On Swapping Diagonal Blocks in Real Schur Form*

Zhaojun Bai[†]
Department of Mathematics
University of Kentucky
Lexington, Kentucky 40506

and

James W. Demmel[‡]
Computer Science Division and Mathematics Department
University of California
Berkeley, California 94720

Submitted by G. W. Stewart

ABSTRACT

We discuss a new version of an existing algorithm for reordering the eigenvalues on the diagonal of a matrix in real Schur form by performing an orthogonal similarity transformation. A detailed error analysis and software description are presented. Numerical examples show the superiority of our algorithm over previous algorithms.

1. INTRODUCTION

The problem of reordering the eigenvalues into a desired order along the (block) diagonal of a quasitriangular real matrix arises in several applications: computing an invariant subspace corresponding to a given group of eigenvalues, estimating condition numbers for a cluster of eigenvalues or their

* This work was supported in part by NSF grant ASC-9005933 and DARPA grant DAAL03-91-C-0047 via a subcontract from the University of Tennessee. The first author was also supported in part by NSF grant ASC-9102963.

[†] E-mail: na.bai@na-net.ornl.gov.

[‡] E-mail: na.demmel@na-net.ornl.gov.

655 Avenue of the Americas, New York, NY 10010 0024-3795/93/$6.00

associated invariant subspace [18, 2], computing partial eigenvalues of a large nonsymmetric matrix by the simultaneous iteration method [14], computing matrix functions [4, 11], solving the linear-quadratic control problem [10], and so on. These problems can be solved in two phases: the first is to compute the Schur decomposition of the given matrix, and the second is to reorder a group of specified eigenvalues to appear at the upper left corner of the matrix. In this paper we describe an algorithm and its implementation for this reordering problem. The software is available in LAPACK [1], a public domain numerical linear algebra library.

Specifically, for a real matrix A, there is a real orthogonal matrix Q such that

$$A = QTQ^{\mathrm{T}}, \tag{1}$$

where T is a real upper quasitriangular matrix, called the *real Schur form*. This means that T is block upper triangular with 1×1 and 2×2 blocks on the diagonal. The 1×1 blocks contain the real eigenvalues of A. The eigenvalues of the 2×2 diagonal blocks are the complex conjugate eigenvalues of A. The real Schur form may be computed using subroutine HQR from EISPACK [13] or subroutine SHSEQR from LAPACK [1]. Here Q provides an orthonormal basis for the invariant subspaces of certain subsets of eigenvalues of the matrix A. If we partition Q and T conformally as

$$Q = [Q_1, \quad Q_2], \qquad T = \begin{bmatrix} T_{11} & T_{12} \\ 0 & T_{22} \end{bmatrix},$$

then from (1) we have

$$AQ_1 = Q_1 T_{11}, \tag{2}$$

and hence Q_1 gives an orthonormal basis for the invariant subspace of A corresponding to the eigenvalues contained in T_{11}.

Unfortunately, the T_{11} given by the QR algorithm will not generally contain the eigenvalues in which we are interested. We must therefore perform some further orthogonal similarities that preserve block triangular form but reorder the desired eigenvalues of A to the upper left corner of the Schur form T. The crux of such a reordering is to swap two adjacent 1×1 or 2×2 diagonal blocks by an orthogonal transformation. Formally, let A_{11} be a $p \times p$ matrix, A_{22} be a $q \times q$ matrix, $p, q = 1$ or 2; we want to compute an orthogonal $(p + q) \times (p + q)$ matrix Q such that

$$Q^{\mathrm{T}} \begin{bmatrix} A_{11} & A_{12} \\ 0 & A_{22} \end{bmatrix} Q = \begin{bmatrix} \tilde{A}_{22} & \tilde{A}_{12} \\ 0 & \tilde{A}_{11} \end{bmatrix}, \tag{3}$$

where $\tilde{A}_{ii}$ is similar to A_{ii}, $i = 1, 2$, so that the eigenvalues are unchanged but their positions are exchanged along the (block) diagonal.

To this end, Stewart [15] has described an iterative algorithm for swapping consecutive 1×1 and 2×2 blocks of a quasitriangular matrix, which we refer to as algorithm EXCHNG. In his method, the first block is used to determine an implicit QR shift. An arbitrary QR step is performed on both blocks to create a dense $(p + q) \times (p + q)$ matrix. Then a sequence of QR steps using the previously determined shift is performed. Theoretically, after one step of QR iteration, the eigenvalues of the first block will emerge in the lower part. But in practice, two or even more QR iterations may still fail to reorder the eigenvalues for some hard problems. This use of QR iteration has been extended by Van Dooren [19] to reordering the eigenvalues of a generalized eigenvalue problem using QZ iteration.

Another algorithm to be further developed in this paper is the so-called *direct swapping method*, which was originally motivated by the work of Ruhe [12], and by that of Dongarra, Hammarling, and Wilkinson (in 1983, although the paper was not finished until 1991 [7]). Ng and Parlett [11] also developed a program to implement the direct swapping algorithm. A similar idea has also been published by Cao and Zhang [6].

This previous work still does not solve the problem satisfactorily. The iterative swapping algorithm has the advantage of guaranteed backward stability, since it just multiplies the data by orthogonal matrices. But it may be inaccurate and even fail to reorder the eigenvalues in ill-conditioned cases. On the other hand, the direct swapping algorithm is simple and can better deal with ill-conditioned cases. But there are examples where these implementations fail to be stable.

In this paper, we further improve the direct swapping algorithm. Various strategies have been designed at each stage of the algorithm to improve its accuracy and robustness. A detailed analysis of the algorithm shows that backward instability is possible only in very ill-conditioned cases, so ill-conditioned in fact that we have been unable to construct a case where it fails. Our goal was to have an absolute stability guarantee, however; we achieved this by directly and cheaply testing for instability and rejecting a swap if it would have been unstable. This can occur only when the eigenvalues are so ill-conditioned as to be indistinguishable in a certain reasonable sense. Numerical experiments show the superiorities of our direct swapping algorithm over previous implementations.

The rest of the paper is organized as follows: Section 2 describes the direct swapping algorithm. The error analysis of the algorithm is carried out in Section 3. The software implementation and numerical experiments are reported in Section 4. Section 5 draws conclusions. All software including test software for the algorithms in this paper can be found in the LAPACK library [1].

We assume that any 2 × 2 diagonal block in the quasitriangular matrix is in standardized form. This means that its diagonal entries are equal and its off-diagonals nonzero and of opposite sign:

$$\begin{bmatrix} \alpha & \beta \\ \gamma & \alpha \end{bmatrix}, \qquad \beta\gamma < 0. \tag{4}$$

For any 2 × 2 block with complex conjugate eigenvalues, we can easily compute an orthogonal similarity transformation to standardize the block.

2. DIRECT SWAPPING ALGORITHM

As we described in the introduction, the crux of reordering the diagonal blocks is to interchange the consecutive diagonal blocks A_{11} and A_{22} in the following block matrix:

$$A = \begin{bmatrix} A_{11} & A_{12} \\ 0 & A_{22} \end{bmatrix}, \tag{5}$$

where A_{11} is $p \times p$, A_{22} is $q \times q$, and p, $q = 1$ or 2. Throughout this paper, we assume that A_{11} and A_{22} have no eigenvalue in common; otherwise, they need not be exchanged. It is seen that the block matrix (5) can be block diagonalized as

$$\begin{bmatrix} A_{11} & A_{12} \\ 0 & A_{22} \end{bmatrix} = \begin{bmatrix} I_p & -X \\ 0 & I_q \end{bmatrix} \begin{bmatrix} A_{11} & 0 \\ 0 & A_{22} \end{bmatrix} \begin{bmatrix} I_p & X \\ 0 & I_q \end{bmatrix},$$

where X is the solution of the Sylvester equation

$$A_{11}X - XA_{22} = A_{12}. \tag{6}$$

Since it is assumed that A_{11} and A_{22} have no eigenvalue in common, the solution X exists and is unique. If we choose an orthogonal matrix Q such that

$$Q^{\mathrm{T}}\begin{bmatrix} -X \\ I_q \end{bmatrix} = \begin{bmatrix} R \\ 0 \end{bmatrix}$$

and conformally partition Q in the form

$$Q = \begin{bmatrix} Q_{11} & Q_{12} \\ Q_{21} & Q_{22} \end{bmatrix},$$

then

$$Q^{\mathrm{T}}\begin{bmatrix} -X & I_p \\ I_p & 0 \end{bmatrix} = \begin{bmatrix} R & Q_{11}^{\mathrm{T}} \\ 0 & Q_{12}^{\mathrm{T}} \end{bmatrix}.$$

Since both matrices on the left are invertible, so are R and Q_{12}^{T}. Thus

$$\begin{aligned} Q^{\mathrm{T}}\begin{bmatrix} A_{11} & A_{12} \\ 0 & A_{22} \end{bmatrix} Q &= Q^{\mathrm{T}}\begin{bmatrix} I_p & -X \\ 0 & I_q \end{bmatrix} \begin{bmatrix} A_{11} & 0 \\ 0 & A_{22} \end{bmatrix} \begin{bmatrix} I_p & X \\ 0 & I_q \end{bmatrix} Q \\ &= \begin{bmatrix} R & Q_{11}^{\mathrm{T}} \\ 0 & Q_{12}^{\mathrm{T}} \end{bmatrix} \begin{bmatrix} A_{22} & 0 \\ 0 & A_{11} \end{bmatrix} \begin{bmatrix} R^{-1} & -R^{-1}Q_{11}^{\mathrm{T}}Q_{12}^{-\mathrm{T}} \\ 0 & Q_{12}^{-\mathrm{T}} \end{bmatrix} \\ &= \begin{bmatrix} RA_{22}R^{-1} & -RA_{22}R^{-1}Q_{11}^{\mathrm{T}}Q_{12}^{-\mathrm{T}} + Q_{11}^{\mathrm{T}}A_{11}Q_{12}^{-\mathrm{T}} \\ 0 & Q_{12}^{\mathrm{T}}A_{11}Q_{12}^{-\mathrm{T}} \end{bmatrix} \\ &\equiv \begin{bmatrix} \tilde{A}_{22} & \tilde{A}_{12} \\ 0 & \tilde{A}_{11} \end{bmatrix}, \end{aligned}$$

where $\tilde{A}_{ii}$ is similar to A_{ii}, $i = 1, 2$, so that the eigenvalues are invariant, but their positions are exchanged. Furthermore, we have the following theorem to specify such orthogonal transformation:

THEOREM 1(Ng and Parlett [11]). *An orthogonal $(p + q) \times (p + q)$ matrix Q swaps A_{11} and A_{22} if and only if*

$$Q^{\mathrm{T}}\begin{bmatrix} -X \\ I_q \end{bmatrix} = \begin{bmatrix} R \\ 0 \end{bmatrix} \tag{7}$$

for some invertible $q \times q$ matrix R where X is defined in (6).

In the presence of rounding errors, the biggest concern is solving the Sylvester equation (6). It could possibly be ill conditioned if A_{11} and A_{22} have close eigenvalues. In the extreme case, if A_{11} and A_{22} have the same eigenvalues, the Sylvester equation is singular and the solution X may be infinite. To prevent possible overflow, we instead solve the equation

$$A_{11}X - XA_{22} = \gamma A_{12} \tag{8}$$

or the corresponding linear system

$$Kx = \gamma b, \tag{9}$$

where γ is a scaling factor ($\gamma \leqslant 1$), $K = I_q \otimes A_{11} - A_{22}^{\mathrm{T}} \otimes I_p$, $\otimes$ is the Kronecker product, $x = \mathrm{col}(X)$, and $b = \mathrm{col}(A_{12})$. Here col(W) denotes the column vector formed by taking columns of W and stacking them atop one another from left to right. Possible overflow of X is taken care of by choosing a small scaling factor γ. In the extreme case, when A_{11} and A_{22} have the same eigenvalues, we choose $\gamma = 0$. Because the linear system (9) can only be 1×1, 2×2, or 4×4, it does not cost too much to use Gaussian elimination with complete pivoting to solve it with better numerical properties (in particular, the pivots are within a modest factor of the singular values of the 4×4 matrix, so setting tiny pivots to a chosen tiny value controls the conditioning of the system and norm of the solution). Applying standard results from [20], a straightforward analysis shows that for the computed solution $\bar{X}$ of the Sylvester equation one has

$$\frac{\|E\|_{\mathrm{F}}}{\|X\|_{\mathrm{F}}} \leqslant \frac{\rho\varepsilon_M(\|A_{11}\|_{\mathrm{F}} + \|A_{22}\|_{\mathrm{F}})}{\mathrm{sep}(A_{11}, A_{22})}, \tag{10}$$

where $E = X - \bar{X}$, ρ is a small constant of order $O(1)$, ε_M is the machine precision, and $\mathrm{sep}(A_{11}, A_{22}) = \sigma_{\min}(K)$ is called the separation of the matrices A_{11} and A_{22}.

In the following error analysis of the algorithm, we will see that the numerical stability is essentially governed by the residual $Y \equiv A_{12} - A_{11}\bar{X} + \bar{X}A_{22} = -A_{11}E + EA_{22}$. Applying standard error analysis of Gaussian elimination [9], we have

$$\|Y\|_{\mathrm{F}} = \|A_{12} - A_{11}\bar{X} + \bar{X}A_{22}\|_{\mathrm{F}} \leqslant \rho\varepsilon_M(\|A_{11}\|_{\mathrm{F}} + \|A_{22}\|_{\mathrm{F}})\|X\|_{\mathrm{F}}. \tag{11}$$

Note that the bound does not involve sep(A_{11}, A_{22}).

Next we form the QR factorization of the matrix $(-\overline{X}^{\mathrm{T}}, \gamma I)^{\mathrm{T}}$ by Householder elementary reflectors, so that

$$\begin{bmatrix} -\overline{X} \\ \gamma I \end{bmatrix} = \overline{Q} \begin{bmatrix} \overline{R} \\ 0 \end{bmatrix}, \tag{12}$$

where $\overline{Q} = Q + \delta Q$, $\|\delta Q\| \approx \varepsilon_M$, $Q^{\mathrm{T}}Q = I$. In other words, the computed matrix $\overline{Q}$ is orthogonal to machine precision [20].

In the next section, we will show that in some pathological cases, the norm of the (2,1) (block) entry of $\overline{Q}^{\mathrm{T}}A\overline{Q}$ may be larger than $O(\varepsilon_M \|A\|)$, i.e., it may be backward unstable if we are forced to treat $\overline{Q}^{\mathrm{T}}A\overline{Q}$ as block upper triangular by setting the (2,1) entry to zero. Therefore we propose to perform adjacent blocks swapping tentatively: if the norm of the (2,1) (block) entry of $\overline{Q}^{\mathrm{T}}A\overline{Q}$ is less than or equal to $O(\varepsilon_M \|A\|)$, we swap the blocks; otherwise we return without performing the swap. This gives an absolute guarantee of backward stability. We can fail to swap only if the eigenvalues A_{11} and A_{22} are so close that a small perturbation of the matrix could make them identical. If $p = q = 1$, then swapping will always succeed.

If the two blocks are exchanged, then an orthogonal similarity transformation is performed on the 2×2 blocks (if any exist) to return them to standard form.

Finally, since the nonsymmetric eigenvalue problem is an ill-conditioned problem, a small perturbation to a 2×2 block (complex conjugate eigenpair) could cause a large perturbation of its eigenvalues. In the extreme case, a 2×2 block could split into two 1×1 blocks if its complex conjugate eigenvalues become real. Carefully designed standardization steps will detect and report such phenomena. All above considerations are summed up in the following algorithm.

DIRECT SWAPPING ALGORITHM SLAEXC.

1. Copy A to T:

$$T = \begin{bmatrix} T_{11} & T_{12} \\ 0 & T_{22} \end{bmatrix} \leftarrow A = \begin{bmatrix} A_{11} & A_{12} \\ 0 & A_{22} \end{bmatrix}.$$

2. Use Gaussian elimination with complete pivoting to solve

$$T_{11}X - XT_{22} = \gamma T_{12},$$

where γ is a scaling factor to prevent overflow. If there is a small diagonal element during Gaussian elimination, set it to roughly machine precision (times the norm of the matrix).

3. Compute the QR factorization $G = (-X^{\mathrm{T}}, \gamma I)^{\mathrm{T}} = QR$ by Householder transformations.
4. Perform swapping tentatively: if the norm of the (2,1) (block) entry of $Q^{\mathrm{T}}TQ$ is of order less than $\varepsilon_M \|T\|_{\mathrm{M}}$, go to the next step, and otherwise exit.
5. If the swap is accepted, replaced A by $Q^{\mathrm{T}}AQ$ and set the (2,1) (block) entry of $Q^{\mathrm{T}}AQ$ to zero.
6. Standardize 2×2 diagonal block(s) if any exist.

In our implementation of SLAEXC in LAPACK, we have chosen $10\varepsilon_M \|A\|_{\mathrm{M}}$ as the stability criterion in step 4, where $\|A\|_{\mathrm{M}} = \max_{i,j}|a_{ij}|$. Finally, we note that we also provide a subroutine STREXC in LAPACK which calls SLAEXC to reorder all the eigenvalues into a user selected order. In particular, the user may select any subset of the spectrum which will be reordered to appear at the top left of the matrix using the fewest possible calls to SLAEXC.

3. ERROR ANALYSIS

In this section, we give an error analysis of the direct swapping algorithm SLAEXC described in the last section. We assume that $p = q = 2$, i.e., we only consider swapping two 2×2 blocks, the hardest case of the problem. In addition, for the sake of exposition, we also assume that the computation of QR factorization and the similarity transformation $Q^{\mathrm{T}}AQ$ are exact, and the scaling factor $\gamma = 1$. Including these rounding errors does not change the conclusion of the analysis, but makes the exposition appear more complicated.

Let $\overline{X}$ be the computed solution of the Sylvester equation, where $\overline{X} = X + E$, X is the exact solution, and E is an error matrix. By the argument of (12) and a result of Stewart [17] on the perturbation of the QR factorization, we know that under mild conditions (such as $\|G^{\dagger}\|_2\|E\|_{\mathrm{F}} < 1$), the QR factorization of $(-\overline{X}^{\mathrm{T}}, I)^{\mathrm{T}}$ can be written as

$$\begin{bmatrix} -X \\ I \end{bmatrix} + \begin{bmatrix} -E \\ 0 \end{bmatrix} = G + \begin{bmatrix} -E \\ 0 \end{bmatrix} = \hat{Q}\hat{R} = (Q + W)\begin{bmatrix} R + F \\ 0 \end{bmatrix}, \quad (13)$$

where W and F are the perturbations of the orthogonal matrix Q and the triangular matrix R, respectively, and $\hat{Q} = Q + W$ is orthogonal. $\|W\|_{\mathrm{F}}$ and

$\|F\|_F$ are essentially bounded by the terms of order $\|G^{\dagger}\|_2\|E\|_F$. From $(Q + W)^T(Q + W) = I$, up to the first order we have $Q^TW = -W^TQ$. When $\hat{Q} = Q + W$ transforms A, ignoring the second order perturbations, we have

$$\begin{aligned}\hat{Q}^TA\hat{Q} &= (Q + W)^TA(Q + W)\\ &= Q^TAQ + W^TAQ + Q^TAW + W^TAW\\ &= \tilde{A} + W^TQ \cdot Q^TAQ + Q^TAQ \cdot Q^TW\\ &= \tilde{A} + \tilde{A}Q^TW - Q^TW\tilde{A}.\end{aligned}$$

Defining $Z = Q^TW$ and partitioning it conformally with $\tilde{A}$ in the form

$$Z = \begin{bmatrix} Z_{11} & Z_{12} \\ Z_{21} & Z_{22} \end{bmatrix},$$

we have

$$\hat{Q}^TA\hat{Q} = \begin{bmatrix} \tilde{A}_{22} & \tilde{A}_{12} \\ 0 & \tilde{A}_{22} \end{bmatrix} + \begin{bmatrix} E_{22} & E_{12} \\ E_{21} & E_{11} \end{bmatrix}, \tag{14}$$

where

$$\begin{aligned}E_{11} &= \tilde{A}_{11}Z_{22} - Z_{22}\tilde{A}_{11} - Z_{21}\tilde{A}_{12},\\ E_{22} &= \tilde{A}_{22}Z_{11} - Z_{11}\tilde{A}_{22} + \tilde{A}_{12}Z_{21},\\ E_{21} &= \tilde{A}_{11}Z_{21} - Z_{21}\tilde{A}_{22}.\end{aligned}$$

E_{11} and E_{22} perturb the eigenvalues directly and do not affect stability. E_{21} is of interest because it measures the numerical stability of swapping. E_{12} is the error in the block $\tilde{A}_{12}$. It is not of interest, since it neither affects the numerical stability of the algorithm nor perturbs the eigenvalues. The task is to give bounds on the norms of E_{11}, E_{22}, and E_{21}. To do so, let us first express Z_{ij} in terms of the blocks Q_{ij} of Q, E, F, and R. From (13), we have

$$\begin{aligned}(I + Q^{\mathrm{T}}W)\begin{bmatrix} R + F \\ 0 \end{bmatrix} &= Q^{\mathrm{T}}\begin{bmatrix} -X \\ I \end{bmatrix} + Q^{\mathrm{T}}\begin{bmatrix} -E \\ 0 \end{bmatrix} \\ &= \begin{bmatrix} R \\ 0 \end{bmatrix} + \begin{bmatrix} -Q_{11}^{\mathrm{T}}E \\ -Q_{12}^{\mathrm{T}}E \end{bmatrix}.\end{aligned}$$

Postmultiplying by $(R + F)^{-1}$ on both sides of the above equation, and noting that $Z = Q^{\mathrm{T}}W$, we get

$$(I + Z)\begin{bmatrix} I \\ 0 \end{bmatrix} = \begin{bmatrix} R - Q_{11}^{\mathrm{T}}E \\ -Q_{12}^{\mathrm{T}}E \end{bmatrix}(R + F)^{-1},$$

so that

$$\begin{aligned} Z_{11} &= -I + (R - Q_{11}^{\mathrm{T}}E)(R + F)^{-1}, \\ Z_{21} &= -Q_{12}^{\mathrm{T}}E(R + F)^{-1}, \end{aligned}$$

and up to the first order perturbations, we have

$$Z_{11} = -Q_{11}^{\mathrm{T}}ER^{-1} - FR^{-1}, \tag{15}$$

$$Z_{21} = -Q_{12}^{\mathrm{T}}ER^{-1}. \tag{16}$$

To express Z_{22}, again from (13),

$$\begin{aligned}(I + QW^{\mathrm{T}})\begin{bmatrix} -X - E \\ I \end{bmatrix} &= Q\begin{bmatrix} R \\ 0 \end{bmatrix} + Q\begin{bmatrix} F \\ 0 \end{bmatrix} \\ &= \begin{bmatrix} -X \\ I \end{bmatrix} + Q\begin{bmatrix} F \\ 0 \end{bmatrix}.\end{aligned}$$

By canceling $(-X^{\mathrm{T}}, I)^{\mathrm{T}}$ from both sides of the equation and premultiplying by Q^{T}, we obtain

$$W^{\mathrm{T}}\begin{bmatrix} -X - E \\ I \end{bmatrix} = \begin{bmatrix} F \\ 0 \end{bmatrix} + Q^{\mathrm{T}}\begin{bmatrix} E \\ 0 \end{bmatrix}.$$

By inserting $Q^{\mathrm{T}}Q = I$ in the left side of the above equation and noting that $W^{\mathrm{T}}Q = -Q^{\mathrm{T}}W = -Z$, we have

$$Z\begin{bmatrix} R \\ 0 \end{bmatrix} - Z\begin{bmatrix} Q_{11}^{\mathrm{T}}E \\ Q_{12}^{\mathrm{T}}E \end{bmatrix} = -\begin{bmatrix} F \\ 0 \end{bmatrix} - \begin{bmatrix} Q_{11}^{\mathrm{T}}E \\ Q_{12}^{\mathrm{T}}E \end{bmatrix}.$$

Thus the "bottom" equation is

$$Z_{21}R - Z_{21}Q_{11}^{\mathrm{T}}E - Z_{22}Q_{12}^{\mathrm{T}}E = -Q_{12}^{\mathrm{T}}E.$$

By (16) and assuming that error matrix E is nonsingular, we get

$$Z_{22} = -Z_{21}Q_{11}^{\mathrm{T}}Q_{12}^{-\mathrm{T}} = Q_{12}^{\mathrm{T}}ER^{-1}Q_{11}^{\mathrm{T}}Q_{12}^{-\mathrm{T}}. \tag{17}$$

From the expressions (15), (16), and (17) for Z_{11}, Z_{12}, and Z_{22}, the expressions for E_{11}, E_{22}, and E_{21} are recast as

$$\begin{aligned} E_{11} &= Q_{11}^{\mathrm{T}}A_{11}Q_{12}^{-\mathrm{T}}Q_{12}^{\mathrm{T}}ER^{-1}Q_{11}^{\mathrm{T}}Q_{12}^{-\mathrm{T}} - Q_{12}^{\mathrm{T}}ER^{-1}Q_{11}^{\mathrm{T}}Q_{12}^{-\mathrm{T}}Q_{12}^{\mathrm{T}}A_{11}Q_{12}^{-\mathrm{T}} \\ &\quad + Q_{12}^{\mathrm{T}}ER^{-1}\left(-RA_{22}R^{-1}Q_{11}^{\mathrm{T}}Q_{12}^{-\mathrm{T}} + Q_{11}^{\mathrm{T}}A_{11}Q_{12}^{-\mathrm{T}}\right) \\ &= Q_{12}^{\mathrm{T}}(A_{11}E - EA_{22})R^{-1}Q_{11}^{\mathrm{T}}Q_{12}^{-\mathrm{T}} \\ &= -Q_{12}^{\mathrm{T}}YR^{-1}Q_{11}^{\mathrm{T}}Q_{12}^{-\mathrm{T}}, \\ E_{22} &= -RA_{22}R^{-1}\left(Q_{11}^{\mathrm{T}}ER^{-1} + FR^{-1}\right) + \left(Q_{11}^{\mathrm{T}}ER^{-1} + FR^{-1}\right)RA_{22}R^{-1} \\ &\quad -\left(-RA_{22}R^{-1}Q_{11}^{\mathrm{T}}Q_{12}^{-\mathrm{T}} + Q_{11}^{\mathrm{T}}A_{11}Q_{12}^{-\mathrm{T}}\right)Q_{12}^{\mathrm{T}}ER^{-1} \\ &= Q_{11}^{\mathrm{T}}(-A_{11}E + EA_{22})R^{-1} - \tilde{A}_{22}FR^{-1} + FR^{-1}\tilde{A}_{22} \\ &= Q_{11}^{\mathrm{T}}YR^{-1} - \tilde{A}_{22}FR^{-1} + FR^{-1}\tilde{A}_{22}, \end{aligned}$$

and

$$\begin{aligned} E_{21} &= -Q_{12}^{\mathrm{T}}A_{11}Q_{12}^{-\mathrm{T}}Q_{12}^{\mathrm{T}}ER^{-1} + Q_{12}^{\mathrm{T}}ER^{-1}RA_{22}R^{-1} \\ &= -Q_{12}^{\mathrm{T}}(-A_{11}E + EA_{22})R^{-1} \\ &= Q_{12}^{\mathrm{T}}YR^{-1}. \end{aligned}$$

We see that E_{11}, E_{22} and E_{21} are essentially related to the residual vector Y of the Sylvester equation solver, R, and the subblocks Q_{11} and Q_{12} of Q. Furthermore, rewriting (7) as

$$\begin{bmatrix} -X \\ I \end{bmatrix} = \begin{bmatrix} Q_{11} & Q_{12} \\ Q_{21} & Q_{22} \end{bmatrix} \begin{bmatrix} R \\ 0 \end{bmatrix},$$

we see that

$$Q_{21} = R^{-1}$$

and

$$R^{\mathrm{T}}R = I + X^{\mathrm{T}}X.$$

Let $\sigma(C)$ denote the set of singular values of matrix C, and $\lambda(C)$ denote the set of eigenvalues of matrix C. Then

$$\sigma^2(R) = \lambda(R^{\mathrm{T}}R) = \lambda(I + X^{\mathrm{T}}X) = 1 + \lambda(X^{\mathrm{T}}X) = 1 + \sigma^2(X).$$

Therefore

$$\|Q_{21}\|_2 = \|R^{-1}\|_2 = \frac{1}{\sigma_2(R)} = \frac{1}{\left[1 + \sigma_2^2(X)\right]^{1/2}}, \tag{18}$$

where $\sigma_1(X) \geqslant \sigma_2(X) \geqslant 0$. Now to estimate the norm of the blocks Q_{ij} of Q, we use the following CS decomposition of a partitioned orthogonal matrix, which was introduced by Stewart [16]. A proof of the existence of the decomposition can be found in [18].

CS DECOMPOSITION. *Let the orthogonal matrix $Q \in \mathbb{R}^{2k\times 2k}$ be partitioned in the form*

$$Q = \begin{array}{c c} & \begin{array}{cc} k & k \end{array} \\ \begin{array}{c} k \\ k \end{array} & \begin{pmatrix} Q_{11} & Q_{12} \\ Q_{21} & Q_{22} \end{pmatrix} \end{array}.$$

The there are orthogonal matrices $U = \mathrm{diag}(U_1, U_2)$ *and* $V = \mathrm{diag}(V_1, V_2)$ *with* $U_1, V_1 \in \mathbb{R}^{k\times k}$ *such that*

$$U^{\mathrm{T}}QV = \begin{array}{c c} & \begin{array}{cc} k & k \end{array} \\ \begin{array}{c} k \\ k \end{array} & \begin{pmatrix} C & S \\ -S & C \end{pmatrix} \end{array},$$

where

$$C = \operatorname{diag}(c_1, c_2, \dots, c_k) \geqslant 0, \qquad S = \operatorname{diag}(s_1, s_2, \dots, s_k) \geqslant 0,$$
$$C^2 + S^2 = I.$$

By the CS decomposition of Q and (18), we have

$$\|Q_{11}\|_2 = \frac{\sigma_1(X)}{\left[1 + \sigma_1^2(X)\right]^{1/2}}$$

and

$$\|Q_{12}\|_2 = \|Q_{21}\|_2, \qquad \|Q_{22}\|_2 = \|Q_{11}\|_2.$$

Thus, for E_{11}, we have

$$\|E_{11}\|_2 \leqslant \|Q_{12}^{\mathrm{T}}\|_2 \|Y\|_{\mathrm{F}} \|R^{-1}\|_2 \|Q_{11}^{\mathrm{T}}\|_2 \|Q_{12}^{-\mathrm{T}}\|_2 = \frac{\sigma_1(X)}{1 + \sigma_2^2(X)} \|Y\|_{\mathrm{F}}.$$

Similarly, for E_{22}, from [17], we have $\|FR^{-1}\|_{\mathrm{F}} \leqslant 2\|G^{\dagger}\|_2 \|E\|_{\mathrm{F}}$; therefore

$$\begin{aligned} \|E_{22}\|_2 &\leqslant \|Q_{11}^{\mathrm{T}}\|_2 \|Y\|_{\mathrm{F}} \|R^{-1}\|_2 + 2\|\tilde{A}_{22}\|_2 \|FR^{-1}\|_{\mathrm{F}} \\ &\leqslant \frac{\sigma_1(X)}{1 + \sigma_2^2(X)} \|Y\|_{\mathrm{F}} + 4\|\tilde{A}_{22}\|_2 \|G^{\dagger}\|_2 \|E\|_{\mathrm{F}}. \end{aligned}$$

Finally, for E_{21}, we have

$$\|E_{21}\|_2 \leqslant \|Q_{12}^{\mathrm{T}}\|_2 \|Y\|_{\mathrm{F}} \|R^{-1}\|_2 = \frac{1}{1 + \sigma_2^2(X)} \|Y\|_{\mathrm{F}}.$$

Hence we have the following theorem.

THEOREM 2. *Let* $Y = A_{12} - A_{11}\bar{X} + \bar{X}A_{22}$, *where* $\bar{X} = X + E$ *is the computed solution of the Sylvester equation* (6), *assume that the error matrix* E *is nonsingular, and let the QR factorization of* $(-\bar{X}^{\mathrm{T}}, I)^{\mathrm{T}}$ *satisfy*

$$\begin{bmatrix} -\bar{X} \\ I \end{bmatrix} = \hat{Q} \begin{bmatrix} \hat{R} \\ 0 \end{bmatrix}.$$

Then

$$\hat{Q}^{T} A \hat{Q} = \begin{bmatrix} \tilde{A}_{22} & \tilde{A}_{12} \\ 0 & \tilde{A}_{11} \end{bmatrix} + \begin{bmatrix} E_{22} & E_{12} \\ E_{21} & E_{11} \end{bmatrix},$$

where $\tilde{A}_{ii}$ *is similar to* A_{ii}, $i = 1, 2$, *and up to the first order perturbation* $O(\|E\|_2)$,

$$\|E_{11}\|_2 \leqslant \frac{\sigma_1(X)}{1+\sigma_2^2(X)}\|Y\|_F, \tag{19}$$

$$\|E_{22}\|_2 \leqslant \frac{\sigma_1(X)}{1+\sigma_2^2(X)}\|Y\|_F + 4\|\tilde{A}_{22}\|_2\|G^{\dagger}\|_2\|E\|_F \tag{20}$$

$$\|E_{21}\|_2 \leqslant \frac{1}{1+\sigma_2^2(X)}\|Y\|_F. \tag{21}$$

Three remarks are in order:

REMARK 1. From the theorem, we see that the departure $\|E_{21}\|_2$ from upper block triangular form (the measure of numerical instability) is bounded by $\|Y\|_F/[1+\sigma_2^2(X)]$. It is easy to see that

$$\|X\|_F \leqslant \frac{\|A_{12}\|_F}{\operatorname{sep}(A_{11}, A_{22})}, \tag{22}$$

where the equality is attained when $\operatorname{col}(A_{12})$ is a left singular vector of K corresponding to the smallest singular value $\sigma_{min}(K) = \operatorname{sep}(A_{11}, A_{22})$. Combining (22), (11), and (21), we have

$$\|E_{21}\|_2 \leqslant \frac{\rho\varepsilon_M(\|A_{11}\|_F + \|A_{22}\|_F)\|A_{12}\|_F}{[1+\sigma_2^2(X)]\operatorname{sep}(A_{11}, A_{22})}.$$

Logically, the above bound indicates that the numerical instability will occur if we have small $\operatorname{sep}(A_{11}, A_{22})$. But in practice, numerical experiments show that this upper bound is very pessimistic. Small $\operatorname{sep}(A_{11}, A_{22})$ does not imply instability. We will discuss this further in the following section.

REMARK 2. Iterative refinement applied to the Sylvester equation will improve the accuracy of computed $\bar{X}$ (unless the Sylvester equation is too close to singular), but it need not improve $\|Y\|_F$, at least when Gaussian elimination with complete pivoting is used to solve the Sylvester equation.

REMARK 3. The factor $\sigma_1(X)/[1 + \sigma_2^2(X)]$ that affects $\|E_{11}\|_2$ and $\|E_{22}\|_2$ is interesting, since it warns that large and ill-conditioned X may endanger accuracy, because of (11) and

$$\frac{\sigma_1(X)}{1 + \sigma_2^2(X)} = \frac{\kappa(X)}{\sigma_2(X) + \sigma_2^{-1}(X)},$$

where $\kappa(X) = \sigma_1(X)/\sigma_2(X)$. How $\kappa(X)$, $\operatorname{sep}(A_{11}, A_{22})$, and the accuracy of the swapped eigenvalues are related in practice needs further investigation.

4. SOFTWARE DEVELOPMENT AND NUMERICAL EXPERIMENTS

In this section, we first discuss the development of software for the swapping algorithm SLAEXC. Then we discuss numerical experiments to show the capability of our software to deal with ill-conditioned cases, compare with Stewart's swapping algorithm EXCHNG, and finally demonstrate the sharpness of our perturbation bounds.

4.1. *Software Development*

A set of FORTRAN subroutines has been developed to implement the direct swapping algorithm described in Section 3. It is part of the LAPACK project [1]. As with other LAPACK routines, this algorithm was designed for accuracy, robustness and portability.

The main subroutine is called STREXC. STREXC moves a given 1 × 1 or 2 × 2 diagonal block of a real quasitriangular matrix to a user specified position. On return, parameter INFO reports whether the given block has moved to the desired position, or whether there are blocks too close to swap, and what is the current position of the given block. The subroutine STREXC is supported by subroutine SLAEXC, which exchanges adjacent blocks. The subroutine SLAEXC is an implementation of the algorithm SLAEXC described in Section 3, where the subproblem of solving the Sylvester equation (8) by Gaussian elimination with complete pivoting is implemented in subroutine SLASY2, and the subproblem of standardizing a 2 × 2 block is implemented in subroutine SLANV2.

In the interest of simplicity, we also used some other subroutines from LAPACK and the BLAS to perform some basic linear algebra operations, such as generating Householder transformations, computing the 2-norm of a vector, and so on.

Finally, a test subroutine has been written to automatically test the subroutine SLAEXC. There are nested loops over different block sizes, different numerical scales, and different conditionings of the problem.

4.2. *Numerical Experiments*

Backward Stability Test. To measure the backward stability of a swapping algorithm, we need to test (I) how close the matrix $\bar{Q}$ is to an orthogonal matrix, and (II) how close $\bar{Q}\check{A}\bar{Q}^{\mathrm{T}}$ is to the original matrix A, where $\check{A}$, is the computed $\tilde{A}$. In other words, we need to test whether the two quantities

$$E_Q = \frac{\|I - \bar{Q}^{\mathrm{T}}\bar{Q}\|_1}{\varepsilon_M}, \qquad E_A = \frac{\|A - \bar{Q}\check{A}\bar{Q}^{\mathrm{T}}\|_1}{\varepsilon_M \|A\|_1}$$

are around 1, where ε_M is the machine precision. To check the changes among eigenvalues is not required to judge the correctness of an algorithm, since we know that there must be at least an $O(\varepsilon_M \|A\|)$ perturbation to the original matrix after swapping, and the nonsymmetric eigenvalue problem is potentially ill conditioned. However, for reasonably conditioned matrices, the changes in the eigenvalues do measure the accuracy of a swapping algorithm. For this reason, in the following numerical examples, we also compare the eigenvalues before and after swapping, besides checking the quantities E_Q and E_A.

All numerical experiments were carried out on a Sun Sparcstation 1 + . The arithmetic is IEEE standard single precision, with machine precision $\varepsilon_M = 2^{-23} \approx 1.192 \times 10^{-7}$.

We have done extensive testing on matrices with various mixtures of the block sizes, scales, and closeness among eigenvalues. More specifically, we show algorithm SLAEXC on the following four types of matrices:

Test Matrix 1: Good separation of A_{11} and A_{22}. The eigenvalues before swapping are

$$\lambda_1 = 0.2000000E + 01 \pm i0.2085666E + 02,$$

$$\lambda_2 = 0.1000000E + 01 \pm i0.2017424E + 02.$$

Test Matrix 2: Moderate separation of A_{11} and A_{22}. The eigenvalues before swapping are

$$\lambda_1 = 0.1000000E + 01 \pm i0.1732051E + 01,$$

$$\lambda_2 = 0.1001000E + 01 \pm i0.1732916E + 01.$$

Test Matrix 3: Close eigenvalues. The corresponding the Sylvester equation is very ill conditioned; the eigenvalues before swapping are

$$\lambda_1 = 0.1000000E + 01 \pm i0.1000000E + 01,$$

$$\lambda_2 = 0.1001000E + 01 \pm i0.1000000E + 01.$$

Test Matrix 4: The extreme case, where the eigenvalues of A_{11} and A_{22} are the same, and theoretically, the Sylvester equation solution is infinite. This matrix is used to test the robustness of our software against overflow. The eigenvalues before swapping are

$$\lambda_1 = 0.1000000E + 01 \pm i0.1732051E + 01,$$

$$\lambda_2 = 0.1000000E + 01 \pm i0.1732051E + 01,$$

Table 1 summarizes the results of algorithm SLAEXC, where $\operatorname{sep}(A_{11}, A_{22})$ is computed by MATLAB, and included here for the sake of theoretical analysis. From Table 1, we see that both the backward stability and the accuracy of algorithm SLAEXC are satisfactory.

Comparison with Stewart's Algorithm EXCHNG. We have done numerical comparisons between the direct swapping algorithm SLAEXC and Stewart's swapping algorithm EXCHNG [15], which uses QR iteration. Both algorithms perform well in most cases, but in certain cases, EXCHNG is inferior to SLAEXC. For example, let

$$A(\tau) = \begin{bmatrix} 7.001 & -87 & 39.4\tau & 22.2\tau \\ 5 & 7.001 & -12.2\tau & 36.0\tau \\ 0 & 0 & 7.01 & -11.7567 \\ 0 & 0 & 37 & 7.01 \end{bmatrix},$$

TABLE 1

NUMERICAL TESTS OF ALGORITHM SLAEXC

Test	Matrix	sep(A_{11}, A_{22})	E_Q	E_A	Eigenvalues after swapping
1	$\begin{pmatrix} 2 & -87 & -20000 & 10000 \\ 5 & 2 & -20000 & -10000 \\ 0 & 0 & 1 & -11 \\ 0 & 0 & 37 & 1 \end{pmatrix}$	3.337×10^{-1}	0.260	0.197	0.1000001E + 01 ± i0.2017424E + 02 0.2000000E + 01 ± i0.2085665E + 02
2	$\begin{pmatrix} 1 & -3 & 3576 & 4888 \\ 1 & 1 & -88 & -1440 \\ 0 & 0 & 1.001 & -3 \\ 0 & 0 & 1.001 & 1.001 \end{pmatrix}$	8.442×10^{-4}	0.625	0.423	0.1001000E + 01 ± i0.1732917E + 01 0.1000000E + 01 ± i0.1732051E + 01
3	$\begin{pmatrix} 1 & -100 & 400 & -1000 \\ 0.01 & 1 & 1200 & -10 \\ 0 & 0 & 1.001 & -0.01 \\ 0 & 0 & 100 & 1.001 \end{pmatrix}$	2.000×10^{-7}	0.417	0.001	0.1000996E + 01 ± i0.1000360E + 01 0.1000003E + 01 ± i0.9995396E + 00
4	$\begin{pmatrix} 1 & -3 & 3 & 2 \\ 1 & 1 & 9 & 0 \\ 0 & 0 & 1 & -3 \\ 0 & 0 & 1 & 1 \end{pmatrix}$	∞	0.687	0.241	0.9999987E + 00 ± i0.1732051E + 01 0.1000002E + 01 ± i0.1732051E + 01

where τ is a parameter. The matrix $A(\tau)$ has the same eigenvalues for all τ:

$$\lambda_1 = 0.7001000E + 01 \pm i0.2085666E + 02,$$

$$\lambda_2 = 0.7010000E + 01 \pm i0.2085660E + 02,$$

and $\operatorname{sep}(A_{11}, A_{22}) = 0.0024$. When $\tau = 1$, the output matrix of algorithm SLAEXC is

$$\check{A} = \begin{pmatrix} 0.70100012E + 01 & -0.86993660E + 02 & -0.39390938E + 02 & -0.22241005E + 02 \\ 0.50003409E + 01 & 0.70100012E + 01 & 0.12191071E + 02 & -0.35999401E + 02 \\ 0.00000000E + 00 & 0.00000000E + 00 & 0.70009995E + 01 & -0.11755549E + 02 \\ 0.00000000E + 00 & 0.00000000E + 00 & 0.37003792E + 02 & 0.70009995E + 01 \end{pmatrix}.$$

The eigenvalues after swapping are

$$\tilde{\lambda}_2 = 0.7010001E + 01 \pm i0.2085661E + 02,$$

$$\tilde{\lambda}_1 = 0.7000999E + 01 \pm i0.2085665E + 02,$$

which are accurate to machine precision. However, the output of algorithm EXCHNG after eight QR iterations is[1] E

$$\check{A} = \begin{pmatrix} 0.28140299E + 02 & -0.81122643E + 02 & -0.39849255E + 02 & -0.15834051E + 02 \\ 0.10856283E + 02 & -0.14087547E + 02 & -0.23942078E + 02 & 0.32877380E + 02 \\ 0.00000000E + 00 & 0.00000000E + 00 & 0.19211971E + 02 & 0.21227583E + 02 \\ 0.00000000E + 00 & 0.00000000E + 00 & -0.27540298E + 02 & -0.52427406E + 01 \end{pmatrix},$$

which has eigenvalues

$$\tilde{\lambda}_2 = 0.7026377E + 01 \pm i0.2085408E + 02,$$

$$\tilde{\lambda}_1 = 0.6984615E + 01 \pm i0.2085919E + 02$$

They only have two decimal digits correct.

Table 2 shows the numerical results with different choices of parameter τ; when $\tau = 10$, it takes 17 QR iterations to converge. It clearly shows the superiority of algorithm SLAEXC. In particular, we note that algorithm EXCHNG is nonconvergent when $\tau = 100$. It means that the eigenvalues are not able to

[1] Where the stopping criterion used in QR iteration is eps $= 1.2 \times 10^{-7}$.

TABLE 2
COMPARISON OF ALGORITHMS SLAEXC and EXCHNG

τ	SLAEXC	EXCHNG
1	$\tilde{\lambda}_2 = 0.7010001\text{E} + 01 \pm i0.2085661\text{E} + 02$ $\tilde{\lambda}_1 = 0.7000999\text{E} + 01 \pm i0.2085665\text{E} + 02$	$\tilde{\lambda}_2 = 0.7026377\text{E} + 01 \pm i0.2085408\text{E} + 02$ $\tilde{\lambda}_1 = 0.6984615\text{E} + 01 \pm i0.2085919\text{E} + 02$
10	$\tilde{\lambda}_2 = 0.7010000\text{E} + 01 \pm i0.2085660\text{E} + 02$ $\tilde{\lambda}_1 = 0.7000999\text{E} + 01 \pm i0.2085665\text{E} + 02$	$\tilde{\lambda}_2 = 0.7063053\text{E} + 01 \pm i0.2086175\text{E} + 02$ $\tilde{\lambda}_1 = 0.6947970\text{E} + 01 \pm i0.2085144\text{E} + 02$
100	$\tilde{\lambda}_2 = 0.7009999\text{E} + 01 \pm i0.2085660\text{E} + 02$ $\tilde{\lambda}_1 = 0.7000999\text{E} + 01 \pm i0.2085665\text{E} + 02$	Not convergent after 30 QR steps

be exchanged by algorithm EXCHNG. But algorithm SLAEXC has no difficulty. This convergence difficulty may reflect recent work of Batterson [3], who has discovered classes of nonsymmetric matrices where QR iteration fails to converge, or converges quite slowly.

On the Upper Bound of $\|E_{21}\|_2$. Finally, in the interest of theoretical analysis, we discuss the sharpness of the bound on $\|E_{21}\|_2$, which controls the numerical stability of algorithm SLAEXC. In most of the test examples, we see that the bound (21) of $\|E_{21}\|_2$ is very pessimistic. However, we do find some examples indicating that the bound in (21) can roughly be attained. Let us consider the following example:[2]

$$A = \begin{array}{c c} & \begin{array}{cc} 2 & 2 \end{array} \\ \begin{array}{c} 2 \\ 2 \end{array} & \begin{pmatrix} A_{11} & A_{12} \\ 0 & A_{22} \end{pmatrix} \end{array}$$

$$= \begin{pmatrix} 1.0000E + 00 & -1.0000E + 02 & 1.9900E + 04 & 1.0201E + 02 \\ 1.0000E - 02 & 1.0000E + 00 & 1.0000E + 02 & -1.9800E + 00 \\ 0 & 0 & 1.0100E + 00 & -1.0000E - 02 \\ 0 & 0 & 1.0000E + 02 & 1.0100E + 00 \end{pmatrix},$$

where $\operatorname{sep}(A_{11}, A_{22}) = 2 \times 10^{-6}$. The A_{12} block of A is designed so that

$$X = \begin{pmatrix} 1.0000E + 00 & -2.0000E + 02 \\ 1.0000E + 00 & -1.0000E + 00 \end{pmatrix}$$

is the solution of the Sylvester equation. Note that $\sigma_1(X) = 200.01$, $\sigma_2(X)$

[2] For brevity, only five digits are displayed for all the data in this section, though we did run in double precision.

= 0.99498. We used MATLAB to compute the different quantities in the bound (where machine precision is doubled: $\varepsilon_M = 2.2204 \times 10^{-16}$). First the norm of the residual matrix Y for the computed solution $\bar{X}$ of the Sylvester equation is

$$\|Y\|_F = \|A_{12} - A_{11}\bar{X} + \bar{X}A_{22}\|_F = 4.0272 \times 10^{-12},$$

which almost reaches the estimated bound (11) of Y:

$$\varepsilon_M(\|A_{11}\|_F + \|A_{22}\|_F)\|X\|_F = 8.8830 \times {}^{-12}.$$

Furthermore, the observed norm of (2,1) block $\check{A}_{21}$ after swapping is

$$\|\check{A}_{21}\|_2 = 1.2973 \times 10^{-12},$$

which also roughly attains the bound (21) for $\|E_{21}\|_2$;

$$\|E_{21}\|_2 \leqslant \frac{1}{1 + \sigma_2^2(X)}\|Y\|_F = 2.0237 \times 10^{-12}.$$

Note that for this example, the algorithm is still backward stable, since

$$\|\check{A}_{21}\|_2 = 1.2973 \times 10^{-12} \leqslant \varepsilon_M\|A\|_F = 4.4189 \times 10^{-12}.$$

After setting $\check{A}_{21} = 0$, then the measures of backward stability are $E_Q = 2.3$ and $E_A = 1.8$.

From Remark 1 after Theorem 2, we might worry that a huge $\|X\|_F$ or tiny $\operatorname{sep}(A_{11}, A_{22})$ could cause numerical instability. However, the following example illustrates how in practice a small separation of A_{11} and A_{22} does not necessarily lead to instability. Let

$$A_{11} = \begin{bmatrix} 1 & -10^{-6} \\ 1 & 1 \end{bmatrix}, \qquad A_{22} = A_{11} + \sqrt{\varepsilon_M}\, I.$$

The the separation of A_{11} and A_{22} is tiny; namely, $\operatorname{sep}(A_{11}, A_{22}) = 2.9802 \times 10^{-14}$. Let A_{12} be chosen such that $\operatorname{col}(A_{12})$ is left the singular vector of K corresponding to the smallest singular value $\sigma_{\min}(K)$, so that the norm of the solution X of the Sylvester equation $A_{11}X - XA_{22} = A_{12}$ reaches its upper bound (22), that is,

$$\|X\|_F = \frac{\|A_{12}\|_F}{\operatorname{sep}(A_{11}, A_{22})} = 3.3554 \times 10^{13}$$

and $\kappa(X) = 10^6$. Hence the estimated bound of the norm of the residual Y is

$$\varepsilon_M(\|A_{11}\|_F + \|A_{22}\|_F)\|X\|_F = 2.5810 \times 10^{-2}.$$

However in practice, the observed residual norm $\|Y\|_F = 3.7253 \times 10^{-9}$. After swapping, it turns out that

$$\|\check{A}_{21}\|_F = 7.3985 \times 10^{-24} \ll \varepsilon_M\|A\|_F = 5.8747 \times 10^{-16}.$$

So the swapping is perfectly stable.

5. CONCLUSIONS

In this paper, we have developed a direct swapping algorithm which reorders the eigenvalues on the diagonal of a matrix in real Schur form by performing an orthogonal similarity transformation. A complete set of FORTRAN subroutines has been developed and included in the LAPACK library [1]. The algorithm is guaranteed to be numerically stable because we explicitly test for instability and do not reorder the eigenvalues if it happens; it can only happen if the eigenvalues are so close as to be numerically indistinguishable. Unfortunately, there is no proof of the backward stability of the algorithm without this explicit test, even though we have not seen an example where instability occurred. The detailed error analysis and numerical examples show how well it deals with ill-conditioned cases, whereas the alternative stable algorithm EXCHNG may occasionally fail to converge.

The authors would like to thank K. C. Ng and B. Parlett for sharing their programs during our initial work on the subject. The valuable comments of G. W. Stewart and B. Parlett during the development of software are gratefully acknowledged. The authors are also indebted to A. Edelman and N. Higham for their valuable comments on the subject.

REFERENCES

1 E. Anderson, Z. Bai, C. Bischof, J. Demmel, J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammarling, A. Mckenney, S. Ostrouchov, and D. Sorensen, *LAPACK Users' Guide, Release 1.0*, SIAM, 1992.

2 Z. Bai, J. Demmel, and A. Mckenney, On computing condition numbers for the nonsymmetric eigenproblem, *ACM Trans. Math. Software*, to appear.

3 S. Batterson, Convergence of the QR algorithm on 3×3 normal matrices, *Numer. Math.* 58:341–352 (1990).

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

5 R. S. Bartels and G. W. Stewart, Solution of the matrix equation $AX + XB = C$, *Comm. ACM* 15:820–826 (1972).

6 Z. Cao and F. Zhang, Direct methods for ordering eigenvalues of a real matrix (in Chinese), *Chinese Univ. J. Comput. Math.* 1:27–36 (1981).

7 J. Dongarra, S. Hammarling, and J. Wilkinson, Numerical considerations in computing invariant subspaces, *SIAM J. Math. Anal. Appl.* 13:145–161 (1992).

8 F. R. Gantmacher, *Theory of Matrices*, Vol. I, Chelsea, New York, 1959.

9 G. Golub, S. Nash, and C. Van Loan, A HessenbergSchur Method for the Problem $AX + XB = C$, *IEEE Trans. Automat. Control* AC-24:909–913 (1979).

10 V. Mehrmann, A symplectic orthogonal method for single input or single output discrete time optimal control problems, in *Linear Algebra in Signals Systems and Control* (B. N. Datta et al., Eds.), SIAM, Philadelphia, 1988, pp. 128–140.

11 K. C. Ng and B. N. Parlett, Development of an accurate algorithm for EXP(Bt), Part I, Programs to swap diagonal block, Part II CPAM-294, Univ. of California, Berkeley, 1988.

12 A. Ruhe, An algorithm for numerical determination of the structure of a general matrix, *BIT* 10:196–216 (1970).

13 B. T. Smith et al., *Matrix Eigensystem Routines—EISPACK Guide*, 2nd ed., Lecture Notes in Comput. Sci. 119, Springer-Verlag, 1976.

14 G. W. Stewart, Simultaneous iteration for computing invariant subspaces of non-Hermitian matrices, *Numer. Math.* 25:12–56 (1976).

15 G. W. Stewart, Algorithm 506 HQR3 and EXCHANG: Fortran subroutine for calculating and ordering the eigenvalues of a real upper Hessenberg matrix [F2], *ACM Trans. Math. Software* 2:275–280 (1976).

16 G. W. Stewart, On the perturbation of pseudo-inverse, projections, and linear least squares problems, *SIAM Rev.*, 19:634–62 (1977).

17 G. W. Stewart, Perturbation bounds for the QR factorization of a matrix, *SIAM J. Numer. Anal.* 14:509–18 (1977).

18 G. W. Stewart and J. Sun., *Matrix Perturbation Theory*, Academic, New York, 1990.

19 P. Van Dooren, Algorithm 590, DSUBSP and EXCHQZ: Fortran subroutines for computing deflating subspaces with specified spectrum, *ACM Trans. Math. Software* 8:376–382 (1982).

20 J. H. Wilkinson, *The Algebraic Eigenvalue Problem*, Oxford U.P., 1965.

Received 30 January 1992; final manuscript accepted 14 June 1992