AIAA JOURNAL
Vol. 32, No. 2, February 1994

New Family of Modal Methods for Calculating Eigenvector Derivatives

Mehmet A. Akgün*
Middle East Technical University, Ankara, Turkey

A new family of modal methods for the calculation of eigenvector derivatives in non-self-adjoint systems with a singular coefficient matrix is developed. The family contains the modal and modified modal methods as a subset. In the family, the component of the mth eigenvector in the expansion of the derivative of the jth eigenvector is multiplied by various powers of the eigenvalue ratio λ_j/λ_m, thereby accelerating convergence. The family of methods is applied to a self-adjoint example problem, namely, a cantilever beam whose root depth is used as the design variable to which the sensitivity of the first four mode shapes is sought. Two different numbers of elements are used to model the beam in two cases in an effort to investigate the effect of the system size on the performance of the new methods. Central processor time and number of modes needed for convergence are determined. For a given problem, one of the methods in the family takes the smallest number of modes and shortest time to converge. The method is applied to a non-self-adjoint system with zero eigenvalues as well. The family is compared with Nelson's method and the modified Rudisill and Chu method on the basis of operation counts and is expected to perform better than the two when more than one eigenvector derivative is of interest.

Nomenclature

$[A]$, $[B]$ = coefficient matrices of the eigenvalue problem
$[A_E]$ = see Eq. (25)
$[B_R]$ = see Eq. (29)
$\{a_{jk}\}$ = eigenvector weights in a column matrix form
a_{jkm}, b_{jkm} = weights of the eigenvectors in the expansion of right and left eigenvector derivatives, respectively; Eq. (12)
$[C]$ = see Eq. (37); described following Eq. (39)
$[D_E]$ = see Eq. (27)
$[F_{ij}]$ = see Eq. (8)
f_j = see Eq. (3)
$[G_{ij}]$ = see Eq. (15)
N = order of the eigenvalue problem
N_R = number of zero eigenvalues
$\hat{N}$ = number of modes used in the specific modal method
n = order of the modal method; Eq. (11)
$[P]$ = see Eq. (32b)
p = number of eigenvector derivatives to be computed
q = number of design parameters to which sensitivity is sought
$[R]$ = see Eq. (34)
r_{jm} = eigenvalue ratio; Eq. (14)
s_j = see Eq. (4)
t = time to compute an eigenvector derivative including eigensolution
t_{ne} = time to compute an eigenvector derivative excluding eigensolution
$[X]$, $[Y]$ = "modal matrices" for the right and left eigenvectors, respectively
$\{x_i\}$, $\{y_i\}$ = ith right and left eigenvectors, respectively
ϵ = relative error in eigenvector derivative; Eq. (40)
κ = sparsity parameter for $[A]_{,k}$
λ_i = ith eigenvalue

Superscripts

T = transpose of a matrix
$*$ = complex conjugate transpose of a matrix
-1 = inverse of a matrix

Subscripts

E = quantity associated with nonzero eigenvalues, "elastic mode"
$,k$ = partial derivative with respect to kth design variable
R = quantity associated with zero eigenvalues, "rigid-body mode"

Introduction

THE need for eigensolution sensitivities in a variety of problems is widely known. Some examples are system identification, development of systems insensitive to parameter variation, and automated structural optimization. Modal frequency and mode shape derivatives with respect to design parameters may, for example, be required in aircraft wing design to tailor vibrational and flutter characteristics.

Computation of eigenvalue derivatives is straightforward, whereas a single universally agreed-on method of computation of eigenvector derivatives that is efficient seems inexistant even though Nelson's method[1] and the modified Rudisill and Chu method[11] have been emerging as superior methods among the others. Reference 2 is a review paper on the methods available for computing sensitivity derivatives. Reference 3 compares four methods of calculating eigenvector derivatives in terms of the central processor time required. The methods compared are finite-difference method, modal method,[4-6] a modified modal method,[7,8] and Nelson's method.[1] The finite-difference method approximates the eigenvector derivative by a difference formula. In the modal method, the derivative is expanded in terms of the eigenvectors, possibly in terms of a subset of them. The modified modal method uses a smaller subset of the eigenvectors and is similar to the mode-acceleration method of dynamic response calculation.[9] The latter

Received March 12, 1992; presented in part as Paper 92-4747 at the AIAA/USAF/NASA/OAI 4th Symposium on Multidisciplinary Analysis and Optimization, Cleveland, OH, Sept. 21-23, 1992; revision received March 3, 1993; accepted for publication March 3, 1993.

*Associate Professor, Aeronautical Engineering Department. Member AIAA.

includes a pseudostatic response term in the dynamic solution, thereby accelerating convergence. Nelson's method is an exact analytical method that yields an exact solution to the equation governing eigenvector derivatives.

Reference 3 applied the foregoing four methods to three example problems. In terms of central processor time, Nelson's method performed better overall than the finite-difference method, though not by a great margin. The modified modal method always converged faster than the modal method, but was much slower than the other two because the number of modes required not being known a priori, many more modes were computed than actually needed for convergence. Reference 3 concluded that the modified modal method could compete with Nelson's method for at least the first mode shape derivative when the number of modes needed in the summation was known before the eignsolution was performed. Reference 11, on the other hand, surveyed methods for sensitivity analysis, proposed a variation of the Rudisill and Chu method[12] based on a different normalizing condition than the original method, and compared the modal method (called the adjoint method in Ref. 11) with the modified Rudisill and Chu method on the basis of operation counts and central processor times. Reference 11 concluded that the superiority of either method they investigated for computing first derivatives of eigenvectors depends on the number of design variables and eigenvalues being differentiated.

This paper presents a new family of modal methods for computing eigenvector derivatives in non-self-adjoint systems with a singular coefficient matrix. To the author's knowledge, no previous study on eigensolution sensitivity exists for systems having zero eigenvalues. The methods developed are similar to the family of mode-superposition methods for dynamic response calculations developed in Ref. 10. The family includes the modal and modified modal methods as a subset. In the modified modal method, the component of the mth eigenvector in the expansion of the derivative of the jth eigenvector is first multiplied by the eigenvalue ratio λ_j/λ_m before being added into the derivative expression that speeds up convergence as compared to the modal method. The information lost by excluding the higher eigenvectors is carried by the pseudostatic response term. In the higher-order modal methods within the family, the so-called modal method being the lowest-order one, the component of the mth eigenvector in the derivative is multiplied by higher powers of λ_j/λ_m that accelerates convergence even further up to a certain order. The information lost in then compensated by additional terms involving the system coefficient matrices. The formal derivation of the family of methods is presented next. For a given problem, one of the methods in the family requires the smallest number of eigenvectors and usually takes the shortest time. The performance of the family is demonstrated with an example whereby the number of modes and central processing time needed for convergence are investigated. The floating-point operation count for the method is given in terms of number of design variables and eigenvalue derivatives of interest and is compared to the counts in Ref. 11. A second example is presented on non-self-adjoint systems with zero eigenvalues.

Eigensensitivity Equations

The non-self-adjoint problem with distinct eigenvalues can be expressed with the following four equations:

$$[A]\{x_j\} = \lambda_j[B]\{x_j\}, \qquad j = 1, \ldots, N \tag{1}$$

$$\{y_i\}^T[A] = \lambda_i\{y_i\}^T[B], \qquad i = 1, \ldots, N \tag{2}$$

$$\{x_j\}^T[B]\{x_j\} = f_j \quad \text{or} \quad \{x_j\}^*[B]\{x_j\} = f_j \qquad j = 1, \ldots, N \tag{3}$$

$$\{y_i\}^T[B]\{x_j\} = \delta_{ij}s_j, \qquad i, j = 1, \ldots, N \tag{4}$$

where $[A]$ and $[B]$ are general complex matrices of order N, δ_{ij} is the Kronecker's delta, and superscripts T and * denote the transpose and complex conjugate transpose, respectively. Although, in most cases, the eigenvalue problem in Eq. (1) may be reduced to one in which $[B]$ is replaced by the identity matrix $[I]$, the problem is treated here in the form just given. When $[B] = [I]$, the first normalizing condition in Eq. (3) may not be suitable as the product may be zero or very small for complex eigenvectors. The second normalizing condition in Eq. (3) for $[B] = [I]$ is always nonzero, but it renders the eigenvector $\{x_j\}$ unique only within a complex factor of unit modulus.[11] Another possible normalizing condition is based on setting a selected element of the eigenvector equal to unity. On the other hand, when $[B] \neq [I]$ and $[B]$ is a real or complex unsymmetrical matrix, it is possible that either condition in Eq. (3) may be zero or very small for both real and complex eigenvectors.

It can be shown that the eigenvalue derivative is given by

$$\lambda_{j,k} = \frac{1}{s_j}\{y_j\}^T([A]_{,k} - \lambda_j[B]_{,k})\{x_j\} \tag{5}$$

where subscript k following a comma indicates a partial derivative with respect to the kth design parameter. Right and left eigenvector derivatives are obtained by differentiating Eqs. (1) and (2), respectively, which leads to

$$([A] - \lambda_j[B])\{x_j\}_{,k} = [F_{jk}]\{x_j\} \tag{6}$$

$$([A]^T - \lambda_j[B]^T)\{y_j\}_{,k} = [F_{jk}]^T\{y_j\} \tag{7}$$

where

$$[F_{jk}] = \lambda_{j,k}[B] + \lambda_j[B]_{,k} - [A]_{,k} \tag{8}$$

The case of a singular $[A]$ is treated later.

Development of the Method for Nonsingular $[A]$

Development of the family of methods follows along the lines of the derivation of the nth-order mode-superposition method used in forced structural response analysis.[10] To that end, Eq. (6) is rearranged in the following manner:

$$\{x_j\}_{,k} = [A]^{-1}\Big([F_{jk}]\{x_j\} + [B](\lambda_j\{x_j\}_{,k})\Big) \tag{9}$$

Equation (9) is then multiplied by λ_j and the result is substituted back into the second term on the right-hand side of Eq. (9), which subsequently becomes

$$\{x_j\}_{,k} = [A]^{-1}\Big(([F_{jk}] + \lambda_j[B][A]^{-1}[F_{jk}])\{x_j\} + [B][A]^{-1}[B](\lambda_j^2\{x_j\}_{,k})\Big) \tag{10}$$

Equation (9) is now multiplied by λ_j^2 and the result is substituted in Eq. (10) that now contains $\lambda_j^3\{x_j\}_{,k}$. This process of multiplying Eq. (9) with increasing powers of the eigenvalue and substituting the result into successive $\{x_j\}_{,k}$ expressions can be repeated n times so that

$$\{x_j\}_{,k} = \sum_{i=0}^{n}(\lambda_j[A]^{-1}[B])^i[A]^{-1}[F_{jk}]\{x_j\} + (\lambda_j[A]^{-1}[B])^{n+1}\{x_j\}_{,k} \tag{11}$$

At this point, the eigenvector derivative on the right-hand side of Eq. (11) is expanded in terms of a truncated set of eigenvectors

$$\{x_j\}_{,k} = \sum_{m=1}^{\hat{N}} a_{jkm}\{x_m\}, \qquad \hat{N} < N \tag{12}$$

where N is the system size. The substitution of Eq. (12) into the right-hand side of Eq. (11), use of Eq. (1), and manipulation of the summation term yield

$$\{x_j\}_{,k} = \frac{(n+1)\lambda_{j,k}}{\lambda_j}\{x_j\} + \sum_{i=0}^{n}(\lambda_j[A]^{-1}[B])^i[A]^{-1}[G_{jk}]\{x_j\} + \sum_{m=1}^{\hat{N}} a_{jkm}(r_{jm})^{n+1}\{x_m\}, \qquad n = 0, 1, 2, \ldots \tag{13}$$

where

$$r_{jm} \equiv \lambda_j/\lambda_m, \qquad [G_{jk}] \equiv \lambda_j[B]_{,k} - [A]_{,k} \tag{14,15}$$

In Eq. (13), $n = 0$ case corresponds to the modified modal method and is analogous to the mode-acceleration method of dynamic response calculation. In this paper, $n = -1$ designates the modal method that is analogous to the mode-displacement method of response calculation.[9] Equation (13) covers the latter case also where the first two terms on the right side are absent. Equation (13) indicates that, for large enough n, only the first $L \le \hat{N}$ modes are needed for which $r_{jL} \ge 1$. Increasing n, however, increases round-off errors. On the other hand, the higher the mode number for which a derivative is required, the larger the round-off errors become for a given n. Optimum n values should therefore be sought. The coefficients a_{jkm} are obtained by substituting Eq. (13) into Eq. (6), premultiplying the result by $\{y_l\}^T$, $l \neq j$, and simplifying the result. Hence,

$$a_{jkm} = \frac{\{y_m\}^T[G_{jk}]\{x_j\}}{s_m(\lambda_m - \lambda_j)}, \qquad j \neq m \tag{16}$$

which is independent of n. Equation (16) is valid for the modal method also. To obtain a_{jkj}, the first normalizing condition in Eq. (3), which leads to a simpler expression, is differentiated once. This results in

$$\{x_j\}^T([B] + [B]^T)\{x_j\}_{,k} + \{x_j\}^T[B]_{,k}\{x_j\} = 0 \tag{17}$$

The second condition in Eq. (3) could be used as well. Substitution of Eq. (13) in Eq. (17) and rearrangement give

$$a_{jkj} = -\frac{1}{2f_j}\{x_j\}^T([B]+[B]^T)\left(\sum_{i=0}^{n}(\lambda_j[A]^{-1}[B])^i[A]^{-1}[G_{jk}]\{x_j\} + \sum_{\substack{m=1\\ m\neq j}}^{\hat{N}} a_{jkm}(r_{jm})^{n+1}\{x_m\}\right) - \frac{(n+1)\lambda_{j,k}}{\lambda_j} - \frac{\{x_j\}^T[B]_{,k}\{x_j\}}{2f_j} \tag{18a}$$

Equation (18a) is valid for all nonnegative n. For $n = -1$, i.e., the modal method, the first summation term is absent. For a self-adjoint problem, Eq. (18a) reduces to

$$a_{jkj} = -\{x_j\}^T[B]_{,k}\{x_j\}/2f_j \tag{18b}$$

Furthermore, when $[B]$ is the identity matrix, $a_{jkj} = 0$; i.e., as a result of the normalizing condition used, an eigenvector derivative is orthogonal to the eigenvector itself.

Similar derivations follow easily for the left eigenvector. $\{y_j\}_{,k}$ is then given by Eq. (13), with x's, $[A]$'s, $[B]$'s, $[G_{jk}]$, and a_{jkm} being replaced by y, $[A]^T$, $[B]^T$, $[G_{jk}]^T$, and b_{jkm}, respectively. b_{jkm} for $j \neq m$ is given by Eq. (16) with subscripts m and j of y and x, respectively, in the numerator interchanged. To find b_{jkj}, Eq. (4) is differentiated to give

$$\{y_j\}_{,k}^T[B]\{x_j\} + \{y_j\}^T[B]_{,k}\{x_j\} + \{y_j\}^T[B]\{x_j\}_{,k} = 0 \tag{19}$$

Upon substitution of Eq. (13) and the similar expression for $\{y_j\}_{,k}$ in Eq. (19), use of Eqs. (1), (2), and (5), and simplification, there results

$$(b_{jkj} + a_{jkj})s_j = -\{y_j\}^T[B]_{,k}\{x_j\} \tag{20}$$

The knowledge of a_{jkj} from Eq. (18) allows b_{jkj} to be solved from Eq. (20). For self-adjoint systems, left and right eigenvectors are identical and all the foregoing equations are valid with y simply replaced by x.

Case of Singular $[A]$

When the $[A]$ matrix is singular,

$$\lambda_j = 0, \quad \lambda_{j,k} = \frac{1}{s_j}\{y_j\}^T[A]_{,k}\{x_j\}, \qquad j = 1, \ldots, N_R \tag{21a}$$

where N_R is the number of zero eigenvalues. For higher modes, the eigenvalue derivative is still given by Eq. (5). Then

$$[A]\{x_j\}_{,k} = (\lambda_{j,k}[B] - [A]_{,k})\{x_j\}, \qquad j = 1, \ldots, N_R \tag{21b}$$

Equation (21b) can be solved with an existing method. The derivative of higher eigenvectors is developed next, again governed by Eq. (6). However, the eigenvector derivative computed from Eq. (6) may contain "rigid-body" components, Hence,

$$\{x_j\}_{,k} = [X_R]\{a_{jkR}\} + [X_E]\{a_{jkE}\}, \qquad j = N_R + 1, \ldots, N \tag{22}$$

where $[X_R]$ and $[X_E]$ are the modal matrices containing the right "rigid" and "elastic" modes, respectively. When Eq. (6) is rewritten as

$$[A]\{x_j\}_{,k} = [F_{jk}]\{x_j\} + [B](\lambda_j\{x_j\}_{,k}) \tag{23}$$

and Eq. (22) is substituted on the left-hand side of Eq. (23) that is then premultiplied by $[Y_E]^T$, the modal matrix containing the left elastic modes, the following results:

$$[A_E]\{a_{jkE}\} = [Y_E]^T\Big([F_{jk}]\{x_j\} + [B](\lambda_j\{x_j\}_{,k})\Big) \tag{24}$$

where

$$[A_E] \equiv [Y_E]^T[A][X_E] = \mathrm{diag}\{\lambda_i s_i\} \tag{25}$$

The elastic part of the derivative in Eq. (22) is obtained from Eq. (24), which is

$$\{x_{jE}\}_{,k} \equiv [X_E]\{a_{jkE}\} = [D_E]\Big([F_{jk}]\{x_j\} + [B](\lambda_j\{x_j\}_{,k})\Big) \tag{26}$$

where

$$[D_E] \equiv [X_E][A_E]^{-1}[Y_E]^T \tag{27}$$

$[D_E]$, which is a singular matrix, is analogous to the elastic flexibility matrix encountered in the computation of the dynamic response of systems possessing rigid-body modes with the mode-accelaration method.[9] Equation (26) is similar to Eq. (9). Hence, the elastic part of the eigenvector derivative is given by Eq. (13) with $[A]^{-1}$ replaced by $[D_E]$, $[B]$ left untouched, and the lower limit on the second summation term replaced by $N_R + 1$. Coefficients a_{jkm} are still given by Eqs. (16) and (18) with $[A]^{-1}$ in Eq. (18a) replaced by $[D_E]$. In Eq. (18a), the first N_R terms within the second summation term are now associated with zero eigenvalues and, therefore, the corresponding r_{jm} are to be taken as unity. The coefficients of these rigid-body components are computed by substituting Eq. (22) into Eq. (6), premultiplying by $[Y_R]^T$, the modal matrix containing the left rigid-body modes, and solving the resulting equation. Thus,

$$\{a_{jkR}\} = -\frac{1}{\lambda_j}[B_R]^{-1}[Y_R]^T[G_{jk}]\{x_j\}; \quad j = N_R + 1, \ldots, N \tag{28}$$

which is a compact version of Eq. (16) for the rigid-body components, where

$$[B_R] \equiv [Y_R]^T[B][X_R] \tag{29}$$

On the other hand, it is known that

$$[Y_R]^T[B]\{x_{jE}\},_k = 0 \tag{30}$$

Therefore, from Eqs. (22) and (30), another expression for $\{a_{jkR}\}$ is obtained

$$\{a_{jkR}\} = [B_R]^{-1}[Y_R]^T[B]\{x_j\},_k \tag{31}$$

The substitution of Eq. (31) into Eq. (22) yields

$$\{x_{jE}\},_k = [P]\{x_j\},_k; \qquad [P] \equiv [I] - [X_R][B_R]^{-1}[Y_R]^T[B] \tag{32a,b}$$

where $[I]$ is the identity matrix of order N.

Matrix $[D_E]$, as given by Eq. (27), requires the knowledge of all 'elastic' modes. It can, however, be computed with a different approach. To that end, Eq. (28) is substituted in Eq. (22), which is then put in Eq. (6) with the result that

$$([A] - \lambda_j[B])\{x_{jE}\},_k = [R][F_{jk}]\{x_j\} \tag{33}$$

where $[G_{jk}]$ in Eq. (28) is replaced by $[F_{jk}]$ since $[Y_R]^T[B]\{x_j\} = \{0\}$ anyway and

$$[R] \equiv [I] - [B][X_R][B_R]^{-1}[Y_R]^T \tag{34}$$

For a self-adjoint system, $[R] = [P]^T$. On the other hand, $[R] = [P]$ when $[B] = [I]$. It can be shown that

$$[R][F_{jk}]\{x_j\} = \lambda_{j,k}([R][B])\{x_j\} + \Big(\lambda_j([R][B]),_k - ([R][A]),_k\Big)\{x_j\} \tag{35}$$

The substitution of Eq. (32a) into Eq. (33) and use of the identities $[A][P] = [R][A]$ and $[B][P] = [R][B]$ lead to

$$([RA] - \lambda_j[RB])\{x_j\},_k = [R][F_{jk}\{x_j\} \tag{36}$$

Equations (6), (8), and (35) reveal that Eq. (36) describes the behavior of a system with matrices $[RA]$ and $[RB]$ instead of the original $[A]$ and $[B]$. In analogy with Eqs. (6) and (9), Eq. (33) can be rewritten as

$$\{x_j\},_k = [C]\Big([R][F_{jk}]\{x_j\} + [R][B](\lambda_j\{x_j\},_k)\Big) \tag{37}$$

which, with the use of Eq. (32a), becomes

$$\{x_{jE}\},_k = [P][C][R]\Big([F_{jk}]\{x_j\} + [B](\lambda_j\{x_j\},_k)\Big) \tag{38}$$

Comparison of Eqs. (26) and (38) leads to

$$[D_E] = [P][C][R] \tag{39}$$

In the mode-acceleration method of dynamic response calculation applied to self-adjoint systems with rigid-body modes, the matrix corresponding to $[C]$ is the flexibility matrix of the system with the rigid-body modes constrained.[9] $[C]$ here may be determined in an analogous manner as follows: A number, which is equal to the rank defect of $[A]$, of rows and columns of $[A]$ are set to zero. The same rows and columns of $[C]$ are also set to zero. The remaining submatrix of $[A]$ is inverted and then placed in the vacant positions of $[C]$. The submatrix in question, however, may still be singular, in which case rows to be set to zero need to be reselected.

Numerical Examples

Two examples are presented in this section to address various issues. The first is a self-adjoint example to investigate the convergence properties of the family of methods. The second is a non-self-adjoint example with zero eigenvalues to validate the formulation with matrix $[C]$.

Self-Adjoint Example

Derivatives of the first four mode shapes of a cantilever beam of rectangular cross section, which is a self-adjoint system, are computed with various methods in the family. To assess the effect of system size on the performance of the new methods, the beam is first modeled with 20, then in the second case, 40 beam elements. Each node in either case has the freedom to translate tranversely and rotate about an axis normal to the plane of motion. Beam mass is lumped at the nodes. The reduced system size in the two cases is thus $N = 20$ and $N = 40$, respectively. The design variable to which the eigensolution sensitivity is sought is the depth of the root element, which is taken to be 15 cm. The relevant cross-sectional area moment of inertia of the beam is 4100 cm^4, except for the root element for which the value is 3900 cm^4. This example is similar to one of the three in Ref. 3, which had 22 degrees of freedom. Convergence of a method is decided by means of a relative error measure that is based on the 2-norm of a vector. The error is thus defined as

$$\epsilon \equiv (\{\delta\}^T\{\delta\})^{1/2}/(\{\alpha\}^T\{\alpha\})^{1/2} \tag{40}$$

where $\{\alpha\}$ is the exact value of the respective eigenvector derivative taken here as the value computed with the modal method with all the system mode shapes used and

$$\{\delta\} \equiv \{x_j\},_k - \{\alpha\} \tag{41}$$

with $\{x_j\},_k$ being the approximate derivative from any method, characterized by the index n, and for any number of modes $\tilde{N}$. Figure 1 shows the convergence of the methods $n = 0$, 1, and 2 for the first four mode shape derivatives for case 1 (i.e., $N = 20$). The error in the modal method ($n = -1$) is outside the range of the plots and is therefore not shown. That error for the first mode shape derivative with two modes, is, e.g., 0.34, and with five modes, 0.056.

Figure 1 reveals that error level is higher for derivatives of higher modes. A second fact is that convergence is faster for higher n up to a certain value of the latter. After a point, however, round-off errors become dominant, where the precision used is double precision, due to the increasing number of matrix operations, and methods with a higher index n converge with increasing values of error. This starts to become evident in the derivative of the fourth mode (Fig. 1d) where the method of order 2 ($n = 2$) converges with an eventual error of 0.0018 (reached with six modes), whereas the eventual error for $n = 1$ is 9.08×10^{-7} (reached with 15 modes). On the other hand, the third-order method is not useful for the derivative of the fourth mode and converges with an error of 7.189×10^{-10}, 4.607×10^{-5}, and 0.0362 for that of the first, second, and third modes, respectively.

The convergence picture and error values in the case of $N = 40$ are similar to those in Fig. 1, for at least the number of modes involved in the figure. The error values in both cases are quite close, but those for $N = 40$ are consistently higher than those for $N = 20$ for the same number of modes. For example, the error in the derivative of the third mode with the first-order method with five modes is 0.00034 and 0.00039 for $N = 20$ and 40, respectively, which is fairly representative of the situation. It should, however, be noted that five modes constitute 25% of all modes of the smaller system, whereas the percentage is halved for the larger system.

Central processor time to converge is a major concern for any method. To compare the methods in the new family in terms of time, convergence is now assumed to be reached when the error ϵ [Eq. (40)] is less than 0.003. Tables 1 and 2 present for the two cases $N = 20$ and 40, respectively, the number of modes $\tilde{N}$ required for convergence with each method up to $n = 2$ and the corresponding execution times

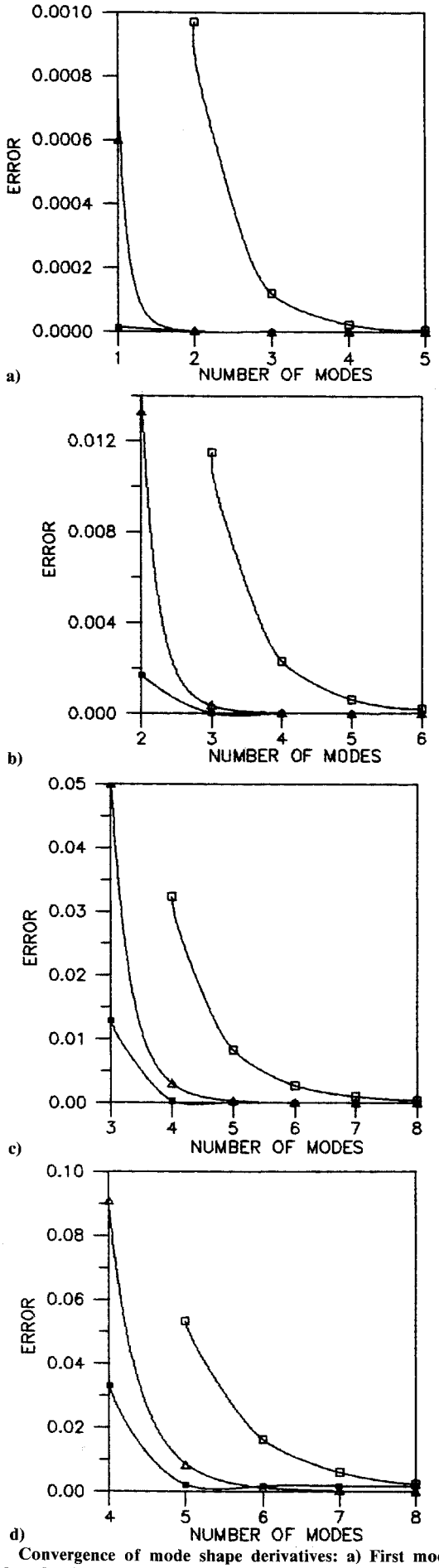

Fig. 1 Convergence of mode shape derivatives: a) First mode, b) second mode, c) third mode, and d) fourth mode. □ : $n = 0$, △ : $n = 1$, ■ : $n = 2$.

recorded on a 80386 microcomputer. t is the total time to read the matrices $[A]$, the stiffness matrix, and $[B]$, the mass matrix, from a data file; to find the $\hat{N}$ eigensolutions (hence, the number of eigenvectors was known a priori), to compute $\partial[A]/\partial h$, where h is the depth of the root element, to perform the sensitivity calculations, and to write the derivative onto the screen. (Writing a vector of order N onto the screen took $0.03N$ s on a one-item-per-line basis, whereas it would take nearly one-fourth of that time to write the same vector in a data file.) The times are accurate to ± 0.05 s.

Situations may exist in which the eigenvectors may already be available in a file and hence need not be computed during the sensitivity calculations. In the tables, t_{ne} is the time it takes to perform the same steps as in t, with the exception of the $\hat{N}$ eigensolutions that are read from a file. The times in the tables are for the computation of a single mode shape derivative at a time. Time savings would be realized if all four derivatives were computed in the same run for a given n because terms such as $([A]^{-1}[B])^i$ in Eq. (13), which are independent of eigenvalues and eigenvectors, would have to be computed only once. The following conclusions are reached from Tables 1 and 2:

1) The number of modes required for a given n is the same, or almost the same, regardless of the system size with the exception of the modal method. A tendency seems to exist, however, for more modes (but a significantly smaller percentage of all the system modes) to be required as the system size grows. The extent of this for large systems is unknown.

2) Derivatives of the first and second modes require one and two modes, respectively, to converge. As the number of the mode being differentiated increases, however, one or two more modes than the number of the mode are required.

3) The modal method ($n = -1$) takes the shortest time if the eigenvectors are already known. However, a large number of mode shapes (50% or more of all the system modes) is required, which makes the method impractical.

4) With the modal method excluded, the so-called modified modal method ($n = 0$) takes the shortest time if the eigenvectors are already known. The number of modes required for convergence with $n = 0$, however, may still be considered high in some cases. The CPU times in the last column of both tables for various values of n differ by small amounts only. The trend, however, is clearly seen in going from the smaller to the larger system size in that time differences are increasing.

5) When the eigenvectors are computed during the same run, either the first- or second-order method takes the shortest time, provided the number of design variables to which sensitivity is sought is small (or 1 as here). In such cases then, the first-order method may be preferable because of smaller round-off errors. This choice is supported by the fact that $n = 2$ does not converge for the derivative of the fourth mode shape in the larger system (Table 2) within the error bound. (It converges with an error of 0.053.) The CPU times for various values of n are again closely spaced but start moving apart as the system size grows, thus suggesting greater savings for larger systems. On the other hand, when the number of design variables is not small, the zeroth-order method may perform better. This point is elaborated on in the next section on computational considerations.

Some of the foregoing conclusions depend on the error limit chosen to define convergence. For example, if an error ϵ of 0.024 is acceptable, the first mode derivative for the system $N = 20$ can be computed using a single mode with the zeroth-order method, in which case the latter would be taking the shortest time instead of the first-order method. On the other hand, the CPU times just given were measured with $\hat{N}$ known a priori. Such knowledge may not be available in practical systems. The examples presented here, however, seem to provide a crude estimate. At most, 50% more modes than the number of the mode being differentiated were required for convergence for both system sizes. Whether this holds for large systems or derivatives of higher modes needs to be investigated. However, even if twice or thrice as many modes as the

number of the mode being differentiated were needed for convergence, the fact that only lower eigenvector derivatives whose number constitutes a small fraction of all the system eigenvectors in a practical large system are usually of interest would lead one to conclude that computing cost would not increase significantly. This point is clarified in the next section, where operation counts of various methods are compared.

Singular Non-Self-Adjoint Example

The system

$$[A] = \begin{bmatrix} 1 & g & -1 & 1 & 0 & 0 \\ -1 & -0.8 & g & g & g+1 & g-1 \\ 0 & -1 & 1 & -1 & -1 & 1 \\ 1 & 0 & -1 & 1 & 0 & 0 \\ -1 & 1 & 0 & -1 & 0 & -1 \\ 0 & -1 & 1 & 0 & 0 & 1 \end{bmatrix}, \quad [B] = [I]$$

has the eigenvalues $\{\lambda\}^T = \{0\ 0\ -1\ 1\ 1.6 + i(1.04)^{1/2}\ 1.6 - i(1.04)^{1/2}\}$ for $g = 2.4$, where i is the complex identity. The right and left eigenvectors are, respectively,

$$[X] = \begin{bmatrix} 1 & -1 & -1.6 & 1 & -0.2261 - i0.5726 & -0.2261 + i0.5726 \\ 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 & 0.6590 + i0.6814 & 0.6590 - i0.6814 \\ 0 & 1 & 0.8 & -1.4 & -1.2928 + i0.1073 & -1.2928 - i0.1073 \\ 0 & -1 & -1.3 & 0.9 & 0.9426 - i0.6478 & 0.9426 + i0.6478 \\ -1 & 0 & 0.5 & 0.5 & 0.3502 + i0.5405 & 0.3502 - i0.5405 \end{bmatrix}$$

$$[Y] = \begin{bmatrix} 0 & -0.3 & 0.1 & 1 & -0.0476 + i0.4856 & -0.0476 - i0.4856 \\ 0 & 1.2 & 0.3 & 1 & 0.4762 + i0.2428 & 0.4762 - i0.2428 \\ 0 & 4.08 & -0.82 & 1 & 0.0190 - i0.1942 & 0.0190 + i0.1942 \\ 1 & 1.5 & -1.74 & 3.4 & 0.9524 + i0.4856 & 0.9524 - i0.4856 \\ 1 & 0 & -1.84 & 2.4 & 1 & 1 \\ 1 & -5.76 & -0.72 & 2 & -0.0286 + i0.2914 & -0.0286 - i0.2914 \end{bmatrix}$$

The matrix $[A]$ has seven nonzero minors of order four for $g = 2.4$, any one of which can be selected to compute $[C]$; Eq. (39). Inversion of the submatrix obtained by eliminating the 1st and 5th rows and columns of $[A]$, e.g., leads to the following $[C]$ matrix:

$$[C] = \tfrac{1}{3} \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 5 & 17 & 5 & 0 & -24 \\ 0 & 0 & -3 & -3 & 0 & 3 \\ 0 & 0 & -3 & 0 & 0 & 3 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 5 & 20 & 8 & 0 & -24 \end{bmatrix}$$

In a vibrating system where $[A]$ is the stiffness matrix, what was just done is equivalent to fixing the 1st and 5th degrees of freedom. $[D_E]$, which should and does come out to be the same regardless of the nonsingular 4×4 submatrix of $[A]$ selected, is then computed to be

$$[D_E] = \begin{bmatrix} 1.1852 & 1.5370 & -1.0741 & -0.7259 & -1.9111 & 0.1111 \\ 0.2778 & 0.5556 & 1.8889 & 5.1111 & 4.8333 & 2.1667 \\ 0.2778 & 0.5556 & 0.8889 & 2.1111 & 1.8333 & 1.1667 \\ -0.7407 & -1.6482 & -0.3037 & -2.2963 & -1.5556 & -1.0444 \\ 0.7407 & 1.6482 & -0.6963 & 0.2963 & -0.4444 & 0.0444 \\ 0.0000 & 0.0000 & 1.0000 & 2.0000 & 2.0000 & 1.0000 \end{bmatrix}$$

For the purpose of illustration, the derivative of the third right eigenvector with respect to parameter g is computed here. It is found that $\lambda_{3,g} = 10/78$ and the exact eigenvector derivative is

$$\{x_3\}_{,g} = \{0.0979\ \ 0.1722\ \ 0.0279\ \ -0.0573\ \ -0.0009\ \ 0.0582\}^T$$

Table 1 Processing time and number of modes for convergence of mode shape derivatives ($N = 20$)

	n	$\hat{N}$	t(s)	t_{ne}(s)
First mode	−1	11	11.75	3.35
	0	2	5.99	3.46
	1	1	5.38	3.52
	2	1	5.49	3.62
Second mode	−1	13	13.02	3.46
	0	4	7.30	3.52
	1	3	6.70	3.57
	2	2	6.15	3.68
Third mode	−1	14	13.73	3.47
	0	6	8.68	3.68
	1	4	7.41	3.63
	2	4	7.58	3.74
Fourth mode	−1	16	15.16	3.68
	0	8	10.11	3.73
	1	6	8.79	3.79
	2	5	8.24	3.68

If the derivative is computed with the first five modes for any n, it erroneously turns out to be a complex value. Therefore, if a complex mode is used in any order modal method, its complex conjugate mode should also be used. The columns of the matrix next given show, from left to right, the third right eigenvector derivative computed with the first four modes with $n = 0$ through $n = 7$, respectively.

$n=0$	1	2	3	4	5	6	7
0.1098	0.1004	0.0924	0.1021	0.0957	0.0987	0.0978	0.0978
0.1672	0.1507	0.1927	0.1600	0.1774	0.1710	0.1719	0.1729
−0.0345	0.0479	0.0274	0.0227	0.0326	0.0251	0.0290	0.0276
−0.0158	−0.0542	−0.0717	−0.0455	−0.0639	−0.0548	−0.0578	−0.0576
−0.0074	−0.0145	0.0129	−0.0095	0.0028	−0.0019	−0.0011	−0.0005
0.0232	0.0687	0.0587	0.0549	0.0611	0.0567	0.0589	0.0581

For $n = 7$, of the six entries in the eigenvector derivative, five have converged with a relative error near or smaller than 1%, except the fifth that has a large error. The fifth entry, however, is itself very small in magnitude. In this problem, one should indeed use all six modes with $n = -1$ since the magnitude of the third eigenvalue is not very small compared to the modulus of the fifth (and, therefore, the sixth) eigenvalue. The latter is the reason why it takes such a high value of n for the third eigenvector derivative to converge with four modes. The example does illustrate a significant point, however, in that the convergence is ultimately achieved even when two modes are ignored in a problem of this character.

Table 2 Processing time and number of modes for convergence of mode shape derivatives ($N = 40$)

	n	$\hat{N}$	t(s)	t_{ne}(s)
First mode	−1	18	50.86	13.02
	0	2	27.91	15.16
	1	1	26.91	15.55
	2	1	27.41	15.99
Second mode	−1	22	56.96	13.57
	0	4	30.87	15.22
	1	3	29.82	15.60
	2	2	28.78	15.98
Third mode	−1	24	59.87	13.56
	0	7	35.32	15.65
	1	4	31.34	15.65
	2	4	31.81	16.09
Fourth mode	−1	25	61.58	13.62
	0	9	38.78	15.71
	1	6	34.33	16.04
	2	never	—	—

Computational Considerations

Reference 11 gives floating-point operations (flops) count to implement the modal method, called the adjoint method there, and a variation of the Rudisill and Chu method.[12] The latter resembles Nelson's method but requires only the right eigenvector, whose sensitivity is needed, to compute the sensitivity. It is thus superior to Nelson's method that requires both the left and right eigenvectors.[11] Like Nelson's method, however, it requires the factorization of a different matrix for each eigenvector derivative. A flop count comparison is made next between the methods in Ref. 11, Nelson's method, and the method of the present paper. The most advantageous feature of the family of modal methods presented here is that $[A]$, or the reduced $[A]$ when there are zero eigenvalues, needs to be factored only once regardless of the number of eigenvector derivatives and design parameters. For efficiency in computing p right eigenvector derivatives when there are q design parameters, Eq. (13) is implemented as follows, with the most significant term in the flop count for each step given for $[B] = [I]$:

1. Compute the first $\hat{N}$ left and right eigenvectors: $3.5N^2\hat{N}$, (count from Ref. 11 using EISPACK subroutine package also used here).
2. LU decompose $[A]$: $N^3/3$.
3. For $j = 1, \ldots, p$ and $k = 1, \ldots, q$:

Compute $[G_{jk}]\{x_j\} = -[A]_{,k}\{x_j\}$: $pqN^2\kappa$, (κ is a sparsity parameter[11] that is unity for full $[A]_{,k}$).

Compute $\{y_m\}^T([G_{jk}]\{x_j\})$ for $m = 1, \ldots, \hat{N}$: $pqN\hat{N}$.
Compute $\{u_0\} \equiv [A]^{-1}([G_{jk}]\{x_j\})$: pqN^2.
Compute $\{u_i\} \equiv \lambda_j([A]^{-1}\{u_{i-1}\})$ for $i = 1, \ldots, n$: $npqN^2$.
Compute the second summation term in Eq. (13): $pqN\hat{N}$.

With $\hat{N} \ll N$, the total number of flops to implement the nth-order modal method is thus $[3.5\hat{N} + N/3 + pq(\kappa + 1 + n)]N^2$. The flop count for computing the eigenvalues is not considered, as the same count will appear in all of the methods to be compared. The flop counts for the modal method with a full set of left and right eigenvectors used (i.e., $\hat{N} = N$), the modified Rudisill and Chu method,[11] and Nelson's method are

$$[3.5N + pq(\kappa + 2)]N^2, \qquad [2p + pN/3 + pq(\kappa + 1)]N^2$$

and

$$[3.5p + pN/3 + pq(\kappa + 1)]N^2$$

respectively. The $n = 1$ method is clearly superior to the $n = -1$ method, i.e., the modal method. Comparison of the nth-order modal method and modified Rudisill and Chu method

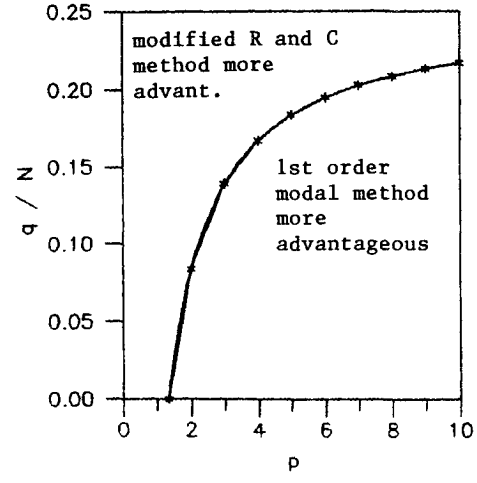

Fig. 2 Regions where the modified Rudisill and Chu method or first-order modal method is superior for $z = 0.25$; Eq. (42).

reveals that the former is superior to the latter, and hence Nelson's method, when

$$nq/N < z - 1/3p; \quad z \equiv 1/3 + (2 - 3.5e)/N, \quad e \equiv \hat{N}/p \quad (42)$$

where e is a measure of the number of modes required for convergence. For the two self-adjoint examples with $N = 20$ and 40 in this paper, e was found to be 1.5 when a derivative of the fourth mode (i.e., $p = 4$) was required (Tables 1 and 2). Inequality (42) is represented in Fig. 2 for $n = 1$ and $z = 0.25$, which is expected to be a conservative value (if e had a value of 3.0 for $N = 100$, e.g., then $z = 0.25$ and z increases for larger system sizes, which makes the nth-order modal method more advantageous). If only the first eigenvector derivative is of interest, the modified Rudisill and Chu method performs better. On the other hand, if, e.g., the first three eigenvector derivatives are required, the first-order modal method performs better, provided the number of design variables q is smaller than 14% of N, the system size. This limit on q increases to 20% of N for seven eigenvector derivatives. On the other hand, the zeroth-order method (i.e., the so-called modified modal method) may perform better than the first-order method for q greater than some value. This depends on the relation between $\hat{N}$ and n.

Concluding Remarks

In the paper, a new family of modal methods for computing eigenvector derivatives in non-self-adjoint systems with zero eigenvalues has been presented. For a given problem, one of the methods in the family requires the smallest number of modes and shortest time to converge. The methods, when applied to systems with no zero eigenvalues, are superior to the modified[11] Rudisill and Chu method and Nelson's method when more than one eigenvector derivative is to be computed. The method is applied, for the first time, to an example system with zero eigenvalues and convergence with a truncated set of modes is demonstrated.

References

[1]Nelson, R. B., "Simplified Calculation of Eigenvector Derivatives," *AIAA Journal*, Vol. 14, No. 9, 1976, pp. 1201–1205.

[2]Adelman, H. M., and Haftka, R. T., "Sensitivity Analysis of Discrete Structural Systems," *AIAA Journal*, Vol. 24, No. 5, 1986, pp. 823–832.

[3]Sutter, T. R., Camarda, C. J., Walsh, J. L., and Adelman, H. M., "Comparison of Several Methods for Calculating Vibration Mode Shape Derivatives," *AIAA Journal*, Vol. 26, No. 12, 1988, pp. 1506–1511.

[4]Fox, R. L., and Kapoor, M. P., "Rates of Change of Eigenvalues and Eigenvectors," *AIAA Journal*, Vol. 6, No. 12, 1968, pp. 2426–2429.

[5]Rogers, L. C., "Derivatives of Eigenvalues and Eigenvectors," *AIAA Journal*, Vol. 8, No. 5, 1970, pp. 943,944.

[6]Plaut, R.H., and Huseyin, K., "Derivatives of Eigenvalues and Eigenvectors in Non-Self-Adjoint Systems," *AIAA Journal*, Vol. 11, No. 2, 1973, pp. 250,251.

[7]Wang, B. P., "An Improved Approximate Method for Computing Eigenvector Derivatives," AIAA/ASME/ASCE/AHS 26th Structures, Structural Dynamics, and Materials Conf., Orlando, FL, April 1985.

[8]Lim, K. B., Junkins, J. L., and Wang, B. P., "Re-examination of Eigenvector Derivatives," *Journal of Guidance, Control, and Dynamics,* Vol. 10, No. 6, 1987, pp. 581–587.

[9]Craig, R. R., *Structural Dynamics*, Wiley, New York, 1981, Chap. 15.

[10]Akgün, M. A., "A New Family of Mode-Superposition Methods for Response Calculations," *Journal of Sound and Vibration*, Vol. 167, No. 2, 1993, pp. 289–302.

[11]Murthy, D. V., and Haftka, R. T., "Derivatives of Eigenvalues and Eigenvectors of a General Complex Matrix," *International Journal for Numerical Methods in Engineering*, Vol. 26, No. 2, 1988, pp. 293–311.

[12]Rudisill, C. S., and Chu, Y., "Numerical Methods for Evaluating the Derivatives of Eigenvalues and Eigenvectors," *AIAA Journal*, Vol. 13, No. 6, 1975, pp. 834–837.