SIAM J. Matrix Anal. Appl.
Vol. 18, No. 3, pp. 566–589, July 1997

AN EFFICIENT IMPLEMENTATION OF THE NONSYMMETRIC LANCZOS ALGORITHM[*]

DAVID DAY[†]

Abstract. Lanczos vectors computed in finite precision arithmetic by the three-term recurrence tend to lose their mutual biorthogonality. One either accepts this loss and takes more steps or re-biorthogonalizes the Lanczos vectors at each step. For the symmetric case, there is a compromise approach. This compromise, known as maintaining semiorthogonality, minimizes the cost of reorthogonalization. This paper extends the compromise to the two-sided Lanczos algorithm and justifies the new algorithm.

The compromise is called *maintaining semiduality.* An advantage of maintaining semiduality is that the computed tridiagonal is a perturbation of a matrix that is exactly similar to the appropriate projection of the given matrix onto the computed subspaces. Another benefit is that the simple two-sided Gram–Schmidt procedure is a viable way to correct for loss of duality.

A numerical experiment is included in which our Lanczos code is significantly more efficient than Arnoldi's method.

Key words. Lanczos algorithm, breakdown, sparse eigenvalue problems, biorthogonalization methods

AMS subject classification. 65F15

PII. S0895479895292503

1. Introduction.

For non-Hermitian matrices approximate eigenvalues from the (two-sided) Lanczos process are much more accurate (for the same elapsed time and starting vectors) than those from the Arnoldi method. Consequently, it is important to implement the Lanczos algorithm as well as possible. This paper summarizes the analysis in [7] and claims to show the best (or nearly best) way to do it.

This article shows that it is not necessary to re-biorthogonalize the Lanczos vectors at every step to approximate the behavior of the algorithm in exact arithmetic. A property of the computed Lanczos vectors called *semiduality* may be imposed (defined in section 3) and suffices for keeping close to the exact algorithm at minimal cost. Semiduality is less expensive to maintain than duality, yet equally effective. Having stated our contribution, we resume the introduction.

Krylov subspace methods determine a useful basis for the Krylov subspace

$$\mathcal{K}_i(q, B) = \operatorname{span}(q, Bq, \ldots, B^{i-1}q).$$

The eigenvalues of certain projections of B onto $\mathcal{K}_i(q, B)$ serve as approximations to eigenvalues of B. The eigenvalues of projections of B are often called Ritz values. They are not Raleigh–Ritz approximations, except in the Hermitian case, but we do not have a better name for them.

[*] Received by the editors September 27, 1995; accepted for publication (in revised form) by M. Gutknecht July 9, 1996. This work was performed by an employee of the U. S. Government or under U. S. Government contract. The U. S. Government retains a nonexclusive, royalty-free license to publish or reproduce the published form of this contribution, or allow others to do so, for U. S. Government purposes. Copyright is owned by SIAM to the extent not limited by these rights.

http://www.siam.org/journals/simax/18-3/29250.html

[†] Applied and Numerical Math, Sandia National Laboratories, Albuquerque, NM 87185-5800 (dday@cs.sandia.gov). The author gratefully acknowledges support from ONR contract N00014-90-J-1372 and the Sandia National Laboratory AMS Fellowship.

The most popular Krylov subspace method for non-Hermitian matrices is the Arnoldi algorithm [27]. An orthonormal basis of $\mathcal{K}_i(q, B)$ is computed. The orthogonal projection of B onto $\mathcal{K}_i(q, B)$ is represented by an $i \times i$ Hessenberg matrix.

The non-Hermitian or two-sided Lanczos algorithm is another Krylov subspace method. Given two starting vectors $p^* = p_1^*$ and $q = q_1$, the two-sided Lanczos algorithm simultaneously computes a basis for the right Krylov subspace $\mathcal{K}_i(q, B)$ and a dual basis for the left Krylov subspace

$$\mathcal{K}^i(p^*, B) = \operatorname{span}(p^*, \ldots, p^* B^{i-1}).$$

The Lanczos algorithm computes the partial reduction of B to tridiagonal form.

In exact arithmetic the Hermitian Lanczos algorithm determines a matrix of orthogonal vectors Q, while the two-sided Lanczos algorithm determines two matrices P and Q such that P^*Q is diagonal. This relation among the Lanczos vectors is often called *biorthogonality* [16].

Lanczos vectors computed in finite precision arithmetic by the three-term recurrence tend to lose their mutual biorthogonality. Two ways to compensate for this phenomenon are known: Lanczos with full re-biorthogonalization (LanFRB) [4] and an acceptance of the loss of biorthogonality which forces more steps to be taken [6, 10]. For the Hermitian case, a compromise is known [11, 19, 20, 28, 29]. This compromise, known as maintaining semiorthogonality, minimizes the number of reorthogonalizations. This article extends the compromise to the two-sided Lanczos algorithm and justifies the new algorithm.

There are known cases when the simple recurrence takes extreme amounts of time [10]. On the other hand, LanFRB is expensive for a long run. The compromise we present in this article is better than either of the two extremes.

Better approximations to eigenvalues of B tend to be computed from a single Krylov subspace of dimension 100 than from four Krylov subspaces of dimension 25. To take advantage of this property, we want to use large Krylov subspaces. The amount of data transfer (from memory to the computational unit) required in Lanczos with full re-biorthogonalization when n is large is significant. In this respect, the compromise is at least twice as fast as maintaining full biorthogonality. Usually it is much faster. See section 6.2.

The state-of-the-art in Lanczos methods for eigenvalue problems is to select from one of four algorithms. For linear solvers there are many more options: two-term versus three-term, CGS, BiCGStab1/2/e, QMR, TFQRM, and so on. But for eigenvalue problems the user first selects either the three-term recurrence or LanFRB. This choice is a trade-off between the low cost per step of the three-term recurrence and the limited number of Lanczos steps taken by LanFRB. Then the user selects an implementation with or without look-ahead [10, 22, 21]. Look-ahead enhances stability while increasing cost modestly. This article does not consider implementations with the look-ahead feature.

The word reorthogonalization in the Hermitian case is ugly enough, but the analogous term re-biorthogonalization goes too far (nine syllables). So we seek a term with fewer syllables. In functional analysis row vectors represent linear functionals and the property $p_i^* q_j = \delta_{ij}$ (Kronecker's delta) says that the ordered sets $\{p_1^*, \ldots, p_j^*\}$ and $\{q_1, \ldots, q_j\}$ are a pair of dual bases for $\mathcal{K}_i(q, B)$ and $\mathcal{K}^i(p^*, B)$. So we use the term dual instead of biorthogonal. Consequently, we speak of maintaining duality, local duality, and semiduality (introduced in section 3).

1.1. Summary. Our results extend earlier work done in the Hermitian case [28, 29], but new issues arise in the non-Hermitian case. In exact arithmetic the Lanczos algorithm determines a tridiagonal–diagonal pencil (T, Ω) such that $\Omega^{-1}T$ is similar to the projection of B onto the spans of the Krylov subspaces. See Definition 2.1. The diagonal elements of Ω are defined to be the inner products of consecutive pairs of normalized Lanczos vectors. In exact arithmetic the algorithm breaks down if $\Omega = \mathrm{diag}(\omega_i)$ is singular. In finite precision arithmetic breakdowns are rare, but near breakdowns are not. It is tempting to require that $|\omega_i| \geq \sqrt{\epsilon}$, where ϵ is the round-off unit, but the rather lengthy analysis of [7] shows that the algorithm is still viable provided that $|\omega_j| \geq (n + 10j)\epsilon$. Below that level the accuracy of the Ritz values does not generally improve if the recurrence continues.

The remainder of this work is organized as follows. Section 2 contains a discussion of what is known about solving eigenvalue problems using the two-sided Lanczos process. The basic properties of the Lanczos algorithm are reviewed, the implementation of the three-term recurrence is outlined, convergence theory is discussed, and our practical experience with implementations of the three-term recurrence is summarized.

With that done, we move on to the tricky issue of when to "re-biorthogonalize" or correct the Lanczos vectors to restore duality. The candidate Lanczos vectors are computed by the three-term recurrence, but at certain steps the loss of duality of the candidate Lanczos vectors to the previous Lanczos vectors is "too large," and then we correct them to obtain the final Lanczos vectors. To obtain a competitive algorithm, correction steps are implemented just like a step of LanFRB. Since the duality of the Lanczos vectors is not maintained to full precision, this process must be justified. The viability of the two sided Gram–Schmidt process is established in section 3.

In section 4 the properties of the Lanczos algorithm *with correction* are developed. In section 4.4 we show how to monitor the loss of duality among the computed Lanczos vectors without significantly increasing the cost of the algorithm. For efficiency the correction steps must be invoked as rarely as possible consistent with maintaining accuracy in the approximations.

In section 5 we prove that an added advantage of maintaining semiduality is that the computed pencil (T, Ω) is a perturbation of a pencil that is exactly equivalent to the projection of the operator onto the computed subspaces. The norm of the perturbation is as small as the data warrants. Section 6 illustrates some of our results with some challenging numerical examples.

2. Two-sided Lanczos.

The two-sided Lanczos algorithm is based on the partial reduction of a non-Hermitian matrix B to tridiagonal form. The Lanczos algorithm starts from an arbitrary pair of vectors $p^* = p_1^*$ and $q = q_1$. After j successful steps, the matrices P_j^* and Q_j are produced. The rows of P_j^* span the Krylov subspace $\mathcal{K}^j(p^*, B)$ and the columns of Q_j span $\mathcal{K}_j(q, B)$. The matrix $T_j = P_j^* B Q_j$ is tridiagonal; $\Omega_j = P_j^* Q_j$ is diagonal. In finite precision arithmetic the latter will no longer be true, and we will set $\Omega_j = \mathrm{diag}(P_j^* Q_j)$.

Certain implementations scale the Lanczos vectors so that $\Omega = I$ [1, 37], and others maintain the unit length of all the Lanczos vectors [4, 10, 22]. Our analysis of the Lanczos algorithm requires that the unit length of all the Lanczos vectors be maintained. Normalizing the Lanczos vectors is necessary in this work because for the resulting more complicated algorithm it is possible to establish certain properties of the quantities computed in finite precision arithmetic which are required to justify the Lanczos algorithm with correction (see [7]); this would be impossible based on less precise models such as [1]. A useful result of our work is that Ω can become nearly singular, $\operatorname{cond}(\Omega) = \mathcal{O}(1/\epsilon)$, without spoiling the algorithm.

The eigenvalues of an *oblique projection* of B are used to approximate the eigenvalues of B.

Definition 2.1. *Let* $Q_j = [q_1, \ldots, q_j]$ *and* $P_j^* = [p_1, \ldots, p_j]^*$ *have full rank. If* $P_j^* Q_j$ *is invertible then*

$$\Pi_j = Q_j \Omega_j^{-1} P_j^*$$

is a projector ($\Pi_j^2 = \Pi_j$). *It is not orthogonal* ($\Pi_j^* \neq \Pi_j$) *in general. We say that* Π_j *is an oblique projector onto* $\operatorname{Range}(Q_j)$. *It is also an oblique projector onto the dual space* $\{u^* \Pi_j : u \in \mathbf{C}^j\} = \operatorname{Range}(P_j)^*$. *Thus* $\Pi_j B \Pi_j$ *is a projection of* B *onto the pair* $\operatorname{Range}(Q_j)$ *and* $\operatorname{Range}(P_j)^*$.

Assuming that Q_n and P_n^* exist (that is, the algorithm does not break down), the representation of B with respect to the basis $\{q_1, \ldots, q_n\}$ is $Q_n^{-1} B Q_n$ and we have $\Pi_n = I$, which implies that $Q_n^{-1} = \Omega_n^{-1} P_n^*$ and $Q_n^{-1} B Q_n = \Omega_n^{-1} T_n$. The tridiagonal $\Omega_j^{-1} T_j$ represents $\Pi_j B \Pi_j$ in the dual bases $\{q_1, \ldots, q_j\}$ and $\{\omega_1^{-1} p_1^*, \ldots, \omega_j^{-1} p_j^*\}$. Similarly, the representation corresponding to P_j^* and $Q_j \Omega_j^{-1}$ is $T_j \Omega_j^{-1}$.

2.1. The three-term recurrences.

The Lanczos vectors satisfy a pair of three-term recurrences

$$\beta_{i+1} p_{i+1}^* = p_i^* B - \frac{\alpha_i}{\omega_i} p_i^* - \frac{\gamma_i \omega_i}{\omega_{i-1}} p_{i-1}^* \tag{2.1}$$

and

$$q_{i+1} \gamma_{i+1} = B q_i - q_i \frac{\alpha_i}{\omega_i} - q_{i-1} \frac{\beta_i \omega_i}{\omega_{i-1}}. \tag{2.2}$$

The coefficients α_i and ω_i are chosen so that the right-hand side of (2.1) annihilates $q_1, \ldots, q_i$ and the right-hand side of (2.2) is annihilated by $p_1^*, \ldots, p_i^*$. The βs and γs come from the normalizing convention. The recurrence stops if $\beta_{j+1} \omega_{j+1} \gamma_{j+1} = 0$. With

$$T_j := \operatorname{tridiag} \begin{pmatrix} & \beta_2 \omega_2, & \cdots & , \beta_j \omega_j & \\ \alpha_1, & & \cdots & & , \alpha_j \\ & \gamma_2 \omega_2, & \cdots & , \gamma_j \omega_j & \end{pmatrix},$$

equations (2.1) and (2.2) may be written in compact form:

$$P_j^* B - T_j \Omega_j^{-1} P_j^* = e_j \beta_{j+1} p_{j+1}^* \tag{2.3}$$

and

$$B Q_j - Q_j \Omega_j^{-1} T_j = q_{j+1} \gamma_{j+1} e_j^*, \tag{2.4}$$

where $e_j = (0, \ldots, 0, 1)^*$.

2.2. Ritz triplet convergence.

The eigenvalues of B are approximated using the eigenvalues of the pair (T_j, Ω_j) for increasing j. Given an eigentriplet $(u_i^{(j)*}, \theta_i^{(j)}, v_i^{(j)})$,

$$u_i^{(j)*} T_j = \theta_i^{(j)} u_i^{(j)*} \Omega_j \quad \text{and} \quad T_j v_i^{(j)} = \Omega_j v_i^{(j)} \theta_i^{(j)},$$

form the Ritz triplet $(x_i^{(j)*}, \theta_i^{(j)}, y_i^{(j)})$ where

$$x_i^{(j)*} = u_i^{(j)*} P_j^* \quad \text{and} \quad y_i^{(j)} = Q_j v_i^{(j)}. \tag{2.5}$$

Ritz triplets approximate eigentriplets of B. In discussions of the analysis of the quantities computed after j Lanczos steps, we omit the superscript (j) for clarity.

The expression $x_i^* \times (2.4) \times v_i$ reduces to

$$x_i^* B y_i = \theta_i x_i^* y_i.$$

In other words, θ_i is the *generalized* Rayleigh quotient corresponding to $x_i^* = u_i^* P_j^*$ and $y_i = Q_j v_i$. See section 11 of [18] for a discussion of generalized Rayleigh quotients.

We now list what is known about the approximations derived from the first j steps of the algorithm.

Multiply (2.4) by v_i from the right and substitute (2.5) to obtain

$$B y_i - y_i \theta_i = q_{j+1} \gamma_{j+1} v_i(j), \tag{2.6}$$

where $v_i(j)$ is the jth component of $v_i = v_i^{(j)}$. The remarkable property of (2.6) is that the right-hand side can be computed without forming y_i. Even in exact arithmetic $\|y_i\|_2$ can be smaller than $\|v_i\|_2$. Thus a small value of $|v_i(j)|$ is a necessary though not sufficient indication that θ_i is close to an eigenvalue of B.

The perturbation theory for the eigenvalue problem is more complicated than in the Hermitian case. The Lanczos algorithm eventually yields approximate eigentriples $(\hat{x}_i^*, \theta_i, \hat{y}_i)$, where $\|\hat{x}_i^*\|_2 = 1 = \|\hat{y}_i\|_2$ such that the corresponding residuals $\|\hat{x}^*(B - \theta I)\|_2$ and $\|(B - \theta I)\hat{y}\|_2$ are small. Such triples exactly solve a nearby eigenvalue problem [14]. The good thing is that the eigenvalues of $\Omega_j^{-1} T_j$ for which the residual norms are small persist as approximate eigenvalues of $\Omega_k^{-1} T_k$ for $k > j$ [14].

The residual norm $\|(B - \theta_i I)\hat{y}_i\|_2$ is a pessimistic estimate of the accuracy of θ_i and a good estimate of the accuracy of $\hat{y}_i$. The accuracy of generalized Rayleigh quotients is proportional to the product of the residual norms. To be precise, if $(\hat{x}_i^*, \theta_i, \hat{y}_i)$ approximates an eigentriple of B that is well separated (see Theorem 2.1 in section 5 of [31]), then the accuracy of θ_i is proportional to

$$\|\hat{x}_i^*(B - \theta_i I)\|_2 \|(B - \theta_i I)\hat{y}_i\|_2.$$

This product divided by

$$\text{gap}(\theta_i, T_j) = \min_{k \neq i} |\theta_i - \theta_k|$$

appears to be a realistic backward error estimate for θ_i [3].

One-sided algorithms, and in particular the Arnoldi algorithm, do not enjoy this property.

To factor the exact shrinkage $\|x\|_2/\|u\|_2$ and $\|y\|_2/\|v\|_2$ into the error estimates to obtain asymptotic error bounds, one must first compute the Ritz triplets. For an $n \times n$ real operator B, after j Lanczos steps the number of real floating point operations (flops) required to compute the matrices of left and right eigenvectors for m Ritz values is $8nmj$. This is often more flops than are required for the Lanczos run.

Fortunately, a realistic lower bound on the shrinkage is available if the duality of the computed Lanczos vectors is maintained. In this case $y = Q_j v$ satisfies $P_j^* y =$

$\Omega_j v$. Also, since the Lanczos vectors are normalized to have unit Euclidean length, $\|P_j^*\|_2 \leq \sqrt{j}$. Combine these two equations to find

$$\|y\|_2 \geq \frac{\|P_j^*\|_2}{\sqrt{j}}\|y\|_2 \geq \frac{\|P_j^* y\|_2}{\sqrt{j}} = \frac{\|\Omega_j v\|_2}{\sqrt{j}}.$$

Similarly, if $x^* = u^* P_j^*$, then $\|x^*\|_2 \geq \|u^*\Omega_j\|_2/\sqrt{j}$.

2.3. Practical experience. Without careful observation, there is no science. This section discusses surprising behavior that has been consistently observed in large-scale scientific computations using the non-Hermitian Lanczos algorithm [10]. In the case $p_1 = q_1$,

- the sequence $\{|\omega_i|\}_{i>0}$ tends to be decreasing, sometimes precipitously,
- even when $|\omega_j|$ has declined to nearly $10n\epsilon$, approximate eigenvalues, eigenvectors, and solutions to linear systems continue to converge,
- even when $|\omega_j|$ has declined to nearly $10n\epsilon$, (T_j, Ω_j) is almost always graded so that the growth factor

$$\Phi_j = \max(\|T_j\Omega_j^{-1}\|_\infty, \|\Omega_j^{-1}T_j\|_1)/\|B\|_2 \tag{2.7}$$

  is of order unity, say less than 10.

As usual, ϵ denotes the machine precision. The scalar Φ_j measures the relative size of the intermediate quantities introduced during the Lanczos algorithm. It is essential to distinguish

$$\|\Omega_j^{-1}\|_2 = 1/\min_i |\omega_i|$$

from Φ_j. The quantity $\|\Omega_j^{-1}\|_2$ can be large, nearly ϵ^{-1}, without necessarily affecting the accuracy of the eigenvalues and eigenvectors. The error in computing the eigenvalues of $\Omega_j^{-1}T_j$ is, among other things, proportional to $\|\Omega_j^{-1}T_j\|_1$. But in the rare case that Φ_j is large, the Lanczos algorithm is unstable due to the introduction of large intermediate quantities. The approximate eigenvalues suffer perturbations like $\epsilon\Phi_j\|B\|_2$. In this case, look-ahead Lanczos is recommended [10, 13, 21]. When $\Phi_j = \mathcal{O}(1)$, we expect our approximations to be as accurate as the data warrants.

3. The viability of the two-sided Gram–Schmidt process. This section studies the central problem of how to maintain adequate duality between the two sequences of Lanczos vectors $\{p_1^*, \ldots, p_j^*\}$ and $\{q_1, \ldots, q_j\}$ at a reasonable expense. It will help to recall the corresponding technical problem in the symmetric case when $p_i = q_i$, $i = 1, \ldots, j$. See [23, 20, 28, 29]. Suppose that the three-term recurrence, in finite precision arithmetic, returns a unit vector q'_{j+1} that is not orthogonal to the previous q_i; i.e., $Q_j^* q'_{j+1}$ is not negligible. The Gram–Schmidt process replaces q'_{j+1} by a normalized version of $(I_j - Q_jQ_j^*)q'_{j+1}$. This procedure is appropriate if $Q_j^*Q_j = I_j$, but can actually make things worse if Q_j's columns are not orthogonal. The interesting question here is how much $\|I_j - Q_j^*Q_j\|_2$ can be permitted to grow and yet guarantee that $(I_j - Q_jQ_j^*)q'_{j+1}$ is orthogonal to range(Q_j) to within working accuracy.

Our problem is similar but more complicated. The formal two-sided Gram–Schmidt operator is $I_j - Q_j\Omega_j^{-1}P_j^*$, where $\Omega_j = \text{diag}(P_j^*Q_j)$. How large can we permit $\|I_j - P_j^*Q_j\Omega_j^{-1}\|_2$ to grow and yet get what we want by applying $I_j - Q_j\Omega_j^{-1}P_j^*$ to $(p'_{j+1})^*$ and q'_{j+1}?

The answer in the symmetric case is that semiorthogonality defined in equation (3.1) suffices: if

$$\|Q_i^* q_{i+1}\|_1 \leq \sqrt{\epsilon} \tag{3.1}$$

holds for $i = 1, \ldots, j-1$ and if $\|Q_j^* q'_{j+1}\|_1 \leq \sqrt{\epsilon}$ then we may take $q_{j+1} = q'_{j+1}$.

We shall give a similar condition in the two-sided case–semiduality suffices. However, the definition of semiduality is not as simple as in the symmetric case, and we postpone it until more notation has been developed. An added benefit of maintaining semiduality is that the computed tridiagonal–diagonal pair of Lanczos matrices (T_j, Ω_j) is equivalent, to within round-off error, to the true "projection" of B, namely, $(P_j^* B Q_j, P_j^* Q_j)$. More precisely, we will show that the no-breakdown condition

$$\min_{i \leq j} |\omega_i| \geq (n + 10j)\epsilon \tag{3.2}$$

and the semiduality condition

$$\max_{i \leq j-1} (\|(p'_{i+1})^* Q_i |\Omega_i|^{-1/2}\|_\infty, \||\Omega_i|^{-1/2} P_i^* q'_{i+1}\|_1) \leq \sqrt{\epsilon} \tag{3.3}$$

suffice to ensure the preservation of (T_j, Ω_j) described in the previous sentence (see Theorem 5.2). It is necessary to strengthen condition (3.3) somewhat to guarantee the viability of two-sided Gram–Schmidt (GS) when (3.2) is nearly an equality. Note that in the symmetric case $\Omega_j = I_j$ and (3.3) reduces to (3.1) as claimed.

The superscript $'$ in p'_{i+1} and q'_{i+1} indicates that these are the candidate Lanczos vectors computed by the three-term recurrence, but not necessarily the actual $i+1$th Lanczos vectors. The vectors p'_{i+1} and q'_{i+1} have been normalized.

3.1. Analysis of GS. We are going to derive a sequence of matrices $\{M_j\}_{j>0}$ whose norm is the "right" factor by which duality is enhanced in GS. Recall from section 2.1 that at the end of step j the Lanczos algorithm has computed dual matrices of Lanczos vectors P_j^* and Q_j, candidate Lanczos vectors $(p'_{j+1})^*$ and q'_{j+1}, and ω_{j+1} denotes the computed value of the inner product $(p'_{j+1})^* q'_{j+1}$. We assume $\omega_{j+1} \neq 0$. Due to the loss of duality, $P_j^* Q_j \neq \Omega_j \equiv \text{diag} P_j^* Q_j$ and off-diagonal entries of $P_j^* Q_j$ could be as large as 1 if the three-term recurrence is not modified.

Suppose that $\|P_j^* q'_{j+1}\|_2$ and $\|(p'_{j+1})^* Q_j\|_2$ are too big (criterion to be discussed later). GS yields new candidates $(\breve{p}_{j+1})^*$ and $\breve{q}_{j+1}$ satisfying

$$(\breve{p}_{j+1})^* = (p'_{j+1})^* (I_j - Q_j \Omega_j^{-1} P_j^*)$$

and

$$\breve{q}_{j+1} = (I_j - Q_j \Omega_j^{-1} P_j^*) q'_{j+1}.$$

Now we examine the new duality situation:

$$\begin{aligned} (\breve{p}_{j+1})^* Q_j &= (p'_{j+1})^* (I_j - Q_j \Omega_j^{-1} P_j^*) Q_j \\ &= (p'_{j+1})^* Q_j (I_j - \Omega_j^{-1} P_j^* Q_j) \end{aligned}$$

and

$$\begin{aligned} P_j^* \breve{q}_{j+1} &= P_j^* (I_j - Q_j \Omega_j^{-1} P_j^*) q'_{j+1} \\ &= (I_j - P_j^* Q_j \Omega_j^{-1}) P_j^* q'_{j+1}. \end{aligned}$$

The factor Ω_j^{-1} in the middle is alarming because we expect some ω_i to become quite small and we fear that off-diagonal entries may rise too close to 1. This feature is absent in the symmetric case. However, the situation is better than it appears.

We can obtain a more balanced expression for the duality of the new vectors $(\breve{p}_{j+1})^*$ and $\breve{q}_{j+1}$ by writing

$$\Omega_j = |\Omega_j|^{1/2}\mathrm{sign}(\Omega_j)|\Omega_j|^{1/2}.$$

Then modified expressions for the duality, namely,

$$\begin{aligned} \breve{p}^*_{j+1}Q_j|\Omega_j|^{-1/2} &= p^{\prime *}_{j+1}Q_j(I_j - \Omega_j^{-1}P_j^*Q_j)|\Omega_j|^{-1/2} \\ &= p^{\prime *}_{j+1}Q_j|\Omega_j|^{-1/2}\mathrm{sign}(\Omega_j^*)(\mathrm{sign}(\Omega_j) - |\Omega_j|^{-1/2}P_j^*Q_j|\Omega_j|^{-1/2}) \qquad (1) \end{aligned}$$

and

$$\begin{aligned} &|\Omega_j|^{-1/2}P_j^*\breve{q}_{j+1} \\ &= (\mathrm{sign}(\Omega_j) - |\Omega_j|^{-1/2}P_j^*Q_j|\Omega_j|^{-1/2})\mathrm{sign}(\Omega_j^*)|\Omega_j|^{-1/2}P_j^*q'_{j+1}, \qquad (3.4) \end{aligned}$$

show that the balanced "reducing factor" after applying GS is $\|M_j\|$, where

$$M_j = \mathrm{sign}(\Omega_j) - |\Omega_j|^{-1/2}P_j^*Q_j|\Omega_j|^{-1/2}. \qquad (3.5)$$

We get no benefit from the cost of GS unless $\|M_j\|$ is much less than 1.

We choose to measure duality using

$$\||\Omega_j|^{-1/2}P_j^*q_{j+1}\|_1 \quad \text{and} \quad \|p^*_{j+1}Q_j|\Omega_j|^{-1/2}\|_\infty$$

and define the effectiveness of GS using the balanced *connection* matrix M_j. Note that $\|x\|_1 = \|x^*\|_\infty$.

To illustrate the advantage of a balanced connection matrix, we applied LanFRB to the matrix B that arises from the finite difference discretization (five-point stencil) of the partial differential operator

$$L[u](\mathbf{x}) = -\Delta u + 50\nabla\cdot(u\mathbf{x}) - 125u$$

on a regular 31×31 grid over the unit square with zero boundary values [35]. Though the eigenvalue problem for B is ill posed (because the coefficients 50 and 125 are enormous compared to the grid size), this example is relevant because breakdown occurs at step 56, and at the previous step

$$|\omega_{55}| \approx 1e-13 \quad \text{and} \quad \|\Omega_{55}^{-1}T_{55}\|_1 \approx 11\|B\|_1.$$

Figures 1 and 2 display the absolute values of the entries of the unbalanced connection matrix $I_{55} - \Omega_{55}^{-1}P_{55}^*Q_{55}$ and the balanced connection matrix M_{55} on a semilog scale. Though $\|I_{55} - \Omega_{55}^{-1}P_{55}^*Q_{55}\|_1 \approx 1e-4$, the norm of the balanced operator is much less, $\|M_{55}\|_1 \approx 2e-11$.

Recall that $(\breve{p}_{j+1})^*$ and $\breve{q}_{j+1}$ are obtained from $(p'_{j+1})^*$ and q'_{j+1} by GS. If

$$\|(p'_{j+1})^*Q_j|\Omega_j|^{-1/2}\|_\infty \le \frac{\epsilon}{\|M_j\|_\infty}, \qquad (3.6)$$

then, by (3.4), one trivially has

$$\|(\breve{p}_{j+1})^*Q_j|\Omega_j|^{-1/2}\|_\infty \le \epsilon.$$

Similarly, if

$$\||\Omega_j|^{-1/2}P_j^*q'_{j+1}\|_1 \le \frac{\epsilon}{\|M_j\|_1}, \qquad (3.7)$$

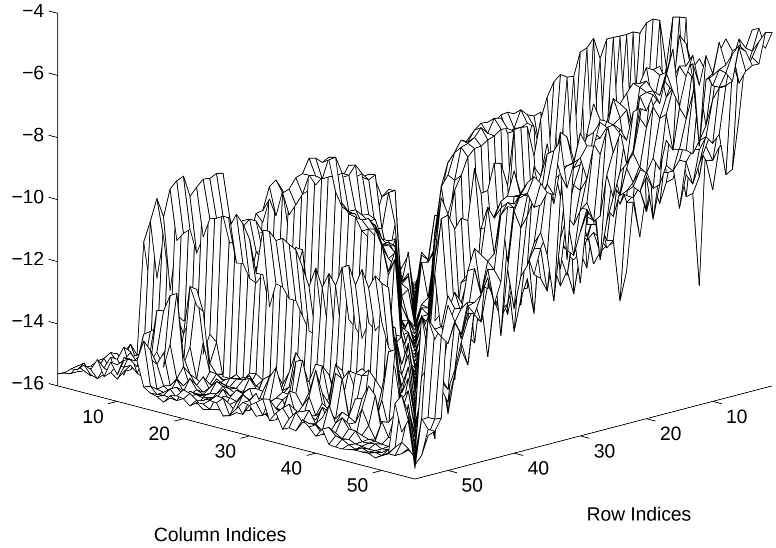

FIG. 1. $I_{55} - \Omega_{55}^{-1} P_{55}^* Q_{55}$ *(unbalanced op.)*; $\|I_{55} - \Omega_{55}^{-1} P_{55}^* Q_{55}\|_1 \approx 1\mathrm{e}-4$.

then, by (3.4),

$$\||\Omega_j|^{-1/2} P_j^* \breve{q}_{j+1}\|_1 \leq \epsilon.$$

The sequence

$$(\max(\|(p'_{j+1})^* Q_j |\Omega_j|^{-1/2}\|_\infty, \||\Omega_j|^{-1/2} P_j^* q'_{j+1}\|_1))_{j\geq 1} \tag{3.8}$$

tends to increase with j gradually until the last term is too big. At that step a correction step is made (that is, $q'_{j+1} \to \breve{q}_{j+1}$ and $p'_{j+1} \to \breve{p}_{j+1}$). This change reduces the latest term in (3.8) to ϵ. Hence semilog graphs of (3.8) look sawtoothed.

We use this perspective on GS to find the "right" definition of semiduality. For overall efficiency we want to minimize the total number of corrections and particularly avoid unnecessary corrections near the end of a Lanczos run. So we seek the weakest conditions that give adequate levels of duality. To this end we explicitly ensure that

$$\max(\|(p'_{j+1})^* Q_j |\Omega_j|^{-1/2}\|_\infty \|M_j\|_\infty, \|M_j\|_1 \||\Omega_j|^{-1/2} P_j^* q'_{j+1}\|_1) \leq \epsilon. \tag{3.9}$$

Condition (3.9) takes no account of ω_{j+1}, and if $|\omega_{j+1}|$ is too small it is essential not to accept $(p'_{j+1})^*$ and q'_{j+1}. So, in addition to (3.9) we must take account of possible growth in $\|M_{j+1}\|_1$. To analyze this, note that the nonzero part of the rightmost column of M_{j+1} is

$$\Omega_j^{-1/2} P_j^* q'_{j+1} |\omega_{j+1}|^{-1/2}.$$

Since

$$\|\Omega_j^{-1/2} P_j^* q'_{j+1} |\omega_{j+1}|^{-1/2}\|_1 \leq \|M_{j+1}\|_1,$$

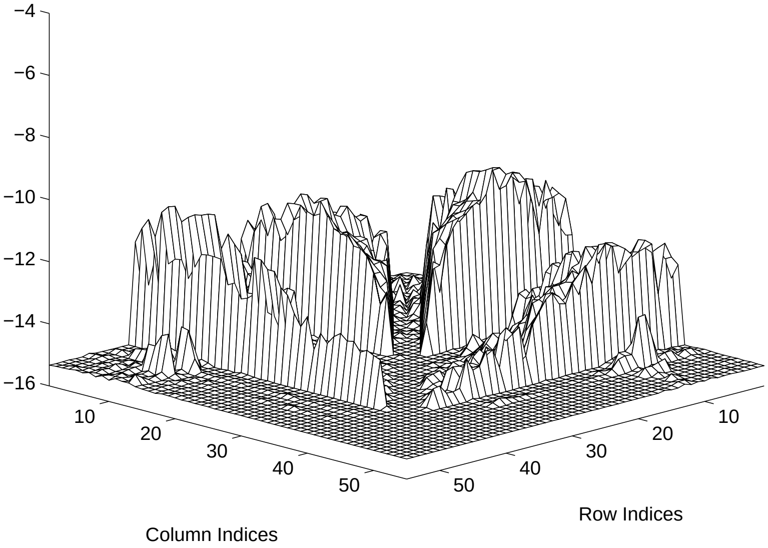

FIG. 2. M_{55} *(balanced op.);* $\|M_{55}\|_1 \approx 2\mathrm{e}-11$.

a necessary condition for (3.9) to hold at step $j+2$ without correction is that

$$\begin{aligned}\frac{\||\Omega_j|^{-1/2}P_j^* q'_{j+1}\|_1}{|\omega_{j+1}|^{1/2}}\||\Omega_{j+1}|^{-1/2}P_{j+1}^* q'_{j+2}\|_1 \\ \le \|M_{j+1}\|_1\||\Omega_{j+1}|^{-1/2}P_{j+1}^* q'_{j+2}\|_1 \le \epsilon.\end{aligned} \tag{3.10}$$

The square root of (3.10) yields

$$(\||\Omega_j|^{-1/2}P_j^* q'_{j+1}\|_1\||\Omega_{j+1}|^{-1/2}P_{j+1}^* q'_{j+2}\|_1)^{1/2} \le \epsilon^{1/2}|\omega_{j+1}|^{1/4},$$

and this is guaranteed by

$$\max(\||\Omega_j|^{-1/2}P_j^* q'_{j+1}\|_1, \||\Omega_{j+1}|^{-1/2}P_{j+1}^* q'_{j+2}\|_1) \le \epsilon^{1/2}|\omega_{j+1}|^{1/4}. \tag{3.11}$$

A similar argument applied to $(p_{j+1}^*)'$ yields our definition of semiduality.

DEFINITION 3.1. *Semiduality holds at step* $j+1$ *if for* $i \le j$,

$$\max(\|p_{i+1}^{'*}Q_i|\Omega_i|^{-1/2}\|_\infty, \||\Omega_i|^{-1/2}P_i^* q'_{i+1}\|_1) \le \epsilon^{1/2}\ |\omega_{i+1}|^{1/4}.$$

4. The Lanczos algorithm with correction. In this section, we present a practical and efficient implementation of the Lanczos algorithm with correction (LanCor hereafter). Several implementation details are addressed and relevant properties of the computed quantities are established. Extensive work from [7] on how to implement the Lanczos recurrences is summarized in section 4.1. In section 4.2 we discuss how to correct the duality loss and what effect this has on the computed quantities. An efficient implementation of correction steps called retroactive correction is discussed and justified in section 4.3. In section 4.4 we will show how to compute $(p'_{j+1})^* Q_j$ and $P_j^* q'_{j+1}$ without accessing P_j^* and Q_j and using only $\mathcal{O}(j)$ floating point operations and storage per step. LanCor is given in section 4.5.

4.1. Implementing the three-term recurrences. A prerequisite to the analyses of later sections is an understanding of how nearly dual consecutive pairs of left and right Lanczos vectors can be. We say that *local duality* holds at step j if

$$\max_{1<i\leq j}(|p_i^* q_{i-1}|, |p_{i-1}^* q_i|) \leq 4\epsilon. \tag{4.1}$$

In [7] it was proved that local duality is maintained to within a (theoretically necessary but generally unrealistic) factor of n by the implementation of the three-term recurrences below called LanLD. LanLD stands for the Lanczos algorithm maintaining local duality.

```
ALGORITHM (LANLD).
Start: $p_1 = p/\|p\|_2, q_1 = q/\|q\|_2, \omega_0 = 1, \beta_1 = \gamma_1 = 0, \omega_1 = p_1^* q_1$.
Iterate: For $i$=1,MaxStep
    1. $r_i^* = p_i^* B - \frac{\gamma_i \omega_i}{\omega_{i-1}} p_{i-1}^*, \quad s_i = Bq_i - q_{i-1}\frac{\beta_i \omega_i}{\omega_{i-1}}$
    2. $\alpha_i \in \{p_i^* s_i, r_i^* q_i\}$
    3. $r_i^* := r_i^* - \frac{\alpha_i}{\omega_i} p_i^*, \quad s_i := s_i - q_i \frac{\alpha_i}{\omega_i}$
    4. $\alpha_i^l = r_i^* q_i, \qquad \alpha_i^r = p_i^* s_i$ /* these are small corrections to $\alpha_i$ */
    5. $r_i^* := r_i^* - \frac{\alpha_i^l}{\omega_i} p_i^*, \quad s_i := s_i - q_i \frac{\alpha_i^r}{\omega_i}$
    6. $\beta_{i+1} = \|r_i^*\|_2, \quad \gamma_{i+1} = \|s_i\|_2$
    7. Check for invariant subspace. See equations (4.3) and (4.4).
    8. $p_{i+1}^* = r_i^*/\beta_{i+1}, \quad q_{i+1} = s_i/\gamma_{i+1}$
    9. $\omega_{i+1} = p_{i+1}^* q_{i+1}$
    10. Check for breakdown: $|\omega_{i+1}| < (n + 10(i+1))\epsilon$
    11. Check for convergence periodically (see section 2.2)
```

Remark 1. The meaning of step 2 is that α_i can be assigned either value $p_i^* s_i$ or $r_i^* q_i$; it does not matter which.

Remark 2. Local duality is maintained by steps 4 and 5.

The properties of the quantities computed by LanLD are summarized as follows. We assume that the no-breakdown condition holds,

$$\min_{1\leq i\leq j} |\omega_i| \geq (n + 10j)\epsilon, \tag{4.2}$$

and that the no-invariant subspace conditions hold: for $1 \leq i \leq j$,

$$\beta_{i+1} \geq \max(\sqrt{\epsilon}(\Phi_i + 1 + \psi)\|B\|_2, |\alpha_i^l/\omega_i|) \tag{4.3}$$

and

$$\gamma_{i+1} \geq \max(\sqrt{\epsilon}(\Phi_i + 1 + \psi)\|B\|_2, |\alpha_i^r/\omega_i|). \tag{4.4}$$

Here Φ_i is the growth factor defined by (2.7), and the constant ψ accounts for the rounding error introduced when the operator B is applied.

In this section, we add two more error bounds to our model of the computed quantities which are proved to be realistic in [7]. First, the Lanczos vectors satisfy the perturbed three-term recurrences

$$\beta_{j+1} p_{j+1}^* = p_j^* B - \frac{\alpha_j + \alpha_j^l}{\omega_j} p_j^* - \frac{\gamma_j \omega_j}{\omega_{j-1}} p_{j-1}^* - f_j^* \tag{4.5}$$

and

$$q_{j+1}\gamma_{j+1} = Bq_j - q_j \frac{\alpha_j + \alpha_j^r}{\omega_j} - q_{j-1}\frac{\beta_j \omega_j}{\omega_{j-1}} - g_j, \tag{4.6}$$

where the matrices $F_j = [f_1, \ldots, f_j]$ and $G_j = [g_1, \ldots, g_j]$ are such that

$$\max(\|F_j\|_2, \|G_j\|_2) \leq \epsilon(\Phi_j + 1)\|B\|_2. \tag{4.7}$$

Second, the tiny refinements to the trailing bits of each α_i to maintain local duality,

$$D_j^l = \operatorname{diag}(\alpha_i^l)_{i=1}^j \quad \text{and} \quad D_j^r = \operatorname{diag}(\alpha_i^r)_{i=1}^j, \tag{4.8}$$

satisfy

$$\max(\|D_j^l\|_2, \|D_j^r\|_2) \leq \epsilon(\Phi_j + 1)\|B\|_2. \tag{4.9}$$

4.2. Properties of the computed quantities. Recall from section 3 that the loss of duality of the candidate Lanczos vectors to the previous Lanczos vectors is corrected using a version of the GS process. Our model of the properties of the quantities computed by LanCor is obtained by amending the model for LanLD to account for correction steps.

To correct the loss of duality of the $(i+1)$st Lanczos vectors to the previous Lanczos vectors at the end of a Lanczos step we first compute

$$x_i^* = (p'_{i+1})^* Q_{i-1}, \quad y_i = P_{i-1}^* q'_{i+1}. \tag{4.10}$$

Next we "re-biorthogonalize" or correct the candidate Lanczos vectors:

$$(\breve{p}_{j+1})^* = (p'_{j+1})^* - x_i^* \Omega_{i-1}^{-1} P_{i-1}^* \tag{4.11}$$

and

$$\breve{q}_{j+1} = q'_{j+1} - Q_{i-1} \Omega_{i-1}^{-1} y_i. \tag{4.12}$$

Let $\mathcal{I}_j$ denote the set of all indices i up to and including j at which correction steps are taken, let e_i denote the ith column of the $j \times j$ identity matrix, and let

$$\Lambda_j = D_j^l + \sum_{i \in \mathcal{I}_j} \beta_{i+1} e_i (x_i^*, 0), \quad \Upsilon_j = D_j^r + \sum_{i \in \mathcal{I}_j} \begin{bmatrix} y_i \\ 0 \end{bmatrix} e_i^* \gamma_{i+1}. \tag{4.13}$$

Recall that D_j^l and D_j^r are defined in equation (4.8).

For the purpose of illustration, $T_{40} + \Upsilon_{40}$ corresponding to a model problem discussed in [25] is displayed on a semilog scale in Figure 3.

The governing equations for LanCor are

$$P_j^* B = (T_j + \Lambda_j)\Omega_j^{-1} P_j^* + e_j \beta_{j+1} p_{j+1}^* + F_j^* \tag{4.14}$$

and

$$B Q_j = Q_j \Omega_j^{-1} (T_j + \Upsilon_j) + q_{j+1} \gamma_{j+1} e_j^* + G_j. \tag{4.15}$$

The matrices Υ_j (for upper) and Λ_j (for lower) are upper and lower triangular matrices of spikes, one spike for each correction step. Local duality, (4.1), (4.7), and

$$\max(\|\Lambda_j \Omega_j^{-1/2}\|_2, \|\Omega_j^{-1/2} \Upsilon_j\|_2) \leq \sqrt{\epsilon}(\Phi_j + 1)\|B\|_2 \tag{4.16}$$

are also realistic for LanCor [7].

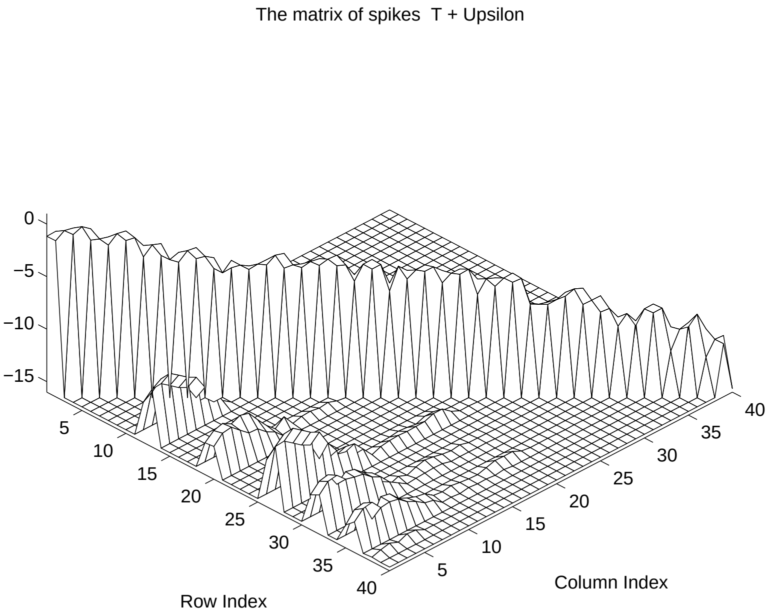

FIG. 3. $T_{40} + \Upsilon_{40}$.

4.3. Retroactive correction. In this section we show how to implement a correction step,

$$p'_{j+1} \to \breve{p}_{j+1}, \quad q'_{j+1} \to \breve{q}_{j+1}.$$

As in the symmetric case, correction steps are taken in pairs,

$$p'_{j+1} \to \breve{p}_{j+1}, \quad q'_{j+1} \to \breve{q}_{j+1},$$

$$p'_{j} \to \breve{p}_{j}, \quad q'_{j} \to \breve{q}_{j}.$$

The reason for correcting the jth Lanczos vectors with the $j+1$th is the same as in the symmetric case, and for the convenience of the reader we revisit the explanation. In practice the loss of duality is gradual,

$$\|P^*_{j-1} q_j\| \approx \|P^*_{j-1} q_{j+1}\|,$$

and thus $\|P^*_{j-1} q_j\|$ is less than but approximately equal to the semiduality threshold. Consider the $j+2$th Lanczos vector

$$q_{j+2}\gamma_{j+2} = Bq_{j+1} - q_{j+1}\frac{\alpha_{j+1}}{\omega_{j+1}} - q_j \frac{\beta_{j+1}\omega_{j+1}}{\omega_j}.$$

Multiply by P^*_{j-1} to obtain

$$P^*_{j-1} q_{j+2}\gamma_{j+2} = P^*_{j-1} B q_{j+1} - P^*_{j-1} q_{j+1}\frac{\alpha_{j+1}}{\omega_{j+1}} - P^*_{j-1} q_j \frac{\beta_{j+1}\omega_{j+1}}{\omega_j}.$$

If the jth Lanczos vectors are not corrected along with the $j+1$th Lanczos vectors, then

$$\|P_{j-1}^* q_{j+2}\|\gamma_{j+2} \approx \|P_{j-1}^* q_j\| \frac{\beta_{j+1}|\omega_{j+1}|}{|\omega_j|},$$

which is often just below the semiduality threshold. Correcting the jth Lanczos vectors with the $j+1$th substantially reduces $\|P_{j-1}^* q_{j+2}\|$, and this postpones the next correction step.

Correction steps are implemented so that each Lanczos vector is transferred from slow storage to fast storage and back again only once. Following [24], we call this *retroactive* correction. That is, correcting the jth Lanczos vectors with the $j+1$th doubles the number of floating point operations per correction step, but the amount of data transfer is the same. Retroactive correction by the two-sided modified GS algorithm is implemented as follows.

```
Algorithm (retroactive correction).
Iterate: For $i = 1 \ldots j-1$,
        1. $p_{j+1} := p_{j+1} - p_i(\omega_i^{-*}(q_i^* p_{j+1}))$
        2. $p_j := p_j - p_i(\omega_i^{-*}(q_i^* p_j))$
        3. $q_{j+1} := q_{j+1} - q_i(\omega_i^{-1}(p_i^* q_{j+1}))$
        4. $q_j := q_j - q_i(\omega_i^{-1}(p_i^* q_j))$
Recover local duality
        1. $p_{j+1} := p_{j+1} - p_j(\omega_j^{-*}(q_j^* p_{j+1}))$
        2. $q_{j+1} := q_{j+1} - q_j(\omega_j^{-1}(p_j^* q_{j+1}))$
```

Retroactive correction changes the relations among the computed quantities. Nonetheless, a careful analysis shows that as long as semiduality is maintained properties (4.1), (4.7), and (4.16) are also realistic for LanCor with retroactive correction [7]. It is important *not* to normalize p_j and q_j after a retroactive correction step. The reason is that normalizing p_j and q_j in this case nonnegligibly alters β_j and γ_j. This needlessly complicates the approximate three-term recurrences among the computed quantities.

4.4. Monitoring the loss of duality. We must correct for the loss of duality when either $p_{j+1}^* Q_j |\Omega_j^{-1/2}|$ or $|\Omega_j^{-1/2}| P_j^* q_{j+1}$ increases to $\sqrt{\epsilon}|\omega_{j+1}^{1/4}|$. To compute these vectors at each step is about as costly as correcting the loss of duality at each step. The same problem arises in the symmetric case. Compromise symmetric Lanczos algorithms avoid this costly step by updating a recurrence estimating the loss of orthogonality at each step [11, 19, 20, 24, 28, 29]. In this section we extend the partial reorthogonalization (PRO) algorithm from the symmetric Lanczos algorithm [28, 29]. Recurrence relations for $p_{j+1}^* Q_j$ and $P_j^* q_{j+1}$ for the unnormalized two-sided Lanczos algorithm based on a model of the properties of the quantities computed in finite precision arithmetic appear in [36].

The sequence of vectors $(p_{i+1}^* Q_i)$ and $(P_i^* q_{i+1})$ satisfy a three-term recurrence which we now derive. Let $\omega_{i,j} = p_i^* q_j$. In this notation, $\omega_i = \omega_{i,i}$.

Suppose that a correction step is not taken at step j (i.e., in computing the $(j+1)$st Lanczos vectors). Multiplying equation (4.5) by Q_j and substituting BQ_j according to (4.15), we have

$$\beta_{j+1} p_{j+1}^* Q_j = p_j^* Q_j \left[\Omega_j^{-1}(T_j + \Upsilon_j) - \frac{\alpha_j + \alpha_j^l}{\omega_j} I \right]$$

$$-\frac{\gamma_j\omega_j}{\omega_{j-1}}p_{j-1}^*Q_j + \omega_{j,j+1}\gamma_{j+1}e_j^* + p_j^*G_j - f_j^*Q_j. \tag{4.17}$$

Similarly multiplying (4.6) by P_j^* on the left and substituting in P_j^*B according to (4.14), we have

$$\begin{aligned} P_j^*q_{j+1}\gamma_{j+1} = &\left[(T_j+\Lambda_j)\Omega_j^{-1} - \frac{\alpha_j+\alpha_j^r}{\omega_j}I\right]P_j^*q_j \\ &-\frac{\beta_j\omega_j}{\omega_{j-1}}P_j^*q_{j-1} + e_j\beta_{j+1}\omega_{j+1,j} + F_j^*q_j - P_j^*g_j. \end{aligned} \tag{4.18}$$

To further reduce these equations, we first need to discuss some additional relations among the computed quantities. First we show that the correction terms

$$p_j^*Q_j\Omega_j^{-1}\Upsilon_j \quad \text{and} \quad \Lambda_j\Omega_j^{-1}P_j^*q_j$$

negligibly affect the loss of duality among the computed Lanczos vectors. For this reason, these matrices are not used to estimate the loss of duality and are not stored. Since j is not a correction step, equation (4.13) implies that

$$\Lambda_j\Omega_j^{-1}p_j^*q_je_j = D_j^le_j = e_j\alpha_j^l.$$

Substitute the definition of semiduality, (3.3), and (4.16) to find

$$\left\|\Lambda_j\Omega_j^{-1}\begin{bmatrix} P_{j-1}^*q_j \\ 0 \end{bmatrix}\right\|_2 \le \|\Lambda_j\Omega_j^{-1/2}\|_2\|\Omega_{j-1}^{-1/2}P_{j-1}^*q_j\|_2 \le (\Phi_j+1)\epsilon\|B\|_2.$$

The analysis of $p_j^*Q_j\Omega_j^{-1}\Upsilon_j$ is similar.

Next we expand terms such as $P_j^*q_j$:

$$p_j^*Q_j = (p_j^*Q_{j-1},0) + \omega_je_j^* \quad \text{and} \quad P_j^*q_j = \begin{bmatrix} P_{j-1}^*q_j \\ 0 \end{bmatrix} + \omega_je_j. \tag{4.19}$$

By equation (4.19) and the definition of T_j, we have

$$\begin{aligned} &p_j^*Q_j\left(\Omega_j^{-1}T_j - \frac{\alpha_j+\alpha_j^l}{\omega_j}I\right) \\ &\quad = (p_j^*Q_{j-1},0)\left[\Omega_j^{-1}T_j - \frac{\alpha_j+\alpha_j^l}{\omega_j}I\right] + \gamma_j\omega_je_{j-1}^t - \alpha_j^le_j^t \end{aligned} \tag{4.20}$$

and

$$\begin{aligned} &\left(T_j\Omega_j^{-1} - \frac{\alpha_j+\alpha_j^r}{\omega_j}I\right)P_j^*q_j \\ &\quad = \left[T_j\Omega_j^{-1} - \frac{\alpha_j+\alpha_j^r}{\omega_j}I\right]\begin{bmatrix} P_{j-1}^*q_j \\ 0 \end{bmatrix} + e_{j-1}\beta_j\omega_j - e_j\alpha_j^r. \end{aligned} \tag{4.21}$$

Since step j is not a correction step, the last row of Υ_j and the last column of Λ_j are zero. That is why they do not appear.

We also need the identities

$$p_{j-1}^* Q_j = (p_{j-1}^* Q_{j-2}, 0, 0) + \omega_{j-1} e_{j-1}^* + \omega_{j-1,j} e_j^* \tag{4.22}$$

and

$$P_j^* q_{j-1} = \begin{bmatrix} P_{j-2}^* q_{j-1} \\ 0 \\ 0 \end{bmatrix} + \omega_{j-1} e_{j-1} + \omega_{j,j-1} e_j . \tag{4.23}$$

Finally, substitute equations (4.22) and (4.20) into equation (4.17) and equations (4.23) and (4.21) into equation (4.18) and the desired recurrences appear:

$$\begin{aligned} \beta_{j+1} p_{j+1}^* Q_j = (p_j^* Q_{j-1}, 0) &\left(\Omega_j^{-1} T_j - \frac{\alpha_j}{\omega_j} I \right) \\ &- \frac{\gamma_j \omega_j}{\omega_{j-1}} ((p_{j-1}^* Q_{j-2}, 0, 0) + \omega_{j-1,j} e_j^t) \\ &+ (\omega_{j,j+1} \gamma_{j+1} + \alpha_j^r - \alpha_j^l) e_j^t + \mathcal{O}(\epsilon(\Phi_j + 1)\|B\|_2) \end{aligned} \tag{4.24}$$

and

$$\begin{aligned} P_j^* q_{j+1} \gamma_{j+1} = &\left(T_j \Omega_j^{-1} - \frac{\alpha_j}{\omega_j} I \right) \begin{bmatrix} P_{j-1}^* q_j \\ 0 \end{bmatrix} \\ &- \frac{\beta_j \omega_j}{\omega_{j-1}} \left[\begin{bmatrix} P_{j-2}^* q_{j-1} \\ 0 \\ 0 \end{bmatrix} + \omega_{j,j-1} e_j \right] \\ &+ e_j (\beta_{j+1} \omega_{j+1,j} + \alpha_j^l - \alpha_j^r) + \mathcal{O}(\epsilon(\Phi_j + 1)\|B\|_2). \end{aligned} \tag{4.25}$$

Note that certain computable terms such as $\omega_{j,j-1}$ and α_j^l which are $\mathcal{O}(\epsilon(\Phi_j + 1)\|B\|_2)$ are not included in $\mathcal{O}(\epsilon(\Phi_j + 1)\|B\|_2)$. This is done because these computable quantities are used to estimate $P_j^* q_{j+1}$ and $p_{j+1}^* Q_j$ in the next section.

4.4.1. An implementation of the monitoring algorithm. LanCor is similar to LanLD, but with additional work done (if necessary) between LanLD iterations to maintain semiduality. The perturbed recurrences (4.24) and (4.25) are invoked to compute $\beta_{j+1} p_{j+1}^* Q_j$ and $P_j^* q_{j+1} \gamma_{j+1}$ after β_{j+1} and γ_{j+1} are computed as in step 6 of LanLD. The decision whether or not to correct duality loss is then made as determined in section 3.1. In this section we show how to implement the recurrences to obtain accurate estimates of the duality loss.

In LanCor at each Lanczos step j the candidate $j+1$th Lanczos vectors are explicitly "dualized" against the $j-1$th Lanczos vectors (extended local duality) and then the jth Lanczos vectors (local duality).

$P_j^* q_{j+1}$ is estimated by h_{j+1}. Initially h_2 and h_3 are exact, and for $j > 2$,

$$h_{j+1} \gamma_{j+1} = \left(T_j \Omega_j^{-1} - \frac{\alpha_j}{\omega_j} I \right) \begin{bmatrix} h_j \\ 0 \end{bmatrix} - \begin{bmatrix} h_{j-1} \\ 0 \\ 0 \end{bmatrix} \frac{\beta_j \omega_j}{\omega_{j-1}} - e_{j-2} \alpha_j^r . \tag{4.26}$$

To account for the perturbation of the three-term recurrences by correction steps, $\epsilon \mathrm{diag}|\Omega_j^{-1} T_j|$ is added to the right-hand side above if the loss of the duality of q_{j-1} and q_j was corrected. The $j-1$ and j entries of the estimate h_{j+1} are assigned the exact values $p_{j-1}^* q_{j+1}$ and $p_j^* q_{j+1}$. Maintaining extended local duality sweeps the α_j^r term from the jth entry to the $j-2$th entry, hence the $e_{j-2} \alpha_j^r$ above. The estimate of $p_{j+1}^* Q_j$ is computed by the similar recurrence.

4.5. The implementation of LanCor. In this section we summarize the implementation of LanCor, the Lanczos algorithm maintaining semiduality.

ALGORITHM (LanCor).

```
Start: $p_1 = p/\|p\|_2, q_1 = q/\|q\|_2, \omega_0 = 1, \beta_1 = \gamma_1 = 0, \omega_1 = p_1^* q_1$.
Iterate: For $i$=1,MaxStep
    1. $r_i^* = p_i^* B - \frac{\gamma_i \omega_i}{\omega_{i-1}} p_{i-1}^*$,    $s_i = Bq_i - q_{i-1} \frac{\beta_i \omega_i}{\omega_{i-1}}$
    2. $\alpha_i = r_i^* q_i$
    3. $r_i^* = r_i^* - \frac{\alpha_i}{\omega_i} p_i^*$,    $s_i = s_i - q_i \frac{\alpha_i}{\omega_i}$
    4. Maintain extended local duality (see section 4.4.1)
    5. $\alpha_i^l = r_i^* q_i$,        $\alpha_i^r = p_i^* s_i$
    6. $r_i^* := r_i^* - \frac{\alpha_i^l}{\omega_i} p_i^*$,    $s_i := s_i - q_i \frac{\alpha_i^r}{\omega_i}$
    7. $\beta_{i+1} = \|r_i^*\|_2$,    $\gamma_{i+1} = \|s_i\|_2$
    8. Check for invariant subspace. See equations (4.3) and (4.4).
    9. $p_{i+1}^* = r_i^* / \beta_{i+1}$,    $q_{i+1} = s_i / \gamma_{i+1}$
   10. $\omega_{i+1} = p_{i+1}^* q_{i+1}$
   11. Check for breakdown: $|\omega_{i+1}| < (n + 10(i+1))\epsilon$
   12. Monitor duality loss (see section 4.4.1)
   13. Correct duality loss only if necessary (see section 4.3)
   14. Check for convergence after a correction step only (see section 2.2)
```

Remark 3. The loss of duality among the computed Lanczos vectors corresponds to either a near breakdown of the algorithm or the convergence of a Ritz value to an eigenvalue of B [1]. For this reason it is more efficient to check for convergence only after correction steps.

5. Preserved quantities.

LanCor applied to an operator B after j successful steps yields matrices P_j^* and Q_j of Lanczos vectors and the reduced tridiagonal–diagonal pencil (T_j, Ω_j). In this section we compare the computed quantities to the corresponding exact quantities determined by B, the row span of P_j^*, and the column span of Q_j. We say that a computed quantity is preserved when it is as close to the corresponding exact quantity as the data warrants.

Our main result is that semiduality suffices to preserve (T_j, Ω_j). See section 5.3.

The analysis is more complicated than in the symmetric case. We must avoid perturbations that are proportional to $\|\Omega_j^{-1}\|_2 = 1/\min_{1 \le i \le j} |\omega_i|$.

5.1. Exact projections. The operator

$$\Pi_j = Q_j (P_j^* Q_j)^{-1} P_j^* \tag{5.1}$$

is the oblique projection corresponding to the computed Lanczos vectors. The projection of B onto the spaces spanned by the Lanczos vectors is the matrix $\Pi_j B \Pi_j$. See Definition 2.1. The Lanczos vectors are dual if and only if $P_j^* Q_j$ is diagonal and nonsingular.

We recover exactly dual bases corresponding to the computed Lanczos vectors by use of the LDU factorization of $P_j^* Q_j$:

$$P_j^* Q_j = L_j \hat{\Omega}_j U_j. \tag{5.2}$$

Recall that Ω_j =diag$(P_j^* Q_j)$. If $\hat{\Omega}_j$ is nonsingular, then substitute (5.2) into (5.1) to obtain

$$\Pi_j = Q_j U_j^{-1} \ \hat{\Omega}_j^{-1} \ L_j^{-1} P_j^*. \tag{5.3}$$

The rows of

$$\hat{P}_j^* = L_j^{-1} P_j^* \tag{5.4}$$

and the columns of

$$\hat{Q}_j = Q_j U_j^{-1} \tag{5.5}$$

are dual since

$$\hat{P}_j^* \hat{Q}_j = L_j^{-1} P_j^* Q_j U_j^{-1} = \hat{\Omega}_j.$$

Two-sided GS applied to P_j^* and Q_j yields $\hat{P}_j^*$ and $\hat{Q}_j$. Next define $\hat{T}_j$ by

$$\hat{T}_j = \hat{P}_j^* B \hat{Q}_j. \tag{5.6}$$

Note that $\hat{T}_j$ is *not* tridiagonal. The representation of $\Pi_j B \Pi_j$ with respect to the bases $\hat{\Omega}_j^{-1} \hat{P}_j^*$ and $\hat{Q}_j$ is $\hat{\Omega}_j^{-1} \hat{T}_j$. This matrix is equivalent to the pencil $(\hat{T}_j, \hat{\Omega}_j)$. This pencil is analogous to the orthogonal projection of the operator onto the span of the computed Lanczos vectors in the symmetric case.

5.2. Conditions for preservation. Each computed Lanczos vector is the sum of the vector which exactly satisfies a three-term recurrence and another vector whose norm is proportional to the machine epsilon ϵ. See equations (4.5) and (4.6). This result is typical of Krylov subspace methods [20]. This perturbation of the Lanczos vectors causes perturbations of the diagonal elements of Ω by approximately ϵ and perturbations of the tridiagonal elements of T by approximately $\epsilon\|B\|$. We show that exactly correcting the loss of duality among the computed Lanczos vectors does not change the pencil (T, Ω) by significantly more than these amounts. We call this property the preservation of T and Ω. The elements of T and Ω are not in general determined to working (or full relative) precision. This implies that $T\Omega^{-1}$ and $\Omega^{-1}T$ are not determined to full absolute precision.

This section addresses the problem of determining necessary and sufficient conditions for three properties of the computed quantities to hold. The three properties are (1) that $W = P^*Q$ admits an LDU factorization, (2) that the diagonal matrix D is approximately Ω, and (3) that L and U are well conditioned. To be precise, we determine realistic sufficient conditions for any complex $n \times n$ matrix W with nonzero diagonal elements to admit an LDU factorization

$$W = LDU \tag{5.7}$$

such that

$$\|\text{diag}(W) - D\|_2 \leq 2\epsilon \tag{5.8}$$

and

$$\max(\|L^{-1}\|_2, \|U^{-1}\|_2) < 2. \tag{5.9}$$

In our case $W = P^*Q$. By equation (5.2) $D = \hat{\Omega}$ holds and (5.8) immediately implies the preservation of Ω. The preservation of T is discussed in section 5.3.

Our results are given in the two theorems below. Theorem 5.1 gives necessary and sufficient conditions for (5.7) and (5.8) to hold. Theorem 5.2 gives sufficient conditions for all three properties to hold which are only slightly stronger than the hypotheses of Theorem 5.1 (i.e., the lower bound on $|\omega_j|$ increases from $2j\epsilon$ to $10j\epsilon$). We ultimately increase the latter lower bound by $n\epsilon$ to $(n+10j)\epsilon$ to account for the discrepancy between the computed and exact values of $P_j^*Q_j$.

For any matrix C let $triu'(C)$ denote its strictly upper triangular part.

THEOREM 5.1. *Let W be a $j\times j$ complex matrix and let $\Omega = \mathrm{diag}(W) = \mathrm{diag}(\omega_i)$. Suppose that $\epsilon > 0$, $j > 2$, $(j-2)\epsilon < 1$, and W satisfies the following hypotheses:*

$$\min_{1\le i\le j} |\omega_i| \ge 2(j-2)\epsilon, \tag{5.10}$$

$$\max(\|\Omega^{-1/2} triu'(W)\|_1, \|\Omega^{-1/2} triu'(W^*)\|_1) \le \sqrt{\epsilon}. \tag{5.11}$$

Then equations (5.7) *and* (5.8) *hold.*

Proof. See [7].

Remark 4. The second hypothesis (5.11) is equivalent to the semiduality condition (3.3). Note that the factor of $|\omega_{i+1}|^{1/4}$ that appears in Definition 3.1 to maintain the viability of the two-sided GS process is not necessary to ensure the preservation of (T,Ω) in Theorems 5.1 and 5.2.

One approach to proving Theorem 5.1 is to apply the perturbation theory for Gaussian elimination. Many papers have recently appeared on this subject [2, 33, 34]. To guarantee condition (5.8), all of these general perturbation bounds require significantly stronger hypotheses than Theorem 5.1.

Theorem 5.1 gives necessary and sufficient conditions for the preservation of Ω (i.e., condition (5.8)). By increasing the lower bound on $|\omega_j|$ by a factor of 5, we will show that the computed quantities are preserved. Due to the difficulty of this task, we must be satisfied with unachievable but realistic bounds.

THEOREM 5.2. *Let W be a $j\times j$ complex matrix and let $\Omega = \mathrm{diag}(W) = \mathrm{diag}(\omega_i)$. Suppose that $\epsilon > 0$, $j > 2$, $(j-2)\epsilon < 1$, and Ω is nonsingular. Suppose in addition that W satisfies the hypothesis* (5.11)*, and that*

$$\min |\omega_i| \ge 10j\epsilon. \tag{5.12}$$

Then equations 5.7 *to* 5.9 *hold.*

Proof. See [7].

The idea of the proofs is to decompose W into the sequence of extensions

$$W_{i+1} = \begin{pmatrix} W_i & y_i \\ x_i^t & \omega_{i+1} \end{pmatrix}. \tag{5.13}$$

We define the sequence $\{\kappa_i\}_{i=1}^j$ corresponding to a norm $\|.\|$ by

$$\max(\|x_i\|, \|y_i\|) = \kappa_i. \tag{5.14}$$

As in [17], we then extract the worst-case information corresponding to $\{\kappa_i\}_{i=1}^j$.

5.3. Preservation of T. The pencil computed by the three-term recurrences can eventually become of larger order than the original matrix and clearly differs from the one that would be produced in exact arithmetic, but this can never happen if semiduality holds. In this section, we show that if semiduality is maintained then the computed pencil (T_j, Ω_j) agrees with the exact oblique projection of the spans of the computed left and right Lanczos vectors to full absolute precision.

Correcting the loss of duality of the $(j+1)$st Lanczos vectors to the previous Lanczos vectors replaces the vectors computed by the three-term recursion with (approximations of) $\hat{p}_{j+1}$ and $\hat{q}_{j+1}$, where $\hat{p}_{j+1}$ denotes column $j+1$ of $\hat{P}_k$ and $\hat{q}_{j+1}$ denotes column $j+1$ of $\hat{Q}_k$. The corresponding elements of T and Ω change to (approximations of) the corresponding elements of the dense matrix $\hat{T}$ and the diagonal matrix $\hat{\Omega}$. We want to know how large this perturbation is and in particular when it is negligible.

The following theorem established the preservation of T for LanCor. Recall that Theorem 5.1 establishes the preservation of Ω. The proof uses the properties of the computed quantities established in section 4 and this section. The hypotheses of Theorem 5.2 are the no-breakdown and no-invariance conditions from section 4.1 and the semiduality condition; for $1 \leq i \leq j$,

$$\max(\||\Omega_i|^{-1/2} P_i^* q_{i+1}\|_1, \||\Omega_i|^{-1/2} Q_i^* p_{i+1}\|_1) \leq \sqrt{\epsilon}. \tag{5.15}$$

THEOREM 5.3. *Let B be a complex $n \times n$ matrix and let P_j^* and Q_j be the matrices of Lanczos vectors computed by LanCor. Let T_j denote the computed tridiagonal and let $\hat{T}_j$ be defined as in equation* (5.6). *If $\Omega_j = \mathrm{diag}(P_j^* Q_j)$ satisfies the no-breakdown condition* (4.2) *the no-invariant subspace conditions* (4.3) *and* (4.4) *and semiduality holds* (*see* (5.15)), *then for Φ_j defined in equation* (2.7)

$$\|\hat{T}_j - T_j\|_2 = \mathcal{O}(j(\Phi_j + 1)\epsilon \|B\|_2)$$

holds.

Proof. See [7].

6. Numerical experiments.

The Lanczos algorithm maintaining local duality only (LanLD), semiduality (LanCor), and full duality (LanFRB) have been applied to many tasks. We present the results for one representative example here. All computations were done in MATLAB on an IBM Power Workstation with machine precision $\epsilon \approx 2 \;\; 10^{-16} = 2\mathrm{e}-16$.

6.1. The Tolosa matrix.

We illustrate the properties of LanCor using the Tolosa matrix A of order $n = 2000$ from the Harwell Boeing sparse matrix collection. The computational task is to compute the largest eigenvalues of A to half precision. We choose to compute the 50 largest eigenvalues because this emphasizes the difference between LanLD and LanCor. A has 5184 nonzero entries and $\|A\|_1 \approx 1\mathrm{e}+6.8$. Since A averages less than three nonzeros per row, the inner products in the three-term recurrences cost nearly as much as the matrix–vector multiplications in terms of floating point operations.

The eigenvalue problem for A is known to be ill conditioned and A is known to possess multiple eigenvalues [26]. For theoretical purposes, we computed the eigenvalues of A by the QR algorithm and observed that the spectral radius of A is approximately $1\mathrm{e}+3.4$. For this reason, the MATLAB function `balance`() was applied to A to obtain a balanced matrix B diagonally similar to A. For this B $\|B\|_1 \approx \|B\|_\infty \approx 1\mathrm{e}+4.0$ holds. Since $\|B\|_2^2 \leq \|B\|_1 \|B\|_\infty$, balancing yields a matrix whose Euclidean norm is within a factor of 4 of its spectral radius. QR applied to B computes three real eigenvalues— -12.098, -24.196, and -36.294—of multiplicity 382; the remaining eigenvalues are distinct and well separated. Though the eigenvalues of A and B are the same (barring underflow), the computed eigenvalues of A and B by QR (without

Table 1

Correction step	238	264	283	295	310	323	333	347	363
No. eigenvalues	14	22	26	28	34	38	42	48	56

balancing) agree to from full to half relative precision. We will compare the eigenvalues of B computed by the QR algorithm to the three implementations of the Lanczos algorithm (LanFRB, LanCor, and LanLD) and Arnoldi's method.

We did not compare the Lanczos algorithm to the implicitly restarted Arnoldi iteration [30, 15]. Arnoldi's method does establish a lower bound for the number of steps required by implicitly restarted Arnoldi iteration. Implicit restarts can be incorporated into the Lanczos algorithm as well as Arnoldi's method [12].

6.2. Results. All three implementations of the Lanczos algorithm computed the requested 50 eigenvalues to the same high accuracy. For Arnoldi's method, LanFRB, and LanLD the Ritz values are checked every 50 iterations. In each case $p_1 = q_1$ is the same random vector (normal distribution). LanFRB and LanCor have identical convergence properties; this is a consequence of the preservation of the pencil (T, Ω) (see section 5). The reward for maintaining semiduality is that fewer Lanczos steps are required to complete the given task. In this case, LanFRB and LanCor required 400 and 363 Lanczos steps, respectively, while LanLD and Arnoldi's method required 450 and 400, respectively. Because convergence is checked after correction steps instead of periodically in LanCOR, fewer Lanczos steps are required than for LanFRB in this experiment.

No copies of converged eigenvalues appear among the Ritz values when semi- or full duality is maintained, but copies do appear among the Ritz values computed by LanLD.

LanCor takes 16 correction steps to compute the requested eigenvalues, 1/25th as many as LanFRB. Table 1 gives the last 9 steps at which the duality loss is corrected in LanCor and the number of converged Ritz values at that step.

For comparison we applied Arnoldi's method with modified GS orthogonalization to this task and computed eigenvalues of B to the same accuracy [27]. The number of converged Ritz values near the end of the run are displayed in Table 2.

Table 2

Arnoldi step	50	100	150	200	250	300	350	400
No. eigenvalues	0	0	0	6	16	30	48	70

In this example Arnoldi's method, LanFRB, and LanCor yield approximate eigenvalues of similar accuracy for a fixed number of steps; the differences in the number of steps is due to the convergence criteria.

LanFRB, LanCor, and LanLD were each stable in the sense that

$$\|\Omega^{-1}T\|_1 \approx 35\|B\|_2,$$

even though $\min|\omega_i| \approx 2\mathrm{e}-5$ (see section 2.3).

The $\log_{10}$ of the number of floating point operations (flops) in these Lanczos and Arnoldi runs are tabulated in Table 3. The flop count for applying the operator (a sparse matrix–vector multiplication in this case) is given in column OP; column EIG displays the flop count for solving the reduced eigenvalue problems by

Table 3

Results for balanced Tolosa.

FLOPS ($\log_{10}$)	OP	EIG	(BI-) ORTH	ALGO	TOTAL
Arnoldi (400)	6.6	9.5	8.8	6.9	9.6
LanFRB (400)	7.0	7.7	**9.1**	7.7	9.1
LanCor (363)	7.0	7.8	**8.1**	7.8	8.4
LanLD (450)	7.0	7.8	None	8.0	8.2

the QR algorithm for Arnoldi's method and by the differential QD algorithm, an algorithm that exploits the tridiagonal structure for the Lanczos-based procedures [8]. (BI-)ORTH gives the flop count for maintaining the duality or orthogonality of the basis vectors, and column ALGO contains the remaining flop count. The number of steps required by each method is given in parenthesis below the algorithm name.

We were surprised at the large number of flops required by the QR algorithm in Arnoldi's method in this example. For comparison note that computing the eigenvalues of B by the QR algorithm requires $1\mathrm{e}+11$ flops.

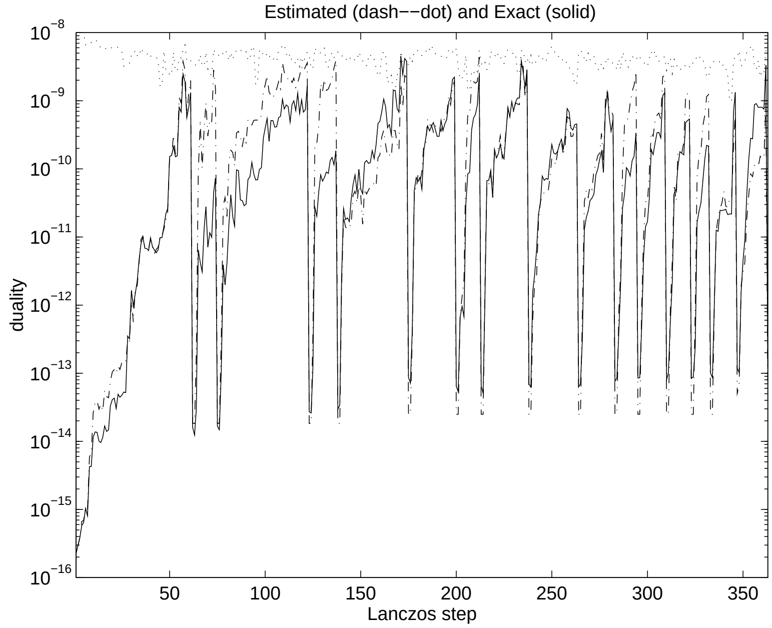

Fig. 4. *Numerical duality for LanCor applied to Tolosa matrix.*

Maintaining semiduality requires an order of magnitude fewer flops than full duality. Also, LanCor has an order of magnitude fewer flops than Arnoldi's method.

LanLD requires the fewest flops and takes the most steps. The low flop count is due to the less rigorous stopping criteria. The error introduced by accepting a Ritz value as an eigenvalue is estimated by the minimum of three quantities: the distance from the Ritz value to the nearest remaining Ritz value and the left and the right unnormalized residuals. Recall from section 2.2 that if v is an eigenvector of $\Omega^{-1}T$, then Qv is used to approximate an eigenvector of B, and reliable accuracy estimates must factor in the shrinkage $\|Qv\|_2/\|v\|_2$. We observed shrinkage, i.e., $\|Qv\|_2/\|v\|_2 \approx .01$, for all the Ritz vectors of interest in this example. For this reason the error estimates based on unnormalized Ritz vectors are 100 times too small. In LanLD the Lanczos vectors are not stored and so the shrinkage of the Ritz vectors is not available. Even if the Lanczos vectors are stored, forming the eigenvectors requires 1e + 8.6 real floating point operations (see section 2.2). That is, if we demand reliability from LanLD similar to that of LanCor, the LanLD flop count will increase above the LanCor flop count.

We conclude by illustrating the effectiveness of the duality-monitoring algorithm of section 4.4.1. Figure 4 compares the $\log_{10}$ of our estimate of

$$\max(\||\Omega_i|^{-1/2}P_i^* q_{i+1}\|_1, \||\Omega_i|^{-1/2}Q_i^* p_{i+1}\|_1) \tag{2}$$

at each step (dash–dot line) i to the exact value (solid line). Each spike indicates a correction step. The dotted line across the top of the figure is the target threshold $\epsilon^{1/2}|\omega_i|^{1/4}$.

REFERENCES

[1] Z. Bai (1992), *Error analysis of the Lanczos algorithm for the nonsymmetric eigenvalue problem*, Math. Comp., 62, pp. 209–226.

[2] A. Barrlund (1991), *Perturbation bounds for the LDL^H and LU decompositions*, BIT, 31, pp. 358–363.

[3] Z. Bai, D. Day, and Q. Ye (1995), *ABLE: An Adaptive Block Lanczos Method of the Eigenvalue Problem*, Technical report 95-07, Mathematics Department, University of Kentucky, Lexington, KY.

[4] D. Boley and G. Golub (1990), *The nonsymmetric Lanczos algorithm and controllability*, Systems Control Lett., 16, pp. 97–105.

[5] J. Cullum and R. Willoughby (1985), *Lanczos Algorithms for Large Symmetric Eigenvalue Computations*, Birkhäuser Boston, Cambridge, MA.

[6] J. Cullum, W. Kerner, and R. Willoughby (1989), *A generalized nonsymmetric Lanczos procedure*, Comp. Phys. Comm., 53, pp. 19–48.

[7] D. Day (1993), *Semi-Duality in the Two-Sided Lanczos Algorithm*, Ph.D. thesis, University of California, Berkeley, CA.

[8] D. Day (1995), *The differential QD algorithm for the tridiagonal eigenvalue problem*, manuscript.

[9] T. Ericsson and A. Ruhe (1980), *The spectral transformation Lanczos method for the numerical solution of large sparse generalized symmetric eigenvalue problems*, Math. Comp., 35, pp. 1251–1268.

[10] R. Freund, M. Gutknecht, and N. Nachtigal (1993), *An implementation of the look-ahead Lanczos algorithm for non-Hermitian matrices*, SIAM J. Sci. Stat. Comput., 14, pp. 137–158.

[11] R. Grimes, J. Lewis, and H. Simon (1994), *A shifted block Lanczos algorithm for solving generalized eigenproblems*, SIAM J. Matrix Anal. Appl., 15, pp. 228–272.

[12] E. Grimme, D. Sorensen, and P. Van Dooren (1994), *Model Reduction of State Space Systems via an Implicitly Restarted Lanczos Method*, TR94-21, Rice University, Houston, TX.

[13] M. Gutknecht (1992), *A completed theory of the unsymmetric Lanczos process and related algorithms*, SIAM J. Matrix Anal. Appl., Part I, 13, pp. 594–639, Part II, 15, pp. 15–58.

[14] W. Kahan, B. Parlett, and E. Jiang (1982), *Residual bounds on approximate eigensystems of nonnormal matrices*, SIAM J. Numer. Anal., 19, pp. 470–484.

[15] R. Lehoucq (1995), *Analysis and Implementation of an Implicitly Restarted Arnoldi Iteration*, TR95-13, Rice University, Houston, TX.

[16] C. Lanczos (1950), *An iteration method for the solution of the eigenvalue problem of linear differential and integral operators*, J. Res. Nat. Bur. Stand., 45, pp. 255–282.

[17] A. Ostrowski (1937), *Über die determinanten mit über wiegender hauptdiagonale*, Commet. Math. Helv., 10, pp. 69–96 (1937) or *Alexander Ostrowski Collected Mathematical Papers* 1, Birkhaüser-Verlag, pp. 31–59 (1983).

[18] B. Parlett (1974), *The Rayleigh quotient algorithm iteration and some generalizations for nonnormal matrices*, Math. Comp., 28, pp. 679–693.

[19] B. Parlett and D. Scott (1979), *The Lanczos algorithm with selective orthogonalization*, Math. Comp., 33, pp. 217–238.

[20] B. Parlett (1980), *The Symmetric Eigenvalue Problem*, Prentice–Hall, Englewood Cliffs, NJ.

[21] B. Parlett, D. Taylor, and Z.-S. Liu (1985), *A look-ahead Lanczos algorithm for unsymmetric matrices*, Math. Comp., 44, pp. 105–124.

[22] B. Parlett (1992), *Reduction to tridiagonal form and minimal realizations*, SIAM J. Matrix Anal. Appl., 13, pp. 567–593.

[23] B. Parlett (1992), *The rewards for maintaining semi-orthogonality among Lanczos vectors*, J. Numer. Linear Algebra Appl., 1, pp. 243–267.

[24] B. Parlett, B. Nour Omid, and Z.-S. Liu (1985), *How to Maintain Semi-Orthogonality Among Lanczos Vectors*, CPAM-420, Center for Pure and Applied Mathematics, University of California, Berkeley, CA.

[25] A. Ruhe (1993), *The two-sided Arnoldi algorithm for nonsymmetric eigenvalue problems*, in Matrix Pencils, LNM 973, B. Kågström and A. Ruhe, eds., Springer-Verlag, Berlin, Heidelberg, New York, pp. 104–120.

[26] A. Ruhe (1995), *Rational Krylov, A Practical Algorithm for Large Sparse Nonsymmetric Matrix Pencils*, Report UCB/CSD-95-871, Computer Science Division (EECS), University of California, Berkeley, CA.

[27] Y. Saad (1980), *Variations of Arnoldi's method for computing eigenelements of large unsymmetric matrices*, Linear Algebra Appl., 34, pp. 269–295.

[28] H. Simon (1984), *Analysis of the symmetric Lanczos algorithm with reorthogonalization*, Linear Algebra Appl., 61, pp. 101–131.

[29] H. Simon (1984), *The Lanczos algorithm with partial reorthogonalization*, Math. Comp., 42, pp. 115–142.

[30] D. Sorensen (1992), *Implicit application of polynomial filters in a k-step Arnoldi process*, SIAM J. Matrix Anal. Appl., 13, pp. 357–385.

[31] G. Stewart and J. Sun (1990), *Matrix Perturbation Theory*, Academic Press, New York.

[32] G. Stewart (1993), *On the perturbation of LU, Cholesky, and QR factorizations*, SIAM J. Matrix Anal. Appl., 14, pp. 1141–1145.

[33] J. G. Sun (1991), *Perturbation bounds for the Cholesky and QR factors*, BIT, 31, pp. 341–352.

[34] J. G. Sun (1992), *Component-wise perturbation bounds for some matrix decompositions*, BIT, 32, pp. 702–714.

[35] C. Tong and Q. Ye (1995), *Analysis of the Finite Precision Bi-Conjugate Gradient Algorithm for Nonsymmetric Linear Systems*, preprint.

[36] H. I. van der Veen and K. Vuik (1995), *Bi-Lanczos with partial orthogonalization*, Computers and Structures, 56, pp. 605–613.

[37] J. Wilkinson (1965), *The Algebraic Eigenvalue Problem*, Oxford University Press, Oxford, UK.