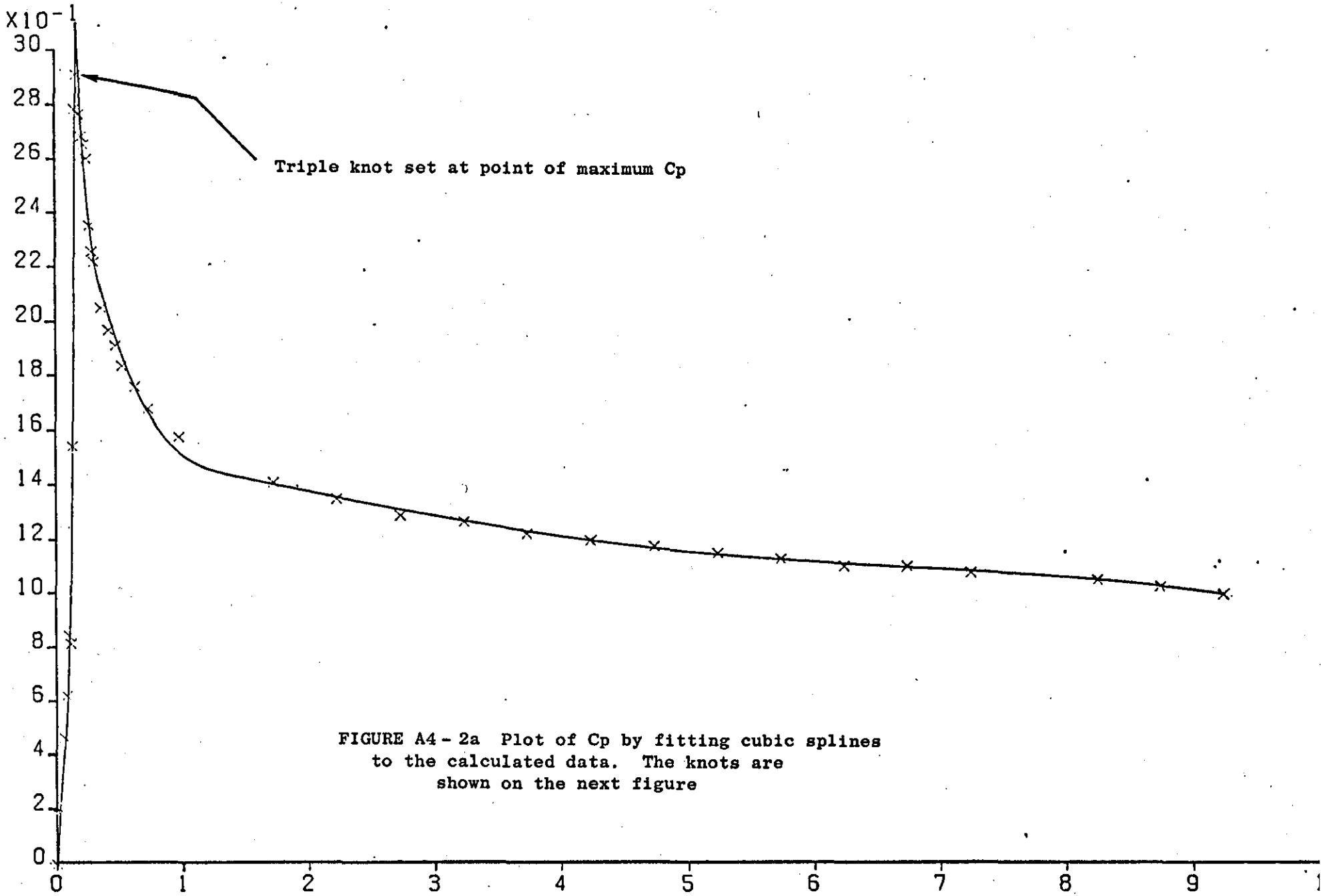
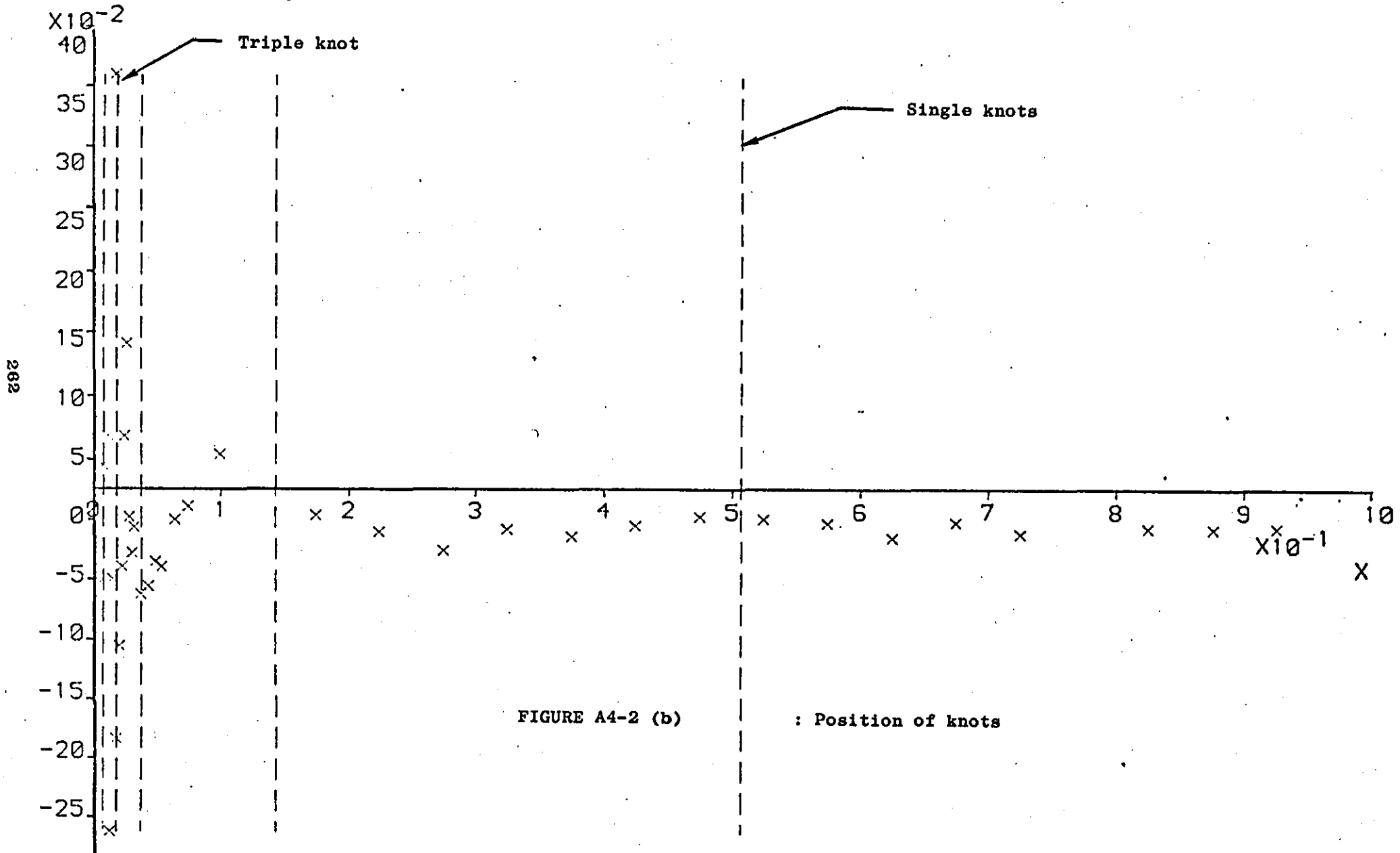


FIGURE A4-2a Plot of Cp by fitting cubic splines to the calculated data. The knots are shown on the next figure

Residual sum squares= .320E 0

Residuals



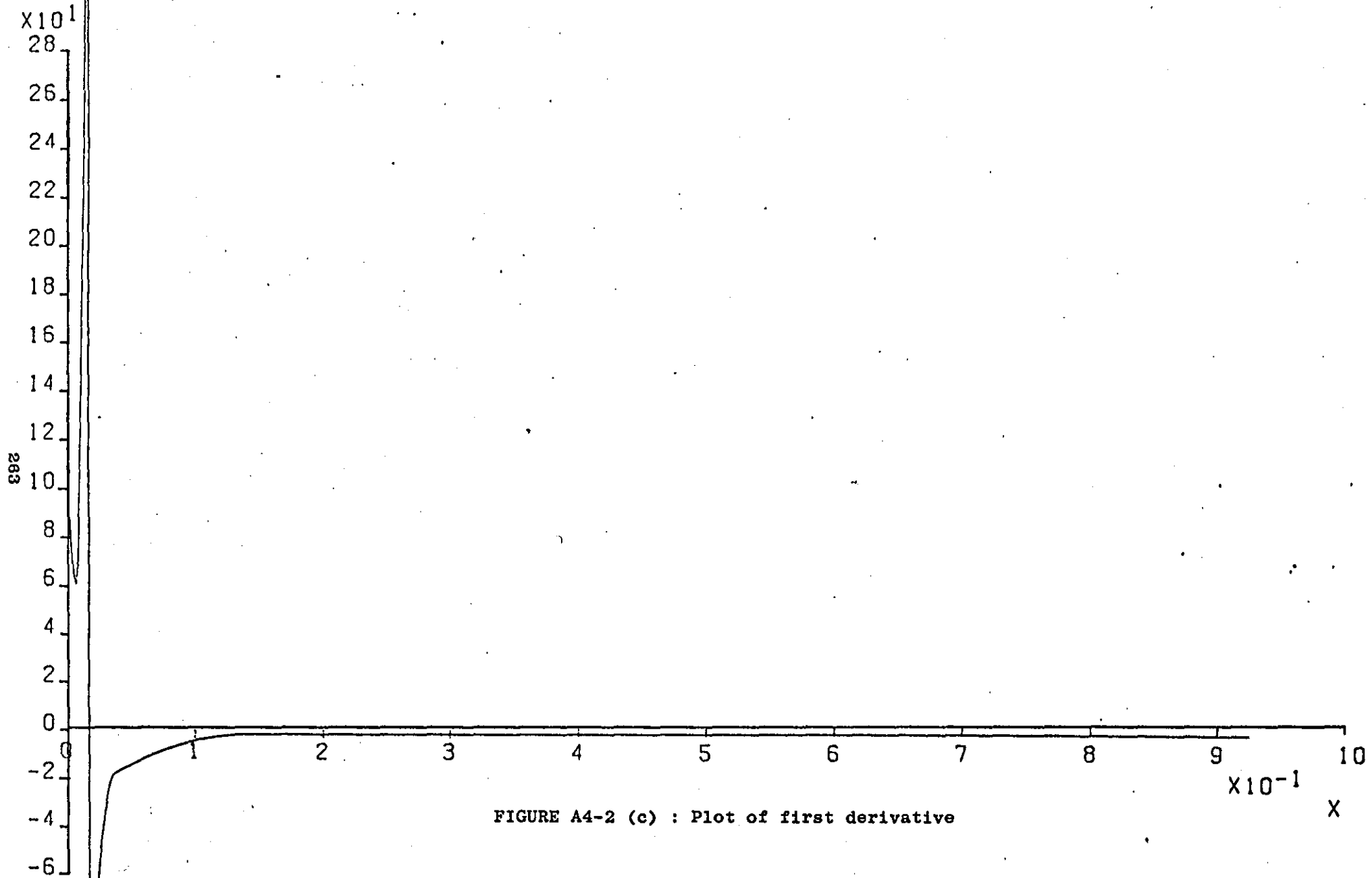
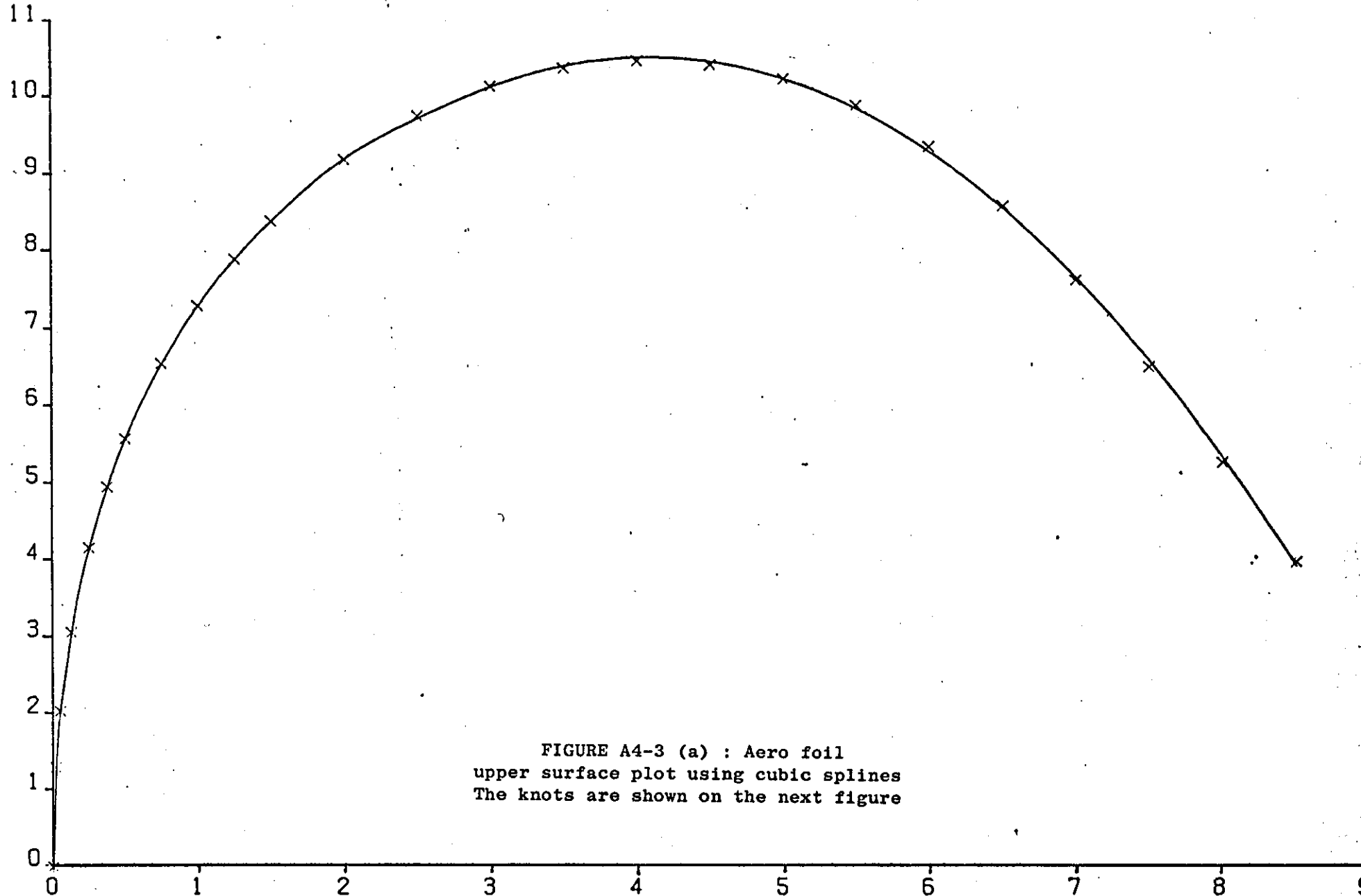


FIGURE A4-2 (c) : Plot of first derivative

Y

$\times 10^{-2}$



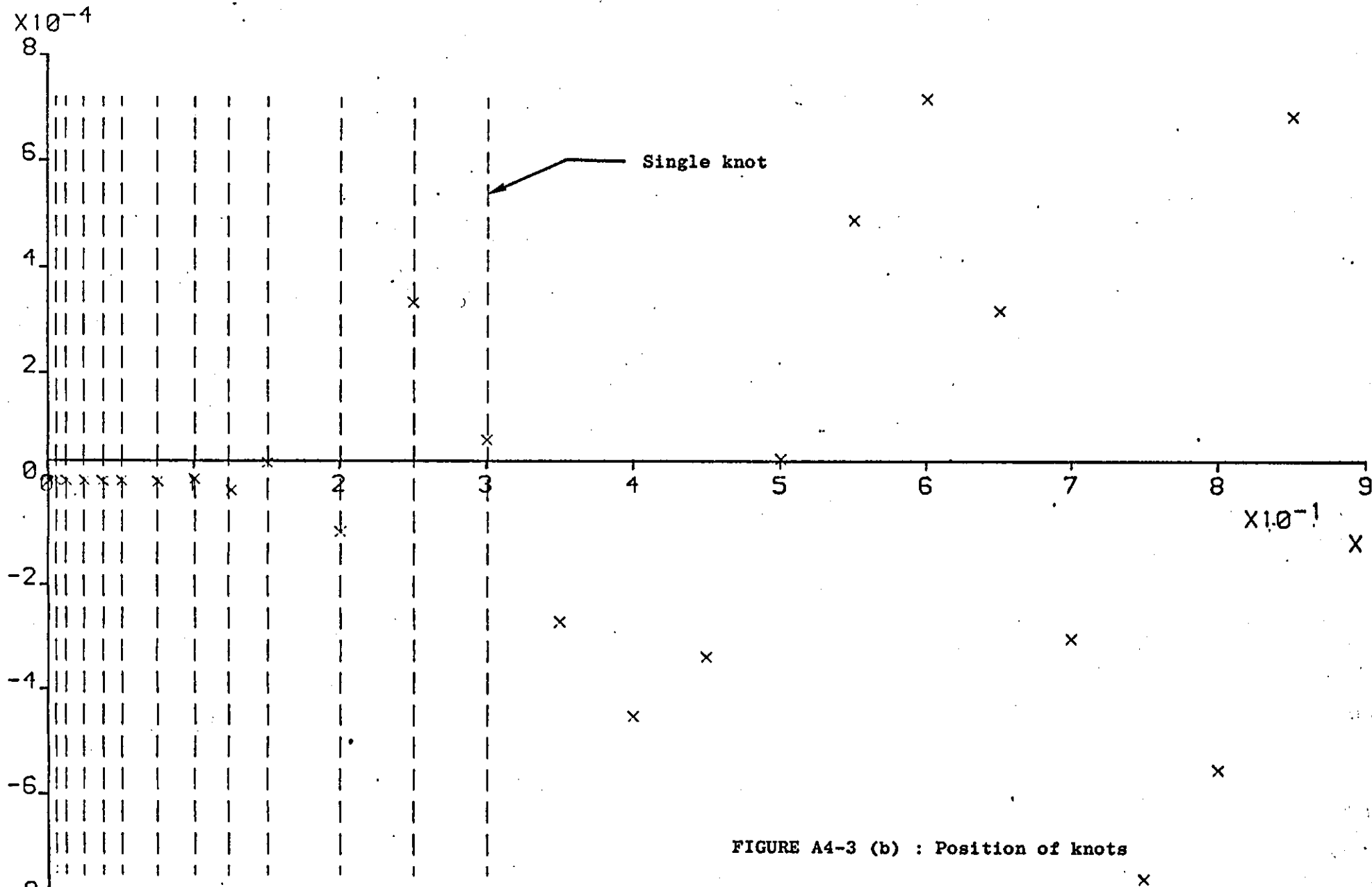
264

FIGURE A4-3 (a) : Aero foil
upper surface plot using cubic splines
The knots are shown on the next figure

x10^-1

Residual sum squares = .281E -5

Residuals



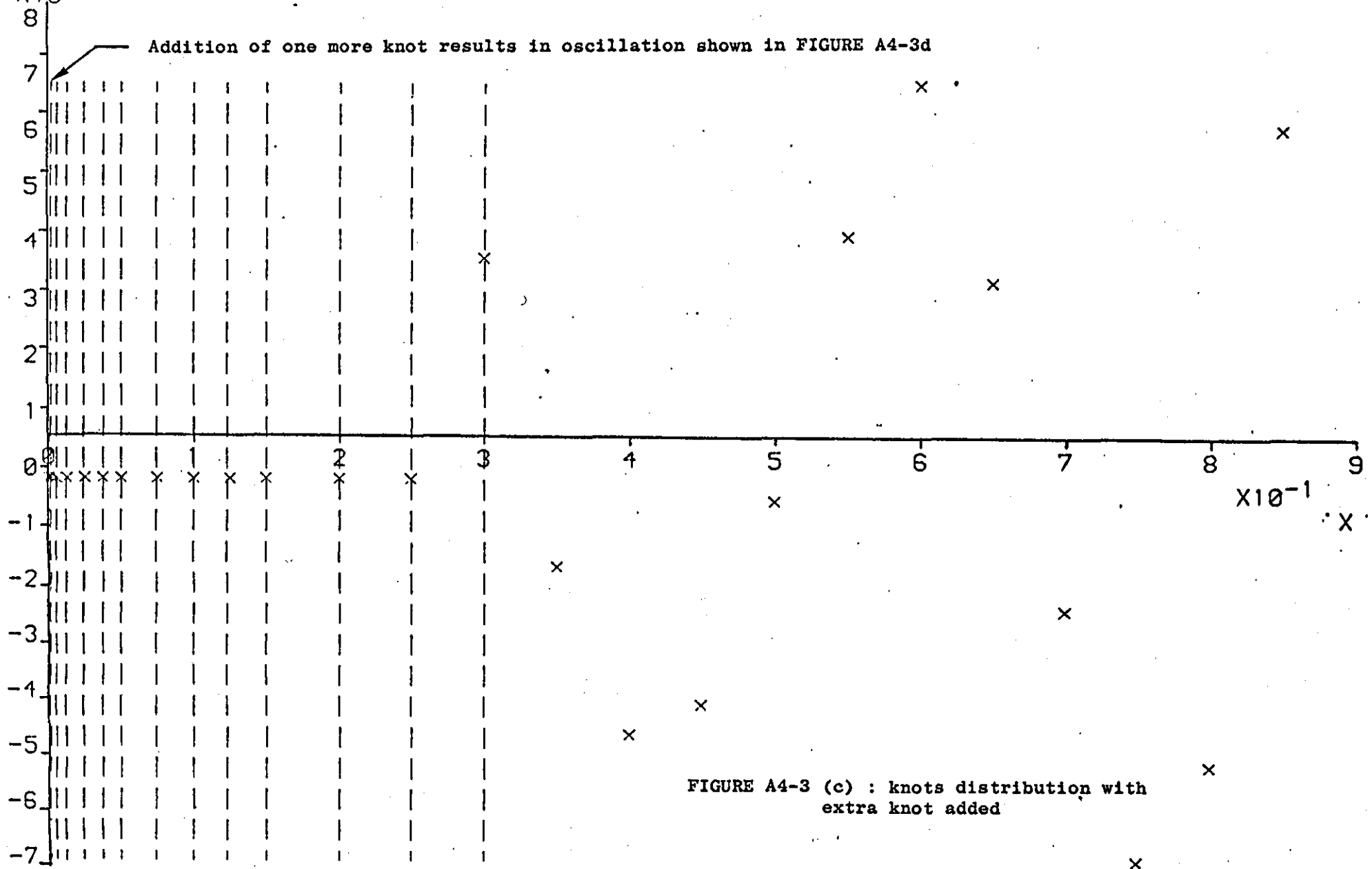
265

FIGURE A4-3 (b) : Position of knots

Residual sum squares = .255E -5

Residuals

$\times 10^{-4}$



288

FIGURE A4-3 (c) : knots distribution with extra knot added

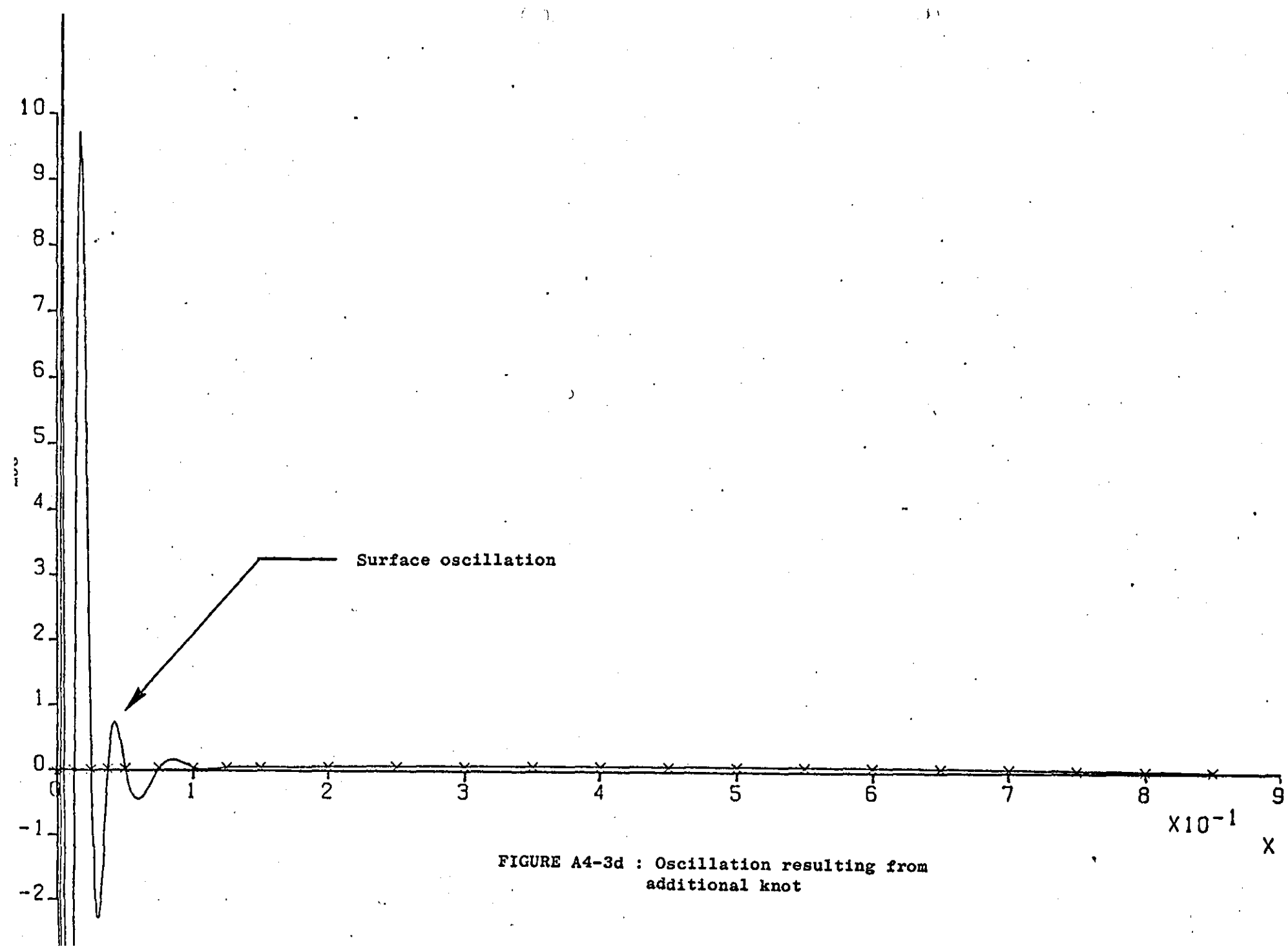


FIGURE A4-3d : Oscillation resulting from additional knot

APPENDIX 5

METHODS FOR SOLVING A SET OF LINEAR EQUATIONS

A5.1 SUCCESSIVE ORTHOGONALISATION METHOD

Let the set of linear equations be

$$\begin{array}{rcl}
 j=1 & a_{11}x_1 + a_{12}x_2 + \dots + a_{1N}x_N + b_1 & = 0 \\
 & \vdots & \\
 j=N & a_{N1}x_1 + a_{N2}x_2 + \dots + a_{NN}x_N + b_N & = 0
 \end{array} \tag{A5.1}$$

where a_{jk} is the coefficient of variable x_k in equation j ($j=1,2,\dots,N$, $k=1,2,\dots,N$).

In Purcell's (91) method the j^{th} equation can be represented by

$$\underline{c}_j \cdot \underline{d} = 0$$

where

$$\begin{aligned}
 \underline{c}_j &= (a_{j1}, a_{j2}, a_{j3}, \dots, a_{jN}, b_j) \\
 \underline{d} &= (x_1, x_2, x_3, \dots, x_N, 1)
 \end{aligned} \tag{A5.2}$$

\underline{c} and \underline{d} being vectors with components as shown above.

The dot product of two vectors is zero if the vectors are orthogonal.

Thus the solution may be found by constructing a vector \underline{d} which is perpendicular to all \underline{c}_j vectors.

The method begins by forming a set of linearly independent vectors

$$\underline{v}_i^1 = \quad (i=1,2,\dots,N)$$

which are orthogonal to \underline{c}_1 . Each of the \underline{v}_i^1 vectors has $N+1$ components and has the form

$$\begin{aligned}
 \underline{v}_1^1 &= (v_{11}^1, 1, 0, \dots, 0) \\
 \underline{v}_2^1 &= (v_{12}^1, 0, 1, \dots, 0) \\
 &\vdots \\
 \underline{v}_N^1 &= (v_{1N}^1, 0, 0, \dots, 1)
 \end{aligned} \tag{A5.3}$$

where v_{ji}^k is the k^{th} component of vector \underline{v}_i^j which is orthogonal to the vector \underline{c}_j whose components are the coefficients of equation j .

Now set

$$\begin{aligned} \underline{v}_i^1 \cdot \underline{c}_1 &= 0 && \text{this gives} \\ v_{1i}^1 a_{11} + a_{1,i+1} &= 0 \text{ or} \\ v_{1i}^1 &= -\frac{a_{1,i+1}}{a_{11}} \end{aligned} \quad (\text{A5.4})$$

The components v_{1i}^1 are stored at this stage. Next, $N-1$ linearly independent vectors \underline{v}_i^2 are constructed from \underline{v}_i^1 . These have the form,

$$\underline{v}_i^2 = B_{i-1}^2 \underline{v}_{i-1}^1 + \underline{v}_{i+1}^1 \quad (\text{A5.5})$$

The new vectors have components

$$\begin{aligned} \underline{v}_1^2 &= (v_{21}^1, v_{21}^2, 1, 0, \dots, 0) \\ \underline{v}_2^2 &= (v_{22}^1, v_{22}^2, 0, 1, \dots, 0) \\ &\vdots \\ \underline{v}_{N-1}^2 &= (v_{2,N-1}^1, v_{2,N-1}^2, 0, 0, \dots, 1) \end{aligned}$$

where

$$\begin{aligned} v_{2i}^1 &= B_{i-1}^2 v_{i-1}^1 + v_{1,i+1}^1 \\ v_{2i}^2 &= B_i^2 \end{aligned} \quad (\text{A5.6})$$

Now let \underline{v}_i^2 be orthogonal to \underline{c}_2 i.e. set

$$\underline{v}_i^2 \cdot \underline{c}_2 = 0$$

this gives

$$v_{2i}^1 a_{21} + v_{2i}^2 a_{22} + a_{2,i+2} = 0$$

$$\therefore (B_{i-1}^2 v_{i-1}^1 + v_{1,i+1}^1) a_{21} + B_i^2 a_{22} + a_{2,i+2} = 0$$

from which

$$B_i^2 = -\frac{a_{2,i+2} + a_{21} v_{1,i+1}^1}{a_{22} + a_{21} v_{11}^1}, \quad (i=1, 2, \dots, N) \quad (\text{A5.7})$$

and hence v_{2i}^1 and v_{2i}^2 can be calculated from equation (A5.6).

The new vectors \underline{v}_i^2 are orthogonal to both \underline{c}_1 and \underline{c}_2 and only the components v_{2i}^1 and v_{2i}^2 are needed for the next stage of the calculation (v_{1i}^1 are no longer required).

This process is repeated for all c_j ; the final vector \underline{v}_1^N contains the solution for \underline{d} .

$$\underline{v}_1^N = (v_{N1}^1, v_{N1}^2, v_{N1}^3, \dots, v_{N1}^N, 1)$$

At the j^{th} stage there are j sets of components to be stored, each of length $N+1-j$. This number reaches a maximum in the middle of the matrix when it has a value

$$\frac{N(N+2)}{4}$$

Since at each stage only the previous set of components $v_{1i}^1, v_{2i}^1, v_{2i}^2$, etc. are needed, these components may be conveniently stored in a single linear array [D] of maximum size given by $\frac{N(N+2)}{4}$, whose elements are updated after each row has been operated on.

As shown above only the coefficients of the j^{th} equation are required at a particular stage so these can be stored in a linear matrix [A] of size $N+1$.

For the first equation (i.e. for $j=1$) the matrix [D] contains the components of all vectors \underline{v}_1^1 orthogonal to \underline{c}_1 .

Thus for $j=1$

$$D(I) = - \frac{A(I+1)}{A(1)} \quad (\text{A5.8})$$

For $j>1$ the following equations apply (70):

$$B(I) = - \frac{\text{NUM}}{\text{DEN}} \quad (\text{A5.9})$$

where

$$\text{NUM} = [A(I+J) + \sum_{K=1}^{J-1} A(K) \times D(M+I)]$$

$$\text{DEN} = A(J) + \sum_{K=1}^{J-1} A(K) \times D(M)$$

$$I=1, 2, \dots, N+1-J$$

$$M=1+(K-1)(N+2-J)$$

Once $B(I)$ has been found the new set of components of matrix [D] is given by

$$D(I+(K-1)(N+1-J)) = B(I) \times P + D(I+1+(K-1)(N+2-J))$$

where $P=D(1+(K-1)(N+2-J))$
 $I=1,2,\dots,N+1-J$
 $K=1,2,\dots,J-1.$

The inner loop represents the components down each column i.e. the v_{21}^1, v_{21}^1 , etc. (see equation A5.6); the outer loop represents each complete column except the last one at a given stage. The last column of stored components is given by

$$D((J-1)(N+1-J)+I) = B(I)$$

$$I=1,2,\dots,N+1-J$$

When all N of the rows of the matrix have been processed then the first N elements of $[D]$ is the solution vector X .

The Fortran coding is given below on page

A5.2 GAUSS ELIMINATION WITH PARTIAL PIVOTING

For linear systems with a small number of equations the method of Gauss with partial pivoting (21) is used. The Fortran coding for this method is also given below.

```

DIMENSION A(10,10),B(10),X(10)
READ(1,*)N
DO 10 I=1,N
READ(1,*)B(I), (A(I,J), J=1,N)
10 CONTINUE
CALL GAUSS2(A,B,X,N)
DO 15 I=1,N
WRITE(1,200)X(I)
200 FORMAT(F16.6)
15 CONTINUE
CALL EXIT
END

C
C
C
SUBROUTINE GAUSS2(A,B,X,N)
DIMENSION A(10,10),B(10),X(10)
NN=N-1
DO 10 K=1,NN
XMAX=0.0
IFLAG=0
DO 20 I=K,N
IF(ABS(A(I,K)).LT.ABS(XMAX))GOTO 20
IFLAG=1
XMAX=A(I,K)
MPIVOT=I
20 CONTINUE
IF(IFLAG.EQ.0)GOTO 40
DO 30 J=1,N
Y=A(K,J)
A(K,J)=A(MPIVOT,J)
A(MPIVOT,J)=Y
30 CONTINUE
Y=B(K)
B(K)=B(MPIVOT)
B(MPIVOT)=Y
40 CONTINUE
KK=K+1
DO 50 I=KK,N
A(I,K)=A(I,K)/A(K,K)
DO 60 J=KK,N
A(I,J)=A(I,J)-A(I,K)*A(K,J)
60 CONTINUE
B(I)=B(I)-A(I,K)*B(K)
50 CONTINUE
10 CONTINUE
X(N)=B(N)/A(N,N)
K=N-1
80 KS=K+1
S=0.0
DO 70 J=KS,N
S=S+A(K,J)*X(J)
70 CONTINUE
X(K)=(B(K)-S)/A(K,K)

```

```
K=K-1  
IF(K. GE. 1)GOTO 90  
RETURN  
END  
BOTTOM
```

APPENDIX 6

VELOCITY INDUCED BY PARABOLIC VORTICITY

DISTRIBUTION ON LINEAR PANEL

A6.1 EQUATION OF INDUCED VELOCITY

Let $\gamma(\alpha) = c_1 + c_2\alpha + c_3\alpha^2$ be the circulation per unit length of the segment ($0 \leq \alpha \leq 1$). Let the straight line segment be of length s and have unit tangential and normal vectors \underline{t} and \underline{n} respectively (DA6.1) and let the

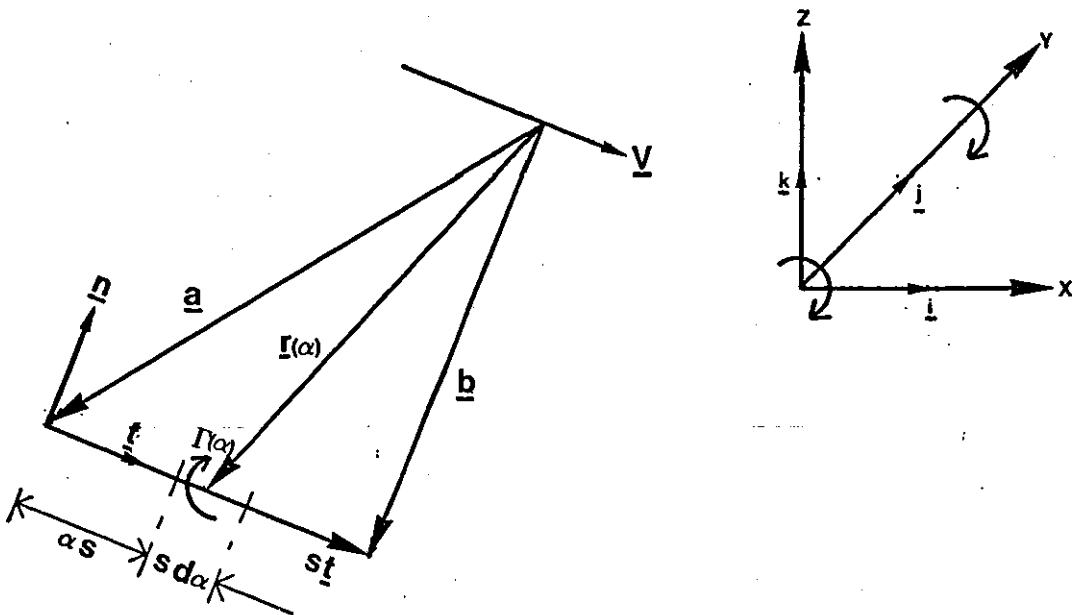


Diagram A6-1

circulation on the element $s d\alpha$ be of strength $\Gamma(\alpha) = \gamma(\alpha) s d\alpha$. Thus,

$$\underline{\Gamma}(\alpha) = \gamma(\alpha) s d\alpha \underline{j} \quad (\text{A6.1})$$

The velocity induced by the circulation on the element $s d\alpha$ is given by

$$d\underline{v} = \frac{\underline{\Gamma}(\alpha)}{2\pi} \frac{\underline{r}(\alpha) \wedge \underline{j}}{(\underline{r}(\alpha))^2} = \frac{1}{2\pi} \frac{\gamma(\alpha) s d\alpha \underline{r}(\alpha) \wedge \underline{j}}{(\underline{r}(\alpha))^2} \quad (\text{A6.2})$$

From vector algebra $\underline{r}(\alpha) = \underline{a} + \alpha s \underline{t}$ and if $\underline{a}' = \underline{a}/s$ then,

$$\underline{r}(\alpha) = s(\underline{a}' + \alpha \underline{t}) \quad (\text{A6.3})$$

Substituting the above in the equation for $d\underline{v}$ gives

$$\underline{dV} = -\frac{j}{2\pi} \frac{s(\underline{a}' + \underline{t}\underline{\alpha})(c_1 + c_2\underline{\alpha} + c_3\underline{\alpha}^2) s d\underline{\alpha}}{s(\underline{a}' + \underline{\alpha}\underline{t}) \cdot s(\underline{a}' + \underline{t}\underline{\alpha})}$$

$$\therefore \underline{dV} = -\frac{j}{2\pi} \frac{(\underline{a}' + \underline{t}\underline{\alpha})(c_1 + c_2\underline{\alpha} + c_3\underline{\alpha}^2) d\underline{\alpha}}{(\underline{a}' \cdot \underline{a}' + 2\underline{\alpha}\underline{a}' \cdot \underline{t} + \underline{\alpha}^2)} \quad (\text{A6.4})$$

and integrating between $\alpha=0$ and 1

$$\underline{V} = -\frac{1}{2\pi} j \int_0^1 \frac{c_1 \underline{a}' + (c_1 \underline{t} + c_2 \underline{a}') \underline{\alpha} + (c_2 \underline{t} + c_3 \underline{a}') \underline{\alpha}^2 + c_3 \underline{t} \underline{\alpha}^3}{(\underline{a}' \cdot \underline{a}' + 2\underline{\alpha}\underline{a}' \cdot \underline{t} + \underline{\alpha}^2)} d\underline{\alpha}$$

$$\therefore \underline{V} = -\frac{1}{2\pi} j [c_1 \underline{a}' I_1 + (c_1 \underline{t} + c_2 \underline{a}') I_2 + (c_2 \underline{t} + c_3 \underline{a}') I_3 + c_3 \underline{t} I_4] \quad (\text{A6.5})$$

and after collecting c_1, c_2 and c_3 terms together,

$$\underline{V} = \frac{1}{2\pi} [c_1 \underline{A}_{-1} + c_2 \underline{A}_{-2} + c_3 \underline{A}_{-3}] \quad (\text{A6.6})$$

which is the velocity induced by a parabolic vorticity distribution on a linear segment.

Taking $c_3=0$ then $\gamma(\alpha)=c_1+c_2\alpha$ which is a linear vorticity distribution.

In this case the velocity is given by

$$\underline{V} = \frac{1}{2\pi} [c_1 \underline{A}_{-1} + c_2 \underline{A}_{-2}] \quad (\text{A6.7})$$

Finally if $c_2=0$ then $\gamma(\alpha)=c_1$. Thus for uniform vorticity distribution on a linear segment

$$\underline{V} = \frac{1}{2\pi} c_1 \underline{A}_{-1} \quad (\text{A6.8})$$

A6.2 EVALUATION OF $\underline{A}_{-1}, \underline{A}_{-2}, \underline{A}_{-3}$

The values of $\underline{A}_{-1}, \underline{A}_{-2}$ and \underline{A}_{-3} are related to the values of I_1, I_2, I_3 and I_4 in (A6.5) through the following equations:

$$\begin{aligned} \underline{A}_{-1} &= (\underline{a}' I_1 + \underline{t} I_2) \wedge \underline{j} \\ \underline{A}_{-2} &= (\underline{a}' I_2 + \underline{t} I_3) \wedge \underline{j} \\ \underline{A}_{-3} &= (\underline{a}' I_3 + \underline{t} I_4) \wedge \underline{j} \end{aligned} \quad (\text{A6.9})$$

where

where

$$\left. \begin{aligned} I_1 &= \int_0^1 \frac{d\alpha}{\underline{a}' \cdot \underline{a} + 2\alpha \underline{a}' \cdot \underline{t} + \alpha^2} \\ I_2 &= \int_0^1 \frac{\alpha d\alpha}{\underline{a}' \cdot \underline{a}' + 2\alpha \underline{a}' \cdot \underline{t} + \alpha^2} \\ I_3 &= \int_0^1 \frac{\alpha^2 d\alpha}{\underline{a}' \cdot \underline{a}' + 2\alpha \underline{a}' \cdot \underline{t} + \alpha^2} \\ I_4 &= \int_0^1 \frac{\alpha^3 d\alpha}{\underline{a}' \cdot \underline{a}' + 2\alpha \underline{a}' \cdot \underline{t} + \alpha^2} \end{aligned} \right\} \quad (\text{A6.10})$$

After integration of the above integrals

$$I_1 = \frac{1}{|\underline{a}' \cdot \underline{n}|} \left[\tan^{-1} \left(\frac{1 + \underline{a}' \cdot \underline{t}}{|\underline{a}' \cdot \underline{n}|} \right) - \tan^{-1} \left(\frac{\underline{a}' \cdot \underline{t}}{|\underline{a}' \cdot \underline{n}|} \right) \right]$$

$$I_2 = L \underline{a}' \cdot \underline{t} I_1$$

where

$$L = \ln \left(\frac{\underline{a}' \cdot \underline{a}' + 2\underline{a}' \cdot \underline{t} + 1}{\underline{a}' \cdot \underline{a}'} \right)^{1/2} = \ln \left| \frac{b}{a} \right|$$

$$I_3 = 1 - 2a'_t L + (a'_t{}^2 - a'_n{}^2) I_1$$

where

$$a'_t = \underline{a}' \cdot \underline{t} \quad , \quad a'_n = \underline{a}' \cdot \underline{n}$$

$$I_4 = \frac{1}{2} - 2a'_t I_3 - a'^2 I_2$$

where

$$a'^2 = \underline{a}' \cdot \underline{a}'$$

(for a more detailed proof see Ref.70, pp.355-357).

Substituting the values of I_1, I_2, I_3, I_4 into the equations for $\underline{A}_1, \underline{A}_2$ and \underline{A}_3 the following expressions are obtained:

$$\underline{A}_1 = -T\underline{t} + L\underline{n}$$

where

$$T = a'_n I_1$$

$$\underline{A}_2 = (Ta'_t - La'_n)\underline{t} + (1 - Ta'_n - La'_t)\underline{n}$$

$$\underline{A}_3 = (Te + Lg - a'_n)\underline{t} + (Tg - Le + \frac{1}{2}a'_t)\underline{n}$$

where

$$e = a'_n{}^2 - a'_t{}^2$$

$$g = 2a'_t a'_n$$

(A6.12)

A6.3 LIMITING VALUES

It was shown (A6.12) that:

$$\begin{aligned}
 T &= a'_n I_{n1} \\
 \therefore T &= \frac{a'_n}{|a'_n|} \left(\tan^{-1} \frac{1+a'_t}{|a'_n|} - \tan^{-1} \frac{a'_t}{|a'_n|} \right) \\
 &= (\text{sign of } a'_n) \left(\tan^{-1} \frac{1+a'_t}{|a'_n|} - \tan^{-1} \frac{a'_t}{|a'_n|} \right)
 \end{aligned}
 \tag{A6.13}$$

Clearly, a limiting value for T has to be found when $a'_n \rightarrow 0$ and also the limiting value of L has to be considered as a or b tend to zero. In either case $\underline{a'_n}$ is zero, thus T is also singular.

A6.3.1 Case 1: Limiting Value for T

In considering the limiting value of T the value of a'_t must be taken into account. Table A6.1 gives all possible cases.

TABLE A6.1

Range for $a'_t = \frac{a}{s} \cdot \underline{t}$	Limit of $T = a'_n I_{n1}$ as $a'_n \rightarrow 0^+$ and $a'_n \rightarrow 0^-$
If $a'_t = 0$ or $a'_t = -1$	$\lim_{a'_n \rightarrow 0^+} T = \frac{\pi}{2}$ then $\lim_{a'_n \rightarrow 0^-} T = -\frac{\pi}{2}$
If $a'_t > 0$ or $a'_t < -1$	$\lim_{a'_n \rightarrow 0^+} T = 0$ then $\lim_{a'_n \rightarrow 0^-} T = 0$
If $-1 < a'_t < 0$	$\lim_{a'_n \rightarrow 0^+} T = \pi$ then $\lim_{a'_n \rightarrow 0^-} T = -\pi$

A6.3.2 Case 2: Limiting Values for L

The quantity L becomes singular as $a'_t = \frac{a}{n s} \cdot n$ tends to zero if $a'_t = 0$ or -1 .

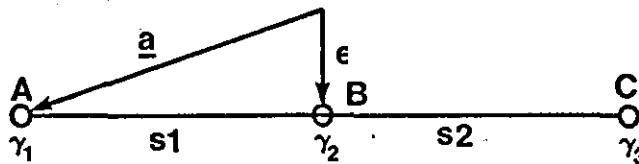
The velocity induced by the isolated segment is then indeterminate at these points (i.e. at either end of the segment, since if $a'_t = 0$ then $\underline{a} = 0$. If $a'_t = -1$ then

$$a'_t + 1 = 0 \quad \therefore \quad \frac{a}{s} \cdot t + 1 = 0 \quad \therefore \quad \underline{a} \cdot t + s = 0$$

$$\therefore \quad (\underline{a} \cdot t) \cdot t + s \cdot t = 0 \quad \therefore \quad \underline{a} + s t = 0$$

Consider the limiting case as $a'_t = \frac{a}{n s} \cdot n$ tends to zero at the junction between two segments of linear vorticity. (Diagram A6-2)

Diagram A6-2



The velocity induced at B due to S_1 is

$$\underline{V}_1 = \frac{1}{2\pi} (c_1 A_1 + c_2 A_2) \quad (A6.14)$$

while that due to S_2 is

$$\underline{V}_2 = \frac{1}{2\pi} (c'_1 A'_1 + c'_2 A'_2) \quad (A6.15)$$

where

$$c_1 = \gamma_1, \quad c_2 = \gamma_2 - \gamma_1$$

$$A_1 = -T_1 t_1 + L_1 n_1$$

$$A_2 = T_1 \left(-\frac{t_1}{s_1} + \frac{\epsilon}{s_1} \frac{n_1}{s_1} \right) - L_1 \left(\frac{\epsilon}{s_1} \frac{t_1}{s_1} - \frac{n_1}{s_1} \right)$$

since

$$a'_{n_1} = -\frac{\epsilon}{s_1}, \quad a'_{t_1} = -1$$

$$T_1 = \tan^{-1} \left(-\frac{s_1}{\epsilon} \right), \quad L_1 = \log_e \left(\frac{\epsilon}{s_1} \right)$$

$$c'_1 = \gamma_2, \quad c'_2 = \gamma_3 - \gamma_2$$

$$a'_{n_2} = -\frac{\epsilon}{s_2}, \quad a'_{t_2} = 0$$

$$A'_1, A'_2, T_2, L_2 \text{ as for } s_1.$$

Thus

$$\begin{aligned} \underline{v}_1 = \frac{1}{2\pi} & \left[\tan^{-1} \left(\frac{s_1}{\epsilon} \right) \left\{ -\gamma_1 \underline{t}_1 + (\gamma_2 - \gamma_1) \left(-\underline{t}_1 + \frac{\epsilon}{s_1} \underline{n}_1 \right) \right\} \right. \\ & \left. + \log_e \left(\frac{\epsilon}{s_1} \right) \left(\gamma_1 \underline{n}_1 - (\gamma_2 - \gamma_1) \left(-\frac{\epsilon}{s_1} \underline{t}_1 - \underline{n}_1 \right) \right) + (\gamma_2 - \gamma_1) \underline{n}_1 \right] \end{aligned} \quad (\text{A6.16})$$

and

$$\begin{aligned} \underline{v}_2 = \frac{1}{2\pi} & \left[-\tan^{-1} \left(\frac{s_2}{\epsilon} \right) \left(-\gamma_2 \underline{t}_2 + (\gamma_3 - \gamma_2) \frac{\epsilon}{s_2} \underline{n}_2 \right) \right. \\ & \left. + \log_e \left(\frac{s_2}{\epsilon} \right) \left(\gamma_2 \underline{n}_2 + (\gamma_3 - \gamma_2) \frac{\epsilon}{s_2} \underline{t}_2 \right) + (\gamma_3 - \gamma_2) \underline{n}_2 \right] \end{aligned} \quad (\text{A6.17})$$

The total induced velocity with ϵ very small is

$$\begin{aligned} \underline{v} = \frac{1}{2\pi} & \left[\gamma_2 \left(\underline{t}_1 \tan^{-1} \left(\frac{s_1}{\epsilon} \right) + \underline{t}_2 \tan^{-1} \left(\frac{s_2}{\epsilon} \right) + \underline{n}_2 \log_e s_2 \right. \right. \\ & \left. \left. - \underline{n}_1 \log_e (s_1) + (\underline{n}_1 - \underline{n}_2) (1 - \log_e (\epsilon)) + \gamma_3 \underline{n}_2 - \gamma_1 \underline{n}_1 \right] \end{aligned}$$

Thus there is only a limiting finite value as $\epsilon \rightarrow 0$ if $\underline{n}_1 = \underline{n}_2$ i.e. if the two segments lie in a straight line. Then

$$\underline{v} = \frac{\gamma_2}{2} \underline{t} + \frac{1}{2\pi} \left(\gamma_2 \log_e \left(\frac{s_2}{s_1} \right) + \gamma_3 - \gamma_1 \right) \underline{n} \quad (\text{A6.18})$$

or when calculating the velocity at a segment edge in such cases then

set

$$\begin{aligned} T = \frac{\pi}{2} \quad \text{and} \quad L = \log_e (b) \quad \text{if } a=0 \\ L_1 = -\log_e (a) \quad \text{if } b=0. \end{aligned}$$

APPENDIX 7

FORMULATION OF THE INFLUENCE COEFFICIENTS

A7.1 INTRODUCTION

Three types of singularities have been used, in the present mathematical model of the potential flow, round the aerofoil and wake system. These singularities are:

1. Linear vorticity
2. Linear source
3. Uniform source.

In Appendix 6 it was first shown how the influence coefficients of a parabolic vorticity distribution on a flat panel, are derived and then the influence coefficients of linear and constant vorticity distributions were considered.

The aim of this appendix is to relate the influence coefficients between corresponding source and vorticity distributions and to give the final formulae for their calculation, in program format.

A7.2 SOURCE-VORTEX RELATION

Let w_s and w_v be the complex potential functions of a source of strength γ_s and a vortex of strength γ_v respectively. From Reference 54:

$$w_s = \frac{\gamma_s}{2\pi} \ln \frac{r}{r_0} e^{i\theta} \tag{A7.1}$$

and

$$w_v = \frac{i\gamma_v}{2\pi} \ln \frac{r}{r_0} e^{i\theta}$$

where the various symbols are illustrated in Diagram A7.1.

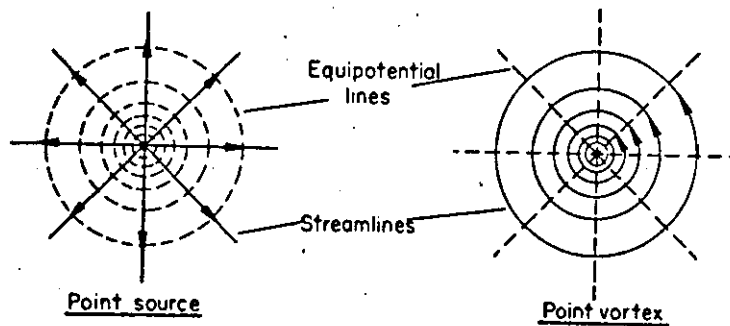


DIAGRAM A7.1

From a direct comparison of the two potential functions it is clear that the complex potential function is the same in each case with the exception of the constant. Thus if:

$$i\gamma_v = \gamma_s \quad (A7.2)$$

then

$$w_s = \frac{i\gamma_v}{2\pi} \ln \frac{r}{r_0} e^{i\theta}$$

and the influence coefficients due to a source γ_s can be obtained from the corresponding ones due to a vortex γ_v .

A7.3.1 Influence Coefficients due to Linear Vorticity

As shown in Appendix 6 the velocity induced by a linear vorticity distribution on a flat panel is given by

$$\underline{V} = \frac{1}{2\pi} [C_1 \underline{A}_1 + C_2 \underline{A}_2] \quad (A7.3)$$

Referring to diagram A7.2 the velocity induced at the control point j due to a linear vorticity distribution on panel k is given by

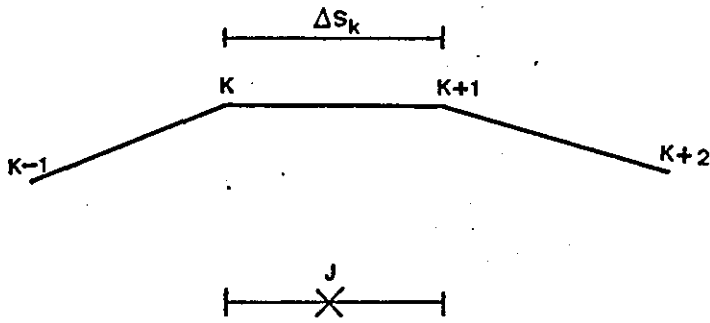


DIAGRAM A7.2

$$\begin{aligned} \underline{v}_{jk} &= \frac{1}{2\pi} \left[\gamma_k \underline{A}_{1jk} + (\gamma_{k+1} - \gamma_k) \underline{A}_{2jk} \right] \\ \underline{v}_{jk} &= \frac{1}{2\pi} \left[\gamma_k (\underline{A}_{1jk} - \underline{A}_{2jk}) + \gamma_{k+1} \underline{A}_{2jk} \right] \end{aligned} \quad (A7.4)$$

where

$\underline{A}_{1jk} - \underline{A}_{2jk}$ is referred to as the influence coefficient of the singularity γ_k located at the end of the panel.

and

\underline{A}_{2jk} is the influence coefficient of singularity γ_{k+1} located at the other end of the panel.

The strength of the singularity γ varies linearly between γ_k and γ_{k+1} .

In program format these influence coefficients can be obtained from the following set of equations (see Figure A7.1):

$$s_k = ((x_{k+1} - x_k)^2 + (z_{k+1} - z_k)^2)^{1/2} \quad (A7.5)$$

$$\bar{z}_k = .5(z_{k+1} + z_k) \quad (A7.6)$$

$$\bar{x}_k = .5(x_{k+1} + x_k) \quad (A7.7)$$

$$\underline{t}_k = \cos \theta_k \underline{i} + \sin \theta_k \underline{k} \quad (A7.8)$$

$$\underline{n}_k = -\sin \theta_k \underline{i} + \cos \theta_k \underline{k} \quad (A7.9)$$

$$F = \{(z_j - \bar{z}_k) \cos \theta_k - (x_j - \bar{x}_k) \sin \theta_k\} / s_k \quad (A7.10)$$

$$E = \{(z_j - \bar{z}_k) \sin \theta_k + (x_j - \bar{x}_k) \cos \theta_k\} / s_k \quad (A7.11)$$

$$T = \tan^{-1} \left(\frac{.5-E}{F} \right) + \tan^{-1} \left(\frac{.5+E}{F} \right) \quad (A7.12)$$

$$L = 0.5 \ln \frac{(.5-E)^2 + F^2}{(.5+E)^2 + F^2} \quad (A7.13)$$

$$F' = F \cos \theta_k - E \sin \theta_k \quad (A7.14)$$

$$E' = E \cos \theta_k + F \sin \theta_k \quad (A7.15)$$

$$B_{x_{jk}} = T \cdot E' + L \cdot F' - \sin \theta_k \quad (A7.16)$$

$$B_{z_{jk}} = L \cdot E' - T \cdot F' + \cos \theta_k$$

$$A_{x_{jk}} = 0.5(T \cos \theta_k - L \sin \theta_k) \quad (A7.17)$$

$$A_{z_{jk}} = 0.5(T \sin \theta_k + L \cos \theta_k)$$

$$G_{x_{v_{jk}}} = A_{x_{jk}} - B_{x_{jk}} + A_{x_{j,k-1}} + B_{x_{j,k-1}} \quad (A7.18)$$

$$G_{z_{v_{jk}}} = A_{z_{jk}} - B_{z_{jk}} + A_{z_{j,k-1}} + B_{z_{j,k-1}}$$

$$\underline{G}_{v_{jk}} = G_{x_{v_{jk}}} \underline{i} + G_{z_{v_{jk}}} \underline{k} \quad (A7.19)$$

$$G_{t_{v_{jk}}} = \underline{G}_{v_{jk}} \cdot \underline{t}_j \quad (A7.20)$$

$$G_{n_{v_{jk}}} = \underline{G}_{v_{jk}} \cdot \underline{n}_j$$

The following limits apply to T and L in the above formulae when the value of F approaches zero

$$\left. \begin{aligned} T &= \pm \pi && \text{for } -.5 < E < .5 \\ T &= \pm \frac{\pi}{2}, L=0 && \text{for } E = \pm .5 \\ T &= 0 && \text{for } |E| > .5 \end{aligned} \right\} \quad (A7.21)$$

where the positive values are taken if F approaches zero with positive value.

A7.3.2 Linear Source Influence Coefficients

In complex number notation the vortex influence coefficient $\underline{G}_{v_{jk}}$ can be written

$$\underline{G}_{v_{jk}} = G_{x_{v_{jk}}} + i G_{z_{v_{jk}}} \quad (A7.22)$$

where $i = \sqrt{-1}$.

Multiplying the above by i gives

$$iG_{-v}{}_{jk} = iG_{xv}{}_{jk} - G_{zv}{}_{jk}$$

Let the influence coefficient of the source singularity be:

$$G_{-s}{}_{jk} = G_{xs}{}_{jk} + iG_{zs}{}_{jk} \quad (A7.23)$$

since

$$G_{-s}{}_{jk} = iG_{-v}{}_{jk}$$

$$G_{-x}{}_{s}{}_{jk} = -G_{zv}{}_{jk}$$

and

$$iG_{xv}{}_{jk} = iG_{zs}{}_{jk}$$

and in terms of normal and tangential components of the influence coefficient

$$\left. \begin{aligned} G_{ns}{}_{jk} &= G_{tv}{}_{jk} \\ G_{ts}{}_{jk} &= -G_{nv}{}_{jk} \end{aligned} \right\} \quad (A7.24)$$

thus the source influence coefficients could be obtained directly from the vortex ones.

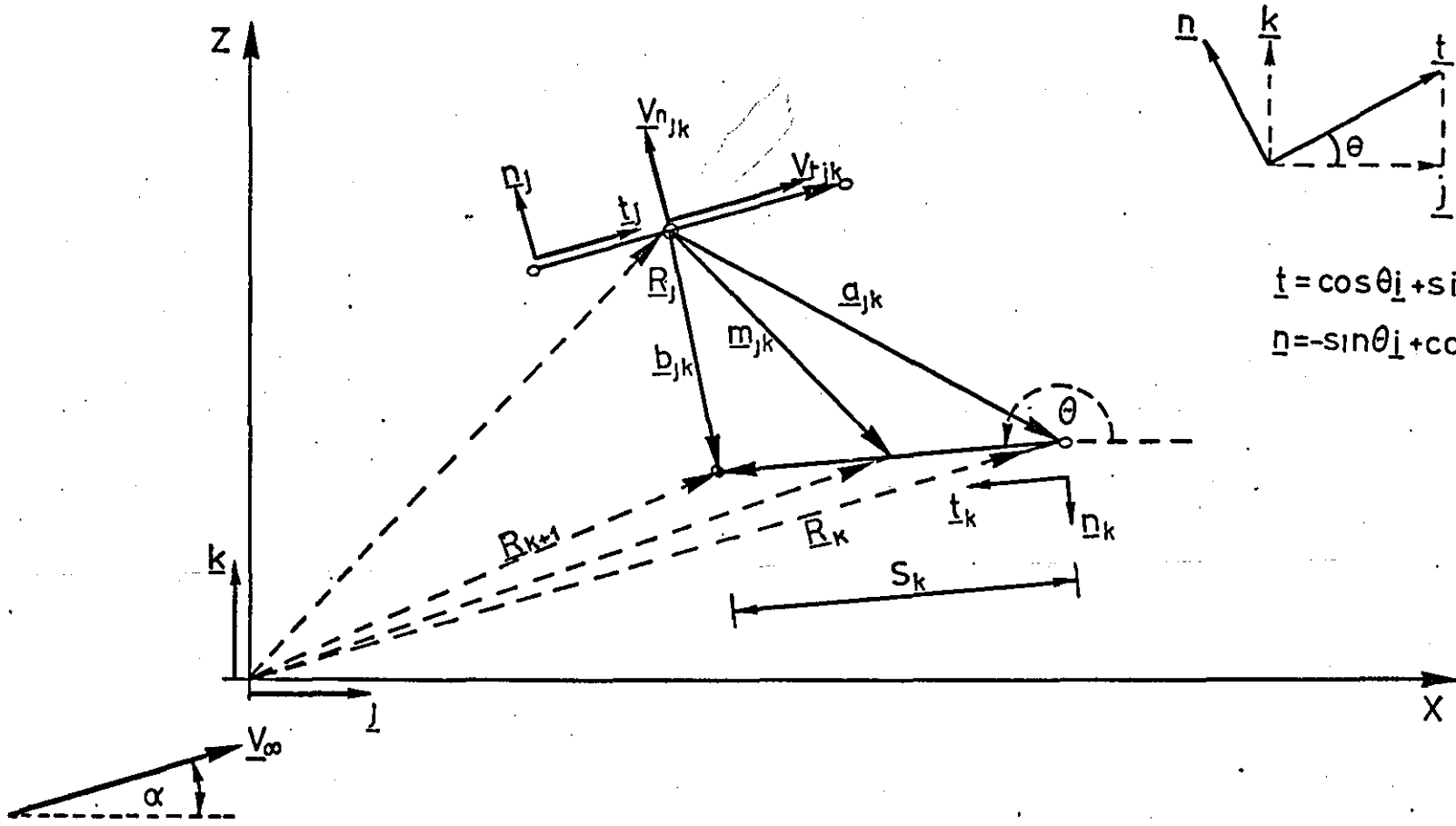
A7.3.3 Influence Coefficients due to a Uniform Source

As for the linear vorticity but the x and z components are given by

$$\left. \begin{aligned} G_{xs}{}_{jk} &= -2A_{zj}{}_{k} \\ G_{zs}{}_{jk} &= 2A_{xj}{}_{k} \\ G_{-s}{}_{jk} &= G_{xs}{}_{jk} \underline{i} + G_{zs}{}_{jk} \underline{k} \end{aligned} \right\} \quad (A7.25)$$

Thus

$$\left. \begin{aligned} G_{ts}{}_{jk} &= G_{-s}{}_{jk} \cdot \underline{t}_j \\ G_{ns}{}_{jk} &= G_{-s}{}_{jk} \cdot \underline{n}_j \end{aligned} \right\} \quad (A7.26)$$



$$\underline{t} = \cos\theta \underline{i} + \sin\theta \underline{k}$$

$$\underline{n} = -\sin\theta \underline{i} + \cos\theta \underline{k}$$

Figure A7.1

APPENDIX 8

LINEAR VORTICITY METHOD APPLIED TO THICK

AEROFOIL SECTIONS

A number of aerofoil sections have been tested using the linear vorticity model. The results presented here give an idea of the performance to be expected from this potential flow model. Generally speaking, provided that the aerofoil has a sharp trailing edge, the shape of the pressure distribution, predicted by the model, is correct. Figures A8.1 and A8.4 show the pressure distribution for five different aerofoils at various angles of attack. The main feature of these graphs is a pressure peak at the trailing edge. Reasons for this phenomenon are given in Appendix 9. Figure A8.2 shows the pressure distribution around a two element aerofoil section. The trend of the pressures is in reasonable agreement with the experimental shape (Ref.135) although the actual values need further improvement. The pressure distributions for the high lift aerofoil section of Figure A8.3 are in good agreement with those of Reference 138, however, a pressure oscillation is present near the trailing edge. Finally Figure A8.5 shows the variation of CL with angle of attack for a NACA4412 aerofoil. Clearly the inviscid theory gives rise to significant errors in predicting the value of CL in cases of high angles of attack, especially when the aerofoil has significant camber. Thus correction for viscous and wake effects is necessary.

2D INVISCID INCOMPRESSIBLE FLOW
COMPUTATION

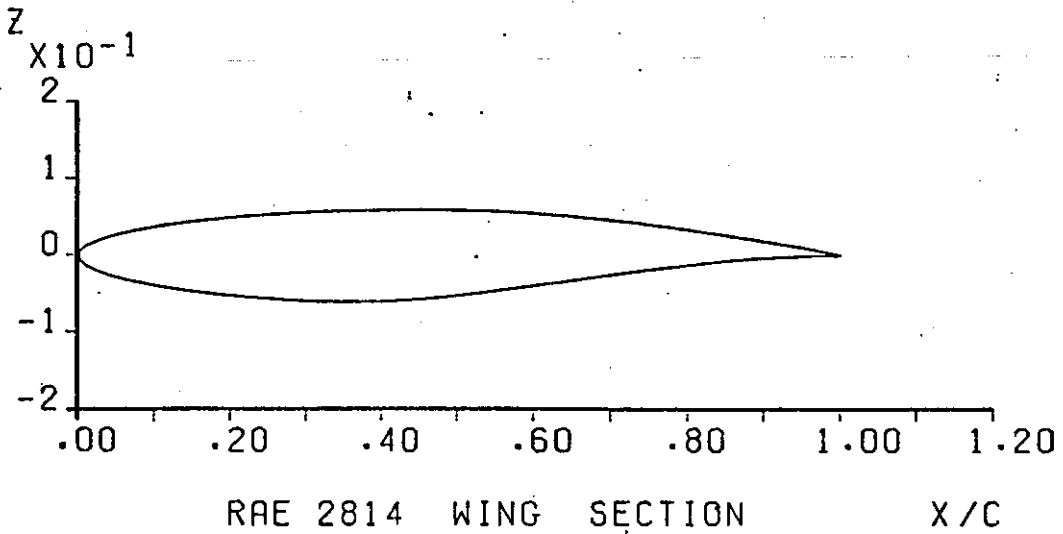
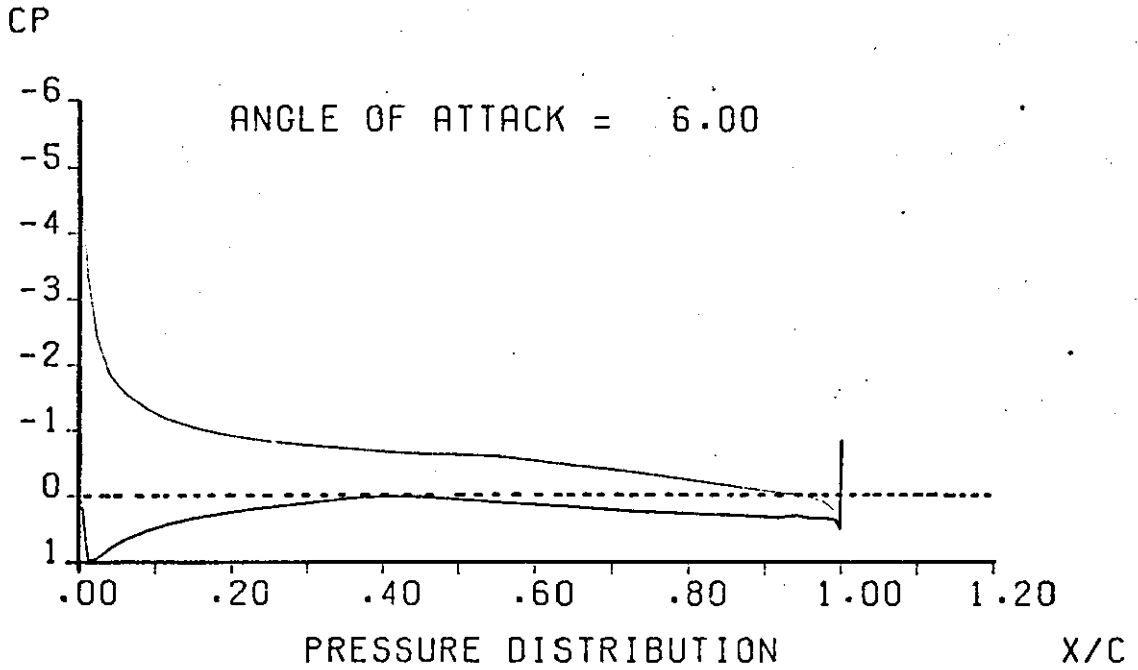


FIGURE A8-1 (a)

2D INVISCID INCOMPRESSIBLE FLOW
COMPUTATION

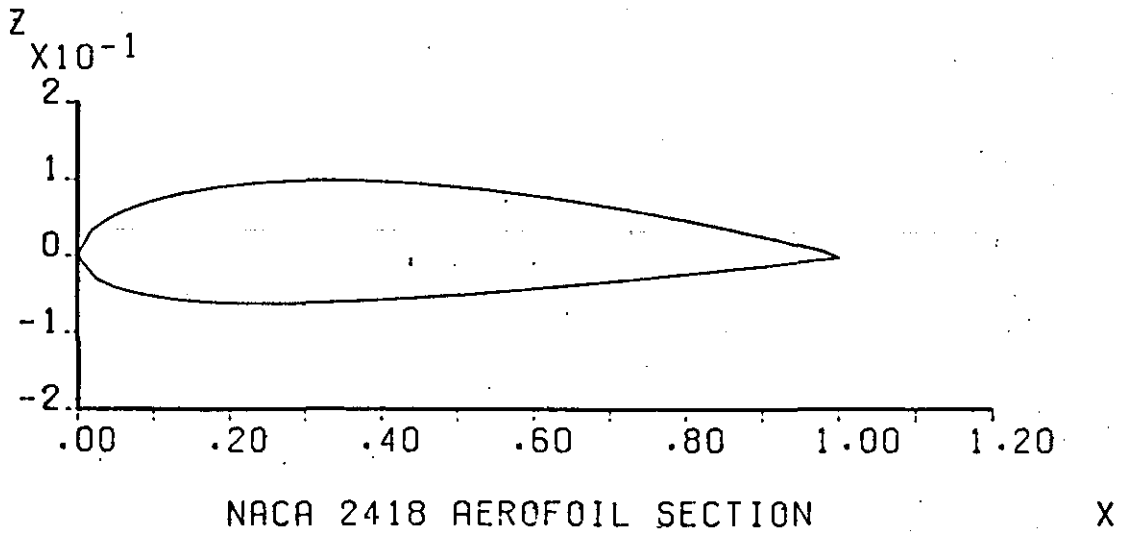
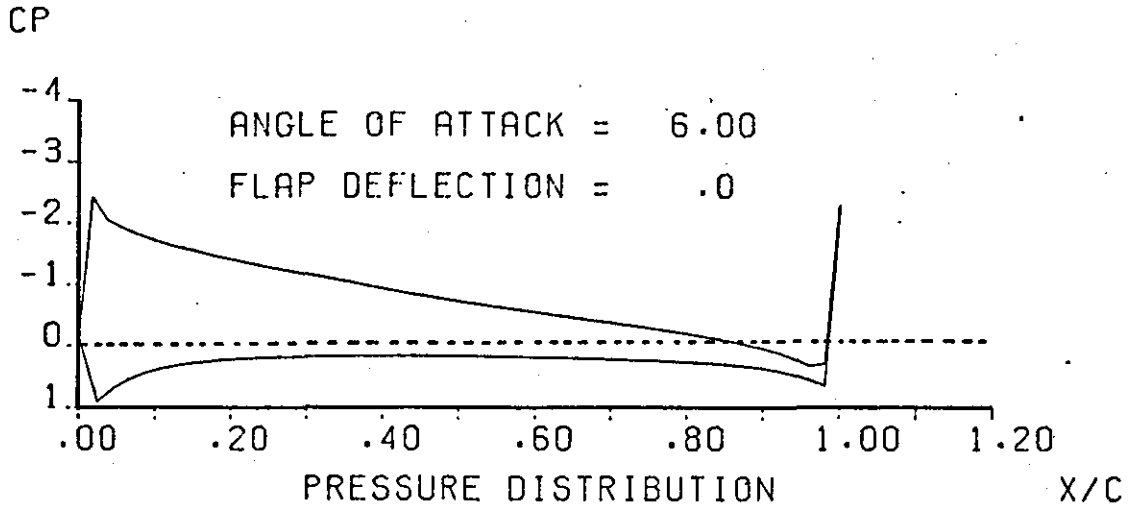


FIGURE A8-1 (b)

2D INVISCID INCOMPRESSIBLE FLOW
COMPUTATION

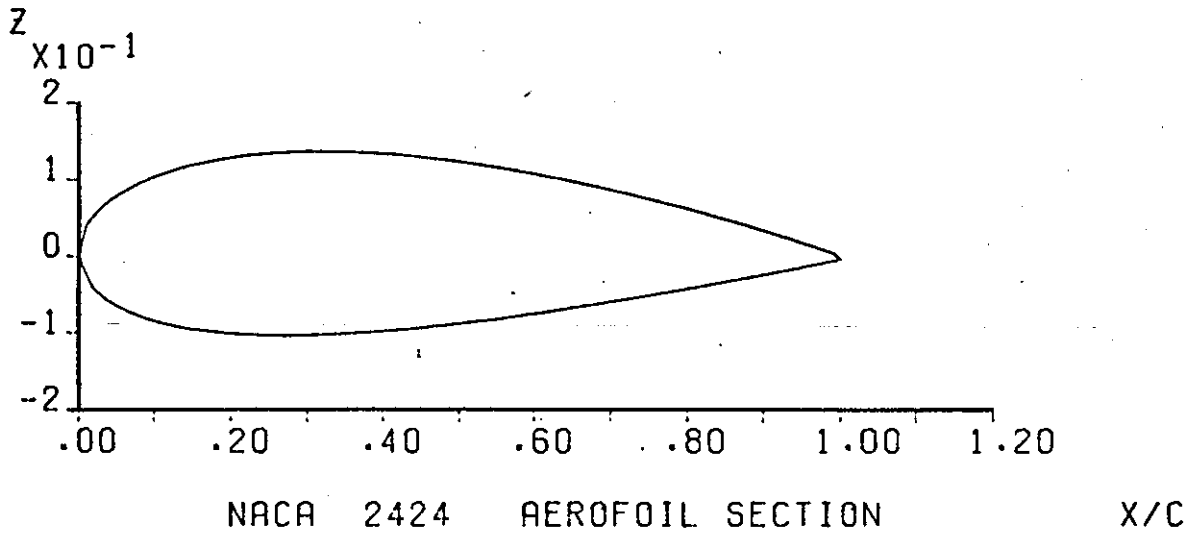
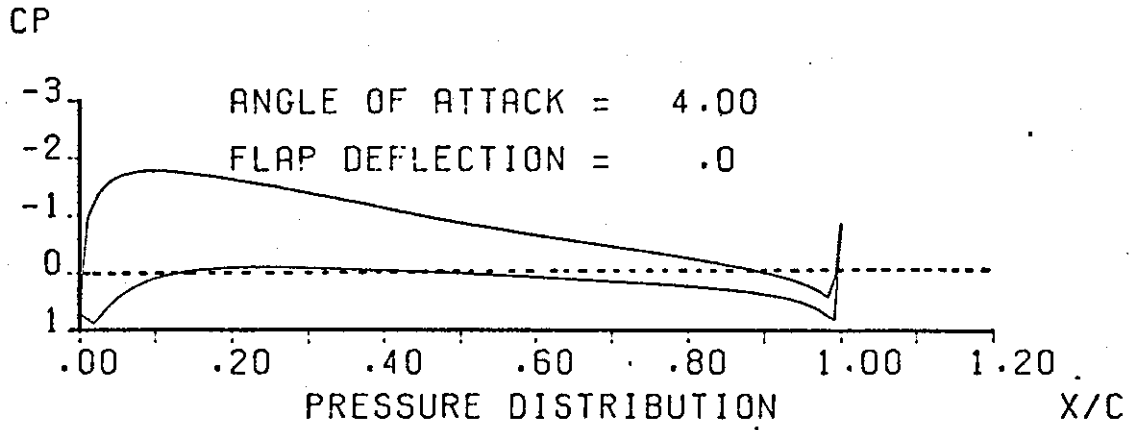


FIGURE A8-1 (c)

2D INVISCID INCOMPRESSIBLE FLOW
COMPUTATION

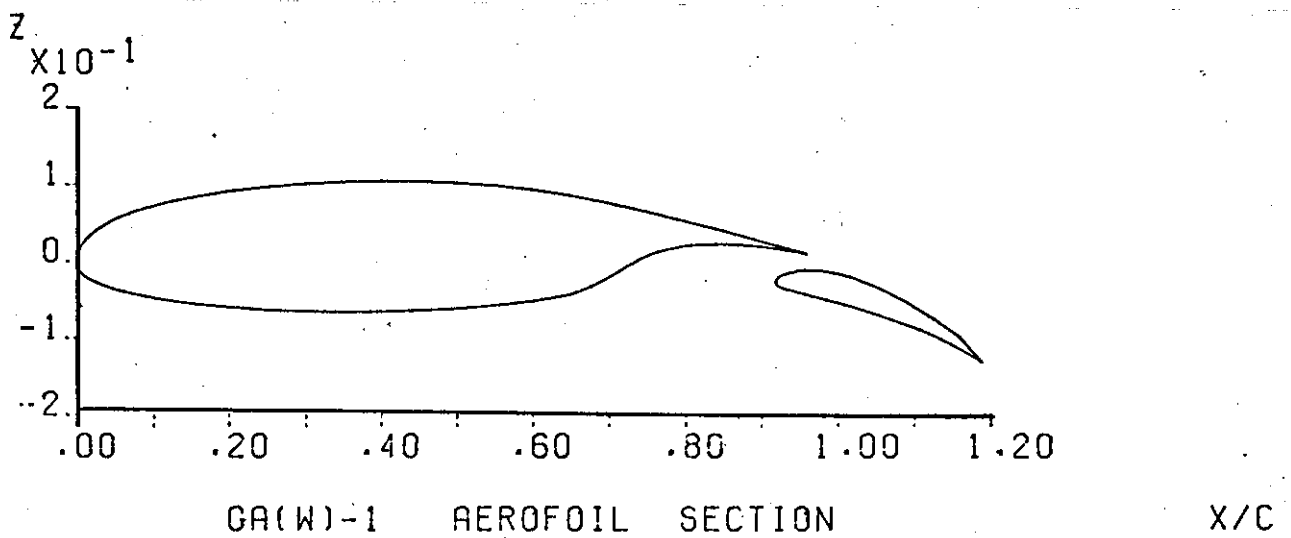
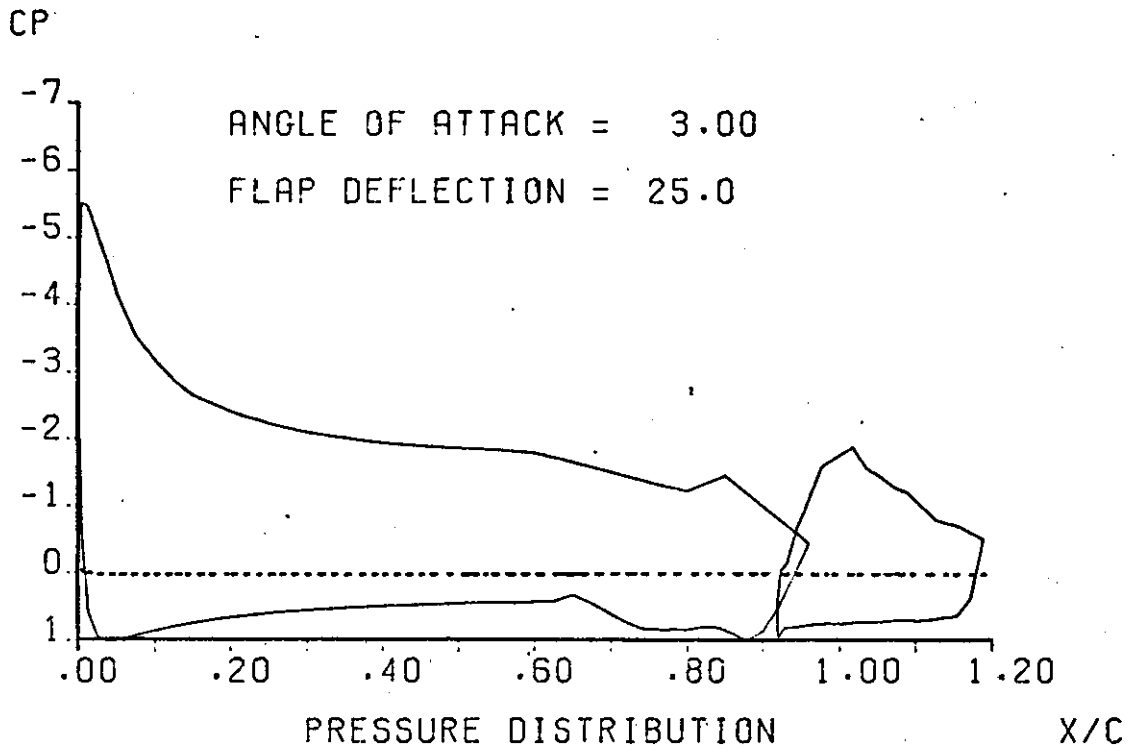


FIGURE A8-2 (b)

2D INVISCID INCOMPRESSIBLE FLOW
COMPUTATION

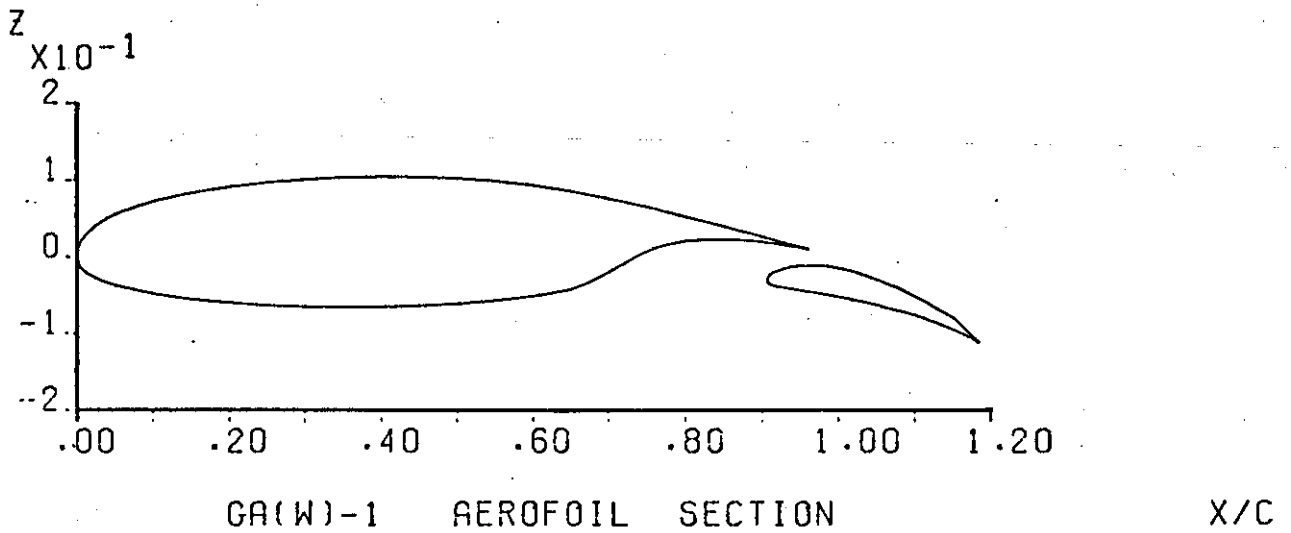
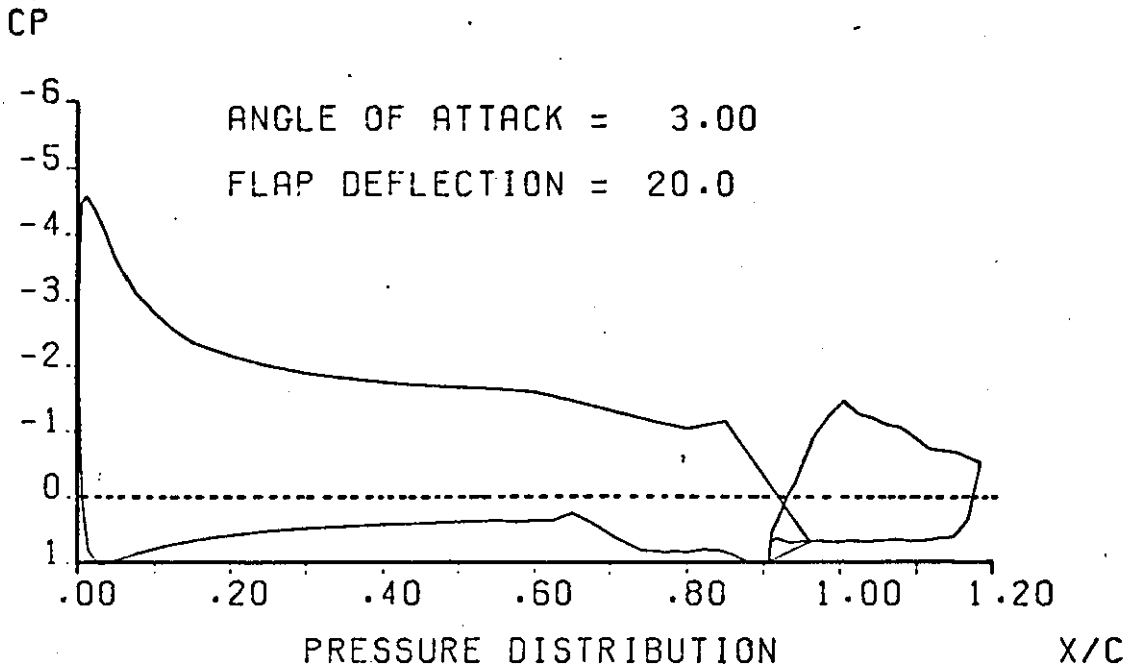


FIGURE A8-2 (a)

2D INVISCID INCOMPRESSIBLE FLOW
COMPUTATION

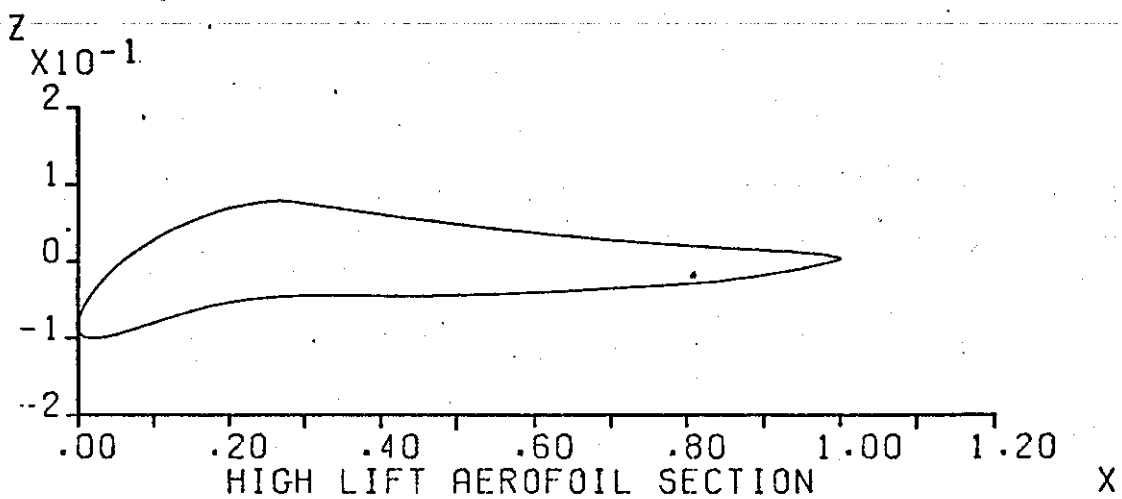
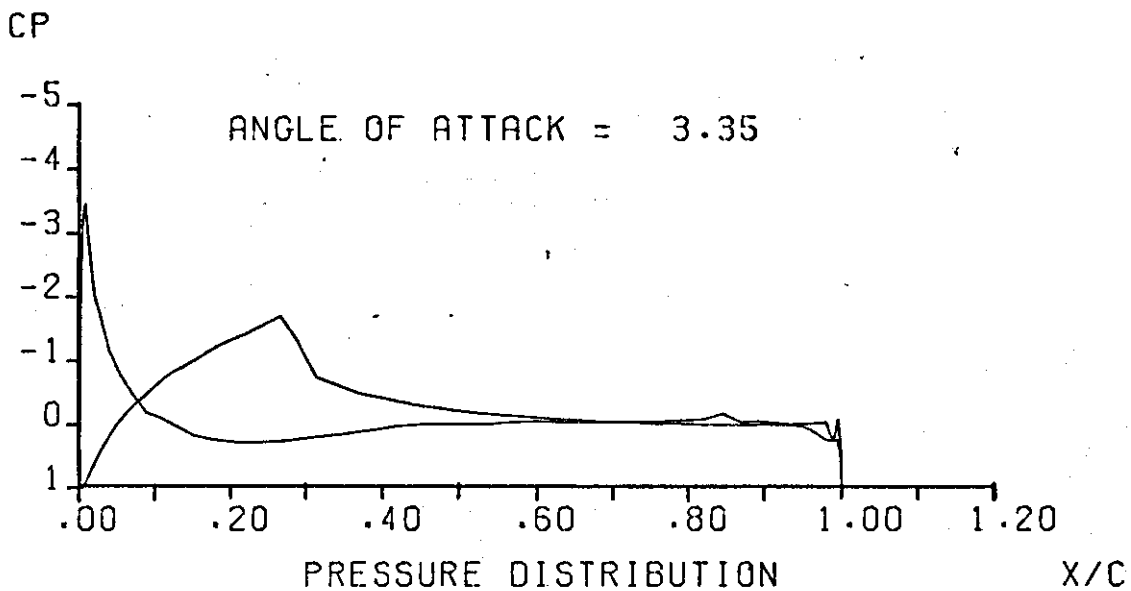


FIGURE A8-3 (a)

2D INVISCID INCOMPRESSIBLE FLOW
COMPUTATION

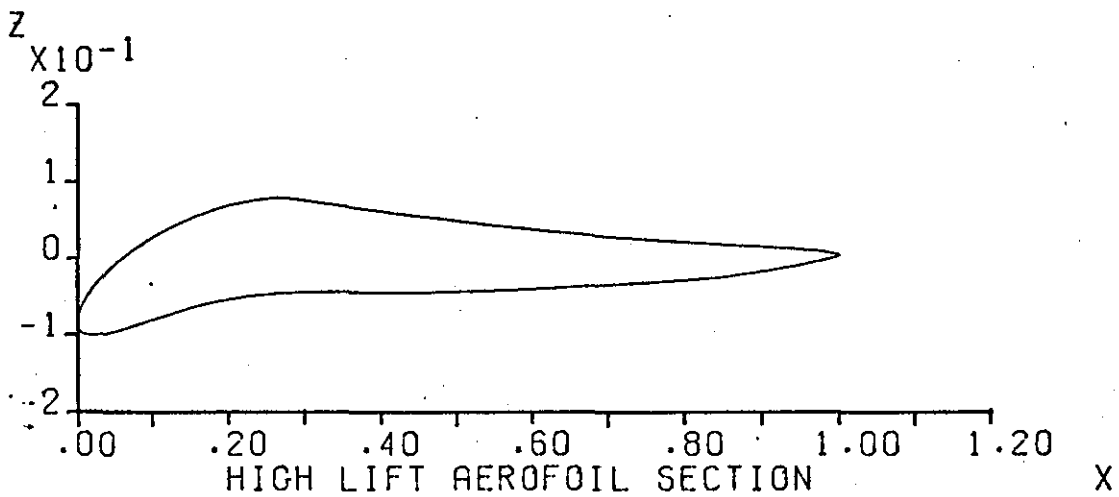
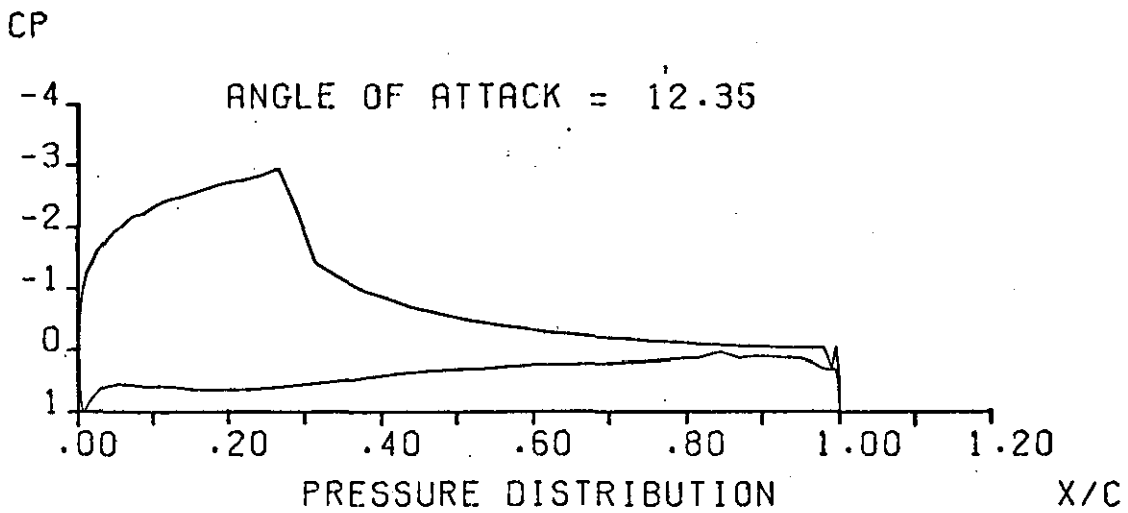


FIGURE A8-3 (b)

2D INVISCID INCOMPRESSIBLE FLOW
COMPUTATION

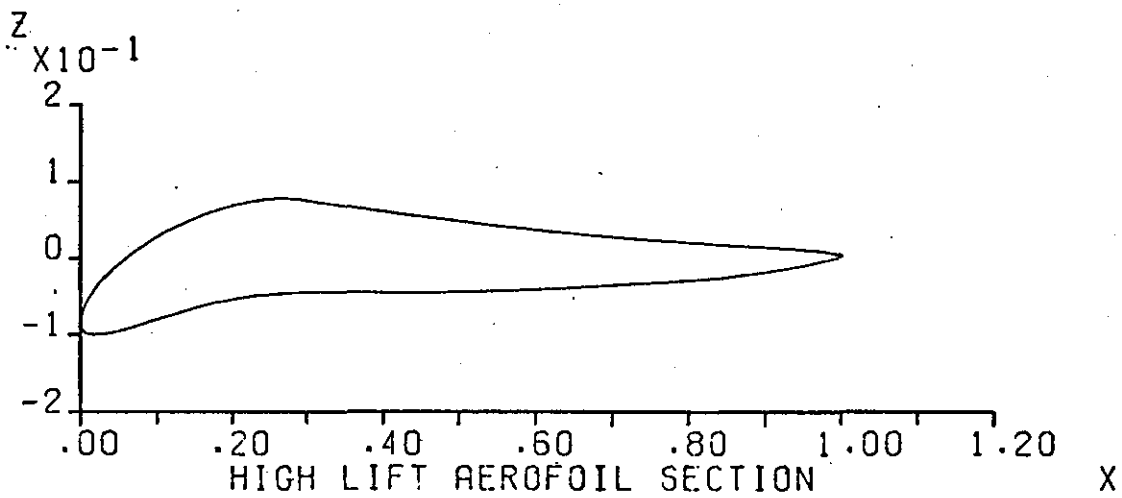
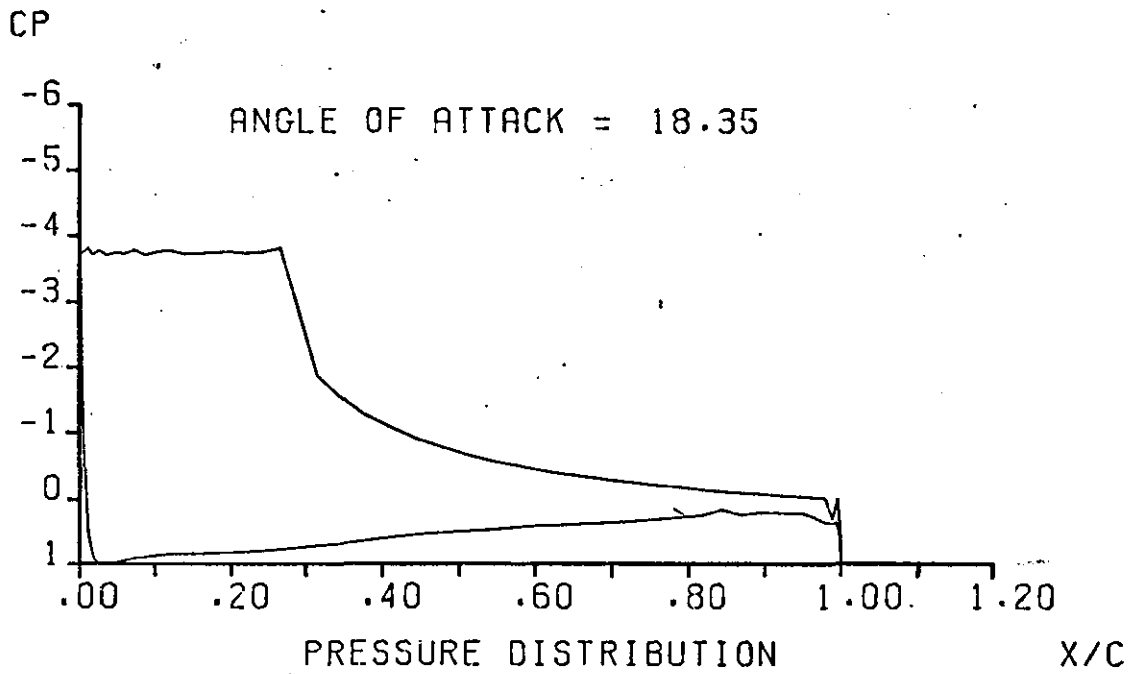


FIGURE A8-3 (c)

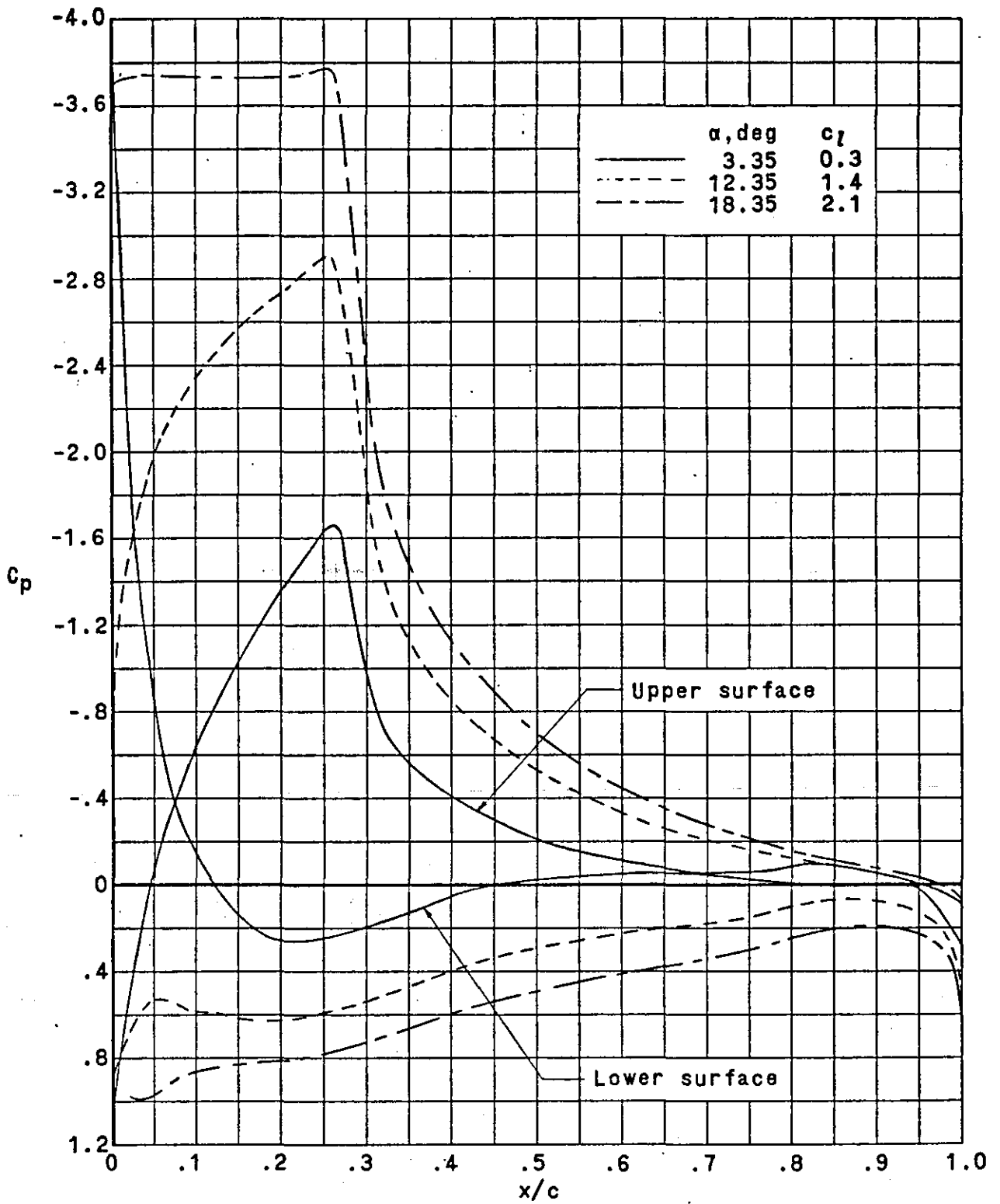


FIGURE A8-3 (d) Theoretical pressure distributions Ref I38

2D INVISCID INCOMPRESSIBLE FLOW
COMPUTATION

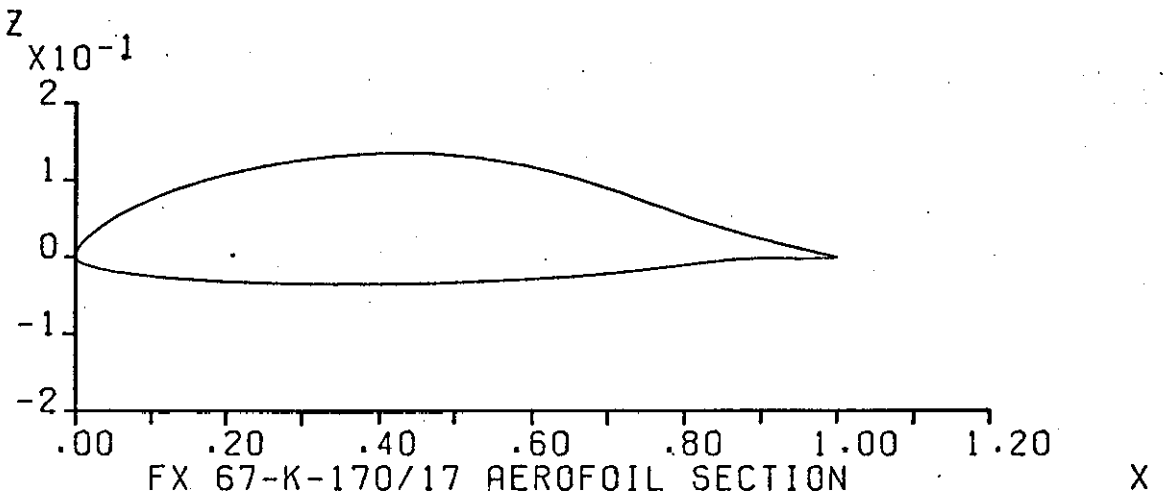
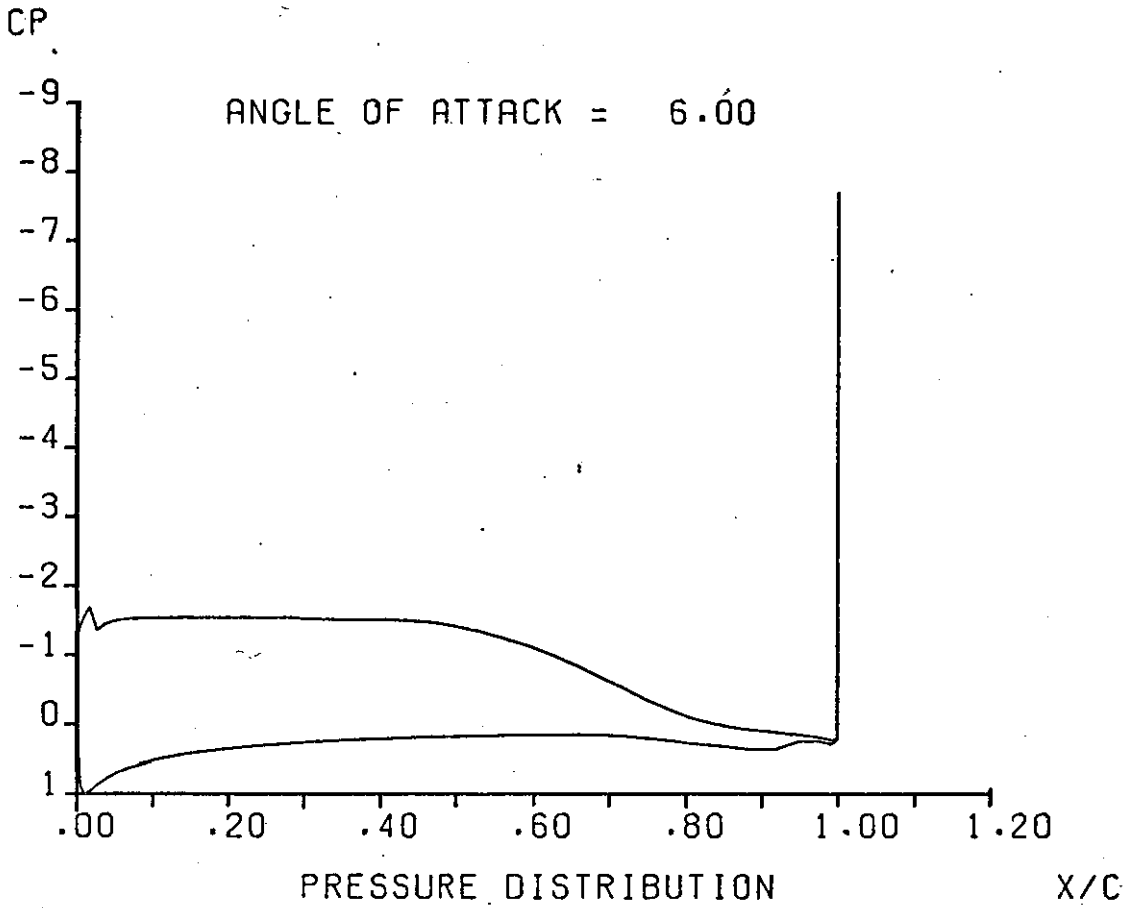


FIGURE A8-4

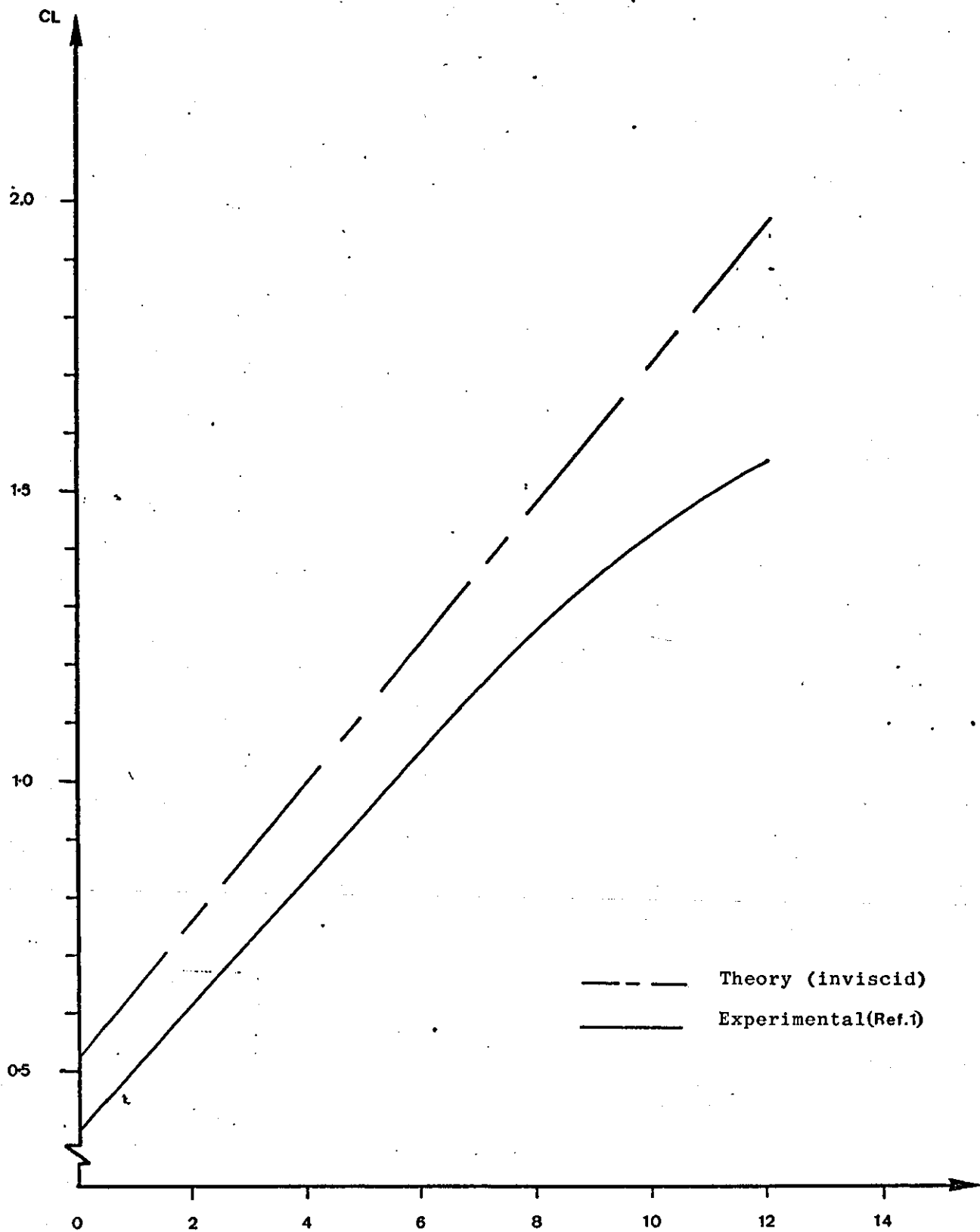


FIGURE A8-5 Lift coefficient versus angle of attack for a NACA4412 aerofoil section

APPENDIX 9

LINEAR VORTICITY MODEL - A GENERAL DISCUSSION

A9.1 INTRODUCTION

During the course of this research some time was devoted to the testing of the linear vorticity model. It was then found that a number of parameters have a significant effect on the performance of the model. These parameters are briefly discussed here:

A9.2 AEROFOIL SURFACE REPRESENTATION

The total number of panels, the panel distribution and the relative size of adjacent panels are important factors to be considered when modelling the aerofoil surface. In principle as the representation of the aerofoil contour improves the value of the lift coefficient should approach its potential flow value asymptotically. The panelling of the aerofoil surface can be improved, firstly by increasing the number of panels which construct the polygon representing the aerofoil contour and secondly by distributing these panels in such a way so that more panels are placed in high curvature areas. An efficient representation combines a sufficient number of panels with a good distribution so that computational time is kept low, while the accuracy of the analysis remains good.

To test the linear vorticity model a cosine distribution of panels was employed. As suggested (Ref.26) such a distribution is simple to apply and it tends to distribute more panels near the leading and trailing edges of the aerofoil. A more dense grouping of panels near the leading

edge is necessary since the surface curvature is considerable at this end. However too many panels near the trailing edge can sometimes cause an instability resulting in large values of the pressure coefficient at the trailing edge. However the effect on the overall value of the lift coefficient is very small.

The cosine distribution is based on the following formulae:

$$x_i = \frac{c}{2} (1 - \cos \phi_i)$$

where $\phi_i = \frac{i\pi}{N}$, $i=1,2,\dots,N$

and N is half the total number of corner points defining the aerofoil polygon.

A distribution which spaces points of the aerofoil surface relative to the local curvature (Figure A9.1) is better than the cosine distribution (Ref. 79) but is more complex to apply.

A9.3 AEROFOIL TRAILING EDGE

It was mentioned earlier that the linear vorticity model can only deal with aerofoils having a sharp trailing. In this section the variation in C_l with number of panels, everything else being constant, is examined for both sharp and blunt trailing edges.

A9.3.1 Sharp Trailing Edge

The aerofoils chosen to test the linear vorticity model were those of the NACA four digit series since they are very simple to generate. In most cases the trailing edge had to be extended very slightly until a sharp one was obtained. The variation of the lift coefficient with number of panels for a NACA 1408 aerofoil is shown in (Figure A9.2). Clearly as the number of panels is increased C_l increases, rather rapidly to begin with but it then levels off once the number of panels has reached sixty.

The increase in C_l can be explained as follows.

Assuming a particular type of distribution, as the panels are increased in number the length of the polygon representing the aerofoil surface approaches the length of the aerofoil contour, thus the circulation is increased and therefore C_l . At the same time the condition of zero normal velocity is satisfied at more points and therefore the distortion of the flow pattern, due to the presence of cross flows between control points, is reduced. Finally since the control points are the mid-panel points i.e. points laying within the contour, the more the panels the nearer to the true aerofoil surface the control points will be. Ideally the condition of zero normal velocity should be satisfied on the aerofoil surface.

A9.3.2 Blunt Trailing Edge

The above test was repeated with a NACA 1408 aerofoil whose trailing edge was not modified this time ($\frac{t}{c}_{TE} = 0.00168$. Ref.1.p.408). From Figure A9.3 it can be seen that C_l varies in a random fashion with the number of panels. This is thought to be due to:

1. The presence of a sharp change in surface curvature at the trailing edge.
2. The type of Kutta condition used.

Sharp corners are reflected as discontinuities in the vortex distribution (Ref.7) and this is probably the main reason for the instability observed.

The Kutta condition applied to this model requires the total vorticity at the trailing edge to be zero. (Diagram A9.1). However this allows the individual vorticity strengths at the trailing edge to reach large values.

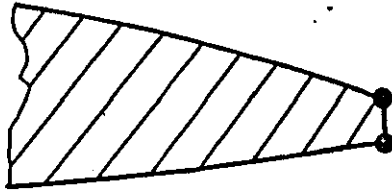


DIAGRAM 9.1

Thus as the number of panels is increased it is a combination of variable surface discontinuity and vorticity distribution which creates the instability observed in the value of C_L .

To solve this problem researchers using this type of potential flow model have tried to modify the Kutta condition (Ref.34). With viscous flow, however, such Kutta conditions increase the complexity of the calculation. If only a potential flow analysis is required matters can easily be improved by simply extending the trailing edge till a sharp one is obtained. Such extensions are obviously valid. If they are short in comparison to the length of the aerofoil chord so that no serious effect on lift arises from the modification of the aerofoil geometry.

A9.4 INACCURATE SURFACE CORNER POINTS

When the number of surface corner points defining the aerofoil polygon is not sufficient one may wish to increase this number by simply interpolating between the available points. However, unless some kind of curve fitting is employed, interpolation may result in panel oscillation (Diagram A9.2). This in turn gives an oscillation in C_p (Figure A9.4). The overall effect, of such inaccurate geometry, on C_L has not been investigated.

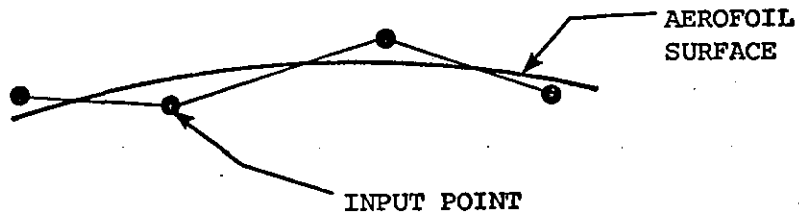


DIAGRAM 9.2

A9.5 THIN AEROFOILS

When the upper and lower surfaces of the aerofoil come very close together the induced velocities can reach high values, due to the nature of the vorticity singularity, thus instability, in the solution for the vortex strength may arise.

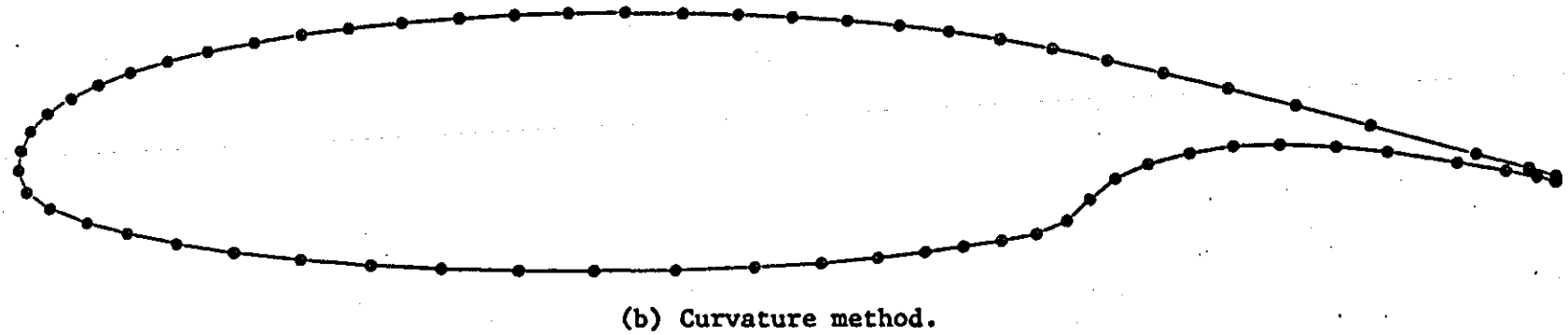
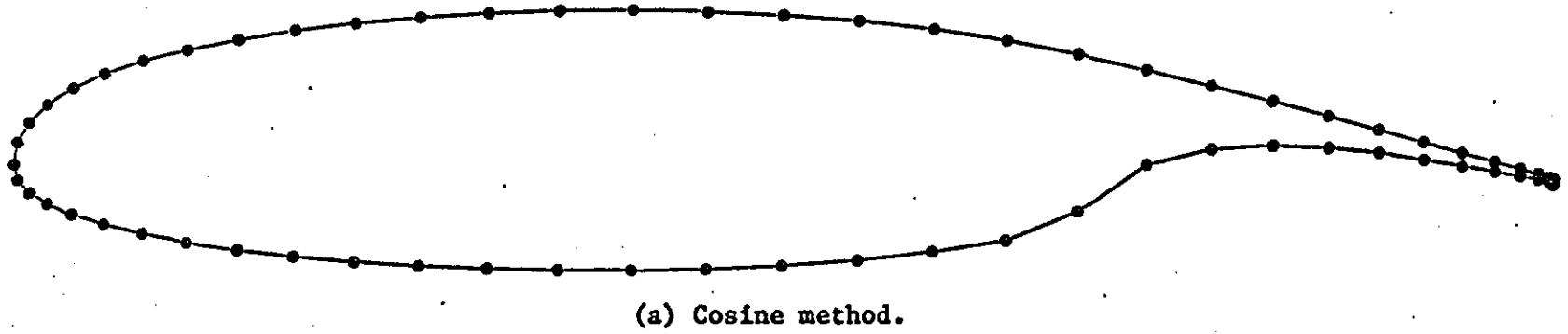


Figure A9.1 Comparison of cosine and curvature methods for distribution of segment corner points.

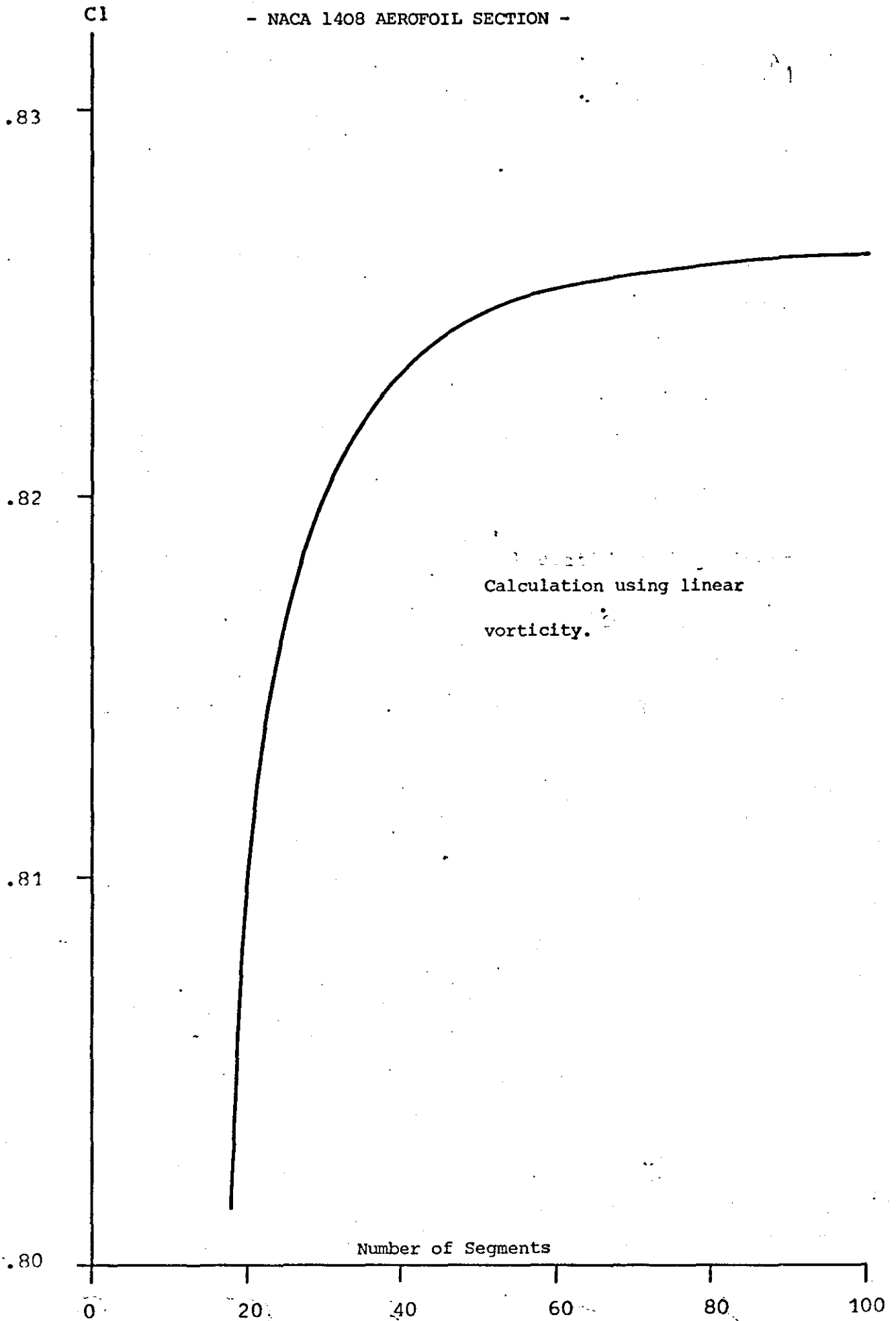


Figure A9.2 Variation of CL with number of panels (Linear vorticity model)

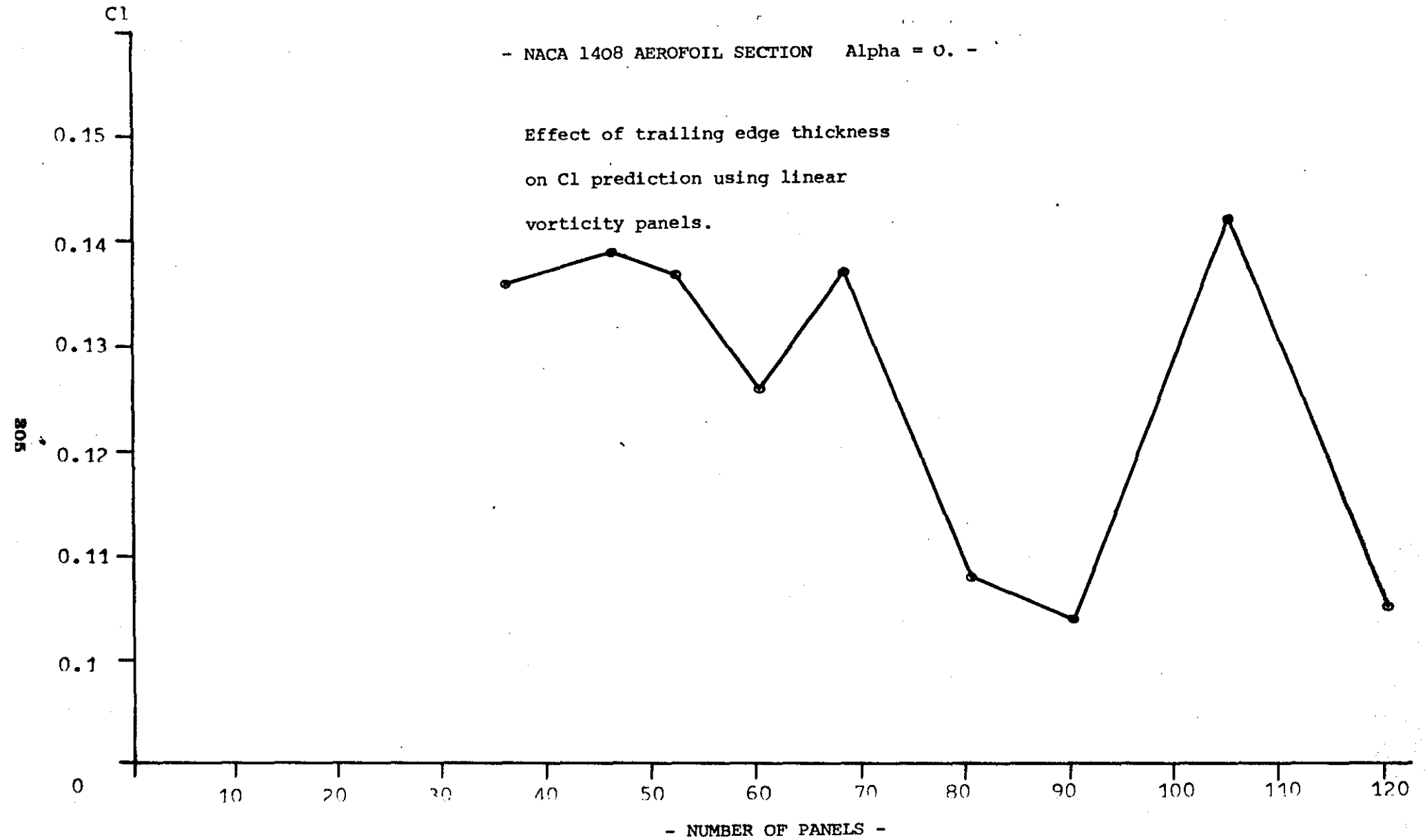


Figure A9.3 Effect of thick trailing edge on CL calculation when using
the linear vorticity method.

2D INVISCID INCOMPRESSIBLE FLOW
COMPUTATION

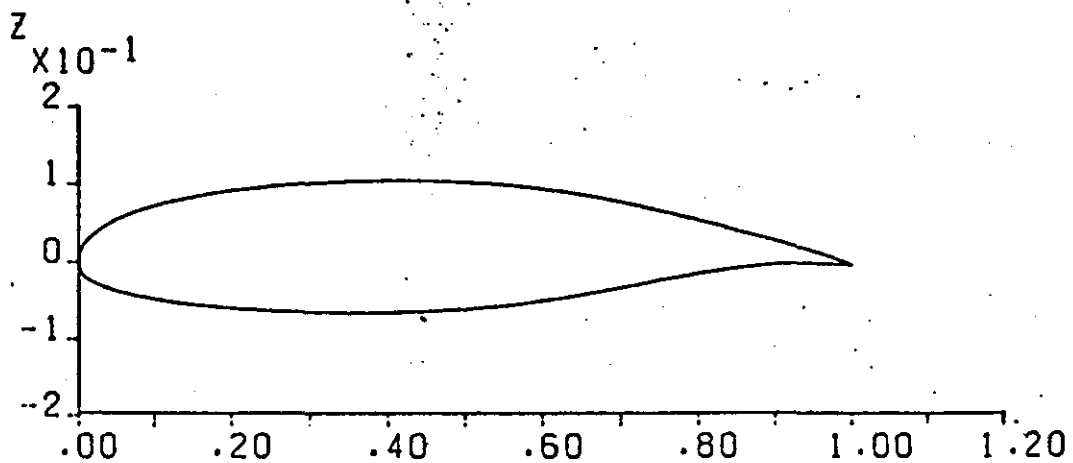
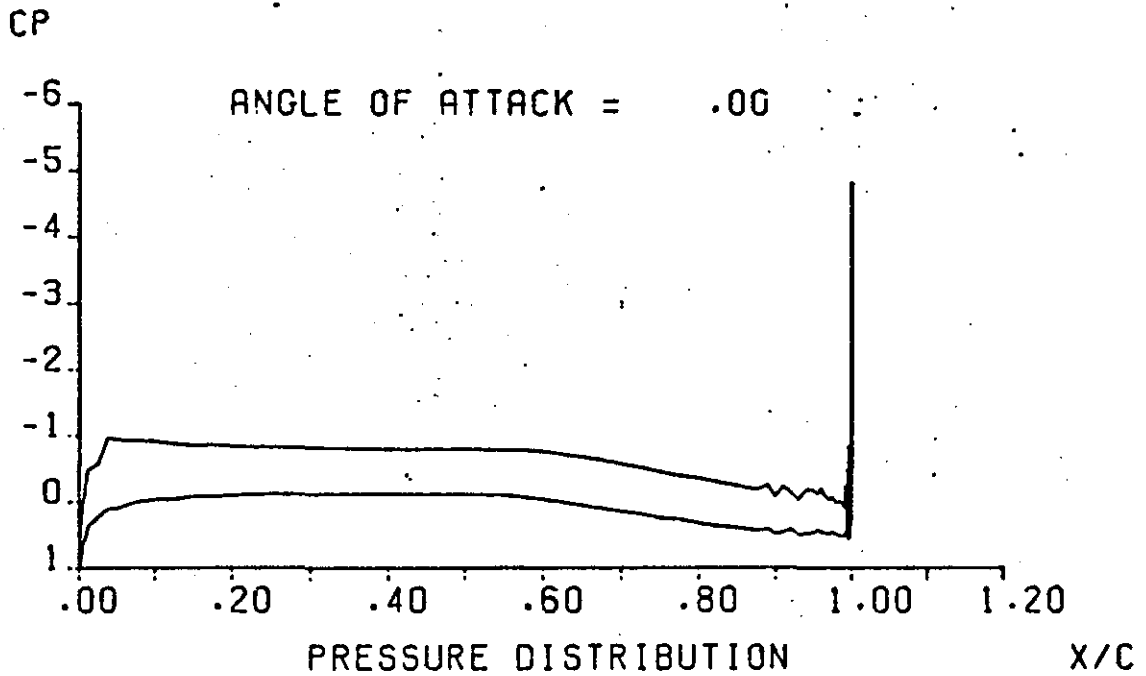


Figure A9.4 Effect of incorrect airfoil surface coordinates on pressure distribution

APPENDIX 10

VISCOUS FLOW BOUNDARY CONDITIONS

In methods which involve separate calculations of the viscous flow over the wing and its wake and of the inviscid flow external to these layers a matching process is required which will enable the mutual interactions of these flows to be determined. The matching is accomplished through the normal velocity at the aerofoil boundary (i.e. aerofoil surface). For the wake both normal and tangential velocity boundary conditions are considered (Diagram A10.1).

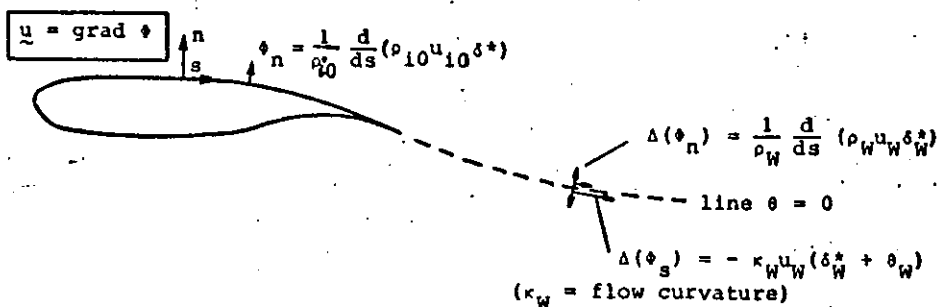


Diagram A10-1 : Boundary conditions
on Aerofoil and wake

A10.1 NORMAL VELOCITY BOUNDARY CONDITION

This condition is derived by integrating the equation of continuity across the boundary layer for both the real flow and the equivalent inviscid flow and matching the two flows at the edge of the boundary layer.

(25).

A10.1.1 Surface Transpiration Method

Consider the continuity equation applied to a viscous layer.

$$\frac{d(\rho u)}{ds} + \frac{d(\rho v)}{dy} = 0 \quad (\text{A10.1})$$

$$\therefore d(\rho v) = - \frac{d(\rho u)}{ds} dy$$

and upon integrating from $y=0$ to $y=\delta$ using $V_{y=0}=0$

$$v_{\delta} = - \frac{1}{\rho_{\delta}} \int_0^{\delta} \frac{d(\rho u)}{ds} dy \quad (\text{A10.2})$$

By definition

$$\delta^* = \int_0^{\delta} \left(1 - \frac{\rho u}{\rho_{\delta} u_{\delta}}\right) dy \quad (\text{A10.3})$$

Hence

$$\delta^* \rho_{\delta} u_{\delta} = \int_0^{\delta} (\rho_{\delta} u_{\delta} - \rho u) dy$$

$$\therefore \delta^* \rho_{\delta} u_{\delta} = \rho_{\delta} u_{\delta} \delta - \int_0^{\delta} \rho u dy$$

and differentiating with respect to s

$$\frac{d}{ds} (\delta^* \rho_{\delta} u_{\delta}) = \frac{d}{ds} (\rho_{\delta} u_{\delta} \delta) - \int_0^{\delta} \frac{d}{ds} (\rho u) dy$$

$$\therefore \frac{d}{ds} (\delta^* \rho_{\delta} u_{\delta}) = \frac{d}{ds} (\delta \rho_{\delta} u_{\delta}) + \rho_{\delta} v_{\delta}$$

$$\therefore v_{\delta} = \frac{1}{\rho_{\delta}} \left[\frac{d}{ds} (\delta^* \rho_{\delta} u_{\delta}) - \frac{d}{ds} (\delta \rho_{\delta} u_{\delta}) \right] \quad (\text{A10.4})$$

For incompressible flow ρ_{δ} is constant and equation (A10.4) reduces to

$$v_{\delta \text{ vis}} = \frac{d}{ds} (\delta^* u_{\delta}) - \frac{d}{ds} (\delta u_{\delta}) \quad (\text{A10.5})$$

Consider now the inviscid incompressible flow, and the normal velocity that exists at a small distance δ above the aerofoil surface.

From the continuity equation,

$$\frac{du}{ds} = - \frac{dv}{dy} \quad \therefore dv = - \frac{du}{ds} dy$$

Integrating with respect to y between 0 and δ and using $V_{y=0}=0$, the above

gives:

$$v_{\delta \text{ inv}} = - \frac{du_{\delta}}{ds} \int_0^{\delta} dy = -\delta \frac{du_{\delta}}{ds}$$

Comparing $v_{\delta \text{ vis}}$ and $v_{\delta \text{ inv}}$ the incremental normal velocity due to viscosity is

$$v_N = \frac{d}{ds}(\delta^* u_\delta) \quad (\text{A10.6})$$

The above equation states that the mass flow normal to the wall in the equivalent inviscid flow is equal to the streamwise rate of change of the 'mass flow deficit' in the boundary layer.

A10.1.2 Equivalent Sources

The additional outflow due to the presence of the boundary layer may also be represented by a surface distribution of sources whose strength (volume flow rate) per unit area is

$$\sigma = \frac{d}{dx}(u_\delta \delta^*) \quad (\text{A10.7})$$

The same considerations are applicable in the wake, where however σ is negative, so that we may consider the rear dividing streamline of the irrotational flow as dotted with sinks (67).

Now the 'new' fluid emitted at the sources just described would fill a region, adjacent to the body, of thickness δ^* ; for the flow of 'new' fluid past any point P (with velocity u_δ) must equal the total outflow from the part of the surface between that point and the point of attachment (Diagram A10.2) and this is

$$\int_0^x \sigma dx = u_\delta \delta^* .$$

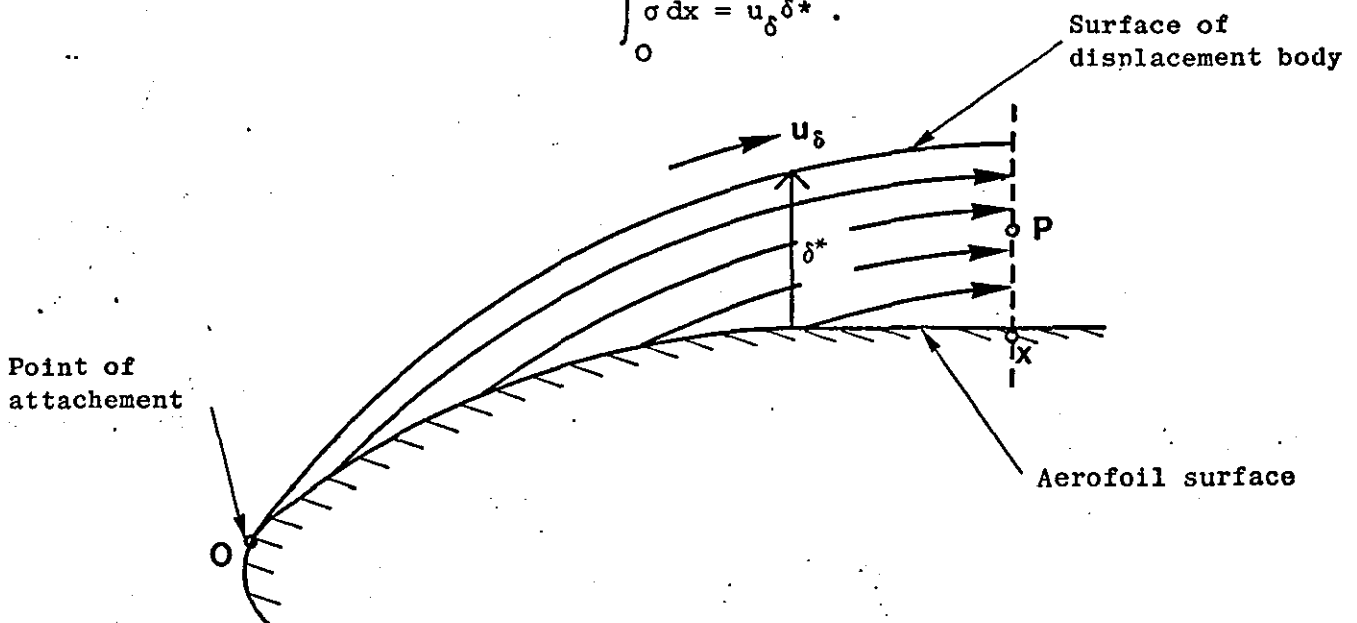


Diagram A10-2

per unit span. But the external flow can be regarded as the irrotational flow about the surface of separation between the fluid from upstream and the 'new' fluid from the sources and this has been shown to be a surface displaced into the fluid through a distance δ^* .

A10.2 TANGENTIAL VELOCITY BOUNDARY CONDITION

The momentum deficit of the viscous layer is given by:

$$M = \int_0^{\delta} \rho_{\delta} u_{\delta}^2 dy - \int_0^{\delta} \rho_{\delta} u_{\delta}^2 dy \quad (A10.8)$$

Equation A10.8 gives the difference between the momentum flow rate which would occur in no boundary layer existed (in this case the velocity in the vicinity of the surface, at the station considered, would be constant and equal to the main stream velocity u_{δ}) and the momentum flow rate within the boundary layer.

By definition

$$\delta^* = \int_0^{\delta} \left(1 - \frac{\rho u}{\rho_{\delta} u_{\delta}}\right) dy$$

$$\theta = \int_0^{\delta} \frac{\rho u}{\rho_{\delta} u_{\delta}} \left(1 - \frac{u}{u_{\delta}}\right) dy$$

Multiplying both δ^* and θ by $\rho_{\delta} u_{\delta}^2$ gives

$$\rho_{\delta} u_{\delta}^2 \delta^* = \int_0^{\delta} (\rho_{\delta} u_{\delta}^2 - \rho u) dy$$

and

$$\rho_{\delta} u_{\delta}^2 \theta = \int_0^{\delta} \rho u (u_{\delta} - u) dy$$

Adding the above two equations gives

$$\rho_{\delta} u_{\delta}^2 (\delta^* + \theta) = \int_0^{\delta} (\rho_{\delta} u_{\delta}^2 - \rho u^2) dy \quad (A10.9)$$

From (A10.9) and (A10.8) the difference in momentum flow rate or momentum deficit in the viscous layer can be written as:

$$M = \rho_{\delta} u_{\delta}^2 (\delta^* + \theta) \quad (A10.10)$$

This momentum deficit must equal the momentum transported outside of the viscous layer by the velocity V_N (Eq. A10.6).

Consider the inner boundary surface s of the viscous layer and the

displacement surface s_1 . Let k be the curvature of s and k_1 the curvature of s_1 . Let s_2 be the flow boundary which is also curved and take an element δs_2 on this boundary (Figure A10.1). The momentum transported to the inviscid outer flow through δs_2 will be turned in direction due to the curvature of the element (25). This will result in an additional normal force being exerted on the outer flow. This force will in fact be exerted on the displacement surface s_1 , since the displacement thickness is the normal distance that the momentum is transported, and it will be greater than that required in potential flow by an incremental force ΔF where

$$\Delta F = \rho_\delta u_\delta^2 (\delta^* + \theta) k_1 \delta s_1$$

and

$$k_1 = \frac{d\epsilon_1}{ds_1} \quad (A10.11)$$

This incremental force must result in a normal pressure gradient in the outer flow and must be related to an incremental tangential velocity Δu_T (25). The normal pressure gradient is given by:

$$\frac{dp}{dn} = -\rho_\delta u_\delta \frac{du_\delta}{dn} \quad (A10.12)$$

since

$$M = \Delta u_T \times \text{mass} = \Delta u_T (\rho_\delta u_\delta \delta s_1)$$

then

$$\Delta u_T = -u_\delta (\delta^* + \theta) \frac{d\epsilon_1}{ds_1} \quad (A10.13)$$

From Figure A10.1 and from the definition of curvature

$$\frac{d\epsilon_1}{ds_1} = \frac{d^2 y_1}{dx_1^2} / \left(1 + \left(\frac{dy_1}{dx_1}\right)^2\right)^{3/2} \quad (A10.14)$$

Referring again to Figure A10.1

$$y_1 = y + \delta^* \cos \epsilon$$

$$x_1 = x - \delta^* \sin \epsilon$$

(A10. 5)

$$\frac{dy}{dx} = \tan \epsilon, \quad \frac{dx}{ds} = \cos \epsilon$$

writing

$$\frac{dy_1}{dx_1} = \frac{dy_1/dx}{dx_1/dx}$$

and using

$$\frac{dy_1}{dx} = \frac{dy}{dx} + \frac{d\delta^*}{dx} \cos \epsilon - \delta^* \sin \epsilon \frac{d\epsilon}{dx}$$

$$\frac{dx_1}{dx} = 1 - \frac{d\delta^*}{dx} \sin \epsilon - \delta^* \cos \epsilon \frac{d\epsilon}{dx}$$

$$dx = \cos \epsilon ds$$

the following result is obtained

$$\frac{dy_1}{dx_1} = \frac{\frac{d\delta^*}{ds} + \tan \epsilon (1 - \delta^* \frac{d\epsilon}{ds})}{1 - \tan \epsilon \frac{d\delta^*}{ds} - \delta^* \frac{d\epsilon}{ds}} \quad (A10.16)$$

which after expanding binomially and retaining first order small terms only gives

$$\frac{dy_1}{dx_1} = \frac{d\delta^*}{ds} \sec^2 \epsilon + \tan \epsilon \quad (A10.17)$$

Note, terms such as $\frac{d\delta^*}{ds}$ and $\delta^* \frac{d\epsilon}{ds}$ are both small. For the latter term at the leading edge $\frac{d\epsilon}{ds}$ may be large but δ^* is very small, elsewhere as δ^* increases $\frac{d\epsilon}{ds}$ becomes very small.

Using a similar approach the two terms of (A10.14) can be written as

$$\left[1 + \left(\frac{dy_1}{dx_1} \right)^2 \right]^{3/2} = \sec^3 \epsilon (1 + 2 \tan \epsilon \frac{d\delta^*}{ds})^{3/2} \quad (A10.18)$$

$$\frac{d^2 y_1}{dx_1^2} = \frac{d}{dx} \left(\frac{dy_1}{dx_1} \right) \bigg/ \frac{dx_1}{dx} = \left(\frac{d^2 \delta^*}{ds^2} + \frac{d\epsilon}{ds} \right) \sec^2 \epsilon \quad (A10.19)$$

Combining equations (A10.14), (A10.18) and (A10.19) the curvature of the displacement surface can be written

$$\frac{d\epsilon_1}{ds_1} = \left[\frac{d\epsilon}{ds} + \frac{d^2 \delta^*}{ds^2} \right] \bigg/ \left[1 + 2 \tan \epsilon \frac{d\delta^*}{ds} \right]$$

and since $\tan \frac{d\delta^*}{ds}$ is very small as explained above

$$\frac{d\epsilon_1}{ds_1} = \frac{d\epsilon}{ds} + \frac{d^2 \delta^*}{ds^2} \quad (A10.20)$$

Finally combining (A10.13) and (A10.20) the incremental tangential velocity (V_T where $V_T = \Delta u_T$) is given by

$$v_T = -u_\delta (\delta^* + \theta) \left(\frac{d\epsilon}{ds} + \frac{d^2 \delta^*}{ds^2} \right) \quad (A10.21)$$

which involves only terms related to the geometry of the surface concerned (i.e. wake or aerofoil) and the corresponding viscous layer parameters.

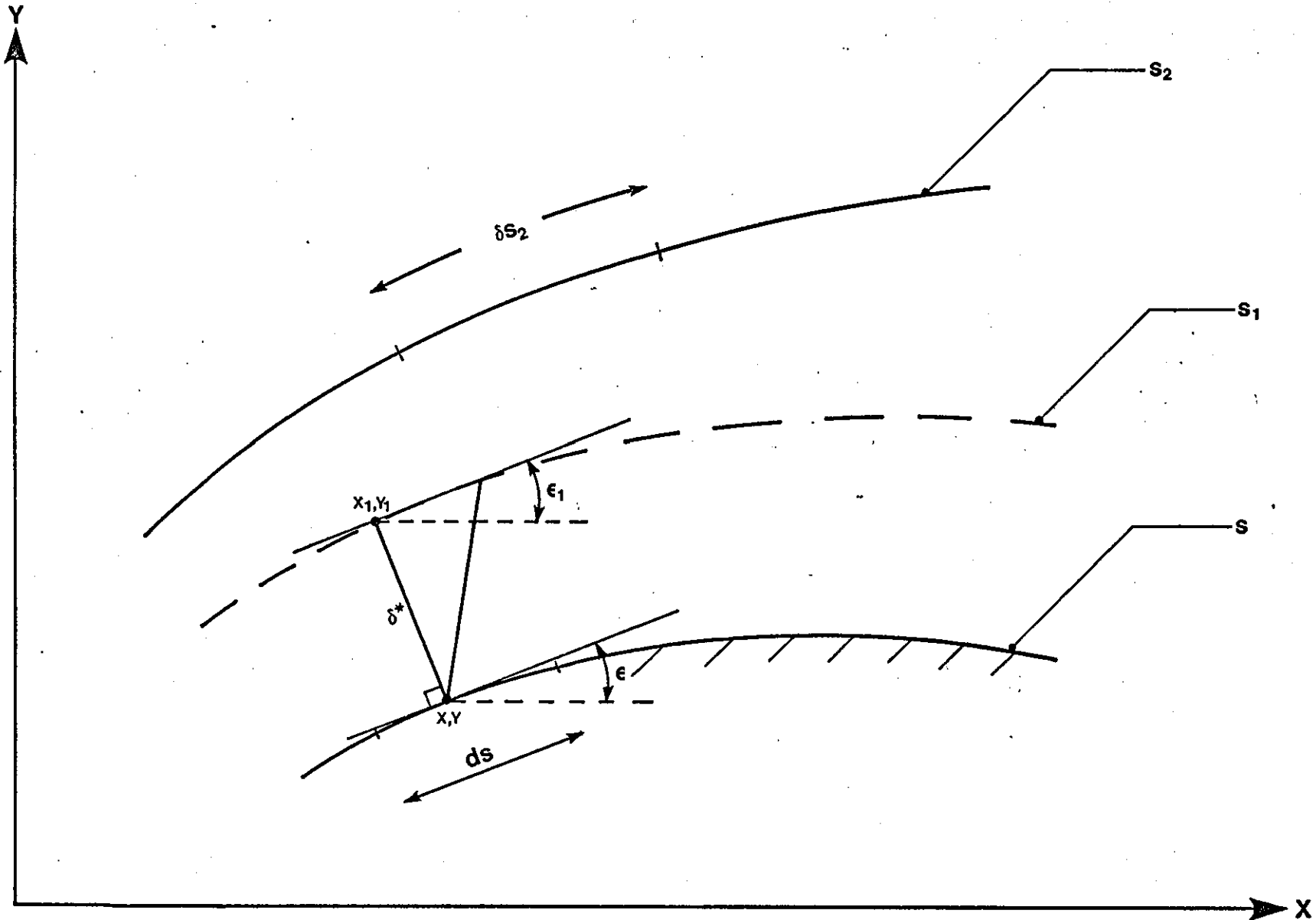


FIGURE A10-1

APPENDIX 11

WAKE GEOMETRY

The procedure for the calculation of the initial shape of the wake singularity sheets is described here.

All.1 WAKE OF MAIN AEROFOIL

The initial position of the wake singularity sheet is taken to be parallel to the flap upper surface, the distance between the two streamlines being,

$$h = (z_T - z_N) + \frac{1}{2} t_{TE} \quad (\text{All.1})$$

where x_T, z_T are the coordinates of the trailing edge of the main aerofoil (see Figure A11-4), t_{TE} is the trailing edge thickness (for blunt trailing edges), x_N, z_N are the coordinates of the point of intersection of the normal from x_T, z_T to the flap upper surface.

Once h is found the coordinates x'_k, z'_k of the wake singularity sheet are calculated by constructing the normal at x_k, z_k on the flap surface and taking a point x'_k, z'_k distance h from x_k, z_k .

The calculation procedure is the same for both negative and positive overlap. For pointed trailing edges $t_{TE} = 0$.

All.1.1 Calculation of x_N, z_N

To find h only x_N, z_N have to be found since x_T, z_T and t_{TE} are known from the aerofoil geometry.

Let d be the length of the normal from x_T, z_T to the flap surface, then d is the minimum distance between x_T, z_T and the flap surface. But

$$d^2 = (x_T - x_N)^2 + (z_T - z_N)^2 \quad (\text{All.2})$$

Starting from a convenient point on the flap surface a search is carried out to find a point x_i, z_i (defined from the input geometry) for which

$$(x_T - x_i)^2 + (z_T - z_i)^2$$

is minimum. Once x_i, z_i has been located a quadratic curve

$$z = ax^2 + bx + c \quad (\text{All.3})$$

is fitted to the points $(x_{i-1}, z_{i-1}), (x_i, z_i), (x_{i+1}, z_{i+1})$. (Diagram All.1).

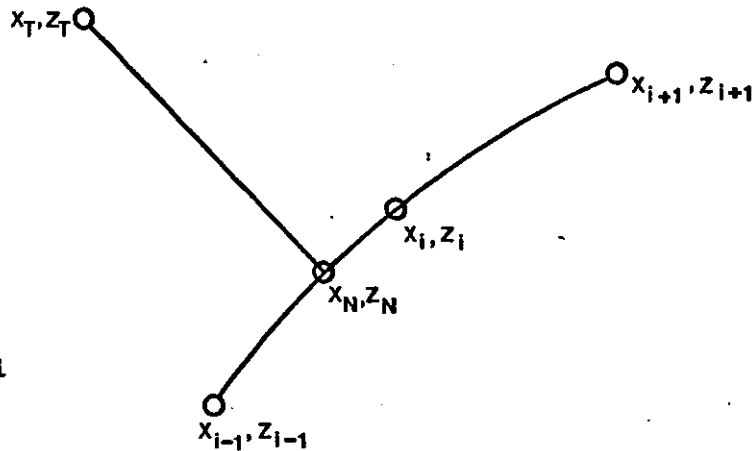


Diagram All-1

The slope at any point on the curve is given by

$$\frac{dz}{dx} = 2ax + b \quad (\text{All.4})$$

and the slope of the normal to this point is given by

$$m = -\frac{1}{\frac{dz}{dx}} = -\frac{1}{2ax+b} \quad (\text{All.5})$$

Let

$$z = mx + e$$

be the equation of the line joining (x_T, z_T) with (x_N, z_N) then:

$$z_N = -\frac{1}{2ax_N+b} x_N + e$$

where

$$e = z_T + \frac{1}{2ax_N+b} x_T$$

Thus

$$z_N = -\frac{x_N}{2ax_N+b} + z_T + \frac{1}{2ax_N+b} x_T \quad (\text{All.6})$$

Since (x_N, z_N) is a point on the curve,

$$z = ax^2 + bx + c \quad (All.7)$$

then
$$z_N = ax_N^2 + bx_N + c \quad (All.8)$$

Combining (All.6) and (All.8) gives

$$ax_N^2 + bx_N + c = \frac{1}{2ax_N + b} (x_T - x_N + z_T (2ax_N + b))$$

which reduces to

$$A_1 x_N^3 + A_2 x_N^2 + A_3 x_N + A_4 = 0 \quad (All.9)$$

where $A_1 = 2a^2$

$$A_2 = 3ab$$

$$A_3 = 2ac + b^2 - 2az_T + 1$$

$$A_4 = b_c - x_T - bz_T$$

Newton-Raphson's method is used to solve the above cubic, starting with an initial value $x_N = x_i$ to find the root x_N such that

$$x_{i-1} \leq x_N \leq x_{i+1}$$

Substituting x_N into equation (All.7) gives the value of z_N .

Once x_N and z_N are known the value of h is found from equation (All.1).

All.1.2 Calculation of Wake Coordinates (x'_k, z'_k)

The following calculation is repeated for all points x_k, z_k on the flap upper surface for which

$$x_k > x_T$$

A quadratic

$$z = ax^2 + bx + c \quad (All.10)$$

is first fitted to the points

$$(x_{k-1}, z_{k-1}), (x_k, z_k), (x_{k+1}, z_{k+1})$$

The slope of the curve at any point is given by

$$\frac{dz}{dx} = 2ax + b,$$

hence

$$\left(\frac{dz}{dx}\right)_{x=x_k} = 2ax_k + b$$

The slope of the normal to the curve at (x_k, z_k) is:

$$m = -1/(2ax_k + b) \quad (All.11)$$

and the equation of this normal is

$$z = mx + e ,$$

where

$$e = z_k - mx_k$$

hence

$$z = mx + z_k - mx_k$$

∴

$$z'_k = mx'_k + z_k - mx_k \quad (All.12)$$

Also

$$(x_k - x'_k)^2 + (z_k - z'_k)^2 = h^2 \quad (All.13)$$

and substituting for z'_k from (All.12) gives a quadratic in x'_k with solutions,

$$x'_k = x_k \pm \frac{h}{\sqrt{m^2 + 1}} \quad (All.14)$$

The two values of x'_k are substituted back in (All.10) and the value of x'_k which satisfies

$$z'_k > z_k$$

is the required value.

All.2 WAKE OF FLAP

The initial position of the flap wake singularity is obtained by simply extending the flap chord length by 100% (8).

Let (x'_f, z'_f) and (x_f, z_f) be the coordinates of the flap leading and trailing edges respectively and let x_w, z_w be the coordinates of the wake trailing edge (Figure All.1). The coordinates x_w, z_w are given by:

$$\begin{aligned} x_w &= 2x_f - x'_f \\ z_w &= 2z_f - z'_f \end{aligned} \quad (All.15)$$

The distance $x_f - x'_f$ is divided into $N/2$ segments where N is the total number of segments defining the flap surface.

All.3 SINGLE ELEMENT AEROFOIL

To estimate the initial position of the wake singularity sheet in this case the method described in Section All.2 is used.

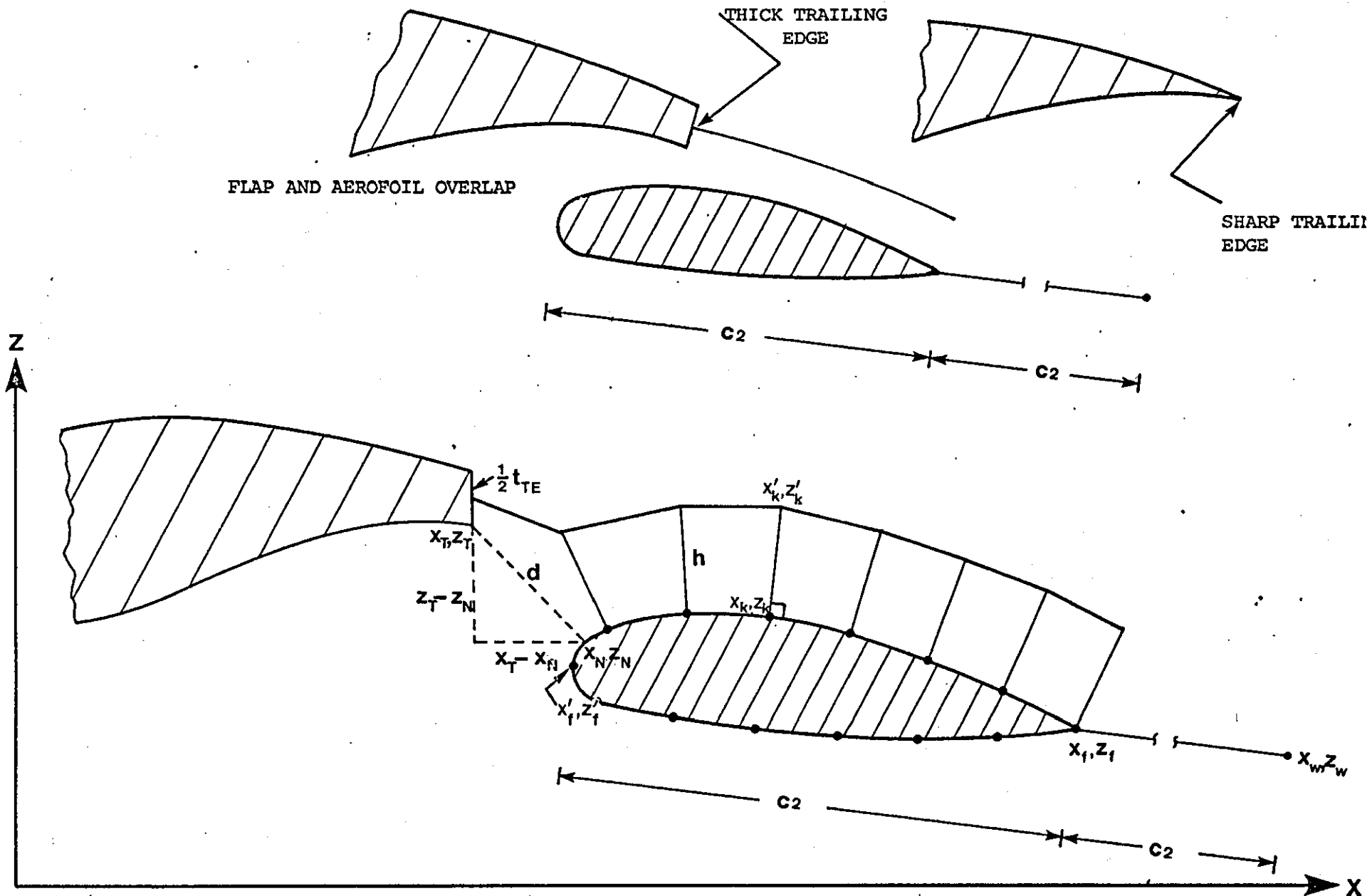


FIGURE A11-1 : Wake Geometry

APPENDIX 12

COMPUTER PROGRAM TO CALCULATE THE

CONFLUENT BOUNDARY LAYER

```

1) DIMENSION A(10,10),B(10)
2) DO 10 I=1,8
3) B(I)=0.0
4) DO 10 J=1,8
5) A(I,J)=0.0
6) 10 CONTINUE
7) A(1,1)=(2.*T5-D5+DS5)/UM
8) A(1,2)=1.
9) A(2,1)=(3.*DSS5-2.*D5+2.*DS5)/UM
10) A(2,3)=1.
11) UMW=UM-UW
12) UMM3=UM*SM3
13) UWM3=UW*SM3
14) A(3,1)=BJ*(2.*UM-UW-4.*UMM3+3.*UWM3+2.*UMW*SM5)+UMW*(D5-DS5)
15) A(3,4)=UM*UMW
16) A(3,5)=-A(3,4)
17) A(3,6)=BJ*(2.*UMM3-UWM3-2.*UMW*SM5)
18) UWE=UW-UE
19) A(4,1)=UWE*(BJ-BJ*SM3+D5-DS5)
20) A(4,4)=UM*UWE
21) A(4,5)=-A(4,4)
22) A(4,6)=BW*UE*SM1+2.*BW*UWE*SM2+BJ*UWE*SM3
23) A(5,1)=DSS
24) A(5,4)=-(UT/RK+UB*.5)
25) A(5,5)=UM
26) A(5,7)=-(D5/RK)
27) A(5,8)=-(D5*.5)
28) A(6,1)=2.*UM*T5+D5*A(5,4)
29) A(6,2)=UM*UM
30) A(6,4)=-A(6,2)*T5/D5
31) A(6,7)=-D5*(UM-RK1*UB-4.*UT/RK)/RK
32) A(6,8)=-D5*(UM*.5-RK1*UT/RK-.75*UB)
33) A(7,1)=3.*UM*UM*DSS5+(9./8.)*D5*UB*UB-2.*D5*UM*UB-4.*D5*UM*UT/RK+
34) +.0*D5*RK1*UT*UB/RK+6.*D5*UT*UT/RK/RK
35) A(7,3)=UM*UM*UM
36) A(7,4)=-A(7,3)*DSS5/D5
37) A(7,7)=-D5*(2.*UM*UM-3.*RK1*UM*UB-12.*UM*UT/RK+3.*RK2*UB*UB+6.*RK3
38) +*UT*UB/RK+18.*UT*UT/RK/RK)/RK
39) A(7,8)=-D5*(15.*UB*UB/16.-9.*UM*UB/4.+UM*UM-3.*RK1*UM*UT/RK+
40) +6.*RK2*UT*UB/RK+3.*RK3*UT*UT/RK/RK)
41) A(8,1)=1.
42) A(8,4)=-UT/RK/D5
43) A(8,7)=-((UM-UB)/UT+1./RK)
44) A(8,8)=-1.
45) Y1=ALOG(RE*UM*T5)
46) H5=DS5/T5
47) A100=ALOG(1.385)-45.79*ALOG(Y1)+17.21*Y1-.918*H5-.743*Y1*Y1
48) A200=ALOG(.889)+23.*ALOG(10.)-158.7*ALOG(Y1)+48.55*Y1-.636*H5-1.8:
49) +*Y1*Y1
50) A300=ALOG(.943)+16.*ALOG(10.)-114.6*ALOG(Y1)+35.68*Y1-1.819*
51) +H5-1.365*Y1*Y1
52) FUN1=EXP(A100)
53) FUN2=EXP(A200)*2.
54) FUN3=EXP(A300)
55) B(1)=-UE*DUE*D5/UM/UM+FUN1
56) B(2)=-2.*UE*DUE*(D5-DS5)/(UM*UM)-2.*(FUN3-FUN1)+FUN2
57) FUN4=.3*FUN3
58) DBJ=.17*UMW/(UM+UW)
59) B(3)=UE*DUE*BJ+UM*UM*FUN4-UM*UM*(FUN3-FUN1)-UMW*(UM-2.*UMM3+UWM3+
60) +UMW*SM5)*DBJ
61) DBW=.185*(UE-UW)/(UE+UW)
62) B(4)=BW*DUE*(UE*SM1-2.*UWE*SM1+2.*UWE*SM2)-UM*UM*FUN4+UWE*(UMW*SM3
63) -UM)*DBJ-UWE*(UE*SM1+UWE*SM2)*DBW
64) CALL EXIT

```

PROGRAMME TO CALCULATE MAIN REGION 2

```

(1) DIMENSION A(10,10),B(10)
(2) DO 10 I=1,7
(3) B(I)=0.0
(4) DO 10 J=1,7
(5) A(I,J)=0.0
(6) 10 CONTINUE
(7) A(1,1)=(2.*T5-D5+DS5)/UM
(8) A(1,2)=1.
(9) A(2,1)=(3.*DSS5-2.*D5+2.*DS5)/UM
(10) A(2,3)=1.
(11) UW=UE
(12) UMW=UM-UW
(13) UMM3=UM*SM3
(14) UWM3=UW*SM3
(15) A(3,1)=BJ*(2.*UM-UW-4.*UMM3+3.*UWM3+2.*UMW*SM5)+UMW*(D5-DS5)
(16) A(3,4)=UM*UMW
(17) A(3,5)=-A(3,4)
(18) A(4,1)=DSS5
(19) A(4,4)=- (UT/RK+UB*.5)
(20) A(4,5)=UM
(21) A(4,6)=- (D5/RK)
(22) A(4,7)=- (D5*.5)
(23) A(5,1)=2.*UM*T5+D5*A(5,4)
(24) A(5,2)=UM*UM
(25) A(5,4)=-A(5,2)*T5/D5
(26) A(5,6)=-D5*(UM-RK1*UB-4.*UT/RK)/RK
(27) A(5,7)=-D5*(UM*.5-RK1*UT/RK-.75*UB)
(28) A(6,1)=3.*UM*UM*DSS5+(9./8.)*D5*UB*UB-2.*D5*UM*UB-4.*D5*UM*UT/RK+
(29) +.0*D5*RK1*UT*UB/RK+6.*D5*UT*UT/RK/RK
(30) A(6,3)=UM*UM*UM
(31) A(6,4)=-A(6,3)*DSS5/D5
(32) A(6,6)=-D5*(2.*UM*UM-3.*RK1*UM*UB-12.*UM*UT/RK+3.*RK2*UB*UB+6.*RK1
(33) +*UT*UB/RK+18.*UT*UT/RK/RK)/RK
(34) A(6,7)=-D5*(15.*UB*UB/16.-9.*UM*UB/4.+UM*UM-3.*RK1*UM*UT/RK+
(35) +6.*RK2*UT*UB/RK+3.*RK3*UT*UT/RK/RK)
(36) A(7,1)=1.
(37) A(7,4)=-UT/RK/D5
(38) A(7,6)=-((UM-UB)/UT+1./RK)
(39) A(7,7)=-1.
(40) Y1=ALOG(RE*UM*T5)
(41) H5=DS5/T5
(42) A100=ALOG(1.385)-45.79*ALOG(Y1)+17.21*Y1-.918*H5-.743*Y1*Y1
(43) A200=ALOG(.887)+23.*ALOG(10.)-158.7*ALOG(Y1)+48.55*Y1-.636*H5-1.1
(44) +*Y1*Y1
(45) A300=ALOG(.943)+16.*ALOG(10.)-114.6*ALOG(Y1)+35.68*Y1-1.819*
(46) +H5-1.365*Y1*Y1
(47) FUN1=EXP(A100)
(48) FUN2=2.*EXP(A200)
(49) FUN3=EXP(A300)
(50) B(1)=-UE*DUE*D5/UM/UM+FUN1
(51) B(2)=-2.*UE*DUE*(D5-DS5)/(UM*UM)-2.*(FUN3-FUN1)+FUN2
(52) FUN4=.3*FUN3
(53) DBJ=.17*UMW/(UM+UW)
(54) B(3)=BJ*DUE*(UE-2.*UMM3+UE*SM3+2.*(UM-UE)*SM5-UM*UM*(FUN3-FUN1)
(55) +-(UM-UE)*(UM-2.*UMM3+UE*SM3+(UM-UE)*SM5)*DBJ
(56) CALL EXIT
(57) END

```

```

(0001) C*****
(0002) C   Calculates confluent boundary layer parameters
(0003) C   for Main region 1 & 2
(0004) C*****
(0005) C
(0006)     DIMENSION BJM(10),BWM(10)
(0007)     D4=D5+BJ+RK2*BW
(0008)     BJM(I)=DBJ*DX+BJM(I-1)
(0009)     BWM(I)=DBW*DX+BWM(I-1)
(0010)     DS=DS5+DSJ+DSW
(0011)     T=T5+TJ+TW
(0012)     SM1=1.178
(0013)     SM2=.786
(0014)     SM3=.5644
(0015)     SM5=.4331
(0016)     UMDE=UM/UE
(0017)     UWDE=UW/UE
(0018)     UC=UMDE-UWDE
(0019)     DSJ=BJ*(1.-UMDE+UC*SM3)
(0020)     TJ=BJ*(UMDE*(1.-UMDE)-(1-UMDE)*UC*SM3+UMDE*UC*SM3-UC*UC*SM5)
(0021)     DSW=BW*SM1*(1-UWDE)
(0022)     TW=BW*((1.-UWDE)*SM1-(1.-UWDE)**2*SM2)
(0023)     CF=2.*(UT/VC)**2
(0024)     CFB=2.*UT*UT
(0025)     CALL EXIT
(0026)     END

```

