AIAA JOURNAL
Vol. 35, No. 2, February 1997

Accelerated Iterative Procedure for Calculating Eigenvector Derivatives

Ouqi Zhang* and Aspasia Zerva†
Drexel University, Philadelphia, Pennsylvania 19104

An iterative method for the computation of eigenvector derivatives of real-valued, symmetric systems has been proposed. The process is shown to converge to the exact solution with any initial values, and it is numerically stable. However, since the convergence rate is determined by the ratios of the eigenvalues, the convergence rate will be prohibitively slow when the ratio is close to 1. Two efficient accelerated algorithms are presented. Numerical examples show that when the eigenvalue ratio is greater than 0.70, the computational effort in the iterative process is drastically reduced. The procedure incorporates the method of frequency shift to deal with a singular stiffness matrix that also provides an additional improvement for the convergence. When the eigenvalue ratios are extremely close to 1, the second generalized stiffness matrix inverse is suggested to reestablish a high convergence rate. The procedure can be used as an exact as well as an approximate method. It requires no more eigenvectors than those whose derivatives are to be calculated and can be applied to systems with repeated eigenvalues.

Nomenclature

A = $\Psi_u \tilde{\Lambda}_u^{-1} \Psi_u^t = K^{-1} - \Psi_a \tilde{\Lambda}_a^{-1} \Psi_a^t$
C_j = modal participation coefficient of ϕ_j^t, defined by $\phi_j^t = \Psi C_j$
C_{ja} = modal participation coefficient of V_{ja}, defined by $V_{ja} = \Psi_a C_{ja}$
C_{ju} = modal participation coefficient of V_{ju}, defined by $V_{ju} = \Psi_u C_{ju}$
F_j = $-[K^t - \lambda_j^t M - \lambda_j M^t]$
I = unit matrix
K, M = real, symmetric stiffness and mass matrices
K^t, M^t = derivatives of K and M
n = system dimension
q = number of available eigenvalues and eigenvectors, or number of eigenvector derivatives demanded
r_k = residual of $(V_{ju})_k$ used as searching direction at kth iteration
U_j = particular solution of Eq. (6)
V_{ja} = component of ϕ_j^t in the range of Ψ_a, $\Psi_a C_{ja}$
V_{ju} = component of ϕ_j^t in the range of Ψ_u, $\Psi_u C_{ju}$
$(V_{ju})_k$ = kth iterative solution for V_{ju}
Z_i = modal participation coefficients of ϕ_i^t in the range of Φ_j, defined by $\phi_i^t = U_i + \Phi_j Z_i$, $i = j, j+1, \ldots, j+m-1; m > 1$; λ_j is an m repeated eigenvalue
α_k = acceleration parameter used in algorithm 1
$(\Delta V_{ju})_k$ = the error of $(V_{ju})_k$, $(V_{ju})_k - V_{ju}$
Λ = diagonal $(\lambda_1, \lambda_2, \ldots, \lambda_n)$, eigenvalue matrix; $\lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_q < \lambda_{q+1} \leq \cdots \leq \lambda_n$
Λ_a = diagonal $(\lambda_1, \lambda_2, \ldots, \lambda_q)$
$\tilde{\Lambda}_a$ = diagonal $(\tilde{\lambda}_1, \tilde{\lambda}_2, \ldots, \tilde{\lambda}_q)$
Λ_u = diagonal $(\lambda_{q+1}, \lambda_{q+2}, \ldots, \lambda_n)$
$\tilde{\Lambda}_u$ = diagonal $(\tilde{\lambda}_{q+1}, \tilde{\lambda}_{q+2}, \ldots, \tilde{\lambda}_n)$
λ_j = jth eigenvalue
$\tilde{\lambda}_j$ = jth effective eigenvalue, $\lambda_j - \lambda_0$
λ_j^t = derivative of the jth eigenvalue
λ_0 = frequency shift
Ξ_k = acceleration parameter used in algorithm 2, $(\xi_{k1} \;\; \xi_{k2})^t$
σ = allowable relative error of the iterative solution $(V_{ju})_\infty$ defined as $\sigma = \|(V_{ju})_\infty - V_{ju}\| / \|V_{ju}\|$ or allowable relative difference of the adjacent iterative solutions defined as $\sigma = \|(V_{ju})_{k+1} - (V_{ju})_k\| / \|(V_{ju})_k\|$
Φ_j = differentiable eigenvector matrix associated with the jth of $m (m > 1)$ repeated eigenvalues, $[\phi_j, \phi_{j+1}, \phi_{j+2}, \ldots, \phi_{j+m-1}]$
ϕ_j = jth differentiable eigenvector
ϕ_j^t = jth eigenvector derivative, $\Psi C_j = V_{ja} + V_{ju} = \Psi_a C_{ja} + \Psi_u C_{ju}$
Ψ = complete eigenvector matrix, $[\phi_1, \phi_2, \ldots, \phi_n]$
Ψ_a = available eigenvector matrix, $[\phi_1, \phi_2, \ldots, \phi_q]$
Ψ_u = unavailable eigenvector matrix, $[\phi_{q+1}, \phi_{q+2}, \ldots, \phi_n]$
ω_k = acceleration parameter used in algorithm 1
$\| \cdot \|$ = vector or matrix norm

I. Introduction

THE dynamic behavior of a structural system is characterized by its eigendata. The partial derivatives of eigenvalues and eigenvectors of structures with respect to design variables or system parameters have attracted extensive attention for the last two decades because of their various applications, such as optimal dynamic design, parameter identification, model modification, machinery failure diagnosis, and system control.

The most straightforward approach for calculating the derivatives is the finite difference method; besides this, there mainly exist three categories in the literature: the modal method,[1–4] the direct method,[1,5–12] and the iterative method.[12–18] Fox and Kapoor[1] derived the direct and modal methods. Nelson[5] simplified the calculation of the direct method. The eigenvector derivatives with repeated eigenvalues are derived by Ojalvo,[6] Mills-Curran,[7,8] and Dailey.[9] Reference 19 presented reviews for the early work in the area. Reference 11 compared the operation counts of the modal method and the modified Rudisile and Chu's iterative algorithm. The relative efficiencies are surveyed in Ref. 20 for the finite difference method, modal method, Wang's modified modal method, and Nelson's direct method on the basis of central processor seconds.

The calculation of the eigenvector derivatives involves extensive computational effort. The direct method is one of the most efficient methods that produces exact solutions and does not need eigenvectors more than those whose derivatives are to be computed. But because its amount of computational effort is proportional to the

Received May 22, 1996; revision received Sept. 23, 1996; accepted for publication Sept. 24, 1996; also published in *AIAA Journal on Disc*, Volume 2, Number 2.

*Research Assistant, Department of Civil and Architectural Engineering. Member AIAA.
†Associate Professor, Department of Civil and Architectural Engineering.

number of eigenvector derivatives required, the application of the method becomes expensive when many eigenvector derivatives are demanded. On the other hand, the truncated modal method has an insuperable efficiency but suffers a serious accuracy problem. To improve the accuracy of the modal method, Wang[2] proposed a modified modal method, which was extended by Liu et al.[17] and Zhang and Zerva[18] to an iterative algorithm that can be used as an exact method as well as an approximate method and, just like the direct method, does not require additional eigenvalues and eigenvectors. The method assumes that the inverse stiffness matrix exists. Recently, Lin and Lim[21] and Zeng[4] presented an approach to deal with singular stiffness matrices. The convergence rate of the iterative method depends mainly on the ratio of the specified eigenvalue to the lowest unavailable one, and when the ratio approaches 1, the convergence rate of the corresponding eigenvector derivative will reduce quickly and the method becomes more expensive than the direct method.

In this paper, an effective accelerated iterative procedure is presented as an extension of the methods in Refs. 2, 17, 18, and 21. After a brief description of the problem and the basic iterative equation, the method proposed by Lin and Lim[21] and Zeng[4] for dealing with systems with zero eigenvalues is introduced in Sec. III. Two accelerated algorithms are formulated in Sec. IV. Several relative problems are discussed in Sec. V, followed by two illustrative examples in Sec. VI. The paper is closed with a brief summary.

II. Preliminaries

Consider a structural vibration system characterized by the real, symmetric stiffness matrix $K \in R^n \times^n$ and mass matrix $M \in R^n \times^n$, which are functions of the design variables. Assume that both K and M are positive definite (this restriction on the stiffness matrix will be released in Sec. III). The eigenproblem is defined by

$$[K - \lambda M]x = 0 \tag{1}$$

where λ and $x \in R^n$ are the eigenvalue and eigenvector, respectively. If the lth eigenvalue λ_l is distinct, the differentiable eigenvector is uniquely defined by Eq. (1). When λ_l is a repeated eigenvalue with $m(m > 1)$ multiplicity, $\lambda_l = \lambda_{l+1}, \ldots = \lambda_{l+m-1}$, but its derivatives are distinct, the unique differentiable eigenvectors are derived by Ojalvo,[6] Mills-Curran,[7,8] and Dailey.[9] Denoting $X \equiv [x_l, x_{l+1}, \ldots, x_{l+m-1}] \in R^n \times^m$ the nonunique eigenvector submatrix associated with λ_l, and $\Phi \equiv [\phi_l, \phi_{l+1}, \ldots, \phi_{l+m-1}] \in R^n \times^m$ the corresponding unique differentiable eigenvector submatrix, there must exist a nonsingular transformation matrix $T \equiv [t_l, t_{l+1}, \ldots, t_{l+m-1}] \in R^m \times^m$ satisfying $\Phi = XT$. Let X and Φ be normalized by

$$X^t M X = I; \qquad \Phi^t M \Phi = T^t X^t M X T = T^t T = I \tag{2}$$

where I stands for identity matrix and the superscript t indicates matrix transpose. It is seen that T is orthogonal. Differentiating Eq. (1) with the differentiable eigenvector ϕ_j, one obtains

$$[K - \lambda_j M]\phi_j' + [K - \lambda_j M]'\phi_j = [K - \lambda_j M]\phi_j' + [K' - \lambda_j' M - \lambda_j M']\phi_j = 0 \qquad j = l, l+1, \ldots, l+m-1 \tag{3}$$

where the superscript $'$ stands for the first-order derivative with respect to the design variable. Substituting $\phi_j = Xt_j$, we have (note $\phi_j' \neq Xt_j'$)

$$[K - \lambda_j M]\phi_j' = -[K' - \lambda_j' M - \lambda_j M']Xt_j \qquad j = l, l+1, \ldots, l+m-1 \tag{4}$$

Premultiplying it with X^t, and noting that $X^t[K - \lambda_j M] = 0$ and $X^t M X = I$, we have

$$(X^t[K' - \lambda_j M']X)t_j = \lambda_j' t_j \qquad j = l, l+1, \ldots, l+m-1 \tag{5}$$

It is recognized that this is an $m \times m$ standard eigenproblem. Since its eigenvalues λ_j' are assumed to be distinct, t_j will be unique, and both can be obtained by solving this problem. Substituting λ_j' and $\phi_j = Xt_j$ obtained from Eq. (5) back into Eq. (3), we have

$$[K - \lambda_j M]\phi_j' = -[K' - \lambda_j' M - \lambda_j M']\phi_j \equiv F_j \phi_j \qquad j = l, l+1, \ldots, l+m-1 \tag{6}$$

where $F_j \equiv -[K' - \lambda_j' M - \lambda_j M']$. Note that the coefficient matrix $[K - \lambda_j M]$ has rank $n - m$ and a kernel spanned by Φ. Therefore, the general solution of ϕ_j' can be expressed by

$$\phi_j' = U_j + \Phi Z_j \qquad j = l, l+1, \ldots, l+m-1 \tag{7}$$

where $U_j \in R^n$ is a particular solution of Eq. (6) and $Z_j \in R^m$ will be determined by U_j and the differentiation of Eqs. (2) and (6) (Refs. 6–8). The expressions for Z_j adopted from Ref. 7 are as follows: for $i = j$,

$$z_{jj} = -\phi_j^t(\tfrac{1}{2}M'\phi_j + MU_j) \qquad j = l, l+1, \ldots, l+m-1 \tag{8}$$

for $i \neq j$,

$$z_{ij} = \frac{\phi_i^t(K'' - 2\lambda_j' M' - \lambda_j M'')\phi_j + 2\phi_i^t F_j U_j}{2(\lambda_j - \lambda_i)} \qquad i, j = l, l+1, \ldots, l+m-1; \quad i \neq j \tag{9}$$

where double prime indicates the second-order derivatives.

In this paper, we address the computation of the particular solution U_j. In the derivation we assumed that the eigenvalue derivatives are distinct, but they can also be repeated, because the particular solution U_j has exactly the same expression and will be solved using the same method whether the eigenvalue derivatives are distinct or not. The eigenvalue and eigenvector derivatives with repeated eigenvalue derivatives are derived in Ref. 22.

Modal Method

Let $\Psi \in R^n \times^n$ be the eigenvector matrix normalized by $\Psi^t M \Psi = I$, and express $\phi_j' = \Psi C_j$ with $C_j \in R^n$ to be determined; the substitution of $\phi_j' = \Psi C_j$ into Eqs. (7) and (6) leads to

$$c_{ij} = \frac{\phi_i^t F_j \phi_j}{\lambda_i - \lambda_j} \qquad j = l, l+1, \ldots, l+m-1 \qquad i \neq j \tag{10}$$

where we assume $\lambda_l = \lambda_{l+1}, \ldots = \lambda_{l+m-1}$, $m \geq 1$; for $i = j$, $c_{ij} = z_{ij}$ that have been given in Eqs. (8) and (9).

Suppose that the first $q (q \ll n)$ eigenvector derivatives are to be calculated, and these q eigenvectors and eigenvalues have been obtained. Let the differentiable eigenvector matrix Ψ and eigenvalue matrix $\Lambda \in R^n \times^n$ be partitioned into

$$\Psi = [\Psi_a \quad \Psi_u]; \qquad \Psi_a \equiv [\phi_1, \phi_2, \ldots, \phi_q] \qquad \Psi_u \equiv [\phi_{q+1}, \phi_{q+2}, \ldots, \phi_n] \tag{11}$$

$$\Lambda = \begin{bmatrix} \Lambda_a & 0 \\ 0 & \Lambda_u \end{bmatrix}; \qquad \Lambda_a \equiv \mathrm{diag}(\lambda_1, \lambda_2, \ldots, \lambda_q) \qquad \Lambda_u \equiv \mathrm{diag}(\lambda_{q+1}, \lambda_{q+2}, \ldots, \lambda_n) \tag{12}$$

where

$$\lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_q < \lambda_{q+1} \leq \cdots \leq \lambda_n \tag{13}$$

Correspondingly, let

$$\phi_j' = \Psi C_j = V_{ja} + V_{ju}; \qquad V_{ja} \equiv \Psi_a C_{ja} \qquad V_{ju} \equiv \Psi_u C_{ju} \tag{14}$$

where

$$C_{ja} \equiv (c_{1j}, c_{2j}, \ldots, c_{qj})^t; \qquad C_{ju} \equiv (c_{q+1,j}, c_{q+2,j}, \ldots, c_{n,j})^t \tag{15}$$

The simple truncated modal method approximates ϕ_j^t by $\phi_j^t \approx V_{ja}$. This is a very rough approximate solution because it neglects V_{ju} completely, but it is an exact solution for V_{ja}.

Iterative Method

The iterative method used in the proposed procedure was derived originally in Ref. 18. Since V_{ja} can be easily and efficiently calculated by the modal method, the iterative method considers the determination of the term V_{ju} only. Let

$$A \equiv \Psi_u \Lambda_u^{-1} \Psi_u^t \tag{16}$$

which, because $K^{-1} = \Psi \Lambda^{-1} \Psi^t = \Psi_a \Lambda_a^{-1} \Psi_a^t + \Psi_u \Lambda_u^{-1} \Psi_u^t$, is actually obtained from

$$A = K^{-1} - \Psi_a \Lambda_a^{-1} \Psi_a^t \tag{17}$$

Noting $\Psi_u \Lambda_u^{-1} \Psi_u^t [K - \lambda_j M] \Psi_a C_{ja} = 0$, and premultiplying Eq. (6) with A, we obtain

$$\begin{aligned} A[K - \lambda_j M]\phi_j &= \Psi_u \Lambda_u^{-1} \Psi_u^t [K - \lambda_j M](V_{ja} + V_{ju}) \\ &= \Psi_u \Lambda_u^{-1} \Psi_u^t [K - \lambda_j M](\Psi_a C_{ja} + \Psi_u C_{ju}) \\ &= \Psi_u \Lambda_u^{-1} \Psi_u^t [K - \lambda_j M] \Psi_u C_{ju} \\ &= \Psi_u C_{ju} - \lambda_j \Psi_u \Lambda_u^{-1} \Psi_u^t M \Psi_u C_{ju} = V_{ju} - \lambda_j A M V_{ju} \\ &= [I - \lambda_j AM] V_{ju} = AF_j \phi_j \end{aligned} \tag{18}$$

i.e.,

$$[I - \lambda_j AM] V_{ju} = AF_j \phi_j \tag{19}$$

It will be shown in the Appendix that the coefficient matrix $[I - \lambda_j AM]$ is nonsingular. Therefore, Eq. (19) can be solved directly as a general linear system, but we will treat it only as an iterative process:

$$(V_{ju})_k = AF_j \phi_j + \lambda_j AM(V_{ju})_{k-1} \qquad j \leq q;\ k = 1, 2, \ldots \tag{20}$$

Equation (20) will be referred to hereafter as the basic iterative equation. It can be shown that after $k (k \geq 1)$ iterations,

$$(V_{ju})_k = \sum_{i=q+1}^{n} \left[1 - \left(\frac{\lambda_j}{\lambda_i} \right)^k \right] \frac{\phi_i^t F_j \phi_j}{\lambda_i - \lambda_j} \phi_i + \sum_{i=q+1}^{n} \left(\frac{\lambda_j}{\lambda_i} \right)^k \phi_j^t M (V_{ju})_0 \phi_i \qquad j \leq q \tag{21}$$

where $(V_{ju})_0$ stands for the initial value. The comparison of Eq. (21) with the exact solution, Eq. (10), shows that the term $(\lambda_j / \lambda_i)^k$ represents the error because of the ith unknown eigenvector. When k tends to infinity, $(\lambda_j / \lambda_i)^k$ vanishes because $\lambda_j / \lambda_i < 1$, and $(V_{ju})_k$ converges to the exact solution with any initial value. Equation (21) also suggests that $(V_{ju})_0$ can be set equal to zero. Note that in each iteration, the roundoff error in the subspace spanned by the lower eigenvectors Ψ_a will be automatically wiped out, which results in a very stable iterative process.

III. Generalized Stiffness Inverse and Systems with Zero Eigenvalues

In the previous sections, we assumed that the stiffness matrix is nonsingular. In this section, we describe a variation of the iterative method to deal with singular stiffness matrices.

The eigenvector derivatives with zero eigenvalues were derived by Akgün.[3] From Akgün's derivation we concluded that the solution of V_{ju} can still be expressed by Eq. (10). Hence, we only need to modify Eq. (19) so that it defines the desired solution using the given singular stiffness matrix.

Following the method proposed by Lin and Lim[21] and Zeng,[4] we construct

$$\tilde{A} \equiv \Psi_u \tilde{\Lambda}_u^{-1} \Psi_u^t = \tilde{K}^{-1} - \Psi_a \tilde{\Lambda}_a^{-1} \Psi_a^t \tag{22}$$

where

$$\begin{aligned} &\tilde{K}^{-1} = [K - \lambda_0 M]^{-1}; \qquad \tilde{\Lambda}_a = \Lambda_a - \lambda_0 \\ &\tilde{\Lambda}_u = \Lambda_u - \lambda_0; \qquad \lambda_0 \neq \lambda_j, \qquad j = 1, \ldots, n \end{aligned} \tag{23}$$

where λ_0 is the frequency shift. We call $\tilde{K}^{-1}$ with $\lambda_0 > 0$ the generalized stiffness inverse that exists whether the original stiffness matrix is singular or not. Note that it preserves the sparsity, if any, of the original system. Premultiplication of Eq. (6) with $\tilde{A}$ gives

$$\begin{aligned} \tilde{A}[K - \lambda_j M] V_j &= \tilde{A}[(K - \lambda_0 M) - (\lambda_j - \lambda_0) M] V_j \\ &= V_{ju} - \tilde{\lambda}_j \tilde{A} M V_{ju} = \tilde{A} F_j \phi_j \end{aligned} \tag{24}$$

or

$$[I - \tilde{\lambda}_j \tilde{A} M] V_{ju} = \tilde{A} F_j \phi_j \tag{25}$$

The differences between Eqs. (19) and (25) are that A is replaced by $\tilde{A}$ and λ_j replaced by $\tilde{\lambda}_j$. Note that Eq. (25) defines the exact solution for V_{ju}.

With this modification, the effective eigenvalue ratios are changed. Let $(V_{ju})_0 = 0$, $|\tilde{\lambda}_j / \tilde{\lambda}_i| < 1$, $j \leq q < i < n$, after $k (k \geq 1)$ iterations, Eq. (21) becomes

$$\begin{aligned} (V_{ju})_k &= \sum_{i=q+1}^{n} [1 - (\tilde{\lambda}_j / \tilde{\lambda}_i)^k] \frac{\phi_i^t F_j \phi_j}{\tilde{\lambda}_i - \tilde{\lambda}_j} \phi_i \\ &= \sum_{i=q+1}^{n} [1 - (\tilde{\lambda}_j / \tilde{\lambda}_i)^k] \frac{\phi_i^t F_j \phi_j}{\lambda_i - \lambda_j} \phi_i \qquad j \leq q \end{aligned} \tag{26}$$

If only the effective eigenvalue ratios $|\tilde{\lambda}_j / \tilde{\lambda}_i| < 1$, $j \leq q < i \leq n$, the basic iterative Eq. (20) will converge to the specific eigenvector derivative. The restriction for the frequency shift to satisfy this inequality is $\lambda_0 < 0.5(\lambda_1 + \lambda_{q+1})$.

Equation (20) shows that the error is ultimately controlled by $\lambda_q / \lambda_{q+1}$. The effective eigenvalue ratio $\tilde{\lambda}_q / \tilde{\lambda}_{q+1}$ in Eq. (26) is reduced by increasing λ_0; however, increasing λ_0 will increase $\tilde{\lambda}_1 / \tilde{\lambda}_{q+1}$. It can be seen that when $\lambda_0 = 0.5(\lambda_1 + \lambda_{q+1})$, the effective eigenvalue ratio $\tilde{\lambda}_q / \tilde{\lambda}_{q+1}$ is decreased to the limit, because $|\tilde{\lambda}_1 / \tilde{\lambda}_{q+1}|$ will be equal to 1.

Considering the efficiency of the iterative method and assuming λ_{q+1} is given, the writers suggest to set $\lambda_0 > \lambda_1$ satisfying $|\tilde{\lambda}_j / \tilde{\lambda}_i| < 0.95$, $j \leq q < i \leq n$, which yields

$$\max \left[\frac{\lambda_1, (\lambda_q - 0.95\lambda_{q+1})}{0.05} \right] < \lambda_0 < \frac{(\lambda_1 + 0.95\lambda_{q+1})}{1.95} \tag{27}$$

If this condition cannot be satisfied, a second generalized stiffness inverse may be necessary. In this case, the available eigenvectors can be divided into two sets, and two generalized stiffness inverses are formulated for calculating the corresponding set of the eigenvector derivatives. If the original stiffness matrix is singular, λ_1 in Eq. (27) is taken as the first nonzero eigenvalue.

Frequency shift has been used to accelerate the convergence of the computation of eigenvector derivatives.[4, 15] When λ_0 is properly selected, and the available eigenvalues are clustered, this generalized stiffness inverse has a substantial effect on the rate of convergence. Note that Eq. (27) defines only the effective range for λ_0. The determination of a proper λ_0 will be given in Sec. V.

For simplicity in notation, we use K^{-1} for $\tilde{K}^{-1}$ and λ_j for $\tilde{\lambda}_j$ in the remainder of this paper. In practice, however, only the generalized stiffness inverse with $\lambda_0 > \lambda_1$, $\lambda_0 \neq \lambda_i$, $i = 1, 2, \ldots, n$, should be used.

IV. Acceleration Techniques for Iterative Process

The degree of efficiency of the proposed procedure depends mainly on the convergence rate of the iterative process. Equation (21) shows that if the eigenvalue ratio $\lambda_j / \lambda_{q+1}$ is small, the basic iterative method converges rapidly. However, when $\lambda_j / \lambda_{q+1}$ is close to 1, the convergence rate will be prohibitively slow. In this section we present two simple but effective acceleration techniques to increase the rate of convergence.

Construct the following scalar function with the vector variable $y \in R^n$:

$$S(y) = \tfrac{1}{2} y^t M[I - \lambda_j AM]y - y^t AF_j\phi_j \tag{28}$$

Note that the matrix $M[I - \lambda_j AM]$ is symmetric positive definite (see the Appendix for details). It is known[23] that the minimum value of $S(y)$ is achieved by setting $y = [I - \lambda_j AM]^{-1} AF_j\phi_j = V_{ju}$. That is, the desired solution of V_{ju} can be obtained by minimizing $S(y)$. Provided that a good searching direction r_k has been determined at the kth iteration, then

$$(V_{ju})_k := (V_{ju})_k + \omega_k r_k \tag{29}$$

will produce a better estimation of the solution, where ω_k is a scalar parameter chosen so as to minimize $S[(V_{ju})_k + \omega_k r_k]$. Substitution of Eq. (29) into Eq. (28) and minimization of Eq. (28) yield

$$\omega_k = \frac{(r_k^t M r_k)}{\left[r_k^t M(I - \lambda_j AM) r_k\right]} \tag{30}$$

The residual of $(V_{ju})_k$ is a good searching direction r_k. Let $(V_{ju})_k = V_{ju} + (\Delta V_{ju})_k$; from Eq. (21), we obtain the error of $(V_{ju})_k$

$$(\Delta V_{ju})_k = \sum_{i=1,\, i\neq j}^{n} \left[-\left(\frac{\lambda_j}{\lambda_i}\right)^k \right] \frac{\phi_i^t F_j \phi_j}{\lambda_i - \lambda_j} \phi_i$$

$$= -\Psi_u \left[\lambda_j \Lambda_u^{-1}\right]^k [\Lambda_u - \lambda_j]^{-1} \Psi_u^t F_j \phi_j \tag{31}$$

from which and from Eq. (16) we have

$$\lambda_j AM(\Delta V_{ju})_k = (\Delta V_{ju})_{k+1} \tag{32}$$

and, noting $[I - \lambda_j AM]V_{ju} = AF_j\phi_j$, we obtain the residual of $(V_{ju})_k$ as

$$\begin{aligned} &AF_j\phi_j - [I - \lambda_j AM](V_{ju})_k \\ &= AF_j\phi_j - [I - \lambda_j AM]\left\{V_{ju} + (\Delta V_{ju})_k\right\} \\ &= -[I - \lambda_j AM](\Delta V_{ju})_k = (\Delta V_{ju})_{k+1} - (\Delta V_{ju})_k \\ &= -\Psi_u \left[\lambda_j \Lambda_u^{-1} - I\right]\left[\lambda_j \Lambda_u^{-1}\right]^k [\Lambda_u - \lambda_j]^{-1} \Psi_u^t F_j \phi_j \end{aligned} \tag{33}$$

From the comparison of Eqs. (31) and (33), it is seen that Eq. (33) gives an approximation for $(\Delta V_{ju})_k$. In the special case of $M = I$, the procedure just above described is, basically, the method of steepest descent.

Also from Eq. (31), we observe that there may exist a strong trend of monotonic convergence, and a multiplier will provide additional improvement:

$$(V_{ju})k := \alpha_k (V_{ju})_k \tag{34}$$

where the scalar multiplier α_k is, again, obtained from the minimization of Eq. (28):

$$\alpha_k := \frac{(r_k^t MAF_j\phi_j)}{\left[r_k^t M(I - \lambda_j AM) r_k\right]} \tag{35}$$

In summary, we have the following accelerated iterative procedure:

$$(V_{ju})_0 = 0$$

For $k = 1, 2, \ldots,$

$$\begin{aligned} &(V_{ju})_k = AF_j\phi_j + \lambda_j AM(V_{ju})_{k-1} \\ &r_k = AF_j\phi_j - (I - \lambda_j AM)(V_{ju})_k \\ &\omega_k = (r_k^t M r_k)/[r_k^t M(I - \lambda_j AM) r_k] \\ &\alpha_k = (r_k^t MAF_j\phi_j)/[r_k^t M(I - \lambda_j AM) r_k] \\ &(V_{ju})_k := \alpha_k[(V_{ju})_k + \omega_k r_k] \end{aligned}$$

This leads to the following algorithm.

Algorithm 1:

$$b := AF_j\phi_j; \qquad \alpha := 0$$

For $k = 1, 2, \ldots,$

$$\begin{aligned} &p_1 := (1 - \alpha)b + \alpha[p_2 + \omega(p_3 - p_2)] \\ &p_2 := b + \lambda_j AMp_1 \\ &r := p_2 - p_1; \text{ if } \|r\|\|p_2\| < \sigma, \text{ stop and let } V_{ju} = p_2; \\ &\quad \text{otherwise} \\ &p_3 := b + \lambda_j AMp_2 \\ &\omega := (r^t Mr)/[r^t M(r - p_3 + p_2)] \\ &\alpha := (r^t Mb)/[r^t M(r - p_3 + p_2)] \\ &k := k + 1 \end{aligned}$$

where σ is a given small number, $\sigma > 0$.

It is noted that, in each iteration, the major computational work amounts in computing $\lambda_j AMp_1$ and $\lambda_j AMp_2$; the term Mr will be available during the computation of these two products. That is, in each iteration, the preceding algorithm needs twice as much computation as the basic iterative method.

Equation (29), rewritten as

$$\begin{aligned} (V_{ju})_k &:= (V_{ju})_k + \omega_k r_k \\ &= \omega_k[AF_j\phi_j + \lambda_j AM(V_{ju})_k] + (1 - \omega_k)(V_{ju})_k \end{aligned} \tag{36}$$

implies that $\omega_k + (1 - \omega_k) = 1$, a seemingly necessary requirement since when $(V_{ju})_k = V_{ju}$, $AF_j\phi_j + \lambda_j AM(V_{ju})_k = V_{ju}$. However, during the iteration, $(V_{ju})_k \neq V_{ju}$, and thus it is slightly overrestricted. We release this restriction in algorithm 1 by the multiplier α_k. A more sound alternative would be the introducionof two variable parameters

$$(V_{ju})_k := \xi_{k1}(V_{ju})_k + \xi_{k2} r_k = [(V_{ju})_k, r_k]\Xi_k \tag{37}$$

where $\Xi_k \equiv (\xi_{k1}\ \xi_{k2})^t \in R^2$ is also determined from the minimization of Eq. (28),

$$\begin{aligned} \Xi = &\left\{[(V_{ju})_k, r_k]^T M(I - \lambda_j AM)[(V_{ju})_k, r_k]\right\}^{-1} \\ &\times [(V_{ju})_k, r_k]^T M[(V_{ju})_k, r_k] \end{aligned} \tag{38}$$

This leads to the following algorithm.

Algorithm 2:

$$b := AF_j\phi_j; \qquad \Xi = (\xi_1 \quad \xi_2)^t := (0 \quad 0)^t$$

For $k = 1, 2, \ldots,$

$$\begin{aligned} &p_1 := b + \xi_1(p_2 - b) + \xi_2(p_3 - b) \\ &p_2 := b + \lambda_j AMp_1 \\ &\rho := [p_1, p_2]; \text{ if } \|p_2 - p_1\|\|p_2\| < \sigma, \text{ stop and let } V_{ju} = p_2; \\ &\quad \text{otherwise} \\ &p_3 := b + \lambda_j AMp_2 \\ &\beta := [(p_1 - p_2 + b), (p_2 - p_3 + b)] \\ &\Xi := [\rho^t M\beta]^{-1}\rho^t Mb \\ &k := k + 1 \end{aligned}$$

In each iteration algorithm 2 also requires twice the computational effort compared with the basic iterative method. The counterpart of the multiplier $\alpha_k = (r_k^t MAF_j\phi_j)/[r_k^t M(I - \lambda_j AM) r_k]$ in algorithm 1 is no longer necessary. In algorithm 2, the residual, $[AF_j\phi_j - (I - \lambda_j AM)(V_{ju})_k]$, is replaced by $AF_j\phi_j + \lambda_j AM(V_{ju})_k$; it can be shown this makes no difference.

V. Discussion

The most reliable convergence criterion is that the relative error between the exact and the iterative solutions is less than a given small number,

$$\frac{\|(V_{ju})_k - V_{ju}\|}{\|V_{ju}\|} < \sigma \tag{39}$$

Because the exact solution is not known, an alternative criterion has to be adopted such as the relative difference of the adjacent iterative results:

$$\frac{\|(V_{ju})_{k+1} - (V_{ju})_k\|}{\|(V_{ju})_k\|} < \sigma \tag{40}$$

This is also suggested in the accelerated iterative algorithms. Generally, a small relative difference does not necessarily mean a small difference to the exact solution. However, for the proposed method the norm of $(V_{ju})_{k+1} - (V_{ju})_k$ can be directly related to the norm of $(V_{ju})_k - V_{ju}$. Denoting $(V_{ju})_{k+1} = AF_j\phi_j + \lambda_j AM(V_{ju})_k$, as defined in the basic iterative equation, and noting $(V_{ju})_k = V_{ju} + (\Delta V_{ju})_k$, from the accelerated algorithms we have

$$r_k = (V_{ju})_{k+1} - (V_{ju})_k = (\Delta V_{ju})_{k+1} - (\Delta V_{ju})_k$$
$$= -[I - \lambda_j AM](\Delta V_{ju})_k = -[I - \lambda_j AM]\{(V_{ju})_k - V_{ju}\} \tag{41}$$

Hence,

$$(V_{ju})_k - V_{ju} = -[I - \lambda_j AM]^{-1}\{(V_{ju})_{k+1} - (V_{ju})_k\} \tag{42}$$

and

$$\|(V_{ju})_k - V_{ju}\| \le \|[I - \lambda_j AM]^{-1}\|\,\|(V_{ju})_{k+1} - (V_{ju})_k\| \tag{43}$$

which shows that Eq. (40) is a reliable convergence criterion. Using two-norm, when $M = I$, $\|[I - \lambda_j AM]^{-1}\| = 1/(1 - \lambda_j/\lambda_{q+1})$.

We show some details about how the accelerated algorithms work. Let $(V_{ju})_k$ be the left-hand side of the first equation in algorithm 1 at kth iteration, and still denote $(V_{ju})_{k+1} = AF_j\phi_j + \lambda_j AM(V_{ju})_k$. Let $\alpha_k \equiv 1$; algorithm 1 gives the residual at $(k+1)$th iteration as

$$r_{k+1} = AF_j\phi_j - [I - \lambda_j AM]\{AF_j\phi_j + \lambda_j AM[(V_{ju})_k + \omega_k r_k]\}$$
$$= \lambda_j AMAF_j\phi_j - [I - \lambda_j AM]\lambda_j AM(V_{ju})_k$$
$$- \omega_k[I - \lambda_j AM]\lambda_j AMr_k$$
$$= \lambda_j AMAF_j\phi_j - \lambda_j AM(V_{ju})_k + \lambda_j AM$$
$$\times\{(V_{ju})_{k+1} - AF_j\phi_j\} - \omega_k[I - \lambda_j AM]\lambda_j AMr_k$$
$$= \lambda_j AM\{(V_{ju})_{k+1} - (V_{ju})_k\} - \omega_k[I - \lambda_j AM]\lambda_j AMr_k$$
$$= [(1 - \omega_k)I + \omega_k\lambda_j AM]\lambda_j AMr_k$$
$$= \Psi_u[(1 - \omega_k)I + \omega_k\lambda_j\Lambda_u^{-1}][\lambda_j\Lambda_u^{-1}]\Psi_u^t Mr_k \tag{44}$$

Taking advantage of Eq. (41), we can write Eq. (44) as

$$(\bar{V}_{ju})_{k+1} - V_{ju} = \Psi_u[(1 - \omega_k)I + \omega_k\lambda_j\Lambda_u^{-1}]$$
$$\times[\lambda_j\Lambda_u^{-1}]\Psi_u^t M\{(V_{ju})_k - V_{ju}\} \tag{45}$$

in which $(\bar{V}_{ju})_{k+1}$ is the left-hand side of the first equation in algorithm 1 at $(k+1)$th iteration. Furthermore, if we let $\omega_k \equiv 1$, algorithm 1 will regress to the basic iterative method and the right-hand side of Eq. (45) becomes

$$(\bar{V}_{ju})_{k+1} - V_{ju} = \Psi_u[\lambda_j\Lambda_u^{-1}][\lambda_j\Lambda_u^{-1}]\Psi_u^t M\{(V_{ju})_k - V_{ju}\} \tag{46}$$

The comparison of Eqs. (45) and (46) shows 1) when $1 < \omega_k < 1/(1 - \lambda_j/\lambda_{q+1})$, the maximum value in the diagonal matrix $[(1 - \omega_k)I + \omega_k\lambda_j\Lambda_u^{-1}]$ reduces from λ_j/λ_{q+1} to $(1 - \omega_k) + \omega_k\lambda_j/\lambda_{q+1}$, but the absolute minimum element increases from λ_j/λ_n to $|(1 - \omega_k) + \omega_k\lambda_j/\lambda_n| \approx \omega_k - 1$, and 2) these increased elements are eliminated effectively by the multiplication with $[\lambda_j\Lambda_u^{-1}]$. This is the reason for the presence of $(V_{ju})_k = AF_j\phi_j + \lambda_j AM(V_{ju})_{k-1}$ in the proposed algorithms.

It can be seen in Eq. (45) that the acceleration parameter ω_k works in a way somewhat similar to the frequency shift in Eq. (25), and for an arbitrarily given ω_k within $1 < \omega_k < 2$, r_k will tend to zero and the process converges. The use of the proposed accelerated algorithms can yield very large values for ω_k. From this point of view, a system with a larger dimension may converge a little faster because it is more likely for the acceleration parameter ω_k to be large.

The number of iterations required to satisfy a given convergence criterion can be well estimated for the basic iterative Eq. (20). From Eqs. (31), (21), and (14), one obtains

$$(V_{ju})_k - V_{ju} = (\Delta V_{ju})_k = -\Psi_u\left[\lambda_j\Lambda_u^{-1}\right]^k[\Lambda_u - \lambda_j]^{-1}\Psi_u^t F_j\phi_j$$
$$= -\Psi_u\left[\lambda_j\Lambda_u^{-1}\right]^k\Psi_u^t M\Psi_u[\Lambda_u - \lambda_j]^{-1}\Psi_u^t F_j\phi_j$$
$$= -\Psi_u\left[\lambda_j\Lambda_u^{-1}\right]^k\Psi_u^t MV_{ju} = -\Psi_u\left[\lambda_j\Lambda_u^{-1}\right]^k\Psi_u^t M\Psi_u C_{ju}$$
$$= -\Psi_u\left[\lambda_j\Lambda_u^{-1}\right]^k C_{ju} \qquad k \ge 1 \tag{47}$$

It follows that

$$\left(\frac{\lambda_j}{\lambda_{q+1}}\right)^k \approx \frac{\|(V_{ju})_k - V_{ju}\|}{\|V_{ju}\|} \qquad k \ge 1 \tag{48}$$

and

$$k \approx \frac{\log[\|(V_{ju})_k - V_{ju}\|/\|V_{ju}\|]}{\log(\lambda_j/\lambda_{q+1})} \qquad k \ge 1 \tag{49}$$

In the examples that follow, we will see that this estimated number of iterations is quite accurate and a little conservative. Note that Eq. (49) applies to the basic iterative method; when the accelerated algorithms are used, it converges much faster. Also note that the iteration number thus evaluated is determined by the relative error to the exact solution.

From Eq. (48) it is seen that the major error of the solution is eliminated by the first several iterations.

Using Eq. (49) to estimate the required iterations for $\tilde{\lambda}_1/\tilde{\lambda}_{q+1}$ and $\tilde{\lambda}_q/\tilde{\lambda}_{q+1}$, and assuming Eq. (27) is satisfied, one can determine a compromised frequency shift λ_0 with the following one-dimensional searching scheme:

$$\delta\lambda_0 := \frac{0.5(\lambda_1 + \lambda_{q+1})}{m}$$

For $k = 1, 2, \ldots, m - 1$,

$$\lambda_0 := k\delta\lambda_0$$
$$\tilde{\lambda}_i := |\lambda_i - \lambda_0| \qquad i = 1, q, q+1$$
$$n_k := \frac{\log\sigma}{[1/\log(\tilde{\lambda}_1/\tilde{\lambda}_{q+1}) + 1/\log(\tilde{\lambda}_q/\tilde{\lambda}_{q+1})]}$$
$$\lambda_0 = \min(n_k)\delta\lambda_0 \ne \lambda_i \qquad i = 1, 2, \ldots, q$$

where σ represents the convergence criterion defined in Eq. (39), and m can be set to 10.

For the first eigenvector derivative, the main computational effort amounts to calculating the generalized stiffness matrix inverse; this computation can be thought of as equivalent to the direct method. From this point of view, the difference between the direct method and the iterative method is that the former solves a different generalized stiffness matrix inverse for each derivative, whereas the latter takes the advantage of the first generalized stiffness inverse to compute the rest of the eigenvector derivatives and thus avoids the formation of a new generalized stiffness inverse each time.[4]

It is not necessary to explicitly calculate the inverse of the stiffness matrix for the iterative procedure, only, say, a Cholesky decomposition[23] of the stiffness matrix is sufficient. It is also not necessary to calculate λ_{q+1}; λ_0 can be simply and, in most of the cases, conservatively set to

$$\lambda_0 = 0.45(\lambda_1 + \lambda_q) \tag{50}$$

The iterative method is an extension of Wang's modified modal method. In fact, the first iterative output $(V_{ju})_1 = AF_j\phi_j$ in Eq. (20) is Wang's solution using the explicit method.[2]

For systems with a relatively large dimension n, and relatively large number of eigenvector derivatives to be calculated, the proposed procedure will be more efficient, because more eigenvector

derivatives are expected to be calculated with small eigenvalue ratios. Note that the convergence speed is independent of the system dimension; this follows from Eq. (49) for the basic iterative method. For the same reason, for relatively small systems, the proposed procedure will be less efficient, because the number of iterations will remain more or less the same.

The iterative process will become less efficient with increasing the number of multiple design variables because the iterative method repeats the same process for every multiple design variable. On the other hand, the direct method needs only little more computational work for multiple design variables than for a single variable because the coefficient matrix requires Cholesky decomposition just one time.

VI. Illustrative Examples

In this section, two examples are presented to illustrate the performance of the proposed procedure. The first example is a 12×12 structural dynamic model with stiffness and mass matrices shown in Tables 1 and 2. The first-order derivatives of the diagonal mass matrix are shown in Table 3, and the derivative of the stiffness matrix is considered to be zero. The system eigenvalues are shown in Table 4. The second example has a dimension of 40×40. Its stiffness and mass matrices K and M are symmetric and positive definite; the derivatives K_I and M_I are symmetric; all are full matrices generated by a random function. The first 12 eigenvalues of the second example are given in Table 5. In the computation, the exact solution of V_{ju} is obtained by solving Eq. (19) directly; V_{ja} is obtained by means of the simple truncated modal method. The convergence criterion is Eq. (39) with $\sigma = 0.005$. Note this is the relative error to the exact solution.

Table 6 gives the exact solutions for the first four eigenvector derivatives ($q = 4$) of the first example and the corresponding element percentage errors to the exact solution by the truncated modal method, Wang's modified modal method (implicit method), and the iterative procedure with and without the acceleration techniques. For the truncated modal method, $(q + 2) = 6$ eigenvectors are utilized. For Wang's and the iterative methods, $q = 4$ eigenvectors, the same as the eigenvector derivatives demanded, are used in the calculations. It should be emphasized that Wang's method is not supposed to be used when only q eigenvectors are given. The solution of Wang's implicit method using q eigenvectors computed herein is used to provide a comparison and help show the performance of the proposed procedure.

From Table 6 it is seen that the truncated modal method yields a very rough approximate solution even though two more eigenvectors are used. Wang's implicit method provides much better results. The powerful effect of the acceleration techniques is clearly shown. Note the last eigenvalue ratio is quite large, $\lambda_q / \lambda_{q+1} = 0.9471$. To satisfy the given accuracy criterion, 97 iterations are estimated based on Eq. (49): $(0.9471)^{97} < 0.005$. The actual number of iterations is 96 using Eq. (20) without acceleration techniques (this is actually the iterative method proposed in Refs. 17 and 18). However, when the acceleration techniques are invoked, only 8 (algorithm 1) or 10 (algorithm 2) equivalent iterations (given by doubling the actual iterations in the accelerated algorithms) lead to convergence. In this case, $(96 - 10)/96 \approx 90\%$ of the computational amounts is saved.

Table 1 Elements of K ($\times 10^4$) (first example)

260	−60	−40	0	0	0	0	0	0	0	0	0
−60	320	−40	−40	0	0	0	0	0	0	0	0
−40	−40	220	−60	−40	0	0	0	0	0	0	0
0	−40	−60	120	−40	−20	0	0	0	0	0	0
0	0	−40	−40	180	−40	−20	0	0	0	0	0
0	0	0	−20	−40	180	−60	−60	0	0	0	0
0	0	0	0	−20	−60	240	−40	−50	0	0	0
0	0	0	0	0	−60	−40	200	−20	−50	0	0
0	0	0	0	0	0	−50	−20	220	−50	−30	0
0	0	0	0	0	0	0	−50	−50	240	−60	−50
0	0	0	0	0	0	0	0	−30	−60	240	−60
0	0	0	0	0	0	0	0	0	−50	−60	180

Table 2 Elements of M[a] (first example)

10	12	22	12	18	18	14	20	22	14	14	8

[a] M is a diagonal matrix; only the diagonal nonzero elements are given in the table.

Table 3 Elements of M_I[a] (first example)

2	1	0	1	0	1	4	2	0	4	1	2

[a] M_I is diagonal.

Table 4 System eigenvalues (first example)

0.2159e+5	0.4260e+5	0.7521e+5	0.1104e+6	0.1166e+6	0.1342e+6
0.1404e+6	0.1956e+6	0.2199e+6	0.2213e+6	0.2671e+6	0.3210e+6

Table 5 First 12 system eigenvalues (second example)

0.70	12.35	22.34	32.12	99.75	103.76
131.60	228.90	262.09	291.14	391.50	461.30

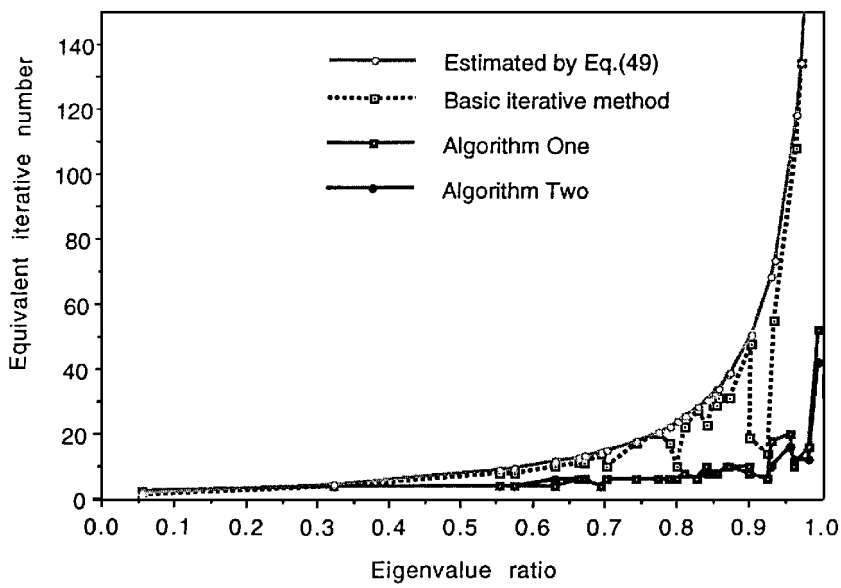

Fig. 1 Comparison of the equivalent number of iterations (the last two points for the estimated number and the basic iterative method are not shown on the plot; see text for details).

Table 6 Error[a] comparison of eigenvector derivatives (first example)

	Exact solution	Modal method (six modes)	Wang's method[b] (four modes)	Iterative basic eq.[c] (four modes)	Iterative alg. 1[d] (four modes)	Iterative alg. 2[d] (four modes)
				($k = 2$)[e]	($k = 4$)[f]	($k = 4$)[f]
ϕ_{1}	−0.001280	19.06	1.53	0.03	0.02	0.02
	−0.002032	10.43	0.11	0.02	0.00	0.00
	−0.006173	10.33	0.42	0.04	0.01	0.00
	−0.007213	14.17	0.61	0.05	0.02	0.00
	−0.006048	6.60	0.21	0.02	0.01	0.00
	−0.002117	11.56	0.79	0.02	0.02	0.00
	0.001189	128.65	1.88	0.13	0.06	0.00
	0.001021	43.23	0.66	0.11	0.01	0.00
	0.000577	95.82	1.16	0.25	0.13	0.00
	0.002379	26.01	0.25	0.05	0.03	0.01
	0.001236	4.56	2.04	0.09	0.01	0.02
	0.001596	22.95	1.14	0.10	0.03	0.01
				($k = 4$)	($k = 4$)	($k = 2$)
ϕ_{2}	0.000523	202.53	16.62	0.37	0.01	1.17
	0.000693	132.50	6.61	0.47	0.00	0.13
	−0.001337	50.65	4.43	0.16	0.00	0.57
	0.005401	48.43	4.00	0.16	0.00	0.24
	0.004655	54.77	7.44	0.33	0.00	0.06
	0.013126	5.37	0.52	0.00	0.00	0.01
	0.006092	35.75	6.26	0.10	0.00	0.03
	0.009676	7.61	0.77	0.00	0.00	0.02
	0.009721	40.10	3.38	0.20	0.00	0.02
	−0.000249	1702.96	3.22	5.03	0.04	2.82
	0.001324	48.85	39.85	1.18	0.01	0.16
	−0.001606	197.85	22.04	1.13	0.00	0.14
				($k = 11$)	($k = 4$)	($k = 2$)
ϕ_{3}	0.000299	369.83	533.08	4.87	0.11	3.56
	0.000191	654.35	419.15	6.22	0.05	3.01
	0.005588	142.97	20.16	0.61	0.01	0.23
	−0.005480	41.14	3.56	0.04	0.01	0.52
	−0.009711	98.39	21.77	0.57	0.00	0.18
	−0.007854	53.81	1.80	0.14	0.00	0.19
	0.011211	89.94	4.34	0.34	0.00	0.12
	0.001543	189.24	21.37	0.37	0.00	0.52
	0.026164	63.19	13.47	0.34	0.00	0.02
	−0.009517	175.44	4.29	0.60	0.01	0.03
	−0.002137	439.70	279.01	3.68	0.05	0.52
	−0.012227	124.85	20.19	0.71	0.00	0.13
				($k = 96$)	($k = 8$)	($k = 10$)
ϕ_{4}	0.004602	76.68	204.01	0.67	0.26	0.41
	0.005416	79.81	111.19	0.48	0.23	0.13
	0.022738	88.14	51.72	0.31	0.08	0.07
	−0.000145	1036.01	110.68	0.41	13.02	0.13
	−0.041439	92.12	57.80	0.29	0.00	0.00
	0.010407	104.31	40.41	0.20	1.14	0.46
	0.031899	96.11	29.49	0.25	0.19	0.05
	−0.008033	174.35	40.02	0.13	1.06	0.47
	0.038599	86.77	81.42	0.49	0.26	0.13
	−0.018709	141.07	101.37	0.66	0.40	0.17
	−0.015733	138.80	166.10	1.07	0.79	0.35
	−0.023164	124.85	120.46	0.81	0.57	0.29

[a]Defined by $100 \times |(\hat{\phi}_{ij} - \phi_{ij})/\phi_{ij}|$, where $\hat{\phi}_{ij}$ is the estimated solution, and ϕ_{ij} is the ith element of ϕ_j.
[b]Equation (22) in Ref. 2, implicit method.
[c]Basic iterative Eq. (20), $(V_{ju})_0 = 0$, without acceleration and frequency shift.
[d]Frequency shift $\lambda_0 = 0.45(\lambda_1 + \lambda_q)$.
[e]The actual number of iterations for the specific eigenvector derivative.
[f]The equivalent numbers of iteration given by doubling the actual iterations in the accelerated algorithms.

Tables 7 and 8 compare the equivalent number of iterations in the first example for various number of eigenvector derivatives demanded ($q = 2, 3, \ldots, 7$; $j = 1, 2, \ldots, q$), with and without frequency shift, respectively.

Table 9 shows the equivalent iterations for the second example. Figure 1 gives the equivalent number of iterations vs the eigenvalue ratios also for the second example ($q = 1, 2, \ldots, 30$; $j = q$). In Fig. 1, the largest two eigenvalue ratios are 0.9807 and 0.9932, i.e., extremely close to 1. Using the basic iterative method, 270 and 774 iterations are required (which is almost identical to the estimated number of iterations: 272 and 777); using the accelerated algorithms, the equivalent iterations are drastically reduced to only 16 and 52 (algorithm 1) or 12 and 42 (algorithm 2).

Several observations can be made from Tables 7–9 and Fig. 1: the estimated number of iterations based on Eq. (49) is consistent with the actual one for the basic iterative method, and the convergence rates of both iterative methods with or without the acceleration techniques are independent of the system dimension; when the eigenvalue ratios λ_j/λ_{q+1} are less than 0.70, the basic iterative method is attractive in term of efficiency; when $\lambda_j/\lambda_{q+1} > 0.70$, 50–90% of the computational effort is saved by the acceleration techniques, and the larger the eigenvalue ratios, the larger the percentage

Table 7 Equivalent iteration numbers (first example) without frequency shift

$q=2$	λ_j/λ_{q+1}	0.2870	0.5664					
	$\hat{k}^{a}$	4	9					
	k^{b}	2_4_2	8_4_4					
$q=3$	λ_j/λ_{q+1}	0.1955	0.3858	0.6812				
	$\hat{k}$	3	6	14				
	k	2_4_2	5_4_4	12_4_4				
$q=4$	λ_j/λ_{q+1}	0.1852	0.3654	0.6451	0.9471			
	$\hat{k}$	3	5	12	97			
	k	2_2_2	4_4_2	11_4_4	96_14_14			
$q=5$	λ_j/λ_{q+1}	0.1609	0.3175	0.5606	0.8229	0.8689		
	$\hat{k}$	3	5	9	27	38		
	k	2_2_2	4_4_2	7_4_4	23_8_8	33_10_12		
$q=6$	λ_j/λ_{q+1}	0.1537	0.3033	0.5355	0.7862	0.8301	0.9553	
	$\hat{k}$	3	4	8	22	28	116	
	k	2_2_2	3_4_2	5_4_2	21_6_6	25_8_8	113_14_18	
$q=7$	λ_j/λ_{q+1}	0.1104	0.2178	0.3845	0.5644	0.5960	0.6859	0.7180
	$\hat{k}$	2	3	6	9	10	14	16
	k	2_2_2	3_2_2	5_4_2	9_4_4	9_4_4	10_4_4	15_6_4

[a]The estimated numbers of iteration calculated by $\hat{k}_j = \log(\sigma)/\log(\lambda_j/\lambda_{q+1})$ with $\sigma = 0.005$.
[b]The equivalent numbers of iteration with acceleration techniques; $\lambda_0 = 0$. The first number is produced by the basic iterative method, the second number by algorithm 1, the third number by algorithm 2.

Table 8 Equivalent iteration numbers (first example) with frequency shift

$q=2$	$\tilde{\lambda}_j/\tilde{\lambda}_{q+1}$	_0.1575	0.2960					
	$\hat{k}^{a}$	3	4					
	k^{b}	1_2_2	3_4_4					
$q=3$	$\tilde{\lambda}_j/\tilde{\lambda}_{q+1}$	_0.3286	_0.0144	0.4734				
	$\hat{k}$	5	1	7				
	k	4_4_2	1_2_2	6_4_4				
$q=4$	$\tilde{\lambda}_j/\tilde{\lambda}_{q+1}$	_0.6613	_0.2939	0.2765	0.8921			
	$\hat{k}$	13	4	4	46			
	k	8_4_4	4_4_2	4_4_2	45_8_10			
$q=5$	$\tilde{\lambda}_j/\tilde{\lambda}_{q+1}$	_0.5638	_0.2720	0.1810	0.6700	0.7557		
	$\hat{k}$	9	4	3	13	19		
	k	8_4_2	3_4_2	2_4_2	11_6_4	16_6_8		
$q=6$	$\tilde{\lambda}_j/\tilde{\lambda}_{q+1}$	_0.6894	_0.3908	0.0728	0.5731	0.6609	0.9108	
	$\hat{k}$	14	6	2	10	13	57	
	k	7_4_2	4_4_2	1_2_2	9_4_4	11_4_6	55_8_10	
$q=7$	$\tilde{\lambda}_j/\tilde{\lambda}_{q+1}$	_0.4183	_0.2471	_0.0187	0.3056	0.3559	0.4992	0.5504
	$\hat{k}$	6	4	1	4	5	8	9
	k	5_4_2	3_2_2	1_2_2	4_4_2	4_4_2	5_4_2	8_4_4

[a,b]The same as Table 7 except $\lambda_0 = 0.45(\lambda_1 + \lambda_q)$.

Table 9 Equivalent iteration numbers (second example) without frequency shift

$q=2$	λ_j/λ_{q+1}	0.0314	0.5528					
	$\hat{k}^{a}$	2	9					
	k^{b}	1_2_2	8_4_4					
$q=3$	λ_j/λ_{q+1}	0.0218	0.3845	0.6954				
	$\hat{k}$	2	6	15				
	k	1_2_2	5_4_4	14_4_4				
$q=4$	λ_j/λ_{q+1}	0.0070	0.1238	0.2240	0.3220			
	$\hat{k}$	1	3	4	5			
	k	1_2_2	2_2_2	3_2_2	4_4_4			
$q=5$	λ_j/λ_{q+1}	0.0068	0.1190	0.2153	0.3096	0.9613		
	$\hat{k}$	1	2	3	5	134		
	k	1_2_2	2_2_2	3_2_2	3_4_4	134_10_12		
$q=6$	λ_j/λ_{q+1}	0.0053	0.0938	0.1698	0.2441	0.7580	0.7885	
	$\hat{k}$	1	2	3	4	19	22	
	k	1_2_2	2_2_2	2_2_2	3_4_4	16_6_6	17_6_6	
$q=7$	λ_j/λ_{q+1}	0.0031	0.0540	0.0976	0.1403	0.4358	0.4533	0.5749
	$\hat{k}$	1	2	2	3	6	7	10
	k	1_2_2	1_2_2	2_2_2	2_2_2	6_4_4	5_4_4	8_4_4

[a,b]The same as Table 7.

computational amounts saved; the two proposed algorithms converge essentially at the same rate; the required number of iterations in the accelerated algorithms depends mainly on the eigenvalue ratio once the relative error is given, and thus it is possible to be estimated a priori.

VII. Concluding Remarks

Assume that the first q eigenvector derivatives of a structural dynamic system are demanded and these q eigenvalues and eigenvectors are available. Only the first several, if any, eigenvector derivatives computed by the truncated modal method meet a mediate accuracy requirement. With the cost of calculating an additional stiffness matrix inverse, Wang's modified modal method greatly improves the accuracy of the solutions, and then more eigenvector derivatives will be acceptable. In Refs. 17 and 18 Wang's method was extended to a basic iterative method. Theoretically, all of the q eigenvector derivatives can be computed exactly using the basic iterative method. The cost is the iterative process. It was shown that, using the basic iterative method, the jth ($j \leq q$) eigenvector derivative converges at the rate of the eigenvalue ratio λ_j/λ_{q+1}. For most of the eigenvector derivatives demanded, $\lambda_j/\lambda_{q+1} (j < q)$ are expected to be small, and the iterative method converges rapidly. However, for the last several eigenvector derivatives, λ_j/λ_{q+1} may be close to 1 and the iterative method becomes expensive. The problem may arise when $\lambda_j/\lambda_{q+1} > 0.70$. In this paper, acceleration techniques are presented for the iterative method. Numerical examples show that when $\lambda_j/\lambda_{q+1} > 0.70$, 50–90% of the iterations will be saved, and the larger the λ_j/λ_{q+1}, the larger the percentage computational effort saved. Like the basic iterative method, no dependence on the system dimension is observed for the accelerated method. The procedure is applicable to systems with zero and repeated eigenvalues.

The proposed procedure is simple, systematic, efficient, and numerically stable.

Appendix: Properties of Eq. (19)

In this Appendix, we show that Eq. (19) defines a unique solution for V_{ju} that can be solved directly as a linear system. Noting that $A = \Psi_u \Lambda_u^{-1} \Psi_u^t$, and $\Psi\Psi^t M = I$, we begin with rewriting Eq. (19) as

$$
\begin{aligned}
[I - \lambda_j AM]V_{ju} &= [I - \lambda_j \Psi_u \Lambda_u^{-1} \Psi_u^t M]V_{ju} \\
&= \left[\Psi\Psi^t M - \Psi\begin{bmatrix} 0 & 0 \\ 0 & \lambda_j \Lambda_u^{-1} \end{bmatrix}\Psi^t M\right] V_{ju} \\
&= \Psi\left[I - \begin{bmatrix} 0 & 0 \\ 0 & \lambda_j \Lambda_u^{-1} \end{bmatrix}\right]\Psi^t M V_{ju} \\
&= \Psi\begin{bmatrix} I & 0 \\ 0 & I - \lambda_j \Lambda_u^{-1} \end{bmatrix}\Psi^t M V_{ju} = AF_j\phi_j
\end{aligned} \tag{A1}
$$

where the matrix $I - \lambda_j \Lambda_u^{-1}$ is diagonal and greater than zero. It is seen that the coefficient matrix of Eq. (A1) has a known eigenstructure: its eigenvectors remain the same as Ψ, and its eigenvalues are given by

$$
\begin{bmatrix} I & 0 \\ 0 & I - \lambda_j \Lambda_u^{-1} \end{bmatrix}
$$

It is also seen that if $M = I$, the coefficient matrix is symmetric and positive definite. Furthermore, premultiplication of M will always give a symmetric and positive definite coefficient matrix.

The unique solution defined by Eq. (A1) is the exact solution given by Eq. (10). From Eq. (A1) it follows that

$$
\begin{aligned}
V_{ju} &= \Psi\begin{bmatrix} I & 0 \\ 0 & [I - \lambda_j \Lambda_u^{-1}]^{-1} \end{bmatrix}\Psi^t MAF_j\phi_j \\
&= \Psi\begin{bmatrix} I & 0 \\ 0 & [I - \lambda_j \Lambda_u^{-1}]^{-1} \end{bmatrix}\Psi^t M\Psi_u \Lambda_u^{-1}\Psi_u^t F_j\phi_j \\
&= \Psi\begin{bmatrix} I & 0 \\ 0 & [I - \lambda_j \Lambda_u^{-1}]^{-1} \end{bmatrix}[0 \quad I]^t \Lambda_u^{-1}\Psi_u^t F_j\phi_j \\
&= \Psi_u[I - \lambda_j \Lambda_u^{-1}]^{-1}\Lambda_u^{-1}\Psi_u^t F_j\phi_j = \Psi_u[\Lambda_u - \lambda_j]^{-1}\Psi_u^t F_j\phi_j \\
&= \sum_{i=1,\, i\neq j}^{n} \frac{\phi_i^t F_j \phi_j}{\lambda_i - \lambda_j}\phi_i
\end{aligned} \tag{A2}
$$

References

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

[2] Wang, B. P., "Improved Approximate Methods for Computing Eigenvector Derivatives in Structural Dynamics," *AIAA Journal*, Vol. 29, No. 6, 1991, pp. 1018–1020.

[3] Akgün, M. A., "New Family of Modal Methods for Calculating Eigenvector Derivatives," *AIAA Journal*, Vol. 32, No. 2, 1994, pp. 379–386.

[4] Zeng, Q.-H., "Highly Accurate Modal Method for Calculating Eigenvectors in Viscous Damping Systems," *AIAA Journal*, Vol. 33, No. 4, 1995, pp. 746–751.

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

[6] Ojalvo, I. U., "Efficient Computation of Modal Sensitivities for Systems with Repeated Frequencies," *AIAA Journal*, Vol. 26, No. 3, 1988, pp. 361–366.

[7] Mills-Curran, W. C., "Calculation of Eigenvector Derivatives for Structures with Repeated Eigenvalues," *AIAA Journal*, Vol. 26, No. 7, 1988, pp. 867–871.

[8] Mills-Curran, W. C., "Comment on 'Eigenvector Derivatives with Repeated Eigenvalues'," *AIAA Journal*, Vol. 28, No. 10, 1990, p. 1846.

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

[10] Zhang, D.-W., and Wei, F.-S., "Computation of Eigenvector Derivatives with Repeated Eigenvalues Using a Complete Modal Space," *AIAA Journal*, Vol. 33, No. 9, 1995, pp. 1749–1753.

[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.

[13] Andrew, A. L., "Convergence of an Iterative Method for Derivatives of Eigensystems," *Journal of Computational Physics*, Vol. 26, No. 1, 1978, 107–112.

[14] Tan, R. C. E., and Andrew, A. L., "Computing Derivatives of Eigenvalues and Eigenvectors by Simultaneous Iteration," *IMA Journal of Numerical Analysis*, Vol. 9, No. 1, 1989, pp. 111–122.

[15] Tan, R. C. E., Andrew, A. L., and Hong, F. M. L., "Iterative Computation of Second-Order Derivatives of Eigenvalues and Eigenvectors," *Communications in Numerical Methods in Engineering*, Vol. 10, No. 1, 1994, pp. 1–9.

[16] Ting, T., "Accelerated Subspace Iteration for Eigenvector Derivatives," *AIAA Journal*, Vol. 30, No. 8, 1992, pp. 2114–2118.

[17] Liu, Z.-S., Chen, S.-H., Yu, M., and Zhao, Y.-Q., "Contribution of Truncated Modes to Eigenvector Derivatives," *AIAA Journal*, Vol. 32, No. 7, 1994, pp. 1551–1553.

[18] Zhang, O., and Zerva, A., "Iterative Method for Calculating Derivatives of Eigenvalues," *AIAA Journal*, Vol. 34, No. 5, 1996, pp. 1088–1090.

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

[20] 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.

[21] Lin, R. M., and Lim, M. K., "Eigenvector Derivatives of Structures with Rigid Body Modes," *AIAA Journal*, Vol. 34, No. 5, 1996, pp. 1083–1085.

[22] Shaw, J., and Jayasuriya, S., "Modal Sensitivities for Repeated Eigenvalues and Eigenvector Derivatives," *AIAA Journal*, Vol. 30, No. 3, 1992, pp. 850–852.

[23] Golub, G. H., and Van Loan, C. F., *Matrix Computations*, Johns Hopkins Univ. Press, Baltimore, MD, 1983.