

ACCURATE EIGENVALUES AND SVDs OF TOTALLY NONNEGATIVE MATRICES*

PLAMEN KOEV†

Abstract. We consider the class of totally nonnegative (TN) matrices—matrices all of whose minors are nonnegative. Any nonsingular TN matrix factors as a product of nonnegative bidiagonal matrices. The entries of the bidiagonal factors parameterize the set of nonsingular TN matrices.

We present new $O(n^3)$ algorithms that, given the bidiagonal factors of a nonsingular TN matrix, compute its eigenvalues and SVD to high relative accuracy in floating point arithmetic, independent of the conventional condition number. All eigenvalues are guaranteed to be computed to high relative accuracy despite arbitrary nonnormality in the TN matrix.

We prove that the entries of the bidiagonal factors of a TN matrix determine its eigenvalues and SVD to high relative accuracy.

We establish necessary and sufficient conditions for computing the entries of the bidiagonal factors of a TN matrix to high relative accuracy, given the matrix entries.

In particular, our algorithms compute all eigenvalues and the SVD of TN Cauchy, Vandermonde, Cauchy–Vandermonde, and generalized Vandermonde matrices to high relative accuracy.

Key words. eigenvalue, singular value, high relative accuracy, totally positive matrix, totally nonnegative matrix, oscillatory matrix, sign regular matrix, bidiagonal matrix

AMS subject classifications. 65F15, 15A18

DOI. 10.1137/S0895479803438225

1. Introduction. The matrices with all minors nonnegative are called *totally nonnegative (TN)*. They appear in a wide area of problems and applications [6, 16, 19, 21, 22, 31] and can be notoriously ill-conditioned—the Hilbert matrix and the Vandermonde matrix with increasing nonnegative nodes are two examples of many.

When traditional algorithms are used to compute the eigenvalues or the singular values of an ill-conditioned TN matrix, only the largest eigenvalues and the largest singular values are computed with guaranteed relative accuracy. The tiny eigenvalues and singular values may be computed with no relative accuracy at all, even though they may be the only quantities of practical interest [8]. Their accurate computation using traditional algorithms is then only possible through an increase in the working precision, which may lead to a drastic increase in the computational time.

As our first major contribution we present new $O(n^3)$ algorithms that compute *all* eigenvalues and singular values of a nonsingular TN matrix to high relative accuracy. In particular, all computed eigenvalues and singular values, including the tiniest ones, must have correct sign and leading digits:

$$|\sigma_i - \hat{\sigma}_i| \leq O(\epsilon)\sigma_i \quad \text{and} \quad |\lambda_i - \hat{\lambda}_i| \leq O(\epsilon)|\lambda_i|.$$

The error bound for the computed eigenvalues must hold despite arbitrary nonnormality of A .

*Received by the editors December 1, 2003; accepted for publication (in revised form) by U. Helmke June 12, 2004; published electronically June 22, 2005. This work was supported by National Science Foundation grant DMS-0314286.

<http://www.siam.org/journals/simax/27-1/43822.html>

†Department of Mathematics, Massachusetts Institute of Technology, Cambridge, MA 02139 (plamen@math.mit.edu).

In contrast, traditional SVD and symmetric eigenvalue algorithms guarantee only small *absolute* errors in the computed singular values and eigenvalues of TN matrices

$$|\sigma_i - \hat{\sigma}_i| \leq O(\epsilon)\sigma_{\max} \quad \text{and} \quad |\lambda_i - \hat{\lambda}_i| \leq O(\epsilon)|\lambda_{\max}|.$$

Previously proposed nonsymmetric eigenvalue algorithms guarantee only the Perron root of a TN matrix to be computed to high relative accuracy [15]. The accuracy in the smaller eigenvalues is also affected by the angle between left and right eigenvectors.

Relative accuracy is lost in the traditional eigenvalue and SVD algorithms because of certain roundoff errors in floating point arithmetic: cancellation of significant digits during subtraction of approximate intermediate quantities. On the other side, relative accuracy is preserved in addition of nonnegative quantities, multiplication, division and taking of square roots.

The idea in our algorithms is to structure the computations in such a way that subtractions are avoided.

Throughout this paper we will use the word *accurately* to mean “to high relative accuracy.” We consider only *nonsingular* TN matrices.

The first step in our quest for accuracy is to choose a parameterization of the TN matrices to work with.

The matrix entries are a poor choice of parameters to represent a TN matrix because tiny relative perturbations in them can cause enormous relative perturbations in the eigenvalues and the singular values. For example, a small ϵ relative perturbation in the $(2, 2)$ entry of the 2×2 TN matrix

$$\begin{bmatrix} 1 & 1 \\ 1 & 1 + \epsilon \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & \\ & \epsilon \end{bmatrix} \cdot \begin{bmatrix} 1 & 1 \\ & 1 \end{bmatrix}$$

from $1 + \epsilon$ to $1 + 2\epsilon$ perturbs the smallest eigenvalue from $\epsilon/2 + O(\epsilon^2)$ to $\epsilon + O(\epsilon^2)$ —a 100% relative change. In other words, errors introduced just from storing a matrix in the computer may change the small eigenvalues or singular values utterly.

Instead, we will represent any nonsingular TN matrix A uniquely as a product of nonnegative bidiagonal matrices

$$A = L^{(1)}L^{(2)} \dots L^{(n-1)}DU^{(n-1)}U^{(n-2)} \dots U^{(1)},$$

where D is diagonal and $L^{(k)}$ and $U^{(k)}$ are lower and upper unit bidiagonal matrices, respectively. We denote this *bidiagonal decomposition* as $\mathcal{BD}(A)$ and discuss its properties in detail in section 2.2. A total of n^2 entries in $\mathcal{BD}(A)$ are nontrivial. These n^2 *independent* nonnegative entries parameterize the set of TN matrices.

Our second major contribution is to show that $\mathcal{BD}(A)$ determines the eigenvalues and the SVD of A accurately. In particular, the relative condition number of any eigenvalue with respect to perturbations in the entries of $\mathcal{BD}(A)$ is at most $2n^2$ independent of the angle between the left and right eigenvectors. The same structured condition number applies for the singular values. This means that our algorithms compute the eigenvalues and the singular values as accurately as they deserve.

Our algorithms take $\mathcal{BD}(A)$ as an input and perform a subtraction-free reduction of both the eigenvalue and the singular value problems to the bidiagonal singular value problem. The bidiagonal singular value problem is then solved to high relative accuracy using known means [12]. The above reduction uses only two types of matrix transformations, which we call *elementary elimination transformations* (EETs):

1. subtraction of a positive multiple of a row/column from the next in order to create a zero;

2. addition of a positive multiple of a row/column to the previous one, followed by scaling of both rows/columns.

Rather than performing the EETs directly on the matrix A , we perform them *implicitly* on the bidiagonal decomposition $\mathcal{BD}(A)$ in such a way that subtractions are not required; hence the accuracy is preserved.

To solve the eigenvalue problem we use a well-known process (which we discuss in section 2.3) to implicitly reduce a TN matrix A to a (nonsymmetric) TN tridiagonal form T using similarity EETs. We then implicitly replace $t_{i-1,i}$ and $t_{i,i-1}$ by $(t_{i-1,i}t_{i,i-1})^{1/2}$, making T symmetric without changing its eigenvalues. We compute the eigenvalues of T as the squares of the singular values of its Cholesky factor.

For the singular value problem we perform an implicit Golub–Kahan bidiagonalization using Givens rotations. The trick in preserving the accuracy is to represent each Givens rotation as a sequence of two EETs.

It is a natural question to ask whether an accurate bidiagonal decomposition may be obtained, given the matrix entries. We know of no simple answer for TN matrices, but for *totally positive* (TP) matrices (matrices all of whose minors are positive), we do. In section 3 we prove that if A is TP, then we can compute $\mathcal{BD}(A)$ accurately if and only if we can compute all initial minors of A accurately. The *initial minors* are minors that are contiguous and include the first row or the first column.

In particular, since accurate formulas exist for the initial minors of TP Vandermonde, Cauchy, Cauchy–Vandermonde, and generalized Vandermonde matrices, we can readily compute the eigenvalues and the singular values of these matrices accurately.

Our algorithms and theory apply with little or no change to TP, *oscillatory* (some power is TP), and inverses of TN matrices.

The idea of using bidiagonal decompositions to perform accurate matrix computations is not new. Such decompositions are used in every Björck–Pereyra method to obtain accurate solutions to structured TN linear systems (see section 2.4). Also, in [14] Parlett and Dhillon discard the entries of the tridiagonal matrix and choose to work with the bidiagonal Cholesky factors.

Organization of the paper. We review the properties of TN matrices and their bidiagonal decompositions in section 2. We present necessary and sufficient conditions for computing accurate bidiagonal decompositions in section 3. We describe how to perform EETs accurately in section 4. We present our new eigenvalue and singular value algorithms in sections 5 and 6, respectively. The perturbation theory and error analysis are in section 7. Finally, we present numerical experiments in section 8.

2. Background. In this section we review some well-known properties of TN matrices, their bidiagonal decompositions, Neville elimination, Björck–Pereyra methods, and the sensitivity of the SVD of a TN matrix to structured perturbations.

2.1. Properties of TN matrices. Excellent references on TN and related matrices are [2, 16, 21, 22, 25, 31]. We mention only some of the relevant properties.

The eigenvalues of a TN matrix are real and positive. The irreducible TN matrices are called *oscillatory*, and their eigenvalues are also distinct, as are their singular values. The product of two TN matrices is TN. A bidiagonal matrix with a positive diagonal and only one nonnegative off-diagonal entry is TN. All TN matrices are obtained by forming a product of such simple bidiagonal matrices, as we describe next.

2.2. Bidiagonal decomposition of TN matrices. A dense matrix A can be reduced to an upper triangular form by a variety of elimination methods. A fixed pivot row is used in Gaussian elimination where at each step an entry (say a_{ij}) is set to zero, resulting in the decomposition $A = K \cdot A'$, where $a'_{ij} = 0$ and K differs from the identity matrix only in its (i, j) th entry $l_{ij} = a_{ij}/a_{jj}$:

$$K = \begin{bmatrix} 1 & & & & \\ & \ddots & & & \\ & & \ddots & & \\ & & & \ddots & \\ l_{ij} & & & & 1 \end{bmatrix}.$$

At the end of this elimination process all matrices K are “assembled” in a unit lower triangular matrix L with subdiagonal entries l_{ij} .

If A is TN, then a different elimination approach, called *Neville elimination*, sheds more light on the structure of A [23, 38]. For completeness of the presentation, we describe this process in detail following [25].

The main difference between Gaussian and Neville eliminations is that the latter uses only adjacent rows or columns for elimination. For example the (i, j) th entry $a_{ij} \neq 0$ is eliminated by subtracting a multiple $m_{ij} = a_{ij}/a_{i-1,j}$ of the $(i-1)$ st row from the i th. In matrix form this results in the decomposition $A = E_i(m_{ij}) \cdot A'$, where E_i differs from the identity only in its $(i, i-1)$ entry:

$$(2.1) \quad E_i(m_{ij}) \equiv \begin{bmatrix} 1 & & & & \\ & \ddots & & & \\ & & 1 & & \\ & & m_{ij} & 1 & \\ & & & & \ddots \\ & & & & & 1 \end{bmatrix}.$$

The total nonnegativity is preserved by Neville elimination [23]; thus $m_{ij} \geq 0$. Once A is reduced to upper triangular form U , the same elimination process is applied to U using only adjacent columns. As a result A is factored as a product of $(n^2 - n)/2$ matrices of type (2.1), a diagonal matrix D , and $(n^2 - n)/2$ transposes of matrices of type (2.1).

The reader may have noticed that we bravely divided by $a_{i-1,j}$ in forming m_{ij} . This will never lead to division by zero because A is TN. Indeed, if $a_{i-1,k}$ is any nonzero (thus positive) entry in the $(i-1)$ st row of A , then the 2×2 minor of A consisting of rows $i-1$ and i and columns j and k ($j < k$) must be nonnegative. Therefore $a_{ij} > 0$ implies $a_{i-1,j} > 0$ and in turn $a_{i-1,j} = 0$ implies $a_{ij} = 0$. Equivalently, $m_{ij} = 0$ implies $m_{i+1,j} = \dots = m_{nj} = 0$, and thus

$$(2.2) \quad m_{ij} = 0 \text{ implies } \begin{cases} m_{kj} = 0 \text{ for all } k > i \text{ if } i > j; \\ m_{ik} = 0 \text{ for all } k > j \text{ if } i < j. \end{cases}$$

The matrices E_i have the following important properties:

$$(2.3) \quad \begin{aligned} E_i^{-1}(x) &= E_i(-x), \\ E_i(x)E_j(y) &= E_j(y)E_i(x), \text{ unless } |i-j| = 1 \text{ and } xy \neq 0, \\ E_i(x)E_i(y) &= E_i(x+y). \end{aligned}$$

$(n - 1)$ st to complete the similarity transformation. In the next step, we subtract a multiple of the $(n - 1)$ st column from the n th to create a zero in position $(1, n)$; then again, we complete the similarity by adding the same multiple of the n th row to the $(n - 1)$ st. The zero in position $(n, 1)$ is not disturbed by the last operation. Next, we introduce zeros in positions $(n - 1, 1)$ and $(1, n - 1)$ in a similar manner, which does not disturb the zeros already created in positions $(n, 1)$ and $(1, n)$. We continue the same process until all but the first two entries of the first row and column are zero. Then we apply the same process to the trailing $(n - 1) \times (n - 1)$ principal submatrix and so on until the matrix is reduced to tridiagonal form.

This process may be unstable for general nonsymmetric matrices, but it is our contribution to show that it is always stable for TN matrices.

Other analogous schemes for reduction to tridiagonal form also exist [37].

2.4. Björck–Pereyra methods. In 1970 Björck and Pereyra presented an algorithm for computing the solution to a Vandermonde system of equations $Vx = b$ in $O(n^2)$ time [4]. Although rooted in the process of Newton interpolation, their algorithm, in essence, computes an accurate decomposition of V^{-1} as a product of bidiagonal matrices and applies it to b . When the nodes of interpolation are nonnegative and increasing (i.e., when V is TN) and b has alternating sign pattern, there is no subtractive cancellation in forming x , and every entry of x is computed accurately. This phenomenon was first observed by Kahan in 1963 [30]; Higham later proved that it applies essentially to all Björck–Pereyra methods [27, 28].

Björck–Pereyra methods were also derived for Cauchy [5], Cauchy–Vandermonde [34], and generalized Vandermonde [13] matrices.

The decomposition $\mathcal{BD}(V)$ is readily and accurately obtainable from this representation of V^{-1} using the formulas (2.3) and inverting D (see also [25]). Therefore by using the algorithms from sections 5 and 6 one can compute accurate eigenvalues and SVD of any TN matrix for which Björck–Pereyra methods exist.

2.5. If A is TN, then $\mathcal{BD}(A)$ determines the SVD of A accurately. By using the Cauchy–Binet identity, Demmel et al. established in [11, section 9] that all minors of a TN matrix are determined accurately by the entries of $\mathcal{BD}(A)$. Thus so is its LDU decomposition resulting from Gaussian elimination with complete pivoting (every element of L , D , and U is a quotient of minors), and in turn so is the SVD. Using these results and ignoring δ^2 and higher-order terms, one can conclude that a δ relative perturbation in the n^2 nontrivial entries of $\mathcal{BD}(A)$ can cause a relative perturbation of at most $O(n^2\delta)$ in any minor (see section 7) and at most $O(n^2\kappa\delta)$ in any singular value. Here $\kappa \equiv \max\{\text{cond}(L), \text{cond}(U)\}$ is a bound on the condition number of L and U , which in practice never exceeds $O(n)$ [39, p. 213]. The angle $\theta(u_i, \hat{u}_i)$ between the i th singular vector and its perturbed counterpart is bounded by

$$\sin \theta(u_i, \hat{u}_i) \leq \frac{O(n^2\kappa\delta)}{\min_{j \neq i} |\sigma_i - \sigma_j| / \sigma_i}.$$

In section 7 we prove a slightly tighter perturbation bound for the singular values.

3. Necessary and sufficient conditions for computing accurate $\mathcal{BD}(A)$.

When can we compute an accurate $\mathcal{BD}(A)$, given A ? Neville elimination is the obvious choice, but it involves subtractions, and therefore likely loss of accuracy. On the other side, if A is TP, the entries of $\mathcal{BD}(A)$ are products and quotients of initial minors of A and can be computed accurately if these minors can be. We recall that a minor is *initial* if it is contiguous and includes the first row or the first column.

We will use MATLAB [35] notation for submatrices: $A(i : j, k : l)$ will denote the submatrix of A consisting of rows i through j and columns k through l .

PROPOSITION 3.1. *If A is TP, then the following expressions are valid for $\mathcal{BD}(A)$:*

$$(3.1) \quad l_i^{(k)} = \frac{\det A(q+1 : i+1, 1 : i-q+1)}{\det A(q+1 : i, 1 : i-q)} \cdot \frac{\det A(q : i-1, 1 : i-q)}{\det A(q : i, 1 : i-q+1)},$$

$$(3.2) \quad u_i^{(k)} = \frac{\det A(1 : i-q+1, q+1 : i+1)}{\det A(1 : i-q, q+1 : i)} \cdot \frac{\det A(1 : i-q, q : i-1)}{\det A(1 : i-q+1, q : i)},$$

$$(3.3) \quad d_i = \frac{\det A(1 : i, 1 : i)}{\det A(1 : i-1, 1 : i-1)},$$

where $q = n - k$; $i \geq q = n - k$ in (3.1) and (3.2), and $l_i^{(k)} = u_i^{(k)} = 0$ for $i < n - k$.

Therefore, being able to compute all n^2 initial minors of A accurately is a necessary and sufficient condition for computing $\mathcal{BD}(A)$ accurately.

Proof. Expressions (3.1), (3.2), and (3.3) follow directly from Lemma 2.6(1) in [23]. For the second part ‘‘sufficiency’’ is obvious, as is ‘‘necessity’’ for the leading principal minors: $A(1 : i, 1 : i) = d_1 d_2 \cdots d_i$. Next we write $m \equiv i - q$ in (3.2) to obtain

$$(3.4) \quad \det A(1 : m+1, q+1 : q+m+1) = \det A(1 : m+1, q : q+m) \\ \times \left[u_i^{(n-q)} \cdot \frac{\det A(1 : m, q+1 : q+m)}{\det A(1 : m, q : q+m-1)} \right].$$

The expression in the brackets involves only initial minors of order m . We now use (3.4) and induction on m from 0 to $n-1$ and q from 1 to $n-m-1$ to conclude that all initial minors of A that include the first row are computable accurately (since (3.4) involves only multiplications and divisions) at only three arithmetic operations per initial minor. Analogously we draw the same conclusion about the initial minors that include the first column by starting with (3.1). \square

Proposition 3.1 allows us to compute accurate bidiagonal decompositions of TP structured matrices. For example, the Vandermonde and Cauchy matrices

$$V = \left[x_i^{j-1} \right]_{i,j=1}^n \quad \text{and} \quad C = \left[\frac{1}{x_i + y_j} \right]_{i,j=1}^n$$

are TP when $0 < x_1 < x_2 < \cdots < x_n$, and $x_1 < x_2 < \cdots < x_n$, $y_1 < y_2 < \cdots < y_n$, $x_1 + y_1 > 0$, respectively. For $\mathcal{BD}(V)$ we have

$$(3.5) \quad d_i = \prod_{j=1}^{i-1} (x_i - x_j), \quad l_i^{(k)} = \prod_{j=n-k}^{i-1} \frac{x_{i+1} - x_{j+1}}{x_i - x_j}, \quad u_i^{(k)} = x_{i+k-n+1},$$

where $l_i^{(k)} = u_i^{(k)} = 0$ for $i < n - k$ and for $\mathcal{BD}(C)$ we have

$$(3.6) \quad d_i = \prod_{k=1}^{i-1} \frac{(x_i - x_k)(y_i - y_k)}{(x_i + y_k)(y_i + x_k)}, \\ l_i^{(k)} = \frac{x_{n-k} + y_{i-n+k+1}}{x_i + y_{i-n+k+1}} \prod_{l=n-k}^{i-1} \frac{x_{i+1} - x_{l+1}}{x_i - x_l} \cdot \prod_{l=1}^{i-n+k-1} \frac{x_i + y_l}{x_{i+1} + y_l}, \\ u_i^{(k)} = \frac{y_{n-k} + x_{i-n+k+1}}{y_i + x_{i-n+k+1}} \prod_{l=n-k}^{i-1} \frac{y_{i+1} - y_{l+1}}{y_i - y_l} \cdot \prod_{l=1}^{i-n+k-1} \frac{y_i + x_l}{y_{i+1} + x_l},$$

where $l_i^{(k)} = u_i^{(k)} = 0$ for $i < n - k$.

The subtractions in formulas (3.5) and (3.6) involve only initial (thus exact) data and will not lead to subtractive cancellation. In section 7 we prove that $\mathcal{BD}(V)$ and $\mathcal{BD}(C)$, and in turn the eigenvalues and the singular values of V and C , will be computed accurately. Similar accurate formulas for $\mathcal{BD}(A)$ may be obtained when A is a TP generalized Vandermonde [13] or a Cauchy–Vandermonde [33] matrix.

The formulas (3.1)–(3.3) are not, in general, valid for TN matrices (which are the closure of TP matrices). We do not know of necessary and sufficient conditions for being able to compute $\mathcal{BD}(A)$ accurately when A is TN, but not TP.

4. Computing with $\mathcal{BD}(A)$. In this section we show how to accurately apply an EET to $\mathcal{BD}(A)$. In other words, if A is TN and A' is obtained from A by applying an EET, we will show how to compute $\mathcal{BD}(A')$ directly from $\mathcal{BD}(A)$ without performing any subtractions.

It is convenient to store the n^2 nontrivial entries of $\mathcal{BD}(A)$ as an $n \times n$ square array. In position (i, j) we store the multiplier m_{ij} used to eliminate the (i, j) th entry in A , and on the diagonal we store D :

$$(\mathcal{BD}(A))_{ij} = \begin{cases} d_i, & i = j, \\ m_{ij}, & i \neq j; \end{cases} \quad \text{equivalently} \quad (\mathcal{BD}(A))_{ij} = \begin{cases} d_i, & i = j, \\ l_{i-1}^{(n-i+j)}, & i > j, \\ u_{j-1}^{(n-j+i)}, & i < j. \end{cases}$$

From now on we will use $\mathcal{BD}(A)$ to refer to either the actual bidiagonal decomposition (2.4) or the $n \times n$ array used for its compact storage, but it should cause no confusion.

Example 4.1. The bidiagonal decomposition of the 3×3 TN matrix

$$\begin{bmatrix} 1 & 2 & 6 \\ 4 & 13 & 69 \\ 28 & 131 & 852 \end{bmatrix} = \begin{bmatrix} 1 & & \\ & 1 & \\ & 7 & 1 \end{bmatrix} \begin{bmatrix} 1 & & \\ 4 & 1 & \\ & 8 & 1 \end{bmatrix} \begin{bmatrix} 1 & & \\ & 5 & \\ & & 9 \end{bmatrix} \begin{bmatrix} 1 & 2 & \\ & 1 & 6 \\ & & 1 \end{bmatrix} \begin{bmatrix} 1 & & \\ & 1 & 3 \\ & & 1 \end{bmatrix}$$

is stored as

$$\mathcal{BD} \left(\begin{bmatrix} 1 & 2 & 6 \\ 4 & 13 & 69 \\ 28 & 131 & 852 \end{bmatrix} \right) = \begin{Bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{Bmatrix}.$$

We can formally transpose $\mathcal{BD}(A)$ to obtain $\mathcal{BD}(A^T)$:

$$A^T = (U^{(1)})^T (U^{(2)})^T \dots (U^{(n-1)})^T D (L^{(n-1)})^T (L^{(n-2)})^T \dots (L^{(1)})^T,$$

which we write as $\mathcal{BD}(A^T) = (\mathcal{BD}(A))^T$.

4.1. Subtracting a positive multiple of a row from the next to create a zero. We now show that in an elimination process of reducing a matrix to upper triangular form, subtracting a row from the next in order to create a zero is equivalent to simply setting an entry in $\mathcal{BD}(A)$ to zero and performing *no arithmetic*.

PROPOSITION 4.2. *Let A be a TN matrix such that $A(k-1 : n, 1 : l) = 0$, except for $a_{k-1,l} > 0$ and $a_{kl} > 0$, where $k > l$. Let A' be obtained from A by subtracting a*

multiple of the $(k-1)$ st row from the k th in order to create a zero in position (k, l) . Then $\mathcal{BD}(A')$ is obtained from $\mathcal{BD}(A)$ by setting the (k, l) th entry of $\mathcal{BD}(A)$ to zero.

Proof. Clearly $A' = E_k(-a_{kl}/a_{k-1,l}) \cdot A$. Because of the zeros in the lower left-hand corner of A , the only way that the process of Neville elimination on A differs from the process of Neville elimination on A' is that the former involves subtracting a multiple $m_{kl} = a_{kl}/a_{k-1,l}$ of the $(k-1)$ st row of A from the k th, whereas the latter does not. Therefore $\mathcal{BD}(A)$ equals $\mathcal{BD}(A')$ except for $(\mathcal{BD}(A))_{kl} = a_{kl}/a_{k-1,l} = m_{kl}$, whereas $(\mathcal{BD}(A'))_{kl} = 0$. \square

4.2. Adding a positive multiple of a column/row to the previous one.

Next, we look at how $\mathcal{BD}(A)$ changes when we add a positive multiple (x/y) of the i th column to the $(i-1)$ st and scale columns $i-1$ and i by y and $1/y$, respectively.

In matrix form this is equivalent to forming the product $AJ_i(x, y)$, where

$$J_i(x, y) \equiv \begin{bmatrix} 1 & & & & & & & & \\ & \ddots & & & & & & & \\ & & y & & & & & & \\ & & x & 1/y & & & & & \\ & & & & \ddots & & & & \\ & & & & & & & & 1 \end{bmatrix}$$

is a lower bidiagonal matrix that differs from the identity matrix only in the entries x , y , and $1/y$ in positions $(i, i-1)$, $(i-1, i-1)$, and (i, i) , respectively.

Adding a multiple of a row to the previous one followed by scaling of both rows is analogous and computed as $\mathcal{BD}(J_i^T(x, y) \cdot A) = \mathcal{BD}(A^T \cdot J_i(x, y))^T$.

THEOREM 4.3. *Let A be TN. Given $x > 0, y > 0$, and $\mathcal{BD}(A)$, the decomposition $\mathcal{BD}(A \cdot J_i(x, y))$ can be computed in at most $4n + 2i + 12$ arithmetic operations without performing any subtractions.*

Proof. We start with

$$(4.1) \quad AJ_i(x, y) = L^{(1)}L^{(2)}L^{(3)} \dots L^{(n-1)}DU^{(n-1)} \dots U^{(2)}U^{(1)} \cdot J_i(x, y).$$

We will consecutively chase the ‘‘bulge’’ $J_i(x, y)$ to the left of the decomposition (4.1), preserving the structure of the bidiagonal factors:

$$(4.2) \quad AJ_i = L^{(1)} \dots L^{(i-2)}L^{(i-1)}L^{(i)} \dots L^{(n-2)}L^{(n-1)}DU^{(n-1)} \dots U^{(3)}U^{(2)}U^{(1)}J_i$$

$$(4.3) \quad = L^{(1)} \dots L^{(i-2)}L^{(i-1)}L^{(i)} \dots L^{(n-2)}L^{(n-1)}DU^{(n-1)} \dots U^{(3)}U^{(2)}J_i^{(1)}U^{(1)}$$

$$= L^{(1)} \dots L^{(i-2)}L^{(i-1)}L^{(i)} \dots L^{(n-2)}L^{(n-1)}DU^{(n-1)} \dots U^{(3)}J_i^{(2)}U^{(2)}U^{(1)}$$

$$= \dots$$

$$(4.4) \quad = L^{(1)} \dots L^{(i-2)}L^{(i-1)}L^{(i)} \dots L^{(n-2)}L^{(n-1)}DJ_i^{(n-1)}U^{(n-1)} \dots U^{(2)}U^{(1)}$$

$$(4.5) \quad = L^{(1)} \dots L^{(i-2)}L^{(i-1)}L^{(i)} \dots L^{(n-2)}L^{(n-1)}J_i^{(n)}DU^{(n-1)} \dots U^{(2)}U^{(1)}$$

$$= L^{(1)} \dots L^{(i-2)}L^{(i-1)}L^{(i)} \dots L^{(n-3)}L^{(n-2)}J_{i+1}^{(n+1)}L^{(n-1)}DU^{(n-1)} \dots U^{(2)}U^{(1)}$$

$$= L^{(1)} \dots L^{(i-2)}L^{(i-1)}L^{(i)} \dots L^{(n-3)}J_{i+2}^{(n+2)}L^{(n-2)}L^{(n-1)}DU^{(n-1)} \dots U^{(2)}U^{(1)}$$

$$= \dots$$

$$(4.6) \quad = L^{(1)} \dots L^{(i-2)}L^{(i-1)}J_n^{(2n-i)}L^{(i+1)} \dots L^{(n-2)}L^{(n-1)}DU^{(n-1)} \dots U^{(2)}U^{(1)}$$

$$(4.7) \quad = L^{(1)} \dots L^{(i-2)}L^{(i-1)}L^{(i)} \dots L^{(n-2)}L^{(n-1)}DU^{(n-1)} \dots U^{(2)}U^{(1)}.$$

The factors that are transformed at each step of the above transformations are underlined. The matrix $J_m^{(k)}$ equals $J_m(x_k, y_k)$ for some x_k and y_k . The matrices $\mathcal{L}^{(k)}$, \mathcal{D} , and $\mathcal{U}^{(k)}$, $k = 1, 2, \dots, n-1$, are unit lower bidiagonal, diagonal and unit upper diagonal, respectively, with nonzero pattern consistent with Theorem 2.1.

We will now explain how to obtain (4.3) from (4.2) and so on until we obtain (4.7). Then we will prove that (4.7) is in fact $\mathcal{BD}(AJ_i(x, y))$.

We start with the transformations (4.2)–(4.4). At each step we are given matrices

$$U = \begin{bmatrix} 1 & u_1 & & & \\ & 1 & u_2 & & \\ & & \ddots & \ddots & \\ & & & 1 & u_{n-1} \\ & & & & 1 \end{bmatrix}, \quad u_j \geq 0, \quad j = 1, 2, \dots, n-1,$$

and $J_i(x, y)$, $x > 0, y > 0$. We need to compute matrices

$$\mathcal{U} = \begin{bmatrix} 1 & u'_1 & & & \\ & 1 & u'_2 & & \\ & & \ddots & \ddots & \\ & & & 1 & u'_{n-1} \\ & & & & 1 \end{bmatrix}, \quad u'_j \geq 0, \quad j = 1, 2, \dots, n-1,$$

and $J_i(x', y')$ such that

$$(4.8) \quad J_i(x', y') \cdot \mathcal{U} = U \cdot J_i(x, y).$$

By comparing the entries on both sides of (4.8), we obtain

$$(4.9) \quad \begin{aligned} x' &= x, \\ y' &= y + u_{i-1}x, \\ u'_{i-2} &= u_{i-2}y, \\ u'_{i-1} &= u_{i-1}/(yy'), \\ u'_i &= u_i y', \\ u'_j &= u_j \text{ for } j \notin \{i-2, i-1, i\}. \end{aligned}$$

The expressions (4.9) involve no subtractions and require only six arithmetic operations. No division by 0 can occur because $y > 0$. The inequalities $u_j \geq 0$, $x > 0$, $y > 0$ imply $u'_j \geq 0$, $x' > 0$, $y' > 0$. Also $u'_j = 0$ only when $u_j = 0$, $j = 1, 2, \dots, n-1$.

All transformations (4.2)–(4.4) are performed according to (4.8).

The reader familiar with the LR algorithm may have noticed that the equation (4.8) is equivalent to one step of LR, performed accurately using differential quotient-difference recurrences [17, 36, 39].

Now we turn our attention to the transformation from (4.4) to (4.5):

$$(4.10) \quad J_i(x', 1) \cdot \mathcal{D} = D \cdot J_i(x, y).$$

This transformation is straightforward and subtraction-free:

$$(4.11) \quad \begin{aligned} \mathcal{D} &= \text{diag}(d_1, d_2, \dots, d_{i-2}, d_{i-1}y, d_i/y, d_{i+1}, \dots, d_n), \\ x' &= d_i x / (d_{i-1}y). \end{aligned}$$

No division by zero can occur in (4.11) because $yd_{i-1} > 0$.

In the transformations (4.5)–(4.7) we are given $x > 0$ and a matrix

$$L = \begin{bmatrix} 1 & & & & & \\ l_1 & 1 & & & & \\ & \ddots & \ddots & & & \\ & & l_{n-2} & 1 & & \\ & & & l_{n-1} & 1 & \end{bmatrix}, \quad l_j \geq 0, \quad j = 1, 2, \dots, n-1.$$

We need to compute $x' \geq 0$ and a matrix

$$\mathcal{L} = \begin{bmatrix} 1 & & & & & \\ l'_1 & 1 & & & & \\ & \ddots & \ddots & & & \\ & & l'_{n-2} & 1 & & \\ & & & l'_{n-1} & 1 & \end{bmatrix}, \quad l'_j \geq 0, \quad j = 1, 2, \dots, n-1,$$

such that

$$(4.12) \quad J_{k+1}(x', 1) \cdot \mathcal{L} = L \cdot J_k(x, 1).$$

By convention we set $J_{n+1} \equiv I$. The bulge chasing is over once $J_{k+1} = I$. This happens when $k = n$ or possibly earlier if x' becomes zero.

By comparing the entries on both sides of (4.12) we obtain

$$(4.13) \quad l'_j = l_j \quad \text{if } j \notin \{k-1, k\},$$

$$(4.14) \quad l'_{k-1} = l_{k-1} + x,$$

$$(4.15) \quad l'_k = l_{k-1}l_k/l'_{k-1} \quad \text{if } k \leq n-1,$$

$$(4.16) \quad x' = xl_k/l'_{k-1} \quad \text{if } k \leq n-1.$$

The expressions (4.13)–(4.16) are subtraction-free and require only four arithmetic operations (we only need compute l_k/l'_{k-1} once in (4.15)–(4.16)). The inequalities $x > 0$ and $l_j \geq 0$ imply $x' \geq 0$ and $l'_j \geq 0$; thus division by 0 cannot occur.

All transformations (4.5)–(4.7) are performed according to (4.12).

To complete the proof we need to show that (4.7) is the unique $\mathcal{BD}(A \cdot J_i(x, y))$. It suffices to verify that the conditions of Theorem 2.1 are satisfied.

Conditions 1, 2, and 3 clearly are satisfied. The second part of condition 4 ($u_j^{(k)} = 0$ implies $u_{j+s}^{(k-s)} = 0$ for $s = 1, 2, \dots, k-1$) is also satisfied, because no zeros or nonzeros are introduced in the upper bidiagonal factors $\mathcal{U}^{(k)}$ in steps (4.2) to (4.5).

We will now establish that the first part of condition 4 holds as well. Let $l'_j^{(m)}, j = 1, 2, \dots, n-1$, be the off-diagonal entries in $\mathcal{L}^{(m)}$, and assume that some $l'_j^{(m)} = 0$. We need to prove that $l'_{j+s}^{(m-s)} = 0$ for $s = 1, \dots, m-1$. It suffices to show that $l'_{j+1}^{(m-1)} = 0$, since the rest will follow by induction on s .

The entry $l'_j^{(m)}$ could not have been computed using (4.14) because $x > 0$. It must have been computed using (4.13) or (4.15).

If (4.13) was used, then (4.13) will also be used to compute $l'_{j+1}^{(m-1)} = l'_{j+1}^{(m)}$ on the next step of (4.12). Condition 4 of Theorem 2.1 along with $l'_j^{(m)} = 0$ implies $l'_{j+1}^{(m-1)} = 0$. Therefore $l'_{j+1}^{(m-1)} = 0$.

If (4.15) was used to compute $l_j^{(m)}$, then $0 = l_j^{(m)} = l_{j-1}^{(m)}l_j^{(m)}/(l_{j-1}^{(m)} + x)$. Thus $l_{j-1}^{(m)} = 0$ or $l_j^{(m)} = 0$. Condition 4 for $\mathcal{BD}(A)$ now implies $l_j^{(m-1)} = 0$ or $l_{j+1}^{(m-1)} = 0$. Either way, on the next step of (4.12), (4.15) gives $l_{j+1}^{(m-1)} = l_j^{(m-1)}l_{j+1}^{(m-1)}/l_j^{(m-1)} = 0$.

To complete the proof it remains only to count the number of arithmetic operations. Since $u_j^{(k)} = 0$ for $j < n - k$, the matrix J_i commutes with $U^{(1)}, U^{(2)}, \dots, U^{(n-i-1)}$. Therefore (4.9) are performed at most i times; (4.11) are performed once, and (4.14)–(4.16) are performed at most $n - i + 1$ times. The total number of arithmetic operations does not exceed $6i + 4 + 4(n - i + 1) = 4n + 2i + 8$. \square

ALGORITHM 4.4. *Given $\mathcal{BD}(A)$, the following algorithm implements the procedure from Theorem 4.3 and accurately computes $\mathcal{BD}(A \cdot J_i(x, y))$ in not more than $4n + 2i + 8$ arithmetic operations.*

```

function B = TNAddToPrevious(B, x, y, i)
n = size(B, 1)
if i < n, then
    B(1, i + 1) = B(1, i + 1) * y
end
for j = 1 : i - 1
    z = y + B(j, i) * x
    if j > 1, then
        B(j - 1, i - 1) = B(j - 1, i - 1) * y
    end
    B(j, i) = B(j, i) / y / z
    if i < n, then
        B(j + 1, i + 1) = B(j + 1, i + 1) * z
    end
    y = z
end
B(i - 1, i - 1) = B(i - 1, i - 1) * y
x = B(i, i) * x / B(i - 1, i - 1)
B(i, i) = B(i, i) / y
j = i
while (j < n) and (x > 0)
    z = B(j, i - 1)
    B(j, i - 1) = z + x
    y = B(j + 1, i) / B(j, i - 1)
    B(j + 1, i) = y * z
    x = y * x
    j = j + 1
end
B(n, i - 1) = B(n, i - 1) + x

```

4.3. Givens rotations. Let A be TN and G be a Givens rotation intended to create a zero in the process of Golub–Kahan bidiagonalization of A . We will now demonstrate how to compute $\mathcal{BD}(GA)$, given $\mathcal{BD}(A)$.

The trick is to represent G as a sequence of two EETs. Indeed, applying a Givens rotation to create a zero in position (say $(n, 1)$) is equivalent to (1) subtracting a (positive) multiple of the $(n-1)$ st row from the n th in order to create a zero in position $(n, 1)$, followed by (2) adding a positive multiple of the n th row to the $(n-1)$ st, and finally (3) scaling the last two rows.

Let

$$G = \begin{bmatrix} c & s \\ -s & c \end{bmatrix},$$

where $s^2 + c^2 = 1$, be a Givens rotation used to set the $(n, 1)$ entry of A to zero. Then

$$G \cdot \begin{bmatrix} a_{n-1,1} & a_{n-1,2} \\ a_{n1} & a_{n2} \end{bmatrix} = \begin{bmatrix} c & s \\ -s & c \end{bmatrix} \cdot \begin{bmatrix} a_{n-1,1} & a_{n-1,2} \\ a_{n1} & a_{n2} \end{bmatrix} = \begin{bmatrix} a_{n-1,1}^{(1)} & a_{n-1,2}^{(1)} \\ 0 & a_{n2}^{(1)} \end{bmatrix}.$$

A simple calculation shows that $c = 1/\sqrt{1+x^2}$ and $s = x/\sqrt{1+x^2}$, where

$$x = a_{n1}/a_{n-1,1} = l_{n-1}^{(1)} = (\mathcal{BD}(A))_{n1}$$

is the only nonzero off-diagonal entry in $L^{(1)}$ of $\mathcal{BD}(A)$. Now write G as

$$G = \begin{bmatrix} 1/c & cx \\ 0 & c \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 \\ -x & 1 \end{bmatrix}.$$

Therefore applying a Givens rotation to the last two rows of A in order to create a zero in position $(n, 1)$ is equivalent to the multiplication

$$A^{(1)} = J_n^T(cx, 1/c) \cdot E_n(-x) \cdot A.$$

Forming $A' = E_n(-x) \cdot A$ is equivalent to subtracting a positive multiple of the $(n-1)$ st row from the n th in order to create a zero in position $(n, 1)$. Therefore we obtain $\mathcal{BD}(A')$ from $\mathcal{BD}(A)$ by setting $(\mathcal{BD}(A))_{n1}$ to zero. The multiplication $A^{(1)} = J_n^T(cx, 1/c) \cdot A'$ is performed using Algorithm 4.4 to compute

$$\mathcal{BD}(A^{(1)}) = (\mathcal{BD}((A')^T \cdot J_n(cx, 1/c)))^T.$$

Applying a Givens rotation on the right is analogous, as is the application of subsequent Givens rotations in the reduction of A to bidiagonal form.

5. The eigenvalue problem. In this section we present our algorithm for computing accurate eigenvalues of a TN matrix A , given $\mathcal{BD}(A)$.

We first implicitly reduce A to tridiagonal form by using similarity EETs as described in section 2.3. We preserve the accuracy in this reduction by applying the EETs to $\mathcal{BD}(A)$ as described in section 4. We obtain the decomposition $\mathcal{BD}(T)$ of a tridiagonal matrix T , which has only three nontrivial factors:

$$T = L^{(n-1)} \cdot D \cdot U^{(n-1)}.$$

The matrix T need not be symmetric and may not have a full set of eigenvectors. It does, however, have the same eigenvalues as the *symmetric* tridiagonal matrix $\bar{T} \equiv \bar{L}^{(n-1)} D \bar{U}^{(n-1)}$, where

$$\bar{l}_i^{(n-1)} = \bar{u}_i^{(n-1)} \equiv \sqrt{l_i^{(n-1)} u_i^{(n-1)}}, \quad i = 1, 2, \dots, n-1.$$

(\bar{T} and T have the same characteristic polynomial; they are, however, only similar if $\bar{l}_i^{(n-1)} = 0$ whenever $u_i^{(n-1)} = 0$ and vice versa.) The eigenvalues of \bar{T} are computed accurately as the squares of the singular values of its Cholesky factor $C = D^{1/2} \bar{U}^{(n-1)}$

using the LAPACK [1] routine `DLASQ1` [12, 17]. (C is accurately obtained from $\mathcal{BD}(T)$ at the cost of $2n - 2$ multiplications and $2n - 1$ square roots.)

The matrix C is thus formed without performing any subtractions at the cost of at most

$$2 \sum_{i=1}^{n-2} \sum_{j=i+2}^n (4n + 2j + 8) + 4n - 3 = \frac{16}{3}n^3 + O(n^2)$$

arithmetic operations. The routine `DLASQ1` costs an additional $O(n^2)$ operations; thus the cost of Algorithm 5.1 is at most $\frac{16}{3}n^3 + O(n^2)$.

In section 7 we prove that all eigenvalues of A are computed accurately independent of the angle between the left and right eigenvectors.

The eigenvectors (when they all exist) may certainly be accumulated, although questions remain about their accuracy in the nonsymmetric case—see section 7.

ALGORITHM 5.1 (accurate eigenvalues of a TN matrix). *Given $B = \mathcal{BD}(A)$ of a TN matrix A , the following algorithm computes the eigenvalues of A accurately.*

```

function TNEigenvalues(B)
  n = size(B,1)
  for i = 1 : n - 2
    for j = n : -1 : i + 2
      x = B(j, i)
      B(j, i) = 0
      B = TNAddToPrevious(B, x, 1, j)
      x = B(i, j)
      B(i, j) = 0
      B = (TNAddToPrevious(BT, x, 1, j))T
    end
  end
  C = diag(sqrt(diag(B)))
  for i = 1 : n - 1
    C(i, i + 1) = sqrt(B(i, i + 1) * B(i + 1, i) * B(i, i))
  end
  TNEigenvalues = (DLASQ1(C))2

```

6. The singular value problem. In this section we present our algorithm for computing accurate singular values of a TN matrix A , given $\mathcal{BD}(A)$.

We first perform an implicit Golub–Kahan bidiagonalization on A by starting with $\mathcal{BD}(A)$ and performing Givens rotations as described in section 4.3. We implicitly introduce zeros in the first column of A below the main diagonal, then in the first row to the right of the (1,2) entry. We then proceed by induction until $\mathcal{BD}(A)$ is reduced to $\mathcal{BD}(F)$, where F is bidiagonal. The decomposition $\mathcal{BD}(F)$ contains only two nontrivial factors: D and $U^{(n-1)}$, so we recover F from $\mathcal{BD}(F)$ at the cost of $n - 1$ multiplications. The computation so far is subtraction-free. Finally, we compute the singular values of F accurately using the LAPACK [1] routine `DLASQ1`.

By performing complexity analysis similar to that of section 5, we conclude that Algorithm 6.1 costs at most $\frac{16}{3}n^3 + O(n^2)$ arithmetic operations for the reduction to bidiagonal form F and $O(n^2)$ operations for `DLASQ1`.

The singular vectors can be computed by accumulating the Givens rotations on the left and the right with the singular vector matrices of F .

In section 7 we prove that the computed singular values and singular vectors are accurate.

ALGORITHM 6.1 (accurate singular values of a TN matrix). *Given $B = \mathcal{BD}(A)$ of a TN matrix A , the following algorithm computes the singular values of A accurately.*

```

function TNSingularValues(B)
    n = size(B,1)
    for i = 1 : n - 1
        for j = n : -1 : i + 1
            x = B(j,i)
            B(j,i) = 0
            c = sqrt(1 + x^2)
            B = (TNAddToPrevious(B^T, x/c, c, j))^T
        end
        for j = n : -1 : i + 2
            x = B(i,j)
            B(i,j) = 0
            c = sqrt(1 + x^2)
            B = TNAddToPrevious(B, x/c, c, j)
        end
    end
    F = diag(diag(B))
    for i = 1 : n - 1
        F(i, i + 1) = F(i, i) * B(i, i + 1)
    end
    TNSingularValues = DLASQ1(F)

```

7. Perturbation theory and error analysis. In this section we prove that if a TN matrix A is represented as a product of nonnegative bidiagonal matrices, then small relative perturbations in the entries of the bidiagonal factors cause only small relative perturbations in the eigenvalues and the SVD of A .

In particular, $\mathcal{BD}(A)$ determines the eigenvalues and the SVD of A accurately, which justifies computing them accurately. We believe that the perturbation result for the eigenvalues is new; the result for the SVD is known (see section 2.5 and Demmel et al. [11, section 9]), but we improve the perturbation bounds for the singular values.

We then use these perturbation results to prove that Algorithms 5.1 and 6.1 compute the eigenvalues and the singular values of a TN matrix accurately in floating point arithmetic; the singular vectors that may be accumulated by Algorithm 6.1 will also be accurate in the appropriate sense. The eigenvector problem remains open in the nonsymmetric case.

We finish the section with the perturbation theory and error analysis for the eigenvalue and singular value problems for TP Vandermonde and TP Cauchy matrices.

For our error analysis we use the standard model of floating point arithmetic [29, section 2.2]:

$$fl(x \odot y) = (x \odot y)(1 + \delta)^{\pm 1}, \quad \text{where } |\delta| \leq \epsilon, \quad \text{and } \odot \in \{+, -, \times, /\}.$$

We will accumulate relative errors or relative perturbations in the style of Higham [29]: If $|\delta_i| \leq \delta \ll 1$ and $\rho_i = \pm 1$, then

$$\left| \prod_{i=1}^k (1 + \delta_i)^{\rho_i} - 1 \right| \leq \frac{k\delta}{1 - k\delta}.$$

Also, if $|\delta_1| \leq m\delta/(1 - m\delta)$ and $|\delta_2| \leq k\delta/(1 - k\delta)$, then

$$|(1 + \delta_1)(1 + \delta_2) - 1| \leq \frac{(m + k)\delta}{1 - (m + k)\delta}.$$

Eigenvalues and singular values. We will extract information about the perturbation of the eigenvalues and the singular values of a TN matrix A from its k th compound or associated matrix $\mathcal{A}^{(k)}$ [20, 21]. The entries of $\mathcal{A}^{(k)}$ are all k th order minors of A . The eigenvalues of $\mathcal{A}^{(k)}$ are all possible products of k eigenvalues of A [21, Thm. 23, p. 65]. We will now prove that the same is true for the singular values.

LEMMA 7.1. *Let $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_n \geq 0$ be the singular values of a matrix A . Then the singular values of the k th compound matrix $\mathcal{A}^{(k)}$ are all possible products of the numbers $\sigma_1, \sigma_2, \dots, \sigma_n$ taken k at a time.*

Proof. Let $A = USV^T$ be the SVD of A , where $S = \text{diag}(\sigma_1, \sigma_2, \dots, \sigma_n)$. The k th compound matrix of a product of matrices equals the product of the k th compound matrices of the factors [21, p. 64]. Therefore $\mathcal{U}^{(k)}$ and $\mathcal{V}^{(k)}$ are orthogonal ($U^T U = I$ implies $(\mathcal{U}^{(k)})^T \mathcal{U}^{(k)} = \mathcal{I}^{(k)}$), and the SVD of $\mathcal{A}^{(k)}$ is

$$\mathcal{A}^{(k)} = \mathcal{U}^{(k)} \mathcal{S}^{(k)} (\mathcal{V}^{(k)})^T.$$

Finally, the matrix $\mathcal{S}^{(k)}$ is diagonal, and its diagonal entries are all possible products of the numbers $\sigma_1, \sigma_2, \dots, \sigma_n$ taken k at a time. \square

THEOREM 7.2. *Let B_1, B_2, \dots, B_s be nonnegative nonsingular bidiagonal matrices, and let $A = B_1 B_2 \dots B_s$. Let x be an entry in some $B_r, 1 \leq r \leq s$, and \hat{A} be obtained from A by replacing x in B_r by $\hat{x} = x(1 + \delta)$, where $|\delta| \ll 1$. Let the eigenvalues and the singular values of A and \hat{A} be $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_n, \sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_n$ and $\hat{\lambda}_1 \geq \hat{\lambda}_2 \geq \dots \geq \hat{\lambda}_n, \hat{\sigma}_1 \geq \hat{\sigma}_2 \geq \dots \geq \hat{\sigma}_n$, respectively. Then for all $k = 1, 2, \dots, n$*

$$\left| \hat{\lambda}_k - \lambda_k \right| \leq \frac{2|\delta|}{1 - 2|\delta|} \lambda_k \quad \text{and} \quad \left| \hat{\sigma}_k - \sigma_k \right| \leq \frac{2|\delta|}{1 - 2|\delta|} \sigma_k.$$

Proof. The values of s and r are of no importance in this theorem; we introduce them only for clarity. Also, any B_j may be upper or lower bidiagonal, or simply diagonal.

The matrix A is TN since it is a product of TN bidiagonal matrices. We will first prove the theorem by assuming that A is TP and then extend the result to the TN case by continuity.

Since $A = B_1 B_2 \dots B_s$, the Cauchy–Binet identity [20, p. 9] implies that any entry $a_{ij}^{(k)}$ of $\mathcal{A}^{(k)}$ is a linear function of x : $a_{ij}^{(k)} = ax + b$ for some $a, b \geq 0$. Similarly, $\hat{a}_{ij}^{(k)} = ax(1 + \delta) + b$. Then for all i, j

$$(7.1) \quad \left| \hat{a}_{ij}^{(k)} - a_{ij}^{(k)} \right| = |\delta|ax \leq |\delta|a_{ij}^{(k)}.$$

The matrix $\mathcal{A}^{(k)}$ is positive with Perron root $\lambda_1 \lambda_2 \dots \lambda_k$ and two-norm $\sigma_1 \sigma_2 \dots \sigma_k$. Since small perturbations in the entries of a positive matrix cause small perturbations in its Perron root [15, Thm. 1] and two-norm [9, Cor. 5.1] the inequality (7.1) implies

$$(7.2) \quad \hat{\lambda}_1 \hat{\lambda}_2 \dots \hat{\lambda}_k = (1 + \delta_1) \lambda_1 \lambda_2 \dots \lambda_k, \quad \hat{\sigma}_1 \hat{\sigma}_2 \dots \hat{\sigma}_k = (1 + \delta_2) \sigma_1 \sigma_2 \dots \sigma_k,$$

and by analogy

(7.3)

$$\hat{\lambda}_1 \hat{\lambda}_2 \cdots \hat{\lambda}_{k-1} = (1 + \delta_3) \lambda_1 \lambda_2 \cdots \lambda_{k-1}, \quad \hat{\sigma}_1 \hat{\sigma}_2 \cdots \hat{\sigma}_{k-1} = (1 + \delta_4) \sigma_1 \sigma_2 \cdots \sigma_{k-1},$$

where $|\delta_i| \leq \delta$ for $i = 1, 2, 3, 4$. The claim now follows directly by substituting (7.3) into (7.2) and using that $|(1 + \delta_1)/(1 + \delta_3) - 1| \leq 2|\delta|/(1 - 2|\delta|)$.

Let A and \hat{A} now be TN. Let the TP matrices A_θ and \hat{A}_θ be obtained from A and \hat{A} by replacing the zero entries in $\mathcal{BD}(A)$ and $\mathcal{BD}(\hat{A})$, respectively, by a tiny quantity $\theta > 0$. Since the eigenvalues of a matrix are continuous functions of the matrix entries, by taking the limit as $\theta \rightarrow 0$ we conclude that the claim of this theorem is true when A is TN as well.

We considered the TP and TN cases separately because we needed $\mathcal{A}^{(k)}$ to be nonnegative and *irreducible* in order to invoke Theorem 1 from [15]. \square

If k entries in the factors of the product $A = B_1 \cdots B_s$ are perturbed and the relative perturbation in each entry is bounded by $0 < \delta \ll 1$, then we can apply Theorem 7.2 repeatedly and accumulate the relative perturbations to conclude that the relative perturbation in each eigenvalue and each singular value of the perturbed product is bounded by $2k\delta/(1 - 2k\delta)$.

COROLLARY 7.3. *Let A, \hat{A} be TN matrices with eigenvalues $\lambda_i, \hat{\lambda}_i$ and singular values $\sigma_i, \hat{\sigma}_i, i = 1, 2, \dots, n$, respectively. If $\mathcal{BD}(\hat{A})$ is a small componentwise relative perturbation of $\mathcal{BD}(A)$: $|(\mathcal{BD}(\hat{A}))_{ij} - (\mathcal{BD}(A))_{ij}| \leq \delta(\mathcal{BD}(A))_{ij}$, where $0 < \delta \ll 1$, then*

$$\left| \hat{\lambda}_i - \lambda_i \right| \leq \frac{2n^2\delta}{1 - 2n^2\delta} \lambda_i \quad \text{and} \quad \left| \hat{\sigma}_i - \sigma_i \right| \leq \frac{2n^2\delta}{1 - 2n^2\delta} \sigma_i.$$

In other words $\mathcal{BD}(A)$ determines the eigenvalues and the singular values of A accurately, and the appropriate structured condition number of each eigenvalue and/or singular value with respect to perturbations in $\mathcal{BD}(A)$ is at most $2n^2$.

We will now use Theorem 7.2 for the error analysis of our eigenvalue and singular value algorithms, Algorithms 5.1 and 6.1. These algorithms reduce the eigenvalue and the singular value problem of a TN matrix to the singular value problem of a bidiagonal matrix. In this reduction the only source of error is in computing the entries of a certain representation of an intermediate TN matrix A as a product of nonnegative bidiagonal matrices $A = B_1 \cdots B_s$ (e.g., any of the representations (4.2) to (4.7)). All arithmetic is subtraction-free; thus every single floating point operation will cause at most ϵ relative perturbation in at most one entry of some B_j . In turn, the relative perturbation of any eigenvalue and singular value of A will be at most $2\epsilon/(1 - 2\epsilon)$. Accumulating the error in each eigenvalue and singular value in the style of Higham as described above, we conclude that the relative perturbation in each eigenvalue and singular value of A after at most $\frac{16}{3}n^3 + O(n^2)$ such arithmetic operations (which is a bound on the cost of reduction to the bidiagonal singular value problem for both algorithms) will not exceed $(\frac{32}{3}n^3 + O(n^2))\epsilon/(1 - (\frac{32}{3}n^3 + O(n^2))\epsilon)$.

COROLLARY 7.4. *Let λ_i and σ_i be the eigenvalues and the singular values of a TN matrix A . Let τ_i and θ_i be the (exact) singular values of the bidiagonal matrices C and F as computed by Algorithms 5.1 and 6.1, respectively, in floating point arithmetic with machine precision ϵ . Then*

$$\left| \lambda_i - \tau_i^2 \right| \leq \frac{(\frac{32}{3}n^3 + O(n^2))\epsilon}{1 - (\frac{32}{3}n^3 + O(n^2))\epsilon} \lambda_i \quad \text{and} \quad \left| \sigma_i - \theta_i \right| \leq \frac{(\frac{32}{3}n^3 + O(n^2))\epsilon}{1 - (\frac{32}{3}n^3 + O(n^2))\epsilon} \sigma_i.$$

The final step of computing the singular values of the bidiagonal matrix using DLASQ1 is known to introduce only a small additional relative error [17, 36].

COROLLARY 7.5. *Under the assumption that the LAPACK routine DLASQ1 computes each singular value of a bidiagonal matrix with a relative error not exceeding $O(n^2)\epsilon$, the relative error in each eigenvalue or singular value of A , as computed by Algorithms 5.1 or 6.1, respectively, does not exceed $\frac{32}{3}n^3 + O(n^2)$ units in the last place.*

Singular vectors. The singular vectors of a TN matrix A are accurately determined by $\mathcal{BD}(A)$ and may be computed accurately by accumulating all Givens rotations in Algorithm 6.1.

PROPOSITION 7.6. *Let $A = U\Sigma V^T$ be the SVD of a TN matrix A and let $\hat{U}\hat{\Sigma}\hat{V}^T$ be the SVD of A as computed by Algorithm 6.1 with accumulation of the singular vector matrices. Then the acute angle $\theta(u_i, \hat{u}_i)$ between the right singular vectors and their computed counterparts is bounded by*

$$(7.4) \quad \sin \theta(u_i, \hat{u}_i) \leq \frac{O(n^3 \kappa \epsilon)}{\min_{j \neq i} |(\sigma_j - \sigma_i) / \sigma_i|},$$

where κ is a bound on the condition number of L and U in the LDU decomposition of A resulting from Gaussian elimination with complete pivoting. A similar result is true for the right singular vectors.

The modest constant hidden in the above big- O notation can be found in [11].

We conjecture that a slightly stronger result is true, namely we can omit the factor κ from the bound (7.4).

Proof. The result follows directly from [11] (see also the discussion in section 2.5). First consider the effect of a single δ relative perturbation in any one entry of $\mathcal{BD}(A)$. As in the proof of Theorem 7.2, from the Cauchy–Binet identity we know that the relative perturbation in any minor of A will be at most δ . The relative perturbation in any entry of L, D , or U will be at most 2δ , since every entry of L, D , and U is a quotient of minors of A . Following [11] we obtain the bound

$$(7.5) \quad \sin \theta(u_i, \hat{u}_i) \leq \frac{O(\kappa \delta)}{\min_{j \neq i} |(\sigma_j - \sigma_i) / \sigma_i|}.$$

Accumulating the perturbations in Algorithm 6.1 using (7.5) and using that for small acute angles $\sin x + \sin y \approx \sin(x + y)$, we obtain (7.4). \square

Eigenvectors. In the symmetric case, the eigenvector matrix of a TN matrix A may be accumulated in Algorithm 5.1 as a product of highly accurate EETs, a diagonal matrix, and the right singular vector matrix of a bidiagonal matrix. Alternatively the eigenvector matrix may be computed using Algorithm 6.1 as the right singular vector matrix of A . Either way, the computed eigenvectors will satisfy the error bound (7.4).

In the nonsymmetric case, however, we know of no relative perturbation results for eigenvectors of nonsymmetric TN matrices, and we make no claims as to the accuracy of the eigenvectors of a nonsymmetric TN matrix that may be computed using Algorithm 5.1.

A nonsymmetric TN matrix may not even have a complete set of eigenvectors. For example a Jordan block corresponding to a positive eigenvalue is TN. If a nonsymmetric TN matrix is irreducible, then it is diagonalizable, and a full set of eigenvectors exists. Although the factors of the eigenvector matrix—EETs, a diagonal matrix, and

the right singular vector of a bidiagonal matrix—are very accurate in the appropriate sense, forming the product will likely involve subtractive cancellation and loss of accuracy in the individual eigenvector components.

The eigenvector matrix of an irreducible TN matrix A is a γ -matrix (i.e., it has an LU decomposition where L and U^{-1} are TN [24]), and the j th eigenvector has $j - 1$ sign changes [21, Thm. 6, p. 87].

Finding algorithms that will guarantee accuracy in the computed eigenvectors and/or the two properties above is still an open problem and a topic of current research.

Vandermonde and Cauchy matrices. We conclude this section with a result about the accuracy and stability of the eigenvalues and singular values of TP Vandermonde and TP Cauchy matrices

$$V(x) = \left[x_i^{j-1} \right]_{i,j=1}^n \quad \text{and} \quad C(x, y) = \left[\frac{1}{x_i + y_j} \right]_{i,j=1}^n,$$

where $0 < x_1 < \dots < x_n$, and $x_1 < \dots < x_n$, $y_1 < \dots < y_n$, and $x_1 + y_1 > 0$, respectively. Define

$$\begin{aligned} \text{rel_gap}_x &\equiv \min_{i \neq j} \frac{|x_i - x_j|}{|x_i| + |x_j|}, \\ \text{rel_gap}_{xy} &\equiv \min_{i,j} \frac{|x_i - y_j|}{|x_i| + |y_j|}, \\ \kappa_V &\equiv \frac{1}{\text{rel_gap}_x}, \\ \kappa_C &\equiv \frac{1}{\text{rel_gap}_x} + \frac{1}{\text{rel_gap}_y} + \frac{1}{\text{rel_gap}_{xy}}. \end{aligned}$$

PROPOSITION 7.7. *Let $V = V(x)$ and $\hat{V} = V(\hat{x})$ be TP Vandermonde matrices, and let $C = C(x, y)$ and $\hat{C} = C(\hat{x}, \hat{y})$ be TP Cauchy matrices. Let $\lambda_i, \mu_i, \hat{\lambda}_i$, and $\hat{\mu}_i$ be the eigenvalues of V, C, \hat{V} , and \hat{C} , respectively. If $\hat{x}_i = x_i(1 + \delta_i)$ and $\hat{y}_i = y_i(1 + \eta_i)$, where $|\delta_i, \eta_i| \leq \delta \ll 1$, then*

$$(7.6) \quad |\hat{\lambda}_i - \lambda_i| \leq O(n^3 \kappa_V \delta) \lambda_i \quad \text{and} \quad |\hat{\mu}_i - \mu_i| \leq O(n^3 \kappa_C \delta) \mu_i.$$

The same condition numbers apply for the singular values of V and C . The quantities $n^3 \kappa_V$ and $n^3 \kappa_C$ are therefore the appropriate structured condition numbers of the eigenvalues and singular values of V and C with respect to perturbations in the data x_i and y_j .

Proof. It suffices to prove the result for the eigenvalues of C , the rest being analogous. We have

$$|(\hat{x}_i - \hat{x}_j) - (x_i - x_j)| = |\delta_i x_i - \delta_j x_j| \leq \delta |x_i - x_j| / \text{rel_gap}_x.$$

Similarly

$$|(\hat{x}_i - \hat{y}_j) - (x_i - y_j)| = |\delta_i x_i - \eta_j y_j| \leq \delta |x_i - y_j| / \text{rel_gap}_{xy}.$$

Accumulating the perturbations in $\mathcal{BD}(C)$ and $\mathcal{BD}(\hat{C})$ using the formulas (3.6) we obtain

$$|(\mathcal{BD}(\hat{C}))_{ij} - (\mathcal{BD}(C))_{ij}| \leq \frac{n \kappa_C \delta}{1 - n \kappa_C \delta} \mathcal{BD}(C)_{ij}.$$

Corollary 7.3 now implies (7.6). \square

PROPOSITION 7.8. *Let V and C be TP Vandermonde and TP Cauchy matrices with eigenvalues λ_i and μ_i , respectively. Let $\hat{\lambda}_i, \hat{\mu}_i$ be the eigenvalues of V and C computed in floating point arithmetic, by first computing $\mathcal{BD}(V)$ and $\mathcal{BD}(C)$ according to (3.5) and (3.6) and then running Algorithm 5.1. Then*

$$(7.7) \quad |\hat{\lambda}_i - \lambda_i| \leq \frac{(12n^3 + O(n^2))\epsilon}{1 - (12n^3 + O(n^2))\epsilon} \lambda_i \quad \text{and} \quad |\hat{\mu}_i - \mu_i| \leq \frac{(18n^3 + O(n^2))\epsilon}{1 - (18n^3 + O(n^2))\epsilon} \mu_i.$$

The same error bounds apply for the singular values computed by Algorithm 6.1.

Proof. The decompositions $\mathcal{BD}(V)$ and $\mathcal{BD}(C)$ can be computed using formulas (3.5) and (3.6) in $\frac{2}{3}n^3 + O(n^2)$ and $\frac{8}{3}n^3 + O(n^2)$ arithmetic operations, respectively. Every floating point operation in evaluating (3.5) or (3.6) can cause at most ϵ relative perturbation in an entry of $\mathcal{BD}(V)$ or $\mathcal{BD}(C)$, respectively. By applying Theorem 7.2 repeatedly, we conclude that the relative perturbations in the eigenvalues and/or singular values of V and C resulting from rounding errors in just computing $\mathcal{BD}(V)$ and $\mathcal{BD}(C)$ will not exceed

$$(7.8) \quad \frac{(\frac{4}{3}n^3 + O(n^2))\epsilon}{1 - (\frac{4}{3}n^3 + O(n^2))\epsilon} \quad \text{and} \quad \frac{(\frac{16}{3}n^3 + O(n^2))\epsilon}{1 - (\frac{16}{3}n^3 + O(n^2))\epsilon},$$

respectively. According to Corollary 7.5, Algorithms 5.1 and 6.1 will introduce at most

$$(7.9) \quad \frac{(\frac{32}{3}n^3 + O(n^2))\epsilon}{1 - (\frac{32}{3}n^3 + O(n^2))\epsilon}$$

additional relative error in each eigenvalue and singular value of V and C . Combining (7.8) and (7.9), we obtain (7.7). \square

The n^3 factor in the error bounds above is only the worst case error bound, which does not appear in numerical experiments we conducted.

8. Numerical results. We performed extensive numerical tests and confirmed the correctness and cost of our algorithms. To confirm the correctness we chose various TN structured matrices (Cauchy, Vandermonde, generalized Vandermonde, etc.) with random nodes and compared the output to the output of other $O(n^3)$ accurate algorithms (for the SVD [10, 11]) and to the output from Mathematica [40] in extended precision arithmetic. Most test matrices had condition numbers well in excess of 10^{16} , so conventional algorithms failed to get any correct digits in the smallest singular values and eigenvalues, whereas Algorithms 5.1 and 6.1 always agreed with the output from Mathematica to at least 14 digits in all singular values and eigenvalues, even the tiniest ones. We present a couple of our test results here.

The Hilbert matrix. In our first numerical example we computed the singular values of the 20×20 Hilbert matrix

$$H = \left[\frac{1}{i+j-1} \right]_{i,j=1}^{20},$$

which is also Cauchy and its bidiagonal decomposition is easily and accurately computed in $O(n^3)$ time using (3.6). The Hilbert matrix is an attractive example because it is a very ill-conditioned TN matrix for which other accurate $O(n^3)$ SVD algorithms exist [10, Alg. 2], which we can compare against.

We computed the singular values of H using the following algorithms:

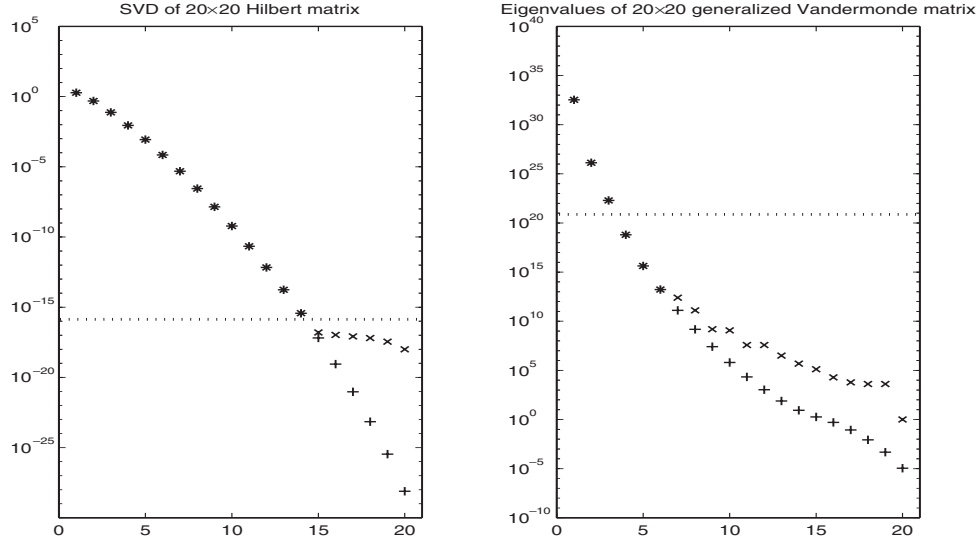


FIG. 8.1. Plots of the singular values of the 20×20 Hilbert matrix and the eigenvalues of a 20×20 generalized Vandermonde matrix; “+” = accurate, “x” = traditional. Data below the dotted lines may be inaccurate for the traditional algorithms.

- Mathematica [40] with 60-digit arithmetic;
- Algorithm 6.1;
- Algorithm 2 from [10];
- traditional SVD algorithm as implemented in MATLAB’s function `svd`.

Since $\kappa(H) \approx 10^{30}$, Mathematica with 60-digit arithmetic computed all singular values with about 30 significant digits of accuracy. Algorithm 2 from [10] guarantees high relative accuracy (close to 16 digits since we are using double precision) in the computed singular values, and its output agreed with the output from Algorithm 6.1 and the output from Mathematica to at least 14 decimal places. In contrast, the traditional algorithm—MATLAB’s function `svd`—returned only the largest singular values accurately with the accuracy gradually decreasing until the singular values smaller than $O(\epsilon)\sigma_1 \approx 10^{-16}$ were computed with no correct digits at all. All these results are in the left plot of Figure 8.1.

Generalized Vandermonde matrix. Next, consider the eigenvalue problem for a 20×20 generalized Vandermonde matrix

$$G = \left[x_i^{j-1+\lambda_{n-j+1}} \right]_{i,j=1}^{20}$$

with nodes $x_i = i$, $i = 1, 2, \dots, 20$, and corresponding to partition $\lambda = (6, 3, 2, 1)$. This Vandermonde matrix is generalized in the sense that the exponents increase nonlinearly—the i th row of G is $(1, x_i, x_i^2, \dots, x_i^{15}, x_i^{17}, x_i^{19}, x_i^{21}, x_i^{25})$. It is typical for the generalized Vandermonde matrices to be described in terms of corresponding partitions λ [13, 32].

We chose this example of a generalized Vandermonde matrix because it incorporates both our technique for computing an accurate bidiagonal decomposition of such a matrix [13] and the eigenvalue algorithm described in this paper. The bidiagonal decomposition of the *inverse* G^{-1} is computable accurately and efficiently using a

Björck–Pereyra method [13], and we permute the elements in the $L^{(i)}$ and $U^{(i)}$ and invert D in the bidiagonal decomposition of G^{-1} in order to obtain $\mathcal{BD}(G)$ [25]. As a benchmark we again used Mathematica. Since G is nonsymmetric, the accuracy of the computed eigenvalues is also affected by the angle between the left (y_i) and right (x_i) eigenvectors. The relative error in each eigenvalue λ_i is bounded by

$$\text{macheps} \cdot \frac{\|A\|_2}{\lambda_i |y_i^T x_i|},$$

where `macheps` is the machine precision in Mathematica. Running Mathematica's `Eigenvalues` routine in 60-digit arithmetic (`macheps` = 10^{-60}) ensured that each eigenvalue is computed with at least 16 correct leading digits (because $\lambda_{\min} \approx 10^{-5}$, $\min_i(\lambda_i |y_i^T x_i|) \approx 10^{-7}$, and $\|A\|_2 \approx 10^{32}$). The results from Algorithm 5.1 agreed with the results from Mathematica to at least 14 decimal digits, even in the tiniest eigenvalues. We also compared our results to the results of the traditional eigenvalue algorithm QR as implemented by MATLAB's routine `eig`. As expected only the largest eigenvalues are computed accurately. MATLAB also returned some negative eigenvalues and eigenvalues with nonzero imaginary part, so in Figure 8.1 we have plotted only their magnitudes.

9. Conclusions and open problems. We have presented new $O(n^3)$ algorithms for computing accurate eigenvalues and SVD of a TN matrix A , given its bidiagonal decomposition $\mathcal{BD}(A)$. With this we settle an open question from [11] not only for the singular value problem but for the eigenvalue problem as well. To the best of our knowledge Algorithm 5.1 is the first algorithm to compute all eigenvalues of a nonsymmetric matrix to high relative accuracy. The question of computing accurate eigenvectors of nonsymmetric TN matrices remains open.

We excluded the class of singular TN matrices because they no longer possess a unique bidiagonal decomposition; the requirement for nonsingularity in Theorem 2.1 is important. The problem of finding accurate eigenvalue and SVD algorithms for singular TN matrices also remains open.

Acknowledgments. I thank James Demmel for suggesting this problem to me and for many helpful discussions in the process of preparing this work; Froilán Dopico and Juan Manuel Peña for making useful suggestions, which lead to improvements in the presentation; and J. Martínez for pointing out the relevance of [33].

REFERENCES

- [1] E. ANDERSON, Z. BAI, C. BISCHOF, S. BLACKFORD, J. DEMMEL, J. DONGARRA, J. DU CROZ, A. GREENBAUM, S. HAMMARLING, A. MCKENNEY, AND D. SORENSEN, *LAPACK Users' Guide*, 3rd ed., Software Environ. Tools 9, SIAM, Philadelphia, 1999.
- [2] T. ANDO, *Totally positive matrices*, Linear Algebra Appl., 90 (1987), pp. 165–219.
- [3] A. BERENSTEIN, S. FOMIN, AND A. ZELEVINSKY, *Parametrizations of canonical bases and totally positive matrices*, Adv. Math., 122 (1996), pp. 49–149.
- [4] Å. BJÖRCK AND V. PEREYRA, *Solution of Vandermonde systems of equations*, Math. Comp., 24 (1970), pp. 893–903.
- [5] T. BOROS, T. KAILATH, AND V. OLSHEVSKY, *A fast parallel Björck-Pereyra-type algorithm for solving Cauchy linear equations*, Linear Algebra Appl., 302/303 (1999), pp. 265–293.
- [6] F. BRENTI, *Combinatorics and total positivity*, J. Combin. Theory Ser. A, 71 (1995), pp. 175–218.
- [7] F. BRENTI, *The applications of total positivity to combinatorics, and conversely*, in Total Positivity and Its Applications, Math. Appl. 359, Kluwer Academic Publishers, Dordrecht, The Netherlands, 1996, pp. 451–473.
- [8] Y. CHEN, *Singular values in electrical impedance tomography*, private communication, 2003.

- [9] J. DEMMEL, *Applied Numerical Linear Algebra*, SIAM, Philadelphia, 1997.
- [10] J. DEMMEL, *Accurate singular value decompositions of structured matrices*, SIAM J. Matrix Anal. Appl., 21 (1999), pp. 562–580.
- [11] J. DEMMEL, M. GU, S. EISENSTAT, I. SLAPNIČAR, K. VESELIĆ, AND Z. DRMAČ, *Computing the singular value decomposition with high relative accuracy*, Linear Algebra Appl., 299 (1999), pp. 21–80.
- [12] J. DEMMEL AND W. KAHAN, *Accurate singular values of bidiagonal matrices*, SIAM J. Sci. Statist. Comput., 11 (1990), pp. 873–912.
- [13] J. DEMMEL AND P. KOEV, *The accurate and efficient solution of a totally positive generalized Vandermonde linear system*, SIAM J. Matrix Anal. Appl., to appear.
- [14] I. S. DHILLON, *A New $O(n^2)$ Algorithm for the Symmetric Tridiagonal Eigenvalue/Eigenvector Problem*, Ph.D. thesis, University of California, Berkeley, CA, 1997.
- [15] L. ELSNER, I. KOLTRACHT, M. NEUMANN, AND D. XIAO, *On accurate computations of the Perron root*, SIAM J. Matrix Anal. Appl., 14 (1993), pp. 456–467.
- [16] S. M. FALLAT, *Bidiagonal factorizations of totally nonnegative matrices*, Amer. Math. Monthly, 108 (2001), pp. 697–712.
- [17] K. FERNANDO AND B. PARLETT, *Accurate singular values and differential qd algorithms*, Numer. Math., 67 (1994), pp. 191–229.
- [18] S. FOMIN AND A. ZELEVINSKY, *Double Bruhat cells and total positivity*, J. Amer. Math. Soc., 12 (1999), pp. 335–380.
- [19] S. FOMIN AND A. ZELEVINSKY, *Total positivity: Tests and parametrizations*, Math. Intelligencer, 22 (2000), pp. 23–33.
- [20] F. GANTMACHER, *The Theory of Matrices*, AMS Chelsea, Providence, RI, 1998.
- [21] F. GANTMACHER AND M. KREIN, *Oscillation Matrices and Kernels and Small Vibrations of Mechanical Systems*, revised edition, AMS Chelsea, Providence, RI, 2002.
- [22] M. GASCA AND C. A. MICCHELLI, EDS., *Total Positivity and Its Applications*, Math. Appl. 359, Kluwer Academic Publishers, Dordrecht, The Netherlands, 1996.
- [23] M. GASCA AND J. M. PEÑA, *Total positivity and Neville elimination*, Linear Algebra Appl., 165 (1992), pp. 25–44.
- [24] M. GASCA AND J. M. PEÑA, *Total positivity, QR factorization, and Neville elimination*, SIAM J. Matrix Anal. Appl., 14 (1993), pp. 1132–1140.
- [25] M. GASCA AND J. M. PEÑA, *On factorizations of totally positive matrices*, in Total Positivity and Its Applications, Kluwer Academic Publishers, Dordrecht, The Netherlands, 1996, pp. 109–130.
- [26] G. A. GEIST, G. W. HOWELL, AND D. S. WATKINS, *The BR eigenvalue algorithm*, SIAM J. Matrix Anal. Appl., 20 (1999), pp. 1083–1098.
- [27] N. J. HIGHAM, *Error analysis of the Björck-Pereyra algorithms for solving Vandermonde systems*, Numer. Math., 50 (1987), pp. 613–632.
- [28] N. J. HIGHAM, *Stability analysis of algorithms for solving confluent Vandermonde-like systems*, SIAM J. Matrix Anal. Appl., 11 (1990), pp. 23–41.
- [29] N. J. HIGHAM, *Accuracy and Stability of Numerical Algorithms*, 2nd ed., SIAM, Philadelphia, 2002.
- [30] W. KAHAN AND I. FARKAS, *Algorithms 167-169*, Comm. ACM, 6 (1963), pp. 164–165. See also the *Certification*, Comm. ACM, 6 (1963), p. 523.
- [31] S. KARLIN, *Total Positivity*, Vol. I, Stanford University Press, Stanford, CA, 1968.
- [32] I. G. MACDONALD, *Symmetric Functions and Hall Polynomials*, 2nd ed., Oxford University Press, New York, 1995.
- [33] J. J. MARTÍNEZ AND J. M. PEÑA, *Factorizations of Cauchy-Vandermonde matrices*, Linear Algebra Appl., 284 (1998), pp. 229–237.
- [34] J. J. MARTÍNEZ AND J. M. PEÑA, *Fast algorithms of Björck-Pereyra type for solving Cauchy-Vandermonde linear systems*, Appl. Numer. Math., 26 (1998), pp. 343–352.
- [35] THE MATHWORKS, INC., *MATLAB Reference Guide*, Natick, MA, 1992.
- [36] B. PARLETT, *The new qd algorithms*, in Acta Numerica, Cambridge University Press, Cambridge, UK, 1995, pp. 459–491.
- [37] J. W. RAINEY AND G. J. HABETLER, *Tridiagonalization of completely nonnegative matrices*, Math. Comp., 26 (1972), pp. 121–128.
- [38] A. M. WHITNEY, *A reduction theorem for totally positive matrices*, J. Analyse Math., 2 (1952), pp. 88–92.
- [39] J. H. WILKINSON, *The Algebraic Eigenvalue Problem*, Oxford University Press, Oxford, UK, 1965.
- [40] S. WOLFRAM, *Mathematica: A System for Doing Mathematics by Computer*, Addison-Wesley, Reading, MA, 1988.