

AD-A266 275



2

NPS-AA-93-001

NAVAL POSTGRADUATE SCHOOL

Monterey, California

DTIC
ELECTE
JUL 01 1993
S A D



A RAPID COMPUTATIONAL MODEL
FOR ESTIMATING THE PERFORMANCE OF
COMPLIANT AIRFOILS IN CASCADES

by

F. Sisto

M. Avila

Technical Report for Period
January 1992 - June 1992

93-15006

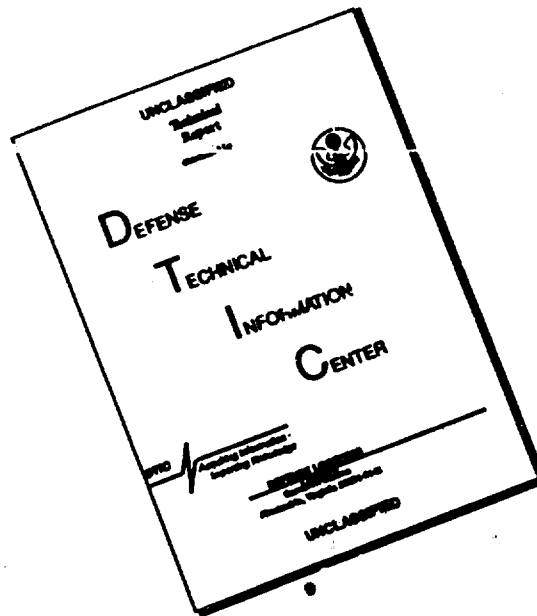


Approved for public release; distribution unlimited

Prepared for: Naval Postgraduate School
Monterey, CA 93943

93

DISCLAIMER NOTICE



THIS DOCUMENT IS BEST QUALITY AVAILABLE. THE COPY FURNISHED TO DTIC CONTAINED A SIGNIFICANT NUMBER OF PAGES WHICH DO NOT REPRODUCE LEGIBLY.

NAVAL POSTGRADUATE SCHOOL
MONTEREY, CA 93943

Rear Admiral T. A. Mercer
Superintendent

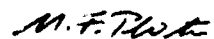
Harrison Shull
Provost

This report was prepared in conjunction with research conducted for the Naval Postgraduate School and funded by the Naval Postgraduate School.

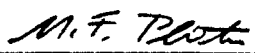
Reproduction of all or part of this report is authorized.

This report was prepared by:


F. SISTO
Senior Research Associate


M. AVILA
LT, U.S. Navy

Reviewed by:


M. F. PLATZER
Director, Navy-NASA Joint Institute

Released by:


PAUL J. MARTO
Dean of Research

REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION UNCLASSIFIED			1b. RESTRICTIVE MARKINGS	
2a. SECURITY CLASSIFICATION AUTHORITY			3. DISTRIBUTION/AVAILABILITY OF REPORT Approved for public release; distribution unlimited	
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE				
4. PERFORMING ORGANIZATION REPORT NUMBER(S) NPS-AA-93-001			5. MONITORING ORGANIZATION REPORT NUMBER(S) NPS-AA-93-001	
6a. NAME OF PERFORMING ORGANIZATION Naval Postgraduate School		6b. OFFICE SYMBOL (If applicable) AA		7a. NAME OF MONITORING ORGANIZATION Naval Postgraduate School
6c. ADDRESS (City, State, and ZIP Code) Monterey, CA 93943			7b. ADDRESS (City, State, and ZIP Code) Monterey, CA 93943	
8a. NAME OF FUNDING/SPONSORING ORGANIZATION Naval Postgraduate School		8b. OFFICE SYMBOL (If applicable) AA		9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER National Research Council
8c. ADDRESS (City, State, and ZIP Code) Monterey, CA 93943			10. SOURCE OF FUNDING NUMBERS	
			PROGRAM ELEMENT NO	PROJECT NO
			TASK NO	WORK UNIT ACCESSION NO
11. TITLE (Include Security Classification) A Rapid Computational Model for Estimating the Performance of Compliant Airfoils in Cascades				
12. PERSONAL AUTHOR(S) F. Sisto and M. Avila				
13a. TYPE OF REPORT Technical		13b. TIME COVERED FROM 1/92 TO 6/92		14. DATE OF REPORT (Year, Month, Day) 92/07/01
15. PAGE COUNT 29				
16. SUPPLEMENTARY NOTATION				
17. COSATI CODES			18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)	
FIELD	GROUP	SUB-GROUP		
			Turbomachinery Aeroelasticity	
19. ABSTRACT (Continue on reverse if necessary and identify by block number)				
We consider the problem of designing the blades in a turbomachine to have a specific schedule of structural stiffness in order to achieve more favorable aerothermodynamic performance characteristics than is possible with rigid blades. This problem of "aero-thermodynamic tailoring" is extremely complex because of the need to couple the flow computation over a deformable surface with the structural dynamics computation. In this report an analysis is presented which accomplishes this objective by means of a two-dimensional incompressible potential flow model using vortex sheet representation of the blades. Details of the analysis and of the resulting computer program together with representative results are described.				
20. DISTRIBUTION/AVAILABILITY OF ABSTRACT ☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT. ☐ DTIC USERS			21. ABSTRACT SECURITY CLASSIFICATION UNCLASSIFIED	
22a. NAME OF RESPONSIBLE INDIVIDUAL M. F. Platzer			22b. TELEPHONE (Include Area Code) (408)656-2058	22c. OFFICE SYMBOL AA/P1

A RAPID COMPUTATIONAL MODEL FOR ESTIMATING THE
PERFORMANCE OF COMPLIANT AIRFOILS IN CASCADES

F. Sisto and M. Avila

SUMMARY

By designing the blades in a turbomachine to have a specific schedule of structural stiffness (typically more compliant than normal) it is possible to endow the end product with an aerothermodynamic performance characteristic that is different from the rigid-bladed counterpart. Presumably the resulting characteristic of the compliant turbomachine, within certain operational limits, is preferable to the more rigid turbomachine. Since there is no active controller involved in changing the blade geometry in response to changing operating conditions, the process may be thought of as passive control by aeroelastic tailoring. In fact, since the objective function is a change in the performance characteristic, the overarching process may be termed "aerothermoelastic tailoring": the principles of static aeroelasticity are applied to a turbomachine to change its performance in a thermodynamic sense.

The problem posed is extremely complex stemming from the need to couple the flow computation over a deformable surface with the structural dynamics computation of the body occupying the other side of the internal boundary. If no aeroelastic instabilities are encountered at a particular operating point the resulting configuration is fixed in time and the static solution (flow and structure) is a special case of the more general dynamic formulation.

In this situation it is useful to develop an approximate model for use in the rapid exploration of many static aeroelastic configurations, described by a host of controlling parameters. In this fashion those regions of parameter space showing potential advantages for the compliant-bladed turbomachine may then be computationally refined using more accurate, but more time-consuming, CFD coupled to discretized structural dynamic codes (FEM). It is the purpose of the present model to permit this rapid exploration of parameter space to delineate the more promising regions for the subsequent detailed studies.

I. AERODYNAMICS

For the rapid computation of the aerodynamic loads acting on airfoils in cascade, a two-dimensional incompressible potential flow model is chosen using a vortex sheet representation of the thin airfoils. Under these assumptions the upwash induced along the chord of the reference (zeroth) airfoil is given by

DTIC QUALITY INSPECTED 6

DTIC QUALITY INSPECTED 6	
Doc	Author's Name
A-1	

$$v(x) = \frac{1}{2\pi} \oint_0^1 g(\xi) \frac{d\xi}{\xi-x} + \frac{1}{2\pi} \int_0^1 g(\xi) F(\xi-x) d\xi \quad (1)$$

Here the variables x (and ξ) are chordwise coordinates made dimensionless w.r.t. the chord, c , and $g(x)$ is the unknown strength of the vortex sheet representing each airfoil. The first integral represents the contribution of the reference airfoil and the second integral represents the contribution from the remainder of the cascade through the "influence function" F under the assumption that every blade has the identical distribution $g(x)$.

Figure 1 shows the geometric configuration under analysis. Eq (1) is derived in Appendix A, as well as the inversion of the integral equation for the solution $g(x)$ under appropriate non-penetration boundary conditions and specified vector mean velocity of flow (V, α) .

The solution is obtained under the classical thin airfoil coordinate transformation

$$x = \frac{1}{2}(1 - \cos\theta) \quad (2)$$

$$\theta = \cos^{-1}(1 - 2x) \quad (2')$$

followed by a discretization of the transformed variable

$$\theta = \frac{\ell\pi}{n}, 0 \leq \ell \leq n \quad (3)$$

where n is the number of segments along the chord, or $n+1$ is the number of gridpoints at which the boundary conditions are satisfied and also the values of $g(\theta)$ are determined.

Using results first obtained in (Sisto, 1967) the distributed circulation can be expressed in the following form

$$g(x) = V \sqrt{\frac{1-x}{x}} f(x) \quad (4)$$

(thus rendering f dimensionless) and the solution for f in matrix format appears as follows:

$$f(\theta) = (A_p)^{-1} (U_p) \quad (5)$$

Here $\underline{A}$ is the $(n+1) \times (n+1)$ aerodynamic matrix and $\underline{U}$ is the upwash column matrix or vector.

The expression for the upwash U depends on the mean flow angle α and y' the camberline slope distribution. The expression for $\underline{A}$ is quite complex. It reflects the geometry of the cascade: s/c , β and, again, y' as a result of the method of inverting the fundamental integral equation. The relevant expressions appear as Eqs. (A2) (A14) and (A16) determining $\underline{A}$ and Eqs. (A14) (A19) (A20) and (A21) for $\underline{U}$. Both have been programmed for solution in the Matlab language and form part of the program test00.m and solve.m appearing as Appendix C.

II. STRUCTURE

Each of the identical airfoil sections comprising the cascade are modelled as two-dimensional plates of small curvature and variable thickness. Each plate may be thought of as the characteristic or representative section of an actual blade at 75 or 80 percent of the span of a typical cantilever blade. As such the elastic displacements may be expressed in terms of the bending moment, M according to

$$\frac{d^2y}{dx^2} = \frac{M(x)}{E(x)} \quad (6)$$

where E is the local stiffness related to Young's modulus, cross-sectional moment-of-inertia and Poisson's ratio if a solid orthotropic material is under consideration. For an aeroelastically tailored structure it is presumed that E is under the control of the designer within limits determined by strength considerations, stability, etc.

Recognizing that changes in stagger related to the untwist of a three-dimensional blade may be an important aeroelastic parameter, the two-dimensional counterpart model will be supported at the leading-edge and near the trailing edge by springs whose stiffness may be specified. In actual construction of a compliant blade these springs represent the restraint to bending provided by the spanwise spars required to support the more compliant material comprising the airfoil shaped profile.

The elastic bending moment $M(x)$ is therefore a function of the aerodynamic loading whereas the elastic displacements depend as well upon the strength of leading edge and rearward springs. The structural configuration is sketched in Fig. 2 where only a single airfoil need be shown, the others being identical. (The case of a shrouded cascade of airfoils would need additional modification of the model).

The aeroelastic coupling is provided by this aerodynamic loading acting on the compliant structure. (There may be a small amount of camber in the unstrained state upon which the elastic displacements are superposed). The aerodynamic load is provided by

$$dL = \zeta V dT = \zeta V g(x) d(cx)$$

$$dL = \zeta V^2 \sqrt{\frac{1-x}{x}} f(x) d(cx) \quad (7)$$

It is this loading which is applied to the plate; the details are developed in Appendix B. It may be noticed that dL has a square root singularity at the leading edge. However, under the coordinate transformation given by Eq (2), the singularity is integrable in the determination of lift and moment, viz

$$dL = qc(1+\cos\theta) f(\theta) d\theta \quad (7')$$

where

$$q = \frac{1}{2}\zeta V^2 \quad (8)$$

is the dynamic pressure based on the vector mean velocity, V . For known aerodynamic loading and structural stiffness, Eq (6) may be solved for the elastic displacements $Y(x)$. Subsequently the unstrained slopes may be augmented by the elastic slopes to obtain the overall slope distribution. This total slope function comprises the input to the aerodynamics subroutine for computation of the aerodynamic loading.

III. COMPUTATIONAL LOGIC

Conceptually the problem at hand is an eigenvalue problem similar in concept to the problem of wing divergence. However, owing to the complexity of the aerodynamic formulation, the method of solution is by iteration. An assumed blade shape produces a specific aerodynamic load. This load applied to the blade structure produces a specific schedule of displacements. These displacements define a new blade shape and completes the first loop of the iteration. If the iterations converge the stable static solution is obtained. If the iterations do not converge the condition of "divergence" states that the initial set of aerodynamic and structural parameters are in supercritical region of parameter space and no stable solution is obtainable.

The looping concept is illustrated in Fig. 3, a flow chart of the computational logic used to iterate the solution.

This chart, along with the actual program statements (expressed in MatLab language) comprise Appendix C.

IV. PARAMETRIC STUDY

A preliminary use of the completed program demonstrated was in the brief parametric study described below. The program solve.m (with input provided from a separate file test00.m) takes the converged result from the loop and computes a number of related parameters.

In order to assess the proximity to stall a very elementary approach is taken. The airfoil cascade is assumed to develop the deflection (lift) and drag provided by Howell's crude and early correlation of the cascaded airfoil. Thus the camber shape, stagger and solidity can be used to enter Howell's charts and predict the profile drag and lift (deflection) coefficients.

These data are developed for the representative section of single rotor compressor at the radius, or spanwise location, of the representative or characteristic section. For this study a design point wheel speed $u=468$ ft/sec, stagger angle $\beta=45^\circ$ and unit solidity are chosen and various graphs of loading coefficient Γ and rotor efficiency, η , can be plotted against flow coefficient, ϕ . These quantities derived from Howell's correlation are not presented as being quantitatively accurate. However, they do represent a consistent system of predicting drag-related performance para-meters, and the relative values are thought to indicate the proximity to stall, or extent of separation. Thus, the rotor efficiency at the same flow coefficient is a measure of stall mitigation if the efficiency of the compliant rotor is higher than the efficiency of the rigid rotor.

In Fig. 4 and the charts following the results of an alternative study are shown for the rigid and compliant rotors having the following characteristics

	Rigid Blades	Compliant Blades
Unstrained Camber, deg.	22	15
Unstrained Stagger, deg.	45	45
Solidity	1.0	1.0
of Elasticity	1.5×10^7 psi	1.0×10^6 psi
Design wheel speed, fps	468	468

In this study the compliant and rigid blades have been tailored so as to have virtually the same performance at the wheel speed, $u=468$ fps. There the system is assumed to have an important second operating condition at 10% overspeed, or $U=515$ fps. The performance at this second condition then discriminates between the two rotors vis a vis stall mitigation.

Since the aerodynamics formulation is inviscid in the interests of rapidity of computation, an alternative assessment of the propensity to stall is made on the basis of several measures: incidence and a leading edge loading parameter taken in conjunction with the camber angle. The last is expected to change significantly with changing operating conditions for the compliant blade.

(lower) for the compliant rotor at all flow coefficients in the overspeed condition. It should be recalled that both rigid and compliant rotors have been configured to have virtually identical geometry and performance at design speed.

The difference shown in Fig. 4 has not been optimized, or maximized. By reducing the rigidity of the compliant blades still further, and increasing their unstrained, or "as fabricated", camber, the leading edge loading differences noted in the figure could be magnified. However, this optimization process has not been carried forward at the time of completing this Final Report. Only the potential of the method has been demonstrated.

REFERENCES

Sisto, F., "Linearized Theory of Nonstationary Cascades at Fully Stalled or Supercavitated Conditions", Zeitschrift fur Angewandte Mathematik und Physik, pp. 531-541, 1967.

Howell, A.R., "Fluid Dynamics of Axial Compressors", Proc. Instn. Mech. Engrs., No. 153, p. 445, 1945

APPENDIX A

CASCADE AERODYNAMICS

Initially we wish to derive the steady aerodynamic load on a cascade of camberlines in incompressible flow. The usual thin airfoil assumption is made that the vorticities (representing the jump in tangential velocity across the airfoils) are distributed along the chordlines and the boundary condition are satisfied along these same straight segments.

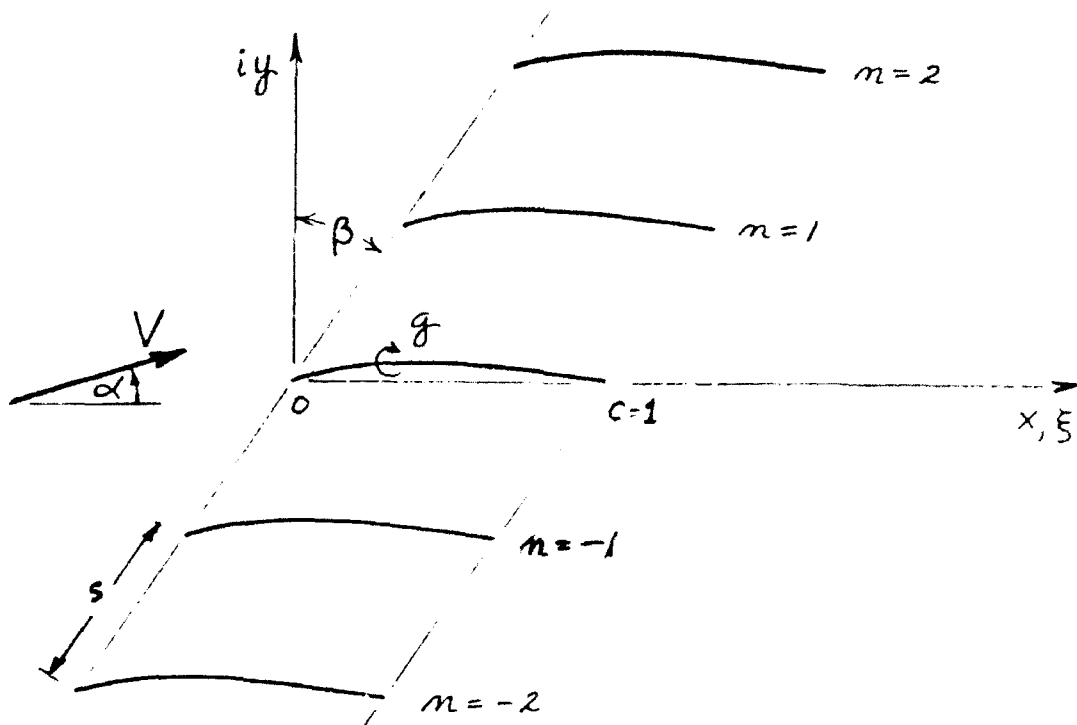


Fig. 1 Cascade Geometry

On the x -axis the velocities induced by the individual airfoils are given by*

$$u_0 - i\psi_0 = \frac{1}{2\pi i} \int_0^c g(\xi) \frac{d\xi}{\xi - x} \pm \frac{1}{2} g(x)$$

$$u_1 - i\psi_1 = \frac{1}{2\pi i} \int_{\Delta}^{\Delta+c} g(\xi_1) \frac{d\xi_1}{\xi_1 - x}$$

*The Biot-Savart Law expresses the conjugate velocity $u-iv$ at a general field point. Subsequent applications of the Plemelj formulae supply the limiting values of $u-iv$ as the camberline is approached from above (+) or below (-).

where $\xi_1 = \xi + \Delta$ and $\Delta = s e^{i(\frac{1}{2}\pi - \beta)}$ is the complex pitch, and so on for each airfoil.

Summing the contribution of all the (infinite number of) airfoils results in

$$\begin{aligned} u(x) - i v(x) &= \pm \frac{1}{2} g(x) + \frac{1}{2\pi i} \sum_{m=-\infty}^{+\infty} \int_{m\Delta}^{m\Delta+C} g(\xi_m) \frac{d\xi_m}{\xi_m - x}, \quad \xi_m = \xi + m\Delta \\ &= \pm \frac{1}{2} g(x) + \frac{1}{2\pi i} \sum_{m=-\infty}^{+\infty} \int_0^C g(\xi) \frac{d\xi}{\xi + m\Delta - x} \quad \because d\xi_m = d\xi \\ &= \pm \frac{1}{2} g(x) + \frac{1}{2\pi i} \int_0^C g(\xi) \left[\frac{\pi e^{i\beta}}{s} \sum_{m=-\infty}^{+\infty} \frac{1}{\frac{\pi e^{i\beta}}{s} (\xi - x) + i m \pi} \right] d\xi \end{aligned}$$

Now a well-known result is $\sum_{m=-\infty}^{+\infty} [\chi + i m \pi]^{-1} = \coth \chi$ so that

$$u(x) - i v(x) = \pm \frac{1}{2} g(x) + \frac{1}{2\pi i} \int_0^C g(\xi') \left\{ P \coth [P(\xi' - x')] \right\} d\xi'$$

where

$$x' = \frac{x}{C}, \quad \xi' = \frac{\xi}{C}.$$

Dropping the prime notation this may be put in the standard form

$$u(x) - i v(x) = \pm \frac{1}{2} g(x) + \frac{1}{2\pi i} \oint_0^C g(\xi) \frac{d\xi}{\xi - x} + \frac{1}{2\pi i} \int_0^C g(\xi) [F(\xi - x) + i G(\xi - x)] d\xi \quad (1)$$

where

$$P = \pi \exp(i\beta) c/s$$

$$F(\xi - x; P) + i G(\xi - x; P) \equiv P \coth [P(\xi - x)] - \frac{1}{\xi - x}$$

$$\text{and } F(0) = 0, G(0) = 0$$

Since g is purely real, separation of Eq. (2) into real and imaginary parts yields

$$u(x) = \pm \frac{1}{2} g(x) + \frac{1}{2\pi} \int_0^1 g(\xi) G(\xi-x) d\xi \quad (3)$$

$$v(x) = \frac{1}{2\pi} \oint_0^1 g(\xi) \frac{d\xi}{\xi-x} + \frac{1}{2\pi} \int_0^1 g(\xi) F(\xi-x) d\xi \quad (4)$$

The boundary condition expresses the fact that the flow direction must be tangential to the blade surface (camberline in this case):

$$\frac{dy}{dx} = \frac{v + V \sin \alpha}{u + V \cos \alpha} \quad (5)$$

where V and α refer to the mean flow and are defined in the figure*. Introduction of Equations (3) and (4) in Equation (5) results in the following integral equation

$$V(-\sin \alpha + \cos \alpha \frac{dy}{dx}) + \frac{1}{2\pi} \int_0^1 g(\xi) [-F(\xi-x) + \frac{dy}{dx} G(\xi-x)] d\xi = \frac{1}{2\pi} \oint_0^1 g(\xi) \frac{d\xi}{\xi-x} \quad (6)$$

Solution of Integral Equation

If the LHS of this equation is presumed defined, then the inversion is well known, and is given by

$$g(x) = -\frac{2}{\pi} \sqrt{\frac{1-x}{x}} \oint_0^1 \sqrt{\frac{\xi}{1-\xi}} \left\{ V[-\sin \alpha + y'(\xi) \cos \alpha] + \frac{1}{2\pi} \oint_0^1 g(\eta) [-F(\eta-\xi) + y'(\xi) G(\eta-\xi)] d\eta \right\} \frac{d\xi}{\xi}$$

which may be written

$$g(x) = g_s(x) + \frac{1}{\pi} \sqrt{\frac{1-x}{x}} \oint_0^1 g(\eta) K(x, \eta) d\eta \quad (7)$$

In the application of the B.C. the term $\pm \frac{1}{2} g(x)$ is omitted from $u(x)$ because of a fundamental approximation in thin airfoil theory*.

* Despite any local nonzero slope of the camberline the component u alone changes from one side of the boundary to the other. However, this jump in u is the local magnitude of the vorticity and physically the vorticity represents the jump in that component of the velocity which is parallel to the local boundary. Hence omitting the local vorticity (or strength of the vortex sheet) at the point where the B.C. are being applied, the remaining components of velocity must sum to a vector parallel to the local boundary. Another way of stating this condition is to say that if all components of the velocity, excepting the local vorticity, sum to a resultant which is parallel to the boundary, the subsequent addition of the vorticity component will not affect that parallelism (only the magnitude).

where g_s is the so-called single airfoil contribution

$$g_s \equiv \frac{2V}{\pi} \sqrt{\frac{1-x}{x}} \int_0^1 \sqrt{\frac{\xi}{1-\xi}} [\sin \alpha - y'(\xi) \cos \alpha] \frac{d\xi}{\xi-x} \quad (8)$$

and the kernel K is given by

$$K(x, \eta) \equiv -\frac{1}{\pi} \int_0^1 \sqrt{\frac{\xi}{1-\xi}} [-F(\eta-\xi) + y'(\xi) G(\eta-\xi)] \frac{d\xi}{\xi-x} \quad (9)$$

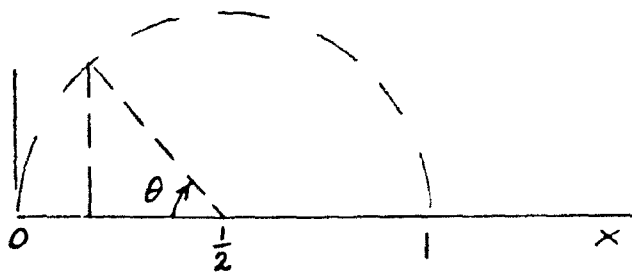
If a modified vorticity function, f , is defined by

$$Vf(x) \equiv g(x) \sqrt{\frac{x}{1-x}} \quad V f_s(x) \equiv g_s(x) \sqrt{\frac{x}{1-x}} \quad (10)$$

then the integral equation (7) takes the alternative form

$$f(x) = f_s(x) + \frac{1}{\pi} \int_0^1 \sqrt{\frac{1-\eta}{\eta}} f(\eta) K(x, \eta) d\eta \quad (11)$$

In order to obtain numerical solutions to this equation the following coordinate transformation, followed by discretization of the variables, is introduced.



$$x = \frac{1}{2}(1 - \cos \theta) \quad \theta = \frac{\ell \pi}{n} \quad (12a)$$

$$\eta = \frac{1}{2}(1 - \cos \phi) \quad \phi = \frac{\mu \pi}{n} \quad (12b)$$

$$\xi = \frac{1}{2}(1 - \cos \psi) \quad \psi = \frac{p \pi}{n} \quad (12c)$$

where ℓ , μ and p are positive integers less than or equal to n , or else zero.

Under this parameterization Eqs. (8) and (9) reduce to

$$f_3(\theta) = -\frac{2}{\pi} \int_0^\pi (1 - \cos \psi) [\sin \alpha - y'(\psi) \cos \alpha] \frac{d\psi}{\cos \psi - \cos \theta} \quad (8')$$

$$K(\theta, \phi) = \frac{1}{\pi} \int_0^\pi (1 - \cos \psi) \left[-F\left(\frac{1}{2} \cos \psi - \frac{1}{2} \cos \phi\right) + y'(\psi) G\left(\frac{1}{2} \cos \psi - \frac{1}{2} \cos \phi\right) \right] \times \\ \times \frac{d\psi}{\cos \psi - \cos \theta} \quad (9')$$

and the integral appearing in Eq. (11) reduces to

$$\int_0^1 \sqrt{\frac{1-\eta}{\eta}} f(\eta) K(x, \eta) d\eta = \frac{1}{2} \int_0^\pi (1 + \cos \psi) f(\psi) K(\theta, \psi) d\psi \quad (13)$$

If the numerators of the integrals in Eqs. (8') and (9') are assumed to be expanded in cosine series, it may be shown that these Glauert-type integrals may be approximated by sums

$$f_3(l) \cong -2 \sum_{p=0}^n \kappa_{lp} \left[1 - \cos \frac{p\pi}{n} \right] [\sin \alpha - y'(p) \cos \alpha] \quad (8'')$$

$$K(l, \mu) \cong \sum_{p=0}^n \kappa_{lp} \left[1 - \cos \frac{p\pi}{n} \right] \left[-F\left(\frac{1}{2} \cos \frac{p\pi}{n} - \frac{1}{2} \cos \frac{\mu\pi}{n}\right) + y'(p) \cdot \right. \\ \left. \times G\left(\frac{1}{2} \cos \frac{p\pi}{n} - \frac{1}{2} \cos \frac{\mu\pi}{n}\right) \right] \quad (9'')$$

where

$$\kappa_{lp} = \frac{1}{2} [2 - \delta_{p0} - \delta_{pn}] \cdot \begin{cases} \frac{1}{n} \frac{1 - (-1)^{l+p}}{\cos \frac{p\pi}{n} - \cos \frac{l\pi}{n}}, & p \neq l \\ 0 & p = l = 0 \text{ or } n \\ n & p = l = 0 \\ -n & p = l = n \end{cases} \quad (14)$$

and δ_{rs} is the Kronecker delta.

In Eq. (13), on the other hand, a direct application of the trapezoidal rule results in

$$\int_0^1 \sqrt{\frac{1-\eta}{\eta}} f(\eta) K(x, \eta) d\eta \approx \frac{\pi}{2m} \sum_{\mu=0}^m (1 - \frac{1}{2} \delta_{\mu 0} - \frac{1}{2} \delta_{\mu m}) (1 + \cos \frac{\mu \pi}{m}) f(\mu) K(\ell, \mu) \quad (13'')$$

When these individual results are combined in integral equation (11) there results a set of $n \times 1$ simultaneous equations, each equation corresponding to a particular value of the index ℓ . These may be expressed in matrix notations as

$$\underline{A} \underline{F} = \underline{U} \quad [A_{\ell \mu}] \{F_{\ell}\} = \{U_{\ell}\} \quad (15)$$

where $[A_{\ell \mu}]$ is a $n \times 1$ square matrix in which ℓ identifies the row and μ the column. The ℓ^{th} term is

$$A_{\ell \mu} = -\delta_{\mu \ell} + \frac{1}{4m} (2 - \delta_{\mu 0} - \delta_{\mu m}) (1 + \cos \frac{\mu \pi}{m}) \sum_{p=0}^m \pi_{\ell p} [1 - \cos \frac{p \pi}{m}] \times \\ \times \left[-F \left(\frac{1}{2} \cos \frac{p \pi}{m} - \frac{1}{2} \cos \frac{\mu \pi}{m} \right) + y'(p) G \left(\frac{1}{2} \cos \frac{p \pi}{m} - \frac{1}{2} \cos \frac{\mu \pi}{m} \right) \right] \quad (16)$$

Both $\{F\}$ and $\{U\}$ are column matrices with $n \times 1$ rows in which the ℓ^{th} rows are given, respectively, by

$$F_{\ell} = f(\ell) \quad (17)$$

$$U_{\ell} = 2^{1/2} \sin \alpha \sum_{p=0}^m \pi_{\ell p} [1 - \cos \frac{p \pi}{m}] - 2 \cos \alpha \sum_{p=0}^m \pi_{\ell p} [1 - \cos \frac{p \pi}{m}] y'(p) \quad (18)$$

Partitioning $\underline{U}$ into two parts

$$U_l = \sin \alpha U_l^{(1)} - \cos \alpha U_l^{(2)} \quad (19)$$

where

$$U_l^{(1)} = 2 \sum_{p=0}^m \mathcal{H}_{lp} \left[1 - \cos \frac{p\pi}{m} \right] = -2 \quad (20)$$

$$U_l^{(2)} = 2 \sum_{p=0}^m \mathcal{H}_{lp} \left[1 - \cos \frac{p\pi}{m} \right] y'(p) \quad (21)$$

The solution of Eq. (15) may be expressed

$$F_l = \sin \alpha F_l^{(1)} - \cos \alpha F_l^{(2)} \quad (22)$$

where

$$\underline{F}^{(1)} = \underline{A}^{-1} \underline{U}^{(1)} \quad (23)$$

$$\underline{F}^{(2)} = \underline{A}^{-1} \underline{U}^{(2)} \quad (24)$$

Cascade Properties

The circulation round one blade is

$$\Gamma = c \int_0^1 g(x) dx = cV \int_0^1 \sqrt{\frac{1-x}{x}} f(x) dx = cV \int_0^\pi \sqrt{\frac{1+\cos\theta}{1-\cos\theta}} f(\theta) \frac{1}{2} \sin\theta d\theta$$

$$\Gamma = \frac{1}{2} cV \int_0^\pi (1+\cos\theta) f(\theta) d\theta$$

$$\frac{\Gamma}{SV} \approx \frac{\pi}{2m} \left(\frac{c}{S} \right) \sum_{l=0}^m \left(1 - \frac{1}{2} \delta_{l0} - \frac{1}{2} \delta_{lm} \right) \left(1 + \cos \frac{l\pi}{m} \right) f(l)$$

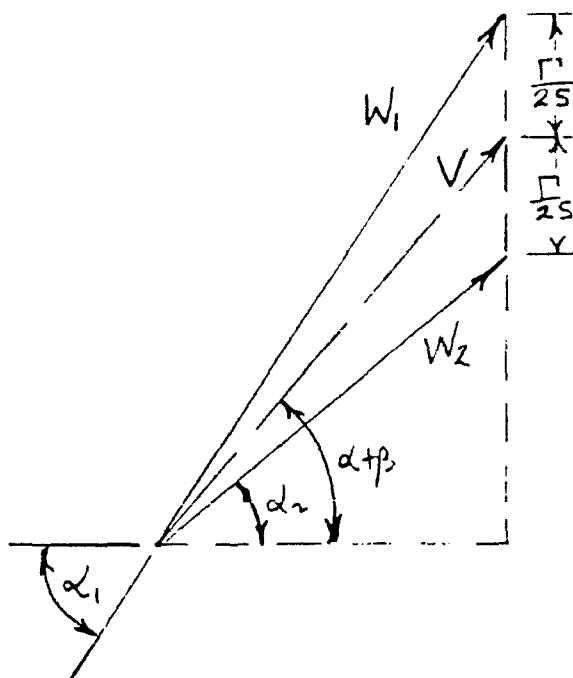
$$\frac{\Gamma}{SV} \approx Q \sin \alpha - R \cos \alpha \quad (25)$$

where

$$Q = \frac{\pi}{2m} \left(\frac{c}{s} \right) \sum_{l=0}^m \left(1 - \frac{1}{2} \delta_{l0} - \frac{1}{2} \delta_{lm} \right) \left(1 + \cos \frac{l\pi}{m} \right) A_l^{(1)} \quad (26)$$

$$R = \frac{\pi}{2m} \left(\frac{c}{s} \right) \sum_{l=0}^m \left(1 - \frac{1}{2} \delta_{l0} - \frac{1}{2} \delta_{lm} \right) \left(1 + \cos \frac{l\pi}{m} \right) A_l^{(2)} \quad (27)$$

Knowing the circulation allows the mean flow velocity V and direction α to be related to the upstream (or downstream) resultant flow. Arranging the cascade along a vertical axis, the various velocity vectors take on the aspect shown in the sketch.



From the geometry of this sketch we deduce

$$-\tan \alpha_2 = \tan \alpha_1 - 2 \tan (\alpha + \beta) \quad (28)$$

whereas the dimensionless circulation for one pitch distance along the cascade is given by

$$\frac{\Gamma}{sV} = \cos (\alpha + \beta) [\tan \alpha_1 - \tan \alpha_2] \quad (29)$$

Replacing $-\tan \alpha_2$ in Eq. (29) by Eq. (28) and then equating Eq. (25) to Eq. (29) results in

$$\tan \alpha = \frac{\frac{1}{2} R + \tan \alpha_1 \cos \beta - \sin \beta}{\frac{1}{2} Q + \tan \alpha_1 \sin \beta + \cos \beta} \quad (30)$$

Presuming that α may be determined by this means, it is then possible to express V in terms of the physically determinate W_1

$$V = \frac{W_1 \cos \alpha_1}{\cos(\alpha + \beta)} = \frac{W_1 \cos \alpha_1 \sqrt{1 + \tan^2 \alpha}}{\cos \beta - \sin \beta \tan \alpha} \quad (31)$$

This results in the interesting conclusion that a given cascade may be solved up to the determination of R and Q without regard to the upstream flow conditions. Subsequently the velocity triangles may be deduced for any upstream flow condition W_1, α_1 , by finding V, α from (30) and (31) and applying (28). In fact these manipulations may be used to derive Merchant's formula

$$\tan \alpha_2 = 2 \frac{R + Q \tan \beta}{2 \sec \beta + Q - R \tan \beta} + \frac{2 \sec \beta - Q + R \tan \beta}{2 \sec \beta + Q - R \tan \beta} \tan \alpha_1 \quad (32)$$

showing the linear dependence of $\tan \alpha_2$ on $\tan \alpha_1$.

APPENDIX B

PLATE MECHANICS

The moment acting on the spring supported plate, see Fig. 2, is expressed by first determining the aerodynamic moment about the leading edge

$$MOM_o = c \int_0^1 x dL = \frac{1}{2} q c^2 \int_0^\pi f(\psi) \sin^2 \psi d\psi \quad (B1)$$

This allows the computation of the rearward spring force immediately from equilibrium considerations

$$f_r = MOM_o / x_{er} \quad (B2)$$

Then computation of the aerodynamic lift

$$Lift = \int dL = q c \int_0^\pi f(\psi) (1 + \cos \psi) d\psi \quad (B3)$$

allows computation of the leading edge spring force

$$f_e = Lift - f_r$$

Knowing the spring constants k_l and k_r the leading edge and rearward support deflections may be computed in turn

$$Y(1,1) = f_e / k_e \quad (B4)$$

$$Y(e_r,1) = f_r / k_r \quad (B5)$$

where Y is the column matrix (vector) of elastic displacement dimensionless w.r.t. the chord and the indices are given in terms of $1 \leq \text{index} \leq m \equiv n+1$.

The bending moment distribution may now be computed rearward from the trailing edge according to

$$\begin{aligned} M &= \int_x^e (f - x) dL + U(e_r - e) f_r (x_r - x) \\ &= \frac{1}{2} q c^2 \int_0^\pi (1 + \cos \psi) (\cos \theta - \cos \psi) f(\psi) d\psi + U(e_r - e) f_r \left(\frac{\cos \theta - \cos \theta_r}{2} \right) \end{aligned} \quad (B6)$$

where U is the unit step function and f , ψ are dummy variables of integration analogous to x , θ .

In these integrals (B1, B6 and others) the integrals are determined at the discrete mesh points; hence the numerical integrations may be evaluated using either a trapezoidal rule, a Simpson's rule or other appropriate numerical technique.

For the approximation to the curvature $d^2y/dx^2 = cd^2Y/dx^2$ several trial models were evaluated. Eventually, in order to perform the integration with uniform steps in the angular variable (θ or ψ) the following approximation was developed.

With $Y=y/c$ and $X=x/c=\frac{1}{2}(1-\cos \theta)$ it is found that

$$\frac{d^2 y}{dx^2} = \frac{4}{c} \left[\sin \theta \frac{d^2 Y}{d\theta^2} - \cos \theta \frac{dY}{d\theta} \right] / \sin^3 \theta$$

and expressing the derivatives in terms of control differences,

$$\frac{c}{4} h^3 \frac{d^2 y}{dx^2} = D =$$

$$\left[h \sin(eu) (Y(e-1,1) - 2Y(e,1) + Y(e+1,1)) - \frac{1}{2} h^2 \cos(eu) (-Y(e-1,1) + Y(e+1,1)) \right] / \sin^3(eu)$$

where $u=\pi/m$ and a common factor h is retained to reduce the size of the elements in the D matrix.

In the application of this result it is found that the first and last rows in the $(n+1) \times (n+1)$ matrix are not required since the bending moment is zero at leading and trailing edges. Recognizing that two values of $Y(e,1)$ are known from previous equilibrium considerations, two columns may be removed and the useful D matrix may be reduced to $(n-1) \times (n-1)$ dimensions. In this reduced form the solution for the remaining Y terms may be obtained by inverting the reduced D matrix and solving for the remaining unknown values of Y

$$Y_r = \text{inv}(D_r) * \left(\frac{M}{E}\right)_r$$

where the subscript r refers to the reduced versions of these matrices.

Upon reassembling the full Y vector it is then possible to estimate the elastic slopes of the camberline using central differences* to correct the stagger angle and the camberline slopes for the next iteration

$$\beta_{\text{new}} = \beta_{\text{old}} + \arctan(Y(m,1) - Y(1,1))$$

$$Y^i \equiv Z$$

$$Z_{\text{new}} = \tan(\arctan Z_{\text{old}} + \arctan (Y(m,1) - Y(1,1)))$$

Adding the elastic slopes to the unstrained camberline slopes the total slopes are then made available as inputs to the aerodynamics subroutine for the next iteration.

*For leading edge and trailing edge resp. forward and backward differences are used.

APPENDIX B
PLATE MECHANICS

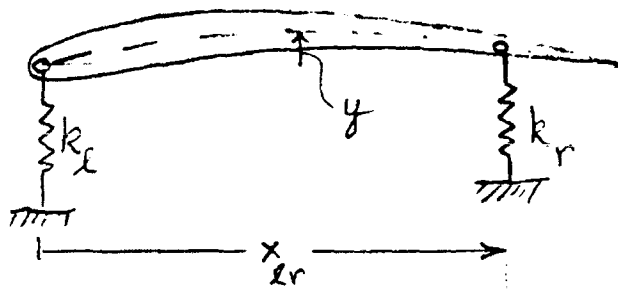
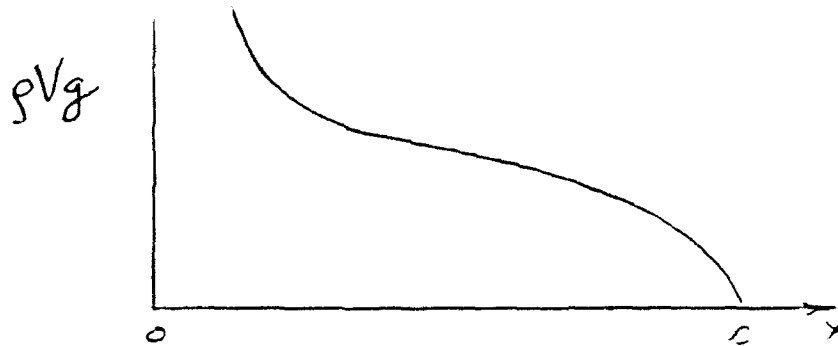


Fig. 2 Structural Model with Aerodynamic Load

APPENDIX C

Computer Programs

test 00.m

solve.m

in MatLab language

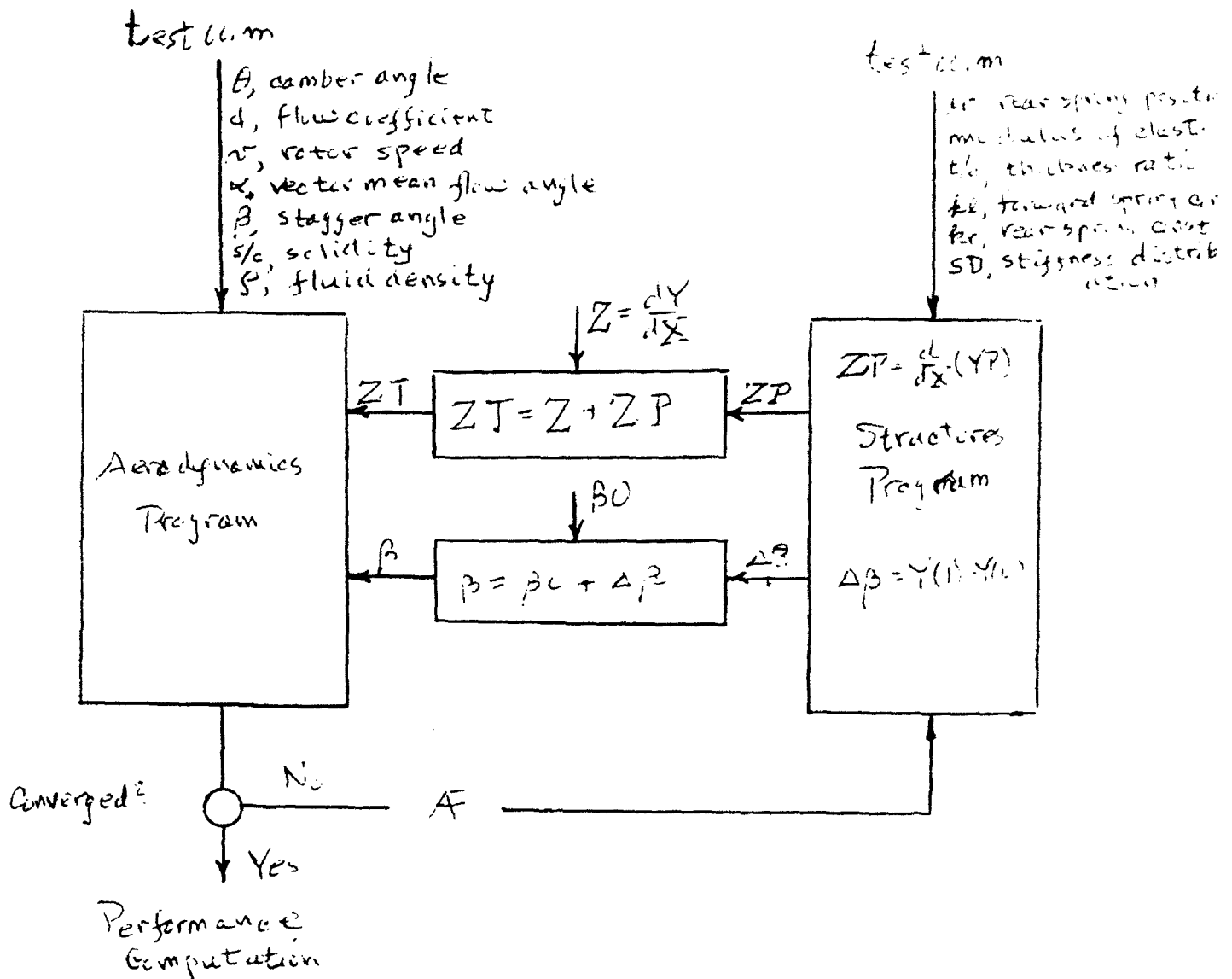


Fig. 3 Computation Flow Chart for solve.m Program (Matlab)

Rigid and Compliant Blade Comparison

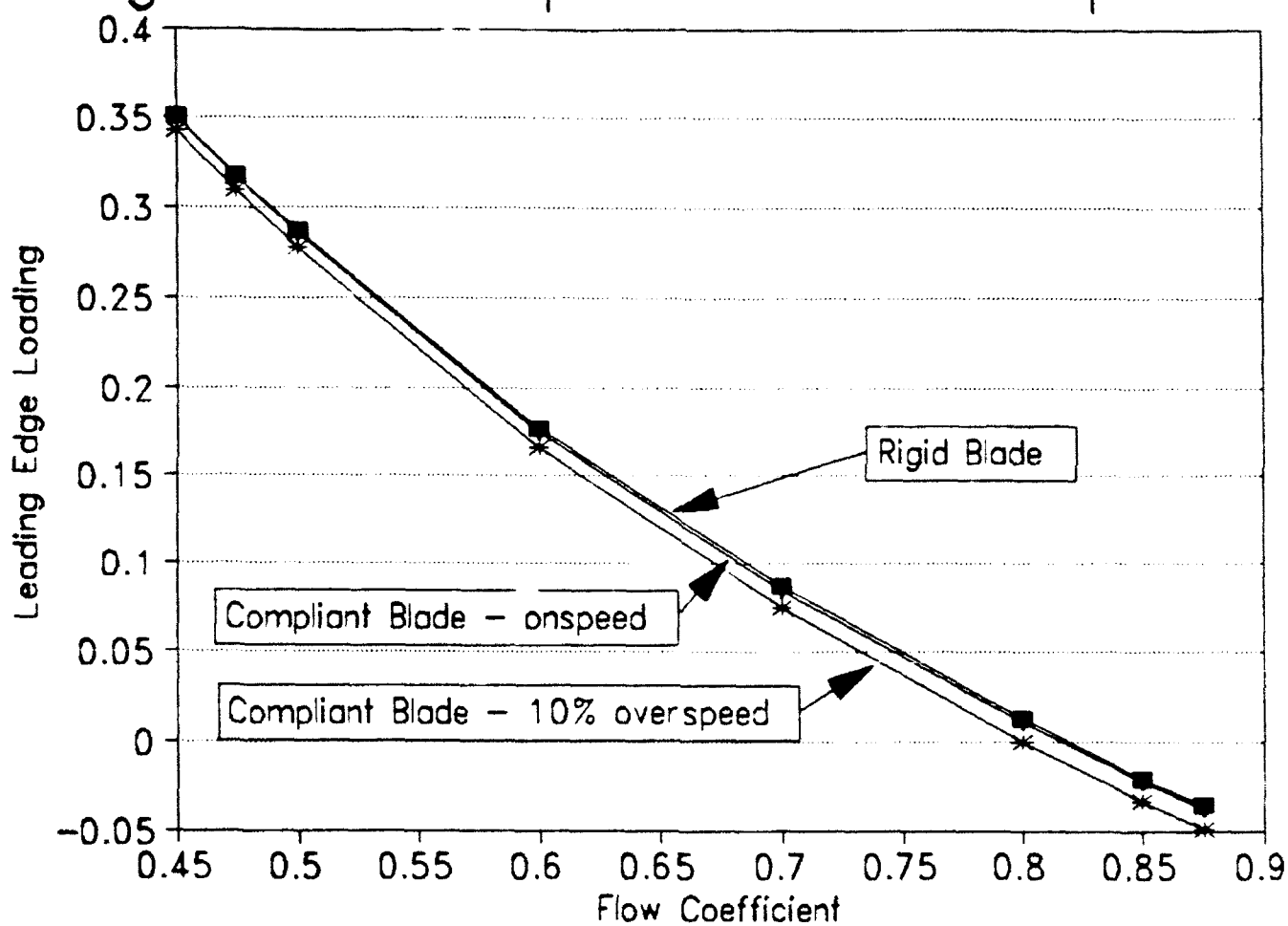


Fig. 4 Leading-Edge Loading Parameter for Rigid and Compliant-Bladed Rotors

```

1  ***** SOLVE.M *****
2  ***** CONSTANTS OUTSIDE OF ITERATION LOOPS *****
3
4  dbeta = 1;
5  dbetaold = 0;
6  beta = beta0;
7  alpha = alfa;
8  alfaiold = 0;
9  n = m - 1;
10  2P = zeros(m,1);
11  ZOLD = ones(m,1);
12  count = 0;
13  account = 0;
14  wps = 0.0001*m;
15  dif = 1;
16  h = pi/n;
17  u = (cos(h)-cos(2*h))/(1-cos(h));
18
19  for a=1:m
20      X(a,1) = 0.5*(1-cos((a-1)*h));
21      Y(a,1) = -0.5/tan(5*theta)*((0.5/(sin(5*theta)))^2-(X(a,1)-.5)^2)^.5;
22      Z(a,1) = (0.5-X(a,1))/Y(a,1)*.5/tan(5*theta);
23  end
24
25  % Initial total center shape
26
27  for a=1:m
28      S02(a,a) = (h^4*(1-5*theta))/tc2/2;
29  end
30
31  T = eye(m);
32  T(1,1) = 5000;
33  T(m,m) = 5000;
34
35  SIMP(1,1) = 2;
36  cc = 0;
37  for a=2:m
38      if cc = 0
39          SIMP(1,a) = (1-cos((a-1)*h))^4;
40          cc = 1;
41      else
42          SIMP(1,a) = (1-cos((a-1)*h))^2;
43          cc = 0;
44      end
45  end
46
47  SIMP = SIMP/3;
48
49  for a=1:m
50      for b=1:m
51          ab = cos((a-1)*h);
52          bb = cos((b-1)*h);
53          if a=b
54              K(a,b) = 0;
55          else
56              K(a,b) = (1-(1-ab)*(1-bb))/tc*(bb-bbald);
57          end
58      end
59  end
60
61  K(a,b) = 0;
62
63  % K = product of K and trapezoidal matrix
64  % G matrix
65  for a=1:m
66      for b=1:m
67          G(a,b) = (1-cos((a-1)*h))^2;
68          G(b,a) = (1-cos((b-1)*h))^2;
69          G(a,a) = 5000;
70          G(b,b) = 5000;
71          G(a,b) = (1-cos((a-1)*h))^4;
72      end
73  end

```

```

72  cc = 1;
73  else
74      S1(a,b) = (ab-bb)*ca)/2/3;
75      cc = 0;
76  end
77
78  % Special Simpson's coefficient
79  for a=2:(n-1)
80      cc = 0;
81      for b=(n-1):n
82          sb = (sin((b-1)*h))^2;
83          cb = cos((b-1)*h);
84          ca = cos((a-1)*h);
85          if cc = 0
86              S1(a,b) = (sb-bb)*ca)/5/4;
87              cc = 1;
88          else
89              S1(a,b) = (sb-bb)*ca);
90              cc = 0;
91          end
92      end
93  end
94
95  a = n-2;
96  b = n-3;
97  sb = (sin((b-1)*h))^2;
98  cb = cos((b-1)*h);
99  ca = cos((a-1)*h);
100  S1(m,(n-1)) = (sb-bb)*ca)/12;
101
102  % S2 matrix
103  cc = 0;
104  for b=2:n
105      sb = (sin((b-1)*h))^2;
106      cb = cos((b-1)*h);
107      for a=2:n
108          if cc = 0
109              S2(a,b) = (ca-cb)*(sb-bb)/4/3;
110              cc = 1;
111          else
112              S2(a,b) = (ca-cb)*(sb-bb)/2/3;
113              cc = 0;
114          end
115      end
116  end
117
118  S = S1 + S2;
119
120  B = zeros(m-2,m);
121  for a=2:n
122      for b=2:n
123          B(a,b) = 0.5*(sin((n-1)*h))^2*(1-cos((a-1)*h))^2/3;
124          B(a,b) = 0.5*(sin((n-1)*h))^2*(1-cos((b-1)*h))^2/3;
125          B(a,b) = 0.5*(sin((a-1)*h))^2*(1-cos((b-1)*h))^2/3;
126      end
127  end
128
129  dif = B(1,1);
130  dif = B(1:n-2,1:n-2);
131  dif = dif*(1-cos(h));
132  dif = dif*(1-cos(h));
133
134  ***** START ITERATION LOOP *****
135
136  while abs(dif) > 1e-6
137
138      % Initial beta for iteration
139      % Initial beta is the unstrained beta
140      % Alfa to be replaced by alfai
141      % Number of panels
142      % Initial strained slopes
143      % Old/previous strained slopes
144      % Iteration counter
145      % Alfa counter
146      % Iteration precision for slope summation
147      % Iteration precision for flow coeff
148      % Angular segment interval
149      % Airfoil shape defined by circular arc
150
151      % New stiffness matrix, m x m
152
153      % Trapezoidal matrix
154
155      % Simpson's 1/3 integration matrix
156
157      % Generate the K matrix
158
159      % K = product of K and trapezoidal matrix
160
161      % G matrix
162
163      % Mean velocity for flow at
164      % Dynamic pressure (psi)
165      % Lift coefficient
166

```



```

285 ind = a;
286 end
287 end
288 maxcamps = 2*(ind,1)*(X((ind,1),1)-X((ind,1),1))/2*(ind,1,1)-2*(ind,1,1));
289 maxcamps = X(ind,1)-maxcamps;
290
291 w1 = (1-sin(alfa+beta)*c/circ/s*(c/circ/s)^2/4)^.5; % Relative flow
292 w2 = (1-sin(alfa+beta)*c/circ/s*(c/circ/s)^2/4)^.5; % velocities
293
294 incld = beta-beta-gamma;
295 delta = beta2-beta-gamma2;
296 nbeta2 = (beta+gamma2)*180/pi/(s*.5)+0.23*(2*maxcamps)^2*cambor*180/pi;
297 nbeta2 = nbeta2/(1/(s*.5)-cambor*180/pi/500);
298
299 DEFTAB(1,1)=[0 .5 1.0 1.5 a];
300 DEFTAB(1,2)=[-10 52.5 39.5 31.7 0];
301 DEFTAB(3,1)=[0 43.3 33.0 26.8 0];
302 DEFTAB(4,1)=[10 36.8 27.9 22.75 0];
303 DEFTAB(5,1)=[30 26.25 19.7 16.2 0];
304 DEFTAB(6,1)=[50 16.4 12.5 10.2 0];
305 DEFTAB(7,1)=[70 7.5 6.15 4.75 0];
306 for c=2:4
307     for a=2:4
308         z=DEFTAB(c,a);
309         for b=2:4
310             z=z*(s-DEFTAB(1,b))/(DEFTAB(1,a)-DEFTAB(1,b));
311             z=z*(s-DEFTAB(1,b))/(DEFTAB(1,a)-DEFTAB(1,b));
312             end
313         end
314     end
315     DEFTAB(c,5)=DEFTAB(c,5)+z;
316 end
317 ndefl=0;
318 for a=2:7
319     z=DEFTAB(a,5);
320     for b=2:7
321         if a=b
322             z=z*(nbeta2-DEFTAB(b,1))/(DEFTAB(a,1)-DEFTAB(b,1));
323             end
324         end
325     end
326     ndefl=ndefl+z;
327 end
328 % Nominal incidence (nincld) and incidence parameter (iparam)
329 nincld = nbeta2 + ndefl - (gamma + beta)*180/pi;
330 iparam = (nincld*180/pi-nincld)/ndefl;
331
332 % Deflection parameter (eparam defl/ndefl)
333
334 aa = -.8461;
335 bb = .6680;
336 cc = 4.5431;
337 dd = -1.7091;
338 ee = 2.3254;
339 ff = 2.5892;
340 eparam = (aa*iparam+bb*(iparam^2)+cc*(iparam^3))
341 eparam = eparam/(1+dd*iparam+ee*(iparam^2)+ff*(iparam^3));
342
343 % profile drag coefficient
344
345 CDTAB(1,1)=[-62-52-42-32-22-12 0; 12-22-32-42-52-62];
346 CDTAB(1,2)=[.083-.082-.0115-.0214-.0168-.0168-.0174-.0182-.022-.046-.061-.121];
347 CDTAB(2,1)=[0];
348 CDTAB(2,2)=[0];
349 for a=1:13
350     for b=1:13
351         z=z*(iparam-CDTAB(b,1))/(CDTAB(a,1)-CDTAB(b,1));
352         z=z*(iparam-CDTAB(b,1))/(CDTAB(a,1)-CDTAB(b,1));
353     end
354 end

```

```

356 profcd=profcd+z;
357 end
358
359 rbeta2 = beta1 - ndefl*eparam*pi/180;
360 mbeta2 = atan(.5*(tan(beta1)+tan(rbeta2)));
361 alfa2 = atan(1/(flowco-tan(beta2)));
362 loadco = 2*stcos(mbeta2)*tan(beta2);
363 lifeco = 2*stcos(mbeta2)*(tan(beta1)-tan(rbeta2));
364 dragco = profcd + 0.009 + 0.018*(lifeco^3);
365 lossco = dragco/s/(cosmbeta2)^3;
366 roff = 1 - lossco*flowco^2/2/loadco;
367 % rm diary % Delete the old diary file
368 diary % Open the diary file for output
369 count
370 dif
371 delalfa1 alfa1 - alfa1old*180/pi
372 % Unstrained slopes
373 % Total slopes (strained + unstrained)
374 ZT
375 cambor = cambor*180/pi
376 stagr = beta*180/pi
377 delta = delta*180/pi
378 alfa = alfa*180/pi
379 alfa = alfa*180/pi
380 maxcamps
381 beta1 = beta1*180/pi
382 beta2 = beta2*180/pi
383 nincld = nincld*180/pi
384 devlath = devlath*180/pi
385 nbeta2
386 nincld
387 nincld
388 iparam
389 eparam
390 ndefl
391 rbeta2
392 alfa2
393 flowco
394 loadco
395 lifeco
396 dragco
397 lossco
398 roff
399 q
400 q
401 diary
402

```

% Close the diary file

```
% Aerodynamic input variables for circular arc unstrained camber
```

```
% Uniform stiffness
```

```
theta = 10*pi/180; % Camber section angle (unstrained)
flowcof = .4500; % Final flow coeff (target)
alpha = 11.4500*pi/180; % Alfa to start iteration
beta0 = 40*pi/180; % Unstrained stagger angle beta
a = 1.00; % a/c - solidity
n = 13; % Number of nodes
rho = 0.002377; % STD density for air (slug/ft.^3)
v = 573.5; % Rotor velocity (ft/sec)
```

```
% Structural input variables
```

```
z = 11; % Position of rear spring (1<ir<m)
modulus = 1.0e6; % Modulus of elasticity
c = 17^-3; % (t/c)^3
kf = 24000; % Forward spring constant
kr = 200; % Rear spring constant
c1 = [1.0;1.0;1.0;1.0;1.0;1.0;1.0;1.0;1.0;1.0;1.0;1.0;1.0]; % Stiffness distribution at m points
```

```
% Solve.m should be run next
```

```

***** SOLVE.M *****
***** CONSTANTS OUTSIDE OF ITERATION LOOPS *****

dbeta = 1;
dbetaold = 0;
beta = beta0;
alfa = alfa0;
alfaold = 0;
m = m - 1;
ZP = zeros(m,1);
ZOLD = ones(m,1);
count = 0;
account = 0;
eps = 0.0001*m;
dif = 1;
b = pi/n;
c = (cos(h)-cos(2*h))/(1-cos(h));

for a=1:m
    % Airfoil shape defined by circular arc
    x(a,1) = 0.5*(1-cos((a-1)*h));
    y(a,1) = -5/tan(.5*theta)+ (1.5/(sin(.5*theta)))^2-(x(a,1)-.5)^2)^.5;
    z(a,1) = (1.5-x(a,1))/(y(a,1)).5/tan(.5*theta);
end

% Initial total camber shape
for a=1:m
    S0/2(a,a)=(h*.43)/S0(a,1)/c/2;
end

% Trapezoidal matrix
I(1,1) = eye(m);
I(1,1) = .5000;
I(m,m) = .5000;
SIMP(1,1)=2;
cc=0;
for a=2:m
    if cc=0
        SIMP(1,a)=(1-cos((a-1)*h))^4;
        cc=1;
    else
        SIMP(1,a)=(1-cos((a-1)*h))^2;
        cc=0;
    end
end

% Generate the K matrix
for a=1:m
    for b=1:m
        ad=cos((a-1)*h);
        bd=cos((b-1)*h);
        if a=b
            ka,b=0;
        else
            K(a,b)=((1-(1)^*(a*b)))/(n*(bb-aa));
        end
    end
end
K(1,1)=0;
K(m,m)=0;
K=K';
% K = product of K and trapezoidal matrix
% S1 matrix
S1=zeros(m,m);
for a=1:m
    for b=(a+1):m
        cb=cos((b-1)*h);
        ca=1-cos((a-1)*h);
        if cc=0
            S1(a,b)=(sb-cb*ca)*4/3;

```

```

72 cc=1;
73 else
74     S1(a,b)=(sb-cb*ca)*2/3;
75 cc=0;
76 end
77 end
78 end
79 for a=2:2:(n-1)
80     cc=0;
81     for b=(n-1):n
82         sb=sin((b-1)*h)^2;
83         cb=1-cos((b-1)*h);
84         ca=1-cos((a-1)*h);
85         if cc=0
86             S1(a,b)=(sb-cb*ca)*5/4;
87             cc=1;
88         else
89             S1(a,b)=(sb-cb*ca);
90             cc=0;
91         end
92     end
93 end
94 a=n-2;
95 b=n-1;
96 sb=sin((b-1)*h)^2;
97 cb=1-cos((b-1)*h);
98 ca=1-cos((a-1)*h);
99 S1(n,(n-1))=(sb-cb*ca)/12;
100
101 S2=zeros(m,m);
102 for a=1:(lr-1)
103     cc=0;
104     for b=2:n
105         sb=sin((b-1)*h)^2;
106         cb=cos((a-1)*h);
107         cl=cos((lr)*h);
108         if cc=0
109             S2(a,b)=(ca-cl)*sb/(1-cl)*4/3;
110             cc=1;
111         else
112             S2(a,b)=(ca-cl)*sb/(1-cl)*2/3;
113             cc=0;
114         end
115     end
116 end
117 S=S1+S2;
118 D = zeros(m-2,m);
119 for a=2:n
120     lr=a-1;
121     D(lr,(a-1))-b/(sin(lr)*h)^2+.5*h^2*cos(lr*h)/(sin(lr)*h)^3;
122     D(lr,a)-2*b/(sin(lr)*h)^2;
123     D(lr,(a+1))-b/(sin(lr)*h)^2-.5*h^2*cos(lr*h)/(sin(lr)*h)^3;
124     end
125
126 d1=D(1,1);
127 d1r=D(1r-1,1r);
128 d2lr=D(1r-1,1r);
129 d3lr=D(1r,1r);
130 D(1,1)=1r;
131
132 ***** START ITERATION LOOP *****
133 while abs(dif) > .0001
134     % v=flowof/constal(a*beta);
135     % Dynamic pressure (pa)
136     % Ratio of q/E
137     % Mean velocity for flowof
138     % Dynamic pressure (pa)
139     % Ratio of q/E
140     % Mean velocity for flowof
141     % Dynamic pressure (pa)
142     % Ratio of q/E

```

```

43 while abs(delta-betaold) > .01
44 while sum(abs(2P-ZOLD)) > eps
45
46 deltaold = delta;
47 count = count + 1;
48 ZOLD = 2P;
49
50 P = pi*s*exp(beta*i);
51
52 for a=1:m
53     for b=1:m
54         aa=cos((a-1)*h);
55         bb=cos((b-1)*h);
56         if a==b
57             H(a,b)=0;
58         else
59             H(a,b) = P/(tanh(0.5*pi*(aa-bb))-2/(aa-bb));
60         end
61     end
62 end
63
64 % Generate the upwash matrix
65 for a=1:m
66     sm=0;
67     for b=1:m
68         sm=(K(a,b)*(1-cos((b-1)*h))*2T(b,1))+sm;
69     end
70     U(a,1)=-2*(sin(alpha)*(cos(alpha)*sm));
71 end
72
73 % Generate aerodynamic matrix
74 for a=1:m
75     for b=1:m
76         sm=0;
77         for c=1:m
78             sm=K(a,c)*(1-cos((c-1)*h))*(F(b,c)-2T(c,1))*G(b,c)+sm;
79         end
80         A(a,b)=(1*cos((b-1)*h))*sm/4/n;
81     end
82 end
83
84 A=A';
85 A=A-eye(m);
86
87 % Aerodynamic force matrix
88 FA = inv(A)*U;
89
90 % Moment about the leading edge
91 fr = sum(FA,1)/(1-cos((1r-1)*h));
92 % Force on rear spring
93 lift = q*h*(SIMP*FA);
94 % Lift generated by section
95 fr = lift - fr;
96
97 % LE deflection
98 Y(1r,1)=fr/kr;
99
100 RHS = qe*SD2*(S*FA);
101
102 RHS(1r,1)=RHS(1r,1)-d11*Y(1r,1);
103 RHS(1r,1)=RHS(1r,1)-d11*Y(1r,1);
104 RHS(1r,1)=RHS(1r,1)-d21*Y(1r,1);
105 RHS(1r,1)=RHS(1r,1)-d11*Y(1r,1);
106 RHS(1r,1)=RHS(1r,1)-d11*Y(1r,1);
107
108 YSM = inv(O)*RHS;
109
110 for a=1:(1r-1)
111     Y(a,1) = YSM((a-1),1);
112 end
113 for a=(1r+1):m
114     Y(a,1) = YSM((a-2),1);
115 end

```

```

214 delta = atan(Y(m,1))-Y(1,1);
215
216 beta = beta0 + delta;
217 alpha = alpha - delta;
218
219 ZP(1,1)=(2+u)*Y(1,1)/(1+u)-(1+u)*Y(2,1)/u+Y(3,1)/u/(1+u);
220 ZP(1,1)=2*ZP(1,1)/(1-cos(h));
221 for a=2:m
222     ZP(a,1)=(-Y((a-1),1)+Y((a+1),1))/h/sin((a-1)*h);
223 end
224 ZP(m,1)=(2+u)*Y(m,1)/(1+u)-(1+u)*Y((m-1),1)/u+Y((m-2),1)/u/(1+u);
225 ZP(m,1)=2*ZP(m,1)/(1-cos(h));
226
227 ZT = Z + ZP;
228
229 % Total camber shape - unstrained and
230 % strained shapes superpositioned
231 for a=1:m
232     ZT(a,1)=tan(tan(ZT(a,1)-delta));
233 end
234
235 % Close ZT Loop
236 % Close delta Loop
237
238 circ = 0.5*lift/q;
239 % Circulation Coeff
240 beta0 = atan(tan(alpha+beta)+0.5*circ/s/cos(alpha+beta));
241 beta = atan(tan(alpha+beta)-0.5*circ/s/cos(alpha+beta));
242 % Relative flow
243 flowco = 1/tan(beta);
244
245 account = account + 1;
246 dif = flowco - flowco;
247
248 if abs(alpha-oldalpha) < .00000001
249     dif = 0;
250 end
251
252 if account==1
253     slope = (dif-olddif)/(alpha-oldalpha);
254     alpha = alpha - dif/slope;
255     alphaold = alpha;
256     difold = dif - delta;
257     difold = dif;
258     deltaold = delta;
259     ZP = zeros(m,1);
260     ZOLD = ones(m,1);
261 end
262
263 if account > 1
264     alphaold = alpha;
265     alpha = alpha - dif/slope;
266     difold = dif - delta;
267     difold = dif;
268     deltaold = delta;
269     ZP = zeros(m,1);
270     ZOLD = ones(m,1);
271 end
272
273 print alpha*180/pi
274
275 % Close flowco loop
276
277 % OUTPUT FROM LOOP
278
279 gamma = atan(ZT(1,1));
280 gamma2 = atan(ZT(m,1));
281 camber = gamma1 - gamma2;
282
283 for a=1:m
284     if ZT(a,1) > 0

```


DISTRIBUTION LIST

Director (2)
Defense Tech Information Center
Cameron Station
Alexandria, VA 22314

Research Office (1)
Code 81
Naval Postgraduate School
Monterey, CA 93943

Library (2)
Code 52
Naval postgraduate School
Monterey, CA 93943

Professor Max F. Platzter (5)
Department of Aeronautics and
Astronautics
Naval Postgraduate School
Monterey, CA 93943

Professor F. Sisto (5)
Department of Mechanical Engineering
Stevens Institute of Technology
Hoboken, NJ 07030-5991