1972-06

A study of supersonic cascade flutter

Chalkley, Henry George

https://hdl.handle.net/10945/16389

Downloaded from NPS Archive: Calhoun
A STUDY OF SUPERSONIC
CASCADE FLUTTER

Henry George Chalkley
A Study of Supersonic Cascade Flutter

by

Henry George Chalkley

Thesis Advisor: M. F. Platzer

June 1972

Approved for public release; distribution unlimited.
A Study of Supersonic
Cascade Flutter

by

Henry George Chalkley
Lieutenant, United States Navy
M.S., Naval Postgraduate School, 1971

Submitted in partial fulfillment of the
requirements for the degree of

AERONAUTICAL ENGINEER

from the

NAVAL POSTGRADUATE SCHOOL
June 1972
ABSTRACT

Supersonic flow past oscillating flat plate cascades with supersonic leading-edge locus is analysed using a linearized method of characteristics valid for arbitrary frequencies and an elementary analytical theory valid only for low frequencies of oscillation. These two methods are extensions of previous work by Teipel and Sauer for the single airfoil in an unbounded supersonic flow to the case of airfoils oscillating in cascade. Included is the determination of pressure distribution and both a two-degree-of-freedom (bending and torsion) flutter analysis and a single-degree-of-freedom (torsion) flutter analysis. Numerically determined flutter boundaries are presented for various primary parameters such as, Mach number, solidity, stagger angle, density ratio, structural damping coefficient, and elastic axis position. Also, results are presented for the related problem of supersonic wind tunnel interference (including the effect of tunnel porosity) and airfoil-airfoil interference.
TABLE OF CONTENTS

I. INTRODUCTION -------------------------------------------- 8

II. PROBLEM FORMULATION ------------------------------------ 11

III. METHOD OF CHARACTERISTICS ----------------------------- 19

IV. ELEMENTARY THEORY FOR SLOWLY OSCILLATING CASCADES ----- 38

V. FLUTTER ANALYSIS ---------------------------------------- 49

VI. COMPUTATIONAL PROCEDURE ------------------------------- 59

VII. RESULTS ----------------------------------------------- 69
     A. COMPARISON OF ELEMENTARY THEORY WITH MILES -------- 69
     B. COMPARISON OF ELEMENTARY THEORY WITH LANE ------- 71
     C. COMPARISON OF ELEMENTARY THEORY WITH DRAKE ----- 71
     D. AIRFOIL-AIRFOIL INTERFERENCE ---------------------- 73
     E. COMPARISON OF THE METHOD OF CHARACTERISTICS WITH ELEMENTARY THEORY ------------------ 75
     F. COMPARISON OF THE METHOD OF CHARACTERISTICS WITH GARRICK-RUBINOW ---------------- 79
     G. METHOD OF CHARACTERISTICS RESULTS --------------- 79

VIII. CONCLUSIONS AND RECOMMENDATIONS ----------------------- 88

APPENDIX A - COMPUTER PROGRAM VARIABLES ------------------- 90

APPENDIX B - FLOW DIAGRAM - PROGRAM A --------------------- 97

COMPUTER PROGRAM A ---------------------------------------- 110

COMPUTER PROGRAM B ---------------------------------------- 133

LIST OF REFERENCES ----------------------------------------- 153

INITIAL DISTRIBUTION LIST ---------------------------------- 155

FORM DD 1473 --------------------------------------------- 156
<table>
<thead>
<tr>
<th>Symbol</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>$d \cot \alpha$</td>
</tr>
<tr>
<td>B</td>
<td>stagger, $d \tan \beta$</td>
</tr>
<tr>
<td>C</td>
<td>Teipel complex velocity of sound perturbation amplitude</td>
</tr>
<tr>
<td>c</td>
<td>velocity of sound perturbation</td>
</tr>
<tr>
<td>$\bar{c}$</td>
<td>local velocity of sound</td>
</tr>
<tr>
<td>$c_\infty$</td>
<td>freestream velocity of sound</td>
</tr>
<tr>
<td>$\hat{c}$</td>
<td>blade chord</td>
</tr>
<tr>
<td>$\cot \alpha$</td>
<td>$\sqrt{M^2 - 1}$</td>
</tr>
<tr>
<td>$C_m$</td>
<td>pitching moment coefficient</td>
</tr>
<tr>
<td>$c_{m\theta}$</td>
<td>pitching moment coefficient amplitude (real)</td>
</tr>
<tr>
<td>$c_{m\theta}^*$</td>
<td>pitching moment coefficient amplitude (imaginary) or pitch damping coefficient</td>
</tr>
<tr>
<td>$C_p$</td>
<td>pressure coefficient amplitude</td>
</tr>
<tr>
<td>$C_h$</td>
<td>blade bending stiffness</td>
</tr>
<tr>
<td>$C_\theta$</td>
<td>blade torsional stiffness</td>
</tr>
<tr>
<td>d</td>
<td>blade distance</td>
</tr>
<tr>
<td>g</td>
<td>structural damping</td>
</tr>
<tr>
<td>h</td>
<td>blade plunging oscillation</td>
</tr>
<tr>
<td>$h_0$</td>
<td>blade plunging oscillation amplitude</td>
</tr>
<tr>
<td>$I_\theta$</td>
<td>blade moment of inertia</td>
</tr>
<tr>
<td>k</td>
<td>reduced frequency $\left( \frac{\omega \hat{c}}{u_\infty} \right)$</td>
</tr>
<tr>
<td>L</td>
<td>aerodynamic lift</td>
</tr>
<tr>
<td>M</td>
<td>Mach number</td>
</tr>
<tr>
<td>$M_b$</td>
<td>blade mass per unit span</td>
</tr>
<tr>
<td>$M_\theta$</td>
<td>aerodynamic pitching moment</td>
</tr>
</tbody>
</table>
\( m \) molecular weight

\( P \) non-dimensional complex pressure perturbation amplitude

\( p \) pressure

\( p_\infty \) freestream pressure

\( r_0 \) blade radius of gyration \( \left( \sqrt{\frac{4I_\theta}{M_b c^2}} \right) \)

\( Q \) universal gas constant

\( s \) entropy

\( S_\theta \) blade static moment about the elastic axis per unit span

\( T \) temperature

\( U \) Teipel complex streamwise velocity perturbation amplitude

\( U_F \) freestream flutter speed

\( u \) streamwise velocity perturbation

\( u \) internal energy in Eq. (59)

\( \bar{u} \) local streamwise velocity

\( u_\infty \) freestream velocity

\( V \) Teipel complex normal velocity perturbation amplitude

\( v \) normal velocity perturbation

\( X \) frequency rate \( \left[ (\frac{\omega_\theta}{\omega})^2 \right] \)

\( x_\theta \) distance between blade center of gravity and elastic axis \( \left( \frac{S_\theta}{M_b} \right) \)

\( x_0 \) elastic axis position

\( x,y \) Cartesian coordinates

\( \alpha \) Mach angle

\( \beta \) stagger angle (complimentary)
\( \gamma \) ratio of specific heats

\( \delta \) interblade phase angle

\( \Phi \) complex perturbation velocity potential

\( \phi \) complex perturbation velocity potential amplitude

\( \theta \) angle of attack

\( \theta_0 \) amplitude of pitch oscillation

\( \mu \) blade density-parameter \( \left( \frac{M_b}{\rho_\infty \hat{c}^2} \right) \)

\( \omega \) frequency of oscillation

\( \omega_F \) flutter frequency

\( \omega_{\theta} \) torsional natural frequency \( \left( \frac{C_{\theta}}{I_{\theta}} \right) \)

\( \omega_h \) bending natural frequency \( \left( \frac{C_h}{M_b} \right) \)

\( \Omega_{\theta} \) factor \( (\mu r_{\theta}^2) \)

\( \Omega \) factor \( [\mu (\frac{\omega_h}{\omega_{\theta}})^2] \)

\( \rho \) local density

\( \rho_\infty \) freestream density

\( \xi, \eta \) characteristic coordinates

\( \sigma \) wind tunnel porosity

\( \chi \) defined in Eq.(118)

\( \psi \) defined in Eq.(118)

\( \nu \) grid fineness ratio
ACKNOWLEDGMENT

This thesis was written with the close support and continual assistance of Associate Professor M. F. Platzer.
I. INTRODUCTION

In order to design an effective supersonic compressor, an analysis of the complex unsteady flow phenomena in this speed regime is necessary. Since the infinite cascade has long been used by the engineer to model two-dimensional compressor flows, the study of oscillating supersonic cascades is important in predicting the flutter characteristics or dynamic response of these compressors. Moreover, the basic understanding of the flow gained by such a study can lead to better design criteria.

Two basic cascade configurations can be distinguished at the outset; i.e., cascades with either subsonic or supersonic leading-edge locus. Although the former is the case of primary interest in current research, in this paper the case of supersonic leading-edge locus is treated in order to form a basis for extension to the more complicated case of subsonic leading-edge locus. This method of approach also has the advantage that the problem of oscillatory wind tunnel interference, already analysed in previous work, is contained in the theory as a special case.

The interference problem of linearized supersonic flow past airfoils oscillating between solid wind tunnel walls was considered by Miles (1956) who derived a solution using Laplace transform techniques. Drake (1956) treated this case for wind tunnels with free jet boundaries and later Drake (1957) gave a solution to this interference problem for
porous-wall wind tunnels also using Laplace transform methods. In an extension of Miles' work, Lane (1957) presented a solution for supersonic flow past oscillating cascades with supersonic leading-edge locus again using Laplace transform techniques. Further work in this area was done by Hamamoto (1962). Using Teipel's (1962) method of characteristics approach, Platzer and Pierce (1970) made an analysis of oscillatory supersonic wind tunnel interference and with the help of the high speed computer were able to predict the pressure distribution along an airfoil oscillating with arbitrary frequency in solid, free jet, or porous-wall wind tunnels. Platzer (1971) will present an elementary analysis of porous-wall wind tunnel interference effects which generalizes Sauer's (1950) solution for a single airfoil oscillating at low frequency in an unbounded linearized supersonic flow. Platzer and Chalkley (1972) further extended this solution to form an elementary theory that together with a method of characteristics procedure (based on Platzer and Pierce, 1970) is used to analyse the supersonic flow past oscillating flat plate cascades having supersonic axial velocities but otherwise arbitrary stagger angle.

In this thesis the detailed description of both the elementary theory and the method of characteristics that was not possible in Platzer and Chalkley (1972) is given. With the aid of the high speed computer, the method of characteristics procedure is used to predict the pressure distributions and aerodynamic forces and pitching moments on a typical cascade.
blade. Both a two-degree-of-freedom (torsion and bending) flutter analysis and a one-degree-of-freedom (torsion) flutter analysis are presented, with numerical results given for the case of the torsional flutter. The elementary theory (valid only for low frequencies of oscillation) is used to provide a convenient check of the characteristics approach as well as to easily predict a number of dynamic instability boundaries for a slowly oscillating airfoil subjected to interference from supersonic wind tunnel walls in one case and from a larger airfoil in close proximity in yet another case.
II. PROBLEM FORMULATION

The cascade is considered to be one of two-dimensional flat plates with each blade performing equal low amplitude, simple harmonic oscillations of the same mode with the same interblade phase angle. As shown in Figure 1, it is aligned in the x-y coordinate system such that one blade lies along the x-axis with its leading-edge at the origin.

![Figure 1](image)

The cascade is further considered to have supersonic leading-edge locus and, except for this restriction, arbitrary stagger, such that,

\[ \tan \beta \leq \cot \alpha \]  

(1)

With this assumption, the flow between any two adjacent blades can be used to describe the flow through the entire cascade. Since disturbances cannot travel upstream of the
Mach lines from the leading-edge of a blade, interference from other than adjacent blades is not possible in the supersonic flow between them. Hence, only the flow between the blade along the x-axis and the one above it is considered. This flow is assumed to be the non-viscous flow of a perfect gas governed by the continuity equation,

$$\frac{D\rho}{Dt} + \rho \left( \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} \right) = 0 \quad (2)$$

the Euler equations,

$$\frac{Du}{Dt} + \frac{1}{\rho} \frac{\partial p}{\partial x} = 0 \quad (3a)$$

$$\frac{Dv}{Dt} + \frac{1}{\rho} \frac{\partial p}{\partial y} = 0 \quad (3b)$$

and the energy equation,

$$\frac{Ds}{Dt} = 0 \quad (4)$$

Because of the assumption of small amplitude oscillations of the cascade blades, all flow quantities are considered to be small perturbations linearly superimposed on freestream quantities. The velocities are written as

$$\bar{u} = u_\infty + u \quad (5a)$$

$$\bar{c} = c_\infty + c \quad (5b)$$

while the pressure and density perturbations are written as

$$\Delta p = p - p_\infty \quad (6a)$$

$$\Delta \rho = \rho - \rho_\infty \quad (6b)$$

With the assumption of isentropic flow, the local velocity of sound can be defined as,

$$c^2 = \frac{dp}{d\rho} \quad (7)$$
further,

\[ \frac{P}{\rho^\gamma} = \text{constant} \]  \hspace{1cm} (8)

Taking the total differential,

\[ \frac{1}{\rho^\gamma} \frac{dP}{d\rho} - \gamma \rho \frac{1}{\rho^{\gamma+1}} d\rho = 0 \]  \hspace{1cm} (9)

or

\[ \frac{dP}{d\rho} = \gamma \frac{P}{\rho} \]  \hspace{1cm} (10)

hence,

\[ \frac{c^2}{\gamma} = \gamma \frac{P}{\rho} \]  \hspace{1cm} (11)

The flow boundary condition along the blade surfaces [as shown in Ch. 5, Bisplinghoff, Ashley, Halfman (1955) for a simple airfoil] is as follows: If the surface over which the fluid flows has the equation,

\[ F(x, y, t) = 0 \]  \hspace{1cm} (12)

particles in contact with the surface must have the same normal velocity as the surface. Stated differently, the rate of change of F is zero when the motion of a particular fluid element is followed along the surface, or

\[ \frac{DF}{Dt} = 0 \]  \hspace{1cm} (13)

If an arbitrary, thin, two-dimensional cascade blade is considered, the equation of the upper surface can be written,

\[ F_u(x, y, t) = y - y_u(x, t) = 0 \]  \hspace{1cm} (14)

where \( y = 0 \) is the mean camber line and \( y_u \) is the distance from the mean camber line to the upper surface. Likewise, the equation of the lower surface can be written,
\[
F_L (x,y,t) = y - y_L (x,t) = 0 \quad (15)
\]

At \(y = y_u\)

\[
\frac{DF_u}{Dt} = - \frac{\partial y_u}{\partial t} - \bar{u} \frac{\partial y_u}{\partial x} + \nu = 0 \quad (16)
\]

and at \(y = y_L\)

\[
\frac{DF_L}{Dt} = - \frac{\partial y_L}{\partial t} - \bar{u} \frac{\partial y_L}{\partial x} + \nu = 0 \quad (17)
\]

since

\[
\frac{\partial y_u}{\partial y} = \frac{\partial y_L}{\partial y} = 1 \quad (18)
\]

Thus, the normal flow velocity can be written,

\[
v = \frac{\partial y_u}{\partial t} + \bar{u} \frac{\partial y_u}{\partial x} \quad \text{at} \quad y = y_u (x,t) \quad (19a)
\]

\[
v = \frac{\partial y_L}{\partial t} + \bar{u} \frac{\partial y_L}{\partial x} \quad \text{at} \quad y = y_L (x,t) \quad (19b)
\]

By applying the assumption of linear perturbations on the freestream, the normal velocity equations become

\[
v = \frac{\partial y_u}{\partial t} + u_\infty \frac{\partial y_u}{\partial x} + u \frac{\partial y_u}{\partial x} \quad \text{at} \quad y = y_u (x,t) \quad (20a)
\]

\[
v = \frac{\partial y_L}{\partial t} + u_\infty \frac{\partial y_L}{\partial x} + u \frac{\partial y_L}{\partial x} \quad \text{at} \quad y = y_L (x,t) \quad (20b)
\]

Since a thin blade is being considered \(y_u\) and \(y_L\) are small, thus the equations become, cancelling the higher order terms,

\[
v = \frac{\partial y_u}{\partial t} + u_\infty \frac{\partial y_u}{\partial x} \quad \text{at} \quad y = y_u (x,t) \quad (21a)
\]

\[
v = \frac{\partial y_L}{\partial t} + u_\infty \frac{\partial y_L}{\partial t} \quad \text{at} \quad y = y_L (x,t) \quad (21b)
\]

This normal velocity can be further written as a Taylor series expansion of the normal flow velocity at \(y = 0\). Thus,
\[
v(x, y_u, t) = v(x, 0^+, t) + y_u \frac{\partial v(x, 0^+, t)}{\partial y}
+ \frac{y_u^2}{2!} \frac{\partial^2 v(x, 0^+, t)}{\partial y^2} + \ldots \quad (22a)
\]

and
\[
v(x, y_L, t) = v(x, 0^-, t) + y_L \frac{\partial v(x, 0^-, t)}{\partial y}
+ \frac{y_L^2}{2!} \frac{\partial^2 v(x, 0^-, t)}{\partial y^2} + \ldots \quad (22b)
\]

Again with the assumption of a thin blade and small linear perturbations on the freestream: \(y_u, y_L\), and \(v\) are small and higher order terms may again be cancelled from the equations leaving:

\[
v(x, y_u, t) = v(x, 0^+, t) \quad (23a)
\]
\[
v(x, y_L, t) = v(x, 0^-, t) \quad (23b)
\]

Thus,
\[
v(x, y, t) = \frac{\partial y_u}{\partial t} + u_\infty \frac{\partial y_u}{\partial x} \quad \text{at } y = 0^+ \quad (24a)
\]
\[
v(x, y, t) = \frac{\partial y_L}{\partial t} + u_\infty \frac{\partial y_L}{\partial x} \quad \text{at } y = 0^- \quad (24b)
\]

With the assumption of simple harmonic motion of the blade,
\[
y_u(x, t) = y_u(x) e^{i\omega t} \quad (25a)
\]
\[
y_L(x, t) = y_L(x) e^{i\omega t} \quad (25b)
\]

and the normal velocity equations can then be written,
\[
v(x, y, t) = [i\omega y_u(x) + u_\infty \frac{\partial y_u}{\partial x}] e^{i\omega t} \quad \text{at } y = 0^+ \quad (26a)
\]
\[
v(x, y, t) = [i\omega y_L(x) + u_\infty \frac{\partial y_L}{\partial x}] e^{i\omega t} \quad \text{at } y = 0^- \quad (26b)
\]

Since the flow is identical between any two adjacent blades,
\[ v(x,y,t) = [i\omega y_L(x) + u_\infty \frac{\partial y_L}{\partial x}] e^{i\omega t} \quad \text{at } y = d^- \quad (27) \]

In plunge, the lower blade oscillation about \( y = 0 \) is,
\[ y = -h_0 e^{i\omega t} \quad (28) \]
hence,
\[ v(x,y,t) = [-i\omega h_0] e^{i\omega t} \quad \text{at } y = 0^+ \quad (29) \]
while the upper blade oscillation about \( y = d \) is,
\[ y = -h_0 e^{i(\omega t+\delta)} \quad (30) \]
where \( \delta \) is the interblade phase angle. Hence,
\[ v(x,y,t) = [\omega \sin \delta h_0 - i\omega \cos \delta h_0] e^{i\omega t} \quad (31) \]
at \( y = d^- \).

In pitch, the lower blade oscillation about \( y = 0 \) is,
\[ y = -\theta_0 (x-x_0) e^{i\omega t} \quad (32) \]
hence,
\[ v(x,y,t) = [-u_\infty \theta_0 - i\omega \theta_0 (x - x_0)] e^{i\omega t} \quad (33) \]
at \( y = 0^+ \), while the upper blade oscillation about \( y = d \) is,
\[ y = -\theta_0 (x - B - x_0) e^{i(\omega t+\delta)} \quad (34) \]
hence,
\[ v(x,y,t) = [-\theta_0 \{u_\infty \cos \delta - \omega \sin \delta (x-B-x_0)\} \]
\[ \quad - i\theta_0 \{u_\infty \sin \delta + \omega \cos \delta (x-B-x_0)\}] e^{i\omega t} \quad (35) \]
at \( y = d^- \).

The pressure distribution on the blades can be written in terms of the local sonic velocity as follows: In terms of perturbation quantities, Eq. (11) can be written,
\[ c_\infty^2 + 2c \ c_\infty = \gamma \frac{1}{\rho_\infty} \frac{P_\infty + \Delta P}{(1 + \frac{\Delta \rho}{\rho_\infty})} \] (36)

cancelling the higher order \( c^2 \) term. Since \( \frac{\Delta \rho}{\rho_\infty} \) is less than one, Eq. (36) can be expanded in the geometric series,

\[ \frac{1}{1 + \frac{\Delta \rho}{\rho_\infty}} = 1 - \frac{\Delta \rho}{\rho_\infty} + \frac{\Delta \rho^2}{\rho_\infty^2} - \ldots = 1 - \frac{\Delta \rho}{\rho_\infty} \] (37)

neglecting the higher order terms. Substituting this in Eq. (36) gives,

\[ 2c \ c_\infty = \gamma \frac{1}{\rho_\infty} (1 - \frac{c_\infty^2}{\frac{\Delta \rho}{\Delta p}} \Delta p) \] (38)

Since \( \Delta \rho \) and \( \Delta p \) are small,

\[ \frac{\Delta p}{\Delta \rho} \approx \frac{1}{\frac{dp}{d\rho}} = \frac{1}{c^2} \] (39)

Substituting this in Eq. (38) gives after some algebraic manipulation,

\[ \frac{2}{(\gamma - 1)} c \ c_\infty \left( \frac{1}{1 + \frac{2c}{(\gamma - 1) c_\infty}} \right) = \frac{1}{\rho_\infty} \Delta p \] (40)

Since \( \frac{2c}{(\gamma - 1) c_\infty} \) is small, \( \frac{1}{1 + \frac{2c}{(\gamma - 1) c_\infty}} \) may be expanded in the geometric series,

\[ \frac{1}{1 + \frac{2c}{(\gamma - 1) c_\infty}} = 1 - \frac{2c}{(\gamma - 1) c_\infty} + \left[ \frac{2c}{(\gamma - 1) c_\infty} \right]^2 - \ldots = 1 - \frac{2c}{(\gamma - 1) c_\infty} \] (41)

neglecting higher order terms. Hence, Eq. (40) can be written, neglecting higher order terms,

\[ p - p_\infty = \frac{2}{\gamma - 1} \rho_\infty c_\infty \left( \bar{c} - c_\infty \right) \] (42)
Finally, any non-dimensional quantities so stated in the paper are made such with reference to the blade chord length and the freestream velocity unless otherwise defined.
III. METHOD OF CHARACTERISTICS

In the method of characteristics it is desirable to ascertain the coordinate system across which all possible discontinuities in flow properties may occur. Along this coordinate system the equations of motion of the flow field can then be treated as ordinary differential equations that are solvable by classical or numerical techniques (e.g., finite differences). To obtain the equations of this coordinate system (the characteristic directions) in the \((x,y)\) plane, the equations of motion are written in terms of the arbitrary intersecting coordinates,

\[
\xi = \xi (x,y) \quad (43a)
\]

and

\[
\eta = \eta (x,y) \quad (43b)
\]

If the first derivatives of the dependent variables, \(\bar{u}, \bar{v},\) and \(p\) (or \(\bar{c}\)) with respect to \(\xi\) are made indeterminate across lines of \(\eta = \text{constant}\), and the first derivatives of the dependent variables with respect to \(\eta\) are made indeterminate across lines of \(\xi = \text{constant}\), then any possible discontinuities in the first derivatives will occur across these lines. These lines are then the characteristics and their equations are obtained in the evaluation of the indeterminacies.

Consider, first, two-dimensional, steady flow. The governing equations of motion are the continuity equation,

\[
\frac{\partial (\rho \bar{u})}{\partial x} + \frac{\partial (\rho \bar{v})}{\partial y} = 0 \quad (44)
\]
the Euler equations,

\[ \frac{\partial \tilde{u}}{\partial x} + v \frac{\partial \tilde{u}}{\partial y} + \frac{1}{\rho} \frac{\partial p}{\partial x} = 0 \]  

\[ \frac{\partial \tilde{v}}{\partial x} + v \frac{\partial \tilde{v}}{\partial y} + \frac{1}{\rho} \frac{\partial p}{\partial y} = 0 \]  

and the energy equation,

\[ \frac{\partial s}{\partial x} + v \frac{\partial s}{\partial y} = 0 \]

Along a streamline this equation may be written,

\[ \frac{\partial p}{\partial x} + v \frac{\partial p}{\partial y} = \frac{1}{c^2} [\frac{\partial \tilde{p}}{\partial x} + v \frac{\partial \tilde{p}}{\partial y}] \]

Substituting in Eq. (44), the continuity equation becomes,

\[ \rho \left[ \frac{\partial \tilde{u}}{\partial x} + \frac{\partial \tilde{v}}{\partial y} \right] = -\frac{1}{c^2} \left[ \tilde{u} \frac{\partial p}{\partial x} + v \frac{\partial p}{\partial y} \right] \]

In terms of the arbitrary coordinates, Eqs. (43), the continuity equation becomes,

\[ \rho \xi \xi \tilde{u}_\xi + \rho \xi \eta \nu_\xi + \frac{1}{c^2} \left[ \tilde{u}_\xi + \nu_\xi \right] p_\xi \]

\[ = -\rho \eta_x \tilde{u}_\eta - \rho \eta_y \nu_\eta - \frac{1}{c^2} \left[ \tilde{u} \eta_x + \nu \eta_y \right] p_\eta \]  

In like manner, the Euler equations become,

\[ \rho [\tilde{u}_\xi + \nu_\xi] \tilde{u}_\xi + \xi_x p_\xi = -\rho [\tilde{u}_\eta + \nu_\eta] \tilde{u}_\eta - p \eta_x \]  

\[ \rho [\tilde{u}_\xi + \nu_\xi] \nu_\xi + \xi_y p_\xi = -\rho [\tilde{u}_\eta + \nu_\eta] \nu_\eta - p \eta_y \]

Eq. (49) and Eqs. (50) form a system of three simultaneous equations in \( \tilde{u}_\xi \), \( \nu_\xi \), and \( p_\xi \). Solving for \( p_\xi \) by Cramer's rule gives a ratio of two determinants. The denominator is the determinant of the matrix of coefficients of \( \tilde{u}_\xi \), \( \nu_\xi \), and \( p_\xi \) formed by Eq. (49) and Eqs. (50), while the numerator is
this determinant with the right hand side of the equations substituted for the coefficients of $p_\xi$.

Since across lines of $\eta = \text{constant}$, $p_\xi$ is indeterminate, both the numerator and the denominator must equal zero. Setting the denominator equal to zero and expanding the determinant gives,

$$[ar{u}\bar{\xi}_x + v\bar{\xi}_y] [(\bar{u}^2 - \bar{c}^2) \xi_x^2 + 2\bar{u}\bar{v} \xi_x \xi_y + (v^2 - \bar{c}^2) \xi_y^2] = 0$$

(51)

The solutions to this equation give the equations of all three characteristics in the physical $(x,y)$ plane. These are,

$$\frac{dy}{dx} = \frac{v}{u} = \tan \zeta \quad (52a)$$

$$\frac{dy}{dx} = \tan (\zeta \pm \alpha) \quad (52b)$$

where $\zeta$ is the angle the streamline makes with the $x$-axis and $\alpha$ is the Mach angle. All three characteristic directions are obtained from Eq. (51) because $\xi$ and $\eta$ are arbitrary and interchangeable: If Eq. (49) and Eqs. (50) were written such that derivatives with respect to $\eta$ are the unknowns, making $p_\eta$ indeterminate across lines of $\xi = \text{constant}$ would result in solutions identical to Eqs. (52), a priori. Eq. (52a) is merely the equation of a streamline while Eqs. (52b) are the equations of the right and left-running Mach lines. Traditionally, Eqs. (52b) are the equations of $\xi = \text{constant}$ and $\eta = \text{constant}$.

The compatibility relation which relates changes in $\bar{u}$, $v$, and $p$ along the $(\xi,\eta)$ characteristics can be obtained by
setting the numerator of $p_\xi$ equal to zero. Expanding the
determinant and solving the resultant equation yields after
some algebraic manipulation,

$$v \frac{\partial u}{\partial \eta} - u \frac{\partial v}{\partial \eta} + \frac{1}{\rho} \cot \alpha \frac{\partial p}{\partial \eta} = 0$$  \hspace{1cm} (53)

This derivation is shown in detail in Chapter 8 of Oswatitsch
(1956). These relations must be satisfied along the ($\xi, \eta$)
characteristics.

The physical significance of these characteristics is
that they act as boundaries across which any discontinuities
will occur. In this flow field where streamlines and Mach
lines are the characteristics, a discontinuity is readily
apparent across the Mach line emanating from the leading-edge
of a slender airfoil or across a slip plane (streamline)
emanating from a Mach stem. Characteristics can further be
described as information carriers in that along them all dis-
turbances propagate and all flow quantities can be determined
given their values in the domain of dependence. It should
also be noted that the characteristic directions are com-
pletely independent of the type of coordinate system used to
describe the flow. They are determined solely by the physical
nature of the flow field as expressed in the equations of
motion.

The problem of two-dimensional, unsteady flow over a flat
plate as shown by Teipel (1962) can be considered in the same
manner as before. The governing equations of motion [Eq. (2),
Eq. (3), and Eq. (4)] are written in an arbitrary coordinate
system across which variation of flow properties are indeter-
minate. The evaluation of these indeterminacies will again
give the characteristic directions of these coordinates.

The arbitrary coordinates are given in Eqs. (48). Eq.(2),
Eq. (3), and Eq. (4) can be rewritten in a form more suitable
for transformation by using the local sonic velocity as an
unknown. Assuming that air is a perfect gas, Eq. (11) can
be written,

\[ \bar{c}^2 = \gamma \frac{R}{m} T \]  (54)

where \( R \) is the universal gas constant and \( m \) is the molecular
weight. Taking the total differential and dividing by \( \bar{c}^2 \)
gives,

\[ 2 \frac{d\bar{c}}{\bar{c}} = \frac{\gamma}{m} \frac{dT}{\bar{c}^2} \]  (55)

Substituting Eq. (54) into this gives,

\[ 2 \frac{d\bar{c}}{\bar{c}} = \frac{dT}{T} \]  (56)

In like manner, the total differential of Eq. (11) is,

\[ 2\bar{c} \frac{d\bar{c}}{\bar{c}} = \gamma \frac{P}{\rho} \frac{dP}{P} - \gamma \frac{P}{\rho} \frac{dP}{\rho} \]  (57)

and from Eq. (11) again,

\[ 2 \frac{d\bar{c}}{\bar{c}} = \frac{dP}{P} - \frac{dP}{\rho} \]  (58)

From the First Law of Thermodynamics and the definition of
entropy,

\[ Tds = du - \frac{P}{\rho^2} d\rho \]  (59)
Since $du = c_v \cdot dt$, this may be written as,

$$Tds = c_v dT - RT \frac{d\rho}{\rho}$$

(60)

where $R$ is the gas constant $(c_p - c_v)$ and $c_p$ and $c_v$ are the specific heats of the gas at constant pressure and volume, respectively. Dividing Eq. (61) by $Tc_v$, substituting from Eq. (56) and Eq. (59), and then taking the substantial derivative of $\frac{1}{c_v}S$ with respect to time gives,

$$\frac{1}{c_v} \frac{D_s}{Dt} = 2\gamma \frac{1}{c} \frac{Dc}{Dt} - (\gamma-1) \frac{1}{p} \frac{D\rho}{Dt} = 0$$

(62)

since entropy is held constant. This can be written substituting from Eq. (11),

$$\frac{Dp}{Dt} = \frac{2}{\gamma-1} \rho c \frac{Dc}{Dt}$$

(63)

Introducing the small perturbation assumption and substituting from Eq. (7) and Eq. (63), Eq. (2) can be written, cancelling higher order terms,

$$\frac{2}{\gamma-1} \frac{\partial \bar{c}}{\partial t} + \frac{2}{\gamma-1} u_\infty \frac{\partial \bar{c}}{\partial x} + c_\infty \frac{\partial u}{\partial x} + c_\infty \frac{\partial v}{\partial y} = 0$$

(64)

This is the form of the continuity equation that will be used in the coordinate transformation. Again dividing Eq. (61) by $Tc_v$, substituting from Eq. (56) and Eq. (58), and then taking the partial derivative with respect to $x$ (holding entropy constant) gives,

$$\frac{1}{\rho} \frac{\partial \bar{p}}{\partial x} = \frac{2}{\gamma-1} \bar{c} \frac{\partial \bar{c}}{\partial x}$$

(65)

Substituting in the first Euler equation, Eq. (3) becomes, cancelling higher order terms,
\[
\begin{align*}
\frac{\partial u}{\partial t} + u_\infty \frac{\partial u}{\partial x} + \frac{2}{\gamma - 1} c_\infty \frac{\partial c}{\partial x} &= 0 \\
\frac{\partial v}{\partial t} + u_\infty \frac{\partial v}{\partial x} + \frac{2}{\gamma - 1} c_\infty \frac{\partial c}{\partial y} &= 0
\end{align*}
\]  

(66)

In like manner, the second Euler equation, Eq. (4), becomes,

\[
\frac{\partial v}{\partial t} + u_\infty \frac{\partial v}{\partial x} + \frac{2}{\gamma - 1} c_\infty \frac{\partial c}{\partial y} = 0
\]

(67)

The continuity equation and the Euler equations thus form a system of three simultaneous equations in \( \bar{u}, \bar{v}, \) and \( \bar{c} \). With assumption of simple harmonic motion, Teipel (1962) introduced his amplitude functions,

\[
\begin{align*}
U(x,y)e^{i\omega t} &= \frac{\bar{u} - u_\infty}{u_\infty} \\
V(x,y)e^{i\omega t} &= \frac{1}{\sqrt{M^2 - 1}} \frac{\bar{v}}{u_\infty} \\
C(x,y)e^{i\omega t} &= \frac{2}{\gamma - 1} \frac{1}{M^2} \frac{\bar{c} - c_\infty}{c_\infty}
\end{align*}
\]

(68) (69) (70)

where \( U, V, \) and \( C \) are complex.

Substituting in Eq. (64), Eq. (66), and Eq. (67) gives,

\[
\begin{align*}
\frac{\partial U}{\partial x} + \sqrt{M^2 - 1} \frac{\partial V}{\partial y} + M^2 \frac{\partial C}{\partial x} + ikM^2 C &= 0 \\
\frac{\partial U}{\partial x} + \frac{\partial C}{\partial x} + ikU &= 0 \\
\frac{\partial V}{\partial x} + \frac{1}{\sqrt{M^2 - 1}} \frac{\partial C}{\partial y} + ikV &= 0
\end{align*}
\]

(71) (72) (73)

where \( k \) is the reduced frequency.

Transforming this system into the new coordinate system, \( \xi(x,y) \) and \( \eta(x,y) \), and considering now a system of three simultaneous equations in unknowns, \( U_\xi, V_\xi, \) and \( C_\xi \), the equations may be written,

\[
\begin{align*}
\xi_x U_\xi + \cot \alpha \xi_y V_\xi + M^2 \xi_x C_\xi &= -U_\eta - \cot \alpha V_\eta \eta_y M^2 C_\eta \eta_x - ikM^2 C \\
\xi_x U_\xi + \xi_x C_\xi &= -U_\eta \eta_x - \eta_\xi \eta_x - ikU
\end{align*}
\]

(74) (75)
\[ \xi_x V_\xi \tan \alpha \xi_y C_\xi = -V, \eta_x - \tan \alpha \eta_y - ikV. \quad (76) \]

As before \( U_\xi, V_\xi, \) and \( C_\xi \) are made indeterminate across lines of \( \eta = \) constant. This forces any discontinuities in \( U, V, \) and \( C \) to lie across lines of \( \eta = \) constant. One condition of indeterminacy is that the determinant of the matrix of coefficients of \( U_\xi, V_\xi, \) and \( C_\xi \) in the above equations be zero. If this determinant is expanded and then set equal to zero, the resulting equation is,

\[ -\xi_x^3 - \xi_x [\xi_y^2 - M^2 \xi_x^2] = 0 \quad (77) \]

As before the solutions to this equation give the three characteristic directions in the physical \((x,y)\) plane:

\[ \frac{dy}{dx} = 0 \quad (78a) \]

\[ \frac{dy}{dx} = \pm \tan \alpha \quad (78b). \]

These coincide with the results of the steady flow analysis when it is remembered that the small perturbation assumption forces the streamlines to be parallel with the \( x \)-axis \((\zeta = 0)\).

In order to find the compatibility relations of the first derivatives of \( U, V, \) and \( C \) along the characteristics, the equations of motion are written along lines of \( \xi = \) constant, \( \eta = \) constant, and along streamlines, that is, for

\[ \text{str}(x,y) = \text{constant}, \quad \frac{\partial y}{\partial x} = 0 \quad (79a) \]

\[ \zeta (x,y) = \text{constant}, \quad \frac{\partial y}{\partial x} = \frac{1}{\sqrt{M^2 - 1}} \quad (79b) \]
\( \eta(x,y) = \text{constant, } \frac{\partial y}{\partial x} = -\frac{1}{\sqrt{M^2 - 1}} \)  

(79c)

If an arbitrary function of \( x \) and \( y \) in the equations is denoted by \( f \) then, \( \frac{\partial f}{\partial x} \) holding \( \text{str} \) constant becomes,

\[
\left(\frac{\partial f}{\partial x}\right)_{\text{str}} = \frac{\partial f}{\partial x}
\]

(80)

Likewise, \( \frac{\partial f}{\partial x} \) holding \( \xi \) constant becomes,

\[
\left(\frac{\partial f}{\partial x}\right)_{\xi} = \frac{\partial f}{\partial x} + \frac{1}{\sqrt{M^2 - 1}} \frac{\partial f}{\partial y}
\]

(81)

and, \( \frac{\partial f}{\partial x} \) holding \( \eta \) constant becomes,

\[
\left(\frac{\partial f}{\partial x}\right)_{\eta} = \frac{\partial f}{\partial x} - \frac{1}{\sqrt{M^2 - 1}} \frac{\partial f}{\partial y}
\]

(82)

Combining these equations and rewriting,

\[
\frac{\partial f}{\partial x} = \left(\frac{\partial f}{\partial x}\right)_{\text{str}}
\]

(83a)

\[
\frac{\partial f}{\partial x} = \frac{1}{2} \left[ \left(\frac{\partial f}{\partial x}\right)_{\xi} + \left(\frac{\partial f}{\partial x}\right)_{\eta} \right]
\]

(83b)

\[
\frac{\partial f}{\partial y} = \frac{1}{2} \sqrt{M^2 - 1} \left[ \left(\frac{\partial f}{\partial x}\right)_{\xi} - \left(\frac{\partial f}{\partial x}\right)_{\eta} \right]
\]

(83c)

Substituting these relations in the equations of motion gives,

\[
\frac{1}{2} \left[ \left(\frac{\partial U}{\partial x}\right)_{\xi} + \left(\frac{\partial U}{\partial x}\right)_{\eta} \right] + \frac{1}{2} (M^2 - 1) \left[ \left(\frac{\partial V}{\partial x}\right)_{\xi} - \left(\frac{\partial V}{\partial x}\right)_{\eta} \right] + \frac{1}{2} M^2 \left[ \left(\frac{\partial C}{\partial x}\right)_{\xi} + \left(\frac{\partial C}{\partial x}\right)_{\eta} \right] + ikM^2 C = 0
\]

(84a)

\[
\frac{1}{2} \left[ \left(\frac{\partial U}{\partial x}\right)_{\xi} + \left(\frac{\partial U}{\partial x}\right)_{\eta} \right] + \frac{1}{2} \left[ \left(\frac{\partial C}{\partial x}\right)_{\xi} + \left(\frac{\partial C}{\partial x}\right)_{\eta} \right] + ikU = 0
\]

(84b)

\[
\left[ \left(\frac{\partial V}{\partial x}\right)_{\xi} + \left(\frac{\partial V}{\partial x}\right)_{\eta} \right] + ikV = 0
\]

(84c)

Subtracting Eq. (84b) from Eq. (84a) gives an equation that is first added to Eq. (84c) and then subtracted from Eq. (84c). The result is two compatibility relations that must be
satisfied along lines of constant $\xi$ and lines of constant $\eta$.

These are,

\[
\left(\frac{\partial V}{\partial x}\right)_{\xi} + \left(\frac{\partial C}{\partial x}\right)_{\xi} + ik [V + \frac{1}{M^2 - 1} (M^2 C - U)] = 0 \tag{85a}
\]

\[
\left(\frac{\partial V}{\partial x}\right)_{\eta} - \left(\frac{\partial C}{\partial x}\right)_{\eta} + ik [V - \frac{1}{M^2 - 1} (M^2 C - U)] = 0 \tag{85b}
\]

The third compatibility relation is obtained by considering Eq. (84b) along a streamline,

\[
\left(\frac{\partial U}{\partial x}\right)_{\text{str}} + \left(\frac{\partial C}{\partial x}\right)_{\text{str}} + ikU = 0 \tag{86}
\]

If irrotationality is assumed, the equation of irrotationality,

\[
\frac{\partial u}{\partial y} - \frac{\partial v}{\partial x} = 0 \tag{87}
\]

may be substituted for the second Euler equation in the system. Again with the assumption of simple harmonic motion, it may be written in terms of Teipel's amplitude functions,

\[
\frac{\partial U}{\partial y} - \sqrt{M^2 - 1} \frac{\partial V}{\partial x} = 0 \tag{88}
\]

The non-dimensional system of equations can then be written as,

\[
\frac{\partial U}{\partial x} + \sqrt{M^2 - 1} \frac{\partial V}{\partial y} + M^2 \frac{\partial C}{\partial x} + ikM^2 C = 0 \tag{89a}
\]

\[
\frac{\partial U}{\partial x} + \frac{\partial C}{\partial x} + ikU = 0 \tag{89b}
\]

\[
\frac{\partial U}{\partial y} - \sqrt{M^2 - 1} \frac{\partial V}{\partial x} = 0 \tag{89c}
\]

By transforming Eqs. (89) to the arbitrary coordinate system, $\xi$ and $\eta$, as before, the system can be considered one of simultaneous equations of $U_{\xi}$, $V_{\xi}$, and $C_{\xi}$. By making any of these
derivatives indeterminate the results of setting the determinant of the matrix of their coefficients equal to zero are Eqs. (78), the characteristic directions.

By writing the equations of motion, as before, along lines of $\xi = \text{constant}$ and $\eta = \text{constant}$ and performing the same algebraic manipulations, the two compatibility relations can be determined:

\[
\left( \frac{\partial U}{\partial x} \right)_\xi - \left( \frac{\partial V}{\partial x} \right)_\xi + \frac{ik}{M^2 - 1} (U - C) = 0 \tag{90a}
\]

\[
\left( \frac{\partial U}{\partial x} \right)_\eta + \left( \frac{\partial V}{\partial x} \right)_\eta + \frac{ik}{M^2 - 1} (U - C) = 0 \tag{90b}
\]

The third compatibility relation is, as before,

\[
\left( \frac{\partial U}{\partial x} \right)_{str} + \left( \frac{\partial C}{\partial x} \right)_{str} + ikU = 0 \tag{86}
\]

The equations of motion are now reduced to a system containing only derivatives with respect to $x$. It is, thus, possible to solve these equations using finite differences in the following manner: Separating the real from the imaginary components, the equations can be written,

\[
\left( \frac{\partial U_R}{\partial x} \right)_{str} + \left( \frac{\partial C_R}{\partial x} \right)_{str} - kU_I = 0 \tag{91a}
\]

\[
\left( \frac{\partial U_I}{\partial x} \right)_{str} + \left( \frac{\partial C_I}{\partial x} \right)_{str} + kU_R = 0 \tag{91b}
\]

\[
\left( \frac{\partial U_R}{\partial x} \right)_\xi - \left( \frac{\partial V_R}{\partial x} \right)_\xi - k \frac{M^2}{M^2 - 1} (U_I - C_I) = 0 \tag{91c}
\]

\[
\left( \frac{\partial U_I}{\partial x} \right)_\xi - \left( \frac{\partial V_I}{\partial x} \right)_\xi + k \frac{M^2}{M^2 - 1} (U_R - C_R) = 0 \tag{91d}
\]

\[
\left( \frac{\partial U_R}{\partial x} \right)_\eta + \left( \frac{\partial V_R}{\partial x} \right)_\eta - k \frac{M^2}{M^2 - 1} (U_I - C_I) = 0 \tag{91e}
\]
\[
\left( \frac{\partial U}{\partial x} \right)_n + \left( \frac{\partial V}{\partial x} \right)_n + k \frac{M^2}{M^2 - 1} (U_R - C_R) = 0
\]  

(91f)

To write these equations in finite difference form, a computational molecule as shown in Figure 2 will be used. With the velocity values at \( P_{11}, P_{12}, \) and \( P_{21} \) known, all the values at \( P_{22} \) can be calculated. The distance \( \Delta x \) is arbitrary.

![Figure 2](image)

If \( F \) is an arbitrary flow quantity, its partial derivatives can thus be written,

\[
\left( \frac{\partial F}{\partial x} \right)_{\text{str}} = \frac{F_{22} - F_{11}}{\Delta x} \]  

(92a)

\[
\left( \frac{\partial F}{\partial x} \right)_\xi = \frac{F_{22} - F_{12}}{\frac{1}{2} \Delta x} \]  

(92b)

\[
\left( \frac{\partial F}{\partial x} \right)_n = \frac{F_{22} - F_{21}}{\frac{1}{2} \Delta x} \]  

(92c)

The values of the flow quantities in the equations are averages, such that,

\[
(F)_{\text{str}} = \frac{1}{2} (F_{11} + F_{22}) \]  

(93a)
(F) \( \xi = \frac{1}{2} (F_{12} + F_{22}) \) \hspace{1cm} (93b)

(F) \( \eta = \frac{1}{2} (F_{21} + F_{22}) \) \hspace{1cm} (93c)

With these relations substituted into the equations, the system becomes, after some algebraic manipulation,

\[U_{22R} + C_{22R} - A_I U_{22I} = K_{12R}\] \hspace{1cm} (94a)

\[U_{22I} + C_{22I} + A_I U_{22R} = K_{12I}\] \hspace{1cm} (94b)

\[U_{22R} - V_{22R} - B_I U_{22I} + B_I C_{22I} = K_{34R}\] \hspace{1cm} (94c)

\[U_{22I} - V_{22I} + B_I U_{22R} - B_I C_{22R} = K_{34I}\] \hspace{1cm} (94d)

\[U_{22R} + V_{22R} - B_I U_{22I} + B_I C_{22I} = K_{56R}\] \hspace{1cm} (94e)

\[U_{22I} + V_{22I} + B_I U_{22R} - B_I C_{22R} = K_{56I}\] \hspace{1cm} (94f)

where:

\[K_{12R} = U_{11R} + C_{11R} + A_I U_{11I}\] \hspace{1cm} (95a)

\[K_{12I} = U_{11I} + C_{11I} - A_I U_{11R}\] \hspace{1cm} (95b)

\[K_{34R} = U_{12R} - V_{12R} + B_I (U_{12I} - C_{12I})\] \hspace{1cm} (95c)

\[K_{34I} = U_{12I} - V_{12I} - B_I (U_{12R} - C_{12R})\] \hspace{1cm} (95d)

\[K_{56R} = U_{21R} + V_{21R} + B_I (U_{21I} - C_{21I})\] \hspace{1cm} (95e)

\[K_{56I} = U_{21I} + V_{21I} - B_I (U_{21R} - C_{21R})\] \hspace{1cm} (95f)

and,

\[A_I = \frac{1}{2} k \Delta x\] \hspace{1cm} (96a)

\[B_I = \frac{1}{4} k \frac{M^2}{M^2 - 1} \Delta x\] \hspace{1cm} (96b)

As shown in Teipel (1962), solving these equations gives,
These are the finite difference equations for a point in the general flow field.

At a point on the top of a cascade blade or solid boundary, the normal velocity $V_{22}$ is prescribed by the movement of the blade. This velocity was given in the Problem Formulation. Here the computational molecule is shown in Figure 3.
The applicable equations are,
\[ U_{22R} + C_{22R} - A_I U_{22I} = K_{12R} \] (98a)
\[ U_{22I} + C_{22I} + A_I U_{22R} = K_{12I} \] (98b)
\[ U_{22R} - B_I U_{22I} + B_I C_{22I} = K_{56R} - K_{34R} \] (98c)
\[ U_{22I} + B_I U_{22R} - B_I C_{22R} = K_{56I} - K_{34I} \] (98d)

where \( K_{12} \) and \( K_{56} \) are as before, but,
\[ K_{34R} = V_{22R} \] (99a)
\[ K_{34I} = V_{22I} \] (99b)

as given by the boundary conditions. Solving for these equations gives,
\[ U_{22R} = \frac{(1-A_I B_I)[K_{56R} - K_{34R} - B_I K_{12I}] + 2B_I[K_{56I} - K_{34I} + B_I K_{12R}]}{(1-A_I B_I)^2 + (2B_I)^2} \] (100a)
\[ U_{22I} = \frac{(1-A_I B_I)[K_{56I} - K_{34I} + B_I K_{12R}] - 2B_I[K_{56R} - K_{34R} - B_I K_{12I}]}{(1-A_I B_I)^2 + (2B_I)^2} \] (100b)
\[ C_{22R} = K_{12R} - U_{22R} + A_I U_{22I} \] (100c)
\[ C_{22I} = K_{12I} - U_{22I} - A_I U_{22R} \] (100d)

\( U_{22R} \) and \( U_{22I} \) are known from the boundary conditions. These are the finite difference equations for the flow velocities at the top of a solid boundary or cascade blade.

Likewise, at the bottom of a solid boundary or cascade blade, the normal velocity, \( V_{22} \) is prescribed by the movement of the blade, and was given in the Problem Formulation. The
computational molecule, different from that at the top of a blade, is shown in Figure 4.

\[ \text{Figure 4} \]

Here, the applicable equations are,

\[ U_{22R} + C_{22R} - A_{1}U_{22I} = K_{12R} \]  
\[ U_{22I} + C_{22I} - A_{1}U_{22R} = K_{12I} \]  
\[ U_{22R} - B_{1}U_{22I} + B_{1}C_{22I} = K_{56R} + K_{34R} \]  
\[ U_{22I} + B_{1}U_{22R} - B_{1}C_{22R} = K_{56I} + K_{34I} \]

where \( K_{12} \) and \( K_{34} \) are as originally stated, but,

\[ K_{56R} = V_{22R} \]  
\[ K_{56I} = V_{22I} \]

as given by the boundary conditions. Thus, the finite difference equations for the flow velocities at the bottom of a solid boundary or cascade blade are,

\[ U_{22R} = \frac{(1-A_{1}B_{1})[K_{56R}+K_{34R}-B_{1}K_{12I}]+2B_{1}[K_{56I}+K_{34I}+B_{1}K_{12R}]}{(1-A_{1}B_{1})^2 + (2B_{1})^2} \]  
\[ (103a) \]
\[ u_{22I} = \frac{(1-A_I B_I)[K_{561} + K_{341} + B_I K_{12R}] - 2B_I [K_{56R} + K_{34R} - B_I K_{12I}]}{(1-A_I B_I)^2 + (2B_I)^2} \]  
\( (103b) \)

\[ c_{22R} = K_{12R} - u_{22R} + A_I u_{22I} \]  
\( (103c) \)

\[ c_{22I} = K_{12I} - u_{22I} - A_I u_{22R} \]  
\( (103d) \)

\( v_{22R} \) and \( v_{22I} \) are given from the boundary condition.

To obtain the conditions on the initial left-running Mach line, assume that \( P_{11} \) and \( P_{21} \) as shown in Figure 2 are just in the freestream and then let \( \Delta x \) shrink to zero. Since \( K_{12} = K_{56} = 0 \) and \( A_I \) and \( B_I \) go to zero as \( \Delta x \) goes to zero, the initial finite difference equations for \( \xi = \text{constant} \) are,

\[ u_{22} + c_{22} = 0 \]  
\( (104a) \)

\[ u_{22} - v_{22} + v_{12} = 0 \]  
\( (104b) \)

\[ u_{22} + v_{22} = 0 \]  
\( (104c) \)

Thus,

\[ u_{22} = -v_{22} = -c_{22} \]  
\( (105) \)

Eq. (90a) can then be written,

\[ \left( \frac{\partial u}{\partial x} \right)_\xi = -ikU \]  
\( (106) \)

Integrating gives,

\[ U = U \text{exp}\left[-ik \frac{M^2}{M^2-1} x\right] \text{ at } x=0 \]  
\( (107) \)

Thus,

\[ u_{22R} = -v_{22R}(0) \cos \left(k \frac{M^2}{M^2-1} x\right) - v_{22I}(0) \sin \left(k \frac{M^2}{M^2-1} x\right) \]  
\( (108a) \)

\[ u_{22I} = -v_{22I}(0) \cos \left(k \frac{M^2}{M^2-1} x\right) + v_{22R}(0) \sin \left(k \frac{M^2}{M^2-1} x\right) \]  
\( (108b) \)
\[ V_{22R} = - U_{22R} \quad (108c) \]
\[ V_{22I} = - U_{22I} \quad (108d) \]
\[ C_{22R} = - U_{22R} \quad (108e) \]
\[ C_{22I} = - U_{22I} \quad (108f) \]

For the initial right-running Mach line, \( P_{11} \) and \( P_{12} \) of Figure 2 are just in the freestream and \( \Delta x \) is brought to zero.

Since \( K_{12} = K_{34} = 0 \), and \( A_I = B_I = 0 \),
\[ U_{22} + C_{22} = 0 \quad (109a) \]
\[ U_{22} - V_{22} = 0 \quad (109b) \]
\[ U_{22} - U_{21} + V_{22} = 0 \quad (109c) \]

Thus,
\[ U_{22} = V_{22} = - C_{22} \quad (110) \]

Eq. (90b) can then be written,
\[ (\frac{\partial U}{\partial x})_\eta = - ikU \quad (111) \]

Integrating gives,
\[ U = V(0) e^{-ik \frac{M^2}{M^2 - 1} x} \quad (112) \]

Thus,
\[ U_{22R} = V_{22R}(0) \cos(k \frac{M^2}{M^2 - 1} x) + V_{22I}(0) \sin(k \frac{M^2}{M^2 - 1} x) \quad (113a) \]
\[ U_{22I} = V_{22I}(0) \cos(k \frac{M^2}{M^2 - 1} x) - V_{22R}(0) \sin(k \frac{M^2}{M^2 - 1} x) \quad (113b) \]
\[ V_{22R} = U_{22R} \quad (113c) \]
\[ C_{22R} = - U_{22R} \quad (113d) \]
The pressure distribution along the surfaces of the blades can be determined from Eq. (42). Dividing both sides by $p_\infty$ and using Teipel's amplitude function for $C$, the equation can be written,

$$P(x,y) = \gamma M^2 C(x,y) \bigg|_{x=0^+}^{x=0^-}$$

where

$$P(x,y)e^{i\omega t} = \frac{P - p_\infty}{p_\infty}$$

Thus the non-dimensional pressure distribution along the top surface and the bottom surface of the cascade blade can be computed in a point by point calculation of $U$, $V$, and $C$ along lines of $\xi = \text{constant}$ and $\eta = \text{constant}$. 

\[ V_{22} \varepsilon = U_{22} \varepsilon \quad (113e) \]

\[ C_{22} \varepsilon = -U_{22} \varepsilon \quad (113f) \]
IV. ELEMENTARY THEORY FOR SLOWLY OSCILLATING CASCADES

Sauer's (1950) solution for an airfoil oscillating at low frequencies in an unbounded supersonic flow can be applied to the flat plate cascade to form a theory suitable for comparison with the method of characteristics. As shown in Garrick (1957), the equations of motion of supersonic flow over a flat plate oscillating at small amplitude can be written, assuming irrotational flow, as

$$\cot^2 \alpha \frac{\partial^2 \phi}{\partial x^2} - \frac{\partial^2 \phi}{\partial y^2} + 2 \frac{1}{c^2} \frac{\partial^2 \phi}{\partial t^2} = 0$$  \hspace{1cm} (115)

With the assumption of simple harmonic motion,

$$\phi(x,y,t) = \phi(x,y,k)e^{ikt}$$  \hspace{1cm} (116)

and the potential equation becomes,

$$\cot^2 \alpha \frac{\partial^2 \phi}{\partial x^2} - \frac{\partial^2 \phi}{\partial y^2} + 2ikM^2 \frac{\partial \phi}{\partial x} - k^2M^2 \phi = 0$$  \hspace{1cm} (117)

where $\phi$ is complex.

If the frequency of oscillation is sufficiently low, the perturbation potential amplitude may be expanded in a Taylor series, such that

$$\phi(x,y,k) = \chi(x,y) + k \psi (x,y)$$  \hspace{1cm} (118)

neglecting higher order terms. Eq. (117) then splits into two simultaneous equations since $k$ cannot appear in any relation, it being independent of $k$. These equations are:

$$\cot^2 \alpha \chi_{xx} - \chi_{yy} = 0$$  \hspace{1cm} (119a)

$$\cot^2 \alpha \psi_{xx} - \psi_{yy} = -2iM^2 \chi_x$$  \hspace{1cm} (119b)
Sauer (1950) showed that a general solution to these two equations is,

\[ \chi = g(Z) \tag{120} \]
\[ \psi = h(Z) - i \frac{M}{\cos \alpha} yg(Z) \tag{121} \]
\[ Z = x - y \cot \alpha \tag{122} \]
\[ \psi = h(Z) + i \frac{M}{\cos \alpha} yg(Z) \tag{123} \]
\[ Z = x + y \cot \alpha \tag{124} \]

where \( g \) and \( h \) are arbitrary functions of position and are equal to zero for \( z \leq 0 \) (\( x, y, z \) and \( d \) are non-dimensional).

To apply this general solution to the cascade with super-sonic leading-edge locus, the flow field is divided into separate zones as shown in Figure 5. The number of these zones is dependent on the size of \( A \). For \( A > 1 \), there is one zone along each blade; for \( 0.5 < A \leq 1 \), there are two; for \( 0.33 < A \leq 0.5 \), there are three; for \( 0.25 < A \leq 0.33 \), there are four; etc.

If oscillation in pitch only is analysed, the boundary conditions on the blade surfaces can be written, in non-dimensional form, from Eq. (33) and Eq. (35),

\[ \phi_y = -1 - ik(x - x_o) \quad \text{at } y = 0 \tag{125} \]
\[ \phi_y = e^{i\delta} [-1 - ik(x - x_o)] \quad \text{at } y = d \tag{126} \]

where \( \theta_o \) is one.
In Zone I there is no interference from the upper blade, hence, Sauer's solution for the single airfoil applies. The boundary condition, however, must be satisfied at $y = 0$, hence,

$$\chi_y = -\cot \alpha g_o'(Z) = -1 \quad (127a)$$
$$\psi_y = -\cot \alpha h_o'(Z) - i \frac{M}{\cos \alpha} g_o(Z) = -i(x-x_o) \quad (127b)$$

This gives,

$$g_o(Z) = Z \tan \alpha \quad (128a)$$
$$h_o(Z) = -i \tan \alpha Z(x_o + \frac{Z}{2} \tan^2 \alpha) \quad (128b)$$

where,

$$\phi = g_o(Z) + k\{h_o(Z) - i \frac{M}{\cos \alpha} y g_o(Z)\} \quad (129)$$

In terms of Teipel's amplitude functions the flow properties can be related to those obtained in Method of Characteristics by,
\[ U = \phi_x \] (130a)
\[ V = \tan \alpha \phi_y \] (130b)
\[ C = -[\phi_x + ik\phi] \] (130c)

In Zone I,
\[ U_I = \tan \alpha \{1 - ik[x_0 + x \tan^2 \alpha + y \cot \alpha]\} \] (131a)
\[ V_I = \tan \alpha \{-1 + ik[x_0 - x + y(\cot \alpha + \frac{M}{\cos \alpha})]\} \] (131b)
\[ C_I = \tan \alpha \{-1 + ik[x_0 + x(\tan^2 \alpha - 1) + 2y \cot \alpha]\} \] (131c)

In Zone I', the perturbation potential can also be written directly from Sauer's single airfoil solution,
\[ \phi = g_o(Z) + k \{h_o(Z) + i \frac{M}{\cos \alpha} y g_o(Z)\} \] (132)

where,
\[ Z = x - A - B + y \cot \alpha \] (133)

This ensures that \( Z \leq 0 \) upstream of the leading-edge of the upper blade.

The boundary condition at \( y = d \) gives,
\[ \chi_y = \cot \alpha \ g_o'(Z) = -e^{i\delta} \] (134a)
\[ \psi_y = \cot \alpha \ h_o'(Z) + i \frac{M}{\cos \alpha} g_o(Z) + i \frac{Md}{\cos \alpha} \cot \alpha g_o'(Z) \] (134b)

hence,
\[ g_o(Z) = -Z \tan \alpha \ e^{i\delta} \] (135a)
\[ h_o(Z) = i \tan \alpha \ Z e^{i\delta} \left\{x_0 + \frac{Z}{2} \tan^2 \alpha + \frac{Md}{\cos \alpha}\right\} \] (135b)

Then,
\[ U_{I'} = -\tan \alpha \ e^{i\delta} \{1 - ik[x_0 + (x-B)\tan^2 \alpha + A - y \cot \alpha]\} \] (136a)
\[ V_{I'} = -\tan \alpha \ e^{i\delta} \{-1 + ik[x_0 - (x-B) + (d-y)(\cot \alpha + \frac{M}{\cos \alpha})]\} \] (136b)
\[ C_{I'} = -\tan \alpha \ e^{i\delta} \{-1 + ik[x_0 + (x-B)(\tan^2 \alpha - 1) + 2A - 2y \cot \alpha]\} \] (136c)
For $0.5 \leq A \leq 1.0$, Zones II and II' will be present in addition to Zones I and I'. In Zone II the perturbation potential is due to the initial waves from Zones I and I' plus the reflected waves at Zone II, hence,

$$
\phi = g_0(Z) + g_0(Z') + g_1(Z_1) + k\{h_0(Z) + h_0(Z') + h_1(Z_1) - i \frac{M}{\cos \alpha} y[g_0(Z) - g_0(Z') - g_1(Z_1)] \} \quad (137)
$$

where $Z$ and $Z'$ are defined in Eq. (122) and Eq. (133), and,

$$
Z_1 = x - 2A + y \cot \alpha \quad (138)
$$

The boundary condition at $y = d$ gives,

$$
\chi_y = - \cot \alpha g'_0(Z) + \cot \alpha g'_0(Z') + \cot \alpha g'_1(Z_1) = -e^{i\delta} \quad (139a)
$$

$$
\psi_y = - \cot \alpha h'_0(Z) + \cot \alpha h'_0(Z') + \cot \alpha h'_1(Z_1)
$$

$$
- i \frac{M}{\cos \alpha} [g_0(Z) - g_0(Z') - g_1(Z_1)]
$$

$$
+ i \frac{Md}{\cos \alpha} \cot \alpha [g'_0(Z) + g'_0(Z') + g'_1(Z_1)]
$$

$$
= - i(x - x_0 - B) e^{i\delta} \quad (139b)
$$

hence, after some algebraic manipulation

$$
g_1(Z_1) = Z_1 \tan \alpha \quad (140a)
$$

$$
h_1(Z_1) = -i \tan \alpha Z_1 \{x_0 + \frac{Z_1}{2} \tan^2 \alpha + 2 \frac{Md}{\cos \alpha} \} \quad (140b)
$$

Then,

$$
U_{II} = \tan \alpha\{2 - e^{i\delta} - ik[(2-e^{i\delta})(x_0 + x \tan^2 \alpha + A) + e^{i\delta}(y \cot \alpha + B \tan^2 \alpha)]\} \quad (141a)
$$

$$
V_{II} = \tan \alpha\{-e^{i\delta} + ik[e^{i\delta}(x_0 - x + (d-y)(\cot \alpha + \frac{M}{\cos \alpha}))
$$

$$
-2(d-y)(\cot \alpha + \frac{M}{\cos \alpha})\} \quad (141b)
$$

$$
C_{II} = -\tan \alpha\{2-e^{i\delta} + ik[e^{i\delta}(x_0 + (x-B)(\tan^2 \alpha - 1) + 2A) - 2A - 2y \cot \alpha - 2(x_0 + x(\tan^2 \alpha - 1) + 2A)\]} \quad (141c)
$$
In Zone II' the perturbation potential is due to the initial waves from Zones I and I' plus the reflected waves at Zones II and II', hence,

$$\phi = g_o(Z) + g_o(\bar{Z}) + g_1(\bar{Z}_1) + k\{h_o(Z) + h_o(\bar{Z}) + h_1(\bar{Z}_1) - \frac{M}{\cos \alpha} y[g_o(Z) - g_o(\bar{Z}) + g_1(\bar{Z}_1)]\} \quad (142)$$

where $Z$ and $\bar{Z}$ are defined in Eq. (122) and Eq. (133), and,

$$\bar{Z}_1 = (x - A - B) - y \cot \alpha \quad (143)$$

The boundary condition at $y = 0$ gives,

$$\chi_y = - \cot \alpha g'_o(Z) + \cot \alpha g'_o(\bar{Z}) - \cot \alpha g'_1(\bar{Z}_1) = -1 \quad (144a)$$

$$\psi_y = - \cot \alpha h'_o(Z) + \cot \alpha h'_o(\bar{Z}) - \cot \alpha h'_1(\bar{Z}_1) - i \frac{M}{\cos \alpha} [g_o(Z) - g_o(\bar{Z}) + g_1(\bar{Z}_1)] = i(x - x_o) \quad (144b)$$

hence,

$$g_1(\bar{Z}_1) = - \bar{Z}_1 \tan \alpha e^{i\delta} \quad (145a)$$

$$h_1(\bar{Z}_1) = i \tan \alpha \bar{Z}_1 e^{i\delta} \left\{ x_o + \frac{\bar{Z}_1}{2} \tan^2 \alpha + \frac{M}{\cos \alpha} \right\} \quad (145b)$$

Then,

$$U_{II'} = \tan \alpha \{ 1 - 2e^{i\delta} + ik[2e^{i\delta}(x_o + (x-B)\tan^2 \alpha + A) - (x_o + x \tan^2 \alpha + y \cot \alpha)] \} \quad (146a)$$

$$V_{II'} = - \tan \alpha \{ -e^{-i\delta} + ik[2e^{i\delta}(y(\cot \alpha + \frac{M}{\cos \alpha}) - (x_o - x + y(\cot \alpha + \frac{M}{\cos \alpha}))] \} \quad (146b)$$

$$C_{II'} = - \tan \alpha \{ 1 - 2e^{i\delta} + ik[2e^{i\delta}(x_o + (x-B)(\tan^2 \alpha - 1) + 2A) - (x_o + x(\tan^2 \alpha - 1) + 2y \cot \alpha)] \} \quad (146c)$$
Because of the algebraic complexity of further analysis, additional zone calculations will be made with the assumption that the blades are oscillating in phase, i.e., \( \delta = 0 \).

For \( 0.33 \leq A \leq 0.5 \) in addition to Zones I, I', II, and II', Zones III and III' will be present in the flow field. In Zone III the perturbation potential is due to the initial waves from Zones I and I' plus the reflected waves at Zones II, II', and III,

\begin{align*}
\phi &= g_0(Z) + g_0(\bar{Z}) + g_1(Z_1) + g_1(\bar{Z}_1) + g_2(Z_2) + \\
&\quad k\{ h_0(Z) + h_0(\bar{Z}) + h_1(Z_1) + h_1(\bar{Z}_1) + h_2(Z_2) \\
&\quad -i \frac{My}{\cos \alpha}[g_0(Z) - g_0(\bar{Z}) - g_1(Z_1) + g_1(\bar{Z}_1) + g_2(Z_2)] \} \quad (147)
\end{align*}

where \( Z, \bar{Z}, Z_1, \bar{Z}_1 \) are defined as before, and,

\[ Z_2 = x - 2A - y \cot \alpha \quad (148) \]

The boundary condition at \( y = 0 \) gives, after considerable algebraic manipulation,

\begin{align*}
g_2 &= Z_2 \tan \alpha \quad (149a) \\
h_2 &= -\tan \alpha Z_2 \{ x_0 + \frac{Z_2^2}{2} \tan^2 \alpha + 2 \frac{M\ddot{y}}{\cos \alpha} \} \quad (149b)
\end{align*}

Then,

\begin{align*}
U_{III} &= \tan \alpha \{ 1 - ik[x_0 + (x-2B)\tan^2 \alpha + 2A + y \cot \alpha] \} \quad (150a) \\
V_{III} &= -\tan \alpha \{ 1 - ik[x_0 - x + y(\cot \alpha + \frac{M}{\cos \alpha})] \} \quad (150b) \\
C_{III} &= -\tan \alpha \{ 1 - ik[x_0 + (x-2B)(\tan^2 \alpha - 1) + 4A + 2y \cot \alpha] \} \quad (150c)
\end{align*}

In Zone III' the perturbation potential is due to initial waves from Zones I and I' plus the reflected waves at Zones II, II', and III'.
\[ \phi = g_o(Z) + g_o(\bar{Z}) + g_1(Z_1) + g_1(\bar{Z}_1) + g_2(Z_2) + g_2(\bar{Z}_2) \]
\[ + k\{h_o(Z) + h_o(\bar{Z}) + h_1(Z_1) + h_1(\bar{Z}_1) + h_2(Z_2) + h_2(\bar{Z}_2) \} \]
\[ - i \frac{My}{\cos \alpha} [g_o(Z) - g_o(\bar{Z}) - g_1(Z_1) + g_1(\bar{Z}_1) - g_2(Z_2) - g_2(\bar{Z}_2)] \]  
(151)

where \( Z, \bar{Z}, Z_1, \) and \( \bar{Z}_1 \) are defined as before, and

\[ \bar{Z}_2 = x - B - 3A + y \cot \alpha \]  
(152)

The boundary condition at \( y = d \) gives, after considerable algebraic manipulation,

\[ g_2(Z_2) = - \bar{Z}_2 \tan \alpha \]  
(153a)

\[ h_2(Z_2) = i \tan \alpha \bar{Z}_2 \{ x_o + \frac{Z_2^2}{2} \tan^2 \alpha + 3 \frac{Md}{\cos \alpha} \} \]  
(153b)

Then,

\[ U_{III'} = - \tan \alpha \{ 1 - ik[x_o + (x-3B) \tan^2 \alpha + 3A - y \cot \alpha ] \} \]  
(154a)

\[ V_{III'} = - \tan \alpha \{ 1 - ik[x_o - (x-B) + (d-y)(\cot \alpha + \frac{M}{\cos \alpha})] \} \]  
(154b)

\[ C_{III'} = \tan \alpha \{ 1 - ik[x_o + (x-3B)(\tan^2 \alpha - 1) + 6A - 2y \cot \alpha] \} \]  
(154c)

For \( 0.25 \leq A \leq 0.33 \), in addition to Zones I, I', II, II', III, and III', Zones IV and IV' will be present in the flow field. In Zone IV the perturbation potential is due to the initial waves from Zones I and I' plus the reflected waves at Zones II, II', III, III',

\[ \phi = g_o(Z) + g_o(\bar{Z}) + g_1(Z_1) + g_1(\bar{Z}_1) + g_2(Z_2) + g_2(\bar{Z}_2) + g_3(Z_3) \]
\[ + k\{h_o(Z) + h_o(\bar{Z}) + h_1(Z_1) + h_1(\bar{Z}_1) + h_2(Z_2) + h_2(\bar{Z}_2) + h_3(Z_3) \} \]
\[ - i \frac{My}{\cos \alpha} [g_o(Z) - g_o(\bar{Z}) - g_1(Z_1) + g_1(\bar{Z}_1) + g_2(Z_2) - g_2(\bar{Z}_2) - g_3(Z_3)] \]  
(155)

where \( Z \) through \( \bar{Z}_2 \) are defined as before, and

\[ Z_3 = x - 4A + y \cot \alpha \]  
(156)
The boundary condition at \( y = d \) gives after considerable algebraic manipulation,

\[
g_3(Z_3) = Z_3 \tan \alpha \tag{157a}
\]

\[
h_3(Z_3) = -i \tan \alpha Z_3 \left\{ x_o + \frac{Z_3}{2} \tan^2 \alpha + 4 \frac{M d}{\cos \alpha} \right\} \tag{157b}
\]

Then,

\[
U_{IV} = \tan \alpha \{ 1 - ik [ x_o + (x+3B) \tan^2 \alpha + 3A + y \cot \alpha ] \} \tag{158a}
\]

\[
V_{IV} = -\tan \alpha \{ 1 - ik [ x_o - (x-B) + (y-d) (\cot \alpha + \frac{M}{\cos \alpha}) ] \} \tag{158b}
\]

\[
C_{IV} = -\tan \alpha \{ 1 - ik [ x_o + (x+3B) (\tan^2 \alpha - 1) + 6A + 2y \cot \alpha ] \} \tag{158c}
\]

In Zone IV' the perturbation potential is due to the initial waves from Zones I and I' plus the reflected waves at Zones II, II', III, III', and IV':

\[
\phi = g_o(Z) + g_o(\bar{Z}) + g_1(Z_1) + g_1(\bar{Z}_1) + g_2(Z_2) + g_2(\bar{Z}_2) + g_3(Z_3) + g_3(\bar{Z}_3) + k \{ h_o(Z) + h_o(\bar{Z}) + h_1(Z_1) + h_1(\bar{Z}_1) + h_2(Z_2) + h_2(\bar{Z}_2) + h_3(Z_3) + h_3(\bar{Z}_3) \}
\]

\[
- \frac{M y}{\cos \alpha} [ g_o(Z) - g_o(\bar{Z}) - g_1(Z_1) + g_1(\bar{Z}_1) + g_2(Z_2) - g_2(\bar{Z}_2) + g_3(Z_3) - g_3(\bar{Z}_3) ] \tag{159}
\]

where \( Z \) through \( \bar{Z}_2 \) are defined as before, and,

\[
\bar{Z}_3 = x - B - 3A - y \cot \alpha \tag{160}
\]

The boundary condition at \( y = 0 \) gives,

\[
g_3(\bar{Z}_3) = - \bar{Z}_3 \tan \alpha \tag{161a}
\]

\[
h_3(\bar{Z}_3) = i \tan \alpha \bar{Z}_3 \left\{ x_o + \frac{\bar{Z}_3}{2} \tan^2 \alpha + 3 \frac{M d}{\cos \alpha} \right\} \tag{161b}
\]

Then, after considerable algebraic manipulation,

\[
U_{IV'} = -\tan \alpha \{ 1 - ik [ x_o + (x-4B) \tan^2 \alpha + 4A - y \cot \alpha ] \} \tag{162a}
\]

\[
V_{IV'} = -\tan \alpha \{ 1 - ik [ x_o - x - y (\cot \alpha + \frac{M}{\cos \alpha}) ] \} \tag{162b}
\]
Further zones may be calculated for values of $A$ less than 0.25 in the same manner.

In the Method of Characteristics it was shown that the non-dimensional pressure, $P$, is a direct function of $C$. By integrating this pressure over the blade surface, the lift and pitching moment on the blade may be determined. The derivation of the lift and moment equation is given in the Flutter Analysis. In terms of Garrick and Rubinow's (1946) lift and moment coefficients, for the case of in-phase blade oscillation ($\delta = 0$),

$$L = L_3 + i L_4 = -\frac{2}{k^2} \left\{ \int_0^1 C(x,0^+)dx - \int_0^1 B C(x,d^-)dx \right\}$$

$$M = M_3 + i M_4 = -\frac{4}{k^2} \left\{ \int_0^1 (x-x_0) C(x,0^+)dx - \int_0^1 B (x-B-x_0)C(x,d^-)dx \right\}$$

For $0.25 \leq A \leq 0.33$,

$$L_3 = 0$$

$$L_4 = 8 \frac{1}{k} \tan \alpha \{ d^2 + 3A^2 + B^2 (\tan^2 \alpha - 1) \}$$

$$M_3 = -16 \frac{1}{k^2} \tan \alpha \{ A^2 - B^2 \}$$

$$M_4 = -16 \frac{1}{k} \tan \alpha \{ x_0 (d^2 + 2A^2 + B^2 \tan^2 \alpha) - 8A^3 - 4d^2 A + 4AB^2 \}$$

If the cascade blades are allowed to oscillate at an arbitrary interblade phase angle, the pitching moment determination can be somewhat simplified by integrating in the following manner: For $0.5 \leq A \leq 1.0$, 

$$C_{IV} = \tan \alpha \{ 1 - ik[x_0 - (x-4B) (\tan^2 \alpha - 1) + 8A - 2y \cot \alpha] \}$$
\[ M = M_3 + i M_4 = -\frac{4}{k^2} \int_0^{A-B} (x-x_o) [C_I(x,0^+)] -
\]
\[ - e^{-i\delta} C_I(x+B,x^-) dx + \int_{A-B}^{A+B} (x-x_o) [C_I(x,0^+)] -
\]
\[ - e^{-i\delta} C_{II}(x+B,d^-) dx + \int_{A+B}^1 (x-x_o) [C_{II}(x,0^+)] -
\]
\[ - e^{-i\delta} C_{II}(x+B,d^-) dx \}
\]
This integration gives
\[ M_3 = 8 \frac{1}{k^2} \tan \alpha \{ \cos \delta [x_o (2-2A)+(A-B)^2-1-2AB] -k B \sin \delta \}
\]
\[ \{2x_o^2 + x_o [2A+2(tan^2\alpha-1)]-4A^2-(tan^2\alpha-1)\}
\]
\[ (A^2+ 1 - \frac{1}{3}B^2)) - x_o + \frac{1}{2} \}
\]
\[ (169a) \]
\[ M_4 = 8 \frac{1}{k^2} \tan \alpha \{ \cos \delta [x_o (2A-2)+ x_o (2A^2 - 4A + 2)
\]
\[ + (A^2- B^2- 1)tan^2\alpha)- \frac{4}{3} A^3+ 2(1-B^2)A - \frac{2}{3}
\]
\[ + \frac{2}{3} (1-A^3)tan^2\alpha] - 2 \frac{1}{k} B \sin \delta [x_o - A]
\]
\[ + x_o^2 - x_o + \frac{1}{3} + (\frac{1}{2}x_o - \frac{1}{3}) \tan^2\alpha \}
\]
\[ (169b) \]
In most airfoil theory, the non-dimensional pitching moment is written as,
\[ C_m = \theta_o [c_{m\theta} + ik c_{m\theta}]e^{ikt}
\]
Relating this to Garrick and Rubinow's (1946) moment coefficients gives (where \( \theta_o \) is one),
\[ c_{m\theta} = - \frac{1}{2} k^2 M_3 \]
\[ (171a) \]
\[ c_{m\theta} = - \frac{1}{2} k M_4 \]
\[ (171b) \]
Lane (1956) showed that the flutter analysis of a cascade can be made by considering only a single blade of the cascade oscillating in the flow field. In order to compute the non-dimensional lift and moment acting on that blade, the pressure difference over the top and bottom surfaces must be integrated over the blade. Thus,

\[ L = \int_0^1 (P(x,0^+) - P(x,0^-)) \, dx \]  
\[ M = \int_0^1 (x - x_o) \{ P(x,0^+) - P(x,0^-) \} \, dx \]

where \( P \) is the non-dimensional pressure as defined by Eq. (114).

Lift here is positive downward as in Garrick and Rubinow (1946). The non-dimensional pressure on the lower surface of this equivalent blade is that of the lower surface of the blade above adjusted for the interblade phase angle and stagger such that,

\[ P(x,0^-) = e^{-i\delta} P(x+B,d) \]

The lift and moment on the equivalent blade can also be written

\[ L = \int_0^1 P(x,0^+) \, dx - \int_0^{1+B} e^{-i\delta} P(x,d) \, dx \]
\[ M = \int_0^1 (x-x_o) P(x,0^+) \, dx - \int_0^{1+B} (x-B-x_o) e^{-i\delta} P(x,d) \, dx \]

Separating these into their real and imaginary parts and writing the equations in terms of the local sonic velocities gives,
\[
L_R = \gamma M^2 \int_0^1 C_R(x,0^+)^2 dx - \int_B^{1+B} [\cos \delta \cdot C_R(x,d) + \sin \delta \cdot C_I(x,d)] dx
\]

(177a)

\[
L_I = \gamma M^2 \int_0^1 C_I(x,0^+)^2 dx - \int_B^{1+B} [\cos \delta \cdot C_I(x,d) - \sin \delta \cdot C_R(x,d)] dx
\]

(177b)

\[
M_R = \gamma M^2 \int_0^1 (x-x_o)^2 C_R(x,0^+) dx - \int_B^{1+B} (x-B-x_o)
\cdot [\cos \delta \cdot C_R(x,d) + \sin \delta \cdot C_I(x,d)] dx
\]

(178a)

\[
M_I = \gamma M^2 \int_0^1 (x-x_o)^2 C_I(x,0^+) dx - \int_B^{1+B} (x-B-x_o)
\cdot [\cos \delta \cdot C_I(x,d) - \sin \delta \cdot C_R(x,d)] dx
\]

(178b)

where \( C_R \) and \( C_I \) are the real and imaginary components, respectively.

To evaluate these integrals numerically, the trapezoidal rule is used:

\[
\int_a^b f(x) dx = \frac{\Delta x}{2} \left[ f(x_0) + 2f(x_1) + 2f(x_2) + 2f(x_3) + \ldots + 2f(x_{n-2}) + 2f(x_{n-1}) + f(x_n) \right]
\]

(179)

If oscillation only in bending and torsion (pitching and plunging) is considered and Garrick and Rubinow's (1946) method of expressing lift and pitching moment is used,

\[
L = -\frac{1}{2} \rho_\infty \hat{c} u_\infty^2 k^2 e^{i\omega t} \left[ \frac{2h_o}{\hat{c}} (L_1 + i L_2) + \theta_o (L_3 + i L_4) \right]
\]

(180)

\[
M = -\frac{1}{4} \rho_\infty \hat{c}^2 u_\infty^2 k^2 e^{i\omega t} \left[ \frac{2h_o}{\hat{c}} (M_1 + i M_2) + \theta_o (M_3 + i M_4) \right]
\]

(181)

where \( k = \frac{\omega \hat{c}}{u_\infty} \) unlike Garrick-Rubinow's, \( h_o \) is the vertical equilibrium position of the elastic axis, and \( \theta_o \) is the
equilibrium angle of attack of the blade. In dimensional form the lift and pitching moment equations are,

\[ L = \hat{c} p_0 e^{i\omega t} \left\{ [L_R + i L_I]_{\text{Plunge mode}} + [L_R + i L_I]_{\text{Pitch mode}} \right\} \]  (182)

\&

\[ M = \hat{c}^2 p_0 e^{i\omega t} \left\{ [M_R + i M_I]_{\text{Plunge mode}} + [M_R + i M_I]_{\text{Pitch mode}} \right\} \]  (183)

Thus,

\[ L_1 + i L_2 = -\frac{\hat{c}}{M^2 k^2 \gamma h_0} [L_R + i L_I]_{\text{Plunge mode}} \]  (184)

Separating the real from the imaginary parts of the equation,

\[ L_1 = -\frac{\hat{c}}{M^2 k^2 \gamma h_0} L_R_{\text{Plunge mode}} \]  (185)

or,

\[ L_1 = -\frac{\hat{c}}{k^2 h_0} \int_0^1 C_R(x, 0^+) dx - \int_0^{1+B} \left[ \cos \delta \cdot C_R(x, d) + \sin \delta \cdot C_I(x, d) \right] dx \] _{\text{Plunge mode}} \]  (186a)

Likewise,

\[ L_2 = -\frac{\hat{c}}{k^2 h_0} \int_0^1 C_I(x, 0^+) dx - \int_0^{1+B} \left[ \cos \delta \cdot C_I(x, d) - \sin \delta \cdot C_R(x, d) \right] dx \] _{\text{Plunge mode}} \]  (186b)

\[ L_3 = -2 \frac{1}{k^2 \theta_0} \int_0^1 C_R(x, 0^+) dx - \int_0^{1+B} \left[ \cos \delta \cdot C_R(x, d) + \sin \delta \cdot C_I(x, d) \right] dx \] _{\text{Pitch mode}} \]  (187a)
L_4 = -2 \frac{1}{k^2 \theta_o} \left\{ \int_0^1 C_I(x,0^+) dx - \int_{1+B} [\cos \delta \cdot C_I(x,d) - \sin \delta \cdot C_R(x,d)] dx \right\} \text{Pitch mode} \tag{187b}

M_1 = -2 \frac{\hat{c}}{k^2 h_o} \left\{ \int_0^1 (x-x_o) C_I(x,0^+) dx - \int_{1+B} (x-B-x_o) \right\} \cdot [\cos \delta \cdot C_R(x,d) + \sin \delta \cdot C_I(x,d)] dx \text{Plunge mode} \tag{188a}

M_2 = -2 \frac{\hat{c}}{k^2 h_o} \left\{ \int_0^1 (x-x_o) C_I(x,0^+) dx - \int_{1+B} (x-B-x_o) \right\} \cdot [\cos \delta \cdot C_R(x,d) - \sin \delta \cdot C_I(x,d)] dx \text{Plunge mode} \tag{188b}

M_3 = -4 \frac{1}{k^2 \theta_o} \left\{ \int_0^1 (x-x_o) C_I(x,0^+) dx - \int_{1+B} (x-B-x_o) \right\} \cdot [\cos \delta \cdot C_R(x,d) + \sin \delta \cdot C_I(x,d)] dx \text{Pitch mode} \tag{189a}

M_4 = -4 \frac{1}{k^2 \theta_o} \left\{ \int_0^1 (x-x_o) C_I(x,0^+) dx - \int_{1+B} (x-B-x_o) \right\} \cdot [\cos \delta \cdot C_I(x,d) - \sin \delta \cdot C_R(x,d)] dx \text{Pitch mode} \tag{189b}

where the integrals are computed using the trapezoidal rule and the two different oscillatory modes are determined by the boundary conditions at the blade surfaces.

Bisplinghoff, Ashley, and Halfman (1955) define flutter as "the dynamic instability of an elastic body in an airstream." In Chapter 9 they describe the physical nature of the flutter phenomenon. Basically, flutter occurs when the characteristic determinant of the equations of motion of the blade vanishes.
If blade oscillation in two degrees of freedom (pitch and plunge) is considered, the equations of motion of the cascade blade can be written as,

\[ M_{h} + S_{\theta} \ddot{\theta} + C_{h}\dot{h} = L \]  
\[ S_{\theta} h + I_{\theta} \ddot{\theta} + C_{\theta}\theta = M_{\theta} \]

where the total inertia force acting on the blade is,

\[ F_{I} = -(M_{h} + S_{\theta} \ddot{\theta}) \]

the elastic restoring force is,

\[ F_{R} = -C_{h}h \]

the sum of the moments of the inertia forces about the elastic axis is,

\[ M_{I} = -(S_{\theta} \ddot{h} + I_{\theta} \ddot{\theta}) \]

and the elastic restoring moment is,

\[ M_{R} = -C_{\theta}\theta \]

\( L \) and \( M_{\theta} \) are the aerodynamic lift and pitching moment about the elastic axis. With the assumption of simple harmonic oscillation, the blade motion can be written,

\[ h = h_{o}e^{i\omega t} \]  
\[ \theta = \theta_{o}e^{i\omega t} \]

The equations of motion thus become,

\[ L = e^{i\omega t}[-\omega^{2}M_{h} - \omega^{2}S_{\theta}\theta_{o} + C_{h}h_{o}] \]  
\[ M_{\theta} = e^{i\omega t}[-\omega^{2}S_{\theta}h_{o} - \omega^{2}I_{\theta}\theta_{o} + C_{\theta}\theta_{o}] \]
Equating these to Garrick-Rubinow's standard lift and moment expressions and following Garrick-Rubinow's (1946) method, the equations can be written as,

\[
\left(\Omega_h X - \mu + L_1 + i L_2\right) \frac{2h^2}{c} + (-\mu X_\theta + L_3 + i L_4) \theta_0 = 0 \quad (200a)
\]

\[
(-\mu X_\theta + M_1 + i M_2) \frac{2h^2}{c} + (\Omega_\theta M - \mu r^2 + M_3 + i M_4) \theta_0 = 0 \quad (200b)
\]

The two simultaneous flutter equations form an eigenvalue problem in \(\frac{2h^2}{c}\) and \(\theta_0\), thus, the determinant of coefficients (the characteristic determinant) must be zero:

\[
\begin{vmatrix}
\Omega_h X - \mu + L_1 + i L_2 & -\mu X_\theta + L_3 + i L_4 \\
-\mu X_\theta + M_1 + i M_2 & \Omega_\theta M - \mu r^2 + M_3 + i M_4 \\
\end{vmatrix} = 0 \quad (201)
\]

This is the flutter determinant with unknown \(X\). If the determinant is expanded, and the real and imaginary parts equated to zero the result is two equations that given a value of \(k\) and \(M\) will yield values of \(X\). For a specific Mach number, the flutter point occurs at the value of \(k\) that produces identical values of \(X\) in the two equations. From this value of \(k\) and \(X\) the flutter speed and frequency can be calculated.

The two flutter equations are:

(Real part)

\[
\Omega_h \Omega_\theta X^2 + [\Omega_\theta (L_1 - \mu) + \Omega_h (M_3 - \mu r^2)]X + C_R = 0 \quad (202)
\]

(Imaginary part)

\[
(\Omega_\theta L_2 + \Omega_h M_4)X + C_I = 0 \quad (203)
\]

where,
\[ C_R = \mu \left[ x_\theta (M_1 + L_3) - (M_3 - \mu r_\theta^2) - L_1 r_\theta^2 - \mu x_\theta^2 \right] + D_R \] (204a)

\[ C_I = \mu \left[ x_\theta (M_2 + L_4) - M_4 - L_2 x_\theta^2 \right] + D_I \] (204b)

and where,

\[ D_R = L_1 M_3 - L_3 M_1 - L_2 M_4 + L_4 M_2 \] (205a)

\[ D_I = L_1 M_4 - L_4 M_1 + L_2 M_3 - L_3 M_2 \] (205b)

Solving for \( X \),

(Real Part)

\[ X = \frac{-[\Omega_\theta (L_1 - \mu) + \Omega_h (M_3 - \mu r_\theta^2)] \pm \sqrt{[\Omega_\theta (L_1 - \mu) + \Omega_h (M_3 - \mu r_\theta^2)]^2 - 4 C_R \Omega_h \Omega_\theta}}{2 \Omega_h \Omega_\theta} \] (206)

(Imaginary Part)

\[ X = \frac{-C_I}{\Omega_\theta L_2 + \Omega_h M_4} \] (207)

The solution to the flutter problem is then to obtain values of \( X \) as a function of \( k \) until the real and imaginary solutions converge. At this "flutter point' the non-dimensional flutter frequency is,

\[ \frac{\omega_F}{\omega_\theta} = \frac{1}{\sqrt{X}} \] (208)

and the non-dimensional freestream flutter speed is

\[ \frac{U_F}{\hat{c} \omega_\theta} = \frac{1}{k \sqrt{X}} \] (209)

It should be noted that the interblade phase angle is an open parameter (as well as \( k \)) in calculating lift and moment on the cascade blade. Lane (1956) showed that the minimum flutter speed of a cascade occurs at a non-zero interblade phase angle and that this is the critical phase angle at.
which the cascade will oscillate. For an unstaggered cascade this critical phase angle was shown to be $180^\circ$, but for a staggered cascade it can only be found by repeated flutter calculations at various phase angles. The problem is solved when the minimum flutter speed is determined.

If only a single degree of freedom of motion (pitch oscillation) exists, the flutter problem is greatly simplified. The equation of motion of the blade can then be written,

$$I_\theta \ddot{\theta} + (1+ig)C_\theta \theta = M_\theta$$  \hspace{1cm} (210)

Here the sum of the moments of the inertia forces about the elastic axis is,

$$M_I = -I_\theta \ddot{\theta}$$  \hspace{1cm} (211)

the elastic restoring moment is,

$$M_R = -(1+ig)C_\theta \theta$$  \hspace{1cm} (212)

and the aerodynamic pitching moment about the elastic axis is,

$$M_\theta = -\frac{1}{4} \rho_\infty \hat{c}^2 u_\infty^2 k^2 \theta o \ e^{i\omega t} (M_3 + i \ M_4)$$  \hspace{1cm} (213)

With the assumption of simple harmonic oscillation as before, the blade motion can be written as,

$$\theta = \theta_o e^{i\omega t}$$  \hspace{1cm} (214)

$$\ddot{\theta} = -\omega^2 \theta_o e^{i\omega t}$$  \hspace{1cm} (215)

The equation of motion then becomes,

$$e^{i\omega t} [I_\theta \omega^2 \theta_o + (1+ig)C_\theta \theta o] = -\frac{1}{4} \rho_\infty \hat{c}^2 u_\infty^2 k^2 \theta o \ e^{i\omega t} (M_3 + i \ M_4)$$  \hspace{1cm} (216)

Separating the equation into its real and imaginary parts gives,
\[-I_0 \omega^2 + C_0 + \frac{1}{4} \rho_\infty a^4 \omega^2 M_3 = 0 \quad (217a)\]
\[g C_0 + \frac{1}{4} \rho_\infty a^4 \omega^2 M_4 = 0 \quad (217b)\]

or as in Garrick and Rubinow (1946)
\[\Omega_0 X - \mu r_0^2 + M_3 = 0 \quad (218a)\]
\[M_4 + g \Omega_0 X = 0 \quad (218b)\]

Thus, the solution to the single-degree-of-freedom problem is to obtain values of \(M_4\) as a function of \(k\) until the second equation is satisfied. The value of \(M_3\) for that \(k\) is then used in the first equation to determine if \(X\) is reasonable:
\[X = 1 - \frac{M_3}{\Omega_0} > 0 \quad (219)\]

The non-dimensional flutter frequency is as before,
\[\frac{\omega_F}{\omega_0} = \frac{1}{\sqrt{X}} \quad (220)\]

and the non-dimensional freestream flutter speed is,
\[\frac{U_F}{c \omega_0} = \frac{1}{k \sqrt{X}} \quad (221)\]

Again it must be noted that in the case of staggered cascades the flutter computations must be repeated with varying interblade phase angle until the minimum flutter speed is found. This will give the critical flutter condition.

As in the analysis of the single airfoil, in the absence of structural damping, aerodynamic instability (flutter) occurs when \(M_4 \leq 0\). When \(M_4\) is written as the pitch damping
derivative, Eq. (171b), it should be noted that aerodynamic undamping occurs when the pitch damping derivative is positive.
VI. COMPUTATIONAL PROCEDURE

The computer programs used to calculate the cascade flow field are based on one used by Platzer and Pierce (1970) to calculate pressure distributions on an airfoil oscillating with wind tunnel interference. The programs here, however, in addition to calculating the pressure distribution on the blade of a cascade integrate the pressure distribution over the blade surface and calculate the flutter frequency and freestream flutter velocity. One program (Program A) is for two-degree-of-freedom (pitch and plunge) flutter calculations while the second (Program B), merely a simplification of the first, is for single-degree-of-freedom (pitch) flutter calculations.

Both programs are valid only for cascades with supersonic leading-edge locus vibrating with small amplitude oscillations. The programs calculate the flow field between two adjacent blades using the method of characteristics finite difference equations developed previously and integrate the pressure distributions on the blade surfaces using the trapezoidal rule.

The computational molecules shown in Figures 2, 3, and 4 are used to calculate the flow field quantities at $P_{22}$ in the general flow field, on the lower blade, and on the upper blade, respectively. The distance $\Delta x$ shown in the figures is determined by,
\[ \Delta x = \frac{2A}{v} \]  \hspace{1cm} (222)

where \( v \) is the grid fineness ratio. This parameter is an arbitrary input variable into the programs and is equal to one less than the total number of grid points on any Mach line that runs from one blade to the other.

Input parameters are entered into the programs by means of the NAMELIST option of FORTRAN. In using this option, input parameters are combined in a titled list and then referred to in the programs by this title. In both Program A and B the NAMELIST title is the same (NAM1); however, the input variables in each are slightly different. The real advantage of using NAMELIST is that each data card need only contain the value of one input variable while the format of the data card is such that the variable's name is punched on the card as well as its value. In this way, when a different value for an input variable is desired, rather than changing the entire data deck, only one easily identifiable data card need be changed.

The format for a NAMELIST data deck is as follows: The first column in each card of the deck is left blank. The first card starts in the second column with the symbol & (ampersand) followed immediately by the title of the NAMELIST (here, NAM1). The next cards list the variables and their input values, one per card, in any order. The format here, starting in the second column and followed by a comma, is: Variable Name = Value. The last card in the data deck and
the signal to the computer that the NAMELIST has ended again starts in the second column with the symbol & (ampersand) followed immediately by END.

The value of an input variable may be written in any format as long as integer variables are written without decimal points and real variables are written with a decimal point. Blanks are taken as zeroes, but a comma must appear somewhere between a desired input value and the next data card.

In both programs, prior to the NAMELIST dataset, the date is entered on a data card in the first twelve spaces.

To illustrate the input procedures, the following two examples are proposed:

<table>
<thead>
<tr>
<th>Two Degree of Freedom Flutter</th>
<th>One Degree of Freedom Flutter</th>
</tr>
</thead>
<tbody>
<tr>
<td>( v = 100 )</td>
<td>( v = 100 )</td>
</tr>
<tr>
<td>( M = \sqrt{2} )</td>
<td>( M = \sqrt{2} )</td>
</tr>
<tr>
<td>( \gamma = 1.4 )</td>
<td>( \gamma = 1.4 )</td>
</tr>
<tr>
<td>( k = 2 )</td>
<td>( k = 2 )</td>
</tr>
<tr>
<td>( x_0 = 0.5 )</td>
<td>( x_0 = 0.5 )</td>
</tr>
<tr>
<td>( d = 0.6 )</td>
<td>( d = 0.6 )</td>
</tr>
<tr>
<td>( \beta = 26.565^\circ )</td>
<td>( \beta = 26.565^\circ )</td>
</tr>
<tr>
<td>( \delta = 180^\circ )</td>
<td>( \delta = 180^\circ )</td>
</tr>
<tr>
<td>( \mu = 500 )</td>
<td>( \mu = 500 )</td>
</tr>
<tr>
<td>( r_\theta = 0.5 )</td>
<td>( r_\theta = 0.5 )</td>
</tr>
<tr>
<td>( x_\theta = 0.1 )</td>
<td>( g = 2% )</td>
</tr>
<tr>
<td>( \omega_H )</td>
<td>( \Delta 2/k = 1 )</td>
</tr>
<tr>
<td>( \omega_\theta )</td>
<td>( \hat{c} = 1.2 )</td>
</tr>
<tr>
<td>( \Delta 2/k = 1 )</td>
<td></td>
</tr>
<tr>
<td>( \hat{c} = 1.2 )</td>
<td></td>
</tr>
</tbody>
</table>
The data cards needed are as follows:

<table>
<thead>
<tr>
<th>Card</th>
<th>Program</th>
<th>Format</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>A &amp; B</td>
<td>16 JUNE 1972</td>
<td>Date</td>
</tr>
<tr>
<td>2</td>
<td>A &amp; B</td>
<td>&amp;NAM1</td>
<td>NAMELIST title</td>
</tr>
<tr>
<td>3</td>
<td>A &amp; B</td>
<td>NGRDFN = 100,</td>
<td>Grid fineness ratio, must be an even integer, less than 400.</td>
</tr>
<tr>
<td>4</td>
<td>A &amp; B</td>
<td>FSTRMN = 1.414214,</td>
<td>Freestream Mach number</td>
</tr>
<tr>
<td>5</td>
<td>A &amp; B</td>
<td>RTOSPH = 1.4,</td>
<td>Ratio of specific heats</td>
</tr>
<tr>
<td>6</td>
<td>A &amp; B</td>
<td>REDFRQ = 2.0,</td>
<td>Reduced frequency</td>
</tr>
<tr>
<td>7</td>
<td>A &amp; B</td>
<td>XSUBO = 0.5,</td>
<td>Elastic axis position</td>
</tr>
<tr>
<td>8</td>
<td>A &amp; B</td>
<td>TNWDST = 0.6,</td>
<td>Blade distance</td>
</tr>
<tr>
<td>9</td>
<td>A &amp; B</td>
<td>STGANG = 26.565,</td>
<td>Stagger angle, ( \tan \beta \leq \cot \alpha ).</td>
</tr>
<tr>
<td>10</td>
<td>A &amp; B</td>
<td>FAZE = 180.0,</td>
<td>Interblade phase angle</td>
</tr>
<tr>
<td>11</td>
<td>A &amp; B</td>
<td>MUU = 500.0,</td>
<td>Blade density parameter</td>
</tr>
<tr>
<td>12</td>
<td>A &amp; B</td>
<td>RSUBA = 0.5,</td>
<td>Radius of gyration</td>
</tr>
<tr>
<td>13</td>
<td>A</td>
<td>XSUBA = 0.1,</td>
<td>Distance from the blade elastic axis to its center of gravity.</td>
</tr>
<tr>
<td>13</td>
<td>B</td>
<td>GSUBA = 0.02,</td>
<td>Structural damping coefficient</td>
</tr>
<tr>
<td>14</td>
<td>A &amp; B</td>
<td>INCRE = 1.0,</td>
<td>2/k increment</td>
</tr>
<tr>
<td>15</td>
<td>A &amp; B</td>
<td>FIN = 1.2,</td>
<td>Chord length, ( \hat{c} \geq 2A ).</td>
</tr>
<tr>
<td>16</td>
<td>A</td>
<td>HAFREQ = 1.5,</td>
<td>Ratio of natural frequencies, bending to torsion.</td>
</tr>
<tr>
<td>16</td>
<td>B</td>
<td>&amp;END</td>
<td>End of NAMELIST</td>
</tr>
<tr>
<td>17</td>
<td>A</td>
<td>&amp;END</td>
<td>End of NAMELIST</td>
</tr>
</tbody>
</table>
The definitions of these parameters are given again in the alphabetical listing of all program variables in Appendix A.

Program A has eight subroutines. They are as follows:

**INPUT**
Reads in all input data and sets up the finite difference grid.

**INTIAL**
Initializes all the flow quantities at the leading-edge, starts the integration, and initializes most of the logic variables.

**MACHLN**
Computes the flow quantities along the initial left and right-running Mach lines at the given grid point.

**HIFOIL**
Computes the flow quantities on the lower surface of the upper airfoil at the given grid point, and continues the integration of the pressure there.

**GENFPT**
Computes the flow quantities at a general flow field grid point.

**LOFOIL**
Computes the flow quantities on the upper surface of the lower airfoil at the given grid point, and continues the integration of the pressure there.

**COMPXY**
Computes the x and y position of the given grid point.

**FLUTER**
Computes the oscillation frequency ratio, X, for both real and imaginary flutter equations.

Program B contains all of these subroutines except FLUTER.

Since the flutter problem is greatly simplified in the single degree of freedom case, the flutter calculations are made in the main program.

Program A contains at least two "DO" loops. The inner one is for calculating the flow field with first pitch oscillation boundary conditions and then plunge oscillation boundary conditions. This loop is not present in Program B.
since only the flow field due to pitch oscillation is desired. The next outer loop is used to increment $2/k$ ($1/k$ in Garrick-Rubinow's (1946) notation) starting at that reduced frequency specified in INPUT. This loop is continued until hopefully the flutter point is found. Any loops outside of these are used to make multiple flutter calculations in one run with varying input parameters.

In INPUT the grid is determined using the grid fineness ratio, blade spacing, and Mach number. The value of $\Delta x$ as shown in Eqs. (96) is obtained. The stagger angle is then adjusted such that the distance the upper blade is staggered back is the nearest multiple number of $\Delta x$ increments. The "Compatible Stagger Angle" is then printed out. The upper blade may be staggered back such that the distance between its leading-edge and its intersection with the initial left-running Mach line from the lower blade is zero, but no further. With less stagger this distance must be greater than or equal to $2\Delta x$, or,

$$A - B \geq 2\Delta x$$

(223)

This allows a minimum of three grid points in the primed flow field zones as shown in Figure 5. This is the minimum number of points required when the value of the last point must be linearly extrapolated from the other two, as is the procedure in the programs.

Flow quantities along the initial left and right-running Mach lines are computed using the equations derived in the Method of Characteristics for initial conditions. Those
quantities along the blade surfaces are computed using the equations derived for the appropriate boundary condition. The values of the normal velocities are derived as a result of the analysis of the flow tangency condition and have been given previously.

Because the programs compute the flow field in rows of right-running Mach lines starting with points on the initial left-running Mach line, provision must be made for the computation of points in the area referred to as Zone I'. The reason is that once the computation reaches the intersection of the initial left-running Mach line from the lower blade and the initial right-running Mach line from the upper blade, the points on the initial left-running Mach line become general flow field points requiring those equations for the flow quantity computation. To use those equations the flow quantities along the last left-running Mach line in Zone I' must be known. They are obtained in two ways depending on the value of the interblade phase angle: If the phase angle is non-zero, the entire flow field in Zone I' (along with the integration over the bottom surface of the top blade) is computed concurrently with the computation of the flow field in Zone I. If the phase angle is zero, only that row in Zone I that is the image of the last left-running Mach line in Zone I' must be determined since the flow quantities of these two rows are the same (except for the signs of U and C).

The pressure distributions on the surfaces of the blades are integrated point by point as they are being calculate
Since the pressure coefficient at the last point in each zone of influence is unknown, its value must be linearly extrapolated from the preceding points. The pressure distribution of the upper blade is adjusted so as to represent the pressure distribution on the bottom surface of the lower blade:

\[ C(x,0^-) = e^{-i\delta} C(x+B,d) \]  

(224)

In this way the result of the integration is the lift and pitching moment on a representative blade of the cascade.

Termination of the flow field calculation starts when the distance from the leading-edge of the lower blade to the point of calculation on the lower blade reaches the prescribed chord length. Because HIFOIL must be called at least once for proper program termination, this chord length has a lower bound:

\[ \delta \geq 2A \]  

(225)

Since the flow field is being calculated along rows of right-running Mach lines, the last zone along the upper blade is then computed to complete the calculation. Although the flow field is terminated when the calculation reaches the prescribed chord length, the integration of the pressure distributions is terminated when the chord length reaches unity.

When the values of total lift and pitching moment have been determined, Program A enters them in the flutter equations, Eq. (206) and Eq. (207) via subroutine FLUTER. Program B, computing only the pitching moment, enters these values in Eqs. (218). Convergence to the flutter point is then determined.
In Program A, the difference between solutions to the real flutter equation and that to the imaginary flutter equation is noted after each computation. If the absolute value of the difference is less than $10^{-6}$, the flutter point is considered to be determined. When the difference changes sign, indicating convergence somewhere in the frequency interval, the $2/k$ increment is divided by five and the computation is started again at a new value of $2/k$ determined by adding the new increment to the previous lower value. Each time the flutter point is passed the increment is made smaller. The flutter point is considered to be ascertained when the increment is less than or equal to $10^{-5}$.

In Program B, the value of $X$ is determined from Eq. (218a). This value is then used along with $M_4$ to determine the value of Eq. (218b). If the absolute value of Eq. (218b) is less than $10^{-6}$, the flutter point is considered to be determined. If Eq. (218b) is negative, indicating convergence somewhere in the frequency interval, the same procedure is followed as in Program A. When the flutter point is determined, the value of $X$ is tested to insure a realistic flutter frequency as shown in Eq. (219).

The output from Program A consists of the input parameters (including such items as $x$ and the compatible stagger angle); the values for $X_R$ and $X_I$ as defined in Eq. (206) and Eq. (207), respectively, for each reduced frequency calculated; and the non-dimensional flutter frequency and flutter speed as defined in Eq. (208) and Eq. (209), respectively. The value of
"1/k" given in the output corresponds to the reciprocal of the reduced frequency as defined by Garrick and Rubinow (1946). This is equivalent to our 2/k.

The output from Program B is identical to that of Program A except that Program B gives the values of \( M_4, M_3, X, \) and the system damping \( (M_4 - g\Omega_0 X) \) as shown in Eq. (218b) for each reduced frequency calculated.

A flow diagram of Program A is given in Appendix B.
VII. RESULTS

A. COMPARISON OF ELEMENTARY THEORY WITH MILES

Miles (1956) presented an analysis of supersonic flow past an oscillating airfoil subjected to wall interference in a solid-wall wind tunnel. Using Laplace transform techniques he derived the following results for the moment coefficient (his Eq. (7.18))

\[ c_{\text{m}a} = 4 \tan \alpha \left( (2N+1) \left[ x_0 - \frac{1}{2} \right] - N(N+1)A x_0 + \frac{1}{6} N(N+1)(2N+1)A^2 \right) \]  
\[ c_{\text{mq}} = -4 \tan \alpha \left( (2N+1) \left[ x_0^2 - x_0 + \frac{1}{3} \right] - N(N+1)A \right) \left[ x_0^2 - x_0 + \frac{1}{2} \right] + \frac{1}{12} N^2 (N+1)^2 A^3 \]  
\[ c_{\text{m} \alpha} = 4 \tan \alpha \left( -(2N+1) \tan^2 \alpha \left[ \frac{1}{2} x_0 - \frac{1}{3} \right] - N(N+1)A \right) \left[ x_0 - \frac{1}{2} \right] + \frac{1}{6} N(N+1)(2N+1)(\tan^2 \alpha + 2) \cdot A^2 x_0 - \frac{1}{12} N^2 (N+1)^2 (2 \tan^2 \alpha + 3)A^3 \]  

(226a)  
(226b)  
(226c)

where \( N \) is the largest integer smaller than \( A^{-1} \) and \( \alpha, \alpha \) subscripts represent angle of attack. Here, for a slowly oscillating airfoil,

\[ C_m = \alpha_0 [c_{\text{m}a} + ik(c_{\text{mq}} + c_{\text{m} \alpha})] e^{ikt} \]  

(227)

Unlike Eq. (170).

As stated by Miles, this case is identical to that of the unstaggered cascade oscillating with an interblade phase angle of \( 180^\circ \), and the values of pitching moment as defined in Eq. (170) and Eq. (171) based on our elementary theory for \( B = 0, \delta = 180, \) and \( 0.5 \leq A \leq 1.0 \), prove to be the same as those shown in Eqs. (226) with \( N = 1 \). The torsional
The large destabilizing influence of interference causing instability over the whole supersonic Mach number range for a wide range of elastic axis positions can be seen. Also shown is the interference-free boundary (A = 1) for the single airfoil in an unbounded supersonic flow (Garrick and Rubinow 1946) exhibiting instability as is well known only for low supersonic Mach numbers. As demonstrated above,
these stability boundaries also apply to the slowly oscillating airfoil subjected to solid-wall wind tunnel interference.

B. COMPARISON OF ELEMENTARY THEORY WITH LANE

Lane (1957) extended Miles' Laplace transform solution to the analysis of oscillating cascades with supersonic leading-edge locus. By expanding the blade pressure distributions (his Eq. (13)) with respect to frequency and retaining only terms up to the first power in frequency, the pressure jump across a slowly oscillating blade can be obtained. This pressure jump is identical to that being integrated in Eq. (168).

C. COMPARISON OF ELEMENTARY THEORY WITH DRAKE

Drake (1956) using Laplace transform techniques presented an analysis of supersonic flow past an oscillating airfoil in a wind tunnel with free jet boundaries. His pitching moment coefficient is defined in the English system, equivalent to Eq. (170), is,

\[ C_m = \alpha_0 \left[ m_\alpha + ik m_\phi \right] e^{ikt} \] (228)

for a slowly oscillating airfoil where \( \alpha \) again refers to angle of attack. Here,

\[ m_\alpha = 4 \tan \alpha \left[ (x_o - 1)h_1(A) + h_2(A) \right] \]

(229a)

\[ m_\phi = 4 \tan \alpha \left[ \frac{\partial}{\partial A} \left( \frac{3}{2} - 1 \right) \right] \left[ \left( x_o - 1 \right) h_2(A) + h_3(A) \right] \]

(229b)

where for \( 0.5 \leq A \leq 1.0 \),
This case is identical to that of the unstaggered cascade oscillating in phase ($\delta = 0$), and the values of pitching moment as defined in Eq. (170) and Eq. (171) based on our elementary theory for $\delta = 0$, $\beta = 0$, and $0.5 \leq A \leq 1.0$ prove to be the same as those shown in Eq. (229).

Drake (1957) extended his work to the case of the airfoil oscillating in a wind tunnel with porous walls. Platzer (1971) will publish an analysis of this problem using our elementary theory. His pitch damping coefficient as defined in Eq. (170), for a slowly oscillating airfoil is,

$$c_m = -4 \tan \alpha \left[ x_o + x_o \left( \frac{1}{2} \tan^2 \alpha + 1 \right) + \frac{1}{3} (1 - \tan^2 \alpha) \right]$$

$$+ \frac{2(1-A)}{1+\sigma \tan \alpha} \left\{ [1-\sigma \tan \alpha] (1-\tan^2 \alpha) + \frac{2\sigma \tan^3 \alpha}{1+\sigma \tan \alpha} \right\}$$

$$X + [(1-\sigma \tan \alpha) (x_o + A(1+\tan^2 \alpha))] [Y]$$

where

$$X = (1-A) \left[ -\frac{1}{2} x_o + \frac{1}{6} (A+2) \right]$$

$$Y = x_o - \frac{1}{2} (A+1)$$

and $0.5 \leq A \leq 1.0$. Here $0 \leq \sigma \leq \infty$ where $\sigma = 0$ represents the solid-wall wind tunnel case and $\sigma = \infty$ represents a free-jet wind tunnel boundary. With these limiting values of porosity substituted in Eq. (231) the comparison with Eq. (171b) is exact for the case of $\delta = 180$ and $\delta = 0$, respectively.
Further the results check with Drake (1957). Stability boundaries for an airfoil oscillating slowly in a porous-wall wind tunnel are given in Figure 7 for a tunnel aspect ratio, $A = 0.5$ and for various values of tunnel porosity. An increase in tunnel porosity is seen to have a large stabilizing influence.

![Diagram](image)

**Figure 7**

**D. AIRFOIL - AIRFOIL INTERFERENCE**

Our elementary solution of the slowly oscillating cascade problem can be used to study other interference problems as well. The interference between two airfoils may be of interest for certain proposed space shuttle configurations, although two-dimensional flow can only give an indication of
trends. Our elementary solution applied to the unstaggered
cascade oscillating with an interblade phase angle of 180°,
as stated before, is equivalent to an airfoil subjected to
solid-wall wind tunnel interference. If the upper wind tun-
nel wall is considered a large stationary airfoil (idealized
as a flat plate at zero angle of attack), the theory can be
used to determine the non-dimensional pressure distribution
along the upper surface of a smaller airfoil mounted closely
below it. Using the non-interference pressure distribution
over the lower surface of the airfoil, the net pressure dis-
tribution over the airfoil can be integrated over the chord
length and the pitch damping coefficient as defined in Eq.
(171b) can be determined as,
\[ c_m = -2 \tan \alpha \left\{ x_o^2(4-2A) - x_o[3A^2 - 4A + 2 \\
+ (A^2 - 2)(\tan^2 \alpha - 1)] + 2A^3 - 2A - \frac{2}{3}(2-A^3) \\
(\tan^2 \alpha - 1) \right\} \\
\text{where} \\
A = 2h \cot \alpha \\
h = \text{airfoil distance} \\
\text{and} \ 0.5 \leq A \leq 1.0.

The stability boundaries for an airfoil undergoing inter-
ference of this type are shown in Figure 8. The effect of
interference is strongly destabilizing. This case further
applies to an airfoil mounted close to only one solid wall
in a wind tunnel.
E. COMPARISON OF THE METHOD OF CHARACTERISTICS WITH ELEMENTARY THEORY

Values of flow field quantities, $U$, $V$, and $C$, both real and imaginary, based on computations using Program B were compared with those quantities using the elementary solution as applied to cascades. Comparisons were made for various values of $A$, stagger, and interblade phase angle. Points along both surfaces of the blades as well as in the flow field were compared. In the case of zero interblade phase angle, values of $A$ as low as 0.25 were compared. The restriction of supersonic leading-edge locus was maintained. All
values compared were within 2% up to reduced frequencies of 0.1. Differences between 1% and 2% were obtained only at Mach numbers near one (\( M = \sqrt{1.25} \)) at a reduced frequency of 0.1. A sample of the flow quantities compared is given in Table 1.

<table>
<thead>
<tr>
<th>#</th>
<th>x</th>
<th>y</th>
<th>C22R (M.O.C.)</th>
<th>C22R (S.S.C.)</th>
<th>C22I (M.O.C.)</th>
<th>C22I (S.S.C.)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.10</td>
<td>0.00</td>
<td>-1.000</td>
<td>-1.000</td>
<td>0.000</td>
<td>0.000</td>
</tr>
<tr>
<td></td>
<td>0.50</td>
<td>0.00</td>
<td>0.659</td>
<td>0.660</td>
<td>1.120</td>
<td>1.120</td>
</tr>
<tr>
<td></td>
<td>0.60</td>
<td>0.00</td>
<td>0.659</td>
<td>0.660</td>
<td>1.120</td>
<td>1.120</td>
</tr>
<tr>
<td></td>
<td>0.10</td>
<td>0.50</td>
<td>0.800</td>
<td>0.800</td>
<td>0.600</td>
<td>0.600</td>
</tr>
<tr>
<td></td>
<td>0.50</td>
<td>0.50</td>
<td>-1.197</td>
<td>-1.200</td>
<td>0.699</td>
<td>0.700</td>
</tr>
<tr>
<td></td>
<td>0.60</td>
<td>0.50</td>
<td>-1.197</td>
<td>-1.200</td>
<td>0.699</td>
<td>0.700</td>
</tr>
<tr>
<td>2</td>
<td>0.20</td>
<td>0.00</td>
<td>-1.000</td>
<td>-1.000</td>
<td>0.000</td>
<td>0.000</td>
</tr>
<tr>
<td></td>
<td>0.50</td>
<td>0.00</td>
<td>0.659</td>
<td>0.660</td>
<td>1.119</td>
<td>1.120</td>
</tr>
<tr>
<td></td>
<td>0.60</td>
<td>0.00</td>
<td>0.659</td>
<td>0.660</td>
<td>1.119</td>
<td>1.120</td>
</tr>
<tr>
<td></td>
<td>0.44</td>
<td>0.50</td>
<td>0.800</td>
<td>0.800</td>
<td>0.600</td>
<td>0.600</td>
</tr>
<tr>
<td></td>
<td>0.74</td>
<td>0.50</td>
<td>-1.197</td>
<td>-1.200</td>
<td>0.699</td>
<td>0.700</td>
</tr>
<tr>
<td></td>
<td>0.84</td>
<td>0.50</td>
<td>-1.197</td>
<td>-1.200</td>
<td>0.699</td>
<td>0.700</td>
</tr>
<tr>
<td>3</td>
<td>0.20</td>
<td>0.00</td>
<td>-1.997</td>
<td>-2.000</td>
<td>0.060</td>
<td>0.060</td>
</tr>
<tr>
<td></td>
<td>0.50</td>
<td>0.00</td>
<td>-1.984</td>
<td>-2.000</td>
<td>0.149</td>
<td>0.150</td>
</tr>
<tr>
<td></td>
<td>0.74</td>
<td>0.00</td>
<td>1.510</td>
<td>1.500</td>
<td>2.203</td>
<td>2.222</td>
</tr>
<tr>
<td></td>
<td>0.80</td>
<td>0.00</td>
<td>1.530</td>
<td>1.522</td>
<td>2.187</td>
<td>2.211</td>
</tr>
<tr>
<td></td>
<td>0.40</td>
<td>1.00</td>
<td>1.627</td>
<td>1.629</td>
<td>1.161</td>
<td>1.162</td>
</tr>
<tr>
<td></td>
<td>0.50</td>
<td>1.00</td>
<td>-2.322</td>
<td>-2.353</td>
<td>1.633</td>
<td>1.650</td>
</tr>
<tr>
<td></td>
<td>0.74</td>
<td>1.00</td>
<td>-2.252</td>
<td>-2.264</td>
<td>1.708</td>
<td>1.738</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>#1</th>
<th>#2</th>
<th>#3</th>
</tr>
</thead>
<tbody>
<tr>
<td>( M = \sqrt{2} )</td>
<td>( M = \sqrt{2} )</td>
<td>( M = \sqrt{1.25} )</td>
</tr>
<tr>
<td>( A = 0.5 )</td>
<td>( A = 0.5 )</td>
<td>( A = 0.5 )</td>
</tr>
<tr>
<td>( \beta = 0.0 )</td>
<td>( \beta = 25.64 )</td>
<td>( \beta = 13.50 )</td>
</tr>
<tr>
<td>( \delta = 36.87 )</td>
<td>( \delta = 36.87 )</td>
<td>( \delta = 36.87 )</td>
</tr>
<tr>
<td>( x_0 = 0.0 )</td>
<td>( x_0 = 0.0 )</td>
<td>( x_0 = 0.0 )</td>
</tr>
</tbody>
</table>

Table 1
Further, comparisons of the out of phase pressure distribution on the upper surface of a staggered cascade blade in supersonic flow of \( M = \sqrt{1.25} \) and \( M = \sqrt{2} \) as predicted by the method of characteristics and the elementary theory are shown in Figure 9 \( (M = \sqrt{2}, B = .12, x_0 = 0, \delta = 0) \) and Figure 10 \( (M = \sqrt{1.25}, B = .12, x_0 = 0, \delta = 0) \). Here, the solid lines indicate the elementary theory, and the dashed lines indicate the method of characteristics.

Figure 9

Agreement between the two methods is quite good for reduced frequencies up to 0.1 at \( M = \sqrt{2} \), but this close agreement is maintained at \( M = \sqrt{1.25} \) only up to a reduced frequency of 0.05. Since both methods are based on the same linearized
equations of motion, this shows that the influence of frequency is greater at the lower supersonic Mach numbers.

Integrated values of lift and moment ($L_3$ and $M_4$) were also compared. Here differences as high as 3% were found at Mach numbers approaching one; however, for $M = \sqrt{2}$, the values were within 2%.

![Figure 10](image-url)

These results were produced using a grid fineness ratio of 100 or one quarter of the capacity of the program in order to reduce the amount of computer time required to make the computations.
F. COMPARISON OF METHOD OF CHARACTERISTICS WITH GARRICK-RUBINOW

Lift and moment calculations obtained from Programs A and B at $A \geq 1$ were compared with those of Garrick and Rubinow (1946) for both pitch and plunge oscillation. Differences were less than 2%. Program B was used to duplicate several different single-degree-of-freedom flutter points shown in Garrick and Rubinow's (1946) Figure 22. Again, the grid fineness ratio used was 100.

G. METHOD OF CHARACTERISTICS RESULTS

The following results were obtained using Program B to compute single-degree-of-freedom flutter speeds and frequencies. Approximately four minutes of CPU time on the IBM-360 were required to generate one flutter point. A grid fineness ratio of 100 was used throughout.

![Figure 11](https://example.com/figure11.png)
In Figure 11 ($M = 1.11, A = 0.95, \delta = 0, x_0 = .5, \mu = 15.708, r_\theta = .5, g = 0$) a typical variation of $M_4$ with $1/k$ is shown for an unstaggered cascade oscillating in phase with only a slight amount of blade interference. Here, the flutter point is at a relatively low value of reduced frequency ($k = 0.02$). That this is not the critical flutter condition is shown in Figure 12 where the non-dimensional flutter speed is plotted versus interblade phase angle for the same parameters as in Figure 11. In complete agreement with Lane (1956), the critical flutter condition, i.e., the lowest flutter speed, for this unstaggered cascade is at an interblade phase angle of 180°.
Figure 13 shows the effect of a small amount of interference on the flutter speed at $M = 1.111$ when plotted against elastic axis position.

Again the blade and flow parameters are those as shown in Figure 11 and Figure 12. It can be seen that a relatively small amount of interference significantly lowers the non-dimensional flutter speed over the entire range of elastic axis positions tested.

In Figure 14 ($A = .95$, $\delta = 180$, $B = 0$, $\mu = 500$, $g = 0$, $r_\theta = .5$) the effect of Mach number on flutter speed is shown for various elastic axis positions. An increase in Mach number to $M = \sqrt{2}$ significantly lowers the non-dimensional flutter speed for a relatively small amount of interference.
In Figure 15 ($\delta = 180, \mu = 15.708, r_0 = .5, g = 0$) the effect of this relatively small amount of interference on the flutter speed is again shown, but here at the higher Mach number ($M = \sqrt{2}$). As before, the non-dimensional flutter speed is significantly lowered over the entire elastic axis range by a small amount of interference.
Figure 16 ($M = \sqrt{2}$, $A = .95$, $B = 0$, $\delta = 180$, $x_o = .5$, $r_\theta = .5$) shows the effect of structural damping on the non-dimensional flutter speed for two values of wing density parameter.

\[
\frac{U_F}{\omega_\theta} = \begin{cases} 
1 & \text{for } \mu = 15.708 \\
2 & \text{for } \mu = 500
\end{cases}
\]

Figure 16

With a relatively small amount of interference, the flutter speed is greatly increased by a large wing density parameter and high structural damping. The high value of wing density parameter is more applicable to the cascade case. As defined in Garrick and Rubinow (1946), its increase may be interpreted as an increase in altitude for a fixed wing density, high values indicating supersonic wings at high altitude. For
typical supersonic cascades, the wing density parameter is of the order of 700.

The remaining results give variations in non-dimensional flutter speed of the staggered cascade ($\beta = 26.565$) at $M = \sqrt{2}$. Results shown are based on a cascade with an interblade phase angle of 180°. It must be re-emphasized that this interblade phase angle is not necessarily the one that gives the critical flutter condition; however, several points at random were chosen and the critical flutter condition found for each. All were within 1% of flutter speed at $\delta = 180$ and within 10° of $\delta = 180$. This was not done for all points shown in an effort to reduce the total amount of computer time used.

In Figure 17 ($\mu = 500$, $g = .02$, $\tau_0 = .5$) the effect of an increasing amount of interference on flutter speed is shown for various elastic axis positions. The cascade is assumed
to have a high wing density parameter and 2% structural damping coefficient. An increase in the amount of interference is seen to lower the non-dimensional flutter speed.

\[ U_F / \Delta \theta \]

Figure 18

Figure 18 (A = .6, \( r_\theta = .5 \)) shows the effect on the non-dimensional flutter speed of increasing the wing density parameter then the structural damping for various elastic axis positions. Without structural damping, an increase in wing density parameter had little effect on the flutter speed. With 2% structural damping, however, the flutter speed was greatly increased. The effect of structural damping is shown further in Figure 19 (A = .6, \( x_0 = .5 \), \( r_\theta = .5 \)).

Here, the variation of flutter speed with structural damping is shown for three values of wing density parameter. As can be seen, increasing the structural damping increases the
difference in flutter speeds for the three values of wing density parameter. Thus, the accurate determination of the wing density parameter becomes increasingly more important with increased structural damping.
Figures 20 and 21 show the same parameters as Figures 18 and 19, respectively, but here, there is more interference \((A = .25)\). The results are relatively the same as those shown in Figures 18 and 19. Flutter speeds overall are lower due to the increase in the amount of interference, and the effect of structural damping is more pronounced.
VIII. CONCLUSIONS AND RECOMMENDATIONS

The elementary theory for cascades with supersonic leading-edge locus is valid for low frequencies and can be used to give rapid calculations of lift forces and pitching moments of cascades as well as their torsional stability boundaries. These results can immediately be applied to a wide variety of wind tunnel and airfoil interference problems.

The method of characteristics can be used in conjunction with the high-speed computer to give rapid, accurate flutter calculations of cascades with supersonic leading-edge locus oscillating at arbitrary frequency. This applies to torsion and bending flutter calculations as well as to torsion flutter alone, for the only differences in the calculations are those due to different values of normal velocity at the blade surface. In each case, torsion or bending, the method of calculating the flow field and determining the lift forces and moments is identical.

Results of single-degree-of-freedom (torsional) flutter calculations show that interference on cascade blades due to reflected Mach waves from adjacent blades lowers flutter speeds significantly and in proportion to the amount of this interference. The presence of structural damping, however, can partially alleviate this adverse situation by increasing flutter speeds, provided the wing density parameter is large.
Recommendations for future study include the development of the elementary theory for bending oscillations, and the generation of numerical results for two-degree-of-freedom flutter using Program A. Of primary importance, however, is the development of a method of characteristics approach for cascades with subsonic leading-edge locus. This is the critical problem today in the design of the supersonic compressor. With this problem properly formulated, the method of characteristics should be able to give rapid, accurate flutter predictions for cascades oscillating in this complicated flow field.
APPENDIX A

COMPUTER PROGRAM VARIABLES

I. LOGIC VARIABLES

The following variables and their definitions, in alphabetical order appear in the logic statements of the programs. The asterisk is placed behind those variables that appear only in Program A.

- **FCONST*** Denotes type of oscillation: TRUE - plunge mode, FALSE - pitch mode.
- **IATNWL** When computation reaches upper airfoil, it takes on the value "1", otherwise, "0".
- **ICO** Takes on the value "1" when lower airfoil computation is complete, otherwise "0".
- **ICOUNT** Incremented only when MACHLN is called by being set equal to JLINE, zeroed when HIFOIL is called.
- **IFIN** Set equal to "1" when integration on lower airfoil is complete, otherwise "0".
- **IHAVEP** The number of the computation in any row starting with "0".
- **IJUNC** The grid point at the intersection of the initial right and left-running Mach lines.
- **IJUNT** Equal to IJUNC, needed to preserve the value initially calculated in case of multiple flow field calculations.
- **ILINE** The number of grid points in the row (right-running Mach line) being computed, plus one.
- **IMAGRT*** If X is negative, set equal to "1", otherwise "0".
- **IREF** Set equal to "1" if lower airfoil point is in Zone I, otherwise "0".
- **ISPLIT** Set equal to 1/2 of the number of grid points minus the number of stagger points.
ISWICH  Set equal to "1" or "2" alternately starting with "1" and changing when upper airfoil calculation goes into a different zone. The upper airfoil zone marker.

ISWTCH  Switch variable used to choose the row being calculated, alternately set equal to "1" or "2". Used as a second subscript.

ITEM  Loop variable for pitch and plunge mode calculation.

IZOT*  Set equal to "1" if flutter frequency ratio is imaginary, otherwise "0".

JCOUNT  Set equal to "0" when in Zone I, incremented when MACHLN is called in an interference region.

JFIN  Set equal to "1" when integration along upper airfoil is complete, otherwise "0".

JLINE  The number of grid points in the row being computed.

JSWICH  The same as ISWICH but used in LOFOIL. The lower airfoil zone marker.

JSWTCH  Switch variable used to choose the preceding line that was calculated, set equal to the opposite value as that of ISWTCH. Used as a second subscript.

JITEM  Loop variable for frequency range incrementing.

KCOUNT  Used in the interference calculation in MACHLN, set equal to "JCOUNT-1".

KOUNT  Set equal to "0" until HIFOIL is called, then set equal to the number of grid points minus one.

LCOUNT  The number starting with "0" of each point in a zone of interference in LOFOIL.

MCOUNT  Same as LCOUNT but used in HIFOIL.
II. QUANTITY VARIABLES

The following variables in the programs take on the values as defined below. All have real values unless otherwise stated. The asterisk is placed behind those variables that appear only in Program A and the accent is placed behind those variables that appear only in Program B. The dimensioned variables are listed first.

\( U_{22R}, U_{22I}, V_{22R}, V_{22I}, C_{22R}, C_{22I}(400,3) \). The velocities, real and imaginary, of a point in the program. They are equivalent to the respective Teipel amplitude functions, \( U_{22R} \) through \( C_{22I} \).

\( U_{33R}, U_{33I}, V_{33R}, V_{33I}, C_{33R}, C_{33I}(200,2) \). Same as above, but used to calculate the flow field in Zone \( I' \) when the phase angle is non-zero.

\( X, Y(400,2) \) The \( x \) and \( y \) coordinates of a point based on the chord length.

\( A_I \) \( A_I \) as defined in Eq. (96a).

\( B_I \) \( B_I \) as defined in Eq. (96b).

\( \text{CAPX}' \) \( X \) as defined in the Table of Symbols.

\( \text{CPI1} \) The imaginary part of \( e^{-i\delta}(C_{22R}+iC_{22I}) \) of the preceding point calculated on the upper airfoil.

\( \text{CPR1} \) The real part of \( e^{-i\delta}(C_{22R}+iC_{22I}) \) of the preceding point calculated on the upper airfoil.

\( \text{CPI2} \) The imaginary part of \( e^{-i\delta}(C_{22R}+iC_{22I}) \) of the point being calculated on the upper airfoil.

\( \text{CPR2} \) The real part of \( e^{-i\delta}(C_{22R}+iC_{22I}) \) of the point being calculated on the upper airfoil.

\( \text{DAMP}' \) The total system damping as defined in Eq. (218b).

\( \text{DELCI} \) The value of \( C_{22I} \) at \( x = 1 \).
DELCPI  The imaginary part of $e^{-i\delta}(\text{DELCR} + i\text{DELCI})$ on the upper airfoil.

DELCPR  The real part of $e^{-i\delta}(\text{DELCR} + i\text{DELCI})$ on the upper airfoil.

DELCR   The value of $C_{22R}$ at $x = 1$.

DELL1, DELL2, DELM1, DELM2.  The change in lift (real and imaginary) and moment, respectively, due to extrapolation to $x = 1$.

DELTA   The phase angle in radians.

DELTAS  The size of the length increment along a Mach line.

DISC*   The discriminant of Eq. (206).

DSTSTR  The size of the $\Delta x$ increment.

FACTOR' Set equal to $\Delta x/k^2$.

FACTOR* In the MAIN, set equal to $\Delta x/k^2$ for the pitch mode, and set equal to $1/2 \Delta x/k^2$ for the plunge mode. In FLUTER, set equal to DISC - $C_R$.

FAZE    The phase angle in degrees.

FIN     The chord length.

FMANGL  The Mach angle in radians.

FNGDPT  The real number value of the number of grid points on the initial Mach line.

FNGRDN  The real number value of the grid fineness ratio, equal to FNGDPT.

FSTRMN  The freestream Mach number.

G1,...,G9 Intermediate factors used in velocity calculations in MACHLN, LOFOIL, GENFPT, and HIFOIL.

GSUBA*  The structural damping coefficient, $g$, as defined in the Table of Symbols.

HAFREQ* The ratio of the natural frequency of the airfoil in plunge to that in pitch.

HDSTRL  The increment, $\Delta x/2$. 

93
INCRE

The amount that $2/k$ is incremented each time the lift and moment is calculated.

K_{12R},...,K_{56I} \quad K_{12R},...,K_{56I} \text{ as defined in Eq. (95).}

L_1, L_2 \quad \text{The real and imaginary lift coefficients, respectively. They become } L_1 \text{ and } L_2 \text{ or } L_3 \text{ and } L_4 \text{ as defined in Eq. (186) and Eq. (187) depending on the type of oscillation.}

L_3, L_4^* \quad \text{The real and imaginary lift coefficients, respectively, due to pitching oscillations.}

M_1, M_2 \quad \text{The real and imaginary moment coefficients, respectively. They become } M_1 \text{ and } M_2 \text{ or } M_3 \text{ and } M_4 \text{ as defined in Eq. (188) and Eq. (189) depending on the type of oscillation.}

M_3, M_4^* \quad \text{The real and imaginary moment coefficients, respectively, due to pitching oscillations.}

MUU \quad \text{The wing density parameter as defined in the Table of Symbols.}

NGDPTS \quad \text{The grid fineness ratio plus one.}

NGRDFN \quad \text{The grid fineness ratio, it must be an even number less than 400. It is an integer.}

NSTPTS \quad \text{The number of stagger grid points, an integer, set equal to } B/Ax \text{ plus one.}

OMEGA \quad \text{The non-dimensional flutter frequency as defined in Eq. (208).}

OMEGAA \quad \Omega_\theta \text{ as defined in the Table of Symbols.}

OMEGAH^* \quad \Omega_\text{h} \text{ as defined in the Table of Symbols.}

REDFRQ \quad \text{The reduced frequency.}

RFREQ \quad 2/k.

ROOTX \quad \text{The square root of } X.

ROOTX_1^* \quad \text{The square root of the first root of Eq. (206).}

ROOTX_2^* \quad \text{The square root of the second root of Eq. (206).}

ROOTXI^* \quad \text{The square root of Eq. (207).}
The radius of gyration, \( r_0 \), as defined in the Table of Symbols.

The ratio of specific heats.

The square root of \( M^2-1 \).

The stagger angle in degrees.

The amount of stagger, \( B \).

\( M^2-1 \).

The difference between the respective roots of Eq. (206) and that of Eq. (207).

The values of \( \text{TEST1} \) and \( \text{TEST2} \), respectively, for the previous frequency computation.

The perpendicular distance between the two airfoils, \( d \).

The chord length of the top airfoil, \( x-B \).

The increment, \( \frac{1}{2} \Delta x \cos \alpha \).

The factor, \( \cos(\frac{k}{M^2-1}x) \).

The non-dimensional flutter speed as defined in Eq. (209).

The values of \( U, U_1, V, V_1, C, C_1 \), respectively, for calculations on the initial Mach line between the point of intersection of the initial left and right-running Mach lines and the upper airfoil.

The factor, \( \sin(\frac{k}{M^2-1}x) \).

The imaginary part of the normal flow velocity at the airfoil surface.

The real part of the normal flow velocity at the airfoil surface.

\( M^2 \).

\( X_1 \) as defined in Eq. (207).

The length of the initial Mach line.

The roots of Eq. (206).
XUBA* The distance from the elastic axis of the airfoil to its center of gravity.

XUBO The distance from the leading-edge of the airfoil to its elastic axis.
APPENDIX B

FLOW DIAGRAM - PROGRAM A

I. MAIN PROGRAM

Read/Write DATE
CALL INPUT
IZOT = 0

1100
KTEM = 1,4

1005
JTEM = 1,50

FCONST = .FALSE.

1002
ITEM = 1,2

ICO = 0
KOUNT = 0
CALL INITIAL
1000
CALL COMPXY

IHAVEP.EQ.0 .AND.
IATNWL.EQ.0

CALL MACHLN

IHAVEP.EQ.0 .AND.
IATNWL.NE.0

CALL HIFOIL

IATNWL.EQ.1 .AND.
IHAVEP.EQ.KOUNT

CALL LOFOIL

IHAVEP.EQ.JLINE

5

97
CALL FLUTER

IF(IMGRT NE.1) THEN
   IZOT = 1
   RFREQ = RFREQ + INCRE
   REDFRQ = 2/RFREQ
   WRITE OMEGA, UF, JTEM
   XSUBO = XSUBO + 0.1
ENDIF

1005

IF(IZOT .NE. 1) THEN
   RFREQ = RFREQ-1
   INCRE = INCRE/5
ENDIF

IF(REDFRQ .LT. 1E-05) THEN
   IF(IZOT .NE. 1) THEN
      RFREQ = RFREQ+INCRE
      REDFRQ = 2/RFREQ
      WRITE OMEGA, UF, JTEM
      XSUBO = XSUBO + 0.1
   ENDIF
   IF(JTEM .NE. 1) OR (IZOT .NE. 1) THEN
      Define TEST3, TEST4
      ABS(TEST1).LE.1E-05 .OR. ABS(TEST2).LE.1E-05
      IF(TEST1.LT.-1E-05) THEN
         TEST3.LT.-1E-05
      ELSE
         TEST2.LT.-1E-05
         TEST4.LT.-1E-05
         IF(TEST3.LT.-1E-05) THEN
            TEST3 = TEST1
            TEST4 = TEST2
         ENDIF
      ENDIF
      IF(TEST4.GT.1E-05) THEN
         TEST4 = TEST2
      ELSE
         TEST4 = TEST2
      ENDIF
      IF(TEST3.GT.1E-05) THEN
         TEST3 = TEST1
      ELSE
         TEST3 = TEST2
      ENDIF
      IF(TEST2.GT.1E-05) THEN
         TEST2 = TEST4
      ELSE
         TEST2 = TEST4
      ENDIF
      IF(TEST1.GT.1E-05) THEN
         TEST1 = TEST2
      ELSE
         TEST1 = TEST3
      ENDIF
   ENDIF
ENDIF

1100

STOP
END
II. SUBROUTINE INPUT

Read/Write NAMELIST
Calculate DSTSTR, NSTPTS

\[
\text{STGR-DSTSTR.GE.0.50} \quad \text{T} \quad \text{NSTPTS = NSTPTS+1}
\]

\[
\text{NGDPTS/2.EQ.NSTPTS-1} \quad \text{T} \quad \text{NSTPTS = NSTPTS-1}
\]

\[
\text{NGDPTS/2.EQ.NSTPTS+1} \quad \text{T} \quad \text{NSTPTS = NSTPTS+1}
\]

Calculate STGR, STGANG, IJUNT, NSTPTS, ISPLIT, S, DELTA, OMEGAA, OMEGAM
Write NAMELIST
RETURN
END

III. SUBROUTINE INITIAL

Define IJUNC, AI, BI

\[
.\text{NOT.FCONST} \quad \text{T}
\]

Define VRPANL, VIPANL

Calculate U22R = C22I(1,1), X(1,1), Y(1,1)

\[
\text{ISPLIT.NE.-1} \quad \text{.AND.}\quad \text{DELTA.LE.1E-05} \quad \text{T}
\]

\[
\text{F}
\]

Calculate L1, L2, M1, M2

\[
64
\]
DELTA.LE.1E-05
.F
.NOT.FCONST
.F
Define VRPANL,VIPANL
Define VRPANL,VIPANL
Calculate U33R -> C33I(1,1)
ISPLIT.EQ.-1
.T
Calculate L1,L2,M1,M2
ISWITCH = 2
JSWITCH = 1
ISWICH = 1
JSWICH = 1
IFIN = 0
JFIN = 0
ILINE = 2
JLINE = 1
IHAVEP = 0
IATNWL = 0
JCOUNT = 0
LCOUNT = 0
MCOUNT = 1
RETURN
END
IV. SUBROUTINE MACHLN

ICOUNT = JLINE

ICOUNT.LT.IJUNC

T

ICOUNT = JCOUNT+1
KCOUNT = KCOUNT-1

F

KCOUNT.NE.0

F

Define UR, UI, VR, VI, CR, CI

Define UR, UI, VR, VI, CR, CI

T

Calculate \( U_{22R} = C_{22I(1,ISWTC)} \)

Define W, T

.NOT. FCONST

T

Define VRPANL, VIPANL

Define VRPANL, VIPANL

F

Calculate \( U_{22R} = C_{22I(1,ISWTC)} \)

ICOUNT.NE.ISPLIT

.OR.

DELTA.GT.1E-05

F

Define \( U_{22R} = C_{22I(1,3)} \)

T

.NOT. FCONST

F

Define VRPANL, VIPANL

Define VRPANL, VIPANL

F

ICOUNT.NE.ISPLIT

T

Define \( U_{22R} = C_{22I(1,3)} \)

F

RETURN

END
V. SUBROUTINE HIFOIL

Define TOPCRD

\( \text{JLINE.EQ.NGDPTS-1} \quad T \quad \rightarrow \quad \text{IATNWL} = 0 \quad \)

\( \text{ICOUNT.EQ.0} \quad T \quad \)

\( \text{JCOUNT.NE.0} \quad T \quad \)

Define \( K12R, K12I, J \)

\( \text{ICOUNT.NE.0} \quad T \quad \)

Define \( I, J, K12R, K12I \)

\( \text{.NOT.FCONST} \quad T \quad \)

Define \( V22R, V22I(1, \text{ISWTCH}) \)

Calculate \( U22R, U22I, C22R, C22I(1, \text{ISWTCH}), \) 
\( \text{CPR1, CPI1, CPR2, CPI2} \)

\( \text{JFIN.EQ.1} \quad T \quad \)

\( \text{LCOUNT.EQ.0 .AND. TOPCRD+DSTSTR.GT.1.0} \quad T \quad \)

\( \text{LCOUNT.EQ.0 .OR. TOPCRD+DSTSTR.GT.1.0} \quad T \quad \)

Calculate \( L1, L2, M1, M2 \)

\( \text{LCOUNT} = \text{LCOUNT}+1 \)

130
ISWICH.NE.1
T

LCOUNT.NE.JCOUNT
T

LCOUNT.EQ.IJUNT
F

JCOUNT.EQ.0
F

TOPCRD+DSTSTR.LE.1.0
T

Calculate L1, L2, M1, M2
LCOUNT = 0

ISWICH = 2

ISWICH.EQ.1
T

F

ISWICH = 1

JFIN = 1

ICOUNT = 0
RETURN
END

ICOUNT.EQ.0
T
VI. SUBROUTINE GENFPT

Define I

IATNWL.EQ.0

T

Define K12R, K12I, K34R, K34I

Define K12R, K12I, K34R, K34I

Calculate U22R → C22I(I,ISWITCH)

ICOUNT.NE.ISPLIT OR. DELTA.GT.1E-05

T

Define U22R → C22I(I,3)

DELTA.LE.1E-05 OR. JLINE.GT.ISPLIT

F


Calculate U33R → C33I(I,ISWITCH)

JLINE.NE.ISPLIT

T

Define U22R → C22I(I,3)

RETURN

END
VII. SUBROUTINE LOFOIL

. NOT.FCONST
  F
  T
  Define V22R,V22I(ILINE,ISWTCH)
  Define V22R,V22I(ILINE,ISWTCH)
  Define K12R, K12I, K34R, K34I
  Define K56R, K56I
  Define K56R, K56I
  IAINWL.EQ.0
    F
    T
    Define K56R, K56I
    Define K56R, K56I
  Calculate U22R, U22I, C22R, C22I(ILINE,ISWTCH)
  ICOUNT.NE.ISPLIT
    OR.
    DELTA.GT.1E-05
      T
      Define U22R → C22I(ILINE,3)
      Define U22R → C22I(ILINE,3)
    F
    DELTA.LE.1E-05
      OR.
      JLINE.GT.ISPLIT
        T
        Define U22R → C22I(ILINE,3)
        Define U22R → C22I(ILINE,3)
      F
      Define V33R,V33I(ILINE,ISWTCH)
      Define V33R,V33I(ILINE,ISWTCH)
      Calculate U33R, U33I, C33R, C33I(ILINE,ISWTCH)
      JLINE.NE.ISPLIT
        T
        Define U22R → C22I(ILINE,3)
        Define U22R → C22I(ILINE,3)
        88
        94
Define CPR1, CPI1, CPR2, CPI2

IFIN.EQ.1   
   T
   Calculate L1, L2, M1, M2

JLINE.NE.ISPLIT        .AND.  
   X(K,L)+DSTSTR.LE.1.0   
   T
   Calculate L1, L2, M1, M2

IFIN.EQ.1   
   F

IREF = 0

IFIN.EQ.1   
   F

MCOUNT.EQ.0   .AND.  
   X(ILINE,ISWTCH)+DSTSTR.GT.1.0   
   T
   Calculate L1, L2, M1, M2

MCOUNT.EQ.0   .OR.  
   X(ILINE,ISWTCH)+DSTSTR.GT.1.0   
   T
   Calculate L1, L2, M1, M2

Calculate L1, L2, M1, M2

IREF.EQ.1   
   T
   130

IREF.EQ.1   
   F

MCOUNT = MCOUNT+1

JSWITH.NE.1   
   F
   110

MCOUNT.EQ.IJUNC   
   F
   120

MCOUNT.NE.ISPLIT+1   
   T
   95

Calculate L1, L2, M1, M2

IREF.EQ.1   
   T
   140

IREF.EQ.1   
   F
Calculate DELL1, DELL2, DELM1, DELM2

DELTA.GT.1E-05

IFIN = 1

RETURN

END
VIII. SUBROUTINE COMPXY

IX. SUBROUTINE FLUTER
PROGRAM A  --  TWO DEGREE OF FREEDOM FLUTTER

THIS PROGRAM COMPUTES THE FLUTTER FREQUENCY AND FLUTTER VELOCITY OF A CASCADE OSCILLATING AT LOW AMPLITUDES WITH ARBITRARY STAGGER, PHASE LAG, BLADE SPACING, AND FREQUENCY. THE CASCADE MUST HAVE SUPersonic Leading Edge LOCUS.

DIMENSION U22R(400,3),U22I(400,3),V22R(400,3),V22I(400,3),
1C22R(400,3),C22I(400,3),X(400,2),Y(400,2)
DIMENSION DATE(3)
DIMENSION U33R(200,2),U33I(200,2),V33R(200,2),V33I(200,2),C33R(200,2),C33I(200,2)

COMMON/BLK1/NGRID,F,STRMN,RTOSPH,REDFRQ,TSUB0,TNWDST,FMANGL,XLENTH
1,DELTAS,NGDPTS,DSTSTR,HDSRRL,TRNGLH,U22R,U22I,V22R,V22I,C22R,C22I
3,M2,K12R,K12I,K34R,K34I,K56R,K56I,VRPANL,VIPANL

COMMON/BLK2/L3,L4,M3,M4
COMMON/BLK3/MUU,RSUBA,VSUBA,HAFREQ,OMEGA1,OMEGA2,INCRE
COMMON/BLK4/ROOTX1,ROOTX2,ROOTX1,OMEGA1,UF,RFREQ,IMGRT
REAL MUU,INCRE

COMMON/BLK5/U33R,U33I,V33R,V33I,C33R,C33I

LOGICAL FCNST
COMMON/FOFX/FCONST
COMMON/FCOR/KOUNT
COMMON/JUNC/JUNT
COMMON/STG/STGR
COMMON/FINE/FIN
COMMON/IFINE/IFIN,JFIN
REAL K12R,K12I,K34R,K34I,K56R,K56I
REAL L1,L2,L3,L4,M1,M2,M3,M4

PRINT NAME OF PROGRAM/DATE OF RUN.

READ(5,7)DATE
7 FORMAT(3A4)
WRITE(6,6)DATE
6 FORMAT(1H1,14(/),14(/),37X,51H OSCILLATING CASCADE FLUTTER PROGRAM)

---RUN OF----,3A4,/,1H1)

PRINT OUT ALL INPUT INFORMATION VIA INPUT.

CALL INPUT
IZOT=0
DO 1005 JTEM=1,50
FCONST=.FALSE.
DO 1002 ITEM=1,2

00000000
00000001
00000002
00000003
00000004
00000010
00000020
00000030
00000040
00000050
00000060
00000070
00000080
00000090
00001000
00001100
00001200
00001300
00001400
00001500
00001600
00001700
00001800
00001900
00002000
00002100
00002200
00002300
00002400
00002500
00002600
00002700
00002800
00002900
00003000
00003100
00003190
00003200
00003300
00003400
ICQ=0
KOUT=0
CALL INITIAL
1000 CONTINUE

COMPXNY COMPUTES THE VALUE FOR X AND Y GIVEN THE PARAMETERS IHAVEP
1 AND ISWITCH FOR INDICES (RESPECTIVELY).
CALL COMPXNY

THE PROGRAM GOES TO 1 IF POINT IS ON INITIAL MACH LINE.
IF((IHAVEP.EQ.0).AND. (IATNWL.EQ.0)) GO TO 1
IF((IHAVEP.EQ.0).AND. (IATNWL.NE.0)) GO TO 2

TEST HERE IF YOU ARE AT A LOW FOIL PT. IF NOT GO TO NEXT POINT.
IF(IATNWL.EQ.1 .AND. IHAVEP.EQ. KOUNT) GO TO 3
IF(IHAVEP.EQ.JLINE) GO TO 3
GO TO 4

INITIAL POINT(MACH LINE) CALC. FOR U,V,AND C AT 1.
1 CALL MACHLN
GO TO 5

HIGH FOILN CALCULATION HERE FOR U,V,AND C AT 2.
2 CALL HIFOIL
GO TO 5

LOW FOIL PT CALCULATION HERE FOR U,V,AND C AT 3.
3 CALL LOFOIL
GO TO 5

GENERAL POINT CALCULATION HERE FOR U,V,AND C AT 4.
4 CALL GENFPT
5 CONTINUE
IF(IATNWL.EQ.1 .AND. IHAVEP.EQ. KOUNT) GO TO 25
IF(IHAVEP.NE.JLINE) GO TO 100

TEST IF AT END OF FOIL IF SO COMPUTE Q AND QUIT.
25 IF(X(IHAVEP+1,ISWITCH)+DSTSTR.GT.FIN) GO TO 101

TEST IF YOU ARE AT A HI AIRFOIL LINE(RIGHT RUNNING MACH) AND IF
ISO NOTE THIS BY IATNW=1 AND JLINE=NGDPTS.

IF(ILINE.EQ.NGDPTS)GO TO 102
INCREMENT FOR NEXT LINES (FIRST OLD ONE (ILINE) BECOMES LAST NEW 1)
ILINE=ILINE+1
JLINE=JLINE+1
IF(ILINE.EQ.NGDPTS) IATNW=1
SWITCH LINES HERE SO FIRST OLD ONE (ILINE) BECOMES LAST NEW ONE.
ZERO OUT NEW LINE POINT INCREMENT COUNTER.

105 IF(ICO.EQ.1) IJUNC=IJUNC-1
IF(ISWTCH.EQ.1) GO TO 103
ISWTCH=1
JSWTCH=2
IHAVEP=0
GO TO 1000

103 ISWTCH=2
JSWTCH=1
IHAVEP=0
GO TO 1000

AT 102 SET UP FOR HERE ON IN AT TUNNEL WALL.

102 IATNW=1
JLINE=NGDPTS
KOUNT=NGDPTS-1
GO TO 105

AT 100 INCREMENT TO NEXT POINT ALONG PRESENT LINE.

100 IHAVEP=IHAVEP+1
FINISHED TOP AIRFOIL? TERMINATE
IF((ICO.EQ.1).AND.(IJUNC.EQ.0)) GO TO 106
PREVENT UNNECESSARY FLOW FIELD CALCULATION.
IF((ICO.EQ.1).AND.(IHAVEP.GT.IJUNC)) GO TO 105
GO TO 1000

IN ZONE 1? TERMINATE

101 IF((JLINE-1).LT.IJUNC) GO TO 106
ICO=1
GO TO 105

106 IF(.NOT.FCONST) GO TO 108

C
C FINISH PLUNGE MODE CALCULATION.
C
FACTOR=HDSTRL/(REDFRQ*REDFRQ)
L1=L1*FACTOR
L2=L2*FACTOR
M1=M1*FACTOR*2.0
M2=M2*FACTOR*2.0
GO TO 110

C
C FINISH PITCH MODE CALCULATION.
C
108 FACTOR=DSTSR/(REDFRQ*REDFRQ)
L3=L1*FACTOR
L4=L2*FACTOR
M3=M1*FACTOR*2.0
M4=M2*FACTOR*2.0
110 FCONST=.TRUE.
1002 CONTINUE
CALL FLUTER
IF(IMAGRT.NE.1) GO TO 200
IZOT=1
GO TO 280
200 IF((JTEM.NE.1).OR.(IZOT.NE.1)) GO TO 210
TEST3=ROOTX1-ROOTXI
TEST4=ROOTX2-ROOTXI
210 TEST1=ROOTX1-ROOTXI
TEST2=ROOTX2-ROOTXI
IF(KAUEQTEST1).LE.1E-05).OR.(ABS(TEST2).LE.1E-05)) GO TO 1006
IF(TEST1.LT.-1E-05) GO TO 220
IF(TEST3.LT.-1E-05) GO TO 250
GO TO 230
220 IF(TEST1.GT.1E-05) GO TO 250
230 IF(TEST2.LT.-1E-05) GO TO 240
IF(TEST4.LT.-1E-05) GO TO 250
GO TO 260
240 IF(TEST4.GT.1E-05) GO TO 250
GO TO 260
250 RFREQ=RFREQ-INCRA
INCRA=INCRA/5.0
IF(INCRE.LE.1E-05) GO TO 1006
GO TO 280
260 TEST3=TEST1
TEST4=TEST2
280 RFREQ=RFREQ+INCRA
REDFRQ=2.0/RFREQ
1005 CONTINUE
1006 WRITE(6,501) OMEGA,UF
501 FORMAT(///,15X,19H FLUTTER FREQUENCY,E20.7,15X,18H FLUTTER VELOC
LITY,E20.7)
WRITE(6,502) JTEM
502 FORMAT(///,12H ITERATIONS=,I5)
STOP
END

SUBROUTINE INPUT

SUBROUTINE INPUT READS ALL INPUT.
NGDFN IS THE FINENESS OF GRID NUMBER.
FSTRMN IS THE FREESTREAM MACH NUMBER.
RTOSPH IS THE RATIO OF SPECIFIC HEATS.
REDRFQ IS THE REDUCED DIMENSIONLESS FREQUENCY.
TNWST IS THE DISTANCE BETWEEN AIRFOILS.
XSUBO IS THE DISTANCE (DIMENSIONLESS) FROM THE LEADING EDGE TO THE ELASTIC AXIS.
STGANG IS THE CASCADE STAGGER ANGLE IN DEGREES.
FAZE IS THE UPPER AIRFOIL PHASE LAG IN DEGREES.
DELTA IS THE UPPER AIRFOIL PHASE LAG IN RADIANS.
MUUV IS THE WING DENSITY PARAMETER.
RSUBA IS THE RADIUS OF GYRATION.
XSUBA IS THE DISTANCE (DIMENSIONLESS) OF THE CENTER OF GRAVITY FROM THE ELASTIC AXIS.
HAFREQ IS THE RATIO OF THE BENDING NATURAL FREQUENCY OF THE AIRFOIL TO THE TORSIONAL NATURAL FREQUENCY.
INCRE IS THE AMOUNT THE REDUCED FREQUENCY IS DECREASED EACH TIME SQRT(X) IS CALCULATED.
OMEGA IS MUUV*RSUBA**2.
OMEGAH IS MUUV*THE RATIO OF HFREQ TO AFREQ**2.
RODTX IS THE RATIO OF AFREQ TO THE FREQUENCY OF OSCILLATION.
HSSUBA & ALPHA0 ARE THE MAXIMUM AMPLITUDE (DIMENSIONLESS) OF THE 1 AIRFOIL PLUNGING & PITCHING OSCILLATION, RESPECTIVELY. EACH IS SET EQUAL TO 1 SINCE THEIR VALUES ARE INDEPENDENT OF THE COMPUTATION.
FMANGL IS THE MACH ANGLE.
XLNGTH IS THE LENGTH OF THE INITIAL MACH LINE.
DELTA IS THE STEP SIZE OF INCREMENTS ALONG THE MACH LINES.
DIMENSION U22R,(400,3),U22I,(400,3),V22R,(400,3),V22I,(400,3),
1C22R,(400,3),C22I,(400,3),X(400,2),Y(400,2)
NGDPTS IS THE NUMBER OF GRID POINTS (INCL. XSUBO AND TW POINT).
DSTSTR IS THE DISTANCE (HORIZONTAL) OF A STREAMLINE FROM ONE GRID.
1POINT TO THE NEXT STREAMLINE GRID POINT.
HDSTRL IS ONE-HALF OF DSTSTR.
TRNGLH IS THE HEIGHT OF Deltas(VERTICAL)
U22R IS THE VARIABLE HOLDING THE REAL PART OF U AT THE RIGHTMOST
GRID POINT. THE SECOND INDEX POINTS TO A LINE(MACH) BEING USED.
SAME TYPE OF MEANING FOR THE OTHER VARIABLES.
AI AND BI ARE USED AS CONSTANTS IN THE DEFINITIONS OF U,V, AND C.
COMPUTE ALL VALUES FOR INITIAL MACH LINES INITIAL POINT.
SET UP INITIAL INTEGRATION VALUES.
ISwITCH IS THE SWITCH VARIABLE FOR CHOOSING THE LINE TO WORK ON.
JSWITCH IS THE OPPOSITE OF ISWITCH. IF ISWITCH=1, JSwITCH=2 VICE-VERSA.
LINE CONTAINS THE NUMBER OF POINTS THE LINE EVENTUALLY SHOULD BE.
JLine=ILINE-1 UNLESS AT HI AIRFOILL THEN JLINE=ILINE(JLINE IS TOTAL POINTS IN THE OTHER LINE).
Ihave IS A COUNTER TELLING THE NUMBER OF PROCESSED POINTS YOU HAVE.
CALCULATED FOR THE LINE YOU ARE WORKING ON.
IATNWl=0 SIGNIFIES YOU ARE NOT AT THE HI AIRFOILL YET. IATNWl WILL
BE SET=1 WHEN FINISHED WITH LINE =NGDPT(S AT HI AIRFOILL).

COMMON/BLK1/NGRDFN, FSTRMN, RTOSPH, REDFRQ, XSUBO, TNW DST, FMANGL, XLMNTH 00001980
1, DELTAS, NGDPTs, DSTSTR, HDSTRL, TRNGLH, U22R, U221, V22R, V221, C22R, C221, 00001990
2X, Y, S, DELTA, ISWITCH, JSWITCH, ILINE, ILINE, IHAVEP, IATNWl, AI, BI, L1, L2, M100002000
3, M2, 0, K12R, K121, K34R, K341, K56R, K561, VRPANL, VIPANL
4, STGANG, NSTPTs, ICOUNT, JCOUNT, IJUNCl, ISPLIT
5, LCOUNT, MCOUNT, ISWITCH, JSWITCH
COMMON/BLK3/MUU, XSUBA, XSUBB, HAFREQ, OMEGAA, OMEGAB, INCRE
COMMON/JUNCl/IJUNCl
COMMON/STG/STGR
COMMON/FINE/FIN
REAL K12R, K121, K34R, K341, K56R, K561
REAL L1, L2, L3, L4, M1, M2, M3, M4
REAL MUU, INCRE
NAMeLIST/NAM1/NGRDFN, FSTRMN, RTOSPH, REDFRQ, XSUBO, TNW DST, STGANG, FASE 00002100
1, MUU, XSUBA, XSUBB, HAFREQ, INCRE, FIN
READ(5, NAM1)
WRITE(6, NAM1)
FMANGL=AR SIN(1.0/FSTRMN)
FMANGL=ASIN(1.0/FSTRMN)
FMANGL=FMANGL/NGRDFN
FMANGL=FMANGL/NGRDFN
FMANGL=FMANGL/NGDPTs
FMANGL=FMANGL/NGDPTs
DEL TAs=XLMNTH/FMGDP T
TRNGLH=TNW DST/FMGDP T
HDSTRL=DEL TAS*COS(FMANGL)
HDSTRL=HDSTRL*2.0
STG R=TNW DST*TAN(STGANG*0.1745329E+01)
NSTPTs=STGR/DSTSTR
NSTPTs=STGR/DSTSTR
NSTPTs=NSTPTs*DSTSTR
IF(STGR-DSTSTR, GE, 0.50) NSTPTs=NSTPTs+1
IF(NGDPTs/2.EQ.,NSTPTs-1) NSTPTs=NSTPTs-1
00001900
IF(NGDPTS/2.EQ.NSTPTS+1) NSTPTS=NSTPTS+1
STGR=NSTPTS*1.0
STGANG=ATAN(STGR/TNWSTD)*57.29578
IJUN=NGDPTS/2+NSTPTS
NSTPTS=NSTPTS+1
ISPLIT=NGDPTS/2-NSTPTS
NGDPTS=NGDPTS+1
S=SQRT(FSTRMN*FSTRMN-1.0)
DELA=FAZE*0.17453290-01
OMEGA=MUU*RSUBA**2
OMEGAH=MUU*HAFREQ**2
WRITE(6,5) NGRDFN, FSTRMN, RTOSPH, REDFRQ, TNWSTD, XSUBO, STGANG, DSTSTR
1
MACH NUMBER=,F20.7,///, 25H RATIO OF SPECIFIC HEATS=,F20.7,///, 34H FREESTREAM 0002410
1 2 REDUCED(DIMENSIONLESS) FREQUENCY=,F20.7,///, 27H DISTANCE BETWEEN 0002430
3 AIRFOILS=,F20.7,///, 52H HORIZONTAL POSITION(DIMENSIONLESS) OF ELAS 0002440
4 IC AXIS=,F20.7,///, 26H COMPATIBLE STAGGER ANGLE=,F20.7,///, 9H DEL 0002450
6E GYRATION=,F20.7,///, 26H DISTANCE(DIMENSIONLESS) FROM EA TO CG=,F20.7,///, 25H U 0002460
720.7,///, 37H BENDING/TORSONAL NATURAL FREQUENCY=,F20.7,///, 25H UP 0002480
5 PER AIRFOIL PHASE LAG=,F20.7)
WRITE(6,6)
6 FORMAT(1H1,10X,8OHVALUES OF FREQUENCY RATIO FOR VARIOUS NON-DIM 0002510
1 TIONAL FREQUENCIES OF OSCILLATION)
RETURN
END

SUBROUTINE INITIAL
INIALIALIZES ALL FLOW FIELD QUANTITIES.

DIMENSION U22R(400,3),U22I(400,3),V22R(400,3),V22I(400,3),
1C22R(400,3),C22I(400,3),X(400,2),Y(400,2)
DIMENSION U33R(200,2),U33I(200,2),V33R(200,2),V33I(200,2),C33R(2000002590
1,2),C33I(200,2)
COMMON/BLK1/NGRDFN,FSTRMN,RTOSPH,REDFRQ,XSUBO,TNWSTD,FMANGL,XLNGTH0002610
1, DELT,STGANG,NSTPTS,DSTSTR,HDSTRL,TNGLH,U22R,U22I,V22R,V22I,C22R,C22I,0002620
2X,Y,S,DELT,ISWIT,JSWIT,ILINE,JLINE,IFAVEP,IATNWL,AL,B1,L1,L2,MA0002630
3,M2,K12R,K12I,K34R,K34I,K56R,K56I,VRPANL,VIPLAN
4,STGANG,NSTPTS,ICOUNT,JCOUNT,IJUNC,ISPLIT
5,LCOUNT,MCOUNT,ISWIT,JSWIT,COMMON/BLK5/U33R,U33I,V33R,V33I,C33R,C33I
LOGICAL FCONST
COMMON/FOFX/FCONST
COMMON/JUNC/IJUNC
COMMON/IFINE/IFIN,JFIN

00002280
00002290
00002300
00002310
00002320
00002330
00002340
00002350
00002360
00002370
00002380
00002390
00002400
00002410
00002430
00002440
00002450
00002460
00002480
00002490
00002500
00002510
00002520
00002530
00002540
00002550
00002560
00002570
00002580
00002590
00002600
00002610
00002620
00002630
00002640
00002650
00002660
00002670
00002680
00002690
00002700
00002701
REAL K12R,K12I,K34R,K34I,K56R,K56I
REAL L1,L2,L3,L4,M1,M2,M3,M4
IJUNC=IJUNT
AI=.5*REDFRQ*DISTR
BI=.25*REDFRQ*(FSTRMN*FSTRMN/(FSTRMN*FSTRMN-1.0))*DISTR

SET UP INITIAL VALUES FOR U, V, AND C AT (0, 0).

IF(.NOT. FCONST) GO TO 50
VRPANL=0.0
VIPANL=-REDFRQ/S
GO TO 60
50 VRPANL=1.0/S
VIPANL=REDFRQ*XSUBO/S
60 U22R(1,1)=-VRPANL
U22I(1,1)=-VIPANL
V22R(1,1)=-U22R(1,1)
V22I(1,1)=-U22I(1,1)
C22R(1,1)=-U22R(1,1)
C22I(1,1)=-U22I(1,1)
X(1,1)=0.0
Y(1,1)=0.0
IF(SPLIT.NE.-1).AND.(DELT.A.LE.1E-05) GO TO 63
L1=-C22R(1,1)
L2=-C22I(1,1)
M1=XSUBO*C22R(1,1)
M2=XSUBO*C22I(1,1)
GO TO 64
63 L1=-2.0*C22R(1,1)
L2=-2.0*C22I(1,1)
M1=2.0*XSUBO*C22R(1,1)
M2=2.0*XSUBO*C22I(1,1)
64 IF(DELT.A.LE.1E-05) GO TO 65
IF(.NOT. FCONST) GO TO 150
VRPANL=REDFRQ*SIN(DELT.A)/S
VIPANL=-REDFRQ*COS(DELT.A)/S
GO TO 160
150 VRPANL=(COS(DELT.A)+REDFRQ*SIN(DELT.A)*XSUBO)/S
VIPANL=(-SIN(DELT.A)-REDFRQ*COS(DELT.A)*XSUBO)/S
160 U33R(1,1)=VRPANL
U33I(1,1)=VIPANL
V33R(1,1)=U33R(1,1)
V33I(1,1)=U33I(1,1)
C33R(1,1)=-U33R(1,1)
C33I(1,1)=-U33I(1,1)
IF(SPLIT.EQ.-1) GO TO 65
L1=L1+C33R(1,1)*COS(DELT.A)+C33I(1,1)*SIN(DELT.A)
L2=L2+C33I(1,1)*COS(DELT.A)-C33R(1,1)*SIN(DELT.A)
M1 = M1 - XSUBO * (C33R(1,1) * COS(Delta) + C33I(1,1) * SIN(Delta))
M2 = M2 - XSUBO * (C33I(1,1) * COS(Delta) - C33R(1,1) * SIN(Delta))

ISWICH = 2
JSWICH = 1
ISWICH = 1
JSWICH = 1
IFIN = 0
JFIN = 0
ILINE = 2
JLINE = 1
IHAVEP = 0
IATNWL = 0
JCOUNT = 0
LCOUNT = 0
MCOUNT = 1
RETURN
END

SUBROUTINE MACHLN
MACHLN COMPUTES THE VALUES OF U, V, AND C ALONG THE INITIAL MACH
LINE AT THE GIVEN X VALUE OF THE MACH LINE.

DIMENSION U22R(400,3), U22I(400,3), V22R(400,3), V22I(400,3),
1 C22R(400,3), C22I(400,3), X(400,2), Y(400,2)
DIMENSION U33R(200,2), U33I(200,2), V33R(200,2), V33I(200,2), C33R(200,2), C33I(200,2),
1,2,33
COMMON/BLK1/NGRDN, FSTRMN, RTOSPHER, REDFRQ, XSUBO, TNWST, FMANGL, XLNGTH
1, DELTAS, NGDPTS, DSTRTR, HDSTRTR, TRNGNH, U22R, U22I, V22R, V22I, C22R, C22I
2, X, Y, S, DELTA, ISWICH, JSWICH, ILINE, JLINE, IHAVEP, IATNWL, AI, BI, L1, L2, M
4, STGANG, NSTPTS, ICOUNT, JCOUNT, IJUNC, ISPLIT
5, LCOUNT, MCOUNT, ISWICH, JSWICH
COMMON/BLK5/U33R, U33I, V33R, V33I, C33R, C33I
LOGICAL FCONST
COMMON/FOFX/FCNST
REAL K12R, K12I, K34R, K34I, K56R, K56I
REAL L1, L2, L3, L4, M1, M2, M3, M4
ICOUNT = JLINE
IF(ICOUNT.LT.IJUNC) GO TO 10
JCOUNT = JCOUNT + 1
KCOUNT = JCOUNT - 1
IF(KCOUNT .NE. 0) GO TO 5
UP = 0.0
UI = 0.0
VR = 0.0
VI = 0.0
CR=0.0
CI=0.0
GO TO 7
5 UR=U22R(KCOUNT,3)
   UI=U22I(KCOUNT,3)
   VR=V22R(KCOUNT,3)
   VI=V22I(KCOUNT,3)
   CR=C22R(KCOUNT,3)
   CI=C22I(KCOUNT,3)
   K12R=AI*UR+CI
   K34R=U22R(1,JSWTCH)-V22R(1,JSWTCH)+BI*(U22I(1,JSWTCH)-C22I(1,JSWTC)
1H)
   K34I=U22I(1,JSWTCH)-V22I(1,JSWTCH)-BI*(U22R(1,JSWTCH)-C22R(1,JSWTC)
1H)
   K56R=U22R(JCOUNT,3)+V22R(JCOUNT,3)+BI*(U22I(JCOUNT,3)-C22I(JCOUNT)
13))
   K56I=U22I(JCOUNT,3)+V22I(JCOUNT,3)-BI*(U22R(JCOUNT,3)-C22R(JCOUNT)
13))
   G1=5*(K34R+K56R)
   G2=BI*KL2I
   G3=1.0-AI*B1
   G4=5*(K34I+K56I)
   G5=BI*KL2R
   G6=2.0*B1
   G7=G3*G3+G6*G6
   G8=G1-G2
   G9=G4+G5
   U22R(1,ISWTCH)=(G8*G3+G9*G6)/G7
   U22I(1,ISWTCH)=-(G8*G6+G9*G3)/G7
   V22R(1,ISWTCH)=-5*(K56R-K34R)
   V22I(1,ISWTCH)=-5*(K56I-K34I)
   C22R(1,ISWTCH)=K12R-U22R(1,ISWTCH)
   C22I(1,ISWTCH)=K12I-U22I(1,ISWTCH)
1I
GO TO 503
10 W=FSTRMN*FSTRMN
   T=FSTRMN*FSTRMN-1.0
   IF(.NOT. FCONST) GO TO 20
   VRPANL=0.0
   VIPANL=REDFRQ/S
   GO TO 30
20 VRPANL=-1.0/S
   VIPANL=REDFRQ*XSUBO/S
30 U=COS(REDFRQ*(W/T)*X(1,ISWTCH))
   V= SIN(REDFRQ*(W/T)*X(1,ISWTCH))
C
C HIERARCHY W,T,S,U,V

U22R(1,ISWTCH)=-VRPANL*U-VIPANL*V
U22I(1,ISWTCH)=-VIPANL*U+VRPANL*V
V22R(1,ISWTCH)=-U22R(1,ISWTCH)
V22I(1,ISWTCH)=-U22I(1,ISWTCH)
C22R(1,ISWTCH)=-U22R(1,ISWTCH)
C22I(1,ISWTCH)=-U22I(1,ISWTCH)
IF((ICOUNT.NE.ISPLIT).OR.(DELTA.GT.1E-05)) GO TO 500
U22R(1,3)=-U22R(1,ISWTCH)
U22I(1,3)=-U22I(1,ISWTCH)
V22R(1,3) = V22R(1,ISWTCH)
V22I(1,3) = V22I(1,ISWTCH)
C22R(1,3)=-C22R(1,ISWTCH)
C22I(1,3)=-C22I(1,ISWTCH)
GO TO 503
500 IF((DELTA.LE.1E-05).OR.(JLINE.GT.ISPLIT)) GO TO 503
IF(.NOT.FCOST) GO TO 120
VRPANL=REDFRQ*SIN(DELTA)/S
VIPANL=REDFRQ*COS(DELTA)/S
GO TO 130
120 VRPANL=-(COS(DELTA)*REDFRQ*SIN(DELTA)*XSUBO)/S
VIPANL=-(SIN(DELTA)*REDFRQ*COS(DELTA)*XSUBO)/S
130 U33R(1,ISWTCH)=VRPANL*U+VIPANL*V
U33I(1,ISWTCH)=VIPANL*U-VRPANL*V
V33R(1,ISWTCH)=U33R(1,ISWTCH)
V33I(1,ISWTCH)=U33I(1,ISWTCH)
C33R(1,ISWTCH)=-U33R(1,ISWTCH)
C33I(1,ISWTCH)=U33I(1,ISWTCH)
IF((ICOUNT.NE.ISPLIT)) GO TO 503
U22R(1,3)=-U33R(1,ISWTCH)
U22I(1,3)=-U33I(1,ISWTCH)
V22R(1,3)=V33R(1,ISWTCH)
V22I(1,3)=V33I(1,ISWTCH)
C22R(1,3)=C33R(1,ISWTCH)
C22I(1,3)=C33I(1,ISWTCH)
GO TO 503
503 CONTINUE
RETURN
END

SUBROUTINE HIFOIL

HIFOIL COMPUTES U,V,AND C AT AN UPPER AIRFOIL POINT.

C DIMENSION U22R(400,3),U22I(400,3),V22R(400,3),V22I(400,3),
1 C22R(400,3),C22I(400,3),X(400,2),Y(400,2)
COMMON/BLK1,NGRDFN,FSTRMN,RTOSPH,REDFRQ,XSUBO,TNWST,FMNGL,XLNGTH/00004480
C22R(1, ISWITCH)=K12R-U22R(1, ISWITCH)+U22I(1, ISWITCH)*AI 00004960
C22I(1, ISWITCH)=K12I-U22I(1, ISWITCH)+U22I(1, ISWITCH)*AI 00004970
CPR1=C22R(1, ISWITCH)*COS(DELTA)+C22I(1, ISWITCH)*SIN(DELTA) 00004980
CPI1=C22I(1, ISWITCH)*COS(DELTA)-C22R(1, ISWITCH)*SIN(DELTA) 00004990
CPR2=C22R(1, ISWITCH)*COS(DELTA)-C22I(1, ISWITCH)*SIN(DELTA) 00005000
CPI2=C22I(1, ISWITCH)*COS(DELTA)+C22R(1, ISWITCH)*SIN(DELTA) 00005010
IF(JFIN.EQ.1) GO TO 130 00005020
AT START OF ZONE AND CLOSE TO END? SKIP LIFT CALCULATION
IF(LCOUNT.EQ.0).AND.(TOPCRD+DSTSTR.GT.1.0)) GO TO 130 00005030
AT START OF ZONE OR END OF AIRFOIL? START/STOP INTEGRATION
IF(LCOUNT.EQ.0).OR.(TOPCRD+DSTSTR.GT.1.0)) GO TO 95 00005040
L1=L1+2.0*CPR2 00005050
L2=L2+2.0*CPI2 00005060
M1=M1+2.0*CPR2*(TOPCRD-XSUBO) 00005070
M2=M2+2.0*CPI2*(TOPCRD-XSUBO) 00005080
GO TO 100 00005090
95 L1=L1+CPR2 00005100
L2=L2+CPI2 00005110
M1=M1+CPR2*(TOPCRD-XSUBO) 00005120
M2=M2+CPI2*(TOPCRD-XSUBO) 00005130
100 LCOUNT=LCOUNT+1 00005140
SET UP FOR END OF ZONE CHECK.
IF(ISWICH.NE.1) GO TO 105 00005150
IF(LCOUNT.EQ.IJUNT) GO TO 110 00005160
GO TO 120 00005170
105 IF(LCOUNT.NE.LJCOUNT) GO TO 120 00005180
AT END OF ZONE, LINEARLY EXTRAPOLATE LAST VALUE AND STOP INTEGRAL 00005190
110 L1=L1+2.0*CPR2-CPR1 00005200
L2=L2+2.0*CPI2-CPI1 00005210
M1=M1+(2.0*CPR2-CPR1)*(TOPCRD+DSTSTR-XSUBO) 00005220
M2=M2+(2.0*CPI2-CPI1)*(TOPCRD+DSTSTR-XSUBO) 00005230
ZERO COUNTER AND SWITCH ZONE MARKERS.
LCOUNT=0 00005240
IF(ISWICH.EQ.1) GO TO 115 00005250
ISWICH=1 00005260
GO TO 120 00005270
115 ISWICH=2 00005280
ALPHA=90-BETA? ELIMINATE ODD ZONE.

IF(JCOUNT.EQ.0) ISWICH=1
CONTINUE
IF(TOPCRD+DSTSTR.LE.1.0) GO TO 130

COMPUTE VALUES OF C22 AT TOPCRD=1.

DECLR=C22R(1,1,ISWTCH)+(C22R(1,1,ISWTCH)-C22R(1,1,JSWTCH))* (1.0-X(1,ISWTCH))
DECLI=C22I(1,1,ISWTCH)+(C22I(1,1,ISWTCH)-C22I(1,1,JSWTCH))* (1.0-X(1,ISWTCH))

DECLPR=DECLR*COS(Delta)+DECLI* SIN(Delta)
DECLPI=DECLI*COS(Delta)-DECLR* SIN(Delta)

COMPUTE CHANGE IN LIFT AND MOMENT DUE TO UNDERSHOOT.

DELL1=(CP12+DECLPR)*(1.0-TOPCRD)/DSTSTR
DELL2=(CP12+DECLPR)*(1.0-TOPCRD)/DSTSTR
DELM1=(CP12*(TOPCRD-XSUBO)+DECLPR*(1.0-XSUBO))*(1.0-TOPCRD)/DSTSTR
DELM2=(CP12*(TOPCRD-XSUBO)+DECLPI*(1.0-XSUBO))*(1.0-TOPCRD)/DSTSTR

L1=L1+DELL1
L2=L2+DELL2
M1=M1+DELM1
M2=M2+DELM2
JFIN=1
CONTINUE
ICOUNT=0
RETURN
END

SUBROUTINE GENFPT

GENFPT COMPUTES U,V, AND C AT A GENERAL FIELD POINT.

 DIMENSION U22R(400,3),U22I(400,3),V22R(400,3),V22I(400,3),
 C22R(400,3),C22I(400,3),X(400,2),Y(400,2)
 DIMENSION U33R(200,2),U33I(200,2),V33R(200,2),V33I(200,2),C33R(200)
 1,2),C33I(200,2)
 COMMON/BLK1/KSDFN,FSTRMN,RTOSPH,REDFRQ,XSUBO,TNWDST,FMANGL,XLNGTH
 1/ETLAS,NGOPTS,DSTSTR,HDSTRLR,TRNGLH,U22R,U22I,V22R,V22I,C22R,C22I
 2,X,Y,S,DELTA,ISWTCH,JSWTCH,ILINE,JLINE,IHAVEP,IAWNL,AL,BL,L1,L2
 3,M2,K12R,K12I,K34R,K34I,K56R,K56I,VRPANL,VRPANL
 4,STGANG,NSTPTS,ICOUNT,JCOUNT,IJUNC,ISPLIT
 5,LCOUNT,MCOUNT,ISWICH,JSWICH
 COMMON/BLK5/U33R,U33I,V33R,V33I,C33R,C33I
 REAL K12R,K12I,K34R,K34I,K56R,K56I

00005320
00005330
00005340
00005350
00005360
00005380
00005390
00005400
00005410
00005420
00005430
00005440
00005450
00005460
00005470
00005480
00005490
00005500
00005510
00005520
00005530
00005540
00005550
00005560

REAL L1,L2,L3,L4,M1,M2,M3,M4
I=IHAVEP+1
IF(IATNW.L.EQ. 0) GO TO 10
K12R=U22R(I,J,JSWTCM)+C22R(I,J,JSWTCM)+AI*U22I(I,J,JSWTCM)
K12I=-AI*U22R(I,J,JSWTCM)+U22I(I,J,JSWTCM)+C22I(I,J,JSWTCM)
K34R=U22R(I+1,J,JSWTCM)+V22R(I+1,J,JSWTCM)+BI*(U22I(I+1,J,JSWTCM)-C22I(I+1,J,JSWTCM))
K34I=U22I(I+1,J,JSWTCM)-V22I(I+1,J,JSWTCM)-BI*(U22R(I+1,J,JSWTCM)-C22R(I+1,J,JSWTCM))
GO TO 12
K34R=U22R(IHAVEP+1,J,JSWTCM)+V22R(IHAVEP+1,J,JSWTCM)+BI*(I22I(IHAVEP+1,J,JSWTCM))
K34I=U22I(IHAVEP+1,J,JSWTCM)-V22I(IHAVEP+1,J,JSWTCM)-BI*(U22R(IHAVEP+1,J,JSWTCM))
12 K56R=U22R(IHAVEP,JSWTCM)+V22R(IHAVEP,JSWTCM)+BI*(I22I(IHAVEP,JSWTCM))
K56I=U22I(IHAVEP,JSWTCM)+V22I(IHAVEP,JSWTCM)-BI*(U22R(IHAVEP,JSWTCM))
G1=-5*(K34R*K56R)
G2=BI*K12I
G3=1.0-AI*BI
G4=5*(K34I*K56I)
G5=BI*K12R
G6=1.0*BI
G7=G3*G5*G6
G8=G1-G2
G9=G4+G5
U22R(I,JSWTCM)=(G8*G3*G9*G6)/G7
U22I(I,JSWTCM)=(-G8*G6*G9*G3)/G7
V22R(I,JSWTCM)=5*(K56R-K34R)
V22I(I,JSWTCM)=5*(K56I-K34I)
C22R(I,JSWTCM)=K12R-U22R(I,JSWTCM)
C22I(I,JSWTCM)=K12I-U22I(I,JSWTCM)
IF((ICOUNT.NE.ISPLIT).OR.(DELTA.GT.1E-05)) GO TO 450
U22R(I,3)=-U22R(I,JSWTCM)
U22I(I,3)=-U22I(I,JSWTCM)
V22R(I,3)=V22R(I,JSWTCM)
V22I(I,3)=V22I(I,JSWTCM)
C22R(I,3)=-C22R(I,JSWTCM)
C22I(I,3)=-C22I(I,JSWTCM)
GO TO 500
450 IF((DELTA.LE.1E-05).OR.(JLINE.GT.ISPLIT)) GO TO 500
K12R=U33R(I HAVEP, JSWTCH)+C33R(I HAVEP, JSWTCH)+AI*U33I(I HAVEP, JSWTCH)
1)
K12I=-AI*U33R(I HAVEP, JSWTCH)+U33I(I HAVEP, JSWTCH)+C33I(I HAVEP, JSWTCH)
1H
K34R=U33R(I HAVEP, JSWTCH)-V33R(I HAVEP, JSWTCH)+BI*(U33I(I HAVEP, JSWTCH)
1H)-C33I(I HAVEP, JSWTCH))
K34I=U33I(I HAVEP, JSWTCH)-V33I(I HAVEP, JSWTCH)-BI*(U33R(I HAVEP, JSWTCH)
1H)-C33R(I HAVEP, JSWTCH))
K56R=U33R(I HAVEP+1, JSWTCH)+V33R(I HAVEP+1, JSWTCH)+BI*(U33I(I HAVEP+1)
1, JSWTCH)-C33I(I HAVEP+1, JSWTCH))
K56I=U33I(I HAVEP+1, JSWTCH)+V33I(I HAVEP+1, JSWTCH)-BI*(U33R(I HAVEP+1)
1, JSWTCH)-C33R(I HAVEP+1, JSWTCH))
G1=5*(K34R+K56R)
G2=BI*K12I
G4=5*(K34I+K56I)
G5=BI*K12R
G8=G1-G2
G9=G4+G5
U33R(I, JSWTCH)=(G8*G3+G9*G6)/G7
U33I(I, JSWTCH)=(-G8*G6+G9*G3)/G7
V33R(I, JSWTCH)=0.5*(K56R-K34R)
V33I(I, JSWTCH)=0.5*(K56I-K34I)
C33R(I, JSWTCH)=K12R-U33R(I, JSWTCH)+U33I(I, JSWTCH)*AI
C33I(I, JSWTCH)=K12I-U33I(I, JSWTCH)-U33R(I, JSWTCH)*AI
IF(JLINE.NE.ISPLIT) GO TO 500
U22R(I, 3)=U33R(I, JSWTCH)
U22I(I, 3)=U33I(I, JSWTCH)
V22R(I, 3)=V33R(I, JSWTCH)
V22I(I, 3)=V33I(I, JSWTCH)
C22R(I, 3)=C33R(I, JSWTCH)
C22I(I, 3)=C33I(I, JSWTCH)
CONTINUE
RETURN
END

SUBROUTINE LOFOIL

LOFOIL COMPUTES THE VALUES OF U, V, AND C AT A LOWER AIRFOIL POINT.

DIMENSION U22R(400, 3), U22I(400, 3), V22R(400, 3), V22I(400, 3),
1C22R(400, 3), C22I(400, 3), X(400, 2), Y(400, 2)
1DIMENSION U33R(200, 2), U33I(200, 2), V33R(200, 2), V33I(200, 2), C33R(20000, 2000, 2),
1C33I(20000, 2)
COMMON/BLK1/NGRDFN, ESTRMN, RTOEIR, REDTRO, XSUBO, TNWDSL, FMANGL, XNLNGTH0000000000
1DELTA, NGDPTS, DSTSTR, HDSSTR, TENGHL, U22R, U22I, V22R, V22I, C22R, C22I, 0000000000
1X, Y, S, DELTA, JSWTCH, JSWTCH, ILINE, JLINE, IHAVEP, IATNWL, AI, BI, L1, L2, M1000000000
4, STG, NSTPS, ICOUNT, NCOUNT, IJUNC, ISPLIT
5, ICOUNT, MCOUNT, ISWICH, JSWICH
COMMON/BLK5/U33R, U33I, V33R, V33I, C33R, C33I
COMMON/PCOR/ICOUNT
COMMON/IFINE/IFIN, JFIN
LOGICAL FCONST
COMMON/FOFX/FCNST
REAL K12R, K12I, K34R, K34I, K56R, K56I
REAL L1, L2, L3, L4, M1, M2, M3, M4
IF(.NOT FCNST) GO TO 999
V22R(ILINE, ISWICH) = 0.0
V22I(ILINE, ISWICH) = -REDFRQ/SQRT(FSTRMN*FSTRMN-1.0)
GO TO 90
999 V22R(ILINE, ISWICH) = -1.0/SQRT(FSTRMN*FSTRMN-1.0)
V22I(ILINE, ISWICH) = (-REDFRQ*X(IHAVEP+1, ISWICH)-XSUBO))/SQRT(FSTRM00006750
0=FSTRMN-1.0)
90 K34R=V22R(ILINE, ISWICH)
K34I=V22I(ILINE, ISWICH)
K12R=U22R(JLINE, JSWICH)+C22R(JLINE, JSWICH)+AI*U22I(JLINE, JSWICH)
K12I=AI*U22R(JLINE, JSWICH)+U22I(JLINE, JSWICH)+C22I(JLINE, JSWICH)
IF(IATNLW.EQ.0) GO TO 91
K56R=U22R(KOUNT, ISWICH)+V22R(KOUNT, ISWICH)+BI*(U22I(KOUNT, ISWICH))
K56I=U22I(KOUNT, ISWICH)
1-C22I(KOUNT, ISWICH))
1-C22R(KOUNT, ISWICH))
GO TO 92
91 K56R=U22R(JLINE, ISWICH)+V22R(JLINE, ISWICH)+BI*(U22I(JLINE, ISWICH))
1-C22I(JLINE, ISWICH))
1-C22R(JLINE, ISWICH))+V22I(JLINE, ISWICH)-BI*(U22R(JLINE, ISWICH))
92 G1=1.0-AI*BI
G2=2.0*BI
G3=G1*G1+G2*G2
G4=K56R-K34R-BI*K12I
G5=K56I-K34I+BI*K12R
U22R(ILINE, ISWICH) = (G4*G1+G5*G2)/G3
U22I(ILINE, ISWICH) = (-G4*G2+G5*G1)/G3
C22R(ILINE, ISWICH) = K12R-U22R(ILINE, ISWICH)+U22I(ILINE, ISWICH)*AI
C22I(ILINE, ISWICH) = K12I-U22I(ILINE, ISWICH)-U22R(ILINE, ISWICH)*AI
K=ILINE
L=ISWICH
IF((ICOUNT .NE. ISPLIT).OR.(DELTA<.GT.1E-05)) GO TO 80
U22R(K,3) = U22R(K,3) - U22R(K,3) = V22R(K,3) - V22I(K,3) - V22I(K,3) = C22R(K,3)
C22I(K,3) = C22I(K,3) - C22I(K,3) = 0.00066620
0.00066300
0.00066400
0.00066500
0.00066510
0.00066600
0.00066700
0.00066800
0.00066900
0.00067000
0.00067100
0.00067200
0.00067300
0.00067400
0.00067500
0.00067600
0.00067700
0.00067800
0.00067900
0.00068000
0.00068100
0.00068200
0.00068300
0.00068400
0.00068500
0.00068600
0.00068700
0.00068800
0.00068900
0.00069000
0.00069100
0.00069200
0.00069300
0.00069400
0.00069500
0.00069600
0.00069700
0.00069800
0.00069900
0.00070000
0.00070100
0.00070200
0.00070300
0.00070400
0.00070500
0.00070600
0.00070700
0.00070800
GO TO 94
80 IF((DELTA.LT.1.E-05).OR.(JLINE.GT.ISPLIT)) GO TO 94
IF(.NOT.FCNST) GO TO 84
V33R(K,L)=REDFRQ*IN(SIN(DELTA))/S
V33I(K,L)=REDFRQ*COS(DELTA)/S
GO TO 86
84 V33R(K,L)=-(COS(DELTA)-REDFRQ*SIN(DELTA)*)((X(K,L)-XSUBO))/S
V33I(K,L)=-(SIN(DELTA)-REDFRQ*COS(DELTA)*)((X(K,L)-XSUBO))/S
86 K12R=US3R(JLINE,JSWITCH)+C33R(JLINE,JSWITCH)+AI*U33I(JLINE,JSWITCH)
K12I=U33I(JLINE,JSWITCH)+C33I(JLINE,JSWITCH)-AI*U33R(JLINE,JSWITCH)
K34R=U33R(JLINE,JSWITCH)+C33R(JLINE,JSWITCH)-BI*(U33I(JLINE,JSWITCH)-C33I(JLINE,JSWITCH))
K34I=U33I(JLINE,JSWITCH)-U33R(JLINE,JSWITCH)-BI*(U33R(JLINE,JSWITCH)-C33R(JLINE,JSWITCH))
K56R=V33R(K,L)
K56I=V33I(K,L)
G4=K56R+K34R-BI*K12I
G5=K56I+K34I+BI*K12R
U33R(K,L)=(G4*G1+G5*G2)/G3
U33I(K,L)=(G4*G2-G5*G1)/G3
C33R(K,L)=K12R-U33R(K,L)+U33I(K,L)*AI
C33I(K,L)=K12I-U33I(K,L)-U33R(K,L)*AI
IF(JLINE.NE.ISPLIT) GO TO 88
U22R(K,3)=U33R(K,L)
U22I(K,3)=U33I(K,L)
V22R(K,3)=V33R(K,L)
V22I(K,3)=V33I(K,L)
C22R(K,3)=C33R(K,L)
C22I(K,3)=C33I(K,L)
C22I(K,3)=C33I(K,L)
88 CPR2=C33R(K,L)*COS(DELTA)+C33I(K,L)*SIN(DELTA)
CP12=C33I(K,L)*COS(DELTA)-C33R(K,L)*SIN(DELTA)
CP11=C33I(JLINE,JSWITCH)*COS(DELTA)+C33I(JLINE,JSWITCH)*SIN(DELTA)
CP11=C33I(JLINE,JSWITCH)*COS(DELTA)-C33R(JLINE,JSWITCH)*SIN(DELTA)
IF(IFIN.EQ.1) GO TO 140
L1=L1+2.0*CPR2
L2=L2+2.0*CP12
M1=M1+2.0*CPR2*(X(K,L)-XSUBO)
M2=M2+2.0*CP12*(X(K,L)-XSUBO)
IF((JLINE.NE.ISPLIT).AND.(X(K,L)+DSTSTR.LE.1.0)) GO TO 94
L1=L1+2.0*CPR2-CPR1
L2=L2+2.0*CP12-CP11
M1=M1+2.0*CPR2-CPR1)*(X(K,L)+DSTSTR-XSUBO)
M2=M2+2.0*CP12-CP11)*(X(K,L)+DSTSTR-XSUBO)
94 IREF=0
IF(IFIN.EQ.1) GO TO 140
C
AT START OF ZONE AND CLOSE TO END? SKIP LIFT CALCULATION.
C
IF((MCOUNT.EQ.0).AND.(X(JLINE,JSWITCH)+DSTSTR.GT.1.0)) GO TO 140
C
00007090
00007100
00007110
00007120
00007130
00007140
00007150
00007160
00007170
00007180
00007190
00007200
00007210
00007220
00007230
00007240
00007250
00007260
00007270
00007280
00007290
00007300
00007310
00007320
00007330
00007340
00007350
00007360
00007370
00007380
00007390
00007400
00007410
00007420
00007430
00007440
00007450
00007460
00007470
00007480
00007490
00007500
00007510
AT START OF ZONE OR END OF AIRFOIL? START/STOP INTEGRATION.

IF(MCOUNT.EQ.0).OR.(X(ILINE,ISWATCH)+DSTSTR.GT.1.0)) GO TO 95

93 L1=1+2.0*C22R(ILINE,ISWATCH)
L2=1+2.0*C22I(ILINE,ISWATCH)
M1=1+2.0*C22R(ILINE,ISWATCH)*(X(ILINE,ISWATCH)-XSUBO)
M2=1+2.0*C22I(ILINE,ISWATCH)*(X(ILINE,ISWATCH)-XSUBO)
IF(IREF.EQ.1) GO TO 130
GO TO 100

95 L1=1+2.0*C22R(ILINE,ISWATCH)
L2=1+2.0*C22I(ILINE,ISWATCH)
M1=1+2.0*C22R(ILINE,ISWATCH)*(X(ILINE,ISWATCH)-XSUBO)
M2=1+2.0*C22I(ILINE,ISWATCH)*(X(ILINE,ISWATCH)-XSUBO)
IF(IREF.EQ.1) GO TO 130
IF(X(ILINE,ISWATCH)+DSTSTR.GT.1.0) GO TO 120

100 MCOUNT=MCOUNT+1

SET UP FOR END OF ZONE CHECK.

IF(JSWICH.NE.1) GO TO 105
IF(MCOUNT.EQ.IJUNC) GO TO 110
GO TO 120

105 IF(MCOUNT.NE.(ISPLIT+1)) GO TO 120

AT END OF ZONE, LINEARLY EXTRAPOLATE LAST VALUE AND STOP INTEGRATION.

110 L1=1+2.0*C22R(ILINE,ISWATCH)+C22R(JLINE,JSWATCH)
L2=1+2.0*C22I(ILINE,ISWATCH)+C22I(JLINE,JSWATCH)
M1=1+2.0*C22R(ILINE,ISWATCH)-C22R(JLINE,JSWATCH)*(X(ILINE,ISWATCH)
M2=1+2.0*C22I(ILINE,ISWATCH)-C22I(JLINE,JSWATCH)*(X(ILINE,ISWATCH)
1+DSTSTR-XSUBO)
M2=M2-(2.0*C22I(ILINE,ISWATCH)-C22I(JLINE,JSWATCH))*(X(ILINE,ISWATCH)
1+DSTSTR-XSUBO)

IN ZONE 1? GET LIFT & MOMENT OF TOP AIRFOIL.

IF(IREF.EQ.1) GO TO 93

ZERO COUNTER AND SWITCH ZONE MARKERS.

MCOUNT=0
IF(JSWICH.EQ.1) GO TO 115

115 JSWICH=1
GO TO 120

ALPHA=90-BETA? ELIMINATE ODD ZONE.

IF(ISPLIT.EQ.-1) JSWICH=1

00007520
00007530
00007540
00007550
00007560
00007570
00007580
00007590
00007600
00007610
00007620
00007630
00007640
00007650
00007660
00007670
00007680
00007690
00007700
00007710
00007720
00007730
00007740
00007750
00007760
00007770
00007780
00007790
00007800
00007810
00007820
00007830
00007840
00007850
00007860
00007870
00007880
NOT IN ZONE 1? SKIP IT.

120 IF((ISPLIT.EQ.-1).OR.(JCOUNT.NE.0).OR.(DELTA.GT.1E-05)) GO TO 130
   IREF=1
   IF(NSTPTS.NE.1) GO TO 125

AT END OF AIRFOIL? ADD FOR TOP AIRFOIL.

   IF(X(ILINE,ISWTCH)+DSTR.GT.1.0) GO TO 95

AT END OF ZONE 1? ADD FOR TOP AIRFOIL.

   IF(MCOUNT.EQ.0) GO TO 110
   GO TO 93
   125 IF(X(ILINE,ISWTCH)+DSTR.GT.1.0) GO TO 128
   IF(MCOUNT.EQ.0) GO TO 130
   IF(MCOUNT-ISPLIT-2)93,95,130
   128 IF(MCOUNT-ISPLIT-1) 95,95,130
   130 CONTINUE
   IF(X(ILINE,ISWTCH)+DSTR.LE.1.0) GO TO 140

COMPUTE VALUES OF C22 AT X = 1.

   DELCR=C22R(ILINE,ISWTCH)+C22R(ILINE,ISWTCH)-C22R(JLINE,JSWTCH))*00008050
   11.0-X(ILINE,ISWTCH))/DSTR
   DELCI=C22I(ILINE,ISWTCH)+C22I(ILINE,ISWTCH)-C22I(JLINE,JSWTCH))*00008070
   11.0-X(ILINE,ISWTCH))/DSTR

COMPUTE CHANGE IN LIFT AND MOMENT DUE TO UNDERSHOOT.

   DELL1=(C22R(ILINE,ISWTCH)+DELCR)*(1.0-X(ILINE,ISWTCH))/DSTR
   DELL2=(C22R(ILINE,ISWTCH)+DELCI)*(1.0-X(ILINE,ISWTCH))/DSTR
   DELM1=(C22R(ILINE,ISWTCH)*(X(ILINE,ISWTCH)-XSUBO)+DELCR*(1.0-XSUBO)00008120
   1.0-X(ILINE,ISWTCH))/DSTR
   DELM2=(C22I(ILINE,ISWTCH)*(X(ILINE,ISWTCH)-XSUBO)+DELCI*(1.0-XSUBO)00008140
   1.0-X(ILINE,ISWTCH))/DSTR

IF NEEDED ADD FOR TOP AIRFOIL NOT COMPUTED IN HIFOIL.

   IF(DELTA.GT.1E-05) GO TO 135
   IF(IREF.EQ.0) GO TO 135
   IF(MCOUNT.GT.(ISPLIT+1)) GO TO 135
   L1=L1-DELL1
   L2=L2-DELL2
   M1=M1-DELM1
   M2=M2-DELM2
   135 L1=L1-DELL1
L2=L2-DELM2
M1=M1-DELM1
M2=M2-DELM2
IFIN=1
140 CONTINUE
RETURN
END

SUBROUTINE COMPXY
DIMENSION U22R(400,3),U22I(400,3),V22R(400,3),V22I(400,3),
C22R(400,3),C22I(400,3),X(400,2),Y(400,2)
COMMON/BLK1/NGRDFN,FRSRMN,RTOSPH,REDFRQ,XSUB0,TNWDST,FMANGL,XLNGTH
1,DELTA,NGDPTS,DSTSTR,HISTR,TRNGLH,U22R,U22I,V22R,V22I,C22R,C22I
2,X,Y,S,DELTAS,NGDPTS,DSTSTR,HISTR,TRNGLH,U22R,U22I,V22R,V22I,C22R,C22I
3,M2,K12R,K12I,K34R,K34I,K56R,K56I,VRPNL,VIPNL
4,STGANG,NSTPNTS,JCOUNT,JCOUNT,JUNC,ISPLIT
5,JCOUNT,NCOUNT,ISWICH,JSWICH
REAL K12R,K12I,K34R,K34I,K56R,K56I
REAL L1,L2,L3,L4,M1,M2,M3,M4

IF FALLS THROUGH, COMPUTE X, Y FOR NEXT ITH POINT ON LINE. (I TH.GT.2)

IF(IHAVEP.EQ.0) GO TO 1
I=IAHAVEP+1
X(1,ISWITCH)=X(IHAVEP,ISWITCH)+HDSTRL
Y(1,ISWITCH)=Y(IHAVEP,ISWITCH)-TRNGLH
GO TO 2

IF FALLS THROUGH ON 1 X, Y IS ON MACH LINE.

1 IF((IATNWL.EQ.1).AND.(JLINE.NE.NGDPTS-1)) GO TO 3
I=IAHAVEP+1
X(1,ISWITCH)=X(1,JSWITCH)+HDSTRL
Y(1,ISWITCH)=Y(1,JSWITCH)+TRNGLH
GO TO 2

IF GONE TO 3 COMPUTE FIRST POINT ON RIGHT RUNNING MACH LINE EMANATING FROM THE TUNNEL WALL.

1 I=IAHAVEP+1
X(1,ISWITCH)=X(1,JSWITCH)+DSTSTR
Y(1,ISWITCH)=Y(1,JSWITCH)
2 CONTINUE
RETURN
END
SUBROUTINE FLUTER

FLUTTER COMPUTES THE FLUTTER SPEED AND FREQUENCY.

DIMENSION U22R(400, 3), U22I(400, 3), V22R(400, 3), V22I(400, 3),
1C22R(400, 3), C22I(400, 3), X(400, 2), Y(400, 2),
COMMON/BLK1/NGRDFN, FSTRMN, RTUSH, REDFREQ, XSUBO, TWDST, FMANGO, XLGHTH
1, DELTAS, NGDPTS, DSTR, HDSTR, TRNJCL, U22R, U22I, V22R, V22I, C22R, C22I, 0.0008670
2X, Y, S, DELTA, ISWITCH, JSWITCH, ILINE, JLINE, IHAPE, IATNW, AI, BI, L1, L2, M1, 0.0008680
3, 6, 7, 8, 9, 10, 11, 12, K12R, K12I, K34R, K34I, K56I, K56I, VRPNL, VJPNL
4, 5, LCOUNT, MCOUNT, ISWITCH, JSWITCH
COMMON/BLK2/L3, L4, M3, M4
COMMON/BLK3/MUu, RSUBA'XSUBA, HAFREQ, OMEGAA, OMEGAH, INCRE
COMMON/BLK4/ROOTXI, ROOTX2, ROOTXI, OMEGA, UF, RFREQ, IMAGRT
REAL K12R, K12I, K34R, K34I, K56R, K56I
REAL L1, L2, L3, L4, M1, M2, M3, M4
REAL MUU, INCRE
IMAGRT=0
RFREQ=2.0/REDFREQ
DR=L1*M3-L3*M1-L2*M4+L4*M2
DI=L1*M4-L4*M1-L2*M3+L3*M2
CR=MUU*(XSUBA*(M1+L3)-M3+OMEGAA-L1*RSUBA**2-MUU*XSUBA**2)+DR
CI=MUU*(XSUBA*(M4+L2)-M4-L2*RSUBA**2)+DI
IF((OMEGAA.LT.1.E-06).OR.(OMEGAA.LT.1.E-06)) GO TO 50
FACTOR=0.5*((L1-MUU)/OMEGAA+(M3-OMEGAA)/OMEGAA)
DISC=FACTOR**2*CR
IF(DISC.LE.-1.0E-06) GO TO 600
XR1=-FACTOR+SQRT(DISC)
XR2=-FACTOR-SQRT(DISC)
GO TO 100
50 XR1=CR/(OMEGAA*(MUU-L1)+OMEGAH*(MUU*RSUBA**2-M3))
XR2=XR1
100 XI=CI/(OMEGAA*L2+OMEGAH*M4)
IF((XR1.LE.-1.0E-06).OR.(XR2.LE.-1.0E-06).OR.(XI.LE.-1.0E-06)) GO TO 600
ROOTXI=SQRT(XR1)
ROOTX2=SQRT(XR2)
ROOTXI=SQRT(XI)
OMEGA=1.0/RDOUTX
UF=OMEGA/REDFREQ
WRITE(6,501)RFREQ
501 FORMAT(/,'5H1/K=,'F20.7)
WRITE(6,502)ROOTXI, ROOTX2, ROOTXI
502 FORMAT(/,'5X,'11H SQRT(XR1)=,'E20.7,5X,'11H SQRT(XR2)=,'E20.7,5X,10H
1SQRT(XI)=,'E20.7')
GO TO 1000
600 IMAGRT=1
WRITE(6,601)RFREQ
601 FORMAT(//,1X,'RFREQ')
602 FORMAT//,1X,'RFREQ')
601  FORMAT('///,5H1/K=,'F20.7,///,32H TOO BAD - SQRT(X) IS IMAGINARY.\) 00009004
1000 CONTINUE
 RETURN
 END
PROGRAM B -- ONE DEGREE OF FREEDOM FLUTTER

THIS PROGRAM COMPUTES THE FLUTTER FREQUENCY AND FLUTTER VELOCITY
OF A CASCADE OSCILLATING AT LOW AMPLITUDES WITH ARBITRARY STAGGER,
PHASE LAG, BLADE SPACING, AND FREQUENCY. THE CASCADE MUST HAVE
SUPersonic LEADING EDGE LOCUS.

DIMENSION U22R(400,3),U22I(400,3),V22R(400,3),V22I(400,3),
1C22R(400,3),C22I(400,3),X(400,2),Y(400,2)
DIMENSION DATE(3)
DIMENSION U33R(200,2),U33I(200,2),V33R(200,2),V33I(200,2),C33R(2000,2)
1,2,C33I(200,2)
COMMON/BLK1/NRGDFN,ESTRMN,RTOSPH,REDFRQ,XSUB0,TNWDST,FMANGL,ELNGTH,
1,DELTA,NGDPTS,DSSTR,HDSTR,TRNGLH,U22R,U22I,V22R,V22I,C22R,C22I,
2X,Y,DELTA,ISWTCH,JSWTCH,LINE,ILINE,ITYAVEP,ITYNWL,AI,BI,M1
3,M2,
1,K12R,K12I,K34R,K34I,K56R,K56I,VRPANL,VIPANL
4,STGANG,NSTPTS,ICOUNT,ICOUNT,IJUNC,ISPLIT
5,LCOUNT,MCOUNT,ISWICH,JSWICH
COMMON/BLK3/MU,MU,RSUBA,GSUBA,INCRE
REAL MUU,INCRE
COMMON/BLK5/U33R,U33I,V33R,V33I,C33R,C33I
COMMON/PCOR/KOUNT
COMMON/JUNC/IJUNT
COMMON/STG/STGR
COMMON/FRN/FIN
COMMON/ITFHN/ITFIN,ITFIN
REAL K12R,K12I,K34R,K34I,K56R,K56I
REAL L1,L2,L3,L4,M1,M2,M3,M4

PRINT NAME OF PROGRAM/DATE OF RUN.

READ(5,7)DATE
7 FORMAT(3A4)
WRITE(6,6)DATE
6 FORMAT(1H1,14(/),14(/),37X,51H OSCILLATING CASCADE FLUTTER PROGRAM
1 ---RUN OF---,3A4/,1H1)

PRINT OUT ALL INPUT INFORMATION VIA INPUT.

CALL INPUT
DO 1005 JTEM=1,50
ICO=0
KOUNT=0
CALL INTIAL
1000 CONTINUE

CUMPXY COMPUTES THE VALUE FOR X AND Y GIVEN THE PARAMETERS ITAVEP
1AND ITSWICH FOR INDICES (RESPECTIVELY).
CALL COMPXY

THE PROGRAM GOES TO 1 IF POINT IS ON INITIAL MACH LINE.

IF((IHAVEP.EQ.0).AND.(IATNW.L.EQ.0))GO TO 1
IF((IHAVEP.EQ.0).AND.(IATNW.L.EQ.0))GO TO 2

TEST HERE IF YOU ARE AT A LOW FOIL PT. IF NOT GO TO NEXT POINT.
IF(IATNW.L .EQ. 1 .AND. IHAVEP .EQ. KOUNT) GO TO 3
IF(IHAVEP.EQ.JLINE)GO TO 3
GO TO 4

INITIAL POINT(MACH LINE) CALC. FOR U,V,AND C AT 1.
1 CALL MACHLN
   GO TO 9

HIGH FOIL CALCULATION HERE FOR U,V,AND C AT 2.
2 CALL HIFOIL
   GO TO 5

LOW FOIL PT CALCULATION HERE FOR U,V,AND C AT 3.
3 CALL LIFOIL
   GO TO 5

GENERAL POINT CALCULATION HERE FOR U,V,AND C AT 4.
4 CALL GENEPT
5 CONTINUE
IF(IATNW.L .EQ. 1 .AND. IHAVEP .EQ. KOUNT) GO TO 25
IF(IHAVEP.NE.JLINE) GO TO 100

TEST IF AT END OF FOIL IF SO COMPUTE Q AND QUIT.
IF(X(IHAVEP+1,ISWTCH)+DSTSTR.GT.FIN) GO TO 101

TEST IF YOU ARE AT A HI AIRFOIL LINE(RIGHT RUNNING MACH) AND IF
1 SO NOTE THIS BY IATNW.L=1 AND JLINE=NGDPTS.
IF(JLINE.EQ.NGDPTS)GO TO 102

INCREMENT FOR NEXT LINES(FIRST OLD ONE(JLINE) BECOMES LAST NEW 1)
ILINE=ILINE+1
JLINE=JLINE+1
IF(JLINE.EQ.NGDPTS) IATNWl=1
C
SWITCH LINES HERE SO FIRST OLD ONE (JLINE) BECOMES LAST NEW ONE.
ZERO OUT NEW LINE POINT INCREMENT COUNTER.
105 IF(ICO.EQ.1) IJUNC=IJUNC-1
IF(ISWITCH.EQ.1) GO TO 103
ISWITCH=1
JSWITCH=2
IHAVEP=0
GO TO 1000
103 ISWITCH=2
JSWITCH=1
IHAVEP=0
GO TO 1000
C
AT 102 SET UP FOR HERE ON IN AT TUNNEL WALL.
102 IATNWl=1
JLINE=NGDPTS
KOUNT=NGDPTS-1
GO TO 105
C
AT 100 INCREMENT TO NEXT POINT ALONG PRESENT LINE.
100 IHAVEP=IHAVEP+1
C
FINISHED TOP AIRFOIL? TERMINATE
IF((ICO.EQ.1).AND.(IJUNC.EQ.0)) GO TO 106
C
PREVENT UNNECESSARY FLOW FIELD CALCULATION.
IF((ICO.EQ.1).AND.(IHAVEP.GT.IJUNC)) GO TO 105
GO TO 1000
C
IN ZONE 1? TERMINATE
101 IF((JLINE-1).LT.IJUNC) GO TO 106
ICO=1
GO TO 105
C
FINISH PITCH MODE CALCULATION.
106 FACTOR=DSTSTR/(REDFRQ*REDFRQ)
M3=M1*FACTOR*2.0
M4=M2*FACTOR*2.0
OMEGA=MMU*RSUBA**2
CA=1.0-M3/OMEGA
DAMP=M4+GSUBA*OMEGA*CAPX
RFREQ=2.0/REDFRQ
WRITE(6,500)RFREQ,M4,M3,CA,DAMP
500
  FORMAT(//,5H1K,=,F17.7,//,4H M4,=,E17.7,5X,3HM3,=,E17.7,5X,2HX,=,E17.7,5X,2HX)
  IF(JTEM.NE.1) GO TO 210
  IF(DAMP.GT.-1E-06) GO TO 210
  IF(DAMP.GT.-1E-06) GO TO 210
  IF(DAMP.GT.-1E-06) GO TO 210
  IF(DAMP.GT.-1E-06) GO TO 210
  IF(INCRE.LE.1E-04) GO TO 240
  GO TO 240
210 IF(ABS(DAMP).LE.1E-06) GO TO 240
  GO TO 240
  GO TO 240
240 IF(CAPX.LT.-1E-06) GO TO 1006
  ROOTX=SQR(T(CAPX)
  OMEGA=1.0/ROOTX
  UF=OMEGA/REDFRQ
  WRITE(6,501)OMEGA,UF
501
  FORMAT(//,15X,19H FLUTTER FREQUENCY=,E20.7,15X,18H FLUTTER VELOC)
  WRITE(*,501)OMEGA
  WRITE(*,502)JTEM
502
  FORMAT(//,12H ITERATIONS=,I5)
  GO TO 1007
280 RFREQ=RFREQ+INCRE
  REDFRQ=2.0/RFREQ
1005 CONTINUE
1006 WRITE(6,503)JTEM
503
  FORMAT(//,8H TOO BAD - THE FREQUENCY RATIO IS IMAGINARY BUT MAY
  BE THE CYCLE IS COMPLETE--ITERATIONS=,I5)
1007 CONTINUE
STOP
END

SUBROUTINE INPUT

C SUBROUTINE INPUT READS ALL INPUT.
C NGDFN IS THE FINENESS OF GRID NUMBER.
C FSTRNM IS THE FREESTREAM MACH NUMBER.
C RTOSPH IS THE RATIO OF SPECIFIC HEATS.
C REDFRQ IS THE REDUCED DIMENSIONLESS FREQUENCY.
C TNWST IS THE DISTANCE BETWEEN AIRFOILS.
XSUBO IS THE DISTANCE (DIMENSIONLESS) FROM THE LEADING EDGE TO THE
1 ELASTIC AXIS.
STGANG IS THE CASCADE STAGGER ANGLE IN DEGREES.
FAZE IS THE UPPER AIRFOIL PHASE LAG IN DEGREES.
UDELTA IS THE UPPER AIRFOIL PHASE LAG IN RADIANS.
MUG IS THE WING DENSITY PARAMETER.
RUSBA IS THE RADIUS OF GRATION.
GSUBA IS THE STRUCTURAL DAMPING COEFFICIENT IN TORSION.
INCRE IS THE AMOUNT THE REDUCED FREQUENCY IS DECREASED EACH TIME
1 SQRT (X) IS CALCULATED.
OMEGA IS MUU-RUBA**2.
ROOTX IS THE RATIO OF AFREQ TO THE FREQUENCY OF OSCILLATION.
HSUBO & ALPHAO ARE THE MAXIMUM AMPLITUDE (DIMENSIONLESS) OF THE
1 AIRFOIL 'PLUNGING & PITCHING OSCILLATION, RESPECTIVELY. EACH IS
2 SET EQUAL TO 1 SINCE THEIR VALUES ARE INDEPENDENT OF THE COMPUTAT
FMANGE IS THE MACH ANGLE.
XLENGTH IS THE LENGTH OF THE INITIAL MACH LINE.
DELTA IS THE STEP SIZE OF INCREMENTING ALONG THE MACH LINES.

DIMENSION U22R(400,3), U22I(400,3), V22R(400,3), V22I(400,3),
1 C22R(400,3), C22I(400,3), X(400,2), Y(400,2)
NGDPTS IS THE NUMBER OF GRID POINTS (INCLUD. XSUBO AND TW POINT)
DSSTSTR IS THE DISTANCE (HORIZONTAL) OF A STREAMLINE FROM ONE GRID
1 POINT TO THE NEXT STREAMLINE GRID POINT.
HDSTSTR IS ONE-HALF OF DSSSTSTR.
TDLTH IS THE HEIGHT OF DELTAS (VERTICAL).
U22R IS THE VARIABLE HOLDING THE REAL PART OF U AT THE RIGHTMOST
1 GRID POINT. THE SECOND INDEX POINTS TO A LINE (MACH) BEING USED
SAME TYPE OF MEANING FOR THE OTHER VARIABLES.
AI AND BI ARE USED AS CONSTANTS IN THE DEFINITIONS OF U, V, AND C.
COMPUTE ALL VALUES FOR INITIAL MACH LINES INITIAL POINT.
SET UP INITIAL INTEGRATION VALUES.
ISWITCH IS THE SWITCH VARIABLE FOR CHOOSING THE LINE TO WORK ON.
JSWITCH IS THE OPPOSITE OF ISWITCH. IF ISWITCH=1, JSWITCH=2 VICE-VERSA.
JLINE CONTAINS THE NUMBER OF POINTS THE LINE EVENTUALLY SHOULD BE.
JLINE=JLINE-1 UNLESS AT HI AIRFOIL THEN JLINE=JLINE(JLINE IS TOTAL)
1 NUMBER OF POINTS CONTAINED IN THE OTHER LINE.
IHAVE IS A COUNTER TELLING THE NUMBER OF PROCESSED POINTS YOU HAVE.
1 CALCULATE FOR THE LINE YOU ARE WORKING ON.
1 IATNW=0 SIGNIFIES YOU ARE NOT AT THE HI AIRFOIL YET. IATNW WILL
1 BE SET =1 WHEN FINISHED WITH LINE=NGDPTS (AT HI AIRFOIL).

COMMON/BLK1/NGRDEFN, FSTRMN, RTOSPH, REDFRO, XSUBO, TNWOST, FMANGL, XLNGTH000000340
1, DELTAS, NGDPTS, DSSSTSTR, HDSTSTR, IRLNGL, U22R, U22I, V22R, V22I, C22R, C22I, 000000350
2 X, Y, S, DELTA, ISWITCH, JSWITCH, JLINE, JLINES, IHAVEP, IATNW, AI, BI, M1
3, M2
4, STGANG, NSTPTS, ICOUNT, JCOUNT, IJUNC, ISPLIT
000000360
5, LCOUNT, MCOUNT, ISWICH, JSWICH
COMMON/BLK3/MUH, RSUBA, GSUBA, INC
COMMON/JOUC/J1UNT
COMMON/STG/STGR
COMMON/FINE/FIN
REAL KJ2R, KI21, K34R, K34I, K56R, K56I
REAL LI, L2, L3, L4, M1, M2, M3, M4
REAL MUH, INC
NAMELIST/NAM1/NGRDFN, FSTRMN, RTOSPH, REDFRQ, XSUBO, TNWDST, STGANG, FZE
00002470
1, MUH, RSUBA, GSUBA, INC, FIN
00002480
READ(5, NAM1)
WRITE(6, NAM1)
FMANGL = ARSIN(1.0/FSTRMN)
FNGRDN = NGRDFN
XLENGTH = TNWDST/SIN(FMANGL)
NGDPTS = NGRDFN
FNGDPT = NGDPTS
DELTA = XLENGTH/FNGDPT
TRWGLH = TNWDST/FNGDPT
HDSTAL = DELTA*COS(FMANGL)
DISTSR = HDSTAL*2.0
STGR = TNWDST*TAN(STGANG*0.1745329E-01)
NSTPTS = STGR/DISTSR
DISTSR = NSTPTS*DSTSR
IF (STGR - DISTSR*GE.0.50) NSTPTS = NSTPTS + 1
IF (NGDPTS/2.E.0.NSTPTS-1) NSTPTS = NSTPTS + 1
IF (NGDPTS/2.E.0.NSTPTS-1) NSTPTS = NSTPTS + 1
STGR = NSTPTS*DSTSR
STGANG = ATAN(STGR/TNWDST)*57.29578
IJUNT = NGDPTS/2 + NSTPTS
NSTPTS = NSTPTS + 1
ISPLIT = NGDPTS/2 - NSTPTS
NGDPTS = NGDPTS + 1
S = SQRT(FSTRMN**2*FSTRMN - 1.0)
DELTA = FZE*0.1745329E-01
WRITE(6,5) NGRDFN, FSTRMN, RTOSPH, REDFRQ, TNWDST, XSUBO, STGANG, DSTSR
00002490
1, MUH, RSUBA, GSUBA, FZE
00002500
5 FORMAT(1H5, 20H GRID FINENESS INPUT NUMBER=, I10, 24H FREESTREAM SOL 00002760
1 MACH NUMBER = F20.7, 25H RATIO OF SPECIFIC HEATS = F20.7, 34H 00002770
2 REDUCED (DIMENSIONLESS) FREQUENCY = F20.7, 27H DISTANCE BETWEEN 00002780
3 AIRFOILS = F20.7, 52H HORIZONTAL POSITION (DIMENSIONLESS) OF ELAS 00002793
4 TIC AXIS = F20.7, 26H COMPATIBLE STAGGER ANGLE = F20.7, 9H 00002800
5 TA X = F20.7, 24H WING DENSITY PARAMETER = F20.7, 20H RADIUS 00002810
6 F GYRATION = F20.7, 42H TORSIONAL STRUCTURAL DAMPING COEFFICIENT 00002820
7 = F20.7, 25H UPPER AIRFOIL PHASE LAG = F20.7)
00002830
WRITE(6, 6)
00002840
6 FORMAT(1H5, 10X, 80H VALUES OF FREQUENCY RATIO FOR VARIOUS NON-DIMEN 00002850
SIONAL FREQUENCIES OF OSCILLATION)
00002860
SUBROUTINE INITIAL

INITIAL INITIALIZES ALL FLOW FIELD QUANTITIES.

DIMENSION U2R(400,3),U2I(400,3),V2R(400,3),V2I(400,3),
C2R(400,3),C2I(400,3),X(400,2),Y(400,2)
DIMENSION U3R(200,2),U3I(200,2),V3R(200,2),V3I(200,2),C3R(200000,2)
1,2),C3I(200,2)
COMMON/BLK1/NGDFN,FSTRMN,RTOSPH,REDRFQ,XSUBO,TNWDST,FMANGL,XLONGTH
1,DELTA,NGDPTS,DSTSTR,HODSTRL,TRNLGH,U2R,U2I,V2R,V2I,C2R,C2I
2,Y,SY,S,DELTA,ISWTCH,JSWTCH,ILINE,ILINE,IATNWL,AL,HI,M1
3,M2
K12R,K12I,K34R,K34I,K56R,K56I,VRPANL,V1PANL
4,STGANG,NSTPTS,ICOUNT,JCOUNT,ISPLIT
5,LCOUNT,MCOUNT,ISWICH,JSWICH
COMMON/BLK5/U3R,U3I,V3R,V3I,C3R,C3I
COMMON/JUNC/IXJUNCT
COMMON/IFINE/IFIN,JFIN
REAL K12R,K12I,K34R,K34I,K56R,K56I
REAL L1,L2,L3,L4,M1,M2,M3,M4
IXJUNCT=1
AI=5*REDRFQ*DSTSTR
BI=25*REDRFQ*(FSTRMN/FSTRMN/(FSTRMN/FSTRMN-1.0))*DSTSTR

SET UP INITIAL VALUES FOR U, V, AND C AT (0,0).

50 VRPANL=-1.0/S
V1PANL=REDRFQ*XSUBO/S
60 U2R(1,1)=-VRPANL
U2I(1,1)=-V1PANL
V2R(1,1)=-U2R(1,1)
V2I(1,1)=-U2I(1,1)
C2R(1,1)=-U2R(1,1)
C2I(1,1)=-U2I(1,1)
X(1,1)=0.0
Y(1,1)=0.0
IF((ISPLIT.NE.-1).AND.(DELTA.LE.1E-05)) GO TO 63
M1=XSUBO*C2R(1,1)
M2=XSUBO*C2I(1,1)
GO TO 64
63 M1=2.0*XSUBO*C2R(1,1)
M2=2.0*XSUBO*C2I(1,1)
GO TO 64
64 IF(DELTA.LE.1E-05) GO TO 65
150 VRPANL=-(COS(DELTA)+REDRFQ*SIN(DELTA)*XSUBO)/S
V1PANL=-(SIN(DELTA)-REDRFQ*COS(DELTA)*XSUBO)/S

160 U33R(1,1)=VRPANL
    U33I(1,1)=VIPANL
    V33R(1,1)=U33R(1,1)
    V33I(1,1)=U33I(1,1)
    C33R(1,1)=V33R(1,1)
    C33I(1,1)=V33I(1,1)
    IF(ISPLIT.EQ.-1) GO TO 65
M1=M1-XXSUB0*(C33R(1,1)*COS(DELTA)+C33I(1,1)*SIN(DELTA))
M2=M2-XXSUB0*(C33I(1,1)*COS(DELTA)-C33R(1,1)*SIN(DELTA))
ISWITCH=2
JSWITCH=1
ISWITCH=1
JSWITCH=1
IF(IN=0)
    IF(LINE=2)
        JLINE=1
        IHAVEP=0
        IATNWL=0
        JCOUNT=0
        LCOUNT=0
        MCOUNT=1
        RETURN
    END

SUBROUTINE MACHLN
MACHLN COMPUTES THE VALUES OF U, V, AND C ALONG THE INITIAL MACH LINE AT THE GIVEN X VALUE OF THE MACH LINE.

DIMENSION U22R(400,3),U22I(400,3),V22R(400,3),V22I(400,3),
C22R(400,3),C22I(400,3),X(400,2),Y(400,2)
DIMENSION U33R(200,2),U33I(200,2),V33R(200,2),V33I(200,2),C33R(2000,0,200)
X(1,2),C33I(200,2)
COMMON/BLK1/NGRDFN,FRSILMN,RTOSPHE,REDFRO,XXSUB0,TNWDST,FMANGL,XLNGTH
1,DELTA:S,OPTS,TSTSTR,HOSTRL,TRNGLH,U22R,U22I,V22R,V22I,C22R,C22I
M2,1,K12R,K12I,K34R,K34I,K56R,K56I,VRPANL,VIPANL
COMMON/BLK5/U33R,U33I,V33R,V33I,C33R,C33I
REAl L1,L2,L3,L4,M1,M2,M3,M4
ICREMENT=JLINE
COMMON/MCOUNT,MCOUNT,MCOUNT,MCOUNT
IF(JCOUNT,LT,1)JUNC GO TO 10
JCOUNT=JCOUNT+1
KCOUNT=KCOUNT+1
00003330
00003340
00003350
00003360
00003370
00003380
00003390
00003400
00003410
00003420
00003430
00003440
00003450
00003460
00003470
00003480
00003490
00003500
00003510
00003520
00003530
00003540
00003550
00003560
00003570
00003580
00003590
00003600
00003610
00003620
00003630
00003640
00003650
00003660
00003670
00003680
00003690
00003700
00003710
00003720
00003730
00003740
00003750
00003760
00003770
00003780
SUBROUTINE HIFOIL

HIFOIL COMPUTES U, V, AND C AT AN UPPER AIRFOIL POINT.

DIMENSION U22R(400,3), U22I(400,3), V22R(400,3), V22I(400,3),
C22R(400,3), C22I(400,3), X(400,3), Y(400,2),
COMMON/BLK1,NGDPTS,OSTSTR,HOSTR,TRNLHL,U22R,U22I,V22R,V22I,
C22R,C22I,DSSTWCH,JSWITCH,LINE,ISWITCH,ILINE,ISHAPE,IATNW,ALPH,MI
1,DELTAS,NGDPTS,OSTSTR,HOSTR,TRNLHL,U22R,U22I,V22R,V22I,C22R,C22I,
2X,Y,S,DELTAS,ISWITCH,JSWITCH,LINE,ISHAPE,ISHAPE,ILINE,ISHAPE,ISWITCH,
3,M2,K12R,K12I,K34R,K34I,K56R,K56I,VRPANL, VIPANL
00004630
00004640
00004650
00004660
00004670
00004680
00004690
00004700
00004710
00004720
00004730
4:STGANG,NSTPTS,ICOUNT,JCOUNT,JUNC,ISPLIT
5:JCOUNT,MCOUNT,ISWICH,JSWICH
COMMON:PCOR/KOUNT
COMMON/STG/STR
COMMON/JUNC/IJUNC
COMMON/IFINE/IFIN,JFIN
REAL K12R,K121,K34R,K34I,K56R,K56I
REAL L1,L2,L3,L4,M1,M2,M3,M4
TOPCRD=X(I,ISWITCH)-STG
IF(JLINE.EQ.
NLGPTS-1) IATNWL=0
IF(JCOUNT.EQ.0) GO TO 80
IF(JCOUNT.NE.0) GO TO 75
K12R=0.0
K12I=0.0
J=1
GO TO 80
75 K12R=U22R(JCOUNT,3)+C22R(JCOUNT,3)+AI*U22I(JCOUNT,3)
K12I=-AI*U22R(JCOUNT,3)+U22I(JCOUNT,3)+C22I(JCOUNT,3)
J=1
80 IF(JCOUNT.NE.0) GO TO 85
I=1
J=2
K12R=U22R(I,JSWITCH)+C22R(I,JSWITCH)+AI*U22I(I,JSWITCH)
K12I=AI*U22R(I,JSWITCH)+U22I(I,JSWITCH)+C22I(I,JSWITCH)
85 V22R(I,JSWITCH)=-(COS(Delta)-REDFRQ*SIN(Delta))*((TOPCRD-XSUB0))%S
V22I(I,JSWITCH)=-(SIN(Delta)+REDFRQ*COS(Delta))*((TOPCRD-XSUB0))%S
90 K56R=V22R(1,JSWITCH)
K56I=V22I(1,JSWITCH)
91 K34R=U22R(J,JSWITCH)-V22R(J,JSWITCH)+BI*(U22I(J,JSWITCH)-U22I(J,JSWITCH))
1C22I(J,JSWITCH)
K34I=U22I(J,JSWITCH)-V22I(J,JSWITCH)-BI*(U22R(J,JSWITCH)-U22R(J,JSWITCH))
1C22R(J,JSWITCH)
92 GL=1.0-AI*BI
G2=2.0*BI
G3=GL*G1+G2*G2
G4=K56R+K34R-BI*K12I
G5=K56I+K34I+BI*K12R
U22R(1,JSWITCH)=G4*G1+G5*G2)/G3
U22I(1,JSWITCH)=-(G4*G2+G5*G1))/G3
C22R(1,JSWITCH)=K12R-U22R(1,JSWITCH)+U22I(1,JSWITCH)*AI
C22I(1,JSWITCH)=K12I-U22I(1,JSWITCH)-U22R(1,JSWITCH)*AI
CPR1=C22R(1,JSWITCH)*COS(Delta)+C22I(1,JSWITCH)*SIN(Delta)
CPR2=C22R(1,JSWITCH)*COS(Delta)-C22I(1,JSWITCH)*SIN(Delta)
CPI1=U22I(1,JSWITCH)*COS(Delta)-C22R(1,JSWITCH)*SIN(Delta)
CPI2=U22I(1,JSWITCH)*COS(Delta)+C22R(1,JSWITCH)*SIN(Delta)
IF(JFIN.EQ.1) GO TO 130
C
C AT START OF ZONE AND CLOSE TO END? SKIP LIFT CALCULATION.
C IF((LCOUNT.EQ.0).AND.(TOPCRD+DSTSTR.GT.1.0)) GO TO 130 00005210
C AT START OF ZONE OR END OF AIRFOIL? START/STOP INTEGRATION. 00005220
C IF((LCOUNT.EQ.0).OR.(TOPCRD+DSTSTR.GT.1.0)) GO TO 95 00005230
C M1=M1+2.0*CPR2*(TOPCRD-XSUBO) 00005240
C M2=M2+2.0*CPI2*(TOPCRD-XSUBO) 00005250
C GO TO 100 00005260
95 M1=M1+CPR2*(TOPCRD-XSUBO) 00005270
C M2=M2+CPI2*(TOPCRD-XSUBO) 00005280
C LCOUNT=LCOUNT+1 00005290
C SET UP FOR END OF ZONE CHECK. 00005300
C IF(ISWICH.NE.1) GO TO 105 00005310
C IF(LCOUNT.EQ.IJINT) GO TO 110 00005320
C GO TO 120 00005330
100 IF(LCOUNT.NE.JCOUNT) GO TO 120 00005340
C AT END OF ZONE; LINEARLY EXTRAPOLATE LAST VALUE AND STOP INTEGRAL. 00005350
C M1=M1+(2.0*CPR2-CPR1)*(TOPCRD+DSTSTR-XSUBO) 00005360
C M2=M2+(2.0*CPI2-CPI1)*(TOPCRD+DSTSTR-XSUBO) 00005370
C ZERO COUNTER AND SWITCH ZONE MARKERS. 00005380
C LCOUNT=0 00005390
C IF(ISWICH.EQ.1) GO TO 115 00005400
C ISWICH=1 00005410
C GO TO 120 00005420
105 ISWICH=2 00005430
C ALPHA=90-BETA? ELIMINATE ODD ZONE. 00005440
C IF(JCOUNT.EQ.0) ISWICH=1 00005450
C CONTINUE 00005460
C IF(TOPCRD+DSTSTR.LE.1.0) GO TO 130 00005470
C COMPUTE VALUES OF C22 AT TOPCRD=1. 00005480
C DELCR=C22R(1,ISWITCH)+(C22R(1,ISWITCH)-C22R(1,JSWITCH))*(1.0-X(1,ISWT)) 00005490
C 1CH+STGR)/DSTSTR 00005500
C DELCI=C22I(1,ISWITCH)+(C22I(1,ISWITCH)-C22I(1,JSWITCH))*(1.0-X(1,ISWT)) 00005510
C 1CH+STGR)/DSTSTR 00005520
C DELCPX=DELCR*COS(Delta)+DELCI*SIN(Delta) 00005530
C DELCPI=DELCI*COS(Delta)-DELCR*SIN(Delta) 00005540
C
SUBROUTINE GENFPT

GENFPT COMPUTES U, V, AND C AT A GENERAL FIELD POINT.

DIMENSION U2(400, 3), U2I(400, 3), V2R(400, 3), V2II(400, 3),
1 C2I2R(400, 3), C2II(400, 3), X(400, 2), Y(400, 2),
DIMENSION U3R(200, 2), U3RI(200, 2), V33R(200, 2), V33RI(200, 2),
1, 2, 3, C3SI(200, 2)
COMMON/BLK1/NORDEN, ESTRMN, RTOPH, REDFRO, XSUBO, TNWDST, FMANGL, XLNGTH
1, DELTAS, NGPDPS, TSSTR, HSSTR, TRNLHL, U22R, U22I, V2R, V2II, C2R, C2II,
2 X, S, DELTA, ISWCH, JTST1, JL1NE, IHAVEP, IATWNL, AI, BI, MI
1, 2, 3, MZ, K21R, K21I, K34R, K34I, K56R, K56I, VROPNL, VIPANL
4, STGANG, NSTPST, ICOUNT, JCOUNT, IJUNC, ISPLIT
5, LCOUNT, MPCTT, ISWCH, JSWCH
COMMON/BLK5/U33R, U33RI, V33R, V33RI, C33R, C33I
REAL K12R, K12I, K34R, K34I, K56R, K56I
REAL L1, L2, L3, L4, M1, M2, M3, M4
1 = IHAVEP + 1
1F(1ATWNL .EQ. 0) GO TO 10
1K12R = U22R(I, JSWCH) + C22R(I, JSWCH) + AI * U22I(I, JSWCH)
1K12I = AI * U22R(I, JSWCH) + U22I(I, JSWCH) + C22I(I, JSWCH)
1K24R = U22R(I+1, JSWCH) - V22R(I+1, JSWCH) + BI * (U22I(I+1, JSWCH) - C22I(I+1, JSWCH))
1 K34I = U22I(I+1, JSWCH) - V22I(I+1, JSWCH) - BI * (U22R(I+1, JSWCH) - C22R(I+1, JSWCH))
G0 TO 10
1K12R = U22R(I, JSWCH) + C22R(I, JSWCH) + AI * U22I(I, JSWCH)
1K12I = AI * U22R(I, JSWCH) + U22I(I, JSWCH) + C22I(I, JSWCH)
1K24R = U22R(I+1, JSWCH) - V22R(I+1, JSWCH) + BI * (U22I(I+1, JSWCH) - C22I(I+1, JSWCH))
1 K34I = U22I(I+1, JSWCH) - V22I(I+1, JSWCH) - BI * (U22R(I+1, JSWCH) - C22R(I+1, JSWCH))
G0 TO 12
1K56R = U22R(I, JSWCH) + V22R(I, JSWCH) + BI * (U22I(I, JSWCH) + C22I(I, JSWCH))
1K56I = AI * U22R(I, JSWCH) + U22I(I, JSWCH) + C22I(I, JSWCH)
1K24R = U22R(I+1, JSWCH) - V22R(I+1, JSWCH) + BI * (U22I(I+1, JSWCH) - C22I(I+1, JSWCH))
1 K34I = U22I(I+1, JSWCH) - V22I(I+1, JSWCH) - BI * (U22R(I+1, JSWCH) - C22R(I+1, JSWCH))
G0 TO 12
U331(I, ISWITCH) = (-G8*G6+G9*C3)/G7
V331(I, ISWITCH) = 0.5*(K56R-K34R)
C331(I, ISWITCH) = K12R-U33R(I, ISWITCH)+U331(I, ISWITCH)*AI
C331(I, ISWITCH) = K12I-U33II(I, ISWITCH)-U33R(I, ISWITCH)*AI
IF(JLINE_4.EQ._ISPLIT) GO TO 500
U22R(I, J) = U33R(I, ISWITCH)
U22I(I, J) = U33I(I, ISWITCH)
V22R(I, J) = V33R(I, ISWITCH)
V22I(I, J) = V33I(I, ISWITCH)
C22R(I, J) = C33R(I, ISWITCH)
C22I(I, J) = C33I(I, ISWITCH)
CONTINUE
RETURN
END

SUBROUTINE LOFOIL

LOFOIL COMPUTES THE VALUES OF U, V, AND C AT A LOWER AIRFOIL POINT.

DIMENSION U22R(400,3), U22I(400,3), V22R(400,3), V22I(400,3),
C22R(400,3), C22I(400,3), X(400,2), Y(400,2)
DIMENSION U33R(200,2), U33I(200,2), V33R(200,2), V33I(200,2), C33R(2000000840
1,2), C33I(200,2)
COMMON/BLKI/NGRDFN, FSTRMN, RTOSPH, REDFRQ, XSUB0, TNWST, FMANGL, XNLGTH
1, DELTAS, NGDPTS, DSTRTR, TDCTST, TRNLGH, U22R, U22I, V22R, V22I, C22R, C22I
2, X, Y, S, DELTA, ISWITCH, JSWITCH, ILINE, JLINE, IHAVEP, IATNWL, AI, BI, M1
3, M2, K12R, K12I, K34R, K34I, K56R, K56I, VRPANL, VIPANL
4, STGANG, NSTPST, ICOUNT, JCOUNT, IJUNC, ISPLIT
5, LCOUNT, MCOUNT, ISWICH, JSWICH
COMMON/BLK5/U33R, U33I, V33R, V33I, C33R, C33I
COMMON/PCOR/KOUNT
COMMON/IFINE/IFIN, JFIN
REAL K12R, K12I, K34R, K34I, K56R, K56I
REAL LI, L2, L3, L4, M1, M2, M3, M4
999 V22R(ILINE, ISWITCH) = -1.0/SQRT(FSTRMN*FSTRMN-1.0)
V22I(ILINE, ISWITCH) = (-REDFRQ*(X(IHAVEP+1, ISWITCH)-XSUB0))/SQRT(FSTRMON*FSTRMN-1.0)
10 K34R = V22R(ILINE, ISWITCH)
K34I = V22I(ILINE, ISWITCH)
K12R = U22R(ILINE, ISWITCH)+C22R(ILINE, JSWITCH)+AI*U22I(ILINE, JSWITCH)
K12I = AI*U22R(ILINE, JSWITCH)+U22I(ILINE, JSWITCH)+C22I(ILINE, JSWITCH)
IF (IATNWL_.EQ._0) GO TO 91
K56R = U22R(KOUNT, ISWITCH)+V22R(KOUNT, ISWITCH)+BI*(U22I(KOUNT, ISWITCH)
1-C22I(KOUNT, ISWITCH))
K56I = U22I(KOUNT, ISWITCH)+V22I(KOUNT, ISWITCH)+BI*(U22R(KOUNT, ISWITCH)
1-C22R(KOUNT, ISWITCH))

00006630
00006640
00006650
00006660
00006670
00006680
00006690
00006700
00006710
00006720
00006730
00006740
00006750
00006760
00006770
00006780
00006790
00006800
00006810
00006820
00006830
00006840
00006850
00006860
00006870
00006880
00006890
00006900
00006910
00006920
00006930
00006940
00006950
00006960
00006970
00006980
00006990
00007000
00007010
00007020
00007030
00007040
00007050
00007060
00007070
00007080
00007090
00007100
00007110
00007120
00007130
CPR1=C33R(JLINE,JSWITCH)*COS(Delta)+C33I(JLINE,JSWITCH)*SIN(Delta)
CPI1=C33I(JLINE,JSWITCH)*COS(Delta)-C33R(JLINE,JSWITCH)*SIN(Delta)
IF(IFIN.EQ.1) GO TO 140
M1=M1+2.0*CPR2*(X(K,L)-XSUBO)
M2=M2+2.0*CPI2*(X(K,L)-XSUBO)
IF((JLINE,NE.ISPLIT),AND,(X(K,L)+DSTSTR.LE.1.0)) GO TO 94
M1=M1+2.0*CPR2-CPR1*(X(K,L)+DSTSTR-XSUBO)
M2=M2+2.0*CPI2-CPI1*(X(K,L)+DSTSTR-XSUBO)
94 IREF=0
IF(IFIN.EQ.1) GO TO 140
AT START OF ZONE AND CLOSE TO END? SKIP LIFT CALCULATION.
IF((MCOUNT.EQ.0).AND.(X(ILINE,ISWTCH)+DSTSTR.GT.1.0)) GO TO 140
AT START OF ZONE OR END OF AIRFOIL? START/STOP INTEGRATION.
IF((MCOUNT.EQ.0).OR.(X(ILINE,ISWTCH)+DSTSTR.GT.1.0)) GO TO 95
M1=M1-2.0*C22R(ILINE,ISWTCH)*(X(ILINE,ISWTCH)-XSUBO)
M2=M2-2.0*C22I(ILINE,ISWTCH)*(X(ILINE,ISWTCH)-XSUBO)
IF(IREF.EQ.1) GO TO 130
GO TO 100
95 M1=M1-C22R(ILINE,ISWTCH)*(X(ILINE,ISWTCH)-XSUBO)
M2=M2-C22I(ILINE,ISWTCH)*(X(ILINE,ISWTCH)-XSUBO)
IF(IREF.EQ.1) GO TO 130
IF(X(ILINE,ISWTCH)+DSTSTR.GT.1.0) GO TO 120
100 MCOUNT=MCOUNT+1
SET UP FOR END OF ZONE CHECK.
IF(JSWITCH.NE.1) GO TO 105
IF(MCOUNT.EQ.IJUNC) GO TO 110
GO TO 120
105 IF(MCOUNT.NE.(ISPLIT+1)) GO TO 120
AT END UF ZONE, LINEARLY EXTRAPOLATE LAST VALUE AND STOP INTEGRAL.
110 M1=M1-2.0*C22R(ILINE,ISWTCH)-C22R(JLINE,JSWITCH)*(X(ILINE,ISWTCH)
1+DSTSTR-XSUBO)
M2=M2-2.0*C22I(ILINE,ISWTCH)-C22I(JLINE,JSWITCH)*(X(ILINE,ISWTCH)
1+DSTSTR-XSUBO)
IN ZONE 1? GET LIFT & MOMENT OF TOP AIRFOIL.
IF(IREF.EQ.1) GO TO 93
ZERO COUNTER AND SWITCH ZONE MARKERS.
MCOUNT=0
IF(JSWICH.EQ.1) GO TO 115
JSWICH=1
GO TO 120
115 JSWICH=2

ALPHA=90-BETA?  ELIMINATE ODD ZONE.
IF(ISPLIT.EQ.-1) JSWICH=1
NOT IN ZONE 1?  SKIP IT.
120 IF((ISPLIT.EQ.-1).OR.(JCOUNT.NE.O).OR.(DELTA.GT.1E-05)) GO TO 130
IF(IREF=1)
IF(NSTPTS.NE.1) GO TO 125

AT END OF AIRFOIL?  ADD FOR TOP AIRFOIL.
IF(X(ILINE,ISWTCH)+DSTSTR.GT.1.0) GO TO 95
AT END OF ZONE1?  ADD FOR TOP AIRFOIL.
IF(MCOUNT.EQ.0) GO TO 110
GO TO 93
125 IF(X(ILINE,ISWTCH)+DSTSTR.GT.1.0) GO TO 128
IF(MCOUNT.EQ.0) GO TO 130
IF(MCOUNT-ISPLIT-2)95,95,130
128 IF(MCOUNT-ISPLIT-1) 95,95,130
130 CONTINUE
GO TO 140

IF(X(ILINE,ISWTCH)+DSTSTR.LE.1.0) GO TO 140

COMPUTE VALUES OF C22 AT X= 1.

DELCR=C22R(ILINE,ISWTCH)+(C22R(ILINE,ISWTCH)-C22R(JLINE,JSWTCH))*(1.0-X(ILINE,ISWTCH))/DSTSTR
DELCI=C22I(ILINE,ISWTCH)+(C22I(ILINE,ISWTCH)-C22I(JLINE,JSWTCH))*(1.0-X(ILINE,ISWTCH))/DSTSTR

COMPUTE CHANGE IN LIFT AND MOMENT DUE TO UNDERSHOOT.

DELM1=(C22R(ILINE,ISWTCH)*(X(ILINE,ISWTCH)-XSUB0)+DELCR*(1.0-XSUB0))/DSTSTR
DELM2=(C22I(ILINE,ISWTCH)*(X(ILINE,ISWTCH)-XSUB0)+DELCI*(1.0-XSUB0))/DSTSTR

IF NEEDED ADD FOR TOP AIRFOIL NOT COMPUTED IN HIFOIL.
IF(DELTA.GT.1E-05) GO TO 135
IF(IREF.EQ.0) GO TO 135
IF(MCOUNT.GT.(ISPLIT+1)) GO TO 135
M1=M1-DELM1
M2=M2-DELM2
135 M1=M1-DELM1
M2=M2-DELM2
IFIN=1
140 CONTINUE
RETURN
END

SUBROUTINE COMPY
DIMENSION U22R(400,3),U22I(400,3),V22R(400,3),V22I(400,3),
1C22R(400,3),C22I(400,3),X(400,2),Y(400,2)
COMMON/BLK1/NGRDFN,EFRM,RROS,RHDFN,RMENG,XMLNG,XMLNGTH
1,DELTAS,NGDPTS,_DSTSTR,HDSTRL,TRNGLH,U22R,U22I,V22R,V22I,C22R,C22I
2X,Y,S,DELTAS,ISWITCH,JSWITCH,IOWNER,IPROP,NEI,IAE,soul
3,M2,K1R,K12R,K1I,K12I,K34R,K34I,K56R,K56I,VRPANL,VIPANL
4,STGANG,NSPTTS,ICOUNT,JCOUNT,IJUNC,ISPLIT
5,SCOUNT,MCOUNT,ISNITCH,JSWITCH
REAL L1,L2,L3,L4,M1,M2,M3,M4
00008620

IF FALLS THROUGH, COMPUTE X,Y FOR NEXT ITH POINT ON LINE.(I TH.GT.2)
00008620
IF(IHAVEP.EQ.0) GO TO 1
I=IHAVEP+1
X(I,ISWITCH)=X(IHAVEP,ISWITCH)+HDSTRL
Y(I,ISWITCH)=Y(IHAVEP,ISWITCH)-TRNGLH
GO TO 2

1 IF((IATWLN.EQ.1).AND.(I LINE.LE.NGDP T-1)) GO TO 3
I=IHAVEP+1
X(I,ISWITCH)=X(1,JSWITCH)+HDSTRL
Y(I,ISWITCH)=Y(1,JSWITCH)+TRNGLH
GO TO 2

2 IF GONE TO 3 COMPUTE FIRST POINT ON RIGHT RUNNING MACH LINE EMANATING FROM THE TUNNEL WALL.
3 IF(IHAVEP.EQ.0) GO TO I
X(I,ISWITCH)=X(I,JSWITCH)+HDSTRL
Y(I,ISWITCH)=Y(I,JSWITCH)+TRNGLH
RETURN

00008630
LIST OF REFERENCES


INITIAL DISTRIBUTION LIST

No. Copies

1. Defense Documentation Center
   Cameron Station
   Alexandria, Virginia  22314  2

2. Library, Code 0212
   Naval Postgraduate School
   Monterey, California  93940  2

3. Professor R. W. Bell, Code 57Be
   Department of Aeronautics
   Naval Postgraduate School
   Monterey, California  93940  1

4. Assoc. Professor M. F. Platzer, Code 57P1
   Department of Aeronautics
   Naval Postgraduate School
   Monterey, California  93940  15

5. LCDR Henry G. Chalkley, USN
   Box 344, CFB Summerside
   Slemon Park, P.E.I.
   Canada  1

6. Dr. H. J. Mueller (AIR - 310)
   Naval Air Systems Command
   Washington, D. C.  20360  1

7. Professor M. H. Vavra, Code 57Va
   Department of Aeronautics
   Naval Postgraduate School
   Monterey, California  93940  1
Supersonic flow past oscillating flat plate cascades with supersonic leading-edge locus is analysed using a linearized method of characteristics valid for arbitrary frequencies and an elementary analytical theory valid only for low frequencies of oscillation. These two methods are extensions of previous work by Teipel and Sauer for the single airfoil in an unbounded supersonic flow to the case of airfoils oscillating in cascade. Included is the determination of pressure distributions and both a two-degree-of-freedom (bending and torsion) flutter analysis and a single-degree-of-freedom (torsion) flutter analysis. Numerically determined flutter boundaries are presented for various primary parameters such as, Mach number, solidity, stagger angle, density ratio, structural damping coefficient, and elastic axis position. Also, results are presented for the related problem of supersonic wind tunnel interference (including the effect of tunnel porosity) and airfoil-airfoil interference.
Cascades
Flutter
Supersonic leading-edge locus
Method of characteristics
Thesis
C33865 Chalkley
c.1 A study of supersonic cascade flutter.
A study of supersonic cascade flutter.