


After setting $\tilde{A}_{21} = 0$, then the ranges of backward stability are $E = 0.23$ and $E = 0.18$.

From rank 1 after Theorem 2, we verify that a huge $\|X\|_F$ or tiny $\text{sep}(A_{11}, A_{22})$ can 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

$$A_{11} = \begin{bmatrix} 1 & -10^{-6} \\ 1 & 1 \end{bmatrix}, \quad A_{22} = A_{11} + \sqrt{\varepsilon} M I,$$

then the separation of $A_{11}$ and $A_{22}$ is tiny, that is $\text{sep}(A_{11}, A_{22}) = 2 \times 10^{-14}$. Let $A_{12}$ be
down such that $A_{12}$ is the left 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 $A_{11} X - X A_{22} = A_{12}$ reaches its upper bound (22), that is

$$\|X\|_F = \frac{\|A_{12}\|_F}{\text{sep}(A_{11}, A_{22})} = 351 \times 10^{-13},$$

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

$$\varepsilon_M(\|A_{11}\|_F + \|A_{22}\|_F) \|X\|_F = 2 \times 10^{-19},$$

However, in practice, the observed residual norm $\|Y\|_F = 3 \times 10^{-9}$, which is $\leq 5 \times 10^{-16}$. After swapping it turns out that

$$\|A_{21}\|_F = 3 \times 10^{-10},$$

so the swapping is perfectly stable.

5 Conclusions

In this paper, we have developed a robust swapping algorithm that handles the eigenvalues out of the domain of a matrix and Shur form preserving a posteriori similarity transformation. A complete set of codes and libraries has been developed and included in the LAPACK library [1]. The algorithm is guaranteed to numerically stable because we explicitly test for instability and do not avoid the eigenvalues if this would be unstable. However, we hope that the eigenvalues are so close that it is not possible to distinguish the instability. Usually, there is no proof of backward stability to the algorithm that this explicit test, so long as we can use the singular values instability will occur. We are concerned with limits on the stability algorithm in practice, but the asymptotic stability algorithm is occasionally found to work.

Acknowledgments: The authors would like to thank K. C. and B. Batte for sharing their program and initial work on the subject. The authors are grateful to G. W. Stewart for his valuable comments on the subject.

References

The 2 exception of algorithms SLAX and IXIG

<table>
<thead>
<tr>
<th>τ</th>
<th>SLAX</th>
<th>IXIG</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>( \lambda_2 = 0 ) ( 7.0001E + 01 \pm 0 ) ( 28569E + 02 )</td>
<td>( \lambda_2 = 0 ) ( 7.0367E + 01 \pm 0 ) ( 28548E + 02 )</td>
</tr>
<tr>
<td></td>
<td>( \lambda_1 = 0 ) ( 7.0009E + 01 \pm 0 ) ( 28561E + 02 )</td>
<td>( \lambda_1 = 0 ) ( 6.8155E + 01 \pm 0 ) ( 28589E + 02 )</td>
</tr>
<tr>
<td>10</td>
<td>( \lambda_2 = 0 ) ( 7.0000E + 01 \pm 0 ) ( 28558E + 02 )</td>
<td>( \lambda_2 = 0 ) ( 7.0367E + 01 \pm 0 ) ( 28675E + 02 )</td>
</tr>
<tr>
<td></td>
<td>( \lambda_1 = 0 ) ( 7.0009E + 01 \pm 0 ) ( 28561E + 02 )</td>
<td>( \lambda_1 = 0 ) ( 6.8155E + 01 \pm 0 ) ( 28589E + 02 )</td>
</tr>
<tr>
<td>100</td>
<td>( \lambda_2 = 0 ) ( 7.0000E + 01 \pm 0 ) ( 28558E + 02 )</td>
<td>not convergent</td>
</tr>
<tr>
<td></td>
<td>( \lambda_1 = 0 ) ( 7.0009E + 01 \pm 0 ) ( 28561E + 02 )</td>
<td>after 30 Q steps</td>
</tr>
</tbody>
</table>

On the upper bound of \( \| E \|_2 \): Finally, in the interest of theoretical analysis, we discuss the shapes of the bounds \( \| E \|_2 \), which obtains the numerical stability of algorithms SLAX and IXIG. For the test cases analyzed, we see that the bound (21) of \( \| E \|_2 \) is very pessimistic. However, we do find some cases indicating that the bound in (21) can roughly be attained. Let us consider the following case:

\[
A = \begin{pmatrix}
2 & 2 \\
A_{11} & A_{12}
\end{pmatrix} = \begin{pmatrix}
1.000E + 00 & -1.000E + 02 \\
0.000E - 02 & 1.000E + 00 \\
0.000E + 00 & 0.000E + 00
\end{pmatrix},
\]

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

\[
X = \begin{pmatrix}
1.000E + 00 & -2.000E + 02 \\
1.000E + 00 & -4.000E + 00
\end{pmatrix}
\]

is the solution of the Sylvester equation. We note that \( \sigma_1(X) = 2.00 \times 10^{-16} \). First we compute the residual matrix \( Y \) for computed solution \( \tilde{X} \) of the Sylvester equation is

\[
\| Y \|_F = \| A_{12} - A_{11} \tilde{X} + \tilde{X} A_{22} \|_F = 4.027 \times 10^{-12},
\]

which shows the estimated bound (1) of \( Y \):

\[
\varepsilon_M(\| A_{11} \|_F + \| A_{22} \|_F) \| X \|_F = 8.880 \times 10^{-12}.
\]

Furthermore, the observed result of (21) block of \( \tilde{A}_{21} \) after swapping is:

\[
\| \tilde{A}_{21} \|_2 = 1.203 \times 10^{-12}.
\]

which also roughly attain the bound (21) for \( \| E \|_2 \):

\[
\| E_{21} \|_2 \leq \frac{1}{1 + \sigma \| \frac{1}{2}(X) \|} \| Y \|_F = 2.027 \times 10^{-12}.
\]

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

\[
\| \tilde{A}_{21} \|_2 = 1.203 \times 10^{-12} \leq \varepsilon_M \| A \|_F = 4.489 \times 10^{-12}.
\]

For brevity, only five digits are displayed for all the data in this section though we did run in double precision.
### The Numerical Tests of Algorithm SLAEQC

<table>
<thead>
<tr>
<th>Test</th>
<th>Matrix $A_{11}, A_{22}$</th>
<th>$\text{sep}(A_{11}, A_{22})$</th>
<th>$E_Q$</th>
<th>$E_A$</th>
<th>Eigenvalues after swapping</th>
</tr>
</thead>
</table>
| 1    | \[
\begin{pmatrix}
2 & -87 & -20000 & 0.1000 \\
5 & 2 & -20000 & -1.0000 \\
0 & 0 & 1 & 1 \\
0 & 0 & 37 & 1
\end{pmatrix}
\] | $335 \times 10^{-1}$ | 0.260 | 0.197 | $0.160000E+01 \pm 0.0201424E+02$ |
| 2    | \[
\begin{pmatrix}
1 & 3 & 3 \times 10^{-6} & 4888 \\
1 & 1 & -3 & 1.4401 \\
0 & 0 & 1 & 1 \times 10^{3} \\
0 & 0 & 100 & 1.001
\end{pmatrix}
\] | $8.442 \times 10^{-4}$ | 0.625 | 0.423 | $0.160000E+01 \pm 0.172201E+01$ |
| 3    | \[
\begin{pmatrix}
1 & -100 & -100 & -100 \\
1 & 1 & 1200 & 0 \\
0 & 0 & 1 & 1 \\
0 & 0 & 100 & 1.001
\end{pmatrix}
\] | $2.00 \times 10^{-7}$ | 0.417 | 0.001 | $0.160000E+01 \pm 0.172201E+01$ |
| 4    | \[
\begin{pmatrix}
1 & 1 & 0 & 2 \\
1 & 1 & 9 & 0 \\
0 & 3 & 1 & 1 \\
0 & 0 & 1 & 1
\end{pmatrix}
\] | $\infty$ | 0.687 | 0.241 | $0.9999987E+00 \pm 0.172201E+01$ |

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

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

The eigenvalues after swapping are

$$\tilde{\lambda}_2 = 0.7010001E+01 \pm 0.2085661E+02,$$
$$\tilde{\lambda}_1 = 0.7000999E+01 \pm 0.2085665E+02,$$

which are accurate to machine precision. However, the output of algorithm EXCHING after 8 QR iterations¹ is

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

which has eigenvalues

$$\tilde{\lambda}_2 = 0.7026377E+01 \pm 0.2085408E+02,$$
$$\tilde{\lambda}_1 = 0.6984615E+01 \pm 0.2085919E+02.$$ 

They only have two decimal digits correct.

Table 2 shows the numerical results with different choices of parameter $\tau$, where when $\tau = 10$, it takes 17 QR iterations to converge. It clearly shows the superiority of algorithm SLAEQC. In particular, we note that algorithm EXCHING is nonconvergent when $\tau = 100$. It means that the eigenvalues are not able to be exchanged by algorithm EXCHING. But the algorithm SLAEQC has no difficulty. This convergence difficulty may reflect recent work of Batterton who has discovered classes of nonsymmetric matrices where QR iteration fails to converge, or converges quite slowly.

¹where the stopping criterion used in QR iteration is $\epsilon_p s = 1.2 \times 10^{-6}$
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 quantities and $E_A$.

All numerical experiments were carried out on a SUNsparc station 14. The arithmetics is IEEE standard single precision, with machine precision $2^{-23} \approx 1.192 \times 10^{-7}$.

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

\[
\lambda_1 = 0.200000E+01 \pm 0.208566E+02, \\
\lambda_2 = 1.000000E+01 \pm 0.201742E+02, 
\]

Test Matrix 2: moderate separation separation of $A_{11}$ and $A_{22}$, the eigenvalues before swapping are:

\[
\lambda_1 = 0.100000E+01 \pm 0.173205E+01, \\
\lambda_2 = 0.100100E+01 \pm 0.173291E+01. 
\]

Test Matrix 3: close eigenvalues, the corresponding the Sylvester equation is very ill-conditioned, the eigenvalues before swapping are

\[
\lambda_1 = 0.100000E+01 \pm 0.100000E+01, \\
\lambda_2 = 0.100000E+01 \pm 0.100000E+01. 
\]

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.

\[
\lambda_1 = 0.100000E+01 \pm 0.173205E+01, \\
\lambda_2 = 0.100000E+01 \pm 0.173205E+01. 
\]

Table 1 summarizes the results of algorithm SLAEXC, where 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 accuracy of the 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 [14], which uses QR iteration. Both algorithms perform well in most cases, but in certain cases, the algorithm EXCHNG is inferior to algorithm SLAEXC. For example, let

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

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

\[
\lambda_1 = 0.700100E+01 \pm 0.208566E+02, \\
\lambda_2 = 0.701000E+01 \pm 0.208566E+02,
\]
Remark 3. The fact of \( X/(1+\sigma \frac{1}{2} X) \) that affects \( \| E_1 \|_2 \) and \( \| E_2 \|_2 \) is interesting, since it warns that large and ill-conditioned \( X \) may endanger accuracy, because of (11) and

\[
\frac{\sigma_1(X)}{1+\sigma \frac{1}{2} X} = \frac{\sigma(X)}{\sigma_2(X) + \sigma \frac{1}{2} X},
\]

where \( \sigma(X) = \sigma_1(X)/\sigma_2(X) \). How \( \sigma(X) \), sep \((A_1, A_2)\), 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. As with other LAPACK routines, this algorithm was designed for accuracy, robustness, and portability.

The main subroutine is called STREXC. STREXC moves a given \( 1 \times 1 \) or \( 2 \times 2 \) diagonal block of a real quasi-triangular 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 \( 2 \times 2 \) block is implemented in subroutine SLAV2.

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 \( \tilde{Q} \) is to an orthogonal matrix, and (II) how close \( \tilde{Q} \tilde{A} \tilde{Q}^T \) is to the original matrix \( A \) where \( \tilde{A} \) is the computed \( \tilde{A} \). In other words, we need to test whether the two quantities

\[
E_Q = \frac{\| I - \tilde{Q}^T \tilde{Q} \|_1}{\varepsilon_M}, \quad E_A = \frac{\| A - \tilde{Q} \tilde{A} \tilde{Q}^T \|_1}{\varepsilon_M \| A \|_1}
\]

are around 1, where \( \varepsilon_M \) is machine precision. To check the changes among eigenvalues is not required to judge the correctness of an algorithm since we know that there must have at least an order of \( \tilde{O}(\varepsilon_M \| A \|) \) perturbation to the original matrix after swapping, and the nonsymmetric
Similarly, for $2\xi$ from $[1 \xi]$, we have $\| F R^T \|_F \leq 2 \| G \|_F$, therefore
\[
\| F_2 \|_2 \leq \| Q_2 \|_2 \| Y \|_F \| R^T \|_2 + 2 \| \tilde{A}_{22} \|_2 \| F R^T \|_F \leq \frac{\sigma_1(X)}{1 + \sigma_2^2(X)} \| Y \|_F + 4 \| \tilde{A}_{22} \|_2 \| G \|_2 \| \xi \|_F.
\]
Finally, for $2\xi$, we have
\[
\| F_1 \|_2 \leq \| Q_1 \|_2 \| Y \|_F \| R \|_2 = \frac{1}{1 + \sigma_1^2(X)} \| Y \|_F.
\]
Hence we have the following theorem

**Theorem 2.** Let $Y = A_{12} - A_{11} X + X A_{22}$, where $X = X + E$ is the computed solution of the Sylvester equation (6), assume that the error matrix $E$ is nonsingular, let the QR factorization of $(-X^T, I)^T$ satisfies
\[
\begin{bmatrix}
-X \\
I
\end{bmatrix} = Q \begin{bmatrix}
\tilde{R} \\
0
\end{bmatrix},
\]
then
\[
\tilde{Q}^T A \tilde{Q} = \begin{bmatrix}
\tilde{A}_{22} & \tilde{A}_{12} \\
0 & \tilde{A}_{11}
\end{bmatrix} + \begin{bmatrix}
F_{22} & F_{12} \\
F_{21} & F_{11}
\end{bmatrix},
\]
where $\tilde{A}_i$ is similar to $A_i, i = 1, 2$, and up to the first order perturbation $O(\| \xi \|_F)$
\[
\| F_1 \|_2 \leq \frac{\sigma_1(X)}{1 + \sigma_2^2(X)} \| Y \|_F, \tag{19}
\]
\[
\| F_2 \|_2 \leq \frac{\sigma_1(X)}{1 + \sigma_2^2(X)} \| Y \|_F + 4 \| \tilde{A}_{22} \|_2 \| G \|_2 \| \xi \|_F \tag{20}
\]
\[
\| F_2 \|_2 \leq \frac{1}{1 + \sigma_2^2(X)} \| Y \|_F. \tag{21}
\]

Three remarks are in order:

**Remark 1.** From the theorem we see that the departure $2\xi \|_2$ from upper block-triangular form (the measure of numerical instability) is bounded by $\| \xi \|_F + \sigma_2(X)$. It is easy to see that
\[
\| X \|_F \leq \frac{\| A_2 \|_F \| A_1 \|_F}{\text{sep}(A_1, A_2)}, \tag{22}
\]
where the equality is attained when $(d_i, 1)$ is a left singular vector of $K$ corresponding to the smallest singular values of $K = \text{sep}(A_1, A_2)$. Combining (22), (11) and (21), we have
\[
\| F_2 \|_2 \leq P \| Y \|_F + \| A_{22} \|_F \| A_2 \|_F.
\]
Logically, the above bound indicates that the numerical instability will occur if we have small $\text{sep}(A_1, A_2)$. But in practice, numerical experiments show that this upper bound is very pessimistic. Small $\text{sep}(A_1, A_2)$ 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 $X$, (unless the Sylvester equation is too close to singular), but it need not improve $\| Y \|$ at least when Gaussian elimination with complete pivoting is used to solve the Sylvester equation.
Swapping Algorithm

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 subblock $Q$ of $Q_{12}$ 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

$$R^T R = I + X^T X
$$

Let $\sigma (Q)$ denote the set of singular values of matrix $Q$ and $\lambda (Q)$ denote the set of eigenvalues of matrix $Q$, then

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

Therefore

$$
\|Q_1\|_2 = \|R^{-1}\|_2 = \frac{1}{\sigma_2(R)} = \left( \frac{1}{1 + \sigma^2(X)} \right)^{1/2},
$$

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

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

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

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

$$U^T Q V = \frac{k}{k} \begin{pmatrix} C & S \\ -S & C \end{pmatrix},
$$

where

$$C = \text{diag}(c_1, c_2, \ldots, c_k) \geq 0, \quad S = \text{diag}(s_1, s_2, \ldots, s_k) \geq 0, \quad C^2 + S^2 = I.
$$

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

$$
\|Q_1\|_2 = \frac{\sigma_1(X)}{\sqrt{1 + \sigma^2(X)}}
$$

and

$$
\|Q_2\|_2 = \|Q_1\|_2; \quad \|Q_2\|_2 = \|Q_1\|_2;
$$

Thus, for $\|L_1\|_2$, we have

$$
\|L_1\|_2 \leq \|Q_2^T\|_2 \|Y\|_F \|R^T\|_2 \|Q_1^T\|_2 \|Q_2^T\|_2 = \frac{\sigma_1(X)}{\sqrt{1 + \sigma^2(X)}} \|Y\|_F.
$$
so that

\[ Z_{11} = -I + (R - Q^T_{12} FR)^{-1}, \]
\[ Z_{21} = -Q^T_{12} BR^{-1}, \]

and up to the first order perturbations, we have

\[ Z_{11} = -Q^T_{11} BR^{-1} - FR^{-1}, \quad (15) \]
\[ Z_{21} = -Q^T_{12} BR^{-1}. \quad (16) \]

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

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

By canceling \((-X, I)^T\) from both sides of the equation, and premultiplying \( W \), we obtain

\[ W^T \begin{bmatrix} -X & E \\ I \end{bmatrix} = \begin{bmatrix} F \\ 0 \end{bmatrix} + Q^T \begin{bmatrix} E \\ 0 \end{bmatrix}. \]

By inserting \( Q^T = I \) in the left side of the above equation and noting that \( QW = -Q^T W = -Z \), we have

\[ Z \begin{bmatrix} R \\ 0 \end{bmatrix} - Z \begin{bmatrix} Q^T_{11} E \\ Q^T_{12} E \end{bmatrix} = - \begin{bmatrix} F \\ 0 \end{bmatrix} - \begin{bmatrix} Q^T_{11} E \\ Q^T_{12} E \end{bmatrix}. \]

Thus the "bottom" equation is

\[ Z_{21} R - Z_{21} Q^T_{11} E - Z_{22} Q^T_{12} E = -Q^T_{12} E. \]

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

\[ Z_{22} = -Z_{21} Q^T_{11} Q^{-T}_{12} = Q^T_{12} BR^{-1} Q^T_{11} Q^{-T}_{12}. \quad (17) \]

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

\[ E_{11} = Q^T_{12} A_{11} Q^{-T}_{12} Q^T_{12} BR^{-1} Q^T_{11} Q^{-T}_{12} - Q^T_{12} BR^{-1} Q^T_{11} Q^{-T}_{12} Q^T_{12} A_{11} Q^{-T}_{12} + Q^T_{12} BR^{-1} (-RA_{22} R^{-1} Q^T_{11} Q^{-T}_{12} + Q^T_{11} A_{11} Q^{-T}_{12}) \]
\[ = Q^T_{12} (A_{11} E_{22} A_{22}) R^{-1} Q^T_{11} Q^{-T}_{12} \]
\[ = -Q^T_{12} YR^{-1} Q^T_{11} Q^{-T}_{12}, \]

and

\[ E_{22} = -RA_{22} R^{-1} (Q^T_{11} BR^{-1} + FR^{-1}) + (Q^T_{11} BR^{-1} + FR^{-1}) RA_{22} R^{-1} \]
\[ -(-RA_{22} R^{-1} Q^T_{11} Q^{-T}_{12} + Q^T_{11} A_{11} Q^{-T}_{12}) Q^T_{12} BR^{-1} \]
\[ = Q^T_{11} (-A_{11} E_{22} A_{22}) R^{-1} - A_{22} FR^{-1} + FR^{-1} A_{22} \]
\[ = Q^T_{11} YR^{-1} - A_{22} FR^{-1} + FR^{-1} A_{22}, \]

and

\[ E_{21} = -Q^T_{12} A_{11} Q^{-T}_{12} Q^T_{12} BR^{-1} + Q^T_{12} BR^{-1} RA_{22} R^{-1} \]
\[ = -Q^T_{12} A_{11} E_{22} A_{22} R^{-1} \]
\[ = Q^T_{12} YR^{-1}. \]
factor $\gamma = 1$. Including these rounding errors does not change the conclusion of the analysis, but makes the exposition appear more complicated.

Let $X$ be the computed solution of the Sylvester equation, where $X + E$ is the exact solution, and $E$ is an error matrix. By the argument of (12), and a result of Stewart in the perturbation of the QR factorization, we know that under mild conditions (such that $\|E\|_F < 1$), the QR factorization of $(X^T, I)^T$ can be written as

$$
\begin{bmatrix}
-X & I \\
-I & 0
\end{bmatrix} + \begin{bmatrix}
-E & 0 \\
0 & 0
\end{bmatrix} = Q + \begin{bmatrix}
R + F & 0 \\
0 & 0
\end{bmatrix},
$$

where $W$ and $F$ are the perturbations of the orthogonal matrix $Q$ and the triangular matrix $R$, respectively, and $Q = Q + W$ is orthogonal. $\|W\|_F$ and $\|F\|_F$ are essentially bounded by the terms of order $\|E\|_2 \|E\|_F$. From $(Q + W)^T(Q + W) = I$, up to the first order, we have $Q^TW = -W^TQ$.

When $Q = Q + W$ transforms $A$ ignoring the second order perturbations we have

$$
\dot{Q}^TA \dot{Q} = (Q + W)^T A (Q + W) = Q^TAQ + W^TQA + QA^TW + WA^T.
$$

Defining $Z = Q^TW$ and partitioning it conformally within the form

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

we have

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

where

$$
E_{11} = \dot{A}_{11}Z_{22} - Z_{22}\dot{A}_{11} - Z_{21}\dot{A}_{12},
$$

$$
E_{22} = \dot{A}_{22}Z_{11} - Z_{11}\dot{A}_{22} + \dot{A}_{12}Z_{21},
$$

$$
E_{21} = \dot{A}_{11}Z_{21} - Z_{21}\dot{A}_{22}.
$$

$E_{11}$ and $E_{22}$ perturb the eigenvalues directly and do not affect stability of interest because it measures the numerical stability of swapping $E$ is the error in the block $2$. 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 $E$, $E_{22}$ and $E_{21}$. To do so, let us first express $Z$ in terms of the blocks $Q$ of $Q$, $E$ $F$ and $R$. From (13), we have

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

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

$$
(I + Z) \begin{bmatrix}
I & 0 \\
0 & 0
\end{bmatrix} = \begin{bmatrix}
R - Q_{11}^TE \\
-Q_{12}^TE
\end{bmatrix} (R + F)^{-1},
$$
Swapping Algorithm

In the next section, we will show that in some pathological cases, the norm of the \( (2,1) \) (block) entry of \( Q^T A Q \) may be larger than \( O(\varepsilon M \| A \|) \), i.e., it may be backward unstable if we are forced to treat \( Q^T A Q \) 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) \( Q^T A Q \) is less than or equal to \( Q \| A \| \), 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 \( A_1 \) 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 \( 2 \times 2 \) 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 \( 2 \times 2 \) block (complex conjugate eigenpair) could cause a large perturbation of its eigenvalues. In the extreme case, a \( 2 \times 2 \) block could split into two \( 1 \times 1 \) 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:**

1. Copy A to \( T \): \( T = \begin{bmatrix} T_{11} & T_{12} \\ 0 & T_{22} \end{bmatrix} \rightarrow A = \begin{bmatrix} A_{11} & A_{12} \\ 0 & A_{22} \end{bmatrix} \).

2. Use Gaussian elimination with complete pivoting to solve

\[
T_{11} X \rightarrow \begin{bmatrix} X_{11} & X_{12} \\ 0 & X_{22} \end{bmatrix} = \gamma T_{12},
\]

where \( \gamma \) 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 \( G = (-W \gamma I)^T = QR \) by Householder transformations.

4. Perform swapping tentatively: if the norm of the \( (2,1) \) (block) entry \( T_{12} Q \) is less than \( O(\varepsilon M \| T \| M) \), go to the next step, and otherwise exit;

5. If the swap is accepted, replace \( A \) by \( Q^T A Q \) and set the \( (2,1) \) (block) entry of \( Q^T A Q \) to zero.

6. Standardize \( 2 \times 2 \) diagonal blocks if any exist.

In our implementation of SLAEXC in LAPACK, we have chosen \( 10 \| A \| M \) as the stability criterion in step 4, where \( M = \max_{ij} |a_{ij}| \). 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 topleft 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 \( 2 \times 2 \) 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 are exact, and the scaling
Swapping Algorithm

where \( A_i \) is similar to \( A_j \) for \( i = 1, 2 \), 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

\[
Q^T \begin{pmatrix}
-X \\
I_q
\end{pmatrix} = \begin{pmatrix}
R \\
0
\end{pmatrix}
\]  

(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

\[
A_{11}X - \mu = \gamma A_{12}
\]  

(8)

or the corresponding linear system

\[
\tilde{K} = \gamma \tilde{b}
\]  

(9)

where \( \gamma \) is a scaling factor \((\gamma \leq 1)\), and \( K_{\gamma} = A_{12} \otimes I_p \otimes I_q \) is the Kronecker product, \( x = \text{col}(X), b = \text{col}(A_{12}) \), \( \text{col}(W) \) denotes the column vector formed by taking columns of \( W \) and stacking them top one another from left to right. Possible overflow of \( X \) is taken care of by choosing a small scaling factor \( \gamma \). In the extreme case, where \( A_{11} \) and \( A_{22} \) have the same eigenvalues, we choose \( \gamma = 0 \). Because the linear system (9) can only be \( 1 \times 1, 2 \times 2 \) or \( 4 \times 4 \), 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 \( 4 \) by \( 4 \) matrix, so setting tiny pivots to a chosen tiny value controls the conditioning of the system and not of the solution). Applying standard results from [20] straightforward analysis shows that for the computed solution \( \hat{X} \) of the Sylvester equation:

\[
\frac{\| \hat{X} \|_F}{\| X \|_F} \leq \frac{\rho \epsilon_M (\| A_1 \|_F + \| A_{22} \|_F)}{\text{sep}(A_{11}, A_{22})},
\]  

(10)

where \( F = X \), \( \rho \) is a small constant of order \( O(\| \epsilon \|_{1, \infty}) \), \( \epsilon \) is machine precision, and \( \text{sep}(A_{11}, A_{22}) = \sigma_{\infty}(K) \) 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 \( Y = A_{11} \hat{X} + \hat{X} A_{22} = -A_{11} \tilde{K} \tilde{b} \tilde{a} \). Applying standard error analysis of Gaussian elimination on \( Y \) have

\[
\| Y \|_F = \| A_{12} - A_{11} \tilde{X} + \tilde{X} A_{22} \|_F \leq \rho \epsilon_M (\| A_1 \|_F + \| A_{22} \|_F) \| \hat{X} \|_F.
\]  

(11)

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

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

\[
\begin{pmatrix}
-\tilde{X} \\
\gamma I
\end{pmatrix} = Q \begin{pmatrix}
\tilde{R} \\
0
\end{pmatrix},
\]  

(12)

where \( \tilde{Q} = Q + \delta Q, \| \delta Q \| \approx \epsilon_M, Q^T \tilde{Q} = I \). In other words, the computed matrix \( \tilde{Q} \) is orthogonal to machine precision [20].
Swapping Algorithm

This means that its diagonal entries are equal and its off diagonal nonzero and of opposite sign:
\[
\begin{bmatrix}
\alpha & \beta \\
\gamma & \alpha
\end{bmatrix}
\quad \beta \gamma < 0.
\] (4)

For any \(2 \times 2\) 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 \(A_1\) and \(A_2\) in the following block matrix
\[
A= \begin{bmatrix}
A_{11} & A_{12} \\
0 & A_{22}
\end{bmatrix}
\] (5)

where \(A_{11}\) is \(p \times p\), \(A_2\) is \(q \times q\), \(p, q = 1\) or \(2\). Throughout this paper, we assume that \(1\) and \(A_2\) 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
\[
A_{11}X - XA_{22} = A_{12}. \quad (6)
\]

Since it is assumed that \(A\) and \(A_2\) have no eigenvalue in common, the solution \(X\) exists and is unique. If we choose an orthogonal matrix \(Q\) such that
\[
Q^T \begin{bmatrix}
-X \\
I_q
\end{bmatrix} = \begin{bmatrix}
R \\
0
\end{bmatrix}
\]

and conformally partition \(Q\) in the form
\[
Q = \begin{bmatrix}
Q_{11} & Q_{12} \\
Q_{21} & Q_{22}
\end{bmatrix},
\]

then
\[
Q^T \begin{bmatrix}
-X & I_p \\
I_q & 0
\end{bmatrix} = \begin{bmatrix}
R & QT_1 \\
0 & QT_2
\end{bmatrix}.
\]

Since both matrices on the left are invertible, so are \(R\) and \(QT_2\) and thus
\[
Q^T \begin{bmatrix}
A_{11} & A_{12} \\
0 & A_{22}
\end{bmatrix} Q = Q^T \begin{bmatrix}
I_p & -X \\
0 & I_q
\end{bmatrix}
\begin{bmatrix}
A_{11} & 0 \\
0 & A_{22}
\end{bmatrix}
\begin{bmatrix}
I_p & X \\
0 & I_q
\end{bmatrix} Q
\]
\[
= \begin{bmatrix}
R & QT_1 \\
0 & QT_2
\end{bmatrix}
\begin{bmatrix}
A_{22} & 0 \\
0 & A_{11}
\end{bmatrix}
\begin{bmatrix}
R^{-1} & -R^{-1}QT_1Q_{12}^T \\
0 & Q_{12}^T
\end{bmatrix}
\]
\[
= \begin{bmatrix}
RA_{22}R^{-1} & -RA_{22}R^{-1}QT_1Q_{12}^T + Q_1T_1A_{11}Q_{12}^T \\
0 & Q_{12}^T
\end{bmatrix}
\equiv \begin{bmatrix}
\tilde{A}_{22} & \tilde{A}_{12} \\
0 & \tilde{A}_{11}
\end{bmatrix}.
\]
and hence $Q_1$ gives an orthonormal basis for the invariant subspace of $A$ corresponding to the eigenvalues contained in $T$

Unfortunately, the $T$ 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 $1 \times 1$ or $2 \times 2$ diagonal blocks by an orthogonal transformation. Formally, let $A_1$ be a $p \times p$ matrix, $A_2$ 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

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

(3)

where $\tilde{A}_{ii}$ is similar to $A_i$ for $i = 1, 2$, 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 $1 \times 1$ and $2 \times 2$ blocks of a quasi-triangular matrix, which we refer to as algorithm EXCITE. 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 converge in the lower part. 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 QR 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 Dąbrowski, Hammarling and Wilkinson in 1983, although the paper was finished later (1991). The algorithm is developed in a program to implement the direct swapping algorithm. A similar idea has also been published by Gao 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 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: describes the direct swapping algorithm. The error analysis of the algorithm is carried out. The software implementation and numerical experiments are reported in §5. §6 draws conclusions. All software including test software for the algorithm in this paper can be found in the LAPACK library [1].

We assume that any $2 \times 2$ diagonal block in the quasi-triangular matrix is in standardized form.
On Swapping Diagonal Blocks in Real Schur Form*

Zhaojun Bai† and James W. Demmel‡

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 quasi-triangular 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 associated invariant subspaces, computing partial eigenvalues of a large nonsymmetric matrix by the simultaneous iteration method [14], computing matrix functions [4], 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 \( \mathbf{A} \) there is a real orthogonal matrix \( \mathbf{Q} \) such that

\[
\mathbf{A} = \mathbf{Q} \mathbf{T} \mathbf{Q}^T,
\]

where \( \mathbf{T} \) is a real upper quasi-triangular matrix, called the real Schur form. This means that \( \mathbf{A} \) is block upper triangular with \( 1 \times 1 \) and \( 2 \times 2 \) blocks on the diagonal. The \( 1 \times 1 \) blocks contain the real eigenvalues of \( \mathbf{A} \). The eigenvalues of the \( 2 \times 2 \) diagonal blocks are the complex conjugate eigenvalues of \( \mathbf{A} \). The real Schur form may be computed using subroutine \texttt{HQR} from \texttt{HSF} [13] or subroutine \texttt{SISQOR} from \texttt{LAPACK} [1]. Here \( \mathbf{Q} \) provides an orthonormal basis for the invariant subspaces of certain subsets of eigenvalues of the matrix \( \mathbf{A} \). If we partition \( \mathbf{Q} \) and \( \mathbf{T} \) conformally as

\[
\mathbf{Q} = [\mathbf{Q}_1; \mathbf{Q}_2], \quad \mathbf{T} = \begin{bmatrix} \mathbf{T}_{11} & \mathbf{T}_{12} \\ \mathbf{0} & \mathbf{T}_{22} \end{bmatrix},
\]

then from (1) we have

\[
\mathbf{A} \mathbf{Q}_1 = \mathbf{Q}_1 \mathbf{T}_{11}
\]