Aeroelastic Design Optimization Program

A. J. Dodd,* K. E. Kadrinka,† M. J. Loikkanen,‡
B. A. Rommel,§ G. D. Sikes,¶ R. C Strong,** and T. J. Tzong††
McDonnell Douglas Corporation, Douglas Aircraft Company, Long Beach, California 90846

A major programming effort to produce an interdisciplinary optimization program for the static, dynamic, and aeroelastic analysis of finite element models has been underway at Douglas Aircraft Company since 1985. The data handling utilities, a high-level array and function manipulation language, and many basic analysis modules are in place and have been tested. The current status of the program is described together with test and production applications and plans for future developments.

Introduction

AN Aeroelastic Design Optimization Program (ADOP) is currently under development at the Douglas Aircraft Company. Full-scale development was authorized in 1985. The initial objective was to provide a capability for the rapid static, dynamic, and aeroelastic evaluation of finite element (FE) structural models of complete aircraft at the advanced design phase. The development team grew out of the company's unsuccessful proposal effort for the Air Force contract that resulted in the Automated Structural Optimization System (ASTROS).

Early in the development, it became clear that the major challenge was not methodology but automated multidisciplinary interfaces to achieve the efficiency required for rapid response at the advanced design and initial design phases. A major effort was expended in writing interface programs with established codes, including the CASES (Computer Aided Sizing and Evaluation System) program for advanced design weights and geometry, PATRAN, and the McDonnell Douglas Corporation (MDC) in-house modeler CGSA (Computer Graphics Structural Analysis) for FE model data and N5KM for doublet lattice aerodynamic influence coefficients (AIC).

It also became obvious that the data management system required for these efficient interfaces would show similar advantages when applied to larger-scale production analyses. Currently, a typical transport aircraft project is represented with models of up to 50,000 degrees of freedom (DOF) at the advanced design level and in excess of 250,000 DOF in the production phase. Models with 50,000 DOF were used for production detailed analysis a decade ago and were considered impractical 20 years ago. The program was therefore modified to allow for manipulation of very large arrays. This extended the development schedule but resulted in a system that is very general and is ideally suited to the ever-increasing demands on computing capacity.

Also adding to the need for greater computing capacity are advances in FE technology, which can now be used to solve problems in nonlinear structural dynamics, vibro-acoustic response, and fatigue and damage tolerance. Structural optimization requires multiple analyses of high-fidelity FE models that closely resemble the structure under design. In the near future, the problems of transonic aeroelasticity will be solved by merging the advances in computational fluid dynamics with those in FE strucutral mechanics.

Another major effort was undertaken to develop an interactive graphics capability based on the Silicon Graphics Iris 3030 workstation. This program is used as an "intermediate" processor for such tasks as displaying and debugging models, grouping or linking of elements, transferring mass data from the weights data base, displaying animated mode shapes, modal splining between aerodynamic and structural models, and displaying stresses, deformations, and design sensitivities. These capabilities greatly increase the overall efficiency of the ADOP system and provide impressive graphics for presentations and proposals.

The user interface with ADOP is via the ADOP Control Language (ACL), which is similar to the DMAP language of NASTRAN and the MAPOL language of ASTROS. High-level language concepts were developed at Douglas in the early 1960s for the FORMAT program but have been expanded and rewritten for ADOP, based on experience gained with NASTRAN and ASTROS.[1] The form of the ACL code is similar to FORTRAN; the main difference is ACL's use of array manipulation and its ability to call specialized ADOP modules. There are currently 79 such commands available, varying from simple matrix manipulations to data translators and complete analysis modules.

With the basic tools in place, the current effort is focused on two tasks. The first is the assembly of a static aeroelasticity module, using ACL. The initial objective is to provide an accurate capability at the advanced and preliminary design phases to examine three major aeroelastic design constraints—lift effectiveness, roll effectiveness, and divergence—and to generate the design sensitivities of these constraints so that least-weight structural changes can be made to a statically optimized FE model to satisfy any violated constraints. The second task is to incorporate the Design Optimization Tool (DOT) code as the initial numerical optimization approach, using a fully stressed design loop to satisfy static constraints and to provide the starting design for a sequence of dynamic, aeroelastic, and flutter optimizations. This development will use the ACL high-level language, which allows the optimization strategy to be rapidly reformulated or tailored for specific applications.

The ADOP development philosophy was to use existing technology where possible and to adapt existing code to the ADOP data handling and programming rules—although in most cases, it has been more efficient to recode rather than adapt existing programs. The ADOP data management system, the ACL language, the graphics module, and the major analysis modules currently in place or under development are described in this paper.

Received Dec. 26, 1989; revision received April 6, 1990; accepted for publication April 18, 1990.

*Senior Staff Engineer. Member AIAA.
†Consultant on loan from Rockwell International. Associate Fellow AIAA.
‡Principal Engineer Scientist.
§Principal Engineer Scientist. Member AIAA.
¶Engineer Scientist Specialist.
**Computing Analyst.
††Senior Engineer Scientist.

Disk and Core Management System

The ADOP Disk and Core Management System (ADACS) is a central input/output and memory manager. The features described below allow very large models to be routinely handled. A region of virtual memory is dynamically allocated at run time as a work area, resulting in a substantially smaller load module and avoiding recompilation of the source code when more memory is required. The work area is managed by a set of routines that partition it into named arrays. Arrays are referenced at the user level through ACL (see below) or by the programmer through the memory management routines. A catalog of the work area contents and attributes is stored in a set of COMMON blocks for central access. Each array may be divided into subarrays. This feature is particularly useful when one needs to write only a portion of an array to file, for example, when shifting converged eigenvectors from the array of approximations to file and replacing them with new vectors. ADACS dynamically allocates files used in ADOP, sparing the user the need for coding JCL. Each file has an associated name and a catalog describing its contents and attributes. Arrays are easily located through a catalog search, which is far more efficient than a search of the file records.

Sequential and direct-access file organizations are available through ADACS. Sequential files are beneficial when the contents are accessed in a sequential fashion and do not require large numbers of data retrievals. Sequential file contents may not be overwritten. Direct-access files are efficient when the contents are frequently accessed and overwritten. The cost of direct access in an IBM environment is the overhead required to format the files. Since IBM direct-access files and the ADOP catalogs associated with sequential and direct-access files are of fixed dimension, a provision was made for automatic allocation of "continuation" files. Each continuation file is related to the original file via a link list, and subsequent file use automatically accesses the continuation files.

Aeroelastic Design Optimization Program

The initial control language of ADOP was a derivative of FORMAT, developed at Douglas in the early 1960s. This was an interpretive language in that each instruction was read and executed before the next one was processed. The drawback of this approach is that semantic errors are detected only at run time so that after correction, the entire program must be rerun. In contrast, ACL is a high-level, compiler-based language.[2] The complier is composed of a parser and lexical analyzer. The entire instruction set is compiled and checked for semantic and logic errors and variable-type inconsistencies. Once error free, the "object code" is then used to govern program execution.

The syntax of ACL is very similar to FORTRAN and includes in-line procedures, logical looping and branching, the standard arithmetic operators, and direct matrix operators. This contrasts with FORTRAN, in which the default data entity is the scalar and the exception is the array. Like FORTRAN, ADOP analysis and function modules are accessed with a CALL statement.

Currently, 79 analysis and function modules exist in ADOP, including static analysis, eigenvalue analysis, fully stressed design, dynamic transient response, flutter analysis, and splining. Each of the modules generates related arrays, which are stored on file and may be used by other analysis modules or accessed by the user through ACL. Arrays may also be input via ACL as direct matrix input (DMI).

Two ACL examples follow. The first (Fig. 1) demonstrates logical looping and branching (arrays are referenced without regard to how they were created).

Figure 2 demonstrates file input/output and the use of an analysis module. The problem is to solve for a set of displacements, given a stiffness matrix and load vector. The pivot tolerance is input to the solution program.

```
IMPLICIT REAL GENERAL MATRIX (A-H, O-Z)
INTEGER I, II, IMAX, ICOUNT
LABEL LAB1, LAB2
" "
IMAX = 12
II = IMAX**2-IMAX/3
DO I = 1, II
  IF (I. EQ .3).OR.(I .EQ. 4) THEN GOTO LAB1
  ICOUNT = ICOUNT + 1
  IF (ICOUNT .GT. 9) THEN GOTO LAB2
  XMAT = XMAT * A
  LAB1:ENDDO
LAB2:END
```

Fig. 1 ACL logical looping and branching.

```
IMPLICIT GENERAL MATRIX (A-H, O-Z)
REAL PIVOT
" "
PIVOT = 1.E-6
" "
"****** OPEN STIFFNESS, LOAD
AND DISPLACEMENT FILES ******"
" "
OPEN STIFFDSK (TYPE = INPUT, &
DSN = 'USERID.STIFF.DATA')
OPEN LOADSK (TYPE = INPUT, &
DSN = 'USERID.LOAD.DATA')
OPEN DISPDSK (TYPE = OUTPUT, &
DSN = 'USERID.DISP.DATA')
" "
"****** READ STIFFNESS MATRIX AND
LOAD VECTOR FROM FILES ******"
READ (STIFFDSK) STIFF
READ (LOADDSK) RLOAD
DISP = SOLVE( STIFF, RLOAD, PIVOT )
WRITE (DISPDSK) DISP
PRINT DISP
END
```

Fig. 2 ACL analysis module call.

Interactive Graphics

The ADOP system does not include an FE modeling program; this is unnecessary in view of the number of highly developed modeling programs available. There are currently two operational modules capable of translating NASTRAN or Computer Aided Structural Design (CASD) bulk data files to ADOP format. CASD is the Douglas in-house static analysis system. NASTRAN or CASD bulk data files are generated by the in-house modeling program CGSA, and PATRAN is commonly used to generate NASTRAN models. These translator modules, although representing a large manpower investment, are of little interest technically and will not be considered in detail here.

In the planning phase for ADOP, it was decided that interactive graphics should be emphasized as an aid to automating the design optimization process. At that time (1983–1984), large-capacity stand-alone color graphics workstations were becoming available, and after some investigation, the Silicon Graphics (SG) 3030 Iris Graphics workstation was chosen. Coding of the graphics module (in C) was greatly simplified by use of the comprehensive SG firmware library of graphics functions, but at the cost of machine dependency. The code has subsequently been updated for the new SG 4D workstation. The workstations are linked to the mainframe for file transfers to and from the main ADOP system. A Seiko copier produces color screen images on paper or transparencies.

The primary functions of the graphics module are 1) display of the assembled FE model in ADOP format to allow visual checking for modeling errors and inconsistencies; 2) definition of mass and weights data; weights data are translated from the existing advanced design CASE model to the ADOP model and redefined as distributed or concentrated masses; masses from other sources can be added at this stage; these data may also be developed interactively via the ADOP mainframe system, but the process is much easier and faster with the aid of the graphics module; 3) display of animated mode shapes and

static deformations; 4) display of color contours of stresses, deformation, and sensitivities; 5) graphical splining display; this feature allows the user to verify that the splining of mode shapes between the structural and aerodynamics models is accurate; and 6) generation of data points for the aerodynamic model (program N5KDRAW).

Static Design Modules

The ADOP static design capability consists of modules for static analysis, stress recovery, model weight calculation, fully stressed design (FSD) resizing, and design sensitivity calculations. These modules are all new code. This was necessary in order to have full control of the source code to enable easy incorporation and modification of features such as finite elements, equation solvers, bandwidth minimizers, and coding updates for data restructuring, vectorizing, and other efficiency enhancements. Two important features of the ADOP static design capability are discussed below.

Finite Element Library

At present ADOP has 14 FE types: a 2-node axial bar, a 2-node beam, a 4-node qualdrilateral plane stress element, a 2-node spring, an 8-node solid element, a lumped mass element, a 4-node quadrilateral shell, a 3-node triangular shell, a 3-node triangular plane stress element, a quadrilateral shear panel, a 4-node composite quadrilateral shell, a 3-node composite triangular shell, and 3- and 4-node composite in-plane elements. The stiffness, stress, and mass matrices are computed for all element types for unit design variables and stored in a disk file in buffered format. Numerical integration is used throughout, except for the beam elements.

The beam and the axial bar elements are conventional types with constant cross section. In the near future, an offset capability will be added to improve the analysis of stringer-stiffened shells and plates. The 3-node in-plane and shell elements use the classical displacement formulation,[2] and the 4-node elements are based on the assumed stress hybrid approach.[4,5] In the hybrid formulation, two interpolation functions are defined for each element: the displacements are defined on the boundary, and the stresses are defined inside the element, giving better accuracy than displacement interpolation alone. For the membrane hybrid quadrilateral, there are eight nodal DOF and consequently eight displacement fields. Isoparametric shape functions are used for displacement interpolation. For stress interpolation, three constant and two linear fields (i.e., linear in the skewed isoparametric system) are used. The bending quadrilateral has 12 nodal DOF, one translation, and two rotations per node. Displacement interpolation is defined along the element boundary. A cubic polynomial defines the transverse deformation, a quadratic polynomial defines the tangent slope, and a linear polynomial defines the normal slope. Nine stress functions are used for the stress interpolation, all of the constant and linear terms are complete, and the symmetric second-order terms are included. The 3-node bending element employs a displacement formulation, but the interpolation has been smoothed to improve accuracy by taking the least-square fit of the curvatures.

There are four different laminated composite FE in ADOP. These are both 3-node and 4-node versions of in-plane (membrane) shell and bending shell elements.[6] For the in-plane elements, the bending action is discarded and, therefore, the input stacking sequence must be symmetric. The 4-node elements are based on the assumed stress hybrid FE formulation, and the 3-node elements use the classical displacement method. The bending stiffness for both element types is based on thin plate theory, i.e., the flexibility due to transverse shear displacement is ignored. The number of laminae is unrestricted. Each lamina can be orthotropic and arbitrarily oriented with respect to the reference axis. The reference surface can be offset from the midsurface of the laminate.

The mass matrix for all of the element types in ADOP is integrated numerically using displacement interpolation.

Fully Stressed Design

The RESIZE module in ADOP uses FSD to resize the FEs in the structural model. The process assumes that in an optimum structure, each FE is subject to the allowable stress under at least one of the applied load conditions. The element sizes are each "scaled" by the ratio of the actual element stress to the allowable stress. The new design variable is computed for all of the applied load conditions and checked against maximum and minimum size allowables, and the maximum permitted size is taken. For each new set of design variables, the stiffness arrays are updated and the model reanalyzed.

If the structure is statically determinate, the process will converge in one iteration to the exact optimum. In general, there is no guarantee that the solution will converge. The number of iterations required is not directly related to the number of elements, although convergence may depend on the degree of redundancy of the model. The model weight is calculated after each FSD iteration, and the process can be stopped after a specified number of iterations or continued until the rate of weight reduction falls below a specified percentage.

With the exception of the bending beam, all the elements in ADOP have only one design variable. For the beam elements, the user may choose from a small number of standard cross sections—circular, square, and rectangular—which can be defined completely by one variable. The specified allowable stress is compared to the equivalent stress at the critical points on the beam, leading to a sixth-order function of the design variable, found by the Newton-Raphson method.

Resizing of the membrane and shell elements is based on the von Mises-Hill criterion for allowable stress. The resize computations define new thicknesses at all four Gauss integration points, at the top and bottom surfaces of the shell, and for all the specified load cases. The maximum of all these is the new thickness. The actual stress values are functions of the in-plane and bending forces and the thickness of the element.

Structural Dynamics Modules

Modal Analysis

An accurate, efficient modal analysis methodology is the heart of any dynamic analysis system. Today, direct modal analysis of large unreduced structural models is possible with modern computer hardware, allowing more accurate representation of the dynamic behavior of the model and greatly reducing the labor involved in defining and interpreting specialized dynamic analysis models.

The large-order modal analysis module in ADOP includes an accelerated subspace iteration scheme with modifications to the numerical error correction technique and some improvements in computational efficiency.[7] An alternate Lanczos method formulation is currently under development but is not discussed here.

For large-order modal analysis, the numerical error growth caused by the amplification effect during iteration can become intolerable. A modified version of the accelerated subspace iteration method was developed to avoid this, including a novel monitoring technique to detect any regeneration of previously converged eigenvectors during iteration in the current subspace.[8] Other techniques, listed below, are incorporated to analyze large-order problems: 1) A blocking scheme partitions large matrices into small blocks, each of which fits into half of the available working memory.[9] 2) A nodal renumbering technique based on the wavefront method is used to minimize the size of the stiffness and consistent mass matrices.[10,11] This technique symbolically performs the solution and leads to an approximate minimum envelope size. 3) Rigid-body modes may be computed as standard structural modes or, to reduce the cost of the eigenvalue solution, they may be computed directly by balancing the inertial loadings to the structure without causing elastic deformation. 4) The classic thin plate-shell and membrane FE are rank deficient. A diagnostic sub-

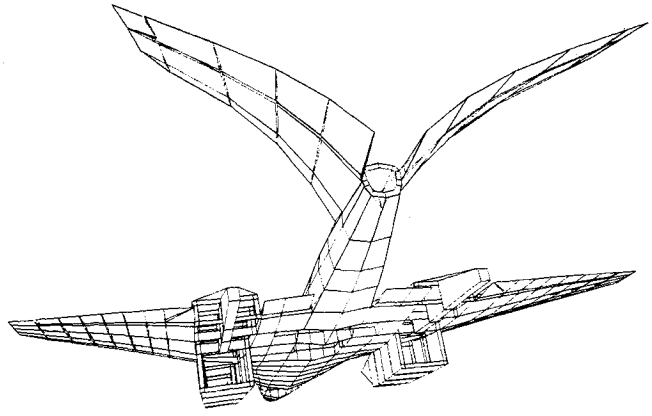

Fig. 3 Tail bending mode, supersonic research aircraft.

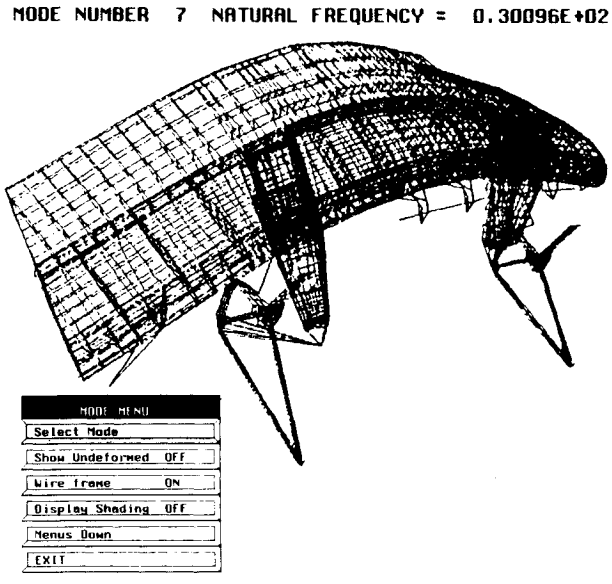

Fig. 4 Seventh mode shape of inboard flap.

routine in ADOP will automatically detect and remove any singularities due to missing stiffness. If a slightly curved surface is made of flat elements, some spurious eigenvalues may be present. A criterion is included in the diagnostic subroutine to avoid such eignvalues. 5) The user can select either a diagonal or consistent mass matrix. 6) The modal analysis module has a restart option. Modes need not be recomputed if the disk files from the earlier run are saved.

Modal Analysis Examples

Two numerical examples are presented to show the integrity of the modal analysis module. An advanced design model of a V-tailed supersonic research aircraft is shown (viewed from the rear) in Fig. 3. The model consists of 3,700 elements and 6,800 DOF. The first 10 vibration modes required 19 interations and no shift. Figure 3 illustrates the third mode, tail bending.

A 35,577-DOF transport aircraft inboard flap model was tested as a large size problem. The first 10 frequencies required 24 iterations. The results compare reasonably well with those computed by MSC/NASTRAN using Guyan reduction. Figure 4 illustrates the seventh mode shape. Figures 3 and 4 were generated by the animated modes option of the IMAGES module.

Dynamic Response

Two independent time-domain transient response procedures are available in ADOP, Newmark-Beta and modal superposition.[12-14] Newmark-Beta is a direct intergration method commonly used in dynamic transient response of structures. The procedure employed is implicit integration where the equilibrium equations of motion, below, are solved step-by-step in time:

$$[M]\ [\ddot{X}] + [C]\ [\dot{X}] + [K]\ [X] = [R] \qquad (1)$$

where $[M]$, $[C]$, $[K]$ are the structural mass, damping, and stiffness matrices; $[\ddot{X}]$, $[\dot{X}]$, $[X]$ are the structural acceleration, velocity, and displacement in the discrete coordinate system; and $[R]$ are time-dependent loadings.

The advantages of this method are that it is unconditionally stable and suitable for nonlinear structural dynamic analysis. The current scheme for Newmark-Beta integration in ADOP can handle matrices in both discrete and modal (generalized) coordinates. The matrices can be in the form of envelope, general (square), banded, or diagonal. A variation of the Newmark-Beta method, the Wilson $-\theta$ scheme, is also available.

The modal superposition method is ideal for linear structural systems. The basis of this approach is that the original set of equilibrium equations, in geometric coordinates, can be transformed to a reduced set of j uncoupled equations in modal (generalized) coordinates:

$$m_n\ddot{Y}_n + c_n\dot{Y}_n + k_nY_n = P_n(t) \qquad n = 1,\ldots,j \qquad (2)$$

where m_n, c_n, k_n are the generalized mass, damping, and stiffness coefficients for mode n; $\ddot{Y}_n$, $\dot{Y}_n$, Y_n are acceleration, velocity, and displacement in the generalized (modal) coordinates; and $P_n(t)$ are the generalized time-dependent loads for mode n.

These are second-order ordinary differential equations, which are easily solved with any time-dependent load condition. Once complete, the modal responses are combined according to mode superposition and the responses computed at structural node points. The advantage of this method is that no forward and backward solution for the entire system during the analysis is needed.

Table 1 Flutter mode types

Model	Young's modulus	Flutter speed, fps	Flutter frequency	Natural frequencies
Wind Tunnel[b]	—	495 (M = 0.45) (0.0%)	120	36, 210, 242
Beam-Stick model	—	509 (+2.83%)	134	—
NASTRAN[b] HA75E[a,c]	8.86E + 6[d]	483 (−2.42%)	113	34.7, 210, 255
ADOP[b]	10.2E + 6[e]	495.42 (+0.08%)	120.03	34.4, 210, 262

[a]Ref. *MSC/NASTRAN Handbook for Aeroelastic Analysis,* Vol. II, Nov. 1987, pp. 6.2-107.
[b]Aluminum alloy, measured material density = 0.097464 lb/in.^{3}
[c]Using NASTRAN KE method of flutter analysis.
[d]Young's modulus ratioed to match measured and NASTRAN-calculated natural frequency.
[e]Young's modulus ratioed to match measured and ADOP-calculated natural frequency.

Flutter and Sensitivities Modules

Flutter is a critical behavior parameter that must be considered very early in the design cycle. The complexity of modern aircraft structures and their models make automation of the flutter design procedure essential.

ADOP flutter constraints are calculated for each flutter mode and each hump mode using approximation techniques. As shown in Table 1, the aeroelastic modal behavior may be characterized by the behavior of the modes on the V-g diagram as stable modes, violent flutter modes, hump modes, or

stable incipient flutter modes. Hump modes and stable incipient flutter modes are similar except that the stable incipient modes do not flutter at the current design point. However, at a small distance from the current design point, these modes may flutter and become hump flutter modes. On the other hand, the hump flutter modes may become stable a short distance away from the current design point. If both behaviors are not tracked and constrained, the optimizer may end up hunting back and forth between design points without converging to any acceptable design.

To further complicate matters, the hump mode behavior is not truly a flutter characteristic but is a behavior of the flutter solution procedure. The K- and P-K methods will give different estimates of the hump mode and stable incipient mode behavior, although near the axis the damping is small and these behaviors will be similar. A maximum value for damping g^H_{max} is therefore required to describe the hump mode behavior. A default value ($g^H_{max} = 0.2$) is set but may be overridden. The parameter g^H_{max} is the upper limit for the classification of a hump mode. If damping at the top of the hump mode is greater than g^H_{max}, the mode may be classified as a violent flutter mode, and only the constraint on flutter velocity is computed. If damping at the top of the hump mode is less than g^H_{max}, design constraints are imposed on both flutter velocity and damping. Stable incipient modes have damping greater than $-g^H_{max}$ but less than zero. Modes which hump near the axis with a maximum damping less than $-g^H_{max}$ are treated as stable and are not constrained. To ensure stability, damping at the top of hump modes and stable incipient modes must be less than or equal to some required damping value $g_R = -0.02$, which the user may override. The constraints on flutter velocity are determined by requiring the flutter speed to be greater than a required value V_R.

The flutter optimization module in ADOP is being developed to perform 1) flutter analysis with a direct search for the flutter velocity and the matched air density, and 2) calculation of flutter design sensitivity for velocity and frequency. If the flutter performance is deficient, these design sensitivities will be processed by the ADOP optimizer to satisfy the flutter constraints with the minimum increase in weight. The DOT program is being installed for this purpose.[15]

Currently, only the K-method flutter analysis is available and only hysteretic-type damping, i.e., nonviscous damping, may be assumed. Hydraulic and other control elements are not allowed. A flutter analysis using the P-K method,[16] which does not have the above restrictions, is near completion.

The K-method flutter equation is written as

$$\{[\bar{K}] - \lambda_m([\bar{M}] + [\overline{AIC}])\}[\Phi_m] = 0 \tag{3}$$

where $[\bar{K}]$ and $[\bar{M}]$ are the modal stiffness and mass matrices, respectively. Hysteretic damping is incorporated in $[\bar{K}]$ using a modal damping ratio. $[AIC]$ is the aerodynamic matrix in modal DOF, which is obtained by premultiplying and postmultiplying the discrete aerodynamic matrix by the splined modal shapes from the structural nodal points to the aerodynamic grid points. The λ_m and Φ_m are the complex eigenvalue and eigenvector, respectively, and $\lambda_m = \omega^2/(1 + ig)$, where g is the damping of the system and ω is the circular frequency. Two surface splining methods, the Harder spline and a beam spline,[17] and a motion axis linear splining method adapted from Kroll and Hirayama[18] are currently available in ADOP.

The order of magnitude of damping for each mode changes with velocity or reduced frequency, making the manual search for the flutter point (zero damping point) troublesome. A direct numerical search for the flutter point using the information from a previous data point is therefore attractive. However, damping is not necessarily a single-valued function of velocity V in the K-method flutter analysis but becomes single-valued when reduced velocity $1/k$ is used, where k is the reduced frequency and is defined as $k = \omega c/V$, and c is half of the reference chord length. In the numerical search, the aerodynamic matrix should be a continuous function of the reduced frequency k. A table of aerodynamic data is computed for discrete k values. The continuous aerodynamic function is then obtained by interpolating from those discrete aerodynamic data using a standard cubic spline along the k axis.

The Laguerre method, which uses second-order derivatives, was used to perform the numerical search for the flutter point.[19] The associated formula is

$$(1/k)_{new} = (1/k)_{old} \pm g/\{(g_{,1/k})^2 - g g_{,1/k\,1/k}\}^{1/2} \tag{4}$$

where $g_{,1/k}$ and $g_{,1/k\,1/k}$ are the first and second derivatives of damping with respect to $1/k$, respectively. The numerical search procedure in the ADOP flutter module starts with a step-by-step calculation of damping along the $1/k$ axis. Every possible root is therefore located between to $1/k$ values. The Laguerre method or the Newton-Raphson method is then used to search for the root. A step-by-step calculation guarantees that a single root exists between two specific $1/k$ values. Root deflation is also employed to avoid repeated searching to the same root.

A search for the matched air density, which results in the lowest flutter velocity matching the airspeed, is necessary for building the flight envelope associated with the structure.[6] The search is similar to that for the flutter velocity; however, the step-by-step solution is avoided since the flutter velocity associated with the starting air density is found.

A simple flat-plate wing with 15 deg of sweepback is used as a flutter analysis test problem.[20] The wing was tested in a wind tunnel for flutter at subsonic speed at a Mach number of 0.45. The aerodynamic matrices were computed by the doublet lattice method.[21] A comparison of results from ADOP, MSC/NASTRAN, and the wind-tunnel test is shown in Fig. 5.

Approximate Flutter Reanalysis

For the initial development of the ADOP flutter optimization capability, the flutter speed and design derivatives are calculated exactly after each numerical search sequence. As shown in Fig. 6, a large number of parameter changes must be investigated in the flutter design of commercial transports, and this may require the use of approximate analysis methods during numerical search optimization. A second-order approximation technique for computing K-method flutter velocities can be written as follows:

$$V_F = V_{F0} + \{\nabla V_{F0}\}^T\{D - D_0\} + \tfrac{1}{2}\{D - D_0\}^T[H_{V_{F0}}]\{D - D_0\} \tag{5}$$

where V_F and V_{F0} are the estimated and original flutter velocities, and D and D_0 are the new original design vectors. The gradient vector $\{\nabla V_{F0}\}$ and the approximate Hessian matrix $[H_{V_{F0}}]$ may be computed using only the first design derivatives of the flutter eigenvalues and the reduced frequency for flut-

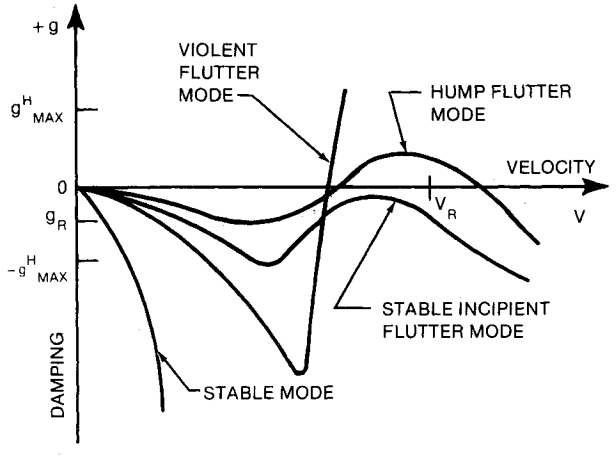

Fig. 5 Flutter analysis comparisons.

ter.[22] In a similar manner, the following equation may be used to approximate the hump mode damping factors:

$$g_H = g_{H0} + \{\nabla V_{g_{H0}}\}^T\{D - D_0\} + \tfrac{1}{2}\{D - D_0\}^T[H_{g_{H0}}]\{D - D_0\} \tag{6}$$

The numerical adequacy of this approximation will be examined more completely in future applications. The gradient vector and Hessian matrix may be computed as described by Rudisill, Bhatia, and Cooper.[19,23-25] This computation requires the AIC derivatives and the flutter eigenvector derivatives.[26-29] The AIC derivatives are computed from the spline fit in reduced frequency space. Experience indicates that 16 or more reduced frequency hard points are required to compute the AIC matrix and its derivatives, many more than was previously thought necessary.

Numerical search optimization requires that the constraint gradients as well as the flutter constraints be computed. Using the above approximations, a large number of flutter constraints and their design gradients can be computed rapidly for a number of parametric conditions of the aircraft, permitting simultaneous design optimization for all conditions. A cumulative quadratic flutter constraint similar to the one described by Haftka may be incorporated in future flutter optimization procedures.[26]

Aeroelasticity and Sensitivities Module

Development of the aeroelasticity analysis and sensitivities module is currently a major ADOP focus. This section, therefore, contains a more detailed presentation than in previous sections.

There have been numerous attempts to optimize cantilevered wing structures to satisfy both strength and flutter requirements. The first organized effort was reported in Ref. 30, and the first practical program was that of Siegel.[31] More advanced programs include TSO, which incorporates a modal approach,[32] and the Flutter and Strength Optimization Program (FASTOP).[33]

The problem of the unconstrained vehicle in free-free flight has been studied by Hancock,[34] Milne,[35,36] Wykes,[37] and Rodden and Love.[38] A modal approach was implemented at Douglas a decade ago by McGrew[39]; the ADOP aeroelastic module is an extension of this work, accounting for aerodynamic lift coefficients, divergence, and roll properties and their sensitivities.

Static Aeroelasticity Equations

For a system in static equilibrium, using the d'Alembert principle, where the inertia force has been replaced by an applied force, the static aeroelastic equations of equilibrium are written in generalized coordinates as

$$\begin{Bmatrix} \Delta\bar{F}_R \\ \Delta\bar{F}_E = 0 \end{Bmatrix} = [\bar{Z}]\begin{Bmatrix} p_R \\ p_E \end{Bmatrix} + \begin{Bmatrix} \bar{f}_R \\ \bar{f}_E \end{Bmatrix} \tag{7}$$

where subscript R denotes the rigid and E the elastic component of the principal generalized coordinate set p, and the generalized aeroelastic coefficient matrix $[\bar{Z}]$ consists of

$$\begin{bmatrix} \bar{Z}_{RR}\bar{Z}_{RE} \\ \bar{Z}_{ER}\bar{Z}_{EE} \end{bmatrix} = -q_{\text{Dyn}}\begin{bmatrix} \overline{AIC}_{RR}\overline{AIC}_{RE} \\ \overline{AIC}_{ER}\overline{AIC}_{EE} \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & \bar{K}_{EE} \end{bmatrix} \tag{8}$$

where q_{Dyn} is the dynamic pressure, and the shorthand notations are typically

$$[\overline{AIC}_{RE}] = [\phi_R]^T[AIC][\phi_E] \tag{9}$$

and

$$[\bar{K}_{EE}] = [\phi_E]^T[K][\phi_E] \tag{10}$$

where the generalized aerodynamic influence coefficient matrix $[\overline{AIC}]$ is evaluated at zero reduced frequency. The $[\phi]$ are both rigid-body and free-free elastic natural mode shapes obtained from the eigensolution of the dynamic system.

Elastic forces $\{\Delta\bar{F}_E\}$ are in internal equilibrium and equal to zero, and the rigid-body forces $\{\Delta\bar{F}_R\}$ are those necessary to keep the system in static equilibrium.

The generalized applied forces $\{\bar{f}\}$ are obtained from wind-tunnel measurements, airfoil camber data, deliberate wing surface deformations due to aeroelastic tailoring requirements, and roll helix angle. They are all obtained using artificial nonorthogonal shapes, identified by the letter J (for jig). The general expressions of these forces are

$$\{\bar{f}_R\} = [\phi_R]^T\{F_J\} = [\phi_R]^T[AIC]\{\phi_J\} = \{\bar{Z}_{RJ}\} \tag{11}$$

and

$$\{\bar{f}_E\} = [\phi_E]^T\{F_J\} = [\phi_E]^T[AIC]\{\phi_J\} = \{\bar{Z}_{EJ}\} \tag{12}$$

where $\{\phi_J\}$ is the nonorthogonal jig shape.

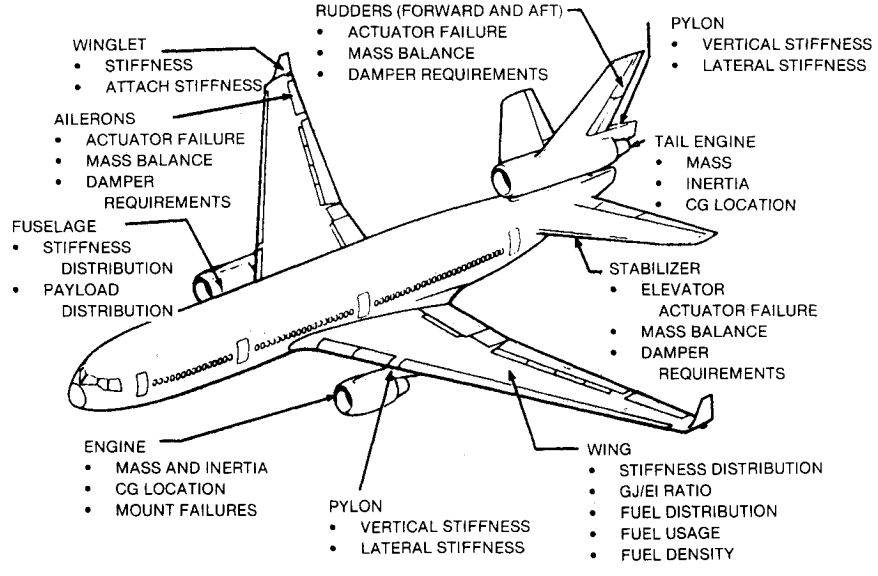

Fig. 6 Flutter behavior parameters.

Isolating the flexible generalized forces $\{p_E\}$ and then back-substituting them, the final operating form of the static aeroelastic equation is obtained:

$$\{\Delta\bar{F}_R\} = \left[[\bar{Z}_{RR}] - [\bar{Z}_{RE}][\bar{Z}_{EE}]^{-1}[\bar{Z}_{ER}] \right] \{p_R\} + \left\{ \{\bar{Z}_{RJ}\} - [\bar{Z}_{RE}][\bar{Z}_{EE}]^{-1} \{\bar{Z}_{EJ}\} \right\} \tag{13}$$

where $\{p_R\}$ are the rigid generalized coordinates, e.g., angle of attack α.

Derivation of the Aerodynamic Coefficients

The static aeroelastic equation of equilibrium in terms of aerodynamic coefficients can now be written as

$$\{\Delta\bar{F}_R\} = \begin{Bmatrix} \text{Aerodynamic Lift} \\ \text{Aerodynamic Moment} \end{Bmatrix} = \{C_J^{\text{Rigid}} + C_J^{\text{Flex}}\} + [C_{pR}^{\text{Flex}} + C_{pR}^{\text{Flex}}]\{p_R\} \tag{14}$$

Equating this with the external forces from the previous paragraph yields the relationships

$$[C_J^{\text{Total}}] = [\bar{Z}_{RJ}] - [\bar{Z}_{RE}]^T[\bar{Z}_{EE}]^{-1}[\bar{Z}_{EJ}] \tag{15}$$

and

$$[C_{pR}^{\text{Total}}] = [\bar{Z}_{RR}] - [\bar{Z}_{RE}]^T[\bar{Z}_{EE}]^{-1}[\bar{Z}_{ER}] \tag{16}$$

From here, the aerodynamic coefficients can be inferred directly. For example, the rigid and the elastic part of the aerodynamic lifting coefficients are

$$C_{L\alpha}^{\text{Rigid}} = \frac{\bar{Z}_{h\alpha}}{S_{\text{Ref}}\, q_{\text{Dyn}}} \tag{17}$$

and

$$C_{L\alpha}^{\text{Flex}} = -\frac{\{\bar{Z}_{hE}\}^T[\bar{Z}_{EE}]^{-1}\{\bar{Z}_{E\alpha}\}}{S_{\text{Ref}}\, q_{\text{Dyn}}} \tag{18}$$

where $\bar{Z}_{h\alpha}$ is a scalar value, S_{Ref} the reference area, L the lift, and h the plunge component.

Effectivity Terms and Their Sensitivities

Effectivity is defined as the ratio between total and rigid aerodynamic effects due to a unit deflection. It can be expressed in terms of aerodynamic coeffcients:

$$\eta = \frac{C^{\text{Total}}}{C^{\text{Rigid}}} = \frac{C^{\text{Rigid}} + C^{\text{Flex}}}{C^{\text{Rigid}}} = +\frac{C^{\text{Flex}}}{C^{\text{Rigid}}} \tag{19}$$

For example, the efficiency expression for the aerodynamic lift coefficient yields

$$\eta_{L\alpha} = 1 - \frac{\{\bar{Z}_{hE}\}^T[\bar{Z}_{EE}]^{-1}\{\bar{Z}_{E\alpha}\}}{\bar{Z}_{h\alpha}} \tag{20}$$

To sustain a steady roll, the rolling moment due to aileron deflection has to oppose an equal aerodynamic roll damping moment. The rolling performance characteristic is defined as the ratio between the rolling moment due to a unit aileron deflection and the aerodynamic damping moment due to a unit helix angle:

$$\eta_{1_R\delta} = \frac{\partial(pl/V)}{\partial\delta} = -\frac{C_{lR\delta}}{C_{lRp}} = -\frac{\bar{Z}_{y\delta} - \{\bar{Z}_{yE}\}^T[\bar{Z}_{EE}]^{-1}\{\bar{Z}_{E\delta}\}}{\bar{Z}_y \frac{py}{V} - \{\bar{Z}_{yE}\}^T[\bar{Z}_{EE}]^{-1}\{\bar{Z}_E \frac{py}{V}\}} \tag{21}$$

where 1_R is the rolling moment and δ is the control surface displacement. The unit helix angle produced by rolling of the lifting surface is py/V, where p indicates the rolling velocity, y the effective distance between the axis of rotation and the aerodynamic boxes, and V is the airstream velocity.

The exact sensitivity derivatives are lengthy but straightforward. A considerable simplification may be obtained by assuming that the initial set of natural modes and those after the optimization step differ only by an infinitesimal amount, so that the derivatives of the natural mode shapes, with respect to the design variables, vanish while the generalized stiffness matrix does not:

$$\frac{d\eta}{dw_i} = -\frac{1}{C^{\text{Rigid}}}\frac{dC^{\text{Flex}}}{dw_i} \tag{22}$$

where C^{Rigid} and C^{Flex} are the rigid and flexible portions of the aerodynamic coefficient, and w_i is the ith element design variable. Since

$$C^{\text{Flex}} = \{\bar{Z}_{hE}\}^T[\bar{Z}_{EE}]^{-1}\{\bar{Z}_{E\alpha}\} \tag{23}$$

its sensitivity derivative becomes

$$\frac{d\eta}{dw_i} = -\frac{1}{C^{\text{Rigid}}}\{\bar{Z}_{\text{left}}\}^T = \left(\frac{d}{dw_i}[\bar{Z}_{EE}]\right)\{\bar{Z}_{\text{right}}\} \tag{24}$$

where

$$\{\bar{Z}_{\text{left}}\}^T = \{\bar{Z}_{hE}\}^T[\bar{Z}_{EE}]^{-1} \tag{25}$$

$$\{\bar{Z}_{\text{right}}\} = [\bar{Z}_{EE}]^{-1}\{\bar{Z}_{E\alpha}\} \tag{26}$$

Noting that for linear elasticity

$$\frac{d}{dw_i}[\bar{Z}_{EE}] = \frac{d}{dw_i}[\bar{K}_{EE}] = [\phi_E]^T[k_i][\phi] \tag{27}$$

where $[k_i]$ is the element stiffness matrix of the ith element for unit design variable, the derivative of the lift efficiency becomes

$$\frac{d\eta_{L\alpha}}{dw_i} = \frac{1}{\bar{Z}_{h\alpha}}\{Q_h\}^T[k_i]\{\nabla_\alpha\} \tag{28}$$

where

$$\{\nabla_\alpha\} = [\phi_E][\bar{Z}_{EE}]^{-1}\{\bar{Z}_{E\alpha}\} \tag{29}$$

$$\{Q_h\}^T = \{\bar{Z}_{hE}\}^T[\bar{Z}_{EE}]^{-1}[\phi_E]^T \tag{30}$$

The expression for the derivative of the rolling performance characteristic follows a similar procedure.

Divergence and Its Sensitivity Expression

The eigenvalue form of the classical or cantilevered divergence problem is given by

$$\left[[\bar{K}_{EE}]^{-1}[\overline{AIC}_{EE}] - \lambda[I] \right]\{\phi_{\text{Div}}\} = \{0\} \tag{31}$$

where $\lambda = 1/q_{\text{Div}}$, and $\{\phi_{\text{Div}}\}$ is the divergence mode shape. The derivative of the above equation yields

$$\frac{d}{dw_i}\left([\bar{K}_{EE}]^{-1}[\overline{AIC}_{EE}]\right)\{\phi_{\text{Div}}^R\} = \frac{d}{dw_i}\lambda\{\phi_{\text{Div}}^R\} \tag{32}$$

where, as before, the right-side eigenvector is assumed to be independent of the design variables. To obtain a solution, the divergence eigenvalue problem is also solved with the resulting eigenvector on the left side $\{\phi_{\text{Div}}^L\}^T$. Premultiplying the derivative equation with the left divergence eigenvector $\{\phi_{\text{Div}}^L\}$ and using the orthonormality property between the left and the right eigenvectors yields the sensitivity expression for the divergence speed in modal coordinates:

$$\frac{dq_{\text{Div}}}{dw_i} = \frac{-q_{\text{Div}}^2}{w_i}\{\phi_{\text{Div}}^L\}^T[\bar{K}_{EE}^{-1}][k_i][\bar{K}_{EE}^{-1}][\overline{AIC}_{EE}]\{\phi_{\text{Div}}^R\} \tag{33}$$

Optimization Strategy

ADOP development is currently in the early stages of incorporating and testing various optimization features. The DOT program is being incorporated as an ADOP module and will be used to exercise the optimization loops and sensitivities modules. Previous experience with numerical optimization techniques indicates that there is an upper limit between 300 and 600 on the dimension of the design space. Above this limit, numerical accuracy reduces to the point where the search becomes meaningless, and computing time and cost becomes impractical. For large FE modes, elements are graphically grouped or linked to reduce the number of design variables.

For optimization of large statically loaded structures, it is felt that the FSD loop, with side constraints to satisfy design variable limits, is currently the most effective and practical approach. In the near future, modifications will be incorporated to handle thermal effects, size-dependent stress constraints (compression panel buckling, for example), and composite elements. Constraints on static deformations are rare in practice, but these will be handled with an optimality criteria algorithm. In the future, optimality criteria techniques will be investigated for specific application to optimization of aeroelastic constraints. It is not currently planned to incorporate shape or configuration optimization in ADOP.

The current optimization strategy is to perform a series of optimization studies rather than attempt the simultaneous satisfaction of multidisciplinary constraints. This reflects the current state of development of ADOP and also parallels the traditional sequence of events in the aerospace structural design and analysis process.

A typical optimization sequence will consist of an FSD sequence to establish a base static design. The final step in the FSD sequence will allow size increase only, to ensure positive stress margins of safety. Static deformation or frequency constraints can now be satisfied. At this point, an outer loop may be invoked to recalculate aerodynamic loads for the deformed model, and the entire process may be repeated. Element linking can now be done interactively, as required for the aeroelastic optimization. Various aeroelastic analysis and sensitivities calculation modules will be called at this stage to evaluate behavior such as flutter, control effectiveness, and wing effectiveness. These constraints, currently investigated individually, may in the future be combined for simultaneous optimization.

The optimization process described above will be initially set up as an ACL sequence. This has the advantage of being easily generated, modified, and tested. In the future it may be practical to "hard code" portions of the ACL sequence as ADOP modules.

Conclusions and Future Plans

With DOT incorporated as a module, all of the major planned features in ADOP will be in place. The development process has been more time-consuming and expensive than originally envisioned, largely due to the definition and coding of interface programs with existing codes, as discussed previously. The ADACS disk and memory management routines are now available as an IBM system library so that future developers can produce code that is automatically compatible with ADOP via ACL, thus eliminating the need for interface programs. It is not generally possible or practical to modify existing code to incorporate ADACS, and the original goal of incorporating existing code in ADOP has, with a few exceptions, not been realized.

Future plans include testing and quality assurance, production applications, documentation and training, and increased user-friendliness by improved input and output (including automated report writing). Proposed technical developments, based on the use of the ACL language, include active controls, vibro-acoustics, nonlinear dynamics, and new or improved interdisciplinary interfaces with supersonic aerodynamics, production loads programs, and advanced design configuration optimization and weights distribution codes.

As large interdisciplinary systems such as ADOP reach production status, various concerns arise that are not directly related to the the purely technical developments. These include development and management of multidisciplinary engineering data bases, integration with other engineering and related disciplines, and support and training of users.

The Packard Commission report of June 1986, "A Quest For Excellence," recommended a Total Quality Management (TQM) strategy for military procurement, with the multidisciplinary engineering aspects expanded from the traditional disciplines of aerodynamics, propulsion, and structures to include engineering system disciplines such as manufacturability, maintainability, life-cycle costs, and mission cost-effectiveness in the automated optimization process. This process is referred to as "Concurrent Engineering." ADOP can now be seen as merely a small part of a global design optimization system, but hopefully the lessons learned and data management features developed for ADOP can be applied to a future concurrent engineering system.

The challenge to the computer manufacturers is to provide ever more computing power at an affordable price, coupled with large working memories and very large permanent storage for the massive FE, CFD, and CAE/CAD/CAM/CALS data bases of the future.

Acknowledgments

Consultants to Douglas Aircraft Company, Universal Analytics, Inc., Playa Del Rey, California, under the direction of David L. Herendeen, have made major contributions to the ADOP code, including the ADOP control language compiler and a NASTRAN-to-ADOP bulk data file translator.

References

[1]Herendeen, D. L., Hoesly, R. L., Johnson, E. H., and Venkayya, V. B., "ASTROS – An Advanced Softwared Environment for Automated Design," AIAA Paper 86-0856, May 1986.

[2]Herendeen, D. L., and Sikes, G. D., "ADOP Executive System – Programmers' Manual," Universal Analytics, Inc., Playa Del Rey, CA, Rpt. UAI-TR88-0003, 1989.

[3]Zienkeiewicz, O. C., *The Finite Element Method,* McGraw-Hill Book Co., London, 1977.

[4]Pian, T. H. H., "Derivation of Element Stiffness Matrices by Assumed Stess Distribution," *AIAA Journal,* Vol. 2, No. 5, 1964, pp. 1333–1336.

[5]Pian, T. H. H., "Alternative Ways for Formulation of Hybrid Stress, Elements," *International Journal of Numerical Methods in Engineering*, Vol. 18, 1984, pp. 1679-1684.

[6]Loikkanen, M. J., "A 4-Node Thin Hybrid Finite Element," *Engineering Computations, U.K.*, Vol. 2, June 1985, pp. 151–154.

[7]Bathe, K. J., and Ramaswamy, S., "An Accelerated Subspace Iteration Method," *Journal of Computer Methods in Applied Mechanics and Engineering*, Vol. 23, 1989, pp. 313-331.

[8]Tzong, T. J., Sikes, G. D., and Loikkanen, M. J., "Large Order Modal Analysis in the Aeroelastic Design Optimization Program (ADOP)," McDonnell Douglas Corp., Long Beach, CA, Rpt. No. K4239, July 1989.

[9]George, A., and Liu, J. W., *Computer Solution of Large Sparse Positive Definite Systems,* Prentice-Hall, Inc., Englewood Cliffs, NJ, 1981.

[10]Hoit, M., and Wilson, E. L., "An Equation Numbering Algorithm Based on a Minimum Front Criteria," *Computers and Structures,* Vol. 16, No. 1-4, Pergamon Press Ltd., Great Britain, 1983, pp. 225-239.

[11]Irons, B., and Ahmad, S., *Techniques of Finite Elements*, Ellis Horwood Limited, Chichester, Great Britain, 1980.

[12]Clough, R. W., and Penzien, J., *Dynamics of Structures,* McGraw-Hill, New York, 1975.

[13]Bathe, K. J., and Wilson, E. L., *Numerical Methods in Finite Element Analysis*, Prentice-Hall, Inc., Englewood Cliffs, NJ 1976.

[14]Sikes, G. D., and Tzong, T. J., "Dynamic Transient Response of Structures in the Aeroelastic Design Optimization Program (ADOP)," McDonnell Douglas Corp., Long Beach, CA, Rpt. No. K4464, Oct. 1989.

[15]*COPES/DOT, A FORTRAN Control Program for Engineering Synthesis, Version 2.00*, Engineering Design Optimization Inc., Santa Barbara, CA, Feb. 1989.

[16]*MSC/NASTRAN Handbook for Aeroelastic Analysis, Version 65, Vol. 1,* Macneal-Schwendler Corp. Los Angeles, CA., Nov. 1987, pp 2.6-1-2.6-8.

[17]Harder, R. L., and Desmarais, R. N., "Interpolation Using Surface Splines," *Journal of Aircraft*, Vol. 9, No. 2, 1972, pp. 189-191.

[18]Kroll, R. I., and Hirayama, M. Y., "Modal Interpolation Program L 215 (INTERP)," NASA CR-2847, 1979.

[19]Bhatia, K. G., "An Automated Method for Determining the Flutter Velocity and the Matched Point," *Journal of Aircraft,* Vol. 11, No. 1, 1974, pp. 21-27.

[20]*MSC/NASTRAN Handbook for Aeroelastic Analysis, Version 65, Vol. II,* Macneal-Schwendler Corp., Los Angeles, CA, Nov. 1987, pp. 6.2-131-6.2-152.

[21]Kalman, T. P., and Giesing, J. P., "Subsonic Steady and Oscillatory Aerodynamics for Multiple Interfering Wings and Bodies," *Journal of Aircraft,* Vol. 26, No. 10, 1972, pp. 693-702.

[22]Rommel, B. A., "The Development of FAST-FLOW," *Aeroelastic Considerations in the Preliminary Design of Aircraft,* AGARD CP-354, Sept. 1983.

[23]Rudisill, C. S., and Bhatia, K. G., "Second Derivatives of the Flutter Velocity and the Optimization of Aircraft Structures," *AIAA Journal,* Vol. 10, No. 12, 1972, pp. 1569-1572.

[24]Rudisill, C. S., and Cooper, J. L., "An Automated Procedure for Determining the Flutter Velocity," *AIAA Journal,* Vol. 10, No. 7, 1973, pp. 1569-1572.

[25]Cooper, J. L., "A Curve Fitting Method for Solving the Flutter Equation," NASA CR-132629, Dec. 1972.

[26]Murthy, D. V., "Survey of Methods for Calculating Sensitivity of General Eigenproblems," NASA CP-2457 Sept. 1986, pp. 177-196.

[27]Ojalvo, I. U., "Modal Sensitivity for Structural Systems with Repeated Frequencies," *Sensitivity Analysis in Engineering,* NASA CP-2457 Sept. 1986, pp. 197-213.

[28]Ojalvo, I. U., "Gradients for Large Structural Models with Repeated Frequencies," Proceedings of SAE Aerospace Technology Conference and Exposition, Long Beach, CA, Oct. 1986.

[29]Dailey, R. L., "Eigenvector Derivatives with Repeated Eigenvalues," *AIAA Journal,* Vol. 27, No. 4, 1989, pp. 486-491.

[30]"Structural Design Application of Mathematical Programming Techniques," AGARD-AG-149-71, AGARDograph-149, N71-20128, Feb. 1971.

[31]Siegel, S., "A Flutter Optimization Program for Aircraft Structural Design," AIAA Paper 72-795, June 1972.

[32]McCullers, L. A., and Lynch, R.W., "Dynamic Characteristics of Advanced Filamentary Composite Structures," AFFEDL-TR-73-111, 1973.

[33]Markowitz, J., and Isakson, G., "FASTOP-3: A Strength, Deflection, and Flutter Optimization Program for Metallic and Composite Structure," AFFDL-TR-78-50, 1978.

[34]Hancock, G. J., "The Static Aeroelastic Deformation of Slender Configurations," *The Aeronautical Quarterly,* Aug. 1961, pp. 293-308.

[35]Milne, R. D., "Dynamics of the Deformable Aeroplane," R. & M. No. 3345, 1964

[36]Milne, R. D., "Some Remarks on the Dynamics of Deformable Bodies," *Journal of Aircraft,* Vol. 5, No. 11, 1968, pp. 556-558.

[37]Wykes, J. H., "Aerothermoelasticity: Its Impact on Stability and Control of Winged Aerospace Vehicles," *Journal of Aircraft*, Vol. 5, No. 11, 1968 pp. 517-526.

[38]Rodden, W. P., and Love, J. R., "Equations of Motions of a Quasisteady Flight Vehicle Utilizing Restrained Static Aeroelastic Characteristics," *Journal of Aircraft*, Vol. 22, No. 9, 1985, pp. 802-809.

[39]McGrew, J. A., "Flutter and Response Analysis Program C4EZ," McDonnell Douglas Corp., Long Beach, CA, Rpt. No. MDC—J6469, June 1978.