Exact Solutions in Oscillating Airfoil Theory

M. H. Williams*
Princeton University, Princeton, N.J.

Introduction

IT is well known that any planar lifting surface problem in linearized aerodynamics can be formulated as an integral equation relating surface pressure and downwash. For two-dimensional airfoils oscillating in an arbitrary subsonic parallel flowfield the integral equation reduces[2] to the type solved in Ref. 1. In the present Note the result obtained in Ref. 1 is reformulated to show that the pressure distribution induced by any deformation can be constructed from the particular solutions for heaving and pitching motions. Specific formulas are given for an oscillating control surface with a sealed gap.

The Integral Equation

The integral equation relating surface pressure and downwash is

$$\int_0^1 \mathrm{d}t\, K(t-x)P(t) = w(x) = \frac{\mathrm{d}f}{\mathrm{d}x} + ikf \tag{1}$$

where $P(x)$ is the loading and $f(x)$ is the spatial amplitude of the local displacement of the airfoil centerline which is assumed to vary harmonically in time with the (reduced) frequency k. The kernel function $K(t-x)$, which represents the downwash induced by a unit concentrated load at the point t, is determined by the undisturbed flowfield and the reduced frequency k. General rules for constructing the kernel for a particular flow are discussed in Refs. 2 and 7. The results given here are valid for any kernel associated with a generally subsonic parallel flow. Equations of the form of Eq. (1) also occur in the theory of vibrating cascades and the response of an airfoil to a sinusoidal gust.

If the undisturbed flow is asymmetric about the plane of the airfoil, source terms involving the thickness distribution may appear on the right-hand side of Eq. (1).[2] However, such terms appear in an unsteady problem only for pulsatile bodies and are neglected here.

The task confronting the aerodynamicist is to evaluate the loading P for a given motion w. The answer is not unique because an arbitrary circulatory motion can be imposed about the airfoil. Therefore the general loading solution [satisfying Eq. (1)] can be expressed as the sum of a particular solution [with circulation determined, for instance, by the Kutta condition, $P(1)=0$] and the homogeneous (circulatory) solution

$$P(x) = aP_0(x) + \int_0^1 \mathrm{d}t\, \Gamma(t,x)w(t) \tag{2}$$

where $P_0(x)$ is the loading for pure circulation and can, without loss of generality, be defined by the relations:

$$\int_0^1 \mathrm{d}t\, K(t-x)P_0(t) = 0\,, \qquad \int_0^1 \mathrm{d}x\, P_0(x) = 1 \tag{3}$$

Thus the quantity a in Eq. (2) is an arbitrary constant. The second term in Eq. (2), then, represents the "particular" solution and can be assumed to satisfy the Kutta condition [$\Gamma(t,1)=0$]. The function $\Gamma(t,x)$, known as the "resolvent kernel," is the pressure loading due to a unit, concentrated downwash at the point t. A complete solution of the lifting problem, Eq. (1), has been achieved when both $P_0(x)$ and $\Gamma(t,x)$ are known explicitly.

Such complete solutions are well known for an unbounded incompressible potential flow[5] and for an airfoil equidistant from two plane walls bounding an incompressible potential flow.[6] In general, though, purely numerical techniques have had to be used to solve Eq. (1), e.g., for subsonic, compressible potential flow.[8]

Recently a solution of the integral equation (1) has been found,[1] wherein the resolvent kernel is constructed explicitly in terms of the circulatory solution P_0 and one additional load distribution, $P_1(x)$, say, which solves Eq. (1) for unit downwash ($w=1$) and satisfies the Kutta condition, $P_1(1)=0$. The purpose of the present Note is to reformulate this result in a way which makes it more accessible to the aerodynamicist: specifically, we shall express the loading due to an arbitrary deformation explicitly in terms of the loadings due to the two rigid body motions (pitching and heaving) of the airfoil. Thus, giving the loadings for the elementary motions of the airfoil, loadings for any other deformation can be found by quadrature. This result is given in Eq. (12), the derivation of which will now be outlined.

In Ref. 1 it is shown that the resolvent kernel is

$$\Gamma(t,x) = P_0(1-t)P_1(x) - \frac{\partial}{\partial t}G(t,x) \tag{4}$$

where

$$G(t,x) = \int_0^t \mathrm{d}s[P_0(1-s)P_1(x-t+s) - P_0(x-t+s)P_1(1-s)] \tag{5}$$

$P_0(x)$ and $P_1(x)$ being the aforementioned solutions of Eq. (1) for $w=0$ and 1, respectively (it should be noted that $P_n(x) = q_n(1-x)$ in the notation of Ref. 1). This formula has been shown to reduce to the known results for steady subsonic unbounded potential flows[1] and unsteady incompressible unbounded potential flows.[4]

For simple heaving motion of amplitude h and pitching motion (about the leading edge), we have the downwashes $w = i\,k\,h$ and $w = \alpha(1+ikx)$, respectively. The loadings induced by these motions will be denoted by $P = hP_h$ and $P = \alpha P_\alpha$, which we take to satisfy the Kutta condition. It follows at once, from Eqs. (2) and (4-5), that

$$P_h(x) = i\,k\,P_1(x) \tag{6}$$

$$P_\alpha(x) = P_1(x)[1+ik(1-x_1)] + ik\int_0^x \mathrm{d}t[P_1(t) - L_1P_0(t)] \tag{7}$$

where

$$x_1 = \int_0^1 \mathrm{d}x\, x\, P_0(x),\quad L_1 = \int_0^1 \mathrm{d}x P_1(x) \tag{8}$$

These relations, which define the load distributions in the rigid body modes in terms of $P_{0,1}$, are easily inverted to give

$$P_1(x) = P_h(x)/i\,k \tag{9}$$

$$P_0(x) = [P_h(x) - \frac{\mathrm{d}}{\mathrm{d}x}(P_\alpha - bP_h)]/L_h \tag{10}$$

where

$$b = (P_\alpha(x)/P_h(x))\Big|_{x=0},\quad L_h = \int_0^1 \mathrm{d}x P_h(x) \tag{11}$$

Thus the complete solution to the lifting problem can be given directly in terms of the loads P_α and P_h. For an arbitrary

Received Dec. 13, 1976; revision recevied March 4, 1977.
Index category: Nonsteady Aerodynamics.
*Research Associate. Member AIAA.

downwash, w, the loading (satisfying the Kutta condition) becomes

$$P(x) = P_h(x)L/L_h + \frac{\mathrm{d}}{\mathrm{d}x}\int_0^1 \mathrm{d}t B(t,x)\mathrm{d}w(t)/\mathrm{d}t \qquad (12)$$

where

$$B(t,x) = \int_0^t \mathrm{d}s[P_h(1-s)P_\alpha(x-t+s) - P_\alpha(1-s)P_h(x-t+s)]/ikL_h \qquad (13)$$

and

$$L = \int_0^1 \mathrm{d}t w(t) P_h(1-t)/ik \qquad (14)$$

The loading for any given motion is easily calculated from Eqs. (10) and (12) once the solutions for pitch and heave have been found. We now give specific results for an oscillating control surface. This is a problem which is difficult to solve by standard numerical methods because it involves a discontinuous downwash distribution. The analytic solution, however, is particularly simple.

Oscillating Control Surface

For a control surface (with sealed gap) oscillating about its hinge, x_0, with amplitude β (the remainder of the airfoil held fixed) we have the downwash

$$w(x) = \beta H(x-x_0)[1+ik(x-x_0)] \qquad (15)$$

where H is the Heaviside step function. Let the loading induced by this motion be $\beta P_\beta(x)$ (taken to satisfy the Kutta condition). Then, from Eqs. (12) and (15) we have

$$P_\beta(x) = C_\beta(x_0)P_h(x) + D_\beta(x_0)P_\alpha(x) + \frac{\mathrm{d}}{\mathrm{d}x}B(x_0,x) + ikB(x_0,x) \qquad (10)$$

where

$$C_\beta(x_0) = \frac{1}{ikL_h}\int_0^{1-x_0} \mathrm{d}x[(1+ik(1-x_0-x))P_h(x) - ikP_\alpha(x)]$$

$$D_\beta(x_0) = \frac{1}{L_h}\int_0^{1-x_0} \mathrm{d}x P_h(x)$$

are constants depending on the hinge location, and $B(x_0,x)$ is defined by Eq. (13). It will be noted that at the hinge line

$$B(x_0,x_0) = \frac{1}{ikL_h}\int_0^{x_0} \mathrm{d}s[P_h(1-s)P_\alpha(s) - P_h(s)P_\alpha(1-s)] \qquad (17)$$

is bounded. Thus the pressure may have at most a weak singularity at the hinge line (which can be shown to be logarithmic if the kernel has a simple pole) arising from the $\mathrm{d}B(x_0,x)/\mathrm{d}x$ term in Eq. (16).

Moments

The total lift and pitching moment for any deformation follow directly from Eq. (12)

$$L = \int_0^1 \mathrm{d}x\, P(x) = \frac{1}{ik}\int_0^1 \mathrm{d}t w(1-t)P_h(t) \qquad (18)$$

$$M = \int_0^1 \mathrm{d}x\, xP(x) = -k^{-2}\int_0^1 \mathrm{d}t w(1-t) \times [(1+ik)P_h(t) - ikP_\alpha(t)] \qquad (19)$$

which are special cases of more general reverse flow theorems.[3] We note that Eq. (18) implies that the lift due to pitching is related to the lift and moment in heave

$$L_\alpha = \frac{1}{ik}(1+ik)L_h - M_h \qquad (20)$$

a relation which has long been recognized to hold for certain kernels K.[3]

For the oscillating control surface the flap lift and hinge moment can be easily evaluated by quadratures of Eq. (16). The resulting formulas, however, are complicated and are not given here.

Conclusions

It has been shown that the loading induced by an arbitrary harmonic deformation of an airfoil can be constructed exactly, using simple quadratures, if the loadings induced by the two rigid body motions of the airfoil are known. This can be done regardless of the structure of the undisturbed flow which only affects the form of the kernel in the governing integral equation.

In many particular cases it may be possible to relate (by symmetry arguments) the solutions for pitching and heaving, thereby reducing the number of "primitives," or functions to be evaluated numerically. An example of this is an unbounded subsonic flow, for which the kernel can be made antisymmetric by using a transformation developed in Ref. 4. For antisymmetric kernels the circulatory solution P_0 is the symmetric part of the unit downwash solution P_1.

For practical purposes the results given here are most easily applied if the load distributions for pitch and heave are represented as series in some suitable set of functions (such as the Chebyshev polynomials commonly used in airfoil theory). If this is done, then the resolvent kernel can be evaluated analytically and the load distribution for any given motion can be found by a single numerical quadrature.

While the present results are valid only for two-dimensional flows, they can easily be incorporated into strip theory calculations for three-dimensional wings. Whether relations similar to those given here may exist in general three-dimensional lifting surface theory is an intriguing question which merits further study.

Acknowledgments

This work was supported by NASA Grant 31-001-197, Ames Research Center.

References

[1]Williams, M. H., "The Resolvent of Singular Integral Equations" (to appear in the *Quarterly of Applied Mathematics*, April 1977).

[2]Williams, M. H., Chi, R., Ventres, C. S., and Dowell, E. H., "Effects of Inviscid Parallel Shear Flows on Steady and Unsteady Aerodynamics and Flutter," AIAA Paper 77-158, Los Angeles, Calif., 1977; submitted to *AIAA Journal*.

[3]Sears, W. R., Ed., "Reciprocity Relations and Reverse Flow Theorems in Aerodynamics," *General Theory of High Speed Aerodynamics*, Vol. VI, Princeton University Press, Princeton, N.J., 1954, pp. 314-327.

[4]Williams, M. H., "Generalized Theodorsen Solution for Singular Integral Equations of the Airfoil Class" (to be published in the *Quarterly of Applied Mathematics*).

[5]Theodorsen, T., "General Theory of Aerodynamic Instability and the Mechanism of Flutter," NACA TR 49, 1935.

[6]Timman, R., "The Aerodynamic Forces on an Oscillating Aerofoil between Two Parallel Walls," *Applied Scientific Research*, Vol. A, 3, 1951, pp. 31-57.

[7]Dowell, E. H. and Ventres, C. S., "Derivation of Aerodynamic Kernel Functions," *AIAA Journal*, Vol. 11, Nov. 1973, pp. 1586-1588.

[8]Landahl, M. T. and Stark, V. J. E., "Numerical Lifting Surface Theory—Problems and Progress," *AIAA Journal*, Vol. 6, Nov. 1968, pp. 2049-2060.

Robust Code for Constrained Optimization

Donald A. Pierre*
Montana State University, Bozeman, Mon.

Introduction

EASON and Fenton[1] and Pappas[3] have evaluated the performance of eighteen optimization codes for solving nonlinear programming problems. All codes were applied to ten problems having inequality constraints, and the results were used to rank the codes in terms of generality and efficiency. Of the eighteen codes, five used information concerning derivatives or derivative approximations; the best of these five codes solved only half of the problems under a specified set of test conditions. As a result, the gradient-based codes ranked low in terms of performance.

Among the recent advances in gradient-based methods have been the development of a variety of augmented Lagrangian methods (also called multiplier methods) which were originally introduced[4,5] for equality constrained problems, but have since been generalized in various ways[7] to account for inequality constraints as well. One of these generalizations[6] led to the development of an algorithm and a FORTRAN code LPNLP[7] which has proved to be very efficient in solving a variety of test problems and application problems; a method using gradient approximations with LPNLP has been evaluated in Ref. 8. In contrast to conventional gradient-based penalty function methods, the augmented Lagrangian methods do not require in general that penalty weights approach infinity as the solution progresses.

The purpose of this Note is to present the results of applying LPNLP to the ten problems cited in Ref. 1. The results show that LPNLP compares favorably with the best of the codes tested in Refs. 1 and 3.

Of interest are problems of the form

$$\text{maximize} f(x) \tag{1}$$

subject to

$$p_i(x) = a_i, \qquad i = 1,2,\ldots,NE < n \tag{2}$$

$$q_j(x) \leq b_j, \qquad j = 1,2,\ldots,NI \tag{3}$$

and

$$c_k \leq x_k \leq d_k, \qquad k = 1,2,\ldots,n \tag{4}$$

where x is an n vector $(x_1, x_2, \ldots, x_n)$; the a_i's, b_j's, c_k's, and d_k's are real constants and are supplied in the data set; and f, p_i's, and q_j's are functions that are assigned in subroutine FXNS. In the analytical gradient (AG) case, the first derivatives of these functions are assigned in subroutine GRAD; but in the discrete gradient (DG) case, subroutine GRAD is standardized[8] and need not be modified from one problem to the next. Because the ten problems in Ref. 1 are minimization problems, LPNLP makes use of the relationship $\min f = -\max(-f)$.

Received Jan. 17, 1977; revision received March 17, 1977.

Index categories: Analytical and Numerical Methods; Performance; Structural Design.

*Professor, Electrical Engineering Department.

Test Conditions and Results

The LPNLP code has two major phases: an unconstrained maximization phase and a multiplier/weight update phase. Four distinct modes of unconstrained maximization are available in LPNLP. Only one of these modes, the Davidon-Fletcher-Power mode with controlled reset (DFP/RESET mode), was used here. From previous experience[7,8] this mode has been the most consistent, but usually not the fastest. The reset rate parameter NSRCH was assigned the value $2n+1$ where n is the number of search variables.

Other control parameters in the code pertain to convergence tests and penalty weights. For all problem solutions, solution parameters ϵ_1, ϵ_2, and ϵ_3 were assigned values of 0.0001, 0.01, and 0.01, respectively. These settings resulted in excess of six place accuracy in the maximum value of f at convergence. In all but one case, penalty weights were assigned initial values of unity, were updated by a factor of four, and were held to a maximum of sixteen. All Lagrange multipliers were initialized at zero. For the discrete gradient approximation, a delta factor[8] of 10^{-8} was applied. The LPNLP code in Refs. 7 and 8 was used as given, with all print statements retained, but with output control flags set so that only initial conditions and terminal conditions were printed. Results were obtained on a time-shared XDX Sigma-7 computer with an automatic double-precision option (equivalent to 15 significant decimal digits of accuracy).

Table 1 lists the analytical gradient (AG) and discrete gradient (DG) results in terms of number of function evaluations. For problems 9 and 10, the problem statements were taken directly from FORTAN code in Ref. 2 (except that the objective functions were the negative of those in Ref. 2), and analytical gradients were not readily available. The NG column in Table 1 gives the number of calls to GRAD, NF gives the number of calls to FXNS, and the number of "equivalent" function evaluations NEF is calculated, NEF = n NG + NF. Note that 5 of 8 NEF/AG values are identical to corresponding NF/DG values, which suggests that solution paths under the two solution modes are very similar.

A summary of timing results is given in Table 2. The normalized time values in Table 2 were calculated by dividing the actual CPU times, obtained by loading and running executable load modules to convergence, by the corresponding CPU time for Colville's timing program.[9] Three of these times, for problems 1 (with 5 variables), 3 (5 variables), and 10 (4 variables), are smaller than corresponding times obtained by any of the eighteen codes tested in Refs. 1 and 3. Problem 4 is the only other problem with 4 variables; the remaining problems have either two or three variables.

In comparing AG timing values to corresponding DG values, the AG values have a slight advantage. This is more than offset, however, by the coding advantage that results from using a standard code for subroutine GRAD in the discrete gradient case. The timing results for problems 9 and 10 could be improved significantly if the numerous trigonometric terms used in FXNS were evaluated in the calling program and made available to FXNS by a labeled common block. For consistency with earlier results, however, this was not implemented.

Timing results merit closer examination. In a time-shared computer environment, the same executable load module can be run at two different times and require different CPU times; a difference of 5% in CPU times is not unusual. Furthermore, the compiling and linking times are not included in timing results. To illustrate the effect of these operations, problem 3 was run in several ways. First, the total normalized time required to compile a short calling program, to link all relevant compiled subroutines, and to run the program, was 0.1724 units. Next, the normalized time required to compile a short calling program, to link the same subroutines, and to simply exit from the calling program (prior to calling on LPNLP to solve the problem), was 0.1457 units – the difference between this value and the preceding one is 0.0267