INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN ENGINEERING, VOL. 26, 293–311 (1988)

DERIVATIVES OF EIGENVALUES AND EIGENVECTORS OF A GENERAL COMPLEX MATRIX

DURBHA V. MURTHY* AND RAPHAEL T. HAFTKA

Department of Aerospace and Ocean Engineering, Virginia Polytechnic Institute and State University, Blacksburg, VA 24061, U.S.A.

SUMMARY

A survey of methods for sensitivity analysis of the algebraic eigenvalue problem for non-Hermitian matrices is presented. In addition, a modification of one method based on a better normalizing condition is proposed. Methods are classified as Direct or Adjoint and are evaluated for efficiency. Operation counts are presented in terms of matrix size, number of design variables and number of eigenvalues and eigenvectors of interest. The effect of the sparsity of the matrix and its derivatives is also considered, and typical solution times are given. General guidelines are established for the selection of the most efficient method.

INTRODUCTION

The behaviour of many physical systems is completely determined by the eigenvalues of the system. Variations in parameters lead to changes in these eigenvalues and hence in response characteristics. Thus, derivatives of eigenvalues and eigenvectors are of immense interest in several fields of physical sciences and engineering.

The derivatives (or synonymously, sensitivities) are of interest for a variety of uses. Design optimization is intimately connected with sensitivity analysis and the cost of calculating derivatives is the dominant contributor to the total cost in many optimization procedures. Most optimization algorithms require many analyses of the system and derivatives can often be used to approximate the eigenvalues and eigenvectors of a modified system and thus reduce the cost of reanalysis, especially in large systems. In addition, derivatives are very useful in design trend studies and for gaining understanding of and insight into the behaviour of physical systems. Finally, derivatives of eigenvalues are valuable in calculating the statistics of eigenvalue locations in probabilistic analyses.

The application of derivatives is not restricted to design-oriented activities. Sensitivity analysis is also playing an increasing role in determining the analytical model itself. In the areas of system identification and analytical model improvement using test results, sensitivity analysis is of growing importance. Much recent work in these fields is directly dependent on the calculation of eigensystem derivatives.

It has been found in certain cases that second order derivatives are very effective in improving accuracy of approximations[1–9] and efficiency of design.[3,7,8] Eigenvalues are usually non-linear functions of design parameters and a second order approximation offers a much wider range of applicability compared to the first order approximation. Intermediate variables which may improve the quality of first order approximations are not generally available for eigenvalue approximations. Also, some optimization algorithms require second order derivatives, and first

*Current addresses: Department of Mechanical Engineering. The University of Toledo, Toledo, 0443606, USA and NASA Lewis Research Center, Cleveland, OH44135, USA.

0029–5981/88/020293–19$09.50
© 1988 by John Wiley & Sons, Ltd.

Received 15 May 1986
Revised 5 January 1987

order derivatives of optimal solutions require second order derivatives of constraints.[9] The use of second derivatives can also greatly reduce the number of reanalyses required for the convergence of an optimization procedure.[7,10] Further, in certain optimization algorithms, second order approximations for eigenvalue constraints can drastically relax the move limits, thus achieving a nearly optimum trajectory, and can virtually eliminate the need for trial and error adjustment of move limits, thus improving the performance of the optimizer.[10] Looking at another aspect, in problems where instabilities are to be avoided, a first order calculation may completely fail to detect instabilities.[2] References 11 and 12 also offer examples of the usefulness of second order derivatives.

The problem of calculating the derivatives of symmetric and Hermitian eigenproblems is relatively simple and solution procedures are well established, see e.g. References 13–17. However, many physical problems give rise to non-self-adjoint formulations and thus lead to general matrices. An important example is aeroelastic stability which requires the solution of eigenproblems with complex, general and fully populated matrices. General systems are also obtained in damped structural systems and in network analysis and control system design, where the eigenvalues are usually called poles. The present study aims to present a comparative analysis of the various methods available for calculating the derivatives of the general eigenproblem and propose some modifications to existing techniques. A considerable amount of literature is available on the subject and a comparative analysis of the various methods will be of value for selecting the most efficient technique for a particular application. The purpose of this paper is to summarize the more efficient techniques proposed so far and to establish guidelines for the selection of the appropriate method for a given problem. Only the essentials of these methods are presented, with details referred to the original references. Attention is restricted to the general eigenproblem, and techniques that are useful only for the self-adjoint problem are not considered. The present discussion is limited to the case of distinct eigenvalues. The efficiency of the algorithms is examined assuming the system matrix to be fully populated as, for example, in aeroelastic or control systems. Thus, the comments on efficiency may not be applicable to the cases for which the system matrix is banded as in finite element vibration analyses.

Problem definition

The matrix eigenproblem is defined as follows:

$$\mathbf{A}\mathbf{u}^{(k)} = \lambda^{(k)}\mathbf{u}^{(k)} \tag{1}$$

and the corresponding adjoint problem is

$$\mathbf{v}^{(k)\mathrm{T}}\mathbf{A} = \lambda^{(k)}\mathbf{v}^{(k)\mathrm{T}} \tag{2}$$

where $\mathbf{A}$ is a general complex matrix of order n and $\lambda^{(k)}$, $\mathbf{u}^{(k)}$ and $\mathbf{v}^{(k)}$ are the kth eigenvalue and right and left eigenvectors respectively. The superscript T denotes the transpose. All eigenvalues are assumed to be distinct.

The matrix $\mathbf{A}$, and hence $\lambda^{(k)}$, $\mathbf{u}^{(k)}$ and $\mathbf{v}^{(k)}$, are functions of the real design parameter vector $\mathbf{p}$ with individual parameters denoted by Greek subscripts, e.g. p_α. Derivatives with respect to p_α are denoted by the subscript $,\alpha$ e.g. $\partial\mathbf{A}/\partial p_\alpha = \mathbf{A}_{,\alpha}$. All the design variables are assumed to be real.

The well-known biorthogonal property of the eigenvectors is given by

$$\mathbf{v}^{(i)\mathrm{T}}\mathbf{u}^{(j)} = 0 \quad \text{iff} \quad i \neq j \tag{3}$$

Note that the left hand side of equation (3) is not an inner product as usually understood, since $\mathbf{v}^{(i)}$ and/or $\mathbf{u}^{(j)}$ may be complex vectors. Note also that the left eigenvectors of $\mathbf{A}$ are the right eigenvectors of $\mathbf{A}^{\mathrm{T}}$ and *vice versa.*

NORMALIZATION OF EIGENVECTORS

The eigenvectors $\mathbf{u}^{(k)}$ and $\mathbf{v}^{(k)}$ are not completely defined by equations (1) and (2). A normalization condition has to be imposed to obtain unique eigenvectors. For brevity, let us consider only the normalization of the right eigenvector. A normalizing condition frequently imposed in the self-adjoint case is the following:

$$\mathbf{u}^{(k)\mathrm{T}}\mathbf{u}^{(k)}=1 \tag{4}$$

However, it is not always possible to use equation (4) for non-self-adjoint problems as $\mathbf{u}^{(k)\mathrm{T}}\mathbf{u}^{(k)}$ can equal zero or a very small number, causing numerical difficulties. This is true even if the matrix A is real. Unfortunately, considerable confusion exists in the literature regarding this point and several authors arbitrarily adopted equation (4) as a normalizing condition for non-self-adjoint problems, see e.g. References 7, 8, 11, 18–21. In this respect, the formulations of these references are not rigorous for general matrices.

One possible way to avoid the above difficulty is to replace equation (4) by

$$\mathbf{u}^{(k)*}\mathbf{u}^{(k)}=1 \tag{5}$$

where superscript* denotes a conjugate-transpose. Equation (5) is not prone to the difficulties of equation (4) because $\mathbf{u}^{(k)*}\mathbf{u}^{(k)}$ is always guaranteed to be non-zero. But, equation (5) is not a complete normalizing condition as it does not render the eigenvector unique. If $\mathbf{u}$ satisfies equation (5), then $\mathbf{w}=\mathbf{u}e^{\mathrm{i}c}$, where $\mathrm{i}=\sqrt{-1}$ and c is an arbitrary real number, also satisfies equation (5).[22]

Another normalization condition, inspired by the biorthogonality property of the left and right eigenvectors, is

$$\mathbf{v}^{(k)\mathrm{T}}\mathbf{u}^{(k)}=1 \tag{6}$$

Equation (6) also does not render the eigenvectors unique. It must be emphasized that if the eigenvector is not unique, neither is its derivative.

The normalization condition

$$u_m^{(k)}=1 \tag{7}$$

is very attractive because it renders the eigenvectors unique and, at the same time, the index m can be chosen easily to avoid ill-conditioning. Apparently, only Nelson[23] used this normalizing condition in obtaining the derivatives of eigenvectors. The index, m, may be chosen such that

$$|u_m^{(k)}|=\max_i \quad |u_i^{(k)}| \tag{8}$$

Another choice for m, used by Nelson[23] is

$$|u_m^{(k)}|\ |v_m^{(k)}|=\max_i \quad |u_i^{(k)}|\ |v_i^{(k)}| \tag{9}$$

The nature of uncertainty of the derivative of the eigenvector is of some interest. Without a normalizing condition, an eigenvector is uncertain to the extent of a non-zero constant multiplier. The derivative of an eigenvector is uncertain to the extent of an additive multiple of that eigenvector. To show this, let $\mathbf{u}^{(k)}$ be an eigenvector so that $\mathbf{w}^{(k)}=c\mathbf{u}^{(k)}$ is also an eigenvector. Then, if p_α is a design parameter,

$$\frac{\partial\mathbf{w}^{(k)}}{\partial p_\alpha}=\frac{\partial(c\mathbf{u}^{(k)})}{\partial p_\alpha}=c\frac{\partial\mathbf{u}^{(k)}}{\partial p_\alpha}+d\mathbf{u}^{(k)} \tag{10}$$

where $d=\partial c/\partial p_\alpha$ is arbitrary.[24] In practice, the constant d depends on the way the eigenvectors $\mathbf{u}^{(k)}$ and $\mathbf{w}^{(k)}$ are normalized.

METHODS OF CALCULATION

The various methods of calculating the derivatives of eigenvalues and eigenvectors can be divided into three categories:

1. *Adjoint Methods*, which use both the right and the left eigenvectors;
2. *Direct Methods*, which use only the right eigenvectors;
3. *Iterative Methods*, which use an iterative algorithm that converges to the required derivatives.

Adjoint Methods

Considering only a single parameter, Lancaster[25] obtained the following expressions for the first and second derivatives of an eigenvalue of a general matrix:

$$\lambda^{(k)}_{,\alpha}=\frac{\mathbf{v}^{(k)\mathrm{T}}\mathbf{A}_{,\alpha}\mathbf{u}^{(k)}}{\mathbf{v}^{(k)\mathrm{T}}\mathbf{u}^{(k)}} \tag{11}$$

$$\lambda^{(k)}_{,\alpha\alpha}=\frac{\mathbf{v}^{(k)\mathrm{T}}\mathbf{A}_{,\alpha\alpha}\mathbf{u}^{(k)}}{\mathbf{v}^{(k)\mathrm{T}}\mathbf{u}^{(k)}}+2\sum_{\substack{j=1\\ j\neq k}}^{n}\left[\frac{(\mathbf{v}^{(k)\mathrm{T}}\mathbf{A}_{,\alpha}\mathbf{u}^{(j)})(\mathbf{v}^{(j)\mathrm{T}}\mathbf{A}_{,\alpha}\mathbf{u}^{(k)})}{(\lambda^{(k)}-\lambda^{(j)})(\mathbf{v}^{(k)\mathrm{T}}\mathbf{u}^{(k)})(\mathbf{v}^{(j)\mathrm{T}}\mathbf{u}^{(j)})}\right] \tag{12}$$

An expression corresponding to equation (11) for a generalized quadratic eigenvalue problem was obtained by Pedersen and Seyranian.[26]

Morgan[27] obtained an expression for the derivative of an eigenvalue without requiring the eigenvectors explicitly. His expression is equivalent to

$$\lambda^{(k)}_{,\alpha}=\frac{\text{trace of }\{[\operatorname{adj}(\mathbf{A}-\lambda^{(k)}\mathbf{I})]\mathbf{A}_{,\alpha}\}}{\text{trace of }\operatorname{adj}(\mathbf{A}-\lambda^{(k)}\mathbf{I})} \tag{13}$$

The corresponding expression for derivatives with respect to matrix elements was derived by Nicholson.[28]

It can, however, be shown that[29]

$$\operatorname{adj}(\mathbf{A}-\lambda^{(k)}\mathbf{I})=t_k\mathbf{u}^{(k)}\mathbf{v}^{(k)\mathrm{T}} \tag{14}$$

where t_k is a constant and that[30]

$$\begin{aligned}\text{trace of }\{[\operatorname{adj}(\mathbf{A}-\lambda^{(k)}\mathbf{I})]\mathbf{A}_{,\alpha}\}&=t_k\mathbf{v}^{(k)\mathrm{T}}\mathbf{A}_{,\alpha}\mathbf{u}^{(k)}\\ \text{trace of }\operatorname{adj}(\mathbf{A}-\lambda^{(k)}\mathbf{I})&=t_k\mathbf{v}^{(k)\mathrm{T}}\mathbf{u}^{(k)}\end{aligned} \tag{15}$$

Thus, in the computation of $\operatorname{adj}[\mathbf{A}-\lambda^{(k)}\mathbf{I}]$, both right and left eigenvectors are implicitly computed, in view of equation (14). Equations (15) also show that Morgan's equation (equation (13)) is equivalent to Lancaster's equation (equation (11)). Woodcock[31] also obtained formulae involving the adjoint matrix for the first and second derivatives of eigenvalues. An operation count shows that calculation of the adjoint matrix is several times more expensive than the explicit calculation of right and left eigenvectors, so that Lancaster's formula is preferable to formulae requiring the adjoint matrix. This conclusion is also supported by sample computations.[32] In addition, although equation (13) was used satisfactorily for small problems,[33,34] numerical difficulties were reported for reasonably large problems.[35] Woodcock's formula for the second derivative of the eigenvalue requires a partial derivative of the adjoint matrix, and this is so complicated that Woodcock himself recommends the finite difference method. Formulae due to Morgan and Woodcock are not therefore considered in the following.

To obtain the second derivatives of eigenvalues, the first derivatives of left and right eigenvectors are calculated either implicitly[8, 11, 25] or explicitly.[1, 7, 12, 36, 37] Since the eigenvalues are assumed to be distinct, the first derivatives of eigenvectors can be expressed as

$$\mathbf{u}^{(k)}_{,\alpha}=\sum_{j=1}^{n} c_{kj\alpha}\mathbf{u}^{(j)} \quad \text{and} \quad \mathbf{v}^{(k)}_{,\alpha}=\sum_{j=1}^{n} d_{kj\alpha}\mathbf{v}^{(j)} \tag{16}$$

Rogers[38] obtained the coefficients $c_{kj\alpha}$ and $d_{kj\alpha}$ as

$$c_{kj\alpha}=\frac{\mathbf{v}^{(j)\mathrm{T}}\mathbf{A}_{,\alpha}\mathbf{u}^{(k)}}{(\lambda^{(k)}-\lambda^{(j)})\mathbf{v}^{(j)\mathrm{T}}\mathbf{u}^{(j)}} \quad k\neq j \tag{17}$$

$$d_{kj\alpha}=\frac{\mathbf{v}^{(k)\mathrm{T}}\mathbf{A}_{,\alpha}\mathbf{u}^{(j)}}{(\lambda^{(k)}-\lambda^{(j)})\mathbf{v}^{(j)\mathrm{T}}\mathbf{u}^{(j)}} \quad k\neq j \tag{18}$$

It can be observed that

$$d_{kj\alpha}=-c_{jk\alpha}\frac{\mathbf{v}^{(k)\mathrm{T}}\mathbf{u}^{(k)}}{\mathbf{v}^{(j)\mathrm{T}}\mathbf{u}^{(j)}} \tag{19}$$

Reddy[39] derived an equivalent expression for the response derivative by casting the derivative as the solution of a forced response problem for the same system.

Note that, in view of equation (10), the coefficients $c_{kk\alpha}$ and $d_{kk\alpha}$ in equation (16) are arbitrary and depend on the normalization of the eigenvectors. For example, if equation (7) is used to normalize the right eigenvectors, then

$$c_{kk\alpha}=-\sum_{\substack{j=1\\ j\neq k}}^{n} c_{kj\alpha}u^{(j)}_{m} \tag{20}$$

and if equation (6) is used to normalized the left eigenvectors, then

$$d_{kk\alpha}=-c_{kk\alpha} \tag{21}$$

It has been proposed[23, 40, 41] that the eigenvector derivative be approximated by using less than the full set of eigenvectors in the expansion of equation (16) so that the evaluation of the eigenvector derivative by the Adjoint Method could become cheaper. This variation of the Adjoint Method has received mixed reports in the literature.[23, 40] The quality of such an approximation is difficult to assess beforehand and the selection of eigenvectors to be retained in the expansion is problem dependent. It has not been considered in this work because a meaningful comparison with other methods cannot be easily made. However, this consideration should not be ignored while implementing the sensitivity calculations for particular problems.

The expressions for the second derivatives of eigenvalues were obtained by Plaut and Huseyin Reference 37. For the sake of simplicity in expressions, let us assume, without loss of generality, that the left eigenvectors are normalized as in equation (6). Plaut and Huseyin[37] obtained the second derivatives of eigenvalues with respect to uncorrelated parameters p_α and p_β as

$$\lambda^{(k)}_{,\alpha\beta}=\mathbf{v}^{(k)\mathrm{T}}\mathbf{A}_{,\alpha\beta}\mathbf{u}^{(k)}+\mathbf{v}^{(k)\mathrm{T}}\mathbf{A}_{,\alpha}\mathbf{u}^{(k)}_{,\beta}+\mathbf{v}^{(k)\mathrm{T}}_{,\beta}\mathbf{A}_{,\alpha}\mathbf{u}^{(k)} \tag{22}$$

which can be equivalently written, without involving the derivative of the left eigenvector, as

$$\lambda^{(k)}_{,\alpha\beta}=\mathbf{v}^{(k)\mathrm{T}}\mathbf{A}_{,\alpha\beta}\mathbf{u}^{(k)}+\mathbf{v}^{(k)\mathrm{T}}(\mathbf{A}_{,\alpha}-\lambda^{(k)}_{,\alpha}\mathbf{I})\mathbf{u}^{(k)}_{,\beta}+\mathbf{v}^{(k)\mathrm{T}}(\mathbf{A}_{,\beta}-\lambda^{(k)}_{,\beta}\mathbf{I})\mathbf{u}^{(k)}_{,\alpha} \tag{23}$$

For a diagonal second derivative, equation (22) is simplified to

$$\lambda^{(k)}_{,\alpha\alpha} = \mathbf{v}^{(k)\mathrm{T}}\mathbf{A}_{,\alpha\alpha}\mathbf{u}^{(k)} + 2\mathbf{v}^{(k)\mathrm{T}}\mathbf{A}_{,\alpha}\mathbf{u}^{(k)}_{,\alpha} \tag{24}$$

When the normalizing condition of equation (6) is used, equation (22) can be rewritten using equations (17) and (18) as

$$\lambda^{(k)}_{,\alpha\beta} = \mathbf{v}^{(k)\mathrm{T}}\mathbf{A}_{,\alpha\beta}\mathbf{u}^{(k)} + \sum_{\substack{j=1\\ j\neq k}}^{n} (\lambda^{(k)} - \lambda^{(j)})(c_{kj\alpha}d_{kj\beta} + c_{kj\beta}d_{kj\alpha}) \tag{25}$$

Crossley and Porter[1,42] derived similar expressions for derivatives with respect to the elements of the matrix. Expressions for Nth order diagonal derivatives were derived by Elrazaz and Sinha.[5]

In calculating the derivatives using equations (11) and (16)–(25):

(i) the first derivative of an eigenvalue requires the corresponding right and left eigenvectors;
(ii) the first derivative of an eigenvector requires *all* the left and right eigenvectors;
(iii) the second derivative of an eigenvalue requires the corresponding right and left eigenvectors and their first derivatives.

Direct Methods

The second category comprises methods that evaluate the derivatives using only the right eigenproblem. Direct Methods typically involve either the evaluation of the characteristic polynomial or the solution of a system of linear simultaneous equations without requiring all the left and right eigenvectors. Methods requiring the evaluation of the characteristic polynomial and the derivative of the determinant[35,43] are $O(n^5)$ processes while other methods considered here are at most $O(n^3)$ processes. In addition, the determination of the characteristic polynomial is, in general, an unsatisfactory process with respect to numercial stability, even when all the eigenvalues are well-conditioned.[44] While numerically stable algorithms have been proposed for evaluation of the characteristic polynomial[45] the computational expense still seems to be formidable. Hence, we do not consider these methods. Methods requiring the solution of a system of equations have the particularly attractive feature that the coefficient matrix needs to be factored only once for each eigenvalue, regardless of the number of parameters and the order of the derivatives required. Thus, they are very useful in applications where higher order derivatives are required.

The earliest method in this class is due to Garg[18] who obtained the first derivatives of the eigenvalue and the eigenvector by solving two systems of $(n+1)$ equations each in the real domain, without requiring any left eigenvectors. However, his formulation involves several matrix multiplications. Rudisill[19] prposed a scheme in which only the corresponding left and right eigenvectors are required to calculate the first derivative of the eigenvalue and the eigenvector. This was refined by Rudisill and Chu[20] to avoid calculating the left eigenvectors altogether. Solution of a system of only $(n+1)$ equations is required (though in the complex domain) to obtain the first derivatives of eigenvalue as well as eigenvector. Extension to higher order derivatives is straightforward. Cardani and Mantegazza[21] proposed solution methods of the same formulation for sparse matrices and extended it to the quadratic eigenproblem.

One weakness that is common to all the above formulations that do not require the left eigenvectors[18–21] is that they rely on the normalization condition given by equation (4), which is unreliable, as discussed earlier.

Nelson[23] circumvented this difficulty by using the normalizing conditions

$$\mathbf{v}^{(k)\mathrm{T}}\mathbf{u}^{(k)} = 1 \quad \text{and} \quad u^{(k)}_m = 1$$

However, the formulation of Rudisill and Chu is superior to Nelson's formulation in that it does not require any left eigenvectors.

In this paper, we propose a variation of the Rudisill and Chu formulation which does not rely on the questionable normalizing condition of equation (4) and at the same time requires no left eigenvectors.

Differentiating equation (1), we get

$$\mathbf{A}\mathbf{u}^{(k)}_{,\alpha}+\mathbf{A}_{,\alpha}\mathbf{u}^{(k)}=\lambda^{(k)}\mathbf{u}^{(k)}_{,\alpha}+\lambda^{(k)}_{,\alpha}\mathbf{u}^{(k)} \tag{26}$$

which can be rewritten in partitioned matrix form as

$$[\mathbf{A}-\lambda^{(k)}\mathbf{I}\,\vert\,-\mathbf{u}^{(k)}]\left\{\frac{\mathbf{u}^{(k)}_{,\alpha}}{\lambda^{(k)}_{,\alpha}}\right\}=\{-\mathbf{A}_{,\alpha}\mathbf{u}^{(k)}\} \tag{27}$$

Now, we impose the normalizing condition of equation (7). Differentiation of equation (7) yields,

$$u^{(k)}_{m,\alpha}=0 \tag{28}$$

Because of equation (28), the mth column of the coefficient matrix in equation (27) can be deleted. Equation (28) also reduces the number of unknowns by one so that equation (27) is now a system of n equations in n unknowns. Equation (27) is rewritten as

$$\mathbf{B}\mathbf{y}_1=\mathbf{r} \tag{29}$$

where

$$\mathbf{B}=[\mathbf{A}-\lambda^{(k)}\mathbf{I}\,\vert\,-\mathbf{u}^{(k)}]_{m\text{th column deleted}}$$

$$\mathbf{y}_1=\left\{\frac{\mathbf{u}^{(k)}_{,\alpha}}{\lambda^{(k)}_{,\alpha}}\right\}\text{ with }m\text{th element deleted}$$

$$\mathbf{r}=-\mathbf{A}_{,\alpha}\mathbf{u}^{(k)} \tag{30}$$

To get second derivatives, differentiate (27) with respect to p_β and obtain

$$(\mathbf{A}-\lambda^{(k)}\mathbf{I})\mathbf{u}^{(k)}_{,\alpha\beta}-\mathbf{u}^{(k)}\lambda_{,\alpha\beta}=-\mathbf{A}_{,\alpha\beta}\mathbf{u}^{(k)}-(\mathbf{A}_{,\alpha}-\lambda^{(k)}_{,\alpha})\mathbf{u}^{(k)}_{,\beta}-(\mathbf{A}_{,\beta}-\lambda^{(k)}_{,\beta})\mathbf{u}^{(k)}_{,\alpha} \tag{31}$$

or, in partitioned matrix form,

$$[\mathbf{A}-\lambda^{(k)}\mathbf{I}\,\vert\,-\mathbf{u}^{(k)}]\left\{\frac{\mathbf{u}^{(k)}_{,\alpha\beta}}{\lambda^{(k)}_{,\alpha\beta}}\right\}=\{-\mathbf{A}_{,\alpha\beta}\mathbf{u}^{(k)}-(\mathbf{A}_{,\alpha}-\lambda^{(k)}_{,\alpha})\mathbf{u}^{(k)}_{,\beta}-(\mathbf{A}_{,\beta}-\lambda^{(k)}_{,\beta})\mathbf{u}^{(k)}_{,\alpha}\} \tag{32}$$

Following the same reasoning as before, equation (32) is written as

$$\mathbf{B}\mathbf{y}_2=\mathbf{s} \tag{33}$$

where

$$\mathbf{y}_2=\left\{\frac{\mathbf{u}^{(k)}_{,\alpha\beta}}{\lambda^{(k)}_{,\alpha\beta}}\right\}\text{ with }m\text{th element deleted}$$

$$\mathbf{s}=-\mathbf{A}_{,\alpha\beta}\mathbf{u}^{(k)}-(\mathbf{A}_{,\alpha}-\lambda^{(k)}_{,\alpha}\mathbf{I})\mathbf{u}^{(k)}_{,\beta}-(\mathbf{A}_{,\beta}-\lambda^{(k)}_{,\beta}\mathbf{I})\mathbf{u}^{(k)}_{,\alpha} \tag{34}$$

Note that, if $\lambda^{(k)}$ is a simple eigenvalue of **A** and if $u^{(k)}_m\neq 0$, then the matrix **A** is of rank $(n-1)$ and the mth column that is deleted is linearly dependent on the other columns. Hence the matrix **B** is non-singular. (A formal proof is given in Appendix I.) The matrix **B** will also be well-conditioned if

$u_m^{(k)}$ is the largest component in the eigenvector $\mathbf{u}^{(k)}$ and the matrix $\mathbf{A}$ is itself not ill-conditioned. The vectors $\mathbf{y}_1$ and $\mathbf{y}_2$ can be obtained by standard solution methods. If the matrix $\mathbf{A}$ is banded or if the derivatives of both right and left eigenvectors are required, it may be more efficient to use a partitioning scheme described in Appendix II.

In summary, we note that, in calculating derivatives by the Direct Method:

(i) left eigenvectors are not used;

(ii) a complete solution of the eigenvalue problem is not required if the derivatives of only a few of the eigenvalues and eigenvectors are sought. This is in contrast to the Adjoint Method which requires all the left and right eigenvectors to calculate the first derivative of any eigenvector;

(iii) calculation of any derivative requires the solution of a system of linear equations;

(iv) only one matrix factorization needs to be performed for all orders of derivatives of an eigenvalue and its corresponding right and left eigenvectors with respect to all design parameters.

Iterative Methods

Andrew[22, 24] proposed an iterative algorithm to calculate the first derivatives of eigenvalues and eigenvectors. This algorithm is a refined and generalized version of the iterative scheme developed by Rudisill and Chu.[20] Based on literature published up to 1986, iterative methods do not appear to be competitive for computation of eigenvalue and eigenvector sensitivities of non-Hermitian matrices and hence are not considered here.*

RELATIVE COMPUTATIONAL COST

In this section, we compare the efficiency of calculating the derivatives of eigensystems as a function of the size of the matrix n, number of design parameters m and number of eigenvalues of interest l.

To start with, let us consider the operation counts(flops) for the Adjoint Methods given by equations (11) and (16)–(25) and the Direct Methods given by equations (29)–(34). They are summarized in Table I. It should be noted that the operation counts represent an estimate of the actual number of operations performed by a solution routine and include only the most significant terms. The actual number of operations will vary slightly, depending on programming details. The effect of the sparsity of the matrix derivative, $\mathbf{A}_{,\alpha}$ is modelled by the parameter κ, defined such that the number of operations in evaluating the product $\mathbf{A}_{,\alpha}\mathbf{u}$ is equal to κn^2(that is, $\kappa = 1$ corresponds to a full $\mathbf{A}_{,\alpha}$).

The eigenvalues are calculated using the EISPACK subroutine package[48] by first reducing the matrix to upper Hessenberg form using unitary similarity transformations and then applying the QR algorithm. The number of operations and CPU time for calculating the eigenvalues are not counted in the following results as they are not relevant in comparing the methods to calculate the derivatives.

The right eigenvectors are calculated by inverse iteration on the same upper Hessenberg matrix used for calculating the eigenvalues and back transformation using standard subroutines in the package EISPACK. The corresponding operation count is given in Table I. For the calculation of left eigenvectors, it is important to note that there is no need to repeat the process with the transposed matrix. The left eigenvectors are obtained cheaply using forward substitution in place of backward substitution in the inverse iteration process. There is also no need to repeat the matrix

*Recent work on iterative methods (e.g. References 46 and 47) indicates improvements in efficiency, especially for the dominant and the first few subdominant eigenvalues.

Table I. Operation counts

Eigenvalues and eigenvectors	
Operation	Operation count
Evaluation of eigenvalues	$8n^3$ to $10n^3$
Evaluation of right eigenvectors	$l(2n^2)$
Evaluation of left eigenvectors	$l(\frac{3}{2}n^2)$
Adjoint Methods	
Operation	Operation count
Evaluation of equation (11)	$lmn^2\kappa$
Evaluation of equations (16), (17), (18)	$lmn^2(\kappa+2)$
Evaluation of equation (25)	$l\binom{m}{2}n^2\kappa$
Direct Methods	
Operation	Operation count
LU decomposition of matrix **B**	$l(n^3/3)$
Formulation and solution of equation (29)	$lmn^2(\kappa+1)$
Formulation and solution of equation (33)	$l\binom{m}{2}n^2(3\kappa+1)$

factorization. A subroutine was written to calculate the right and left eigenvectors in this manner and the corresponding operation count is given in Table I.

Table I gives the operation counts of evaluating the individual steps. To obtain the number of operations involved in evaluating the derivatives, we must add the operation counts for all the steps required in the calculations. These counts are given in Appendix III and are discussed in the following section.

CPU TIME STATISTICS

In the following tables, computational cost for the calculation of the first and second derivatives of eigensystems is compared for matrices of order 20, 40 and 60. The CPU time statistics are obtained on the IBM 3084 computer using the VS-FORTRAN compiler with no compiler optimization. The correlation between operation counts and CPU times is shown in Tables II and III. The ratio of operation count(OC) and CPU time for various operations, tabulated in Tables II and III, is about 10^5 operations per CPU second with a variability of 27 per cent.

The matrices are generated for the dynamic stability analysis of a compressor stage rotor with mistuned blades. The geometric and structural parameters of the rotor and formulation and method of analysis are the same as those of NASA Test Rotor 12 described in Reference 49 except that the number of blades and the torsional frequencies are varied. The torsional frequency values are selected randomly from a population of mean 1·0 and standard deviation 0·01. The standard deviations of the actual samples are slightly different.

Calculation of first derivatives of eigenvalues only

It is clear from the operation count (Table IV) that the Adjoint Method, which is an $O(n^2)$ process, is superior to the Direct Method, an $O(n^3)$ process. The number of design variables and the

Table II. Correlation between operation count(OC) and CPU time

Calculation of right eigenvectors

n	l	OC/CPU seconds (× 10^4)
60	60	8·6
60	10	8·3

Calculation of right and left eigenvectors

n	l	OC/CPU seconds (× 10^4)
60	60	8·5
60	10	9·2

Evaluation of equation (11)

n	l	m	OC/CPU seconds (× 10^4)
60	60	10	10·7
60	60	5	10·7
60	10	10	10·7
60	10	5	10·7

Evaluation of equations (16), (17), (18)

n	l	m	OC/CPU seconds (× 10^4)
60	60	10	12·0
60	60	5	11·9
60	10	10	11·9
60	10	5	11·9

Table III. Correlation between operation count (OC) and CPU time

Evaluation of equation (25)

n	l	m	OC/CPU seconds (× 10^4)
60	60	10	9·1
60	10	5	7·4

Decomposition of matrix **B**

n	l	OC/CPU seconds (× 10^4)
60	60	8·5
60	10	9·2

Evaluation of equation (29)

n	l	m	OC/CPU seconds (× 10^4)
60	60	10	12·9
60	10	5	13·0

number of eigenvalues of interest have no bearing on this conclusion. As the order of the matrix increases, the Direct Method becomes more expensive. For example, for 5 design variables and 10 eigenvalues of interest, the CPU time for the Direct Method is 2·3 times more expensive than for the Adjoint Method for $n=20$, and for $n=60$, the ratio is 3·0.

Calculation of first derivatives of eigenvalues and eigenvectors

When the derivatives of both eigenvalues and right eigenvectors are required, the choice of method is dependent on the values of l and m. When very few eigenvalues are of interest, the Direct

Table IV. Operation count: calculation of first derivatives of eigenvalues only

Adjoint Method	$l(\frac{7}{2}n^2+\kappa mn^2)$
Direct Method	$l\left[\frac{n^3}{3}+(\kappa+1)mn^2\right]$

Method is cheaper. When many eigenvalues are of interest, the Direct Method is more expensive than the Adjoint Method. However, this effect of the number of eigenvalues of interest is less significant when the number of design variables is large. As the number of design variables increases, the Direct Method becomes more competitive, even when all eigenvalues are of interest. For a 60×60 full ($\kappa=1$) matrix, this is illustrated in Figure 1.

The operation count (Table V) shows that the computation by Adjoint Method of eigenvector derivative, which is necessary for the second derivative of the eigenvalue, is an $O(n^3)$ process and is more expensive than the computation of the eigenvector itself, which is an $O(n^2)$ process. This fact is significant as some authors have stated the opposite.[2,3]

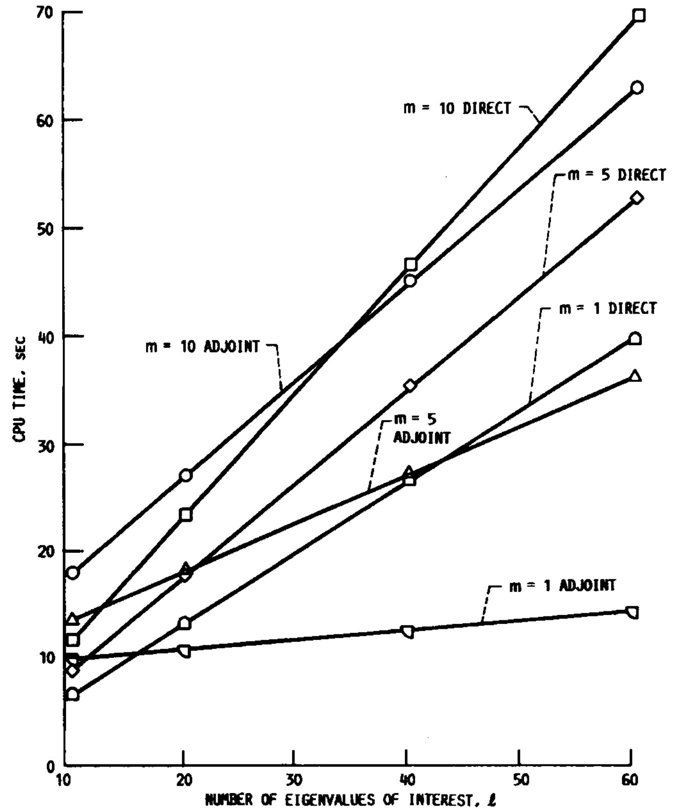

Figure 1. CPU times for calculation of first derivatives of eigenvalues and eigenvectors for a 60×60 matrix

Table V. Operation count: calculation of first derivatives of eigenvalues and eigenvectors

Adjoint Method	$\frac{7}{2}n^3+lmn^2(\kappa+2)$
Direct Method	$\frac{ln^3}{3}+lmn^2(\kappa+1)$

Table VI. Operation count: calculation of first and second derivatives of eigenvalues only

Adjoint Method	$\frac{7}{2}n^3+(\kappa+1)mn^3+l\binom{m}{2}n^2\kappa$
Direct Method	$l\frac{n^3}{3}+l\binom{m}{2}n^2(3\kappa+1)$
Direct–Adjoint Method	$l\frac{n^3}{3}+l\binom{m}{2}n^2\kappa+lmn^2(2\kappa+1)$

Calculation of first and second derivatives of eigenvalues only

The Direct–Adjoint Method denotes the calculation of the eigenvector derivatives by the Direct Method and the eigenvalue derivatives by the Adjoint Method. The third term in the operation count for the Direct–Adjoint Method is significant only when m is small. From the operation count (Table VI), it is seen that the Direct–Adjoint Method is always cheaper than the Direct Method. Hence, the choice is between the Adjoint Method and the Direct–Adjoint Method, Here, considerations similar to those of the last section hold and the choice of method depends on the values of l and m. When few eigenvalues are of interest, the Direct–Adjoint Method is cheaper. When many eigenvalues are of interest, the Adjoint Method is superior. But this advantage of the Adjoint Method diminishes as the number of design variables increases. This is again illustrated for a 60×60 full matrix ($\kappa=1$) in Figure 2.

CONCLUDING REMARKS

The normalization of the eigenvector needs to be properly related to its derivative. In practice, this means that the derivative of the eigenvector is to be normalized before it is used, to conform to the normalization of the eigenvector itself. When the eigenvector is not normalized in a unique manner, its derivative cannot be evaluated. Fixing one of the components of the eigenvector is the best normalizing condition for computation of the derivative. The methods found in the literature are extended to apply to eigenvectors normalized in this manner.

Various methods for calculation of derivatives of eigenvalues and eigenvectors are surveyed and classified as Direct or Adjoint. Adjoint Methods use both the left and the right eigenvectors whereas the Direct Methods use only the right eigenvectors. Their relative efficiency is evaluated as a function of matrix size, number of eigenvalues of interest and the number of design parameters. General recommendations are made for the cases when (a) eigenvalue first derivatives are required, (b) eigenvalue and eigenvector first derivatives are required and (c) eigenvalue second derivatives are required.

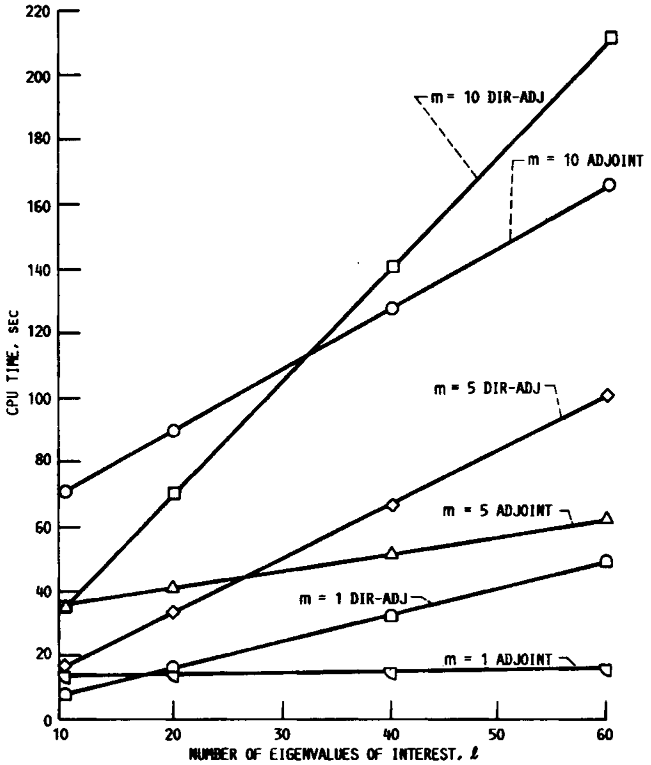

Figure 2. CPU times for calculation of second derivatives of eigenvalues for a 60 × 60 matrix

When only eigenvalue first derivatives are required, the calculation of left eigenvectors is worth the expense as the Adjoint Method is shown to be superior to the Direct Method. When first derivatives of eigenvectors are also required, the decision is dependent on the problem size, the number of design variables and the number of eigenvalues of interest. When the first and second derivatives of eigenvalues are required, similar considerations hold. It is also shown that once the first derivatives of eigenvectors are calculated, the second derivatives of eigenvalues are calculated more efficiently by the Adjoint Method than by the Direct Method.

ACKNOWLEDGEMENT

This work was sponsored by NASA grants NAG-3-347 and NAG-1-224.

APPENDIX I

Proof that the matrix **B** *of equation (29) is non-singular*

The matrix **B** is defined, following equation (29), as

$$\mathbf{B} = [\mathbf{A} - \lambda^{(k)}\mathbf{I} \,\vdots\, -\mathbf{u}^{(k)}]_{m\text{th column deleted}}$$

Let $\mathbf{b}_i$ denote the ith column of $\mathbf{A}-\lambda^{(k)}\mathbf{I}$. $\mathbf{B}$ is not singular if its columns are linearly independent. Assume the contrary. That is, suppose that we can find a non-zero vector $\mathbf{c}$ with components c_i such that

$$c_1\mathbf{b}_1+\ldots+c_{m-1}\mathbf{b}_{m-1}+c_m\mathbf{u}^{(k)}+c_{m+1}\mathbf{b}_{m+1}+\ldots+c_n\mathbf{b}_n=0 \tag{A1}$$

Equation (2) implies that

$$\mathbf{v}^{(k)\mathrm{T}}\mathbf{b}_i=0 \quad i=1,\ldots,n \tag{A2}$$

Multiplying equation (A1) by $\mathbf{v}^{(k)\mathrm{T}}$, we get

$$c_m\mathbf{v}^{(k)\mathrm{T}}\mathbf{u}^{(k)}=0 \tag{A3}$$

and since $\mathbf{v}^{(k)\mathrm{T}}\mathbf{u}^{(k)}\neq 0$ (see equation (3)), we find that $c_m=0$. Then, equation (A1) may be written as

$$(\mathbf{A}-\lambda^{(k)}\mathbf{I})\mathbf{c}=\mathbf{0} \tag{A4}$$

so that $\mathbf{c}$ is an eigenvector of $\mathbf{A}$ corresponding to $\lambda^{(k)}$. However, this is impossible because $\mathbf{c}$ cannot be proportional to $\mathbf{u}^{(k)}$ (as $c_m=0$ and $u_m^{(k)}\neq 0$) and $\lambda^{(k)}$ is a simple eigenvalue.

Hence $\mathbf{c}=\mathbf{0}$ and the columns of $\mathbf{B}$ are linearly independent.

APPENDIX II

Modification of Direct Method for banded matrices

Equations (29) and (33) can be written as

$$(\mathbf{A}-\lambda^{(k)}\mathbf{I})_{m\text{th column deleted}}\mathbf{u}^{(k)}_{,\alpha\, m\text{th row deleted}}-\lambda_{,\alpha}\mathbf{u}=\mathbf{r} \tag{A5}$$

Let $\mathbf{u}^{(k)}$ be normalized so that $u_m^{(k)}=1$.

Equation (A5) is a system of n equations. Writing the mth equation separately, we have, if the superscript (k) is omitted for notational convenience,

$$\mathbf{C}\mathbf{x}_{,\alpha}-\lambda_{,\alpha}\mathbf{x}=\mathbf{t} \tag{A6}$$

and

$$\mathbf{a}_m^{\mathrm{T}}\mathbf{x}_{,\alpha}-\lambda_{,\alpha}=r_m \tag{A7}$$

where

$$\mathbf{C}=(\mathbf{A}-\lambda\mathbf{I})_{m\text{th row and column deleted}}$$

$$\mathbf{x}_{,\alpha}=\mathbf{u}_{,\alpha\, m\text{th row deleted}}$$

$$\mathbf{x}=\mathbf{u}_{m\text{th row deleted}}$$

$$\mathbf{t}=\mathbf{r}_{m\text{th row deleted}}$$

$\mathbf{a}_m^{\mathrm{T}}=m$th row of $\mathbf{A}$ with the mth column deleted

From (A7),

$$\lambda_{,\alpha}=\mathbf{a}_m^{\mathrm{T}}\mathbf{x}_{,\alpha}-r_m \tag{A8}$$

From (A6),

$$\mathbf{x}_{,\alpha}=\mathbf{C}^{-1}(\lambda_{,\alpha}\mathbf{x}+\mathbf{t}) \tag{A9}$$

Eliminating $\mathbf{x}_{,\alpha}$, we have

$$\lambda_{,\alpha} = \frac{\mathbf{t}^{\mathrm{T}}\mathbf{b}_m - r_m}{1 - \mathbf{x}^{\mathrm{T}}\mathbf{b}_m} \tag{A10}$$

where

$$\mathbf{b}_m = [\mathbf{C}^{\mathrm{T}}]^{-1}\mathbf{a}_m$$

Proceeding in a similar manner for the left eigenvector,

$$\mathbf{y}_{,\alpha} = [\mathbf{C}^{\mathrm{T}}]^{-1}(\lambda_{,\alpha}\mathbf{y} + \mathbf{t}_l) \tag{A11}$$

where

$$\mathbf{y}_{,\alpha} = \mathbf{v}_{,\alpha\ m\text{th row deleted}}$$

$$\mathbf{y} = \mathbf{v}_{m\text{th row deleted}}$$

$$\mathbf{t}_l = (\mathbf{r}_l)_{m\text{th row deleted}}$$

$\mathbf{r}_l$ being the appropriate right hand side.

Thus, the following procedure can be used to obtain the derivatives $\lambda_{,\alpha}$ and $\mathbf{u}_{,\alpha}$.

1. Form an LU decomposition of the matrix $\mathbf{C}$.
2. Solve $\mathbf{b}_m = [\mathbf{C}^{\mathrm{T}}]^{-1}\mathbf{a}_m$ by forward substitution.
3. Calculate $\lambda_{,\alpha}$ from (A10).
4. Calculate $\mathbf{x}_{,\alpha}$ from (A9) by backward substitution.
5. Expand $\mathbf{x}_{,\alpha}$ to $\mathbf{u}_{,\alpha}$ setting $u_{m,\alpha} = 0$.

If the derivatives $\mathbf{v}_{,\alpha}$ of the left eigenvectors are also required, only three further steps are needed.

6. Calculate $\mathbf{y}_{,\alpha}$ from (A11) by forward substitution.
7. Expand $\mathbf{y}_{,\alpha}$ to $\mathbf{v}_{,\alpha}$ setting $v_{m,\alpha} = 0$.
8. Normalize $\mathbf{v}_{,\alpha}$ appropriately, depending on the normalization of $\mathbf{v}$. For example, to obtain the derivative of the left eigenvector that satisfies the normalization condition of equation (6), subtract $(\mathbf{v}^{\mathrm{T}}\mathbf{u}_{,\alpha} + \mathbf{v}^{\mathrm{T}}_{,\alpha}\mathbf{u})\mathbf{v}$.

The matrix $\mathbf{C}$ needs to be factored only once. Also, the matrix $\mathbf{C}$ retains the bandedness characteristics of the original matrix $\mathbf{A}$. Furthermore, higher derivatives can be obtained by merely substituting an appropriate right hand side vector, $\mathbf{r}$. However, higher order derivatives can suffer in accuracy because of accumulated round-off error.

The conditioning of matrix $\mathbf{C}$ needs some comment. Note that $\mathbf{C}$ is obtained from the singular matrix $(\mathbf{A} - \lambda^{(k)}\mathbf{I})$ by deleting both the row and column corresponding to index m. Hence, for matrix $\mathbf{C}$ to be non-singular, one must make sure that the mth row is linearly dependent on the other rows as well as that the mth column is linearly dependent on the other columns. In other words, $\mathbf{C}$ is non-singular iff $u_m^{(k)} \neq 0$ and $v_m^{(k)} \neq 0$. If $v_m^{(k)}$ is very small compared to the largest element in $\mathbf{v}^{(k)}$, steps 2 and 4 in the above procedure will give inaccurate results even if $u_m^{(k)}$ is the largest element in $\mathbf{u}^{(k)}$. In general, it is not possible to make a good choice for m without the knowledge of the left eigenvector. Since the calculation of the left eigenvector by forward substitution is cheap, it is suggested that the left eigenvector be calculated and the index m be chosen as in equation (9). This is the same criterion used by Nelson[23] and will assure as well-conditioned a matrix $\mathbf{C}$ as possible.

APPENDIX III

Details of operation counts

The operation counts for eigensensitivity analysis using various procedures are derived below by adding the operation counts for the component tasks. The range of the subscripts and superscripts is given by (except where noted)

$$k=1,\ldots,l$$
$$\alpha=1,\ldots,m$$
$$\beta=1,\ldots,m$$
$$j=1,\ldots,n$$

The order of the matrix is n.

1. First derivatives of l eigenvalues with respect to m design variables by Adjoint Method

(1) Computation of $\mathbf{u}^{(k)}$	$l(2n^2)$
(2) Computation of $\mathbf{v}^{(k)}$	$l(\frac{3}{2}n^2)$
(3) Computation of $\lambda^{(k)}_{,\alpha}$ by equation (11)	$lmn^2\kappa$

Adding, the total operation count is $l(\frac{7}{2}n^2+\kappa mn^2)$.

2. First derivatives of l eigenvalues with respect to m design variables by Direct Method
This procedure is identical to the procedure 4.

3. First derivatives of l eigenvalues and eigenvectors with respect to m design variables by Adjoint Method

(1) Computation of $\mathbf{u}^{(j)}$	$2n^3$
(2) Computation of $\mathbf{v}^{(j)}$	$\frac{3}{2}n^3$
(3) Computation of $\mathbf{A}_{,\alpha}\mathbf{u}^{(k)}$	$lmn^2\kappa$
(4) Computation of $\mathbf{v}^{(j)\mathrm{T}}\mathbf{A}_{,\alpha}\mathbf{u}^{(k)}$	lmn^2
(5) Computation of $c_{kj\alpha}$, $\lambda^{(k)}_{,\alpha}$ by equation (17)	lmn
(6) Computation of $\mathbf{u}^{(k)}_{,\alpha}$ by equation (16)	lmn^2

The operation count for step (5) is not significant in comparison to those of steps (4) and (6). Hence, the total operation count is $\frac{7}{2}n^3+lmn^2(\kappa+2)$.

4. First derivatives of l eigenvalues and eigenvectors with respect to m design variables by Direct Method

(1) Computation of $\mathbf{u}^{(k)}$	$l(2n^2)$
(2) *LU* decomposition of $\mathbf{B}$	$l(n^3/3)$
(3) Formulation of right hand side in equation (29)	$lmn^2\kappa$
(4) Solution of equation (29)	lmn^2

The operation count for step (1) is not significant in comparison to that of step (2).

Hence, the total operation count is $l\left(\dfrac{n^3}{3}\right)+lmn^2(\kappa+1)$.

5. First and second derivatives of l eigenvalues with respect to m design variables by Adjoint Method

(1) Computation of $c_{kj\alpha}$, $k=1,\ldots,n$ and $d_{kj\alpha}=-c_{kj\alpha}$ (See procedure 3)	$\frac{7}{2}n^3+mn^3(\kappa+1)$
(2) Computation of $\mathbf{v}^{(k)\mathrm{T}}\mathbf{A}_{,\alpha\beta}\mathbf{u}^{(k)}$	$l\binom{m}{2}n^2\kappa$
(3) Computation of $\lambda^{(k)}_{,\alpha\beta}$ by equation (25)	$3l\binom{m}{2}n$

The operation count for step (3) is not significant in comparison to that for step (2).

Hence, the total operation count is $\frac{7}{2}n^3+(\kappa+1)mn^3+l\binom{m}{2}n^2\kappa$.

6. First and second derivatives of l eigenvalues with respect to m design variables by Direct Method

(1) Computation of $\mathbf{u}^{(k)}_{,\alpha}$ by Direct Method (See procedure 4)	$l\frac{n^3}{3}+lmn^2(\kappa+1)$
(2) Formulation of right hand side in equation (33)	$3l\binom{m}{2}n^2\kappa$
(3) Solution of equation (33)	$l\binom{m}{2}n^2$

Adding, the total operation count is $l\left(\frac{n^3}{3}\right)+l\binom{m}{2}n^2(3\kappa+1)$.

7. First and second derivatives of l eigenvalues with respect to m design variables by Direct–Adjoint Method

(1) Computation of $\mathbf{u}^{(k)}_{,\alpha}$ by Direct Method (See procedure 4)	$l\frac{n^3}{3}+lmn^2(\kappa+1)$
(2) Computation of $\mathbf{v}^{(k)\mathrm{T}}\mathbf{A}_{,\alpha\beta}\mathbf{u}^{(k)}$	$l\binom{m}{2}n^2\kappa$
(3) Computation of $\mathbf{v}^{(k)\mathrm{T}}(\mathbf{A}_{,\alpha}-\lambda^{(k)}_{,\alpha}\mathbf{I})$	$lmn^2\kappa$
(4) Computation of $\lambda^{(k)}_{,\alpha\beta}$ by equation (23)	$2l\binom{m}{2}n$

The operation count for step (4) is not significant in comparison to that for step (2).

Hence, the total operation count is $l\frac{n^2}{3}+(2\kappa+1)lmn^2+l\binom{m}{2}n^2\kappa$.

REFERENCES

1. T. R. Crossley and B. Porter, 'High-order eigenproblem sensitivity methods: theory and applications to the design of linear dynamical systems', *Int. J. Control*, **10**, 315–329 (1969).
2. H. M. Zein El-Din, R. T. H. Alden and P. C. Chakravarti, 'Second order eigenvalue sensitivities applied to multivariable control systems', *Proc. IEEE*, **65**, 277–278 (1977).
3. H. M. Zein El-Din and R. T. H. Alden, 'Second order eigenvalue sensitivities applied to power system dynamics', *IEEE Trans. Power Apparat. Syst.*, **PAS-96**, 1928–1936 (1977).
4. A. H. Flax, 'Comment on Estimation of fundamental frequencies of beams and plates with varying thickness', *AIAA J.*, **16**, 1022–1024 (1978).
5. Z. Elrazaz and N. K. Sinha, 'Dynamic stability analysis of power systems for large parameter variations', *IEEE PES Summer Meeting*, Vancouver, Canada, July 12–20, 1979.
6. Z. Elrazaz and N. K. Sinha, 'Dynamic stability evaluation for multimachine system: an efficient eigenvalue approach', *IEE Proc. D*, **128**, 268–274 (1981).
7. C. S. Rudisill and K. G. Bhatia, 'Second derivatives of the flutter velocity and the optimization of aircraft structures', *AIAA J.*, **10**, 1569–1572 (1972).

8. K. G. Bhatia, 'An automated method for determining the flutter velocity and the matched point', *J. Aircraft*, **11**, 21–27 (1974).
9. R. T. Haftka and M. P. Kamat, *Elements of Structural Optimization*, Martinus Nijhoff, The Netherlands, 1985.
10. H. Miura and L. A. Schmit, 'Second order approximation of natural frequency constraints in structural synthesis, *Int. j. numer. methods eng.*, **13**, 337–351 (1978).
11. C. S. Rudisill and J. L. Cooper, 'An automated procedure for determining the flutter velocity', *J. Aircraft*, **10**, 442–444 (1973).
12. J. L. Junkins, D. S. Bodden and M. P. Kamat, 'An eigenvalue optimization approach for feedback control of flexible structures', *Proc. SECTAM XII*, May 10–11, 1984, Vol. II, pp. 303–308.
13. C. G. J. Jacobi, 'Uber ein leichtes Verfahren die in der Theorie der Saecular stoerungen vorkommenden Gleichungen numerisch aufzuloesen', *Z. Reine unde Angewandte Math.*, **30**, 51–95 (1846).
14. W. H. Wittrick, 'Rates of change of eigenvalues with reference to buckling and vibration problems', *J. Royal Aeronaut. Soc.*, **66**, 590–591 (1962).
15. R. L. Fox and M. P. Kapoor, 'Rates of change of eigenvalues and eigenvectors', *AIAA J.*, **6**, 2426–2429 (1968).
16. M. S. Zarghamee, 'Minimum weight design with stability constraint', *J. Struct. Div. ASCE*, **96**, 1697–1710 (1970).
17. R. L. Fox, *Optimization Methods for Engineering Design*, Addison-Wesley, Reading, Massachusetts, 1971.
18. S. Garg, 'Derivatives of eigensolutions for a general matrix', *AIAA J.*, **11**, 1191–1194 (1973).
19. C. S. Rudisill, 'Derivatives of eigenvalues and eigenvectors for a general matrix', *AIAA J.*, **12**, 721–722 (1974).
20. C. S. Rudisill and Y. Chu, 'Numerical methods for evaluating the derivatives of eigenvalues and eigenvectors', *AIAA J.*, **13**, 834–837 (1975).
21. C. Cardani and P. Mantegazza, 'Calculation of eigenvalue and eigenvector derivatives for algebraic flutter and divergence problems', *AIAA J.*, **17**, 408–412 (1979).
22. A. L. Andrew, 'Convergence of an iterative method for derivatives of eigensystems', *J. Comp. Phys.*, **26**, 107–112 (1978).
23. R. B. Nelson, 'Simplified calculation of eigenvector derivatives', *AIAA J.*, **14**, 1201–1205 (1976).
24. A. L. Andrew, 'Iterative computation of derivatives of eigenvalues and eigenvectors', *J. Inst. Mathematics and Its Applications*, **24**, 209–218 (1979).
25. P. Lancaster, 'On eigenvalues of matrices dependent on a parameter', *Numerische Math.*, **6**, 377–387 (1964).
26. P. Pedersen and A. P. Seyranian, 'Sensitivity analysis for problems of dynamic stability', *Int. J. Solids Struct.*, **19**, 315–335 (1983).
27. B. S. Morgan, 'Computational procedure for sensitivity of an eigenvalue', *Electron. Lett.*, **2**, 197–198 (1966).
28. H. Nicholson, 'Eigenvalue and state-transition sensitivity of linear systems', *Proc. IEE* (*London*), **114**, 1991–1995 (1967).
29. R. A. Frazer, W. J. Duncan and A. R. Collar, *Elementary Matrices*, Cambridge University Press, London, 1938.
30. D. C. Reddy, 'Evaluation of the sensitivity coefficient of an eigenvalue', *IEEE Trans. Automat. Control.* **AC-12**, 792 (1967).
31. D. L. Woodcock, 'The rates of change of eigenvalues of a lambda matrix and their use in flutter investigations', *J. Royal Aeronaut. Soc.*, **70**, 364–365 (1966).
32. W. J. Vetter, 'Reply to R. G. Rains' comments on 'An extension to gradient matrices', *IEEE Trans. Syst., Man Cybernetics*, **SMC-2**, 285 (1972).
33. V. T. Nagraj, 'Changes in the vibration characteristics due to changes in the structure', *J. Spacecraft*, **7**, 1499–1501 (1970).
34. P. N. Paraskevopoulos, M. A. Christodoulou and M. A. Tsakiris, 'Eigenvalue eigenvector sensitivity analysis of linear time-invariant singular systems', *IEEE Trans. Automat. Control*, **AC-29**, 344–346 (1984).
35. J. B. Nail, J. R. Mitchell and W. L. McDaniel, Jr., 'Determination of pole sensitivities by Danilevskii's method', *AIAA J.*, **15**, 1525–1527 (1977).
36. D. L. Taylor and T. R. Kane, 'Multiparameter quadratic eigenproblems', *J. Appl. Mech. ASME*, **42**, 478–483 (1975).
37. R. H. Plaut and K. Huseyin, 'Derivatives of eigenvalues and eigenvectors in non-self-adjoint systems', *AIAA J.* **11**, 250–251 (1973).
38. L. C. Rogers, 'Derivatives of eigenvalues and eigenvectors', *AIAA J.*, **8**, 943–944 (1970).
39. D. C. Reddy, 'Eigenfunctions and the solution of sensitivity equations', *Electron. Lett.*, **4**, 262–263 (1968).
40. T. K. Hasselman and G. C. Hart, 'Modal analysis of random structural systems', *J. Eng. Mech. Div. ASCE*, **98**, 561–579 (1979).
41. H. M. Adelman and R. T. Haftka, 'Sensitivity analysis for discrete structural systems', *AIAA J.*, **24**, 823–832 (1986).
42. T. R. Crossley and B. Porter, 'Eigenvalue and eigenvector sensitivities in linear systems theory', *Int. J. Control*, **10**, 163–170 (1969).
43. J. R. Mitchell, J. B. Nail and W. L. McDaniel, Jr., 'Computation of cofactors of $[sI-A]$ with applications', *IEEE Trans. Automat. Control*, **AC-24**, 637–638 (1979).
44. J. H. Wilkinson, *The Algebraic Eigenvalue Problem*, Clarendon Press, Oxford, 1965.
45. W. Werner, 'A generalized companion matrix of a polynomial and some applications', *Linear Algebra and Its Applications*, **55**, 19–36 (1983).

46. R. C. E. Tan, 'Accelerating the convergence of an iterative method for derivatives of eigensystems', *J. Comp. Phys.*, **67**, 230–235 (1986).

47. R. C. E. Tan, 'An extrapolation method for computing derivatives of eigensystems', *Int. J. Comp. Math.*, to appear.

48. B. T. Smith, J. M. Boyle, B. S. Garbow, Y. Ikebe and V. C. Kleme, 'Matrix eigensystem routines—EISPACK guide', *Lecture Notes in Computer Science*, Vol. 6, Springer-Verlag, Heidelberg, 1974, pp. 95–97.

49. K. R. V. Kaza and R. E. Kielb, 'Flutter and response of a mistuned cascade in incompressible Flow,' *AIAA J.*, **20**, 1120–1127 (1982).