JOURNAL OF COMPUTATIONAL PHYSICS 26, 107–112 (1978)

Note

Convergence of an Iterative Method for Derivatives of Eigensystems

1. Introduction

Rudisill and Chu [1] have proposed an iterative method for numerical solution of the following problem, which arises, for example, in the optimal design of systems where the dynamic stability or response of the system is a function of several design parameters. (See [1] for references concerning both applications in aeronautical engineering and other numerical methods.) Given an $n \times n$ matrix A which depends on m parameters, $P_1,..., P_m$, it is required to find the partial derivatives with respect to these parameters of one or more of the eigenvalues, λ_i, and corresponding normalized eigenvectors, x_i, of A. Rudisill and Chu reported that in their experience the method converged, but they gave neither any guarantee that this would always be the case nor any estimate of the rate of convergence.

In this note a simple proof is given of the convergence of the algorithm, essentially under the conditions mentioned in [1], and its rate of convergence is determined. It is shown that the limit of the sequence of approximations always gives the true value for the derivatives of the eigenvalues but it gives values for the derivatives of the eigenvectors which depend on the initial approximations chosen. A special choice is recommended. It is noted that a minor change should be made in the algorithm when the eigenvectors are complex. The insight obtained by the theoretical analysis given here is used to derive a more rapidly convergent iterative scheme. As in [1] the existence of all partial derivatives required is assumed. Some results concerning existence of derivatives of eigenvalues and eigenvectors are proved in [2].

2. Proof of Convergence

It is given that

$$(A - \lambda_i I)x_i = 0, \qquad i = 1,..., n, \tag{1}$$

and

$$x_i^* x_i = 1, \qquad i = 1,..., n, \tag{2}$$

where I is the identity matrix and the asterisk denotes the complex conjugate transpose. The basic algorithm considered here is

$$\mu(k) = x_i^* A_{,j} x_i + x_i^*(A - \lambda_i I)\, u(k), \tag{3}$$

$$u(k+1) = \{[A_{,j} - \mu(k)I]\, x_i + Au(k)\}/\lambda_i, \tag{4}$$

0021-9991/78/0261-0107$02.00/0

where the subscript $,j$ denotes the partial derivative with respect to the jth parameter P_j and $u(0)$ is the initial approximation for $x_{i,j}$. We examine the convergence of $u(k)$ and $\mu(k)$ to $x_{i,j}$ and $\lambda_{i,j}$, respectively, as $k \to \infty$.

Substitution of (3) into (4) yields the nonhomogeneous linear recurrence relation

$$u(k+1) + [x_i^* Au(k)/\lambda_i]x_i - Au(k)/\lambda_i - [x_i^* u(k)]x_i = [A_{,j}x_i - (x_i^* A_{,j}x_i)x_i]/\lambda_i . \quad (5)$$

If $u(k) = x_{i,j}$ then, as shown in [1], $u(k+1) = x_{i,j}$. Hence a particular solution of (5), independent of k, is given by $u(k) = x_{i,j}$. Hence the general solution of (5) is given by

$$u(k) = x_{i,j} + \alpha v(k),$$

where α is a constant scalar and $v(k)$ satisfies the homogeneous equation

$$v(k+1) + [x_i^* Av(k)/\lambda_i]x_i - Av(k)/\lambda_i - [x_i^* v(k)]x_i = 0. \quad (6)$$

Hence as $k \to \infty$, $u(k)$ approaches a limit if and only if $v(k)$ does and $u(k) \to x_{i,j}$ if and only if $v(k) \to 0$. (Roundoff errors will ensure that the case $\alpha = 0$ has no special significance in actual numerical calculation.)

We now make the assumption that the set of eigenvectors $\{x_1, \ldots, x_n\}$ is linearly independent. (Some results in [1] require the stronger assumption that the eigenvalues are all distinct.) This implies that for each choice of $u(0)$ there exist scalars $\beta_p(k)$ such that

$$v(k) = \sum_{p=1}^{n} \beta_p(k)\, x_p , \qquad k = 0, 1, 2, \ldots. \quad (7)$$

Substitution of (7) in (6) shows that

$$\beta_p(k+1) = \beta_p(k)\lambda_p/\lambda_i , \qquad \text{for} \quad p \neq i. \quad (8)$$

Also it follows from (2) and (6) that

$$x_i^* v(k+1) = x_i^* v(k). \quad (9)$$

Finally we make the assumption that

$$|\lambda_1| > |\lambda_2| \geqslant \cdots \geqslant |\lambda_n|. \quad (10)$$

If $i = 1$ then (7)–(10) show that there is a scalar, γ, depending on $u(0)$, such that

$$v(k) \to \gamma x_i \qquad \text{as} \quad k \to \infty, \quad (11)$$

and (9) shows that $\gamma = 0$ if and only if $x_i^* v(0) = 0$. By (2), Re $x_i^* x_{i,j} = 0$. When all quantities are real it follows that if (and only if)

$$x_i^* u(0) = 0 \quad (12)$$

then $u(k) \to x_{i,j}$ as $k \to \infty$ and, by (8) and (10),

$$\| u(k) - x_{i,j} \| = O(| \lambda_{i+1}/\lambda_i |^k). \tag{13}$$

Moreover, (1) (3), and (11) now show that $\mu(k) \to \lambda_{i,j}$ and

$$| \mu(k) - \lambda_{i,j} | = O(| \lambda_{i+1}/\lambda_i |^k) \tag{14}$$

as $k \to \infty$, whether or not $u(0)$ satisfies (12).

There is no difficulty in satisfying (12) as an arbitrary initial estimate $u(0)$ could be replaced by $u(0) - x_i^* u(0) x_i$. Like $v(k)$, $u(k)$ satisfies (9), but in an actual numerical calculation roundoff errors will prevent the computed values of $u(k)$ from doing so exactly. This difficulty may be offset by replacing u_k, the computed value of $u(k)$, by $u_k - (x_i^* u_k) x_i$ for the final value of k.

When A is real, complex eigenvalues occur in complex conjugate pairs and hence if the first r inequalities in (10) are strong then $x_1, \ldots, x_r$ are all real. It is perhaps for this reason that, instead of (2), Rudisill and Chu used the normalization

$$x_i^t x_i = 1, \tag{2'}$$

where the superscript t denotes the simple transpose, and, consequently, they also had x_i^t instead of x_i^* in (3). When x_i is real (2) and (2') are of course equivalent but the example

$$A = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}$$

shows that it is not always possible to normalize complex eigenvectors by (2').

When x_i is complex (2) does not imply $\operatorname{Im} x_i^* x_{i,j} = 0$, but without loss of generality we may take $x_i^* x_{i,j}$ to be zero so that (12) again gives the appropriate criterion for choosing $u(0)$. The reason is that (2) does not specify an eigenvector uniquely. If an eigenvector x satisfies (2) and $z(P_j) = x(P_j) \exp(icP_j)$, where c is an arbitrary real number, then the eigenvector z satisfies (2) but $z^*(0) z_{,j}(0) = x^*(0) x_{,j}(0) + ic$. Indeed since any nonzero scalar multiple of an eigenvector is also an eigenvector, it follows from (11) that, for any $u(0)$, the limit of the sequence $\{u(k)\}$ can be regarded as *a* derivative of x_1 when no normalization is required.

The algorithm (3), (4) will not normally converge if $i \neq 1$ as by (8) and (10) some unwanted $\beta_p(k)$ will grow with k. In the case when A is Hermitian and all inequalities in (10) are strict, an extension to (3), (4) given in [1] enables all $x_{i,j}$ and $\lambda_{i,j}$ to be calculated in order of increasing i. The method, which is based on the orthogonality of the different eigenvectors in that case, consists of alternating the step (3), (4) with a step which subtracts the unwanted growing terms. Convergence of this extension may also be proved by an obvious modification of the preceding argument. Again the approximations will satisfy (14), and if (12) is satisfied they will satisfy (13).

This extension may be generalized to the non-Hermitian case, using the fact that a right-hand eigenvector is always orthogonal to a left-hand eigenvector corresponding

to a different eigenvalue. This generalization, which requires computing the derivatives of both left- and right-hand eigenvectors, is not mentioned in [1], possibly because it is not economic unless the number of unnecessary eigenvectors whose derivatives are required is very small compared with the order of A. The disadvantage of requiring additional eigenvectors and their derivatives is only partly offset by the fact that $\lambda_{i,j}$ can then be calculated immediately from the well-known equation $\lambda_{i,j} = y_i^* A_{,j} x_i / y_i^* x_i$, where y_i^* is a left-hand eigenvector of A corresponding to λ_i, and that the term $[A_{,j} - \mu(k)I]x_i$ in (4) may be replaced by $(A_{,j} - \lambda_{i,j}I)x_i$, which need only be calculated once.

3. Acceleration of Convergence

Convergence of (3), (4) may be accelerated by using a shift of origin analogous to that used in some standard methods for computing eigenvalues [3]. This effectively replaces A by a matrix with each λ_i increased by a constant, hence changing the rate of convergence in (13), (14), while $\lambda_{i,j}$, x_i, and $x_{i,j}$ are unchanged. Then (4) is replaced by the more general equation

$$u(k+1) = \{[A_{,j} - \mu(k)I]x_i + [A - \sigma(k)I]\, u(k)\}/[\lambda_i - \sigma(k)], \tag{15}$$

where the $\sigma(k)$ are scalars to be chosen. Essentially the same analysis may be given for (3), (15) as was given for the special case (3), (4). Again (9) is satisfied but (8) must be replaced by

$$\beta_p(k+1) = \beta_p(k)\,[\lambda_p - \sigma(k)]/[\lambda_i - \sigma(k)], \qquad p \neq i.$$

With a judicious choice of $\sigma(k)$, the ratios $|[\lambda_i - \sigma(k)]/[\lambda_1 - \sigma(k)]|$ $(i \neq 1)$ may generally all be made considerably smaller than $|\lambda_2/\lambda_1|$. If all $\sigma(k)$ are given the same value σ_0 then when $i = 1$ the convergence estimates (13) and (14) must be replaced by

$$\| u(k) - x_{1,j} \| = O(r^k) \qquad \text{and} \qquad |\mu(k) - \lambda_{1,j}| = O(r^k),$$

where

$$r = \max_{p \neq 1} |(\lambda_p - \sigma_0)/(\lambda_1 - \sigma_0)|.$$

Similarly calculation of $\lambda_{i,j}$ and $x_{i,j}$ for $i \neq 1$ by the extension of (3), (4) mentioned in the last section may be made more rapidly convergent by shifts of origin.

In the classical problem of calculating eigenvalues, techniques for efficient choice of origin shifts have been developed which do not require a priori knowledge of the other eigenvalues [3, pp. 576–578], and these techniques should be useful for selecting $\sigma(k)$ in (15). Choice of $\sigma(k)$ is especially easy when approximations to the other eigenvalues are known, as the value of $\sigma(k)$ which gives the fastest possible convergence may be expressed in terms of two of the other eigenvalues [3]. Note that while λ_1

and x_1 must be known with great accuracy for the algorithm to give accurate values of $\lambda_{1,j}$ or $x_{1,j}$, use of very crude estimates of other eigenvalues in choosing $\sigma(k)$ does not affect the accuracy of the result but merely gives a rate of convergence which, while usually faster than would be obtained with (3), (4), is slower than the fastest possible.

Recall that the simple algorithm (3), (4) is convergent only when $i = 1$. However, (3), (15) can generally be used to find the derivatives of two distinct λ_i and the corresponding x_i without recourse to the tedious extension used in [1]. For example when all the eigenvalues are real, the greatest and the least can each be made greatest in absolute value by appropriate origin shifts.

4. Simultaneous Iteration

Motivated by the success of simultaneous iteration techniques for the classical eigenvalue problem [4], we consider the following generalization of (3), (4). Let $\Lambda = \mathrm{diag}(\lambda_1, \ldots, \lambda_r)$ and let X be the $n \times r$ matrix whose ith column is x_i where $1 < r \ll n$. Define matrices $M(k)$, $U(k)$ by

$$M(k) = (X^*X)^{-1}\,[X^*A_{,j}X + X^*AU(k) - X^*U(k)\Lambda], \tag{16}$$

$$U(k+1) = [A_{,j}X + AU(k) - XM(k)]\Lambda^{-1}, \qquad k = 0, 1, 2, \ldots. \tag{17}$$

Premultiplication by $X(X^*X)^{-1}X^*$ represents orthogonal projection onto $\mathrm{sp}(x_1, \ldots, x_r)$, the space spanned by $x_1, \ldots, x_r$. Hence essentially the same argument as was used for (3), (4) shows that if $|\lambda_{r+1}| < |\lambda_r|$, then, even if all other inequalities in (10) are weak,

$$M(\infty) = \lim_{k\to\infty} M(k) \qquad \text{and} \qquad U(\infty) = \lim_{k\to\infty} U(k)$$

both exist, the rate of convergence is given by $|\lambda_{r+1}/\lambda_r|^k$, and all columns of $(U(\infty) - X_{,j})$ lie in $\mathrm{sp}(x_1, \ldots, x_r)$, regardless of the choice of $U(0)$. Hence, in view of the nonuniqueness, noted in Section 2, of the derivatives of eigenvectors when unique normalization is not required, the columns of $U(\infty)$ could be regarded as a basis for $\mathrm{sp}(x_{1,j}, \ldots, x_{r,j})$.

However, since $X^*U(k+1) = X^*U(k)$, $U(\infty) = X_{,j}$ if and only if

$$X^*U(0) = X^*X_{,j}\,. \tag{18}$$

Unfortunately, choice of $U(0)$ to satisfy (18) is much harder than when $r = 1$. Even when A is Hermitian and hence X^*X is constant, Re $X^*X_{,j}$ is not generally known in advance. If (16), (17) is to be used to obtain individual $x_{i,j}$, it must be combined with some device, such as that described at the end of Section 2, for eliminating the unwanted term, in this case the orthogonal projection of $U(k) - X_{,j}$ on $\mathrm{sp}(x_1, \ldots, x_r)$. This may well prove easier than the extension to (3), (4) for nondominant eigenvalues given

in [1] since with (16), (17) the unwanted term is not an unbounded function of k. Clearly if $U(\infty) = X_{,j}$ then $M(\infty) = \text{diag}(\lambda_{1,j}, \ldots, \lambda_{r,j})$.

5. Efficiency Considerations

Since the products $x_i^* A_{,j} x_i$, $x_i^*(A - \lambda_i I)$, and $A_{,j} x_i$ are independent of k, by far the most expensive step in the algorithm (3), (4) is the computation of $Au(k)$ which needs n^2 multiplications for each iterative step if all elements of A are nonzero. (As is usual with iterative methods, full advantage can be taken of any sparsity.) For a given number of iterative steps, (3), (15) involves exactly the same number of multiplications and divisions as (3), (4). Since it should require fewer iterations than (3), (4) it will be more efficient.

By way of comparison, it is easy to show that a direct method given in [1, Sect. 3] requires just over $n^2(n + 6)/3$ multiplications and divisions to calculate both $x_{i,j}$ and $\lambda_{i,j}$ once x_i, λ_i, $A_{,j}$ (and of course A) are accurately known, when the equations are solved by triangular decomposition [3, Chap. 4]. If only $x_{1,j}$ and $\lambda_{1,j}$ are required, the iterative method is likely to be more economical when the number of iterations required is less than $n/3$ (and possibly considerably more if A is sparse). Even when considerably more iterations are required, (3), (4) will often be more efficient than many previously published direct methods, some of which, unlike that given in [1], require the accurate calculation of a complete set of eigenvectors in order to calculate a single $x_{i,j}$.

Another direct method is given in [5]. More references are given in [6].

Acknowledgment

The author thanks one of the referees for a suggestion that led to the inclusion of Section 4.

References

1. C. S. Rudisill and Y.-Y. Chu, *AIAA J.* **13** (1975), 834.
2. P. Lancaster, *Numer. Math.* **6** (1964), 377.
3. J. H. Wilkinson, "The Algebraic Eigenvalue Problem," Clarendon Press, Oxford, 1965.
4. G. W. Stewart, *Numer. Math.* **25** (1976), 123.
5. R. B. Nelson, *AIAA J.* **14** (1976), 1201.
6. J. S. Arora, *J. Structural Div. ASCE* **102** (1976), 783.

Received: November 22, 1976

Alan L. Andrew
Mathematics Department,
La Trobe University,
Bundoora, Victoria 3083, Australia