SIAM J. MATRIX ANAL. APPL.
Vol. 16, No. 1, pp. 79–92, January 1995
© 1995 Society for Industrial and Applied Mathematics

Downloaded 01/04/13 to 150.135.135.70. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

A DIVIDE-AND-CONQUER ALGORITHM FOR THE BIDIAGONAL SVD*

MING GU† AND STANLEY C. EISENSTAT‡

Abstract. The authors present a stable and efficient divide-and-conquer algorithm for computing the singular value decomposition (SVD) of a lower bidiagonal matrix. Previous divide-and-conquer algorithms all suffer from a potential loss of orthogonality among the computed singular vectors unless extended precision arithmetic is used. A generalization that computes the SVD of a lower banded matrix is also presented.

Key words. singular value decomposition, divide-and-conquer, bidiagonal matrix

AMS subject classification. 65F15

1. Introduction.

Given an $(N+1) \times N$ lower bidiagonal matrix[1]

$$B = \begin{pmatrix} \alpha_1 & & & \\ \beta_1 & \alpha_2 & & \\ & \ddots & \ddots & \\ & & \beta_{N-1} & \alpha_N \\ & & & \beta_N \end{pmatrix}, \tag{1}$$

its *singular value decomposition* (SVD) is

$$B = X \begin{pmatrix} \Omega \\ 0 \end{pmatrix} Y^T,$$

where X and Y are $(N+1) \times (N+1)$ and $N \times N$ orthogonal matrices, respectively, Ω is an $N \times N$ nonnegative diagonal matrix and 0 is a row of zero elements. The columns of X and Y are the *left singular vectors* and the *right singular vectors* of B, respectively, and the diagonal entries of Ω are the *singular values* of B. This problem arises when one computes the SVD of a general matrix by first reducing it to bidiagonal form [10], [12]. In this paper, we propose a bidiagonal divide-and-conquer algorithm (BDC) for solving this problem.

BDC first partitions B as

$$B = \begin{pmatrix} B_1 & \alpha_k e_k & 0 \\ 0 & \beta_k e_1 & B_2 \end{pmatrix},$$

where B_1 and B_2 are lower bidiagonal matrices, each of which is a submatrix of B. Next it recursively computes the SVDs of B_1 and B_2 and computes orthogonal matrices $(Q,\ q)$ and W such that

$$B = \begin{pmatrix} Q & q \end{pmatrix} \begin{pmatrix} M \\ 0 \end{pmatrix} W^T,$$

* Received by the editors December 31, 1992; accepted for publication (in revised form) by J. R. Bunch, September 1, 1993. This research was supported in part by U. S. Army Research Office contract DAAL03-91-G-0032.

† Department of Mathematics and Lawrence Berkeley Laboratory, University of California, Berkeley, California 94720 (minggu@math.berkeley.edu).

‡ Department of Computer Science, Yale University, Box 208285, New Haven, Connecticut 06520-8285 (eisenstat-stan@cs.yale.edu).

[1] An $N \times N$ lower bidiagonal matrix can be put into the form (1) by appending a zero row; we consider this case since it simplifies the recursion.

Downloaded 01/04/13 to 150.135.135.70. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

where M is an $N \times N$ matrix with nonzero elements only in the first column and on the diagonal. Finally it finds the singular values of B by computing the SVD $M = U\Omega V^T$, where U and V are orthogonal matrices, and then computes the singular vector matrices of B as (QU, q) and WV, respectively.

Since error is associated with computation, a *numerical* SVD of B or M is usually defined as a decomposition of the form

$$B = \hat{X} \begin{pmatrix} \hat{\Omega} \\ 0 \end{pmatrix} \hat{Y}^T + O(\epsilon \|B\|_2) \quad \text{or} \quad M = \hat{U}\hat{\Omega}\hat{V}^T + O(\epsilon \|M\|_2), \tag{2}$$

where ϵ is the machine precision, $\hat{\Omega}$ is diagonal, and $\hat{X}$ and $\hat{Y}$ or $\hat{U}$ and $\hat{V}$ are *numerically orthogonal*. An algorithm that produces such a decomposition is said to be *stable*.

While the singular values of B and M are always well conditioned with respect to perturbations, the singular vectors can be extremely sensitive [13, pp. 419–420]. That is, $\hat{\Omega}$ must be close to Ω, but $\hat{X}$, $\hat{Y}$, $\hat{U}$, and $\hat{V}$ can be very different from X, Y, U, and V, respectively. Thus one is usually content with stable algorithms for computing the SVD of B or M.

Jessup and Sorensen [22] present a divide-and-conquer method that uses basically the same dividing strategy and computes the SVD of M using an algorithm based on the work in [5], [6], and [9]. While it can compute the singular values of M to high absolute accuracy, in the presence of close singular values it can have difficulties in computing numerically orthogonal singular vectors unless extended precision arithmetic is used [22], [23], [27].

In this paper we develop a new algorithm for computing the SVD of M based on the work in [16] and [19]. It uses an approach similar to that of Jessup and Sorensen for computing the singular values, but it uses a completely different approach for computing the singular vectors, one that is stable. The amount of work is roughly the same, yet it does not require the use or simulation of extended precision arithmetic. Since it uses this algorithm, BDC is stable as well. Moreover, BDC uses a new procedure for handling deflation that makes it up to twice as fast asymptotically as the Jessup and Sorensen method.

There are three other divide-and-conquer algorithms for the bidiagonal SVD. Arbenz and Golub [3] follow the Jessup and Sorensen approach, but divide B by removing a column rather than a row. Arbenz [1] and Gragg, Thornton, and Warner [14] (see also Borges and Gragg [4]) each use a divide-and-conquer method for the symmetric tridiagonal eigenproblem to compute a spectral decomposition of a symmetric permutation of the matrix $\left(\begin{smallmatrix} 0 & B^T \\ B & 0 \end{smallmatrix}\right)$ while taking advantage of its special structure. All three algorithms can be unstable as noted above, unless extra precision arithmetic is used. The techniques presented here can be used to stabilize (and speed up deflation in) these algorithms as well.

BDC computes all the singular values in $O(N^2)$ time and all the singular values and singular vectors in $O(N^3)$ time. By using the fast multipole method of Carrier, Greengard, and Rokhlin [7], [15], BDC can be accelerated to compute all the singular values in $O(N \log_2 N)$ time and all the singular values and singular vectors in $O(N^2)$ time (see [16] and [17] for details). These asymptotic times are better than the corresponding worst-case times ($O(N^2)$ and $O(N^3)$) for the Golub–Kahan algorithm [10], [12] and bisection with inverse iteration [20], [21].

Section 2 presents the dividing strategy; §3 presents an algorithm for computing the SVD of M; §4 presents the deflation procedure; and §5 generalizes BDC to

Downloaded 01/04/13 to 150.135.135.70. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

compute the SVD of a lower banded matrix.

We take the usual model of arithmetic[2]

$$\mathrm{fl}(\alpha \circ \beta) = (\alpha \circ \beta)\,(1+\xi),$$

where α and β are floating-point numbers, $\circ$ is one of $+, -, \times$ and $\div$, $\mathrm{fl}(\alpha \circ \beta)$ is the floating-point result of the operation $\circ$, and $|\xi| \leq \epsilon$. We also require that

$$\mathrm{fl}(\sqrt{\alpha}) = \sqrt{\alpha}\,(1+\xi)$$

for any positive floating-point number α. For simplicity, we ignore the possibility of overflow and underflow.

2. "Dividing" the matrix. Given an $(N+1) \times N$ lower bidiagonal matrix B, we divide B into two subproblems as follows (cf. [22]):

$$B = \begin{pmatrix} B_1 & \alpha_k e_k & 0 \\ 0 & \beta_k e_1 & B_2 \end{pmatrix}, \tag{3}$$

where $1 < k < N$, B_1 and B_2 are $k \times (k-1)$ and $(N-k+1) \times (N-k)$ lower bidiagonal matrices, respectively, and e_j is the jth unit vector of appropriate dimension. Usually k is taken to be $\lfloor N/2 \rfloor$.

Let

$$B_i = \begin{pmatrix} Q_i & q_i \end{pmatrix} \begin{pmatrix} D_i \\ 0 \end{pmatrix} W_i^T$$

be the SVD of B_i. Let l_1^T and λ_1 be the last row and last component of Q_1 and q_1, respectively, and let f_2^T and φ_2 be the first row and first component of Q_2 and q_2, respectively. Substituting into (3), we get

$$B = \begin{pmatrix} q_1 & Q_1 & 0 & 0 \\ 0 & 0 & Q_2 & q_2 \end{pmatrix} \begin{pmatrix} \alpha_k\lambda_1 & 0 & 0 \\ \alpha_k l_1 & D_1 & 0 \\ \beta_k f_2 & 0 & D_2 \\ \beta_k\varphi_2 & 0 & 0 \end{pmatrix} \begin{pmatrix} 0 & W_1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & W_2 \end{pmatrix}^T.$$

There is only one nonzero element in the first and last rows of the middle matrix. Applying a Givens rotation to zero out $\beta_k\varphi_2$, we have

$$\begin{aligned} B &= \left(\begin{pmatrix} c_0 q_1 & Q_1 & 0 \\ s_0 q_2 & 0 & Q_2 \end{pmatrix} \begin{pmatrix} -s_0 q_1 \\ c_0 q_2 \end{pmatrix} \right) \begin{pmatrix} r_0 & 0 & 0 \\ \alpha_k l_1 & D_1 & 0 \\ \beta_k f_2 & 0 & D_2 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} 0 & W_1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & W_2 \end{pmatrix}^T \\ &\equiv \begin{pmatrix} Q & q \end{pmatrix} \begin{pmatrix} M \\ 0 \end{pmatrix} W^T, \end{aligned} \tag{4}$$

where

$$r_0 = \sqrt{(\alpha_k\lambda_1)^2 + (\beta_k\varphi_2)^2}, \qquad c_0 = \frac{\alpha_k\lambda_1}{r_0}, \quad \text{and} \quad s_0 = \frac{\beta_k\varphi_2}{r_0}. \tag{5}$$

[2] This model excludes machines such as CRAY and CDC Cyber that do not have a guard digit. BDC can easily be modified for such machines.

Thus we have reduced B to $\binom{M}{0}$ by orthogonal transformations $(Q,\ q)$ and W, and M has nonzero elements only in the first column and on the diagonal.

Let $U\Omega V^T$ be the SVD of M computed using the algorithm described in §3. Substituting into (4) we obtain

$$B = \begin{pmatrix} Q & q \end{pmatrix} \begin{pmatrix} U\Omega V^T \\ 0 \end{pmatrix} W^T \equiv X \begin{pmatrix} \Omega \\ 0 \end{pmatrix} Y^T.$$

The singular values of B are the diagonal elements of Ω, and the singular vectors of B are the columns of X and Y. To compute the SVDs of B_1 and B_2, this process ((3) and (4)) can be recursively applied until the subproblems are sufficiently small. These small subproblems are then solved using the Golub–Kahan algorithm [10], [12]. There can be at most $O(\log_2 N)$ levels of recursion.

Equations (3) and (4) also suggest a recursion for computing only the singular values. Let f_1^T and φ_1 be the first row of Q_1 and the first component of q_1, respectively; let l_2^T and λ_2 be the last row of Q_2 and last component of q_2, respectively; let f^T and φ be the first row of Q and first component of q, respectively; and let l^T and λ be the last row of Q and the last component of q, respectively. Suppose that D_i, f_i, λ_i, l_i, and φ_i are given for $i = 1, 2$. Then we can compute Ω, f, λ, l, and φ by computing r_0, s_0, and c_0 using (5), computing the SVD of M, and computing

$$f^T = \begin{pmatrix} c_0\varphi_1 & f_1^T & 0 \end{pmatrix} U, \quad \lambda = c_0\lambda_2, \quad l^T = \begin{pmatrix} s_0\lambda_2 & 0 & l_2^T \end{pmatrix} U, \quad \text{and} \quad \varphi = -s_0\varphi_1.$$

There is a similar recursion for the divide-and-conquer algorithms in [8], [16],and [18] for the symmetric tridiagonal eigenproblem.

3. Computing the SVD of M. In this section we present a stable and efficient algorithm for finding the SVD of the $n \times n$ matrix

$$M = \begin{pmatrix} z_1 & & & \\ z_2 & d_2 & & \\ \vdots & & \ddots & \\ z_n & & & d_n \end{pmatrix},$$

where $D = \operatorname{diag}(d_1, \ldots, d_n)$, with[3] $0 \equiv d_1 \le d_2 \le \cdots \le d_n$, and $z = (z_1, \ldots, z_n)^T$. We further assume that

$$d_{j+1} - d_j \ge \tau\|M\|_2 \quad \text{and} \quad |z_j| \ge \tau\|M\|_2, \tag{6}$$

where τ is a small multiple of ϵ to be specified later. Any matrix of this form can be reduced to one that satisfies these conditions by using the deflation procedure described in §4.1 and a simple permutation.

The following lemma characterizes the singular values and singular vectors of M.

LEMMA 3.1 (Jessup and Sorensen [22]). *Let $U\Omega V^T$ be the* SVD *of M with*

$$U = (u_1, \ldots, u_n), \quad \Omega = \operatorname{diag}(\omega_1, \ldots, \omega_n), \quad \textit{and} \quad V = (v_1, \ldots, v_n).$$

Then the singular values $\{\omega_i\}_{i=1}^n$ satisfy the interlacing property

$$0 = d_1 < \omega_1 < d_2 < \cdots < d_n < \omega_n < d_n + ||z||_2$$

[3] We introduce d_1 to simplify the presentation.

Downloaded 01/04/13 to 150.135.135.70. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

Downloaded 01/04/13 to 150.135.135.70. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

and the secular equation

$$f(\omega) = 1 + \sum_{k=1}^{n} \frac{z_k^2}{d_k^2 - \omega^2} = 0.$$

The singular vectors are given by

$$u_i = \left(\frac{z_1}{d_1^2 - \omega_i^2}, \ldots, \frac{z_n}{d_n^2 - \omega_i^2} \right)^T \Bigg/ \sqrt{\sum_{k=1}^{n} \frac{z_k^2}{\left(d_k^2 - \omega_i^2\right)^2}}, \tag{7}$$

$$v_i = \left(-1, \frac{d_2 z_2}{d_2^2 - \omega_i^2}, \ldots, \frac{d_n z_n}{d_n^2 - \omega_i^2} \right)^T \Bigg/ \sqrt{1 + \sum_{k=2}^{n} \frac{(d_k z_k)^2}{\left(d_k^2 - \omega_i^2\right)^2}}. \tag{8}$$

On the other hand, given D and all the singular values, we can reconstruct M up to the signs of the z_i (cf. Löwner [25]).

LEMMA 3.2. *Given a diagonal matrix* $D = \operatorname{diag}(d_1, \ldots, d_n)$ *and a set of numbers* $\{\hat{\omega}_i\}_{i=1}^n$ *satisfying the interlacing property*

$$0 = d_1 < \hat{\omega}_1 < d_2 < \cdots < d_n < \hat{\omega}_n, \tag{9}$$

there exists a matrix

$$\hat{M} = \begin{pmatrix} \hat{z}_1 & & & \\ \hat{z}_2 & d_2 & & \\ \vdots & & \ddots & \\ \hat{z}_n & & & d_n \end{pmatrix}$$

whose singular values are $\{\hat{\omega}_i\}_{i=1}^n$. *The vector* $\hat{z} = (\hat{z}_1, \hat{z}_2, \ldots, \hat{z}_n)^T$ *is given by*

$$|\hat{z}_i| = \sqrt{\left(\hat{\omega}_n^2 - d_i^2\right) \prod_{k=1}^{i-1} \frac{\left(\hat{\omega}_k^2 - d_i^2\right)}{\left(d_k^2 - d_i^2\right)} \prod_{k=i}^{n-1} \frac{\left(\hat{\omega}_k^2 - d_i^2\right)}{\left(d_{k+1}^2 - d_i^2\right)}}, \tag{10}$$

where the sign of $\hat{z}_i$ *can be chosen arbitrarily.*

Proof. Assume that $\hat{M}$ (and thus $\hat{z}$) exists. Then

$$\det\left(D^2 + \hat{z}\hat{z}^T - \omega^2 I\right) = \det\left(\hat{M}\hat{M}^T - \omega^2 I\right) = \prod_{k=1}^{n} \left(\hat{\omega}_k^2 - \omega^2\right).$$

On the other hand,

$$\begin{aligned} \det\left(D^2 + \hat{z}\hat{z}^T - \omega^2 I\right) &= \det\left(I + \left(D^2 - \omega^2 I\right)^{-1} \hat{z}\hat{z}^T\right) \det\left(D^2 - \omega^2 I\right) \\ &= \left(1 + \sum_{k=1}^{n} \frac{\hat{z}_k^2}{d_k^2 - \omega^2}\right) \prod_{k=1}^{n} \left(d_k^2 - \omega^2\right). \end{aligned}$$

Combining these two equations,

$$\prod_{k=1}^{n} \left(\hat{\omega}_k^2 - \omega^2\right) = \left(1 + \sum_{k=1}^{n} \frac{\hat{z}_k^2}{d_k^2 - \omega^2}\right) \prod_{k=1}^{n} \left(d_k^2 - \omega^2\right).$$

Downloaded 01/04/13 to 150.135.135.70. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

Setting $\omega = d_i$, we get

$$\hat{z}_i^2 = \prod_{k=1}^{n} (\hat{\omega}_k^2 - d_i^2) \Big/ \prod_{k \neq i} (d_k^2 - d_i^2).$$

Because of the interlacing property (9), the expression on the right-hand side is positive. Taking square roots we get (10). Working backward, if $\hat{z}$ is given by (10), then the singular values of $\hat{M}$ are $\{\hat{\omega}_i\}_{i=1}^n$. □

3.1. Computing the singular vectors. If ω_i were given, then we could compute each difference $d_k^2 - \omega_i^2$ in (7) and (8) to high relative accuracy as $(d_k - \omega_i)(d_k + \omega_i)$. We could also compute each product, each ratio, and each sum to high relative accuracy and thus compute u_i and v_i to componentwise high relative accuracy.

In practice we can only hope to compute an approximation $\hat{\omega}_i$ to ω_i. But problems can arise if we approximate u_i and v_i by

$$\hat{u}_i = \left(\frac{z_1}{d_1^2 - \hat{\omega}_i^2}, \ldots, \frac{z_n}{d_n^2 - \hat{\omega}_i^2} \right)^T \Big/ \sqrt{\sum_{k=1}^{n} \frac{z_k^2}{(d_k^2 - \hat{\omega}_i^2)^2}}$$

and

$$\hat{v}_i = \left(-1, \frac{d_2 z_2}{d_2^2 - \hat{\omega}_i^2}, \ldots, \frac{d_n z_n}{d_n^2 - \hat{\omega}_i^2} \right)^T \Big/ \sqrt{1 + \sum_{k=2}^{n} \frac{(d_k z_k)^2}{(d_k^2 - \hat{\omega}_i^2)^2}}$$

(i.e., replace ω_i by $\hat{\omega}_i$ in (7) and (8) as in [22]). For even if $\hat{\omega}_i$ is close to ω_i, the approximate ratios $z_k/(d_k^2 - \hat{\omega}_i^2)$ and $d_k z_k/(d_k^2 - \hat{\omega}_i^2)$ can still be very different from the exact ratios $z_k/(d_k^2 - \omega_i^2)$ and $d_k z_k/(d_k^2 - \omega_i^2)$, resulting in $\hat{u}_i$ and $\hat{v}_i$ very different from u_i and v_i. And when all the approximate singular values $\{\hat{\omega}_i\}_{i=1}^n$ are computed and all the corresponding singular vectors are approximated in this manner, the resulting singular vector matrices may not be orthogonal.

Lemma 3.2 allows us to overcome this problem. After we have computed all the approximate singular values $\{\hat{\omega}_i\}_{i=1}^n$ of M, we find a *new* matrix $\hat{M}$ whose *exact* singular values are $\{\hat{\omega}_i\}_{i=1}^n$ and then compute the singular vectors of $\hat{M}$ using Lemma 3.1. Note that each difference

$$\hat{\omega}_k^2 - d_i^2 = (\hat{\omega}_k - d_i)(\hat{\omega}_k + d_i) \quad \text{and} \quad d_k^2 - d_i^2 = (d_k - d_i)(d_k + d_i)$$

in (10) can be computed to high relative accuracy, as can each ratio and each product, and we can choose the sign of $\hat{z}_i$ to be the sign of z_i. Thus $\hat{z}_i$ can be computed to high relative accuracy. Substituting the *exact* singular values $\{\hat{\omega}_i\}_{i=1}^n$ and the computed $\hat{z}$ into (7) and (8), each singular vector of $\hat{M}$ can be computed to componentwise high relative accuracy. Consequently, after all the singular vectors are computed, the singular vector matrices of $\hat{M}$ will be numerically orthogonal.

To ensure the existence of $\hat{M}$, we need $\{\hat{\omega}_i\}_{i=1}^n$ to satisfy (9). But since the exact singular values of M satisfy the same interlacing property (see Lemma 3.1), this is only an accuracy requirement on the computed singular values and is not an additional restriction on M.

We can use the SVD of $\hat{M}$ as an approximation to the SVD of M. And since $\|M - \hat{M}\|_2 = \|z - \hat{z}\|_2$, such a substitution is stable (see (2)) as long as $\hat{z}$ is close to z (cf. [16], [19]).

Downloaded 01/04/13 to 150.135.135.70. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

3.2. Computing the singular values. To guarantee that $\hat{z}$ is close to z, we must ensure that the approximations $\{\hat{\omega}_i\}_{i=1}^n$ to the singular values are sufficiently accurate. The key is the stopping criterion for the root-finder, which requires a slight reformulation of the secular equation (cf. [5], [16], [19]).

Consider the singular value $\omega_i \in (d_i, d_{i+1})$, where $1 \le i \le n-1$; we consider the case $i = n$ later.

First assume that[4] $\omega_i \in \left(d_i, \frac{d_i+d_{i+1}}{2}\right)$. Let $\delta_j = d_j - d_i$ and let

$$\psi_i(\mu) \equiv \sum_{j=1}^{i} \frac{z_j^2}{(\delta_j - \mu)(d_j + d_i + \mu)} \quad \text{and} \quad \phi_i(\mu) \equiv \sum_{j=i+1}^{n} \frac{z_j^2}{(\delta_j - \mu)(d_j + d_i + \mu)}.$$

Setting $\omega = d_i + \mu$, we seek the root $\mu_i = \omega_i - d_i \in (0, \delta_{i+1}/2)$ of the reformulated secular equation

$$g_i(\mu) \equiv f(\mu + d_i) = 1 + \psi_i(\mu) + \phi_i(\mu) = 0.$$

Note that we can compute each ratio $z_j^2/((\delta_j - \mu)(d_j + d_i + \mu))$ in $g_i(\mu)$ to high relative accuracy for any $\mu \in (0, \delta_{i+1}/2)$. Indeed, either $\delta_j - \mu$ is a sum of negative terms or $|\mu| \le |\delta_j|/2$, and $d_j + d_i + \mu$ is a sum of positive terms. Thus, since both $\psi_i(\mu)$ and $\phi_i(\mu)$ are sums of terms of the same sign, we can bound the error in computing $g_i(\mu)$ by

$$\eta n(1 + |\psi_i(\mu)| + |\phi_i(\mu)|),$$

where η is a small multiple of ϵ that is independent of n and μ.

Now assume that $\omega_i \in \left[\frac{d_i+d_{i+1}}{2}, d_{i+1}\right)$. Let $\delta_j = d_j - d_{i+1}$ and let

$$\psi_i(\mu) \equiv \sum_{j=1}^{i} \frac{z_j^2}{(\delta_j - \mu)(d_j + d_{i+1} + \mu)} \quad \text{and} \quad \phi_i(\mu) \equiv \sum_{j=i+1}^{n} \frac{z_j^2}{(\delta_j - \mu)(d_j + d_{i+1} + \mu)}.$$

Setting $\omega = d_{i+1} + \mu$, we seek the root $\mu_i = \omega_i - d_{i+1} \in [\delta_i/2, 0)$ of the equation

$$g_i(\mu) \equiv f(\mu + d_{i+1}) = 1 + \psi_i(\mu) + \phi_i(\mu) = 0.$$

For any $\mu \in [\delta_i/2, 0)$, we can compute each ratio $z_j^2/((\delta_j - \mu)(d_j + d_{i+1} + \mu))$ to high relative accuracy (either $\delta_j - \mu$ is a sum of positive terms or $|\mu| \le |\delta_j|/2$; and $d_j + d_i + \mu = d_j + (d_{i+1} + \mu)$, where $|\mu| \le d_{i+1}/2$) and we can bound the error in computing $g_i(\mu)$ as before.

Finally consider the case $i = n$. Let $\delta_j = d_j - d_n$ and let

$$\psi_n(\mu) \equiv \sum_{j=1}^{n} \frac{z_j^2}{(\delta_j - \mu)(d_j + d_n + \mu)} \quad \text{and} \quad \phi_n(\mu) \equiv 0.$$

Setting $\omega = d_n + \mu$, we seek the root $\mu_n = \omega_n - d_n \in (0, \|z\|_2)$ of the equation

$$g_n(\mu) \equiv f(\mu + d_n) = 1 + \psi_n(\mu) + \phi_n(\mu) = 0.$$

[4] This can be checked by computing $f\left(\frac{d_i+d_{i+1}}{2}\right)$. If $f\left(\frac{d_i+d_{i+1}}{2}\right) > 0$, then $\omega_i \in \left(d_i, \frac{d_i+d_{i+1}}{2}\right)$, otherwise $\omega_i \in \left[\frac{d_i+d_{i+1}}{2}, d_{i+1}\right)$.

Downloaded 01/04/13 to 150.135.135.70. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

Again, for any $\mu \in (0, \|z\|_2)$, we can compute each ratio $z_j^2/((\delta_j - \mu)(d_j + d_n + \mu))$ to high relative accuracy, and we can bound the error in computing $g_n(\mu)$ as before.

In practice, the root-finder cannot make any progress at a point μ where it is impossible to determine the sign of $g_i(\mu)$ numerically. Thus we propose the stopping criterion

$$|g_i(\mu)| \leq \eta n \left(1 + |\psi_i(\mu)| + |\phi_i(\mu)|\right), \tag{11}$$

where, as before, the right-hand side is an upper bound on the round-off error in computing $g_i(\mu)$. Note that for each i, there is at least one floating-point number that satisfies this stopping criterion numerically, namely, $\mathrm{fl}(\mu_i)$.

We have not specified the method for finding the root of $g_i(\mu)$. We can use the bisection method or the rational interpolation strategies in [4], [5], [14], [24]. What is most important is the stopping criterion and the fact that, with the reformulation of the secular equation given above, we can find a μ that satisfies it.

3.3. Numerical stability. We now show that $\hat{z}$ is close to z.

THEOREM 3.3. *If* $\tau = 2\eta n^2$ *in* (6) *and each* $\hat{\mu}_i$ *satisfies* (11), *then*

$$|\hat{z}_i - z_i| \leq 4\eta n^2 \|z\|_2. \tag{12}$$

The proof is nearly identical to that of the analogous result in [19]. As argued there, the factor n^2 in τ and (12) is likely to be $O(n)$ in practice.

4. Deflation.

4.1. Deflation for M. Let

$$M = \begin{pmatrix} z_1 & & & \\ z_2 & d_2 & & \\ \vdots & & \ddots & \\ z_n & & & d_n \end{pmatrix},$$

where $D = \mathrm{diag}(d_1, \ldots, d_n)$ with $d_1 \equiv 0$ and $d_i \geq 0$, and $z = (z_1, \ldots, z_n)^T$. We now show that we can stably reduce M to a matrix of the same form that satisfies

$$|d_i - d_j| \geq \tau \|M\|_2 \quad \text{for } i \neq j \quad \text{and} \quad |z_i| \geq \tau \|M\|_2$$

(cf. (6)), where τ is specified in §3.3. We illustrate the reductions for $n = 3$, $i = 3$, and $j = 2$. Similar reductions appear in [22].

Assume that $|z_1| < \tau\|M\|_2$. Changing z_1 to $\tau\|M\|_2$ perturbs M by $O(\tau\|M\|_2)$:

$$M = \begin{pmatrix} z_1 & & \\ z_2 & d_2 & \\ z_3 & & d_3 \end{pmatrix} = \begin{pmatrix} \tau\|M\|_2 & & \\ z_2 & d_2 & \\ z_3 & & d_3 \end{pmatrix} + O(\tau\|M\|_2). \tag{13}$$

The perturbed matrix has the same structure as M and satisfies $|z_1| \geq \tau\|M\|_2$.

Next assume that $|z_i| < \tau\|M\|_2$ for $i \geq 2$. Changing z_i to zero perturbs M by $O(\tau\|M\|_2)$:

$$M = \begin{pmatrix} z_1 & & \\ z_2 & d_2 & \\ z_3 & & d_3 \end{pmatrix} = \begin{pmatrix} z_1 & & \\ z_2 & d_2 & \\ 0 & & d_3 \end{pmatrix} + O(\tau\|M\|_2). \tag{14}$$

Downloaded 01/04/13 to 150.135.135.70. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

In the perturbed matrix, d_i is a singular value and can be deflated, and the $(n-1) \times (n-1)$ leading principle submatrix has the same structure as M.

Now assume that $|d_i - d_1| = |d_i| < \tau\|M\|_2$ for $i \geq 2$. Changing d_i to zero and applying a Givens rotation G to zero out z_i perturbs the matrix GM by $O(\tau\|M\|_2)$:

$$GM = \begin{pmatrix} c & & s \\ & 1 & \\ -s & & c \end{pmatrix} \begin{pmatrix} z_1 & & \\ z_2 & d_2 & \\ z_3 & & 0 \end{pmatrix} + O(\tau\|M\|_2) \tag{15}$$
$$= \begin{pmatrix} r & & \\ z_2 & d_2 & \\ 0 & & 0 \end{pmatrix} + O(\tau\|M\|_2),$$

where $r = \sqrt{z_i^2 + z_1^2}$, $s = z_i/r$, and $c = z_1/r$. In the perturbed matrix, 0 is a singular value and can be deflated, and the $(n-1) \times (n-1)$ leading principle submatrix has the same structure as M.

Finally assume that $|d_i - d_j| < \tau\|M\|_2$ for $i, j \geq 2$. Changing d_j to d_i and symmetrically applying a Givens rotation G to zero out z_i perturbs the matrix GMG^T by $O(\tau\|M\|_2)$:

$$GMG^T = \begin{pmatrix} 1 & & \\ & c & s \\ & -s & c \end{pmatrix} \begin{pmatrix} z_1 & & \\ z_2 & d_3 & \\ z_3 & & d_3 \end{pmatrix} \begin{pmatrix} 1 & & \\ & c & -s \\ & s & c \end{pmatrix} + O(\tau\|M\|_2) \tag{16}$$
$$= \begin{pmatrix} z_1 & & \\ r & d_3 & \\ 0 & & d_3 \end{pmatrix} + O(\tau\|M\|_2),$$

where $r = \sqrt{z_i^2 + z_j^2}$, $s = z_i/r$, and $c = z_j/r$. In the perturbed matrix, d_i is a singular value and can be deflated, and the $(n-1) \times (n-1)$ leading principle submatrix has the same structure as M.

4.2. Local deflation. In the dividing strategy for BDC (see (4)), we write

$$B = \begin{pmatrix} Q & q \end{pmatrix} \begin{pmatrix} M \\ 0 \end{pmatrix} W^T = \begin{pmatrix} QU & q \end{pmatrix} \begin{pmatrix} \Omega \\ 0 \end{pmatrix} (WV)^T, \tag{17}$$

where

$$Q = \begin{pmatrix} c_0 q_1 & Q_1 & 0 \\ s_0 q_2 & 0 & Q_2 \end{pmatrix}, \quad M = \begin{pmatrix} r_0 & 0 & 0 \\ \alpha_k l_1 & D_1 & 0 \\ \beta_k f_2 & 0 & D_2 \end{pmatrix}, \quad \text{and} \quad W = \begin{pmatrix} 0 & W_1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & W_2 \end{pmatrix},$$

l_1^T is the last row of Q_1, f_2^T is first row of Q_2, and $U\Omega V^T$ is the SVD of M.

Note that both Q and W are block matrices with some zero blocks. Since the main cost of BDC is in computing the matrix-matrix products QU and WV, we would like to take advantage of this structure. In this subsection we describe a deflation procedure for BDC that gets a speedup of roughly a factor of two by doing so. This approach is not used in [22].

If $|r_0| < \tau\|M\|_2$, then we apply reduction (13). If the vector $(\alpha_k l_1^T, \beta_k f_2^T)$ has some components with small absolute value, then we apply reduction (14). In both cases the block structure of Q and W is preserved. If D_1 has a small diagonal element, then we apply reduction (15), and if D_1 has two close diagonal elements, then we apply

reduction (16). Again in both cases the block structure is preserved. We do the same when D_2 has a small diagonal element or has two close diagonal elements.

However, when D_1 has a diagonal element that is close to a diagonal element in D_2 and we apply reduction (16), the block structure of Q and W is changed. To illustrate, assume that after applying a permutation, the first diagonal element of D_1 is close to the last diagonal element of D_2. Let

$$Q_1 = (\tilde{q}_1 \;\; \tilde{Q}_1), \quad Q_2 = (\tilde{Q}_2 \;\; \tilde{q}_2), \quad W_1 = (\tilde{w}_1 \;\; \tilde{W}_1), \quad \text{and} \quad W_2 = (\tilde{W}_2 \;\; \tilde{w}_2);$$

and let

$$\alpha_k l_1 = \begin{pmatrix} z_2 \\ \tilde{z}_1 \end{pmatrix}, \quad \beta_k f_2 = \begin{pmatrix} \tilde{z}_2 \\ z_N \end{pmatrix}, \quad D_1 = \operatorname{diag}(d_2, \tilde{D}_1), \quad \text{and} \quad D_2 = \operatorname{diag}(\tilde{D}_2, d_N).$$

Changing d_2 to d_N and applying a Givens rotation G to zero out z_N, we get

$$GMG^T = \begin{pmatrix} r_0 & & & & \\ r & d_N & & & \\ \tilde{z}_1 & & \tilde{D}_1 & & \\ \tilde{z}_2 & & & \tilde{D}_2 & \\ 0 & & & & d_N \end{pmatrix} + O(\tau\|M\|_2) \equiv \begin{pmatrix} \tilde{M}_1 & 0 \\ 0 & d_N \end{pmatrix} + O(\tau\|M\|_2),$$

where $r = \sqrt{z_2^2 + z_N^2}$, $c = z_2/r$, and $s = z_N/r$. Substituting into (17), we have

$$\begin{aligned} B &= (QG^T \;\; q) \begin{pmatrix} GMG^T \\ 0 \end{pmatrix} (WG^T)^T \\ &= (\tilde{X}_1 \;\; \tilde{x} \;\; q) \begin{pmatrix} \tilde{M}_1 & 0 \\ 0 & d_N \\ 0 & 0 \end{pmatrix} (\tilde{Y}_1 \;\; \tilde{y})^T + O(\tau\|M\|_2), \end{aligned}$$

where

$$\tilde{X}_1 = \begin{pmatrix} c_0 q_1 & c\tilde{q}_1 & \tilde{Q}_1 & 0 \\ s_0 q_2 & s\tilde{q}_2 & 0 & \tilde{Q}_2 \end{pmatrix} \quad \text{and} \quad \tilde{x} = \begin{pmatrix} -s\tilde{q}_1 \\ c\tilde{q}_2 \end{pmatrix}$$

and

$$\tilde{Y}_1 = \begin{pmatrix} 0 & c\tilde{w}_1 & \tilde{W}_1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & s\tilde{w}_2 & 0 & \tilde{W}_2 \end{pmatrix} \quad \text{and} \quad \tilde{y} = \begin{pmatrix} -s\tilde{w}_1 \\ 0 \\ c\tilde{w}_2 \end{pmatrix}.$$

d_N is an approximate singular value of B and can be deflated. The corresponding approximate left and right singular vectors are $\tilde{x}$ and $\tilde{y}$, respectively. The matrix $\tilde{M}_1$ has the same structure as M and can be deflated in a similar fashion until no diagonal element of $\tilde{D}_1$ is close to a diagonal element of $\tilde{D}_2$.

Thus, ignoring permutations of the columns of Q_i and W_i and the diagonal entries of D_i, after a series of these interblock deflations B can be written as

$$B = (\tilde{X}_1 \;\; \tilde{X}_2 \;\; q) \begin{pmatrix} \tilde{M}_1 & 0 \\ 0 & \tilde{\Omega}_2 \\ 0 & 0 \end{pmatrix} (\tilde{Y}_1 \;\; \tilde{Y}_2)^T + O(\tau\|B\|_2).$$

Downloaded 01/04/13 to 150.135.135.70. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

Downloaded 01/04/13 to 150.135.135.70. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

$\tilde{\Omega}_2$ is a diagonal matrix whose diagonal elements are the deflated singular values, and the columns of $\tilde{X}_2$ and $\tilde{Y}_2$ are the corresponding approximate left and right singular vectors. $\tilde{M}_1$ is of the form

$$\tilde{M}_1 = \begin{pmatrix} r_0 & & & \\ \tilde{z}_0 & \tilde{D}_0 & & \\ \tilde{z}_1 & & \tilde{D}_1 & \\ \tilde{z}_2 & & & \tilde{D}_2 \end{pmatrix},$$

where the dimension of $\tilde{D}_0$ is the number of deflations, $\tilde{D}_1$ and $\tilde{D}_2$ contain the diagonal elements of D_1 and D_2 not affected by deflation, and $\tilde{z}_0$, $\tilde{z}_1$, and $\tilde{z}_2$ are defined accordingly. $\tilde{X}_1$ and $\tilde{Y}_1$ are of the form

$$\tilde{X}_1 = \begin{pmatrix} c_0 q_1 & \tilde{Q}_{0,1} & \tilde{Q}_1 & 0 \\ s_0 q_2 & \tilde{Q}_{0,2} & 0 & \tilde{Q}_2 \end{pmatrix} \quad \text{and} \quad \tilde{Y}_1 = \begin{pmatrix} 0 & \tilde{W}_{0,1} & \tilde{W}_1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & \tilde{W}_{0,2} & 0 & \tilde{W}_2 \end{pmatrix}, \tag{18}$$

where the column dimension of $\tilde{Q}_{0,1}$, $\tilde{Q}_{0,2}$, $\tilde{W}_{0,1}$, and $\tilde{W}_{0,2}$ is the number of deflations, and the columns of $\tilde{Q}_1$, $\tilde{Q}_2$, $\tilde{W}_1$, and $\tilde{W}_2$ are those of Q_1, Q_2, W_1, and W_2 not affected by deflation.

Let $\tilde{U}_1 \tilde{\Omega}_1 \tilde{V}_1^T$ be the SVD of $\tilde{M}_1$. Then

$$\begin{aligned} B &= \begin{pmatrix} \tilde{X}_1 & \tilde{X}_2 & q \end{pmatrix} \begin{pmatrix} \tilde{U}_1 \tilde{\Omega}_1 \tilde{V}_1^T & 0 \\ 0 & \tilde{\Omega}_2 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} \tilde{Y}_1 & \tilde{Y}_2 \end{pmatrix}^T + O(\tau \|B\|_2) \\ &= \begin{pmatrix} \tilde{X}_1 \tilde{U}_1 & \tilde{X}_2 & q \end{pmatrix} \begin{pmatrix} \tilde{\Omega}_1 & 0 \\ 0 & \tilde{\Omega}_2 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} \tilde{Y}_1 \tilde{V}_1 & \tilde{Y}_2 \end{pmatrix}^T + O(\tau \|B\|_2). \end{aligned}$$

Thus $(\tilde{X}_1 \tilde{U}_1, \tilde{X}_2, q)$ and $(\tilde{Y}_1 \tilde{V}_1, \tilde{Y}_2)$ are approximate left and right singular vector matrices of B, respectively. The matrices $\tilde{X}_1 \tilde{U}_1$ and $\tilde{Y}_1 \tilde{V}_1$ can be computed while taking advantage of the block structure of $\tilde{X}_1$ and $\tilde{Y}_1$ in (18).

We refer to these as local deflations since they are associated with individual subproblems.

4.3. Global deflation. To illustrate *global* deflation, we look at two levels of the dividing strategy (see (4)):

$$B = \begin{pmatrix} B_1 & \alpha_{i+j} e_{i+j} & \\ & \beta_{i+j} e_1 & B_2 \end{pmatrix} = \begin{pmatrix} B_{1,1} & \alpha_i e_i & & \\ & \beta_i e_1 & B_{1,2} & \alpha_{i+j} e_j \\ & & & \beta_{i+j} e_1 \quad B_2 \end{pmatrix}, \tag{19}$$

where B_1, B_2, $B_{1,1}$ and $B_{1,2}$ are principle submatrices of B of dimensions $(i+j) \times (i+j-1)$, $(N-i-j+1) \times (N-i-j)$, $i \times (i-1)$, and $j \times (j-1)$, respectively.

Let $X_{1,2} \left(\begin{smallmatrix} D_{1,2} \\ 0 \end{smallmatrix}\right) W_{1,2}^T$ be the SVD of $B_{1,2}$, and let $(f_{1,2}^T, \varphi_{1,2})$ and $(l_{1,2}^T, \lambda_{1,2})$ be the first and last rows of $X_{1,2}$, respectively. Then

$$B = \bar{X} \begin{pmatrix} B_{1,1} & \alpha_i e_i & & \\ & \beta_i f_{1,2} & D_{1,2} & \alpha_{i+j} l_{1,2} \\ & \beta_i \varphi_{1,2} & 0 & \alpha_{i+j} \lambda_{1,2} \\ & & & \beta_{i+j} e_1 \quad B_2 \end{pmatrix} \bar{Y}^T, \tag{20}$$

Downloaded 01/04/13 to 150.135.135.70. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

where $\bar{X} = \mathrm{diag}(I_i,\ X_{1,2},\ I_{N-i-j+1})$ and $\bar{Y} = \mathrm{diag}(I_{i-1},\ 1,\ W_{1,2},\ 1,\ I_{N-i-j})$.

Let $\bar{d}_s$ be the sth diagonal element of $D_{1,2}$, and let $\bar{f}_s$ and $\bar{l}_s$ be the sth components of $f_{1,2}$ and $l_{1,2}$, respectively. Then ignoring all zero components, the $(i+s)$th column and row of the middle matrix in (20) are $(\bar{d}_s)$ and $(\beta_i \bar{f}_s,\ \bar{d}_s,\ \alpha_{i+j}\bar{l}_s)$, respectively. Thus if both $|\beta_i \bar{f}_s|$ and $|\alpha_{i+j}\bar{l}_s|$ are small, then we can perturb them to zero. $\bar{d}_s$ is a singular value of the perturbed matrix and the $(i+s)$th columns of $\bar{X}$ and $\bar{Y}$ are the corresponding left and right singular vectors, respectively. This singular value and its singular vectors can be deflated from all subsequent subproblems. We call this *global* deflation.

Consider the deflation procedure for computing the SVD in §4.2. If $|\beta_i \bar{f}_s|$ is small, then it can be perturbed to zero. This is a local deflation if only $|\beta_i \bar{f}_s|$ is small and a global deflation if $|\alpha_{i+j}\bar{l}_s|$ is also small.

5. Computing the SVD of a banded matrix. We now generalize BDC to compute the SVD of a lower banded matrix. This problem arises when one uses the block Lanczos algorithm to compute the SVD of a sparse matrix [11], [13]. Arbenz [2] has similarly generalized a divide-and-conquer algorithm for the symmetric tridiagonal eigenproblem to solve the symmetric banded eigenproblem.

Let B be an $(N+K)\times N$ lower $(K+1)$-diagonal matrix with $K \ll N$. We divide B into two subproblems as follows:

$$B = \begin{pmatrix} B_{1,1} & B_{1,2} & 0 \\ 0 & B_{2,2} & B_{2,3} \end{pmatrix}, \tag{21}$$

where $1 < k < N$, $B_{1,1}$ and $B_{2,3}$ are $(k+K)\times k$ and $(N-k)\times(N-K-k)$ lower $(K+1)$-diagonal matrices, respectively, $B_{1,2}$ is a $(k+K)\times K$ matrix with nonzero elements only on the lowest K diagonals, and $B_{2,2}$ is an $(N-k)\times K$ matrix with nonzero elements only on the highest K diagonals. Usually k is taken to be $\lfloor (N-K)/2 \rfloor$.

Let

$$B_{1,1} = \begin{pmatrix} Q_1 & S_1 \end{pmatrix} \begin{pmatrix} D_1 \\ 0 \end{pmatrix} W_1^T \quad \text{and} \quad B_{2,3} = \begin{pmatrix} Q_2 & S_2 \end{pmatrix} \begin{pmatrix} D_2 \\ 0 \end{pmatrix} W_2^T$$

be the SVDs of $B_{1,1}$ and $B_{2,3}$, respectively. Substituting into (21), we have

$$B = \begin{pmatrix} S_1 & Q_1 & & \\ & & Q_2 & S_2 \end{pmatrix} \begin{pmatrix} Z_{0,1} & 0 & 0 \\ Z_1 & D_1 & 0 \\ Z_2 & 0 & D_2 \\ Z_{0,2} & 0 & 0 \end{pmatrix} \begin{pmatrix} I_K & & W_1 & \\ & & & W_2 \end{pmatrix}^T, \tag{22}$$

where $Z_{0,1} = S_1^T B_{1,2}$, $Z_1 = Q_1^T B_{1,2}$, $Z_2 = Q_2^T B_{2,2}$, and $Z_{0,2} = S_2^T B_{2,2}$. There exists a $2K \times 2K$ orthogonal matrix

$$\begin{pmatrix} G_{1,1} & G_{1,2} \\ G_{2,1} & G_{2,2} \end{pmatrix}$$

such that

$$\begin{pmatrix} Z_{0,1} \\ Z_{0,2} \end{pmatrix} = \begin{pmatrix} G_{1,1} & G_{1,2} \\ G_{2,1} & G_{2,2} \end{pmatrix} \begin{pmatrix} Z_0 \\ 0 \end{pmatrix},$$

Downloaded 01/04/13 to 150.135.135.70. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

where Z_0 is a $K \times K$ lower triangular matrix. Substituting into (22), we have

$$B = \begin{pmatrix} S_1 G_{1,1} & Q_1 & & S_1 G_{2,1} \\ S_2 G_{1,2} & & Q_2 & S_2 G_{2,2} \end{pmatrix} \begin{pmatrix} Z_0 & 0 & 0 \\ Z_1 & D_1 & 0 \\ Z_2 & 0 & D_2 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} & W_1 & \\ I_K & & \\ & & W_2 \end{pmatrix}^T. \tag{23}$$

The middle matrix in (23) is lower triangular and can have nonzero elements only in the first K columns and on the diagonal. Partition

$$\begin{pmatrix} Z_0 \\ Z_1 \\ Z_2 \end{pmatrix} = \begin{pmatrix} \tilde{Z}_0 & 0 \\ \tilde{Z} & z \end{pmatrix},$$

where $\tilde{Z}_0$ is a $(K-1) \times (K-1)$ lower triangular matrix and $z = (r_0, z_1^T, z_2^T)^T$, with z_i being the last column of Z_i and r_0 being the last diagonal element of Z_0. Let $U\Omega V^T$ be the SVD of

$$M = \begin{pmatrix} r_0 & 0 & 0 \\ z_1 & D_1 & 0 \\ z_2 & 0 & D_2 \end{pmatrix}$$

computed using the algorithm described in §3. Then the middle matrix in (23) can be rewritten as

$$\begin{pmatrix} \tilde{Z}_0 & 0 \\ \tilde{Z} & U\Omega V^T \end{pmatrix} = \begin{pmatrix} I_{K-1} & \\ & U \end{pmatrix} \begin{pmatrix} \tilde{Z}_0 & 0 \\ U^T\tilde{Z} & \Omega \end{pmatrix} \begin{pmatrix} I_{K-1} & \\ & V \end{pmatrix}^T,$$

where the middle matrix is lower triangular and can have nonzero elements only in the first $K-1$ columns and on the diagonal. Thus the SVD of the middle matrix in (23) can be computed by applying this procedure K times.

To compute the SVDs of $B_{1,1}$ and $B_{2,3}$, we can recursively apply (21) and (23) to $B_{1,1}$ and $B_{2,3}$ until the subproblems are sufficiently small. These small subproblems are then solved using the Golub–Kahan algorithm [10], [12]. There can be at most $O(\log_2 N)$ levels of recursion. This algorithm takes $O(KN^3)$ time to compute both the singular values and the singular vectors. Similar to the bidiagonal case, there is an $O(K^2N^2)$ time divide-and-conquer algorithm for computing only the singular values. These times can be reduced to $O(KN^2)$ and $O(K^2N \log_2 N)$, respectively, by using the fast multipole method [16], [17]. These reduced times are better than the corresponding worst-case times ($O(N^3)$ and $O(KN^2)$) for the banded QR algorithm [26, p. 172].

REFERENCES

[1] P. Arbenz, *Divide-and-conquer algorithms for the computation of the SVD of bidiagonal matrices*, in Vector and Parallel Computing, J. Dongarra, I. Duff, P. Gaffney, and S. McKee, eds., Ellis Horwood, Chichester, 1989, pp. 1–10.

[2] ———, *Divide-and-conquer algorithms for the bandsymmetric eigenvalue problem*, Parallel Comput., 18 (1992), pp. 1105–1128.

[3] P. Arbenz and G. H. Golub, *On the spectral decomposition of Hermitian matrices modified by low rank perturbations with applications*, SIAM J. Matrix Anal. Appl., 9 (1988), pp. 40–58.

[4] C. F. Borges and W. B. Gragg, *A parallel divide and conquer algorithm for the generalized real symmetric definite tridiagonal eigenproblem*, in Numerical Linear Algebra and Scientific Computing, L. Reichel, A. Ruttan, and R. S. Varga, eds., de Gruyter, Berlin, 1993, pp. 10–28.

[5] J. R. BUNCH AND C. P. NIELSEN, *Updating the singular value decomposition*, Numer. Math., 31 (1978), pp. 111–129.
[6] J. R. BUNCH, C. P. NIELSEN, AND D. C. SORENSEN, *Rank-one modification of the symmetric eigenproblem*, Numer. Math., 31 (1978), pp. 31–48.
[7] J. CARRIER, L. GREENGARD, AND V. ROKHLIN, *A fast adaptive multipole algorithm for particle simulations*, SIAM J. Sci. Statist. Comput., 9 (1988), pp. 669–686.
[8] J. J. M. CUPPEN, *A divide and conquer method for the symmetric tridiagonal eigenproblem*, Numer. Math., 36 (1981), pp. 177–195.
[9] G. H. GOLUB, *Some modified matrix eigenvalue problems*, SIAM Rev., 15 (1973), pp. 318–334.
[10] G. H. GOLUB AND W. KAHAN, *Calculating the singular values and pseudo-inverse of a matrix*, SIAM J. Numer. Anal., 2 (1965), pp. 205–224.
[11] G. H. GOLUB, F. T. LUK, AND M. OVERTON, *A block Lanczos method for computing the singular values and corresponding singular vectors of a matrix*, ACM Trans. Math. Soft., 7 (1981), pp. 149–169.
[12] G. H. GOLUB AND C. REINSCH, *Singular value decomposition and least squares solutions*, Numer. Math., 14 (1970), pp. 403–420.
[13] G. H. GOLUB AND C. F. VAN LOAN, *Matrix Computations*, The Johns Hopkins University Press, Baltimore, MD, 1989.
[14] W. B. GRAGG, J. R. THORNTON, AND D. D. WARNER, *Parallel divide and conquer algorithms for the symmetric tridiagonal eigenproblem and bidiagonal singular value problem*, in Modelling and Simulation, W. G. Vogt and M. H. Mickle, eds., Vol. 23, Part 1, University of Pittsburgh School of Engineering, Pittsburgh, 1992, pp. 49–56.
[15] L. GREENGARD AND V. ROKHLIN, *A fast algorithm for particle simulations*, J. Comput. Phys., 73 (1987), pp. 325–348.
[16] M. GU, *Studies in Numerical Linear Algebra*, Ph.D. thesis, Department of Computer Science, Yale University, New Haven, CT, 1993.
[17] M. GU AND S. C. EISENSTAT, *A fast algorithm for updating the singular value decomposition*, manuscript.
[18] ———, *A divide-and-conquer algorithm for the symmetric tridiagonal eigenproblem*, SIAM J. Matrix Anal. Appl., 16 (1995), pp. 172–191.
[19] ———, *A stable and efficient algorithm for the rank-one modification of the symmetric eigenproblem*, SIAM J. Matrix Anal. Appl., 15 (1994), pp. 1266–1276.
[20] E. R. JESSUP, *Parallel Solution of the Symmetric Tridiagonal Eigenproblem*, Ph.D. thesis, Department of Computer Science, Yale University, New Haven, CT, 1989.
[21] E. R. JESSUP AND I. C. F. IPSEN, *Improving the accuracy of inverse iteration*, SIAM J. Sci. Statist. Comput., 13 (1992), pp. 550–572.
[22] E. R. JESSUP AND D. C. SORENSEN, *A parallel algorithm for computing the singular value decomposition of a matrix*, SIAM J. Matrix Anal. Appl., 15 (1994), pp. 530–548.
[23] W. KAHAN, *Rank-1 perturbed diagonal's eigensystem*, manuscript, July 1989.
[24] R.-C. LI, *Solving secular equations stably and efficiently*, Working Paper, Department of Mathematics, University of California at Berkeley, Oct. 1992.
[25] K. LÖWNER, *Über monotone Matrixfunktionen*, Math. Z., 38 (1934), pp. 177–216.
[26] B. N. PARLETT, *The Symmetric Eigenvalue Problem*, Prentice Hall, Englewood Cliffs, NJ, 1980.
[27] D. C. SORENSEN AND P. T. P. TANG, *On the orthogonality of eigenvectors computed by divide-and-conquer techniques*, SIAM J. Numer. Anal., 28 (1991), pp. 1752–1775.

Downloaded 01/04/13 to 150.135.135.70. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php