AIAA Journal
Vol. 43, No. 6, June 2005

Technical Notes

TECHNICAL NOTES are short manuscripts describing new developments or important results of a preliminary nature. These Notes should not exceed 2500 words (where a figure or table counts as 200 words). Following informal review by the Editors, they may be published within a few months of the date of receipt. Style requirements are the same as for regular contributions (see inside back cover).

Davidson Method for Eigenpairs and Their Derivatives

Huiqing Xie*
East China University of Science and Technology, 200237 Shanghai, People's Republic of China
and
Hua Dai†
Nanjing University of Aeronautics and Astronautics, 210016 Nanjing, People's Republic of China

Nomenclature

A^T	=	transpose of matrix A
$\tilde{A}$	=	approximation of matrix A
col (V_k)	=	number of columns of matrix V_k
$R^{m\times n}$	=	set of all $m \times n$ real matrices
$SR^{n\times n}$	=	set of all real symmetric matrices of order n
$v_i^{(k)}$	=	ith column of matrix V_k
$\|\ \|$	=	Euclidean vector norm
$\langle\ \rangle$	=	inner product, $\langle x, y\rangle = x^T y$

Introduction

EIGENPAIR derivatives with respect to system parameters are widely used in many fields such as model updating,[1] damage detection,[2] and structural optimization.[3] The computation of eigenvalue derivatives is simple and straightforward. However, calculation of eigenvector derivatives is much more complicated. At present, methods for calculating eigenvector derivatives mainly include the finite difference method,[4] modal method,[5,6] Nelson method,[7] etc. The Nelson method yields an accurate solution of eigenpair derivative, and the method was widely accepted. Using Lanczos vectors, Ojalvo and Zhang[8] developed a method for computation of eigenvector derivatives. However, this method failed in some cases because it neglected the derivatives of Lanczos vectors. Based on the Lanczos method, Yuan et al.[9] present a procedure for synchrocalculation of eigenpairs and their derivatives.

The Davidson method (see Ref. 10) is an efficient way to compute the extreme eigenvalues and their corresponding eigenvectors of a large, sparse, and symmetric matrix. Compared with the Lanczos method, the Davidson method is usually more efficient because it applies preconditioning techniques in solving eigenvalue problems. Based on the Davidson method, this Note proposes a new method for simultaneously calculating the eigenpairs and their derivatives of the symmetric matrices. When the proposed method is used, the system of equations for eigenvector derivatives can be greatly reduced from the original matrix sizes. Thus, the efficiency of computing eigenvector derivatives is improved. Numerical results also show that the proposed method is more efficient than the synchro-Lanczos method (see Ref. 9) for computing several eigenpair derivatives of the large symmetric matrices.

Davidson Method for Eigenpairs of Symmetric Matrices

Let A be a real symmetric matrix of order n, and let $\lambda_1 \leq \cdots \leq \lambda_n$ be its eigenvalues. The basic idea of Davidson method (see Ref. 10) for computing the eigenpairs of A is to build a sequence $\{\mathcal{V}_k\}$ of subspaces with nondecreasing dimensions; to calculate Rayleigh matrix H_k, that is, the projected matrix of A onto $\mathcal{V}_k$; and to take Ritz pairs as the approximations of the desired eigenpairs of A. Note that the Rayleigh matrix is similar to A when the dimension of subspace reaches n and that Ritz pairs are also the eigenpairs of A. However, n is generally large. To be efficient, it is necessary to set a maximum dimension $m (m \ll n)$ of subspaces and restart the process with the last approximations of the desired eigenvectors as the initial vectors when the dimension of subspace equals m.

The Davidson method that computes the l smallest eigenpairs of A is as follows in Algorithm 1 (see Ref. 10):

1) Choose two integers $m, s (m > s, s \geq l)$, a positive real number ε, approximate matrix $\tilde{A} \in SR^{n\times n}$, and an initial orthonormal matrix $V_0 = [v_1, \ldots, v_s] \in R^{n\times s}$.

2) For $k = 0, 1, 2, \ldots$, perform the following:

a) Compute the Rayleigh matrix $H_k = V_k^T A V_k$ and the s smallest eigenpairs $(\mu_i^{(k)}, y_i^{(k)})$, $i = 1, \ldots, s$, of H_k, where $\mu_1^{(k)} \leq \cdots \leq \mu_s^{(k)}$.

b) Compute the Ritz vectors $\boldsymbol{x}_i^{(k)} = V_k \boldsymbol{y}_i^{(k)}$, $i = 1, \ldots, s$, and residuals $\boldsymbol{r}_i^{(k)} = \boldsymbol{A}\boldsymbol{x}_i^{(k)} - \mu_i^{(k)} \boldsymbol{x}_i^{(k)}$, $i = 1, \ldots, s$.

c) If $\|r_i^{(k)}\| \leq \varepsilon$, $i = 1, \ldots, l$, then exit.

d) If $\mathrm{col}(V_k) > m - s$, then go to step 2a with $V_0 = [\boldsymbol{x}_1^{(k)}, \ldots, \boldsymbol{x}_s^{(k)}]$ and $k = 0$.

e) Compute $\omega_i^{(k)}$, $i = 1, \ldots, s$, from equations

$$\left[\tilde{A} - \mu_i^{(k)} I\right]\omega_i^{(k)} = r_i^{(k)}, \qquad i = 1, \ldots, s$$

f) V_{k+1} = modified Gram–Schmidt (MGS) $(V_k, \omega_1^{(k)}, \ldots, \omega_s^{(k)})$, $k = k + 1$. Go to step 2a.

Because $V_k^T V_k = I$, the procedure $\mathrm{MGS}(V_k, \omega_1^{(k)}, \ldots, \omega_s^{(k)})$ can be described as follows:

1) For $i = 1, \ldots, \mathrm{col}(V_k)$, compute

$$\omega_j^{(k)} = \omega_j^{(k)} - \left\langle \omega_j^{(k)}, v_i^{(k)} \right\rangle v_i^{(k)}, \qquad j = 1, \ldots, s$$

2) For $i = 1, \ldots, s$, compute

$$\alpha = \left\| \omega_i^{(k)} \right\|$$

$$\omega_i^{(k)} = \omega_i^{(k)} \big/ \alpha$$

For $j = i + 1, \ldots, s$, compute

$$\omega_j^{(k)} = \omega_j^{(k)} - \left\langle \omega_j^{(k)}, \omega_i^{(k)} \right\rangle \omega_i^{(k)}$$

Synchrocalculation of Eigenpairs and Their Derivatives

Let $A(p) \in SR^{n\times n}$ be analytic in a neighborhood of $p^* \in R^N$, where $p = (p_1, \ldots, p_N)^T \in R^N$. Without loss of generality, we

Received 30 July 2004; revision received 16 December 2004; accepted for publication 18 December 2004.

*Assistant Professor, Department of Mathematics.
†Professor, Department of Mathematics.

may assume that the point p^* is the origin of R^N. Suppose that $(\lambda_i, \boldsymbol{x}_i)(\|\boldsymbol{x}_i\| = 1, i = 1, \ldots, l)$ are the l smallest eigenpairs of $A(0)$ and $\lambda_1 < \cdots < \lambda_l$ are the simple eigenvalues. By the results in Ref. 11, there exist analytic real-valued functions $\lambda_1(p), \ldots, \lambda_l(p)$ and analytic vector-valued functions $\boldsymbol{x}_1(p), \ldots, \boldsymbol{x}_l(p)$ such that

$$A(p)\boldsymbol{x}_i(p) = \lambda_i(p)\boldsymbol{x}_i(p), \qquad i = 1, \ldots, l \tag{1}$$

Moreover, $\lambda_i(0) = \lambda_i$ and $\boldsymbol{x}_i(0) = \boldsymbol{x}_i$, $i = 1, \ldots, l$. In this section, a method for synchrocomputation of (λ_i, x_i), $(\partial\lambda_i(0)/\partial p_t, \partial x_i(0)/\partial p_t)$, $i = 1, \ldots, l$, is presented.

Differentiating both sides of Eq. (1), we obtain the following equation for the eigenvector derivative:

$$[A(0) - \lambda_i I]\frac{\partial \boldsymbol{x}_i(0)}{\partial p_t} = -\left[\frac{\partial A(0)}{\partial p_t} - \frac{\partial \lambda_i(0)}{\partial p_t} I\right]\boldsymbol{x}_i \tag{2}$$

Based on Algorithm 1, we transfer Eq. (2) to a system of equations with smaller size. The basic idea of our method is to restart Algorithm 1 with the last approximations of the desired eigenvectors as the initial vectors and to compute Ritz pairs and their derivatives simultaneously if the residual norms become small but do not satisfy the desired accuracy. Thus, by our method, we can obtain the desired eigenpairs and their derivatives at the same time.

At each iteration of Algorithm 1, the following eigenproblem needs to be solved:

$$\left[V_k^T A(0) V_k - \mu_i^{(k)} I\right] y_i^{(k)} = 0$$

Differentiating both sides of the preceding equation, we have

$$\left[H_k - \mu_i^{(k)} I\right]\frac{\partial y_i^{(k)}}{\partial p_t} = -\left[\frac{\partial H_k}{\partial p_t} - \frac{\partial \mu_i^{(k)}}{\partial p_t} I\right] y_i^{(k)} \tag{3}$$

where $H_k = V_k^T A(0) V_k$, $(\partial H_k/\partial p_t) = (\partial V_k^T/\partial p_t) A(0) V_k + V_k^T (\partial A(0)/\partial p_t) V_k + V_k^T A(0)(\partial V_k/\partial p_t)$, and $\partial\mu_i^{(k)}/\partial p_t$ and $\partial y_i^{(k)}/\partial p_t$ can be obtained from Eq. (3) by the Nelson method.[7] Obviously, the size of Eq. (3) is much smaller than that of Eq. (2).

Because the approximation of the eigenvector of $A(0)$ is

$$\boldsymbol{x}_i^{(k)} = V_k y_i^{(k)} \tag{4}$$

the approximation of the eigenvector derivative is

$$\frac{\partial \boldsymbol{x}_i^{(k)}}{\partial p_t} = \frac{\partial V_k}{\partial p_t} y_i^{(k)} + V_k \frac{\partial y_i^{(k)}}{\partial p_t} \tag{5}$$

and the approximation of the eigenpair derivative is $(\partial\mu_i^{(k)}/\partial p_t, \partial \boldsymbol{x}_i^{(k)}/\partial p_t)$. If the approximations of eigenpairs and their derivatives do not satisfy the convergent requirement, then compute the residual

$$r_i^{(k)} = A(0)\boldsymbol{x}_i^{(k)} - \mu_i^{(k)}\boldsymbol{x}_i^{(k)} \tag{6}$$

From Eq. (6), it is easily seen that

$$\frac{\partial r_i^{(k)}}{\partial p_t} = \left[A(0) - \mu_i^{(k)} I\right]\frac{\partial \boldsymbol{x}_i^{(k)}}{\partial p_t} + \left[\frac{\partial A(0)}{\partial p_t} - \frac{\partial \mu_i^{(k)}}{\partial p_t} I\right]\boldsymbol{x}_i^{(k)} \tag{7}$$

Let $\tilde{A}(p) \in SR^{n\times n}$ be an approximation of $A(p)$. Solve $\omega_i^{(k)}$ from the following preconditioning equation:

$$\left[\tilde{A}(0) - \mu_i^{(k)} I\right]\omega_i^{(k)} = r_i^{(k)} \tag{8}$$

Differentiating both sides of Eq. (8) leads to the following equation for $\partial\omega_i^{(k)}/\partial p_t$:

$$\left[\tilde{A}(0) - \mu_i^{(k)} I\right]\frac{\partial \omega_i^{(k)}}{\partial p_t} = \frac{\partial r_i^{(k)}}{\partial p_t} - \left[\frac{\partial \tilde{A}(0)}{\partial p_t} - \frac{\partial \mu_i^{(k)}}{\partial p_t} I\right]\omega_i^{(k)} \tag{9}$$

When the MGS procedure is performed and the expressions in MGS are simultaneously differentiated, V_{k+1} and $\partial V_{k+1}/\partial p_t$ can be obtained.

The algorithm for synchrocalculation of $(\lambda_i, \boldsymbol{x}_i)$, $(\partial\lambda_i(0)/\partial p_t, \partial \boldsymbol{x}_i(0)/\partial p_t)$, $i = 1, \ldots, l$, is given as follows in Algorithm 2:

1) Choose two integers $m, s (m > s, s \geq l)$, a positive real number ε_1, an initial orthonormal matrix $V_0 = [v_1, \ldots, v_s] \in R^{n\times s}$, and matrix-valued functions $\tilde{A}(p)$ approximating to $A(p)$. Use Algorithm 1 to calculate Ritz vectors $\tilde{\boldsymbol{x}}_i$, $i = 1, \ldots, l$, such that residual norms $\|r_i\| \leq \varepsilon_1$, $i = 1, \ldots, l$.

2) Choose integer M. Let $V_0 = [\tilde{x}_1, \ldots, \tilde{x}_l]$, $\partial V_0/\partial p_t = 0$, and $k = 0$.

3) Compute the l smallest eigenpairs $(\mu_i^{(k)}, y_i^{(k)})$, $i = 1, \ldots, l$, and their derivatives $(\partial\mu_i^{(k)}/\partial p_t, \partial y_i^{(k)}/\partial p_t)$, $i = 1, \ldots, l$, of Rayleigh matrix $H_k = V_k^T A(0) V_k$.

4) Compute Ritz vectors $\boldsymbol{x}_i^{(k)}$, $i = 1, \ldots, l$, and their derivatives $\partial\boldsymbol{x}_i^{(k)}/\partial p_t$, $i = 1, \ldots, l$, from Eqs. (4) and (5), respectively.

5) If $\mathrm{col}(V_k) > M - l$, then stop. Here, $(\mu_i^{(k)}, x_i^{(k)})$, $i = 1, \ldots, l$, are the approximations of the l smallest eigenpairs $(\lambda_i, \boldsymbol{x}_i)$, $i = 1, \ldots, l$, and $(\partial\mu_i^{(k)}/\partial p_t, \partial\boldsymbol{x}_i^{(k)}/\partial p_t)$, $i = 1, \ldots, l$, are the approximations of the l smallest eigenpair derivatives $(\partial\lambda_i(0)/\partial p_t, \partial\boldsymbol{x}_i(0)/\partial p_t)$, $i = 1, \ldots, l$.

6) Compute the residuals $r_i^{(k)}$, $i = 1, \ldots, l$, and their derivatives $\partial r_i^{(k)}/\partial p_t$, $i = 1, \ldots, l$, from Eqs. (6) and (7), respectively.

7) Solve $\omega_i^{(k)}$, $\partial\omega_i^{(k)}/\partial p_t$, $i = 1, \ldots, l$, from Eqs. (8) and (9), respectively.

8) Compute V_{k+1} and $\partial V_{k+1}/\partial p_t$.

a) For $i = 1, \ldots, \mathrm{col}(V_k)$, compute

$$\frac{\partial \omega_j^{(k)}}{\partial p_t} = \frac{\partial \omega_j^{(k)}}{\partial p_t} - \left\langle \frac{\partial \omega_j^{(k)}}{\partial p_t}, v_i^{(k)} \right\rangle v_i^{(k)} - \left\langle \omega_j^{(k)}, \frac{\partial v_i^{(k)}}{\partial p_t} \right\rangle v_i^{(k)}$$

$$- \left\langle \omega_j^{(k)}, v_i^{(k)} \right\rangle \frac{\partial v_i^{(k)}}{\partial p_t}, \qquad j = 1, \ldots, l$$

$$\omega_j^{(k)} = \omega_j^{(k)} - \left\langle \omega_j^{(k)}, v_i^{(k)} \right\rangle v_i^{(k)}, \qquad j = 1, \ldots, l$$

b) For $i = 1, \ldots, l$, compute

$$\alpha = \left\|\omega_i^{(k)}\right\|, \qquad \frac{\partial \omega_i^{(k)}}{\partial p_t} = \frac{\partial \omega_i^{(k)}}{\partial p_t}\Big/\alpha - \alpha^{-3}\left\langle \frac{\partial \omega_i^{(k)}}{\partial p_t}, \omega_i^{(k)} \right\rangle \omega_i^{(k)}$$

$$\omega_i^{(k)} = \omega_i^{(k)}\big/\alpha$$

For $j = i + 1, \ldots, l$, compute

$$\frac{\partial \omega_j^{(k)}}{\partial p_t} = \frac{\partial \omega_j^{(k)}}{\partial p_t} - \left\langle \frac{\partial \omega_j^{(k)}}{\partial p_t}, \omega_i^{(k)} \right\rangle \omega_i^{(k)} - \left\langle \omega_j^{(k)}, \frac{\partial \omega_i^{(k)}}{\partial p_t} \right\rangle \omega_i^{(k)}$$

$$- \left\langle \omega_j^{(k)}, \omega_i^{(k)} \right\rangle \frac{\partial \omega_i^{(k)}}{\partial p_t}, \qquad \omega_j^{(k)} = \omega_j^{(k)} - \left\langle \omega_j^{(k)}, \omega_i^{(k)} \right\rangle \omega_i^{(k)}$$

9) $V_{k+1} = [V_k, \omega_1^{(k)}, \ldots, \omega_l^{(k)}]$, $\partial V_{k+1}/\partial p_t = [\partial V_k/\partial p_t, \partial\omega_1^{(k)}/\partial p_t, \ldots, \partial\omega_l^{(k)}/\partial p_t]$, $k = k + 1$. Go to step 3.

Numerical Example

Let $p \in R$, $A(p) = [a_{ij}(p)] \in SR^{900\times 900}$, where $A(p)$ is tridiagonal except that $a_{1,900}(p) \equiv a_{900,1}(p) \equiv -100$. The diagonal elements are

$$a_{ii}(p) = \begin{cases} 20{,}000 + 1000 \times i, & i \neq 4, 5, \quad i = 1, \ldots, n \\ 10{,}000 + 1000 \times i + p, & i = 4, 5, \quad i = 1, \ldots, n \end{cases}$$

and the minor diagonal elements are

$$a_{i,i-1}(p) = \begin{cases} -10{,}000 & i \neq 5, \quad i = 2, \ldots, n \\ -p & i = 5, \quad i = 2, \ldots, n \end{cases}$$

We compute the 10 smallest eigenpairs and their derivatives of $A(p)$ at $p^* = 10{,}000$ by Algorithm 2, and compare its numerical results with that of the synchro-Lanczos method (see Ref. 9).

Table 1 Comparison of results

	N_r		E_λ		E_x		T_c, s	
J	NM	SLM	NM	SLM	NM	SLM	NM	SLM
1	1.4628e−8	1.1482e−7	1.9641e−14	9.6418e−14	9.7713e−6	2.6919e−3		
2	2.7720e−8	2.8620e−7	5.9840e−15	4.0891e−14	1.5145e−5	1.5529e−3		
3	4.8965e−8	7.4500e−7	1.2292e−13	4.0415e−14	4.4690e−5	3.8890e−3		
4	6.7182e−8	2.8480e−6	1.4567e−13	2.8134e−12	6.6056e−5	1.9149e−2		
5	7.9757e−8	3.4330e−6	4.9124e−15	2.8929e−11	6.7428e−5	6.1060e−2	41.109	162.013
6	1.5436e−7	3.4703e−6	6.7246e−14	1.8126e−12	6.8173e−5	1.0700e−2		
7	1.0199e−6	2.3305e−6	6.8581e−14	6.2442e−12	2.9170e−4	1.5695e−2		
8	1.9751e−6	1.5826e−6	9.1824e−14	1.8809e−12	1.4971e−3	1.8017e−2		
9	2.5679e−6	8.9919e−6	3.4499e−13	1.4144e−11	3.2772e−3	4.5128e−2		
10	7.2404e−6	1.9618e−5	7.9757e−13	2.0829e−9	2.0764e−2	5.8762e−1		

In Algorithm 2, we set $\varepsilon_1 = 0.1$, $s = 10$, $m = 90$, $M = 70$, and choose $\tilde{A}(p)$ as the diagonal part of $A(p)$.

The error of eigenpair derivatives calculated by Algorithm 2 and the synchro-Lanczos method (see Ref. 9) are defined by

$$E_\lambda = |\tilde{\lambda}' - \lambda'/\lambda'| \tag{10}$$

$$E_x = \|\tilde{x}' - x'\|/\|x'\| \tag{11}$$

respectively, where $\tilde{\lambda}'$ and $\tilde{\boldsymbol{x}}'$ are eigenpair derivatives calculated by Algorithm 2 or the method in Ref. 9, and λ' and $\boldsymbol{x}'$ are eigenpair derivatives calculated by the Nelson method.

A summary of the results are given in Table 1, for the new method (NM), that is, Algorithm 2 and the synchro Lanczos method (SLM) (see Ref. 9). J denotes the order of eigenpair, N_r denotes the residual norms, E_λ denotes the errors of eigenvalue derivatives, E_x denotes the errors of eigenvector derivatives, and T_c denotes the computational time.

Numerical results show that Algorithm 2 is more efficient than the SLM (see Ref. 9) for calculating several smallest eigenpair derivatives of large symmetric matrices.

Conclusions

A new method based on the Davidson method for solving eigenvalue problems has been presented to compute eigenpair derivatives. This new algorithm calculates eigenpairs and their derivatives simultaneously. The systems of equations for solving eigenvector derivatives can be greatly reduced from the original matrix sizes. This algorithm can be very useful for the calculation of several eigenpair derivatives of large symmetric matrices.

References

[1]Zhang, D. W., and Zhang, L. M., "Matrix Transformation Method for Updating Dynamic Model," *AIAA Journal*, Vol. 30, No. 5, 1992, pp. 1440–1443.

[2]Messina, A., Williams, E. J., and Contursi, T., "Structural Damage Detection by a Sensitivity and Statistical-Based Method," *Journal of Sound and Vibration*, Vol. 216, No. 5, 1998, pp. 791–808.

[3]Arora, J. S., and Huang, E. J., "Methods of Design Sensitivity Analysis in Structural Optimization," *AIAA Journal*, Vol. 17, No. 9, 1979, pp. 970–974.

[4]Iott, J., Haftka, R. T., and Adelman, H. M., "On a Procedure for Selecting Step Sizes in Sensitivity Analysis by Finite Differences," NASA TM-86382, Aug. 1985.

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

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

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

[8]Ojalvo, I. U., and Zhang, L. M., "Efficient Eigenvector Sensitivities by a New Procedure Based on Lanczos Vectors," *AIAA Journal*, Vol. 34, No. 11, 1996, pp. 2392–2394.

[9]Yuan, X. R., Chen, E. L., and Yang, S. P., "Lanczos Method in Synchro Calculation of Eigenpairs and Their Derivatives," *AIAA Journal*, Vol. 36, No. 5, 1998, pp. 874, 875.

[10]Crouzeix, M., Philippe, B., and Sadkane, M., "The Davidson Method," *SIAM Journal on Scientific Computing*, Vol. 15, No. 1, 1994, pp. 62–76.

[11]Sun, J. G., "Eigenvalues and Eigenvectors of a Matrix Dependent on Several Parameters," *Journal of Computational Mathematics*, Vol. 3, No. 4, 1985, pp. 351–364.

A. Berman
Associate Editor