SIAM J. MATRIX ANAL. APPL.
Vol. 13, No. 1, pp. 145–161, January 1992

NUMERICAL CONSIDERATIONS IN COMPUTING INVARIANT SUBSPACES*

JACK J. DONGARRA†, SVEN HAMMARLING‡, AND JAMES H. WILKINSON§

Dedicated to Gene Golub on the occasion of his 60th birthday

Abstract. This paper describes two methods for computing the invariant subspace of a matrix. The first method involves using transformations to interchange the eigenvalues. The matrix is assumed to be in Schur form and transformations are applied to interchange neighboring blocks. The blocks can be either one by one or two by two. The second method involves the construction of an invariant subspace by a direct computation of the vectors, rather than by applying transformations to move the desired eigenvalues to the top of the matrix.

Key words. invariant subspaces, eigenvalues, ill-conditioned eigenvalues

AMS(MOS) subject classification. 65F15

1. Introduction.

In this paper, we consider the computation of the invariant subspace of a matrix corresponding to some given group of eigenvalues. Potentially, the Schur factorization provides a method for computing such invariant subspaces, with the important numerical property that it provides an orthonormal basis for such spaces. Let us denote the Schur factorization of the real matrix A as

$$A = QTQ^T,$$

where Q is orthogonal and T block upper triangular, with 1×1 and 2×2 blocks on the diagonal, the 2×2 blocks corresponding to complex conjugate pairs of eigenvalues. Since

$$AQ = QT,$$

Q, of course, provides an orthonormal basis for the invariant subspace of the complete eigenvalue spectrum of A. Numerically, Q is a much more satisfactory basis than the eigenvectors and principal vectors of A, which may well be almost linearly dependent. If we partition Q and T as

$$Q = (Q_1 \; Q_2), \qquad T = \begin{pmatrix} T_{11} & T_{12} \\ 0 & T_{22} \end{pmatrix},$$

then

$$AQ_1 = Q_1 T_{11},$$

and Q_1 gives an orthonormal basis for the invariant subspace of A corresponding to the eigenvalues contained in T_{11}. It is therefore a common requirement to reorder T so that T_{11} has eigenvalues with some desired property. For example, we might require T_{11} to contain all the stable eigenvalues.

* Received by the editors October 12, 1990; accepted for publication (in revised form) January 22, 1991.

† Department of Computer Science, University of Tennessee, Knoxville, Tennessee 37996-1301, and the Mathematical Sciences Section, Oak Ridge National Laboratory, Oak Ridge, Tennessee 37831-8083. The work of this author was supported in part by the Applied Mathematical Sciences subprogram of the Office of Energy Research, U.S. Department of Energy contract DE-AC05-84OR21400, and in part by the Science Alliance, a state-supported program at the University of Tennessee (dongarra@cs.utk.edu).

‡ Numerical Algorithms Group Ltd., Wilkinson House, Jordan Hill Road, Oxford OX2 8DR, United Kingdom (na.hammarling@na-net.ornl.gov).

§ Work on this paper was started as a joint effort with James H. Wilkinson in 1983. After Jim's untimely death, the work lay unfinished for a number of years. The authors recently came across parts of Jim's handwritten manuscript and completed the work.

Unfortunately, unless we know the required group of eigenvalues in advance and accordingly modify the standard shift strategy of the QR algorithm, T_{11} will not normally contain the required eigenvalues on completion of the computation of the Schur factorization. We must therefore perform some further computation to reorder the eigenvalues. Indeed in most applications we perform an initial Schur factorization in order to compute the eigenvalues, which then gives us information on the required grouping.

An example of the application is the computation of matrix functions via the block diagonal form of a matrix. In computing the block diagonal form, it is essential to include "close" eigenvalues in the same diagonal block [3].

To this end, Stewart [9] has described an iterative algorithm for interchanging consecutive 1 × 1 and 2 × 2 blocks of the block triangular matrix. The first block is used to determine an implicit QR shift. An arbitrary QR step is performed on both blocks to eliminate the uncoupling between them. Then a sequence of QR steps using the previously determined shift is performed on both blocks. Except in ill-conditioned cases, the two blocks will interchange their positions.

In this paper, we present two other methods for constructing the invariant subspace. The first involves applying transformations directly to interchange the eigenvalues. The second method involves direct computation of the vectors.

2. Interchanging eigenvalues.

The reordering of the eigenvalues can be achieved by successively interchanging neighboring blocks in the Schur factor T.

Suppose, in a given T, we have decided to group λ_p, λ_q, λ_r together. We know that there exists a unitary matrix $\hat{Q}$ such that $\hat{T} = \hat{Q} T \hat{Q}^H$ is still upper triangular but has λ_p, λ_q, λ_r in the first three positions. Such a Q can be readily determined as the product of a finite number of plane rotations. We merely need an algorithm which will enable us to interchange consecutive blocks on the diagonal by means of a plane rotation. Repeated application of this algorithm can then bring any selected set of eigenvalues into the leading positions.

The algorithm we describe could be used on a complex triangular matrix. However, since we are interested here in real matrices, and since complex conjugate eigenvalues will be represented by 2 × 2 real diagonal blocks, we describe first the algorithm for interchanging two consecutive real eigenvalues.

2.1. Single past single.

Suppose λ and μ are in positions p and $p+1$. A similarity rotation in planes p and $p+1$ will alter only rows and columns p and $p+1$ and will retain the triangular form apart from the possible introduction of a nonzero in position $(p+1, p)$. The rotation can be chosen so as to interchange λ and μ while retaining the zero in $(p+1, p)$. Clearly the rotation is determined solely by the 2 × 2 matrix, which we denote by

$$\begin{pmatrix} \lambda & \alpha \\ 0 & \mu \end{pmatrix}.$$

We have

$$\begin{pmatrix} \lambda & \alpha \\ 0 & \mu \end{pmatrix}\begin{pmatrix} \alpha \\ \mu-\lambda \end{pmatrix} = \mu\begin{pmatrix} \alpha \\ \mu-\lambda \end{pmatrix}, \tag{1}$$

i.e., $(\alpha, \mu-\lambda)^T$ is the eigenvector corresponding to μ. If Q is chosen so that

$$Q\begin{pmatrix} \alpha \\ \mu-\lambda \end{pmatrix} = \begin{pmatrix} r \\ 0 \end{pmatrix}, \tag{2}$$

then

$$Q\begin{pmatrix}\lambda & \alpha\\ 0 & \mu\end{pmatrix}Q^TQ\begin{pmatrix}\alpha\\ \mu-\lambda\end{pmatrix}=\mu Q\begin{pmatrix}\alpha\\ \mu-\lambda\end{pmatrix},$$

and hence, using (2) and dividing by r, we have

$$Q\begin{pmatrix}\lambda & \alpha\\ 0 & \mu\end{pmatrix}Q^T\begin{pmatrix}1\\ 0\end{pmatrix}=\mu\begin{pmatrix}1\\ 0\end{pmatrix}=\begin{pmatrix}\mu\\ 0\end{pmatrix}.$$

This states that the first column of the transformed 2 × 2 is in the required form. Hence we may write

$$Q\begin{pmatrix}\lambda & \alpha\\ 0 & \mu\end{pmatrix}Q^T=\begin{pmatrix}\mu & \beta\\ 0 & \gamma\end{pmatrix}.$$

Since the trace and Frobenius norm are invariant,

$$\lambda+\mu=\mu+\gamma, \qquad \lambda^2+\mu^2+\alpha^2=\mu^2+\gamma^2+\beta^2,$$

giving

$$\gamma=\lambda \quad \text{and} \quad \beta=\pm\alpha.$$

A rotation giving (2) is defined by

$$\text{(3)} \qquad \cos\theta=\alpha/r, \qquad \sin\theta=(\mu-\lambda)/r, \qquad r=+[\alpha^2+(\mu-\lambda)^2]^{1/2},$$

and it will readily be verified that this gives $\beta=+\alpha$.

If the original T has been determined from a matrix A by means of an orthogonal transformation, the matrix defining this transformation must be updated by multiplication with the plane rotations used in the reordering process. Note that in this method, wherever two eigenvalues that we have decided to place in the same group are interchanged, a selected eigenvalue is moved up only past eigenvalues with which it is not to be associated. Moreover, having determined the rotation, we shall apply it to rows and columns p and $p+1$ but not to the 2 × 2 itself. There we shall merely interchange λ and μ and do no computation. Moving 1 × 1 blocks is discussed in [8].

2.2. Single past double. In bringing a selected real eigenvalue to a leading position, we shall, in general, need to pass 2 × 2 blocks on the diagonal corresponding to complex conjugate pairs. Hence we must be able to interchange a real eigenvalue with a real 2 × 2 block by means of an orthogonal similarity transformation. Obviously, the transformation is determined by the relevant 3 × 3 diagonal block which, for simplicity, we write as

$$\text{(4)} \qquad \left(\begin{array}{cc|c} * & * & b\\ * & * & c\\ \hline 0 & 0 & \lambda_3\end{array}\right)=\left(\begin{array}{cc|c} \multicolumn{2}{c|}{B} & b\\ & & c\\ \hline 0 & 0 & \lambda_3\end{array}\right).$$

The same principle may be used as in the single past single case. If

$$\text{(5)} \qquad \begin{pmatrix}x_1\\ x_2\\ 1\end{pmatrix}$$

denotes the eigenvector corresponding to λ_3, then we require a Q such that

$$Q\begin{pmatrix}x_1\\ x_2\\ 1\end{pmatrix}=\begin{pmatrix}r\\ 0\\ 0\end{pmatrix}$$

and then, as before,

$$QTQ^T = \left(\begin{array}{c|cc} \lambda_3 & x & x \\ \hline 0 & x & x \\ 0 & x & x \end{array}\right) = \left(\begin{array}{c|cc} \lambda_3 & x & x \\ \hline 0 & \multicolumn{2}{c}{\multirow{2}{*}{C}} \\ 0 & & \end{array}\right). \tag{6}$$

Note that the general principle we are using is the one commonly employed to establish the Schur canonical form by induction. The 2×2 matrix C in the bottom of (6) is not the same as B in (4), but it will, of course, have the same eigenvalues. However, B and C will not, in general, be orthogonally similar.

The matrix Q can be determined as one Householder matrix or as the product of two Givens rotations. Since λ_3 is real and B has complex conjugate eigenvalues, B can have no eigenvalues in common with λ_3; hence, a unique eigenvector of the form (5) will exist. As the two eigenvalues of B approach the real λ_3, their imaginary parts become small, and the eigenvector (5) will have progressively larger components in the first two positions, i.e., the normalized version will have a progressively smaller third component.

2.3. Double past single. When a pair of complex conjugate eigenvalues is included in the selected group, the associated 2×2 diagonal block has to be moved into a leading position on the diagonal. On the way up it will, in general, pass both single eigenvalues and 2×2 blocks with which it is not to be associated. We consider first taking a complex pair past a real eigenvalue. In other words, in terms of the relevant 3×3 matrix, we require an orthogonal Q such that

$$QTQ^T = \left(\begin{array}{c|cc} \lambda_1 & x & x \\ \hline 0 & \multicolumn{2}{c}{\multirow{2}{*}{B}} \\ 0 & & \end{array}\right) Q^T = \left(\begin{array}{cc|c} \multicolumn{2}{c|}{\multirow{2}{*}{C}} & x \\ & & x \\ \hline 0 & 0 & \lambda_1 \end{array}\right).$$

Here the selected eigenvalues are those of B, a complex conjugate pair. The eigenvalues of C will be the same pair, but in general C and B will be different matrices and will not be orthogonally similar. If we think in terms of moving λ_1 to the bottom, we may use much the same principle as before but now we work in terms of a left-hand eigenvector. If

$$y^T T_3 = \lambda y^T \quad \text{with } y^T = (1, y_2, y_3),$$

we determine a Q such that

$$y^T Q = (0, 0, x).$$

Then $Q^T T_3 Q$ has $(0, 0, \lambda_1)$ as its last row, and the objective has been achieved.

2.4. Double past double. Finally, we may need to move a selected 2×2 matrix past an unrelated 2×2. If we denote the relevant 4×4 matrix T_4 by

$$\left(\begin{array}{cc|cc} b_1 & b_2 & x & x \\ b_3 & b_4 & x & x \\ \hline \multicolumn{2}{c|}{\multirow{2}{*}{0}} & c_1 & c_2 \\ & & c_3 & c_4 \end{array}\right) = \left(\begin{array}{c|c} B & X \\ \hline 0 & C \end{array}\right),$$

then we require an orthogonal Q so that

$$\tilde{T}_4 = QT_4Q^T = \left(\begin{array}{c|c} \tilde{C} & \tilde{X} \\ \hline 0 & \tilde{B} \end{array}\right)$$

where B and C have the same eigenvalues as $\tilde{B}$ and $\tilde{C}$, respectively.

The same general principle may be used. We compute generators of the invariant subspace corresponding to C in the form

$$(x, y) = \begin{pmatrix} * & * \\ * & * \\ 1 & 0 \\ 0 & 1 \end{pmatrix}$$

by solving

$$T_4(x, y) = (x, y)C = (x, y)\begin{pmatrix} c_1 & c_2 \\ c_3 & c_4 \end{pmatrix}. \tag{7}$$

This gives us four equations for the four top components in (x, y). If we now determine a Q such that

$$Q(x, y) = \begin{pmatrix} * & * \\ 0 & * \\ 0 & 0 \\ 0 & 0 \end{pmatrix} = \begin{pmatrix} R \\ 0 \end{pmatrix},$$

then QT_4Q^T will be of the required form. Such a Q may be determined as the product of two Householder matrices or four Givens rotations.

To see how $\tilde{C}$ is related to C, we observe that (7) implies that

$$QT_4Q^TQ(x, y) = Q(x, y)C,$$

giving

$$QT_4Q^T\begin{pmatrix} R \\ 0 \end{pmatrix} = \begin{pmatrix} R \\ 0 \end{pmatrix}C,$$

that is,

$$QT_4Q^T\begin{pmatrix} I \\ 0 \end{pmatrix} = \begin{pmatrix} I \\ 0 \end{pmatrix}(RCR^{-1}).$$

This last equation states that the first two columns of QTQ^T are

$$\begin{pmatrix} RCR^{-1} \\ 0 \end{pmatrix},$$

and hence $\tilde{C} = RCR^{-1}$. We shall not, of course, compute $\tilde{C}$ via R!

3. Numerical considerations.

In each of the four cases discussed above, we determine either an eigenvector or two independent generators of an invariant subspace.

3.1. Single past single.

When taking a single past a single, the formulae giving the components of the vectors are of a particularly simple form. For consistency with the other three cases, the eigenvector in (1) should perhaps have been expressed in the form

$$(\alpha/(\mu - \lambda), 1)^T.$$

This emphasizes the fact that when $\mu - \lambda$ is very small compared with α, the first component of the eigenvector is very large, i.e., in the normalized form, the second component is very small. However, in this case λ and μ should almost certainly have been associated together, and we should not be trying to interchange them!

This remark has more force than might be imagined when the full $n \times n$ quasitriangular matrix has been produced from a general matrix A by an orthogonal similarity transformation. In this case, the elements below the diagonal elements are in no sense true zeros. They are at best negligible to working accuracy.

As an example, consider the matrix

$$(8) \qquad \begin{pmatrix} 1-\varepsilon & 1 \\ 0 & 1+\varepsilon \end{pmatrix}, \quad \lambda_1 = 1-\varepsilon, \quad \lambda_2 = 1+\varepsilon.$$

A perturbation $-\varepsilon^2$ in the (2, 1) element gives modified eigenvalues $\bar{\lambda}_1 = \bar{\lambda}_2 = 1$, and the matrix is defective. Suppose we are working on a ten-digit computer and $\varepsilon = 10^{-6}$. We may not think of 1 ± 10^{-6} as unduly close, but a perturbation of -10^{-12} gives coincident eigenvalues, and this perturbation is well below the negligible level. If we think in terms of perturbations of order 10^{-10} (i.e., computer noise level), all we can say is that the true eigenvalues are (roughly) in a disk centered on $\lambda = 1$ and of radius 10^{-5}. Thus a perturbation $+10^{-10}$ in (2, 1) gives eigenvalues $1 \pm i(.99)^{1/2}10^{-5}$, while a perturbation of -10^{-10} gives eigenvalues $1 + (1.01)^{1/2}10^{-5}$. To attempt to distinguish between $1 + 10^{-6}$ and $1 - 10^{-6}$, and to interchange them, makes no sense. They have no separate identity, and different rounding errors in the triangularization program giving T might well have led to complex eigenvalues and have a 2×2 block rather than that in (8).

For several moderately close eigenvalues, the remark has even greater force. Thus, if

$$T = \begin{pmatrix} 1-\varepsilon & 1 & 0 \\ & 1 & 1 \\ & & 1+\varepsilon \end{pmatrix},$$

$\lambda_1 = 1 - \varepsilon$, $\lambda_2 = 1$, $\lambda_1 = 1 + \varepsilon$, and $\varepsilon = 10^{-6}$. A perturbation even as small as 10^{-12} in (3, 1) gives three eigenvalues of the form $1 + O(10^{-4})$. This problem is discussed in considerable detail in [10], [12], [13]. Clearly, deciding which eigenvalues should be grouped together cannot be done on the superficial basis of "looking at the separations."

The remarkable fact is that in the single past single case, the $\cos\theta$ and $\sin\theta$ are always given with very low relative errors on a computer with correct rounding or chopping. On such computers, $\mu - \lambda$ is always computed without rounding errors even when severe cancellation takes place. Thus, if

$$\begin{pmatrix} .832567 & .912863 \\ 0 & .832569 \end{pmatrix},$$

we have on a six-digit computer $\mu - \lambda = .000002$, and this has no error. (This will be true even when, e.g., $\lambda = .999999$ and $\mu = 10^1(.100001)$, that is, when close λ and μ have different exponents.) Six-figure floating-point computation using (3) gives

$$\cos\theta = 10^1(.100000), \qquad \sin\theta = 10^{-5}(.219091),$$

and both of these have relative errors on the order of machine precision (10^{-12}) in spite of severe cancellation having taken place. Hence, if we actually do the computation of the 2×2 matrix (in practice, we would not; we could merely insert μ, λ, and α in the appropriate places), we find that the coupled (1, 1), (1, 2) and (2, 2) elements are correct to working accuracy and that the (2, 1) element is well below the negligible level. This is comforting because we shall be applying the transformation to the rest of the matrix.

This is an impressively good result. In many situations, not dissimilar from this, we would have to be satisfied with a matrix that is exactly similar to a T with a perturbation of order 10^{-6} in its elements and such a matrix could have eigenvalues agreeing with λ and μ in only the first three figures, a disaster from the point of view of effecting an interchange of λ and μ!

3.2. Single past double or double past single. When we turn to the other three cases, the situation is not so simple. Let us consider the algorithm for moving a single past a double. If we denote the eigenvector in (5) by

$$x = (x_1, x_2, 1)^T,$$

then $T_3 x = \lambda_3 x$ gives

$$(9) \qquad \begin{aligned} (t_{11} - \lambda_3)x_1 + t_{12}x_2 + t_{13} &= 0, \\ t_{21}x_1 + (t_{22} - \lambda_3)x_2 + t_{23} &= 0. \end{aligned}$$

The matrix of coefficients $\tilde{T}$ of this system of equations is

$$\tilde{T} = \begin{pmatrix} t_{11} - \lambda_3 & t_{12} \\ t_{21} & t_{22} - \lambda_3 \end{pmatrix},$$

which can be singular only if λ_3 is an eigenvalue of the leading 2×2 matrix of T_3. This possibility is specifically excluded since λ_3 is real and the 2×2 has complex eigenvalues (otherwise we would have triangularized it). When λ_3 is very well separated from the two complex eigenvalues, $\tilde{T}$ will be very well conditioned and x_1 and x_2 will not be large; hence, in the normalized version of x the third component will not be small. If we compute the transformation and apply it to the full 3×3 matrix, the top element will be λ_3 to high accuracy, the two complex eigenvalues will be accurately preserved, and the (3, 1) and (3, 2) elements will be negligible. The computed results will be very close to those derived by exact arithmetic.

As λ_3 approaches an eigenvalue of the 2×2 block, however (notice that this means that the imaginary parts of the complex eigenvalues must be small since λ_3 is real, and hence we are really moving towards a triple eigenvalue), the matrix $\tilde{T}$ will become progressively more ill conditioned, and in general, x_1 and x_2 will be larger. In the limiting situation, the eigenvector will have a zero third component and will be an eigenvector of the leading 2×2 matrix rather than one corresponding to λ_3 in the 3×3 matrix. The matrix Q is merely a plane rotation in the (1, 2) plane and does not affect λ_3. It is difficult to view this in terms of bringing the (3, 3) element into the leading position! Indeed, we are merely recognizing the fact that the upper 2×2 now has a double real root, and we are triangularizing it. Since the real roots that it has are the same as λ_3, however, the illusion of having moved λ_3 into the leading position is preserved. Thus, if

$$T_3 = \left(\begin{array}{cc|c} 1 & -1 & 0 \\ 1 & -1 & 1 \\ \hline 0 & 0 & 0 \end{array}\right), \qquad \lambda_1 = \lambda_2 = \lambda_3 = 0,$$

the only eigenvector is $(1, 1, 0)^T$; there is no eigenvector of the form $(x, x, 1)^T$. For the rotation in the (1, 2) plane, $\theta = \pi/4$ and the transformed matrix is

$$\begin{pmatrix} 0 & -2 & 1/\sqrt{2} \\ 0 & 0 & 1/\sqrt{2} \\ 0 & 0 & 0 \end{pmatrix} = \left(\begin{array}{c|cc} \lambda_3 & x & x \\ \hline 0 & \multicolumn{2}{c}{\multirow{2}{*}{C}} \\ 0 & & \end{array}\right).$$

The matrix is in the required form, with λ_3 in the leading position, zeros in the first column, and C given by

$$C=\begin{pmatrix}0 & 1/\sqrt{2}\\ 0 & 0\end{pmatrix}, \tag{10}$$

which is similar to the original 2×2, but certainly not orthogonally similar since it has a different Euclidean norm. However, considering how it has come about, it would be perverse to describe it as "bringing λ_3 past the 2×2."

Suppose now that we perturb the (2, 1) entry of the matrix by ε^2 to give

$$\begin{pmatrix}1 & -1 & 0\\ 1+\varepsilon^2 & -1 & 1\\ 0 & 0 & 0\end{pmatrix}, \quad \lambda_1,\lambda_2=\pm i\varepsilon, \quad \lambda_3=0.$$

Then there is an eigenvector x corresponding to λ_3 of the form

$$x^T=(-1/\varepsilon^2,-1/\varepsilon^2,1)=(-1/\varepsilon^2)(1,1,-\varepsilon^2).$$

The normalized version of this vector has a very small third component. If we perform our algorithm exactly, it gives a (2, 3) rotation with an angle of order ε^2 (the corresponding matrix is almost the identity matrix) while the (1, 2) rotation has an angle of almost exactly $\pi/4$. The resulting matrix has $\lambda_3=0$ in the leading position and the 2×2 matrix C is almost exactly as in (10), but has small perturbations that make its eigenvalues $\pm\varepsilon$.

The simplicity of this discussion is slightly obscured by the use of plane rotations and their introduction of irrationals. If we think in terms of nonorthogonal transformations, then to convert $(1, 1, -\varepsilon^2)$ to $(1, 0, 0)$, we perform a similarity with the unit lower triangular matrix

$$M=\begin{pmatrix}1 & & \\ -1 & 1 & \\ \varepsilon^2 & 0 & 1\end{pmatrix}$$

and obtain as our transformed matrix

$$\left(\begin{array}{c|cc}0 & -1 & 0\\ \hline 0 & 0 & 1\\ 0 & -\varepsilon^2 & 0\end{array}\right).$$

The zero eigenvalue is brought to the top and the eigenvalues $\pm i\varepsilon$ moved to the bottom in a transparently obvious way. When $\varepsilon=0$, the transformation operates only on row and column 1 and 2, and λ_3 is not involved. Nonetheless, the transformed matrix is

$$\left(\begin{array}{c|cc}0 & -1 & 0\\ \hline 0 & 0 & 1\\ 0 & 0 & 0\end{array}\right),$$

and our "objective" (inappropriate though it is) has been achieved.

The relevance of this discussion to the performance of our algorithm is the following. When we attempt to bring a single past a double having eigenvalues that are fairly close to it, we risk placing too much reliance on the effect achieved by the very small third component in the normalized version of the unique eigenvector corresponding to λ_3. In the analogous single past single case, the solution was determined with considerable accuracy. Here, however, the solution is not nearly as simple. Moreover, when the transformation has been computed, we shall need to apply it to the 3×3 matrix itself, as well as to the remainder of those relevant rows and columns, since the new 2 × 2 is not determined in a trivial manner, as were the elements in the single past single case.

Clearly the set of equations must be solved with some care. It is essential that the normalized version of

$$(x_1, x_2, 1), \quad \text{i.e.,} \quad (\hat{x}_1, \hat{x}_3, \hat{x}_3)$$

should be such that

$$(t_{11}-\lambda_3)\hat{x}_1+t_{12}\hat{x}_2+t_{13}\hat{x}_3=\varepsilon_1,$$
$$t_{21}\hat{x}_1+t_{22}\hat{x}_2+t_{23}\hat{x}_3=\varepsilon_2$$

be true with ε_1 and ε_2, which are at noise level relative to the coefficients on the left-hand side (ε_1 and ε_2 would be zero with exact computation). The solution of the system by Gaussian elimination with pivoting ensures just that; it produces x_1 and x_2 with errors that are so correlated that the normalized versions give residuals at noise level.

In place of Gaussian elimination with pivoting, we could use any stable direct method to solve the system, e.g., Givens triangulation. However, if we were to solve the system by an unstable method such as Cramer's rule in standard floating-point arithmetic, we would obtain a computed x_1 and x_2 with errors that are uncorrelated, and the residual corresponding to the normalized vector would not then be at noise level.

Assuming, then, that we have a normalized eigenvector giving negligible residuals, the process is satisfactory. Indeed, it is merely the method of deflation by orthogonal similarity transformations that is used after finding an eigenvector of a general matrix (see, e.g., [11, § 20, Chap. 9]). This is a stable deflation, provided the eigenvector has negligible residuals (independent of its absolute accuracy); the deflated matrix is exactly orthogonally similar to a matrix that differs from the original by a matrix E, which is at noise level relative to it. This is true even when we insert (without computation) the computed eigenvalue in the leading position and zero in the rest of the first column. Such a result is the most we can reasonably expect, though it falls somewhat short of the super-stability of the single past single case.

We have naturally concentrated on the case when we are attempting to move a real eigenvalue λ_3 past a complex conjugate pair, each of which is near λ_3, because numerical stability there needs serious investigation. Of course, when λ_3 is "too close," we usually include all three eigenvalues in the same space. However, when we move a single eigenvalue λ_3 past a complex conjugate pair $\lambda \pm i\mu$ such that $\lambda - \lambda_3$ is not small but μ is small, that pair will be close, and hence, in general, very sensitive to perturbations. The 2 × 2 block will itself be subjected to a similarity transformation, and small rounding errors will make substantial changes in the eigenvalues. Thus, if we have the matrix

$$\begin{pmatrix} .431263 & .516325 \\ -.000003 & .431937 \end{pmatrix}$$

with the ill-conditioned eigenvalues $.431600 \pm i(.001198)$, and subject it to a plane rotation with angle $\pi/4$, the exact transform gives

$$\begin{pmatrix} .689761 & .258501 \\ -.257827 & .173439 \end{pmatrix}$$

with, of course, precisely the same eigenvalues. If rounding errors produced

$$\begin{pmatrix} .689760 & .258501 \\ -.257827 & .173440 \end{pmatrix}$$

(i.e., changes of -1 and $+1$ in the last figures of the (1, 1) and (2, 2) elements), the eigenvalues become $.431600 \pm i(.001397)$, a substantial change in the imaginary parts. Yet in this example we have used an orthogonal similarity transformation that is favorable to numerical stability. In general, the bypassed matrix will be subjected to a nonorthogonal similarity transformation.

3.3. Double past double. Finally, we turn to the problem of moving a double past a double. Since two pairs of complex conjugate eigenvalues $\lambda_1 \pm \mu_1$ and $\lambda_2 \pm i\mu_2$ are involved, it is not possible for just one eigenvalue in the lower pair to agree with one in the upper pair. If, for convenience, we denote the relevant 4×4 matrix and the invariant subspace by

$$\left(\begin{array}{c|c} T_{11} & T_{12} \\ 0 & T_{22} \end{array}\right) \quad \text{and} \quad \begin{pmatrix} X \\ I \end{pmatrix},$$

respectively, where T_{11}, T_{12}, T_{22}, and X are 2×2 matrices, then we have

$$T_{11}X + T_{12} = XT_{22}.$$

It is well known that if T_{11} and T_{22} have no eigenvalue in common, then this is a nonsingular system.

For the case when T_{11} and T_{22} share an eigenvalue, consider the matrix

$$T = \begin{pmatrix} T_{11} & T_{12} \\ & T_{22} \end{pmatrix} = \left(\begin{array}{cc|cc} 1 & -1 & 0 & 0 \\ 1 & -1 & 1 & 0 \\ \hline & & 0 & 1 \\ & & 0 & 0 \end{array}\right), \quad \lambda_i = 0, \quad i = 1, \cdots, 4.$$

If we try to find an invariant subspace of the form $\binom{X}{I}$, we fail; the elements of X turn out to be infinite. There is no invariant subspace of dimension 2 of the required form. (The particular form chosen for T_{12} is not critical, though, of course, if we take T_{12} to be null, such an invariant subspace does exist with $X = 0$; T is then derogatory.) However,

$$T\begin{pmatrix} I \\ 0 \end{pmatrix} = \begin{pmatrix} T_{11} \\ 0 \end{pmatrix} = \begin{pmatrix} I \\ 0 \end{pmatrix} T_{11},$$

and hence we now have an invariant subspace, which we think of as belonging to T_{11}. But

$$QT_{11}Q^T = \begin{pmatrix} 0 & -2 \\ 0 & 0 \end{pmatrix} \quad \text{when} \quad Q = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ -1 & 1 \end{pmatrix} \text{ (a rotation).}$$

Hence

$$T\begin{pmatrix} I \\ 0 \end{pmatrix} Q^T = \begin{pmatrix} I \\ 0 \end{pmatrix} Q^T (QT_{11}Q^T),$$

i.e.,

$$T\begin{pmatrix} Q^T \\ 0 \end{pmatrix} = \begin{pmatrix} Q^T \\ 0 \end{pmatrix}\begin{pmatrix} 0 & -2 \\ 0 & 0 \end{pmatrix} = \begin{pmatrix} Q^T \\ 0 \end{pmatrix} M.$$

But

$$\begin{pmatrix} -\frac{1}{2} & \\ & 1 \end{pmatrix} M \begin{pmatrix} -2 & \\ & 1 \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} = T_{22}, \quad \text{i.e.,} \quad DMD^{-1} = T_{22}$$

and hence

$$T\begin{pmatrix} Q^T \\ 0 \end{pmatrix} D^{-1} = \begin{pmatrix} Q^T \\ 0 \end{pmatrix} D^{-1}(DMD^{-1}) = \begin{pmatrix} Q^T \\ 0 \end{pmatrix} D^{-1} T_{22},$$

i.e.,

$$T\begin{pmatrix} Q^T D^{-1} \\ 0 \end{pmatrix} = \begin{pmatrix} Q^T D^{-1} \\ 0 \end{pmatrix} T_{22}.$$

The columns of $\left(\begin{smallmatrix} Q^T D^{-1} \\ 0 \end{smallmatrix}\right)$ are orthogonal, but not orthonormal. It looks as though we have an orthogonal basis of an invariant subspace "belonging to T_{22}," but we should not really speak in these terms.

Nevertheless, if we consider

$$T(\varepsilon) = \left(\begin{array}{cc|cc} 1 & -1 & 0 & 0 \\ 1 & -1 & 1 & 0 \\ \hline & & 0 & 1 \\ & & -\varepsilon^2 & 0 \end{array}\right) = \left(\begin{array}{c|c} T_{11} & T_{12} \\ \hline & T_{22}(\varepsilon) \end{array}\right), \quad \lambda_1, \lambda_2 = 0, \quad \lambda_3, \lambda_4 = \pm i\varepsilon,$$

then there is a subspace of the form $\left(\begin{smallmatrix} X(\varepsilon) \\ I \end{smallmatrix}\right)$ which we could justifiably describe as "belonging to $T_{22}(\varepsilon)$," provided $\varepsilon \neq 0$. The elements of $X(\varepsilon)$ will tend to ∞ as $\varepsilon \to 0$ so that any normalized version of this invariant subspace will have very small components in its lower 2×2 matrix. In fact, since

$$T(\varepsilon)\begin{pmatrix} Q^T D^{-1} \\ 0 \end{pmatrix} = T\begin{pmatrix} Q^T D^{-1} \\ 0 \end{pmatrix},$$

we observe that

$$\begin{aligned} T(\varepsilon)\begin{pmatrix} Q^T D^{-1} \\ 0 \end{pmatrix} - \begin{pmatrix} Q^T D^{-1} \\ 0 \end{pmatrix} T_{22}(\varepsilon) &= T\begin{pmatrix} Q^T D^{-1} \\ 0 \end{pmatrix} - \begin{pmatrix} Q^T D^{-1} \\ 0 \end{pmatrix} T_{22}(\varepsilon) \\ &= T\begin{pmatrix} Q^T D^{-1} \\ 0 \end{pmatrix} - \begin{pmatrix} Q^T D^{-1} \\ 0 \end{pmatrix}\left(T_{22} + \begin{pmatrix} 0 & 0 \\ -\varepsilon^2 & 0 \end{pmatrix}\right) \\ &= \begin{pmatrix} Q^T D^{-1} \\ 0 \end{pmatrix}\begin{pmatrix} 0 & 0 \\ -\varepsilon^2 & 0 \end{pmatrix}. \end{aligned}$$

When ε is small, this invariant subspace gives negligible residuals "corresponding to $T_{22}(\varepsilon)$."

Can we expect $X(\varepsilon)$ to be $Q^T D^{-1}$ apart from a scale factor? Unfortunately we cannot. In fact, we have

$$\varepsilon^2\begin{pmatrix} X(\varepsilon) \\ I \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 1 & -1 \\ \varepsilon^2 & 0 \\ 0 & \varepsilon^2 \end{pmatrix}.$$

4. A direct method for computing invariant subspaces. In this section, we consider the construction of an invariant subspace by a direct computation of the vectors, rather than by applying transformations to move the desired eigenvalues to the top of the matrix T. We assume that the matrix T is derived from some square general matrix A. Suppose λ_k is the kth eigenvalue along the diagonal of T and T_{kk} is the leading $k \times k$ minor in the matrix T.

If λ_k is a simple eigenvalue, we just solve

$$(T-\lambda_k I)x=0.$$

This gives $x_{k+1}, x_{k+2}, \cdots, x_n = 0$. Next, we take $x_k = 1$ and solve

$$(T_{kk}-\lambda_k I)x=0$$

for $x_{k-1}, x_{k-2}, \cdots, x_2, x_1$, so the vector x will have the form

$$x=(x_1, x_2, \cdots, x_{k-1}, 1, 0, \cdots, 0)^T.$$

Now suppose α is a multiple eigenvalue, say a triple, such that

$$\alpha=\lambda_p=\lambda_q=\lambda_r, \qquad (p<q<r).$$

In general, there will be only one eigenvector corresponding to α (unless T is derogatory). First, we find the eigenvector x corresponding to λ_p by solving

$$(T_{pp}-\alpha I)x=0.$$

Next we attempt to find y corresponding to λ_q by taking $y_q = 1$ and attempting to solve

$$(T_{qq}-\lambda_q I)y=0, \quad \text{i.e.,} \quad (T_{qq}-\alpha I)y=0.$$

All is fine until we reach the determination of y_p. We have

$$0y_p+t_{p,p+1}y_{p+1}+\cdots+t_{p,q-1}y_{q-1}+t_{p,q}=0.$$

If we let

$$t_{p,p+1}y_{p+1}+\cdots t_{p,q-1}y_{q-1}+t_{p,q}=d,$$

then

$$0y_p+d=0.$$

If d happens to be zero, then y_p is arbitrary.

It is simplest to take $y_p = 0$. Hence, when $d = 0$, we obtain

$$x=(x_1, x_2, \cdots, x_{p-1}, 1, 0, \cdots, 0, 0, \cdots, 0)^T,$$

$$y=(y_1, y_2, \cdots, y_{p-1}, 0, y_{p+1}, \cdots, y_{q-1}, 1, \cdots, 0)^T.$$

These two vectors are obviously linearly independent. Hence we have two eigenvectors corresponding to α. Both satisfy $(T-\alpha I)x = 0$ and $(T-\alpha I)y = 0$.

If we had taken y_p to be m instead of zero, the solution would have been $y + mx$. This is fine since $y + mx$ is also an eigenvector. We could have chosen $y + mx$ orthogonal to x,

$$x^H(y+mx)=0, \qquad m=(-x^H y/x^H x).$$

That the matrix will be derogatory is much less probable than that it will be defective. In fact, even if A were exactly derogatory, T would probably not be, even if it still had exact multiple eigenvalues.

Suppose now $d \neq 0$. To get y_p, we would need to solve

$$0y_p=-d.$$

Hence we cannot get a second eigenvector. Note that if λ_q were $\lambda_p + \varepsilon$ instead of λ_p, we would be solving

$$\varepsilon y_p=-d$$

at this stage, giving an erroneous value of y_p. Obviously, in this case the first p components of y would be essentially $(-d/\varepsilon)x$ + (vector that is not too large). As $\varepsilon \to 0$, the vector y tends to a multiple of x with a relatively negligible amount of interference. In the limit we find that y and x are in exactly the same direction; the last $q-p$ components of y are negligible compared with the rest when q is small, and arbitrarily vanish altogether in the normalized y.

We cannot find a second eigenvector. We can, however, find a vector y such that

$$(T-\lambda_q I)y = dx.$$

Hence the determination of y proceeds as before, from y_q to y_{p+1}, since x is zero in these components. We now have

$$0y_p + t_{p,p+1}y_{p+1} + \cdots + t_{p,q-1}y_{q-1} + t_{p,q} = dx_p = d,$$

so that

$$t_{p,p+1}y_{p+1} + \cdots + t_{p,q-1}y_{q-1} + t_{p,q} = d,$$

giving

$$0\,y_p = 0.$$

Again y_p is arbitrary, and it is simplest to take y_p to be zero. There are no further problems, and we have

$$x = (x_1, x_2, \cdots, x_{p-1}, 1, 0, \cdots, 0, 0, 0, \cdots, 0)^T,$$

$$y = (y_1, y_2, \cdots, y_{p-1}, 0, y_{p+1}, \cdots, y_{q-1}, 1, 0, \cdots, 0)^T$$

with $(T-\alpha I)x = 0$, $(T-\alpha I)y = dx$, or

$$T(x,y) = (x,y)\begin{pmatrix} \alpha & d \\ & \alpha \end{pmatrix}.$$

Now for the third vector, we shall ignore the possibility of its being derogatory for the moment. We attempt to solve

$$(T_{rr} - \alpha I)z = 0$$

starting with $z_r = 1$. We proceed as usual until we reach z_q. At this stage we have

$$0z_q + t_{q,q+1}z_{q+1} + \cdots + t_{q,r-1}z_{r-1} + t_{q,r} = 0$$

so that

$$t_{q,q+1}z_{q+1} + \cdots + t_{q,r-1}z_{r-1} + t_{q,r} = e.$$

Hence, we solve

$$(T_{rr} - \alpha I)z = ey.$$

This does not affect the components already computed since $y_i = 0$, $(i > q)$.

For convenience we then take $z_q = 0$. We continue until reaching z_p. We now have

$$0z_p + t_{p,p+1}z_{p+1} + \cdots t_{p,r-1}z_{r-1} + t_{p,r} = ey_p,$$

i.e.,

$$t_{p,p+1}z_{p+1} + \cdots + t_{p,r-1}z_{r-1} + t_{p,r} = f.$$

If $f \neq 0$, we would get $z_p = 0$. To avoid this situation, we solve

$$(T_\alpha - \alpha I)z = ey + fx.$$

This does not affect previous components since $x_i = 0$ for $i > p$. The equation for z_p then becomes

$$0z_p = 0.$$

If we take $z_p = 0$ and determine $z_{p-1}, z_{p-2}, \cdots, z_1$, we then have

$$(T - \alpha I)x = 0, \quad (T - \alpha I)y = dx, \quad (T - \alpha I)z = ey + fx,$$

or

$$(11) \qquad (T - \alpha I)(x, y, z) = (x, y, z)\begin{pmatrix} \alpha & d & f \\ & \alpha & e \\ & & \alpha \end{pmatrix} = (x, y, z)T_\alpha.$$

$$x = (x_1, x_2, \cdots, x_{p-1}, 1, 0, 0, \cdots, 0, 0, 0, \cdots, 0)^T,$$
$$y = (y_1, y_2, \cdots, y_{p-1}, 0, y_{p+1}, y_{p+2}, \cdots, y_{q-1}, 1, 0, \cdots, 0)^T,$$
$$z = (z_1, z_2, \cdots, z_{p-1}, 0, z_{p+1}, z_{p+2}, \cdots, z_{q-1}, 0, z_{q+1}, \cdots, z_{r-1}, 1, 0, \cdots, 0)^T.$$

Clearly, x, y, z are linearly independent, and they span the three-dimensional invariant subspace associated with α. They are not orthogonal, in general, but we could develop an orthogonal basis from this. Specifically, if

$$(x, y, z) = (q_1, q_2, q_3)\begin{pmatrix} r_{11} & r_{12} & r_{13} \\ & r_{22} & r_{23} \\ & & r_{33} \end{pmatrix} = Q_3 R_3,$$

then

$$(T - \alpha I)Q_3 R_3 = Q_3 R_3 T_\alpha$$

or

$$(T - \alpha I)Q_3 = Q_3[R_3 T_\alpha R_3^{-1}] = Q_3 M.$$

Q_3 is now an orthogonal basis, and M has α as a triple eigenvalue.

A derogatory matrix will be revealed by zero values among d, e, f. Thus if $d = e = f = 0$, we get three independent eigenvectors, and

$$T(x, y, z) = (x, y, z)\begin{pmatrix} \alpha & & \\ & \alpha & \\ & & \alpha \end{pmatrix}.$$

If $d = f = 0$ and $e \neq 0$, we have

$$T(x, y, z) = (x, y, z)\begin{pmatrix} \alpha & & \\ & \alpha & e \\ & & \alpha \end{pmatrix}.$$

Then we have a linear divisor $(\lambda - \alpha)$ and one quadratic, $(\lambda - \alpha)^2$.

If all computations are exact and T comes from exact computation, then we associate only the eigenvalues that are truly equal, and the vectors obtained in the way we have described are truly independent. In practice, however, T will rarely be an exact matrix. Usually it will have been obtained from a matrix A by, say, the QR algorithm. Even if A had defective eigenvalues, T will usually not have any repeated diagonal elements. A real problem is to decide which diagonal elements to associate together. We may need to associate eigenvalues that are by no means pathologically close. If we have decided which eigenvalues we wish to associate, then we proceed exactly as described.

So far in this section we have tacitly assumed that T is exactly triangular, but the QR algorithm may give 2×2's on the diagonal. If a 2×2 corresponds to a pair of real eigenvalues, we can get rid of it by an orthogonal transformation. If it corresponds to a complex conjugate pair, we cannot. We assume then that all 2×2's correspond to complex conjugate eigenvalues.

We turn now to the case of 2×2 blocks. If we associate only real eigenvalues in an invariant subspace, there are no real new points. We merely need to know how to get the two components of any of our vectors in the position of a 2×2 block in the matrix. Clearly we solve a 2×2 system of equations for the two components. The technique for getting the generators and the M is unchanged.

Now, consider obtaining a pair of vectors spanning the two-space associated with complex conjugate pairs of eigenvalues, assuming for the moment that we are not associating it with any other eigenvalues. For T, illustrated by

$$T = \begin{pmatrix} * & * & * & * & * & * \\ & * & * & * & * & * \\ & & * & * & * & * \\ & & & * & * & * \\ & & & * & * & * \\ & & & & & * \end{pmatrix},$$

we merely solve the equations

$$(12) \qquad T(x_p, x_{p+1}) = (x_p, x_{p+1}) \begin{pmatrix} t_{p,p} & t_{p,p+1} \\ t_{p+1,p} & t_{p+1,p+1} \end{pmatrix}$$

and take

$$(x_p, x_{p+1}) = \begin{pmatrix} * & * \\ * & * \\ * & * \\ 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ \vdots & \vdots \end{pmatrix}$$

so that they are certainly independent. The two back substitutions for determining x_p and x_{p+1} are done as before. We determine $x_i^{(p)}$ and $x_i^{(p+1)}$ from the pair of equations obtained by equating row i on both sides of (12). This gives a well-separated pair of vectors even when the two eigenvectors are close, provided the earlier eigenvalues are well separated from them. Thus, for

$$\begin{pmatrix} 3 & 1 & 2 \\ 0 & 1 & 1 \\ 0 & -10^{-10} & 1 \end{pmatrix} \begin{pmatrix} x_1 & y_2 \\ 1 & 0 \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} x_1 & y_1 \\ 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 1 \\ -10^{-10} & 1 \end{pmatrix},$$

the eigenvalues are $1 \pm i10^{-5}$; they are close, but well separated from the other eigenvalue $\lambda_1 = 3$. The components x_1 and y_1 satisfy

$$3x_1 + 1 = x_1 - 10^{-10} y_1, \qquad 3y_1 + 2 = x_1 + y_1.$$

To eight decimals, $x_1 = -\frac{1}{2}$ and $y_1 = -\frac{5}{4}$. The vectors are extremely well separated and

$$T(x, y) - (x, y) \begin{pmatrix} 1 & 1 \\ -10^{-10} & 1 \end{pmatrix} = O(10^{-10}).$$

If, when computing the two vectors corresponding to a complex pair, we encounter another 2 × 2 block, say, in position $i, i+1$, then components i and $i+1$ of x and y are determined by solving a set of four linear equations derived by equating rows i and $i+1$ of (12). This will be a well-conditioned 4 × 4 system if λ_i, λ_{i+1} are well separated from λ_p, λ_{p+1}.

When we wish to associate $(\lambda_p, \lambda_{p+1})$ with some of the earlier eigenvalues (for which we have already done the back substitution), the solution is quite clear. When we encounter a real eigenvalue λ_i that is to be associated with them, we solve from that point on, namely,

$$T(x_p, x_{p+1}) = (x_p, x_{p+1})\begin{pmatrix} t_{p,p} & t_{p,p+1} \\ t_{p+1,p} & t_{p+1,p+1} \end{pmatrix} + (x_i)(d_1, d_2)$$

and we chose d_1 and d_2 so that the ith component of x_p and x_{p+1} are zero. This gives us a pair of equations for d_1 and d_2. If λ_p, λ_{p+1}, and λ_i were the only three to be associated, we would have for the invariant three-space

$$T(x_i, x_p, x_{p+1}) = \begin{pmatrix} t_{i,i} & d_1 & d_2 \\ 0 & t_{p,p} & t_{p,p+1} \\ 0 & t_{p+1,p} & t_{p+1,p+1} \end{pmatrix}.$$

If during the back substitution for x_p, x_{p+1} we encounter a pair λ_i, λ_{i+1} that we wish to associate with them, we solve from that point on

$$T(x_p, x_{p+1}) = (x_p, x_{p+1})\begin{pmatrix} t_{p,p} & t_{p,p+1} \\ t_{p+1,p} & t_{p+1,p+1} \end{pmatrix} + (x_i, x_{i+1})\begin{pmatrix} d_{i,i} & d_{i,i+1} \\ d_{i+1,i} & d_{i+1,i+1} \end{pmatrix},$$

where the four d's are chosen so as to make components i and $i+1$ of x_p and x_{p+1} equal to zero.

For example, suppose we group $(\lambda_9, \lambda_8), \lambda_6, (\lambda_4, \lambda_3)$ where (λ_9, λ_8) and (λ_4, λ_3) are complex pairs. We have

$$(x_3, x_4, x_6, x_8, x_9) = \left(\begin{array}{cc|c|cc} * & * & * & * & * \\ * & * & * & * & * \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ \hline 0 & 0 & * & * & * \\ 0 & 0 & 1 & 0 & 0 \\ \hline 0 & 0 & 0 & * & * \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ \hline 0 & 0 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 0 \end{array}\right)$$

and finally

$$T(x_3, x_4, x_6, x_8, x_9) = (x_3, x_4, x_6, x_8, x_9)\begin{pmatrix} t_{33} & t_{34} & d_{36} & d_{38} & d_{39} \\ t_{43} & t_{44} & d_{46} & d_{48} & d_{49} \\ & & t_{66} & d_{68} & d_{69} \\ & & & t_{88} & t_{89} \\ & & & t_{98} & t_{99} \end{pmatrix}.$$

The elements named d_{35} and d_{46} would have been determined when computing x_6 when we reached rows 3 and 4; the elements d_{68} and d_{69} would have been determined when computing x_8 and x_9 when we reached element 6; and the elements d_{48}, d_{49}, d_{38}, d_{39} would have been determined when we reached elements 4 and 3.

If we have made a good decision about our grouping, rows of the vectors will not be large, though this would not be sufficient to decide that the grouping is complete. First, there may be some λ_i which should also be associated with these five. Second, the vectors x_3, x_4, x_6, x_8, and x_9 might not be as linearly independent as we would like.

Other approaches have been suggested for computing the invariant subspace directly; see [4]–[6]. These are likely to be more stable but more expensive to compute.

5. Conclusions. The methods described in § 2 have been improved and generalized by Ng and Parlett [7] and implemented in LAPACK [1]. The LAPACK implementation includes tolerance checks and scaling to ensure numerical stability [2]. This is essentially achieved by not swapping blocks that are regarded as being too close.

We have discussed numerical issues concerned with the computation of invariant subspaces and proposed two methods related to their computation. The method discussed for swapping diagonal blocks can readily by extended to the generalized eigenvalue problem.

REFERENCES

[1] E. Anderson, Z. Bai, C. Bischof, J. Demmel, J. Dongarra, J. DuCroz, A. Greenbaum, S. Hammarling, A. McKenney, and D. Sorensen, LAPACK: *A portable linear algebra library for high-performance computers*, in Supercomputer 90, Institute of Electrical and Electronics Engineers Press, New York, 1990.

[2] Z. Bai, J. Demmel, and A. McKenney, *On the conditioning of the nonsymmetric eigenproblem: Theory and software*, Computer Science Department Tech. Report CS-89-86, University of Tennessee, Knoxville, TN, October 1990. (LAPACK Working Note 13.)

[3] C. Bavely and G. W. Stewart, *An algorithm for computing reducing subspaces by block diagonalization*, SIAM J. Numer. Anal., 16 (1979), pp. 359–367.

[4] J. Demmel, *Three methods for refining estimates of invariant subspaces*, Computing, 38 (1987), pp. 43–57.

[5] B. Kågström and A. Ruhe, *Algorithm 560: An algorithm for numerical computation of the Jordan normal form of a complex matrix*, ACM Trans. Math. Software, 6 (1980), pp. 398–419.

[6] ———, *An algorithm for numerical computation of the Jordan normal form of a complex matrix*, ACM Trans. Math. Software, 6 (1980), pp. 437–443.

[7] K. C. Ng and B. N. Parlett, *Programs to swap diagonal blocks*, Center for Pure and Applied Mathematics Tech. Report 381, University of California, Berkeley, CA, 1988.

[8] A. Ruhe, *Perturbation bounds for means of eigenvalues and invariant subspaces*, BIT, 10 (1970), pp. 343–354.

[9] G. W. Stewart. *Algorithm 406:* HQR3 *and* EXCRNG: *Fortran subroutines for calculating and ordering eigenvalues of a real upper Hessenberg matrix*, ACM Trans. Math. Software, 2 (1976), pp. 275–280.

[10] H. J. Symm and J. H. Wilkinson, *Realistic error bounds for a simple eigenvalue and its associated eigenvector*, Numer. Math., 35 (1980), pp. 113–126.

[11] J. H. Wilkinson, *The Algebraic Eigenvalue Problem*, Oxford University Press, London, 1965.

[12] ———, *Sensitivity of eigenvalues*, Utilitas Math., 25 (1984), pp. 5–76.

[13] ———, *Sensitivity of eigenvalues* II, Utilitas Math., 30 (1986), pp. 243–275.