SOLVING REAL LINEAR SYSTEMS WITH THE COMPLEX SCHUR DECOMPOSITION *

CARLA D. MORAVITZ MARTIN † AND CHARLES F. VAN LOAN ‡

Abstract. If the complex Schur decomposition is used to solve a real linear system, then the computed solution generally has a complex component because of roundoff error. We show that the real part of the computed solution that is obtained in this way solves a nearby *real* linear system. Thus, it is "numerically safe" to obtain real solutions to real linear systems via the complex Schur decomposition. This result is useful in certain Kronecker product situations where fast linear equation solving is made possible by reducing the involved matrices to their complex Schur form. This is critical because in these applications one cannot work with the real Schur form without greatly increasing the volume of work.

Key words. Linear Systems, Schur decomposition, Back-substitution, Kronecker products

AMS subject classifications. 15A06, 65F05, 65G50

1. Introduction.

The *Schur decomposition* states that if $A \in \mathbb{R}^{n\times n}$, then there exists a unitary $Q \in \mathbb{C}^{n\times n}$ so that $Q^H AQ = T$ is upper triangular. The eigenvalues that appear along the diagonal of T can be arbitrarily ordered. See [3, p.313].

This decomposition, coupled with back-substitution and matrix-vector multiplication, can be used to solve a real linear system $Ax = b$. Indeed, since $Q^H b = (Q^H AQ)(Q^H x) = T(Q^H x)$ we have

```
Algorithm SchurSolve
    Step 1. Compute the Schur decomposition $Q^H AQ = T$.
    Step 2. Form $c = Q^H b$.
    Step 3. Solve $Ty = c$ by back-substitution.
    Step 4. Set $x = Qy$.
```

Ordinarily, it is preferred to work with the *LU* factorization because it is much cheaper. However, there are settings involving Kronecker products when this is not the case. For example, the *Sylvester equation*

$$FX + XG^T = B \qquad F \in \mathbb{R}^{m\times m},\ G \in \mathbb{R}^{n\times n},\ B \in \mathbb{R}^{m\times n}$$

can be reshaped as $Ax = b$ where $A = I_n \otimes F + G \otimes I_m$, $x = \text{vec}(X)$, and $b = \text{vec}(B)$. (Here, vec$(\cdot)$ makes a column vector out of a matrix by stacking its columns.) The *LU* factorization of A involves $O(m^3n^3)$ flops. But if we compute the Schur decompositions $Q_F^H FQ_F = R$ and $Q_G^H GQ_G = S$ and set $Q = Q_F \otimes Q_G$, then $Q^H AQ = I_n \otimes R + S \otimes I_m$ is the Schur decomposition of A and `SchurSolve` requires $O(n^3+m^3)$ flops if the Kronecker structure is exploited. See [3, p.367].

A problem with `SchurSolve` is that complex arithmetic arises whenever A has complex eigenvalues. This increases the volume of work. Moreover, the computed solution vector x will inevitably have a complex component because of roundoff error.

*This work was supported by NSF grant CCR-9901988.

†Mathematics, Cornell University, 227 Malott Hall, Ithaca, NY 14853-7510, `carlam@cam.cornell.edu`

‡Computer Science, Cornell University, 4130 Upson Hall, Ithaca, NY 14853-7510, `cv@cs.cornell.edu`

These problems can be avoided by working with the *real* Schur decomposition. In this factorization we find a real orthogonal Q so that $Q^TAQ = T$ is upper quasi-triangular, i.e., block triangular with 1-by-1 and 2-by-2 diagonal blocks. Because T is "almost" triangular, the `SchurSolve` philosophy essentially applies, except that a quasi-triangular system is solved in Step 3 of `SchurSolve`, rather than a (complex) triangular system.

Therefore, an algorithm to solve a linear system using the real Schur decomposition appears to involve a simple modification of `SchurSolve`. However, there are situations where the real Schur decomposition is much more expensive to compute than the (complex) Schur decomposition. Consider (again) the Sylvester equation problem. If we have computed the real Schur decompositions $Q_{_F}^TFQ_{_F} = R$ and $Q_{_G}^TGQ_{_G} = S$ and set $Q = Q_{_F} \otimes Q_{_G}$, then $Q^TAQ = I_n \otimes R \, + \, S \otimes I_m$ is *not* the real Schur decomposition of A. Attempting to compute the canonical form would destroy the Kronecker structure and would greatly increase the volume of work. Fortunately, there is a way of handling the subdiagonal blocks of $I_n \otimes R \, + \, S \otimes I_m$ using clever permutations so that the overall procedure remains $O(m^3 + n^3)$. (See [3, p.367].)

However, in [4] we describe another Kronecker product situation where the permutation "device" does not work – specifically, the shifted Kronecker product system

$$\left(A^{(p)} \otimes \cdots \otimes A^{(1)} - \lambda I_{_N}\right) x = b \qquad \lambda \in \mathbb{R},\ b \in \mathbb{R}^{^N},\ N = n_1 \cdots n_p\ , \tag{1.1}$$

where $A^{(i)} \in \mathbb{R}^{n_i \times n_i}$ for $i = 1, \ldots, p$. After computing the real Schur decompositions of the $A^{(i)}$, a fast recursive procedure exists to solve for x if the $A^{(i)}$ have real eigenvalues. However, if the $A^{(i)}$ have complex eigenvalues, the resulting p-fold Kronecker product of quasi-triangular matrices has a complicated and very problematic block structure below the diagonal, thus increasing the volume of work needed by the recursive procedure. This impasse brings us back to `SchurSolve` and the main contribution of this paper. In particular, we examine the properties of the real part of the computed solution $\hat{x}$.

The analysis to determine if a computed solution of a system solves a nearby system of the same form is illustrative of recent work in the general area of "structured" perturbations and error analysis. For example, in [5, 6] the conditioning of structured linear systems is examined where the structure includes symmetric, Toeplitz, circulant, and Hankel matrices. In addition, [1] analyzes the stability of algorithms for solutions of symmetric indefinite systems. In our paper, we show that the real part of the computed solution solves a nearby *real* system, a type of structured perturbation. We are not the first to examine complex algorithms for real problems. For example, [2] compares the condition of a complex eigenvalue of a real matrix under real and complex perturbations in order to analyze the accuracy of real algorithms versus complex algorithms. Our work is in this spirit, expanding what we know about structured perturbations for the case when "structure" means real data.

In §2 we show that `SchurSolve` produces a complex solution that solves a nearby, but complex, linear system. This result is not new but is included for the sake of completeness. We then proceed to prove a perturbation theorem in §3. It shows that when a real linear system is subjected to complex perturbations, then the real part of the solution to the perturbed system solves a nearby real linear system. This is followed by a brief summary in §4.

Throughout this paper we use the 2-norm. The 2-norm condition of a matrix M is denoted by $\kappa(M)$. The unit roundoff is designated by $\mathbf{u}$. We repeatedly use the fact that if $M \in \mathbb{C}^{m \times n}$, then both $\| \operatorname{Re}(M) \|$ and $\| \operatorname{Im}(M) \|$ are bounded by $\| M \|$.

2. Backward Error Analysis. We show that if $\hat{x}$ is the solution produced by `SchurSolve` when floating point arithmetic is used, then

$$(A + \Delta A)\,\hat{x} = b + \Delta b \tag{2.1}$$

$$\| \Delta A \| \le \delta_A \| A \| \tag{2.2}$$

$$\| \Delta b \| \le \delta_b \| b \| \tag{2.3}$$

where the δ's are modest multiples of the unit roundoff $\mathbf{u}$. To present an uncluttered but sufficiently rigorous analysis, we adopt the convention that all the δ's below are $O(\mathbf{u})$ in magnitude. The floating point result of a matrix calculation is indicated by fl$(\cdot)$. The floating point properties associated with the Schur decomposition, back substitution, and other basic computations can be found in [3].

In Step 1 the computed Schur decomposition of $A \in \mathbb{R}^{n\times n}$ produces a "nearly" unitary $\hat{Q} \in \mathbb{C}^{n\times n}$. That is, there is an exactly unitary $Q \in \mathbb{C}^{n\times n}$ such that

$$Q = \hat{Q} + \Delta Q \qquad \| \Delta Q \| \le \delta_1\ . \tag{2.4}$$

The computed Schur form $\hat{T}$ satisfies

$$\hat{T} = Q^H(A + H)Q \qquad \| H \| \le \delta_2 \| A \|\ , \tag{2.5}$$

where $H \in \mathbb{C}^{n\times n}$. Accounting for the roundoff error in Step 2, there exists $\Delta b \in \mathbb{C}^n$ such that

$$\hat{c} = \mathrm{fl}(\hat{Q}^H b) = Q^H(b + \Delta b) \qquad \| \Delta b \| \le \delta_b \| b \| \tag{2.6}$$

while in Step 3 the computed solution to the triangular system satisfies

$$(\hat{T} + G)\hat{y} \;=\; \hat{c} \qquad \| G \| \le \delta_3 \| \hat{T} \| \le \delta_4 \| A \|\ . \tag{2.7}$$

In the last step the computed solution $\hat{x}$ can be related to $\hat{y}$ as follows:

$$\hat{x} = \mathrm{fl}(\hat{Q}\hat{y}) \;=\; Q(\hat{y} + g) \qquad \| g \| \le \delta_5 \| \hat{y} \| \le \delta_6 \| \hat{x} \|\ . \tag{2.8}$$

Now let us combine these results. From (2.6) and (2.7) we have

$$(\hat{T} + G)\hat{y} \;=\; Q^H(b + \Delta b)$$

and so by (2.5) and (2.8)

$$b + \Delta b \;=\; Q\left(\hat{T} + G\right)Q^H Q\hat{y} \;=\; \left(A + H + QGQ^H\right)(\hat{x} - Qg)\,.$$

If $M = A + H + QGQ^H$, then

$$\begin{aligned} b + \Delta b = M(\hat{x} - Qg) \;&=\; \left(M - \frac{MQg\hat{x}^H}{\hat{x}^H\hat{x}}\right)\hat{x} \\ &= \left(A + H + QGQ^H - \frac{MQg\hat{x}^H}{\hat{x}^H\hat{x}}\right)\hat{x}. \end{aligned}$$

and so if we define

$$\Delta A = H + QGQ^H - \frac{MQg\hat{x}^H}{\hat{x}^H\hat{x}} \tag{2.9}$$

then $(A+\Delta A)\,\hat{x} \;=\; b+\Delta b$, i.e., $\hat{x}$ solves a perturbed system. From (2.6) we know that Δb satisfies (2.3). Thus, the verification of (2.1)-(2.3) is complete once we show that $\|\, \Delta A \,\|$ is sufficiently small. Towards that end we note that

$$\begin{aligned} \|\, M \,\| = \|\, A+H+QGQ^H \,\| &\le \|\, A \,\| + \|\, H \,\| + \|\, G \,\| \\ &\le (1+\delta_2+\delta_4)\|\, A \,\| = (1+\delta_7)\|\, A \,\|. \end{aligned}$$

It follows from (2.5), (2.7), (2.8) and (2.9) that

$$\begin{aligned} \|\, \Delta A \,\| \le \|\, H \,\| + \|\, G \,\| + \|\, M \,\| \frac{\|\, g \,\|}{\|\, \hat{x} \,\|} \\ \le \delta_2 \|\, A \,\| + \delta_4 \|\, A \,\| + \delta_6(1+\delta_7)\|\, A \,\| \end{aligned}$$

The inequality (2.2) is established by setting $\delta_A = \delta_2 + \delta_4 + \delta_6(1+\delta_7)$.

3. A Perturbation Theorem.

In this section we prove a result that will enable us to say something very favorable about the real part of the computed `SchurSolve` solution.

THEOREM 3.1. *Suppose* $0 < \epsilon \le 1/6$ *and that* $A \in \mathbb{R}^{n\times n}$ *and* $b \in \mathbb{R}^n$ *with* $\epsilon \cdot \kappa(A) \le 1/2$. *If*

$$(A+E)z = b+f \tag{3.1}$$

where

$$\begin{aligned} E &= E_1 + iE_2\,, & E_1,\, E_2 &\in \mathbb{R}^{n\times n}\,, & \|\, E \,\| &\le \epsilon \|\, A \,\| \\ f &= f_1 \,+\, if_2\,, & f_1,\, f_2 &\in \mathbb{R}^n, & \|\, f \,\| &\le \epsilon \|\, b \,\| \\ z &= z_1 \,+\, iz_2\,, & z_1,\, z_2 &\in \mathbb{R}^n, \end{aligned}$$

then there exists a real matrix $\tilde{E} \in \mathbb{R}^{n\times n}$ *such that*

$$\left(A+\tilde{E}\right) z_1 \;=\; b+f_1 \tag{3.2}$$

and

$$\|\, \tilde{E} \,\| \le 4\epsilon \|\, A \,\| \tag{3.3}$$

$$\|\, f_1 \,\| \le \epsilon \|\, b \,\|\;. \tag{3.4}$$

Proof. Since

$$\|\, f_1 \,\| \le \|\, f_1 + if_2 \,\| = \|\, f \,\| \le \epsilon \|\, b \,\|,$$

the inequality (3.4) holds. Note that if $b = 0$ then $\|\, f \,\| = 0$ and so $\|\, f_1 \,\| = 0$. Expanding (3.1) we get

$$(A+E_1+iE_2)(z_1+iz_2) \;=\; b+f_1+if_2$$

from which follows

$$(A+E_1)z_1 \;-\; E_2 z_2 \;=\; b \;+\; f_1 \tag{3.5}$$

$$(A+E_1)z_2 \;+\; E_2 z_1 \;=\; f_2\;. \tag{3.6}$$

If $b = 0$ and $z_1 = 0$, then any such $\tilde{E}$ such that $\| \tilde{E} \| \leq 4\epsilon \| A \|$ completes the proof. If $z_1 \neq 0$, equation (3.5) can be rewritten as

$$\left(A + E_1 - \frac{E_2 z_2 z_1^T}{z_1^T z_1} \right) z_1 \ = \ b + f_1$$

and so (3.2) holds with

$$\tilde{E} = E_1 - \frac{E_2 z_2 z_1^T}{z_1^T z_1}. \tag{3.7}$$

Now to establish (3.3), we start by taking norms in (3.7):

$$\| \tilde{E} \| \leq \| E_1 \| + \| E_2 \| \frac{\| z_2 \|}{\| z_1 \|} \ \leq \ \epsilon \| A \| \left(1 + \frac{\| z_2 \|}{\| z_1 \|} \right) \ . \tag{3.8}$$

Looking at (3.3), we must confirm that $\| z_2 \|$ is not too much bigger than $\| z_1 \|$. From (3.6) we have

$$z_2 = (A + E_1)^{-1} \left(f_2 - E_2 z_1 \right) \ = \ (I + A^{-1} E_1)^{-1} A^{-1} \left(f_2 - E_2 z_1 \right)$$

and so

$$\| z_2 \| \ \leq \ \| (I + A^{-1} E_1)^{-1} \| \| A^{-1} \| \left(\| f_2 \| + \| E_2 \| \| z_1 \| \right) \ .$$

The assumption $\epsilon \cdot \kappa(A) < 1/2$ implies

$$\| (I + A^{-1} E_1)^{-1} \| \ \leq \ \frac{1}{1 - \| A^{-1} E_1 \|} \ \leq \ \frac{1}{1 - \epsilon \cdot \| A \| \| A^{-1} \|} \ \leq \ 2$$

and thus

$$\| z_2 \| \leq 2\epsilon \| A^{-1} \| \left(\| b \| + \| A \| \| z_1 \| \right) \ . \tag{3.9}$$

By rearranging (3.5) we see that $b = (A + E_1) z_1 - E_2 z_2 - f_1$ and therefore

$$\begin{aligned} \| b \| &\leq \ (\| A \| + \| E_1 \|) \| z_1 \| \ + \ \| E_2 \| \| z_2 \| \ + \ \| f_1 \| \\ &\leq \ (1 + \epsilon) \| A \| \| z_1 \| \ + \ \epsilon \| A \| \| z_2 \| \ + \ \epsilon \| b \| \\ &\leq \ \frac{1+\epsilon}{1-\epsilon} \| A \| \| z_1 \| \ + \ \frac{\epsilon}{1-\epsilon} \| A \| \| z_2 \| \ . \end{aligned}$$

By substituting this inequality into (3.10) and using the assumption that $\epsilon \leq 1/6$ we get

$$\begin{aligned} \| z_2 \| &\leq 2\epsilon \| A^{-1} \| \left(\frac{1+\epsilon}{1-\epsilon} \| A \| \| z_1 \| \ + \ \frac{\epsilon}{1-\epsilon} \| A \| \| z_2 \| \ + \ \| A \| \| z_1 \| \right) \\ &= 2\epsilon \kappa(A) \left(\frac{2}{1-\epsilon} \| z_1 \| \ + \ \frac{\epsilon}{1-\epsilon} \| z_2 \| \right) \\ &\leq \left(\frac{2}{1-\epsilon} \| z_1 \| \ + \ \frac{\epsilon}{1-\epsilon} \| z_2 \| \right) \\ &\leq \frac{2}{1-2\epsilon} \| z_1 \| \ \leq \ 3 \| z_1 \| \ . \end{aligned}$$

The inequality (3.3) follows from this and (3.9).

The proof will be complete after we address whether z_1 can be zero. By way of contradiction, assume $z_1 = 0$. Then (3.5) and (3.6) become $-E_2z_2 = b + f_1$ and $(A + E_1)z_2 = f_2$, respectively. So

$$\begin{aligned} b &= -f_1 - E_2z_2 \\ &= -f_1 - E_2(A + E_1)^{-1}f_2. \end{aligned}$$

This implies that

$$\begin{aligned} \| b \| &\leq \| f_1 \| + \| E_2 \| \| (A + E_1)^{-1} \| \| f_2 \| \\ &\leq \epsilon \| b \| + 2\epsilon \| A \| \| A^{-1} \| (\epsilon \| b \|) \\ &\leq 2\epsilon \| b \| \end{aligned}$$

which means that $1 \leq 2\epsilon$, a contradiction. Since z_1 is nonzero, the proof is now complete.

□

In §2 we showed that `SchurSolve` produces a computed solution $\hat{x}$ that exactly solves a (*complex*) linear system that is "within roundoff" of the original. Thus,

$$\frac{\| \hat{x} - x \|}{\| x \|} \approx \mathbf{u}\kappa(A) \; .$$

Since $\| \operatorname{Re}(\hat{x}) - x \| \leq \| \hat{x} - x \|$ it follows that

$$\frac{\| \operatorname{Re}(\hat{x}) - x \|}{\| x \|} \approx \mathbf{u}\kappa(A)$$

which is what we would expect from a stable linear equation solving process. But we can say more in light of Theorem 3.1. The following Corollary to Theorem 3.1 shows that using the complex Schur decomposition to solve a real problem results in a computed solution whose real part solves a nearby real system.

COROLLARY 3.2. *Suppose $A \in \mathbb{R}^{n\times n}$ and $b \in \mathbb{R}^n$, and $\hat{x}$ is the computed solution to $Ax = b$ using `SchurSolve`. In addition suppose that $\epsilon = \max(\delta_A, \delta_b) \leq 1/6$. Then there exists $\Delta A \in \mathbb{R}^{n\times n}$ and $\delta b \in \mathbb{R}^n$ such that*

$$(A + \Delta A)\operatorname{Re}(\hat{x}) = b + \delta b \tag{3.10}$$

where

$$\| \Delta A \| \leq \delta_A \| A \| \tag{3.11}$$

$$\| \delta B \| \leq \delta_b \| b \|. \tag{3.12}$$

4. Summary. We have illustrated certain situations where the complex Schur decomposition is preferred to using the real Schur decomposition when solving a real system. Thus, we show that it is numerically safe to obtain solutions to the real system by introducing complex arithmetic. In particular, Theorem 3.1 and Corollary 3.2 show that the real part of the computed solution obtained using `SchurSolve` solves a nearby *real* linear system.

REFERENCES

[1] James R. Bunch, James W. Demmel, and Charles F. Van Loan. *The Strong Stability of Algorithms for Solving Symmetric Systems*, SIAM J. Matrix Anal. Appl, 10 (1989), 494-499.

[2] R. Byers and D. Kressner. *On the Condition of a Complex Eigenvalue under Real Perturbations*, BIT Numerical Mathematics, 44 (2004), 209-214.

[3] Gene H. Golub and Charles F. Van Loan. *Matrix Computations.* Third Edition, Johns Hopkins University Press, Baltimore, MD, 1996.

[4] Carla D. Moravitz Martin and Charles F. Van Loan. *Shifted Kronecker Product Systems*, To appear: SIAM J. Matrix Anal. Appl. (2006).

[5] Siegfried M. Rump. *Structured Perturbations Part I: Normwise Distances*, SIAM J. Matrix Anal. Appl., 25 (2003), 1-30.

[6] Siegfried M. Rump. *Structured Perturbations Part II: Componentwise Distances*, SIAM J. Matrix Anal. Appl., 25 (2003), 31-56.