SIAM J. Matrix Anal. Appl.
Vol. 28, No. 4, pp. 1126–1156

ACCURATE SYMMETRIC RANK REVEALING AND EIGENDECOMPOSITIONS OF SYMMETRIC STRUCTURED MATRICES*

FROILÁN M. DOPICO[†] AND PLAMEN KOEV[‡]

Abstract. We present new $O(n^3)$ algorithms that compute eigenvalues and eigenvectors to high relative accuracy in floating point arithmetic for the following types of matrices: symmetric Cauchy, symmetric diagonally scaled Cauchy, symmetric Vandermonde, and symmetric totally nonnegative matrices when they are given as products of nonnegative bidiagonal factors. The algorithms are divided into two stages: the first stage computes a symmetric rank revealing decomposition of the matrix to high relative accuracy, and the second stage applies previously existing algorithms to this decomposition to get the eigenvalues and eigenvectors. Rank revealing decompositions are also interesting in other problems, such as the numerical determination of the rank and the approximation of a matrix by a matrix with smaller rank.

Key words. eigenvalue, eigenvector, high relative accuracy, symmetric rank revealing factorization, Cauchy matrix, Vandermonde matrix, totally nonnegative matrix

AMS subject classifications. 65F15, 65F30

DOI. 10.1137/050633792

1. Introduction.

When traditional algorithms are used to compute the eigenvalues and eigenvectors of ill-conditioned *real symmetric matrices* in floating point arithmetic, only the eigenvalues with largest absolute values are computed with guaranteed relative accuracy. The tiny eigenvalues may be computed with no relative accuracy at all—and even with the wrong sign. The eigenvectors are computed with small error with respect to the absolute eigenvalue gap. This means that if ϵ is the machine precision, and v_i and $\hat{v}_i$ are, respectively, the exact and computed eigenvectors corresponding to an eigenvalue λ_i, then the acute angle between these vectors is bounded as $\theta(v_i, \hat{v}_i) \leq O(\epsilon)/\text{gap}_i$, where $\text{gap}_i = (\min_{j \neq i} |\lambda_i - \lambda_j|)/\max_k |\lambda_k|$. This implies that if there is more than one tiny eigenvalue, then the corresponding eigenvectors are computed with large errors, even if the tiny eigenvalues are well separated in the relative sense. See [1, section 4.7] for a survey on errors bounds for the symmetric eigenproblem.

Our goal is to derive algorithms for computing eigenvalues and eigenvectors of some structured $n \times n$ symmetric matrices *to high relative accuracy by respecting the symmetry of the problem*, and with cost $O(n^3)$, i.e., roughly the same cost as traditional algorithms for dense symmetric matrices. By *high relative accuracy* we mean that the eigenvalues λ_i, the eigenvectors v_i, and their computed counterparts

*Received by the editors June 16, 2005; accepted for publication (in revised form) by I. C. F. Ipsen April 19, 2006; published electronically December 18, 2006. This research was partially supported by the Ministerio de Educación y Ciencia of Spain through grant BFM-2003-00223, by the PRICIT Program of the Comunidad de Madrid through SIMUMAT Project grant S-0505/ESP/0158 (Froilán M. Dopico), and by NSF grant DMS-0314286 (Plamen Koev). This research was partially performed while P. Koev was visiting the Universidad Carlos III de Madrid in December 2004–January 2005 and was supported by the Ph.d. Program of Ingeniería Matemática.

http://www.siam.org/journals/simax/28-4/63379.html

[†] Departamento de Matemáticas, Universidad Carlos III de Madrid, Avda. Universidad 30, 28911 Leganés, Spain (dopico@math.uc3m.es).

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

$\hat{\lambda}_i$ and $\hat{v}_i$ will satisfy

$$(1) \qquad |\hat{\lambda}_i - \lambda_i| \leq O(\epsilon)|\lambda_i| \quad \text{and} \quad \theta(v_i, \hat{v}_i) \leq \frac{O(\epsilon)}{\min\limits_{j\neq i} \left|\frac{\lambda_i - \lambda_j}{\lambda_i}\right|} \quad \text{for} \quad i = 1, \dots, n.$$

These conditions guarantee that the new algorithms compute *all* eigenvalues, including the tiniest ones, with correct sign and leading digits. Moreover, the eigenvectors corresponding to relatively well separated tiny eigenvalues are accurately computed. In the case of a multiple eigenvalue, or a cluster of very close eigenvalues in the relative sense, the previous bound for $\theta(v_i, \hat{v}_i)$ becomes infinite or very large. In this case, we understand by high relative accuracy that the sines of the canonical angles between the unperturbed and the perturbed invariant subspaces corresponding to the cluster of eigenvalues are bounded by $O(\epsilon)$ over the relative gap between the eigenvalues inside the cluster and those outside the cluster [27]. This means that the new algorithms compute accurate bases of invariant subspaces corresponding to cluster of eigenvalues well separated in the relative sense from the rest of the eigenvalues.

In this work, we focus on the following classes of symmetric matrices: diagonally scaled Cauchy matrices (this class includes usual symmetric Cauchy matrices), Vandermonde matrices, and nonsingular totally nonnegative (TN) matrices. Symmetric diagonally scaled Cauchy matrices are defined through two ordered sets of real numbers, $\{x_1, x_2, \dots, x_n\}$ and $\{s_1, s_2, \dots, s_n\}$, and they are of the form

$$C = SC'S, \quad \text{where} \quad C'_{ij} = \frac{1}{x_i + x_j}, \quad 1 \leq i, j \leq n, \quad \text{and} \quad S = \operatorname{diag}(s_1, s_2, \dots, s_n);$$

i.e., they are the two-sided product of a usual symmetric Cauchy matrix C' times a diagonal matrix S. *It should be noticed that if S is the identity matrix, then $C = C'$, and C is just a usual symmetric Cauchy matrix.* Symmetric Vandermonde matrices depend only on one real parameter a, and they are defined as

$$A = \left[a^{(i-1)(j-1)}\right]_{i,j=1}^{n}.$$

This is the only type of Vandermonde matrices that is symmetric. As far as we know, this is the first time that the class of symmetric Vandermonde matrices has been studied in the literature. TN matrices are the matrices with all minors nonnegative. For symmetric diagonally scaled Cauchy matrices, we assume that the parameters $\{x_i\}_{i=1}^n$ and $\{s_i\}_{i=1}^n$ are given, i.e., we are not given just the entries of the matrices. This is a very natural assumption in situations where Cauchy matrices appear, such as, for instance, in rational interpolation theory. For symmetric Vandermonde matrices, we adopt the (also natural) assumption that the parameter a is given. In the case of TN matrices, we assume that the TN structure is explicitly revealed; i.e., any TN matrix is represented as a product of nonnegative bidiagonal matrices [18, 19]. This bidiagonal decomposition is particularly attractive because its nontrivial entries determine the eigenvalues of the matrix with high relative accuracy, and it can be computed very accurately for many important classes of TN matrices [26]. To finish this short presentation of the type of matrices we are dealing with, we want to stress that the symmetric diagonally scaled Cauchy and the symmetric Vandermonde matrices are, in general, indefinite matrices, while the symmetric nonsingular TN matrices are positive definite.

There exist $O(n^3)$ algorithms for computing eigendecompositions of symmetric diagonally scaled Cauchy and symmetric Vandermonde matrices with high relative accuracy, *but these algorithms do not respect the symmetry of the problem.* They are based on the idea of rank revealing decomposition (RRD): an RRD of $G \in \mathbb{R}^{m\times n}$, $m \geq n$, is a factorization $G = XDY^T$, where $D \in \mathbb{R}^{r\times r}$ is diagonal and nonsingular, and $X \in \mathbb{R}^{m\times r}$ and $Y \in \mathbb{R}^{n\times r}$ are well-conditioned matrices of full column rank (notice that this implies $r = \operatorname{rank}(G)$). Demmel et al. presented in [6] an algorithm for computing the singular value decomposition (SVD) of G with high relative accuracy when the computed factors $\widehat{X}, \widehat{D}$, and $\widehat{Y}$ of an RRD satisfy the following forward error bounds:

$$\begin{aligned} |D_{ii} - \widehat{D}_{ii}| &= O(\epsilon)|D_{ii}|, \\ \|X - \widehat{X}\|_2 &= O(\epsilon)\|X\|_2, \\ \|Y - \widehat{Y}\|_2 &= O(\epsilon)\|Y\|_2, \end{aligned} \tag{2}$$

where $\|\cdot\|_2$ is the spectral, or two-norm. Throughout this paper we will use the expression *accurate RRD* to mean an RRD that satisfies the error bounds (2). Algorithms for computing accurate RRDs of general diagonally scaled Cauchy and Vandermonde matrices were derived in [5], and therefore it is possible to compute the SVD of these matrices with high relative accuracy. Finally, an algorithm for computing a high relative accuracy eigendecomposition of a symmetric matrix, given an SVD computed with high relative accuracy, was developed in [11]. We note that when these algorithms are used, the symbols $O(\epsilon)$ appearing in (1) should be replaced with $O(\max\{\kappa_2(X), \kappa_2(Y)\}\,\epsilon)$, where $\kappa_2(X) \equiv \|X\|_2 \cdot \|X^{-1}\|_2$ is the spectral condition number of X.

The process outlined in the previous paragraph does not respect the symmetry of the problem in two stages. First, the RRDs of diagonally scaled Cauchy and Vandermonde matrices computed in [5] are not symmetric, i.e., $X \neq Y$, when G is symmetric. Second, even when G is symmetric and $X = Y$, the algorithm in [6] computes the SVD of G without respecting the symmetry of the problem. Respecting the symmetry is a very important property of eigenvalue algorithms (as well as other computations in the field of numerical linear algebra) because it often leads to increased speed, decreased storage requirements, and improved stability properties [3, 10, 21].

As two of our major contributions we present algorithms for computing accurate *symmetric* RRDs of symmetric diagonally scaled Cauchy matrices and symmetric Vandermonde matrices, i.e., decompositions $G = XDX^T$ with X well conditioned and D diagonal, which satisfy the bounds (2). In this context, it is important to stress that RRDs have been computed in practice as LDU factorizations provided by Gaussian elimination with complete pivoting (GECP) [6]. As can be seen in [21, section 4.4] and [22, Chapter 11], just preserving the symmetry of general dense symmetric indefinite matrices in a stable factorization of LU type requires much more complicated algorithms and pivoting strategies than the usual Gaussian elimination. In our algorithms, we need to preserve the symmetry and *also* attain the accuracy (2). This demands a careful exploitation of the structure of the problems that allows us to get important benefits from the point of view of operational cost. The algorithm we present for computing RRDs of symmetric diagonally scaled Cauchy matrices needs only half the operations required by the general nonsymmetric algorithm presented in [5]. In the case of symmetric Vandermonde matrices, the improvements are much more significant: the cost of the algorithm in [5] is $O(n^3)$ and requires complex arithmetic, and the cost of the algorithm we develop is $2n^2$ and requires only real arithmetic. We note, however, that for symmetric Vandermonde matrices our algorithm computes accurate RRDs only if $|a| \leq \frac{2}{3}$ or $|a| \geq \frac{3}{2}$. For the rest of the values of the parameter, i.e., $\frac{2}{3} < |a| < \frac{3}{2}$, our algorithm computes LDL^T factorizations with componentwise relative errors of $O(\epsilon)$, but they are not RRDs because L may be ill conditioned. This means that the factorizations $A = LDL^T$ we compute of symmetric Vandermonde matrices cannot be used to compute accurate eigendecompositions for values of $|a|$ close to one. However, they can be potentially useful in other contexts such as, for instance, in fast solvers of systems of linear equations $Ax = b$, where A is a symmetric Vandermonde matrix. The operational savings we have just described may not be of primary interest for computing accurate eigendecompositions, because in that case an $O(n^3)$ algorithm with high cost has to be applied to the RRD, but they are very important in other applications of RRDs.

Once an accurate symmetric RRD of a symmetric indefinite matrix G is computed, the J-orthogonal algorithm, introduced in [35] and carefully analyzed in [33], can be used to compute an eigendecomposition of G to high relative accuracy, preserving the symmetry of the process. Also, the signed SVD algorithm of [11] may be used, but then the symmetry is lost in this second stage. It should be noticed that the error bounds for the J-orthogonal algorithm [33] are not exactly of type (1) because the $O(\epsilon)$ symbols are rigorously $\kappa\epsilon$, where κ is the maximum of the condition numbers of some intermediate matrices appearing in the algorithm, which has not been bounded by any moderate magnitude. The error bounds for the signed SVD algorithm [11] are exactly of type (1) because the error for the eigenvectors depends on a different, smaller eigenvalue relative gap than the one in (1). However, in practice, both the J-orthogonal and signed SVD algorithms compute the eigenvalues and eigenvectors to high relative accuracy.

Our third major contribution is to develop algorithms for computing accurate RRDs of a nonsingular TN matrix whenever its bidiagonal factors are given. RRDs of general, not necessarily symmetric, TN matrices can be computed by combining algorithms in [26] and in [6], but the computation of *symmetric RRDs* requires a new approach. It should be remarked that algorithms for computing eigenvalues and singular values of general nonsingular TN matrices already have been presented in [26]. If the TN matrix is symmetric, the techniques in [26] allow us to modify these algorithms to compute eigenvalues to high relative accuracy *respecting the symmetry*. However, the algorithms in [26] do not use RRDs computed by a finite process.

Nonsingular symmetric TN matrices are positive definite; thus a symmetric RRD $A = XDX^T$ has positive elements on the diagonal matrix D. In this case we can compute an accurate eigendecomposition of A starting from this RRD, using a simpler and more efficient approach than the J-orthogonal or signed SVD algorithms. To do so, we compute the singular values and left singular vectors of $XD^{1/2}$ by using the one-sided Jacobi method with the rotations applied on the left [10, section 5.4.3] (see also the seminal reference [9]). This yields eigenvalues and eigenvectors with high relative accuracy as in (1), where the $O(\epsilon)$ symbols are replaced with $O(\epsilon\, \kappa_2(X))$. Obviously, this process preserves the symmetry.

In the previous paragraphs we have stressed the essential role of accurate RRDs in computing spectral problems to high relative accuracy. However, the computation of accurate RRDs is an interesting problem in its own right that can be used in other problems, such as the numerical determination of the rank, and the approximation of a matrix by a matrix with smaller rank [34, Chapter 5]. This is one of the reasons why reducing the cost in computing accurate RRDs is an important issue.

The three classes of symmetric matrices we consider—diagonally scaled Cauchy, Vandermonde, and TN—require three very different techniques for computing their accurate symmetric RRDs. In this regard, in [30] accurate symmetric RRDs of total signed compound and diagonally scaled totally unimodular matrices are computed by using an approach related to the one we used for diagonally scaled Cauchy matrices, i.e., combining accurate computation of Schur complements with the Bunch–Parlett pivoting strategy for the diagonal pivoting method [4]. Two other interesting classes of structured matrices for which there are algorithms for computing accurate RRDs are weakly diagonally dominant M-matrices [7, 31] and polynomial Vandermonde matrices [8]. For *symmetric* weakly diagonally dominant M-matrices, the general algorithm presented in [7] for nonsymmetric matrices respects the symmetry because it performs only diagonal pivoting. The algorithm in [8] does not preserve the symmetry for symmetric matrices, but the symmetric polynomial Vandermonde matrices are nonsymmetric, except in very special cases.

The paper is organized as follows. In section 2 we study how the eigenvalues and eigenvectors of a symmetric matrix are changed by errors of type (2) in a symmetric RRD. In section 3 we present the algorithm, and its error analysis, for computing symmetric RRDs of symmetric diagonally scaled Cauchy matrices. The same is done in section 4 for symmetric Vandermonde matrices. Section 5 includes the algorithms for computing accurate RRDs (symmetric and nonsymmetric) of nonsingular TN matrices. We present numerical experiments in section 6. Finally, in the appendix the technical proof of Theorem 3.1 for the rounding error analysis of diagonally scaled Cauchy matrices is carefully developed in a more general setting.

2. Perturbation properties of symmetric RRDs.

Let G be an $m \times n$ matrix, and let $G = XDY^T$ be an RRD of G. It was shown in [6, Theorem 2.1] that the RRD of G determines its SVD to high relative accuracy; i.e., small relative normwise perturbations of X and Y, and small relative componentwise perturbations of D, produce small relative changes in all singular values of G, and produce small changes in the singular vectors with respect to the singular value relative gap. Next we prove that a symmetric RRD of a symmetric matrix determines its eigenvalues and eigenvectors to high relative accuracy.

THEOREM 2.1. *Let $A = XDX^T$ and $\widetilde{A} = \widetilde{X}\widetilde{D}\widetilde{X}^T$ be RRDs of the real symmetric $n \times n$ matrices A and $\widetilde{A}$. Let $\lambda_1 \geq \cdots \geq \lambda_n$ be the eigenvalues of A and $\tilde{\lambda}_1 \geq \cdots \geq \tilde{\lambda}_n$ be the eigenvalues of $\widetilde{A}$. Let $q_1, \ldots, q_n$ and $\tilde{q}_1, \ldots, \tilde{q}_n$ be the corresponding orthonormal eigenvectors. Let us assume that*

$$\frac{\|\widetilde{X} - X\|_2}{\|X\|_2} \leq \beta,$$

$$\frac{|\widetilde{D}_{ii} - D_{ii}|}{|D_{ii}|} \leq \beta \quad \text{for all } i,$$

where $0 \leq \beta < 1$. Let $\eta = \beta\,(2 + \beta)\,\kappa_2(X)$ be smaller than 1; then

$$|\lambda_i - \tilde{\lambda}_i| \;\leq\; (2\eta + \eta^2)\,|\lambda_i|, \qquad 1 \leq i \leq n,$$

and

$$\sin\theta(q_i, \tilde{q}_i) \leq \frac{\eta}{1-\eta}\left(1 + \frac{2+\eta}{\min_{j \neq i} \frac{|\tilde{\lambda}_i - \lambda_j|}{|\lambda_j|}}\right), \qquad 1 \leq i \leq n,$$

where $\theta(q_i, \tilde{q}_i)$ is the acute angle between q_i and $\tilde{q}_i$. In the case of multiple eigenvalues, or clusters of very close eigenvalues in the relative sense, a similar bound holds for the sines of the canonical angles of the corresponding invariant subspaces.

Proof. The proof is similar to that of Theorem 2.1 in [6]. The main idea is to express $\widetilde{A}$ as a *symmetric* multiplicative perturbation of A, i.e., $\widetilde{A} = (I+E)A(I+E)^T$. This is combined with [12, Theorem 2.1] and [27, Theorem 3.1]. □

A more general version of Theorem 2.1, including similar perturbation results for invariant subspaces [27], can be developed. These bounds are useful when several eigenvalues form a tight cluster, well separated from the remaining eigenvalues, because in this case the invariant subspace is well conditioned, while the individual eigenvectors are very ill conditioned. It is also possible to present perturbation results for eigenvectors with the relative gap defined using exclusively eigenvalues of A, at the cost of bounding the sine of the double angle, i.e., $\sin 2\theta(q_i, \tilde{q}_i)$ [10, Theorem 5.7], [28, Theorem 2.2].

3. Symmetric diagonally scaled Cauchy matrices.

For a real symmetric matrix A, the LU factorization computed using Gaussian elimination, with partial or complete pivoting, does not always preserve the symmetry of the problem. Symmetric pivoting strategies, i.e., permuting rows and columns in the same way, may be unstable or may not exist. A trivial instance is when all the entries on the main diagonal are zero. The most widely used factorization [1, 21, 22] for symmetric matrices is the following special block LU factorization:

$$PAP^T = L\,D_b\,L^T,$$

where P is a permutation matrix, L is unit lower triangular, and D_b is block diagonal with diagonal blocks of dimension 1 or 2. The 2×2 diagonal blocks are symmetric indefinite matrices, and the corresponding diagonal blocks of L are the 2×2 identity matrix. This method is sometimes called the *diagonal pivoting method* [22] and can be implemented with partial, complete, or rook pivoting. We are interested in computing a symmetric RRD; therefore we will focus on the *Bunch–Parlett complete pivoting strategy* [4], which in practice[1] produces a well-conditioned matrix L. Notice that $L\,D_b\,L^T$ is not an RRD because D_b is not diagonal. To get an RRD, we will perform a spectral factorization of each of the 2×2 blocks of D_b; thus $D_b = VDV^T$ with D diagonal and V orthogonal and block diagonal as D_b. Finally,

$$PAP^T = L\,D_b\,L^T = (LV)D(LV)^T \equiv XDX^T \tag{3}$$

is a symmetric RRD. This procedure has been essentially introduced in [32] to compute a symmetric indefinite decomposition GJG^T, where $J = \operatorname{diag}(\pm 1)$. Notice that a GJG^T factorization can be easily computed from XDX^T as $(X\sqrt{|D|})\,J\,(\sqrt{|D|}X^T)$. Moreover, if XDX^T is accurately computed, then GJG^T is also accurately computed, and vice versa. In the rest of the paper we will focus on RRDs XDX^T from the point of view of both algorithms and error analysis.

To be more specific, the method can be described as follows. Let Π be a permutation matrix such that

$$\Pi A \Pi^T = \begin{bmatrix} E & C^T \\ C & B \end{bmatrix}, \tag{4}$$

[1] It can be proven that $\kappa_\infty(L) < n\,(3.78)^n$, by using Theorem 8.12 and Problem 8.5 in [22]. This bound is similar to that appearing in GECP. Therefore, there exists a remote possibility of the Bunch–Parlett pivoting strategy failing to compute a well-conditioned factor L.

where E is a 1×1 or a 2×2 nonsingular matrix. The pivot E and the permutation Π are chosen by comparing the numbers $\mu_0 = \max_{i,j} |a_{ij}| \equiv |a_{rs}|$ $(r \geq s)$ and $\mu_1 = \max_i |a_{ii}| \equiv |a_{pp}|$. If $\mu_1 \geq \alpha\mu_0$, where α is a parameter $(0 < \alpha < 1)$, then $E = a_{pp}$, and if $\mu_1 < \alpha\mu_0$, then E has dimension 2 and $E_{21} = |a_{rs}|$. The classical value for the parameter is $\alpha = (1+\sqrt{17})/8$ (≈ 0.64). Then we can factorize

$$\Pi A \Pi^T = \begin{bmatrix} I & 0 \\ CE^{-1} & I \end{bmatrix} \begin{bmatrix} E & 0 \\ 0 & B - CE^{-1}C^T \end{bmatrix} \begin{bmatrix} I & E^{-1}C^T \\ 0 & I \end{bmatrix}. \tag{5}$$

If E is a 2×2 matrix, let $E = U\Lambda U^T$ be its orthogonal spectral factorization computed by the Jacobi procedure [21, section 8.4]. Then

$$\Pi A \Pi^T = \begin{bmatrix} U & 0 \\ CU\Lambda^{-1} & I \end{bmatrix} \begin{bmatrix} \Lambda & 0 \\ 0 & B - CE^{-1}C^T \end{bmatrix} \begin{bmatrix} U^T & \Lambda^{-1}U^TC^T \\ 0 & I \end{bmatrix}. \tag{6}$$

The process is recursively repeated on the Schur complement $B - CE^{-1}C^T$.

In the case of diagonally scaled Cauchy matrices, it was shown in [5] how to compute all the Schur complements with an entrywise small relative error. Therefore, to compute an accurate symmetric RRD, the remaining task is to show that in (6) the orthogonal diagonalization $E = U\Lambda U^T$ of the 2×2 pivot and the matrix $CU\Lambda^{-1}$ can be accurately computed for each Schur complement.

Let us summarize some key results in [5]. The entries of an $n \times n$ symmetric diagonally scaled Cauchy matrix C are $C_{ij} = s_i s_j/(x_i + x_j)$, where the s_i and x_i, $1 \leq i \leq n$, are given real floating point numbers. Let $S^{(m)}$ be the mth Schur complement of C $(S^{(0)} \equiv C)$. We enumerate the elements of $S^{(m)}$ as the corresponding elements of C. The recurrence relation,

$$S^{(m)}_{rs} = S^{(m-1)}_{rs} \frac{(x_r - x_m)(x_s - x_m)}{(x_m + x_s)(x_r + x_m)} \qquad \text{for} \qquad m+1 \leq r, s \leq n, \tag{7}$$

allows us to compute accurately each Schur complement from the previous one. This is what we need when the Bunch–Parlett pivoting strategy selects a 1×1 pivot. If a 2×2 pivot is selected, we apply (7) twice to obtain

$$S^{(m+1)}_{rs} = S^{(m-1)}_{rs} \frac{(x_r - x_m)(x_s - x_m)}{(x_m + x_s)(x_r + x_m)} \cdot \frac{(x_r - x_{m+1})(x_s - x_{m+1})}{(x_{m+1} + x_s)(x_r + x_{m+1})}. \tag{8}$$

Combining (7) and (8) with (6), we get the following algorithm to compute a symmetric RRD of a symmetric diagonally scaled Cauchy matrix.[2]

Algorithm 1. Symmetric RRD of a symmetric diagonally scaled Cauchy matrix.

```
Input:  $S = \{s_1, \dots, s_n\}$;  $x = \{x_1, \dots, x_n\}$
Output:
    $D$ is a rank $\times$ rank diagonal matrix, where rank is the rank of
    the diagonally scaled Cauchy matrix defined by $S$ and $x$.
    $X$ is an $n \times$ rank block lower triangular matrix, with diagonal
    blocks of dimension $1 \times 1$ or $2 \times 2$.
    IPIV is an $n$-dimensional vector containing a permutation of
    $\{1, \dots, n\}$ such that, if $Q = I_n$ and $P = Q(\text{IPIV}, :)$, then
```

$$P \left[\frac{s_i s_j}{x_i + x_j} \right]_{i,j=1}^{n} P^T = XDX^T.$$

[2] We will use MATLAB [29] notation for submatrices, e.g., $A(i:j, k:l)$ will indicate the submatrix of A consisting of rows i through j and columns k through l, and $A(:, k:l)$ will indicate the submatrix of A consisting of columns k through l.

```
% Initializing variables
  $\alpha = (1+\sqrt{17})/8 \approx 0.64$
  rank $= n$
  IPIV = $1 : n$
  $D$ = zeros($n$)
  for $p = 1 : n$ and $q = 1 : p$
      $A(p,q) = s_p s_q/(x_p + x_q)$
      $A(q,p) = A(p,q)$
  endfor
% Main loop
  $k = 1$
  while $k \leq n$
      $\mu_0$ = maximum entry of $|A(k:n, k:n)| \equiv |A(r,s)|$ $(r \geq s)$
      $\mu_1$ = maximum entry of $\mathrm{diag}(|A(k:n,k:n)|) \equiv |A(p,p)|$
      if $\mu_1 \geq \alpha\mu_0$
          if $\mu_1 = 0$
              rank $= k - 1$
              $k = n + 1$
          else
              swap entries $k \leftrightarrow p$ in IPIV
              swap entries $k \leftrightarrow p$ in x
              swap rows $k \leftrightarrow p$ and swap columns $k \leftrightarrow p$ in $A$
              for $r = k+1 : n$ and $s = k+1 : r$
                  $A(r,s) = A(r,s)\dfrac{(x_r - x_k)(x_s - x_k)}{(x_k + x_s)(x_r + x_k)}$
                  $A(s,r) = A(r,s)$
              endfor
              $D(k,k) = A(k,k)$
              $A(k:n,k) = A(k:n,k)/A(k,k)$
              $A(k,k+1:n) =$ zeros($1, n-k$)
              $k = k + 1$
          endif
      else
          swap entries $k \leftrightarrow s$ and swap entries $k+1 \leftrightarrow r$ in IPIV
          swap entries $k \leftrightarrow s$ and swap entries $k+1 \leftrightarrow r$ in x
          swap rows $k \leftrightarrow s$ and swap rows $k+1 \leftrightarrow r$ in A
          swap columns $k \leftrightarrow s$ and swap columns $k+1 \leftrightarrow r$ in A
          for $r = k+2 : n$ and $s = k+2 : r$
              $A(r,s) = A(r,s)\dfrac{(x_r - x_k)(x_s - x_k)(x_r - x_{k+1})(x_s - x_{k+1})}{(x_k + x_s)(x_r + x_k)(x_{k+1} + x_s)(x_r + x_{k+1})}$
              $A(s,r) = A(r,s)$
          endfor
% Orthogonal diagonalization of the $2 \times 2$ pivot $A(k:k+1, k:k+1)$
          $z = (A(k+1,k+1) - A(k,k))/A(k+1,k)/2$
          if $z = 0$
              $t = 1$
          else
              $t = \mathrm{sign}(z)/\left(\mathrm{abs}(z) + \sqrt{1+z^2}\right)$
          endif
```

```
                $cs = 1/\sqrt{1+t^2}$
                $sn = t \cdot cs$
                $U = \left[\begin{smallmatrix} cs & sn \\ -sn & cs \end{smallmatrix}\right]$
                $D(k,k) = A(k,k) - t \cdot A(k+1,k)$
                $D(k+1,k+1) = A(k+1,k+1) + t \cdot A(k+1,k)$
                $A(k:k+1,k:k+1) = U$
                $A(k+2:n,k:k+1) = A(k+2:n,k:k+1) \cdot U$
                                  $\cdot \operatorname{diag}[\frac{1}{D(k,k)}, \frac{1}{D(k+1,k+1)}]$
                $A(k:k+1,k+2:n) = \texttt{zeros}(2, n-k-1)$
                $k = k+2$
            endif
        endwhile
        $X = A(:,1:\texttt{rank})$
        $D = D(1:\texttt{rank},1:\texttt{rank})$
        $Q = \texttt{eye}(n)$
        $P = Q(IPIV,:)$
```

The cost of Algorithm 1 is $4\,n^3/3 + O(n^2)$ flops, or $2\,n^3/3 + O(n^2)$ if all n^2 possible values of $(x_r - x_m)$ and $1/(x_r + x_m)$ are precomputed. Next, we show that the computed symmetric RRD is accurate.

THEOREM 3.1. *Let*

$$C = \left[\frac{s_i s_j}{x_i + x_j}\right]_{i,j=1}^{n}$$

be a real symmetric diagonally scaled Cauchy matrix, where $s_1, \ldots, s_n$ *and* $x_1, \ldots, x_n$ *are floating point numbers. Let* P, $\widehat{X}$, *and* $\widehat{D}$ *be the matrices of the factorization* (3) *computed by Algorithm* 1 *applied to* C *in floating point arithmetic with machine precision* ϵ. *Let us apply Algorithm* 1 *in exact arithmetic to* C, *but choosing the same dimensions and positions for the pivots as those selected in floating point arithmetic. Let* X *and* D *be the exact factors; thus* $PCP^T = XDX^T$. *If*

$$\frac{648\,(n+2)\,\epsilon}{1 - 648\,(n+2)\,\epsilon} < 1,$$

then

1.
$$|\widehat{D}_{ii} - D_{ii}| \leq \frac{146\,(n+4)\,\epsilon}{1 - 146\,(n+4)\,\epsilon} |D(i,i)| \qquad \textit{for all} \qquad i = 1, \ldots, n.$$

2.
$$||\widehat{X} - X||_F \leq 13\,\frac{684\,(n+2)\,\epsilon}{1 - 684\,(n+2)\,\epsilon}\,||X||_F.$$

If, moreover,

$$\frac{12481\,n\,\epsilon}{1 - 12481\,n\,\epsilon} < \frac{1}{2},$$

then

3.

$$||\widehat{X}(:,j) - X(:,j)||_2 \leq 144\,\sqrt{n}\,\frac{684\,(n+2)\,\epsilon}{1-684\,(n+2)\,\epsilon}\,||X(:,j)||_2 \quad \textit{for all} \quad j=1,\ldots,n.$$

According to Theorem 2.1 and (2), the third item in Theorem 3.1 is not necessary for computing accurate eigenvalues and eigenvectors. It is included for the sake of completeness and because it allows us to state error bounds for the column scaling of X with minimum condition number. We remark that the numerical constants appearing in the bounds above are not optimal: we have sometimes overestimated the constants to get simpler bounds. However, the order of magnitude is correct up to a factor smaller than 10. Theorem 3.1 remains valid if the rank, say ρ, of the matrix is less that n. In this case, the last $n-\rho$ diagonal elements of D are exactly computed to be zero, and the corresponding columns of X are just the $n-\rho$ columns of the identity matrix, and they are also exactly computed.

The proof of Theorem 3.1 is technical and is presented in the appendix. However, the argument explaining why Algorithm 1 accurately computes a symmetric rank revealing factorization of the diagonally scaled Cauchy matrix C can be easily understood. In the first place, the recurrence relation (7) allows us to compute the entries of the Schur complements with a relative error bounded by $8n\epsilon/(1-8n\epsilon)$. Therefore, the elements of D and the entries of the columns of X corresponding to 1×1 pivots are also computed with small relative errors. For the quantities corresponding to 2×2 pivots, the error analysis heavily depends on the properties of these pivots. As we will prove in the appendix, the 2×2 pivots selected by the Bunch–Parlett complete pivoting strategy are very well-conditioned indefinite matrices (with a spectral condition number less than 4.6 for the value $\alpha = 0.64$ used in Algorithm 1), and the entries of their unitary eigenvectors are greater than 0.47 (again for $\alpha = 0.64$). Therefore, the Jacobi algorithm computes with small relative error the eigenvalues (i.e., the elements of D) and the entries of the eigenvectors of the 2×2 pivots. According to (6), the upper 2×2 block of the corresponding two columns of X is just the eigenvector matrix U, and therefore its entries are accurately computed. The rest of the elements of these two columns of X are obtained through multiplying by U and by Λ^{-1}, but these two matrices are well conditioned and all their entries have been computed with small relative error. This last step does not guarantee small entrywise relative errors but it does guarantee small normwise relative errors for X.

4. Symmetric Vandermonde matrices.

A symmetric Vandermonde matrix is defined as

$$A = \left[a^{(i-1)(j-1)}\right]_{i,j=1}^{n} = \begin{bmatrix} 1 & 1 & 1 & \ldots & 1 \\ 1 & a & a^2 & \ldots & a^{n-1} \\ 1 & a^2 & a^4 & \ldots & a^{2(n-1)} \\ \vdots & \vdots & \vdots & \ldots & \vdots \\ 1 & a^{n-1} & a^{2(n-1)} & \ldots & a^{(n-1)^2} \end{bmatrix}, \tag{9}$$

where a is a real number. The class of symmetric Vandermonde matrices depends only on one parameter, and it is the only class of matrices which are, simultaneously, symmetric and of Vandermonde type. Symmetric Vandermonde matrices with $n > 2$ are singular when $a = 0$, $a = 1$, and $a = -1$. In these cases they have only, 2, 1, and 2, respectively, nonzero eigenvalues that can be accurately computed by any standard symmetric eigenvalue algorithm because they are of similar magnitudes. In fact, when

$a = 1$, the only nonzero eigenvalue is equal to n. We assume that a is different from $0, 1$ and -1. The matrix A is positive definite if $a > 1$ and, in this case, A is also totally positive. Therefore, when $a > 1$, an accurate bidiagonal factorization of A can be computed [26, section 3], and its eigenvalues can be obtained to high relative accuracy with the method presented in [26]. The algorithm we introduce in section 5 for computing accurate symmetric RRDs of symmetric totally positive matrices can also be applied to symmetric Vandermonde matrices with $a > 1$.

In this section, we present a method for computing an accurate RRD of A, in the sense of (2), by respecting the symmetry of A. This allows us to compute eigenvalues and eigenvectors to high relative accuracy, as explained in the introduction.

The method we present to compute an accurate RRD of A is very different from the one we used for diagonally scaled Cauchy matrices. The Schur complement of a Vandermonde matrix does not inherit the Vandermonde structure. Moreover, row and column permutations coming from any pivoting strategy also destroy the symmetric Vandermonde structure. Our approach avoids the computation of the Schur complements and, also, avoids pivoting. To be more precise, in the case $|a| < 1$, we use exact formulas for the elements of the LDL^T factorization of A, where L is unit lower triangular and D is diagonal, and we prove that the condition number of L in the 1-norm is $O(n^2)$ when $|a| \leq \frac{2}{3}$. In the case $|a| > 1$, we use exact formulas for the elements of the $\bar{L}\bar{D}\bar{L}^T$ factorization of the *converse* of A, i.e., $A^{\#} \equiv \left[A_{n-i+1,n-j+1}\right]_{i,j=1}^{n}$, and we will prove that $\kappa_1(\bar{L}) = O(n^2)$ when $|a| \geq \frac{3}{2}$. Note that in both cases $|a| \leq \frac{2}{3}$ and $|a| \geq \frac{3}{2}$, we are dealing with matrices whose elements vary widely and in which the largest elements are in the first positions. This is the reason why we are able to get RRDs without using pivoting strategies. The formulas we use allow us to compute accurate LDL^T factorizations for any value of a, but only when $|a| \leq \frac{2}{3}$ or $|a| \geq \frac{3}{2}$ can we guarantee that they are RRDs. These limits are somewhat arbitrary since we can consider values of a closer to $|a| = 1$ at the cost of increasing the bound for $\kappa_1(L)$. However, it should be stressed that we cannot consider values of a as close as we want to $|a| = 1$ because $\kappa_1(L)$ approaches 2^n as $|a|$ approaches 1.

In plain words, there are three limits for which the matrix A is extremely ill conditioned and that have eigenvalues that can vary widely: $|a|$ small enough, $|a|$ large enough, and $|a|$ close enough to 1. We are able to compute eigenvalues and eigenvectors of A to high relative accuracy *by respecting the symmetry* only in the first two cases, i.e., when A contains elements with very different magnitudes. Eigenvalues and eigenvectors *for any value of a* can be computed to high relative accuracy by combining the algorithm presented in [5] to compute a nonsymmetric RRD of A with the signed SVD (SSVD) algorithm in [11], *at the cost of not respecting the symmetry* of the problem.

Consider first the case $|a| < 1$. We start with the LDU decomposition $A = LDL^T$. The entries of L and D are quotient of minors of A [15, section 1.II]:

$$d_i = \frac{\det A(1:i, 1:i)}{\det A(1:i-1, 1:i-1)} = a^{\frac{1}{2}(i-2)(i-1)} \cdot \prod_{t=1}^{i-1}(a^t - 1), \tag{10}$$

$$l_{ij} = \frac{\det A([1:j-1, i], 1:j)}{\det A(1:j, 1:j)} = \prod_{t=1}^{j-1} \frac{1 - a^{i-j+t}}{1 - a^t}. \tag{11}$$

Next, we prove that when $|a| \leq \frac{2}{3}$, the entries of L and L^{-1} are bounded by e^6; thus L is well conditioned.

LEMMA 4.1. *If* $0 \le x \le \frac{2}{3}$ *and* $j \ge 1$, *then*

$$\prod_{t=1}^{j-1} \frac{1}{1-x^t} \le e^6.$$

Proof. We start by observing that $\log(1-x^t) \ge -3x^t$ for $t \ge 1$: If $f(z) = \log(1-z) + 3z$, then $f'(z) = \frac{1}{z-1} + 3 = \frac{3z-2}{z-1} \ge 0$, meaning $f(z)$ is increasing on $[0, \frac{2}{3}]$ and $f(z) \ge f(0) = 0$ on the same interval. Therefore,

$$\log\left(\prod_{t=1}^{\infty}(1-x^t)\right) \ge -3\sum_{t=1}^{\infty} x^t = \frac{-3x}{1-x} \ge -6$$

and

$$\prod_{t=1}^{j-1} \frac{1}{1-x^t} \le \prod_{t=1}^{\infty} \frac{1}{1-x^t} \le e^6. \qquad \square$$

Next, we bound the entries l_{ij} of L: If $0 < a \le \frac{2}{3}$, or $-\frac{2}{3} \le a < 0$ and $i-j$ is even, we have

$$\frac{1-a^{i-j+t}}{1-a^t} \le \frac{1}{1-|a|^t},$$

and using (11) and Lemma 4.1 we get

$$l_{ij} = \prod_{t=1}^{j-1} \frac{1-a^{i-j+t}}{1-a^t} \le \prod_{t=1}^{j-1} \frac{1}{1-|a|^t} \le e^6.$$

Otherwise, if $-\frac{2}{3} \le a < 0$ and $i-j$ is odd, we again have

$$l_{ij} = \prod_{t=1}^{j-1} \frac{1-a^{i-j+t}}{1-a^t} = \frac{1-a^{i-1}}{1-a^{i-j}} \cdot \prod_{t=1}^{j-1} \frac{1-a^{i-j-1+t}}{1-a^t} \le \frac{1+|a|^{i-1}}{1+|a|^{i-j}} \cdot e^6 \le e^6.$$

Either way, $l_{ij} \le e^6$ and $\|L\|_1 \le e^6 n$.

The entries of L^{-1} are also quotients of minors of A, as we now describe. From the LDU decomposition $A = LDU$ we get $A^{-T\#} = L^{-T\#}D^{-T\#}U^{-T\#}$. Therefore, by formula (1.31) in [2],

$$\begin{aligned}
\left(L^{-1}\right)_{ij} &= \left(L^{-T\#}\right)_{n-j+1,n-i+1} \\
&= \frac{\det A^{-T\#}([1:n-i, n-j+1], 1:n-i+1)}{\det A^{-T\#}(1:n-i+1, 1:n-i+1)} \\
&= \frac{\det A^{-1}(i:n, [j, i+1:n])}{\det A^{-1}(i:n, i:n)} \\
&= (-1)^{i+j} \cdot \frac{\det A([1:j-1, j+1:i], 1:i-1)}{\det A(1:i-1, 1:i-1)} \\
&= (-1)^{i+j} \cdot a^{\frac{1}{2}(i-j-1)(i-j)} \cdot \prod_{t=1}^{j-1} \frac{1-a^{i-j+t}}{1-a^t}. \qquad (12)
\end{aligned}$$

Similarly, $\left|(L^{-1})_{ij}\right| \leq e^6$ when $|a| \leq \frac{2}{3}$, and

$$\kappa_1(L) = \|L\|_1 \cdot \|L^{-1}\|_1 \leq e^{12} n^2;$$

i.e., L is well conditioned when $|a| \leq \frac{2}{3}$. The constant e^{12} and the factor n^2 in the previous bound are pessimistic, and the true values of $\kappa_1(L)$ are much smaller. They are shown in the following table for some values of a in 30×30 Vandermonde matrices:

a	$-2/3$	-0.5	-0.3	-0.05	0.05	0.3	0.5	$2/3$
$\kappa_1(L)$	92.12	79.25	69.83	61.50	64.16	126.98	379.12	2694.99

When $|a| > 1$ we consider the *converse* of A:

$$A^{\#} \equiv \left[A_{n-i+1,n-j+1}\right]_{i,j=1}^n = \left[a^{(n-i)(n-j)}\right]_{i,j=1}^n.$$

The matrices A and $A^{\#}$ are similar via an orthogonal similarity transformation,

$$A = JA^{\#}J,$$

where the matrix $J = [\delta_{n-i+1,j}]_{i,j=1}^n$ is the *reverse identity* (which is orthogonal and involutary: $J = J^T = J^{-1}$). Therefore, it suffices to compute an accurate RRD of $A^{\#}$. Consider the LDU decomposition $A^{\#} = \bar{L}\bar{D}\bar{L}^T$. The entries of $\bar{L}$ and $\bar{D}$ are quotients of minors of $A^{\#}$; thus, after some long but elementary manipulations, we get

$$\bar{d}_i = a^{(n-i)^2 - \frac{i(i-1)}{2}} \cdot \prod_{t=1}^{i-1}(a^t - 1), \tag{13}$$

$$\bar{l}_{ij} = a^{(n-1)(j-i)} \prod_{t=1}^{j-1} \frac{a^{i-j+t} - 1}{a^t - 1}. \tag{14}$$

For $|a| \geq \frac{3}{2}$, the entries $\bar{l}_{ij}$ are bounded as

$$\bar{l}_{ij} \leq \prod_{t=1}^{j-1} \frac{1}{1 - |a|^{-t}} \leq e^6.$$

For the entries of $\bar{L}^{-1}$, we obtain analogously to (12),

$$\left(\bar{L}^{-1}\right)_{ij} = (-1)^{i+j} \cdot a^{(j-i)(n-\frac{1}{2}(i-j+1))} \cdot \prod_{t=1}^{j-1} \frac{a^{i-j+t} - 1}{a^t - 1}.$$

Finally, since $n - \frac{1}{2}(i - j - 1) \geq j - 1$ we have

$$\left|\left(\bar{L}^{-1}\right)_{ij}\right| = a^{(j-i)(n-\frac{1}{2}(i-j+1))} \cdot \prod_{t=1}^{j-1} \frac{a^{i-j+t} - 1}{a^t - 1} \leq \prod_{t=1}^{j-1} \frac{1}{1 - |a|^{-t}} \leq e^6.$$

Again, $\kappa_1(\bar{L}) \le e^{12}n^2$. Therefore, $\bar{L}$ is well conditioned when $|a| \ge \frac{3}{2}$, and $A = (J\bar{L})\bar{D}(J\bar{L})^T$ is an RRD of A. The true values of $\kappa_1(\bar{L})$ are much smaller than the bound—in particular, for 30×30 matrices $\kappa_1(\bar{L}) = 13.37$ for $a = \frac{3}{2}$ and $\kappa_1(\bar{L}) = 2.35$ for $a = -\frac{3}{2}$. We have observed that $\kappa_1(\bar{L})$ decreases as $|a|$ increases.

In order to guarantee high relative accuracy in each computed entry of L, $\bar{L}$, D, and $\bar{D}$, we compute all expressions $a^i - 1$ to high relative accuracy as $a^i - 1$ when $a^i < 0$ and as $(|a| - 1)(|a|^{i-1} + |a|^{i-2} + \cdots + 1)$ when $a^i > 0$.

The cost of computing factorizations with the formulas (10) and (11), or (13) and (14), is $O(n^2)$. We need n^2 flops to compute a^i for $i = 1, 2, \ldots, n^2$, and n flops to compute $\sum_{p=0}^{j} |a|^p$ for $j = 1, 2, \ldots, n$. With this, at most n extra flops are needed to compute $a^i - 1$ for $i = 1, 2, \ldots, n$. All the diagonal elements d_i, $i = 1, 2, \ldots, n$, are computed in $6n$ flops. If $i - j = k$, the $n - k$ off-diagonal elements l_{ij} are computed in $2(n - k)$ flops. Taking into account that $k = 1, 2, \ldots, n - 1$, $n^2 + O(n)$ flops are needed to compute all off-diagonal elements l_{ij}. The total cost of computing the LDL^T factorization using (10) and (11) is $2n^2 + O(n)$ flops. A similar argument shows that the total cost of computing the $\bar{L}\bar{D}\bar{L}^T$ factorization using (13) and (14) is $2n^2 + O(n)$ flops. This extremely fast performance is important in its own right, but for the purpose of computing eigenvalues and eigenvectors to high relative accuracy the cost of applying the J-orthogonal or SSVD algorithms to the RRD is $O(n^3)$, and the cost $O(n^2)$ in the RRD computation does not significantly improve the total cost.

5. Computing an RRD of a TN matrix. The matrices with all minors nonnegative are called *totally nonnegative* (*TN*). They appear in a wide range of problems and applications (see [2, 14, 17, 24, 26] and references therein). One of the most important application is to one-dimensional oscillatory problems [16].

It has been recently shown [26, 25] that many accurate computations with nonsingular $n \times n$ TN matrices are possible when these matrices are appropriately represented as products of nonnegative bidiagonal matrices:

$$A = L^{(1)} \cdot L^{(2)} \cdots L^{(n-1)} \cdot D \cdot U^{(n-1)} \cdots U^{(2)} \cdot U^{(1)}, \tag{15}$$

where D is diagonal. This decomposition was introduced in [18, 19], and it is a unique, intrinsic representation for any nonsingular TN matrix A. This *bidiagonal decomposition* will be denoted by $\mathcal{BD}(A)$. We refer to [26, section 2.2] for a detailed explanation of the structure of the factorization (15), and also for the essential relationship between this factorization and *Neville elimination*, an alternative process to Gaussian elimination that allows one to compute (15) and to check whether a matrix is TN or not.

The numerical virtues of $\mathcal{BD}(A)$ are discussed at length in [26, 25]. This decomposition reveals the TN structure of A, and its nontrivial entries accurately determine the eigenvalues, the SVD, the inverse, and other properties of a nonsingular TN matrix. Starting with the representation (15), one can perform many highly accurate matrix computations with nonsingular TN matrices [26, 25], and, in particular, the SVD of a TN matrix A can certainly be computed given (15) (see Algorithm 6.1 from [26]). The SVD is, of course, an RRD. This approach, however, relies on the convergence properties of an algorithm for computing the SVD of a bidiagonal matrix.

Our goal in this section is to design algorithms that compute an accurate RRD of a nonsingular TN matrix given its bidiagonal factorization (15) in $O(n^3)$ time by using a *finite* process and respecting the symmetry; i.e., a symmetric TN matrix will have a symmetric RRD. This last requirement forces us to develop two algorithms: one for general TN matrices and another specifically for symmetric TN matrices.

5.1. RRD of a nonsymmetric TN matrix. Given the bidiagonal decomposition (15) of a nonsingular TN matrix A, we can accurately compute a decomposition $A = QBH^T$, where Q and H are orthogonal and B is bidiagonal, using the first part of Algorithm 6.1 from [26]. All entries of B are computed with relative errors of order ϵ, while Q and H are computed by accumulating Givens rotations with normwise errors of order ϵ, i.e., $\|Q - \widehat{Q}\|_2 = O(\epsilon)$. A similar bound holds for H. If $B = \bar{D}\bar{U}$, where $\bar{D}$ is diagonal and $\bar{U}$ is unit upper bidiagonal, then $B = \bar{D}\bar{U}$ need not be an RRD of B.

How do we compute an RRD of B? We can simply run GECP on B. Since B is acyclic (the bipartite graph of B does not have any cycles), the process of Gaussian elimination with complete pivoting will not involve any subtractions and will therefore be highly accurate (see section 6 and Algorithm 10.1 in [6]). More precisely, if P_1 and P_2^T are the permutation matrices coming from the complete pivoting strategy and $B = P_1 LDUP_2^T$, with L unit lower triangular, U unit upper triangular, and D diagonal, then all the entries of the L, D, and U factors are computed with relative errors of order ϵ.

Once we have $B = P_1 LDUP_2^T$, we obtain an RRD of A:

$$A = (QP_1L) \cdot D \cdot (UP_2^T H^T) \equiv XDY^T.$$

A direct and standard error analysis shows that the computed factors satisfy the error bounds (2). The cost of computing B is at most $\frac{16}{3}n^3 + O(n^2)$ flops [26], and forming Q and H requires not more than $6n^3 + O(n^2)$ flops. The cost of GECP on B does not exceed $\frac{2}{3}n^3 + O(n^2)$ flops and $\frac{n^3}{3}$ comparisons. Finally, the last two matrix multiplications require not more than $2n^3$ flops. The total cost does not exceed $14\,n^3 + O(n^2)$ flops and $\frac{n^3}{3}$ comparisons.

5.2. RRD of a symmetric TN matrix. The techniques of section 5.1 can certainly be used to compute an RRD of a nonsingular symmetric TN matrix given its bidiagonal decomposition. This approach does not, however, respect the symmetry of the matrix. In this subsection we present a different RRD algorithm, which does respect the symmetry.

Let the bidiagonal decomposition of a symmetric and nonsingular TN matrix A be given. Then in (15) we have $L^{(i)} = (U^{(i)})^T$. We can use the techniques of [26] to apply highly accurate Givens rotations to A and reduce A to tridiagonal form T:

$$A = QTQ^T,$$

where Q is orthogonal and $T = LDL^T$ is TN. All entries in the lower unit bidiagonal factor L and in the diagonal factor D are computed with relative errors of order ϵ, while the error in Q is $\|Q - \widehat{Q}\|_2 = O(\epsilon)$. Notice that the previous process computes $\mathcal{BD}(T)$ and Q starting from $\mathcal{BD}(A)$, and that the decomposition $T = LDL^T$ need not reveal the rank of T since L need not be well conditioned.

The remaining task in getting an accurate symmetric RRD is to compute, given $\mathcal{BD}(T)$, an accurate RRD of T by using symmetric GECP:

$$T = P\bar{L}\bar{D}\bar{L}^T P^T,$$

where P is a permutation matrix, $\bar{L}$ is unit lower triangular, and $\bar{D}$ is diagonal. Then the symmetric RRD of A is

$$A = (QP\bar{L})\bar{D}(QP\bar{L})^T.$$

We will show how to compute P and all the entries of $\bar{L}$ and $\bar{D}$ with relative errors of order ϵ. Our approach is based on two key ideas: the first is that T is positive definite, and thus the pivoting strategy in GECP will be diagonal, and the second is that the elements of $\bar{L}$ and $\bar{D}$ are signed quotients of minors of T. We will proceed in three steps as follows: (a) The bidiagonal factorization of a principal submatrix of T is accurately computed starting from $\mathcal{BD}(T)$ in Algorithm 3; (b) this is used in Algorithm 4 to compute accurate minors of T; and (c) the elements of $\bar{L}$ and $\bar{D}$ are computed as quotients of minors in Algorithm 5, together with P.

We can summarize the algorithm to compute a symmetric RRD of a nonsingular symmetric TN matrix A as follows.

Algorithm 2. Computing a symmetric RRD $A = XDX^T$ of a symmetric nonsingular TN matrix A given $\mathcal{BD}(A)$.

1. Apply Givens rotations as in [26, section 4.3] to compute an orthogonal matrix Q and $\mathcal{BD}(T)$ of a symmetric TN tridiagonal matrix T such that $A = QTQ^T$.
2. Compute a symmetric RRD of $T = P\bar{L}\bar{D}\bar{L}^TP^T$ using Algorithm 5.
3. Multiply to get $A = (QP\bar{L})\bar{D}(QP\bar{L})^T \equiv XDX^T$.

Step 1 requires not more than $\frac{8}{3}n^3 + O(n^2)$ flops to get $\mathcal{BD}(T)$ (see [26]) and not more than $3n^3 + O(n^2)$ additional flops to compute Q. We will see that the cost of step 2 does not exceed $14\frac{1}{3}n^3 + O(n^2)$. Finally, the cost of step 3 does not exceed n^3. The total cost of Algorithm 2 does not exceed $21n^3 + O(n^2)$ flops.

We will show that the computation of the symmetric RRD $T = P\bar{L}\bar{D}\bar{L}^TP^T$ is subtraction free. Combining this with the errors in Q, $\mathcal{BD}(T)$, and matrix multiplication, it can be easily shown that the computed RRD satisfies (2).

Once a symmetric RRD, $A = XDX^T$, of the TN matrix A is computed, the eigenvalues and eigenvectors of A can be accurately computed by using the one-sided Jacobi algorithm to get the singular values and left singular vectors of $XD^{1/2}$ [10, section 5.4.3], [9]. The Jacobi rotations in this procedure have to be applied on the left, and the whole process respects the symmetry. The techniques introduced in [26] allow us to develop another symmetric method to compute accurate eigenvalues of a nonsingular symmetric TN matrix A: First, step 1 of Algorithm 2 is performed to get $T = LDL^T$; next, the differential quotient-difference algorithm with shifts (dqds) [13] is applied on the Cholesky factor $LD^{1/2}$ to compute its accurate singular values. This approach does not use RRDs.

5.2.1. The bidiagonal decomposition of a principal submatrix of a TN tridiagonal symmetric matrix.

Let T be a nonsingular symmetric TN tridiagonal matrix[3] and S be a principal submatrix of T. The purpose of this section is to accurately compute $\mathcal{BD}(S)$ given $\mathcal{BD}(T)$.

Consider first the simple special case when the principal submatrix S is obtained

[3] The results of this section remain valid for positive definite tridiagonal matrices because a positive definite tridiagonal matrix is TN if and only if its off-diagonal elements are nonnegative [16, p. 81]. Therefore, any positive definite tridiagonal matrix is similar to a TN matrix through a diagonal similarity transformation with elements ±1.

by erasing the ith row and the ith column of T:

$$T = \begin{bmatrix} t_{11} & t_{12} & & \\ t_{21} & \ddots & \ddots & \\ & \ddots & \ddots & t_{n-1,n} \\ & & t_{n,n-1} & t_{nn} \end{bmatrix};$$

$$\begin{aligned} S &= T([1:i-1, i+1:n], [1:i-1, i+1:n]) \\ &= \left[\begin{array}{cccc|cccc} t_{11} & t_{12} & & & & & & \\ t_{21} & \ddots & \ddots & & & & & \\ & \ddots & \ddots & t_{i-2,i-1} & & & & \\ & & t_{i-1,i-2} & t_{i-1,i-1} & & & & \\ \hline & & & & t_{i+1,i+1} & t_{i+1,i+2} & & \\ & & & & t_{i+2,i+1} & \ddots & \ddots & \\ & & & & & \ddots & \ddots & t_{n-1,n} \\ & & & & & & t_{n,n-1} & t_{nn} \end{array}\right]. \end{aligned}$$

Once we figure out how to compute $\mathcal{BD}(S)$ from $\mathcal{BD}(T)$, we can proceed by induction and erase other rows and columns of S to obtain the bidiagonal decomposition of any principal submatrix of T.

Since the process of Neville elimination of S and T does not differ for the first $i-1$ rows and columns, we have $\mathcal{BD}(S(1:i-1, 1:i-1)) = \mathcal{BD}(T(1:i-1, 1:i-1))$, and we need only compute $\mathcal{BD}(S(i+1:n, i+1:n))$. Therefore, we may assume that $i = 1$ without any loss of generality.

Let $\mathcal{BD}(T)$ and $\mathcal{BD}(S)$ be given as

$$T = LDL^T \quad \text{and} \quad S = T(2:n, 2:n) = \bar{L}\bar{D}\bar{L}^T,$$

where $D = \operatorname{diag}(d_i)_{i=1}^n$, $\bar{D} = \operatorname{diag}(\bar{d}_i)_{i=2}^n$, and the unit lower bidiagonal matrices L and $\bar{L}$ have off-diagonal elements l_i, $i = 1, 2, \ldots, n-1$, and $\bar{l}_i$, $i = 2, 3, \ldots, n-1$, respectively. From $T = LDL^T$ we have

$$t_{11} = d_1; \;\; t_{ii} = l_{i-1}^2 d_{i-1} + d_i; \;\; t_{i-1,i} = l_{i-1} d_{i-1}, \quad i = 2, 3, \ldots, n, \tag{16}$$

and from $T(2:n, 2:n) = \bar{L}\bar{D}\bar{L}^T$ we get

$$t_{22} = \bar{d}_2; \;\; t_{ii} = \bar{l}_{i-1}^2 \bar{d}_{i-1} + \bar{d}_i; \;\; t_{i-1,i} = \bar{l}_{i-1} \bar{d}_{i-1}, \quad i = 3, 4, \ldots, n. \tag{17}$$

By comparing (16) and (17), we obtain

$$\begin{aligned} \bar{d}_2 &= l_1^2 d_1 + d_2, \\ \bar{d}_i &= d_i + l_{i-1}^2 d_{i-1} - \bar{l}_{i-1}^2 \bar{d}_{i-1}, \quad i = 3, 4, \ldots, n, \\ \bar{l}_i \bar{d}_i &= l_i d_i, \quad i = 2, 3, \ldots, n-1. \end{aligned} \tag{18}$$

We introduce auxiliary variables $z_i \equiv \bar{d}_i - d_i$ and get rid of the subtraction in (18):

$$\begin{aligned} z_2 &= l_1^2 d_1, \\ \bar{d}_2 &= z_2 + d_2, \\ \bar{l}_i &= l_i d_i / \bar{d}_i, \quad i = 2, \ldots, n-1, \\ z_{i+1} &= \bar{d}_{i+1} - d_{i+1} = l_i^2 d_i - \bar{l}_i^2 \bar{d}_i = (\bar{d}_i - d_i) l_i^2 d_i / \bar{d}_i = l_i \bar{l}_i z_i, \qquad i = 2, \ldots, n-1, \\ \bar{d}_{i+1} &= z_{i+1} + d_{i+1}, \quad i = 2, \ldots, n-1. \end{aligned} \tag{19}$$

The iterations (19) need only be performed for those $i \geq 2$ for which $l_i \neq 0$. These iterations therefore cost $5(j-1)$, where $j < n$ is the smallest index such that $l_j = 0$ (or $j = n$ if $l_k \neq 0$ for $k = 1, 2, \ldots, n-1$). In the general case, when we remove the ith row and the ith column of T, the cost is $5(j-i)$, where $j \geq i$ is defined as above.

We now implement the recurrences (19).

Algorithm 3. Let $T = LDL^T$ be a nonsingular symmetric TN tridiagonal matrix, where $D = \operatorname{diag}(d_i)_{i=1}^n$, $d_i > 0$, $i = 1, 2, \ldots, n$, and L is a unit lower bidiagonal matrix with off-diagonal entries $l_i \geq 0, i = 1, 2, \ldots, n-1$. Let $\alpha = \{\alpha_1, \alpha_2, \ldots, \alpha_r\}$, $1 \leq \alpha_1 < \alpha_2 < \cdots < \alpha_r \leq n$ be a subset of indices. Given the vectors $d = (d_1, d_2, \ldots, d_n)$ and $l = (l_1, \ldots, l_{n-1})$, the following subtraction-free algorithm computes the decomposition $T(\alpha, \alpha) = \bar{L}\bar{D}\bar{L}^T$ in at most $5r$ time:

```
function $[\bar{d}, \bar{l}] = $ TNTridiagSubmatrix$(d, l, \alpha)$
$n = $ length$(d)$
$\bar{d} = d;\ \bar{l} = l;\ \bar{l}_n = 0$
Let $\beta$ be the complement of $\alpha$ in the set $\{1, 2, \ldots, n\}$
(In MATLAB notation: $\beta = [1:n];\ \beta(\alpha) = 0;\ \beta = \beta(\beta > 0)$)
for $k = $ length$(\beta) : -1 : 1$
    if $\beta_k < n$
        $z = d_{\beta_k} l_{\beta_k}^2$
        $j = \beta_k + 1$
        $\bar{d}_j = z + d_j$
        while $\bar{l}_j \neq 0$
            $\bar{l}_j = l_j d_j / \bar{d}_j$
            $z = l_j \bar{l}_j z$
            $\bar{d}_{j+1} = z + d_{j+1}$
            $j = j + 1$
        end
    end
    $\bar{l}_{\beta_k - 1} = 0;\ \bar{l}_{\beta_k} = 0$
end
$\bar{d} = \bar{d}(\alpha);\ \bar{l} = \bar{l}(\alpha);\ \bar{l} = \bar{l}(1 : (r-1))$
```

5.2.2. A minor of a TN tridiagonal symmetric matrix. Next we consider the problem of accurately computing the value of any minor of a nonsingular symmetric TN tridiagonal matrix T:

$$T(\alpha, \beta) = T([i_1, \ldots, i_k], [j_1, \ldots, j_k]),$$

where $\alpha = [i_1, i_2, \ldots, i_k]$, $1 \leq i_1 < i_2 < \ldots < i_k \leq n$, and $\beta = [j_1, j_2, \ldots, j_k]$, $1 \leq j_1 < j_2 < \ldots < j_k \leq n$.

Let $1 \leq k_1 < k_2 < \cdots < k_r \leq k$ be all indices such that $i_{k_s} \neq j_{k_s}$, $s = 1, 2, \ldots, r$, and let $\gamma = \{i_1, i_2, \ldots, i_k\} \backslash \{i_{k_1}, \ldots, i_{k_r}\}$. Then [16, p. 80]

$$\det T(\alpha, \beta) = \det T(\gamma, \gamma) t_{i_{k_1} j_{k_1}} \cdots t_{i_{k_r} j_{k_r}}. \tag{20}$$

The minor $\det T(\gamma, \gamma)$ can be computed by first computing the bidiagonal decomposition of $T(\gamma, \gamma)$ using Algorithm 3 (then $\det T(\gamma, \gamma) = \bar{d}_1 \bar{d}_2 \cdots \bar{d}_{k-r}$). Any entry $t_{i_{k_s} j_{k_s}}$, $i_{k_s} \neq j_{k_s}$, equals either zero, $t_{m,m+1}$, or $t_{m+1,m}$. The latter two are easily computed from $T = LDL^T$: $t_{m,m+1} = t_{m+1,m} = d_m l_m$. The total cost of computing any minor $\det T(\alpha, \beta)$ following this procedure does not exceed $6k$ flops.

Remark 1. A set of indices $\mathtt{z} \subset \{1, 2, \ldots, n\}$ can be sorted in increasing order in $4n$ time by using the following MATLAB commands:

```
x=1:n; x(z)=0; y=1:n; z=y(x==0);
```

therefore, we can sort the index sets in $T(\alpha, \beta)$ in $8n$ time and allow index sets in arbitrary order in Algorithm 4 below.

Algorithm 4 (minor of a TN tridiagonal matrix). Let $T = LDL^T$ be a nonsingular symmetric TN tridiagonal matrix with notation as in Algorithm 3. Given the vectors d and l, and two sets of indices α and β, the following subtraction-free algorithm computes $|\det T(\alpha, \beta)|$ to high relative accuracy in at most $14n$ time:

```
function $f = \mathtt{TNTridiagMinor}(d, l, \alpha, \beta)$
    ...first sort $\alpha$ and $\beta$ in increasing order (see Remark 1 above)...
$f = 1$; $\gamma = [\,]$
for $i = 1 : \mathtt{length}(\alpha)$
    if $\alpha_i = \beta_i$
        $\gamma = [\gamma, \alpha_i]$
    elseif $|\alpha_i - \beta_i| = 1$
        $f = f d_s l_s$, where $s = \min(\alpha_i, \beta_i)$
    else
        $f = 0$; return
    end
end
$[\bar{d}, \bar{l}] = \mathtt{TNTridiagSubmatrix}(d, l, \gamma)$
$f = f \bar{d}_1 \bar{d}_2 \cdots \bar{d}_s$, where $s = \mathtt{length}(\gamma)$
```

5.2.3. Computing an RRD of a TN tridiagonal symmetric matrix. In this section we present an $O(n^3)$ algorithm which, given the factorization $T = LDL^T$ of a nonsingular symmetric TN tridiagonal matrix T, computes an accurate, symmetric RRD of T. The RRD in question is the LDU decomposition of T resulting from GECP, with L (resp., U) being a unit lower (resp., upper) triangular matrix. We compute each entry of this LDU decomposition as a quotient of minors of T. We compute each minor of T accurately using Algorithm 4.

Since T is positive definite, the pivoting in GECP will be diagonal. The pivot order is determined by comparing the diagonal entries in the Schur complements; if $\gamma = [\gamma_1, \gamma_2, \ldots, \gamma_k]$ is the current pivot order at step k, and $\alpha = \{1, 2, \ldots, n\} \backslash \gamma$, then the diagonal entries of the kth Schur complement are[4]

$$\frac{\det T([\gamma, \alpha_j], [\gamma, \alpha_j])}{\det T(\gamma, \gamma)}, \qquad j = 1, 2, \ldots, n - k. \tag{21}$$

We need only compare the numerators in (21) and we compute those using Algorithm 4.

[4] These expressions for the entries of the Schur complements are valid if for each step of Gaussian elimination the row and column containing the chosen pivot are moved to the first positions in the corresponding Schur complement and the rows and columns between the first and the ones containing the pivot are displaced down one position. This is not the usual implementation of pivoting in Gaussian elimination, which simply interchanges the first row and the first column with the pivot row and the pivot column, respectively [21, 22]. Obviously both implementations produce similar bounds on the elements of L and U, and, therefore, they are equivalent from the point of view of computing RRDs.

Once we obtain the pivot order γ, the entries of the LDU decomposition $T = P\bar{L}\bar{D}\bar{L}^T P^T$ resulting from GECP are computed as

$$\bar{D}_{ii} = \frac{\det T(\gamma(1:i), \gamma(1:i))}{\det T(\gamma(1:i-1), \gamma(1:i-1))}; \tag{22}$$

$$\bar{L}_{ji} = \frac{\det T(\gamma(1:i), \gamma([1:i-1, j]))}{\det T(\gamma(1:i), \gamma(1:i))}, \qquad j > i, \tag{23}$$

with each minor in (22) and (23) computed using Algorithm 4.

The sign of the minor $\det T(\gamma(1:i), \gamma([1:i-1, j]))$ equals $\operatorname{sgn}(\gamma(1:i)) \cdot \operatorname{sgn}(\gamma([1:i-1, j]))$. Here $\operatorname{sgn}(\delta)$ is the *sign* of $[\delta_1, \delta_2, \dots]$ as a permutation of the ordered set $\{\delta_1, \delta_2, \dots\}$, defined as $\operatorname{sgn}(\delta) \equiv (-1)^t$, where $t \equiv \#\{(k,l) | k < l \text{ and } \delta_k > \delta_l\}$; i.e., t is the minimum number of transpositions necessary to sort the elements of δ in increasing order. The first $i-1$ entries of $\gamma(1:i)$ and $\gamma([1:i-1, j])$ coincide; therefore the sign of $\det T(\gamma(1:i), \gamma([1:i-1, j]))$ equals $(-1)^s$, $s = \sum_{k=1}^{i-1} \mathtt{xor}(\gamma_i < \gamma_k, \gamma_j < \gamma_k)$.

Algorithm 5 (GECP on a TN tridiagonal matrix). Let $T = LDL^T$ be a nonsingular symmetric TN tridiagonal matrix. Given the vectors d and l (defined in Algorithm 3), the following subtraction-free algorithm computes the decomposition of $T = P\bar{L}\bar{D}\bar{L}^T P^T$ resulting from Gaussian elimination with complete pivoting. Every entry of $\bar{D}$ and $\bar{L}$ is computed to high relative accuracy, and the total cost does not exceed $14\frac{1}{3}n^3 + O(n^2)$.

```
function $[P, \bar{L}, \bar{D}] = \mathtt{TNTridiagGECP}(d, l)$
$n = \mathtt{length}(d)$
$\bar{L} = \mathtt{eye}(n)$; $P = \mathtt{eye}(n)$; $\bar{D} = \mathtt{eye}(n)$;
$\alpha = 1:n$; $\gamma = [\,]$
            ...First, determine the pivot order...
for $i = 1:n$
    for $j = 1:n-i+1$
        $z_j = \mathtt{TNTridiagMinor}(d, l, [\gamma, \alpha_j], [\gamma, \alpha_j])$
    end
    Let $m$ be such that $z_m = \max_{1 \le j \le n-i+1} z_j$
    $\gamma_i = \alpha_m$
    $\alpha = \alpha([1:m-1, m+1:n-i+1])$
    $t_i = z_m$
end
            ...Next, compute the entries of $\bar{D}$ and $\bar{L}$ using (22) and (23)...
for $i = 1:n$
    $\bar{D}_{ii} = t_i / t_{i-1}$   ( ...assume $t_0 = 1$)
    for $j = i+1:n$
            ...Compute the sign of $\bar{L}_{ji}$...
        $s = 1$
        for $k = 1:i-1$
            $s = s \cdot (-1)$^$\mathtt{xor}(\gamma_i < \gamma_k, \gamma_j < \gamma_k)$
        end
        $\bar{L}_{ji} = s \cdot \mathtt{TNTridiagMinor}(d, l, \gamma(1:i), \gamma([1:i-1, j])) / t_i$
    end
end
$P = P(:, \gamma)$
```

6. Numerical experiments. We performed extensive numerical tests and confirmed the accuracy and cost of our algorithms. More precisely, we combined Algorithm 1 and the one-sided J-orthogonal algorithm [33, Algorithm 3.3.1, page 66] to compute, preserving the symmetry, accurate eigenvalues and eigenvectors of symmetric diagonally scaled Cauchy matrices with different dimensions and several distributions of random Cauchy parameters. The output was compared with that of another $O(n^3)$ accurate algorithm (nonsymmetric RRD computed as in [5] combined with the SSVD algorithm from [11]), and also with the output from the MATLAB `eig` function in variable precision arithmetic (with precision set to $\log_{10} \kappa_2(C) + 20$ decimal digits, guaranteeing at least 16 correct significant digits in each eigenvalue). The output of all three algorithms agreed to at least 14 digits for all the eigenvalues, including the ones with tiniest absolute values. The computed eigenvectors also satisfied the bounds (1). Most test matrices had condition numbers well in excess of 10^{16}, so conventional eigenvalue algorithms (e.g., the MATLAB function `eig` in double [23] precision) failed to get any correct digits in the eigenvalues with tiniest absolute values and in the direction of the eigenvectors corresponding to these eigenvalues (when at least two tiny eigenvalues λ_i such that $|\lambda_i| \leq 10^{-16}\|C\|_2$ were present). We performed similar tests on symmetric Vandermonde matrices for several dimensions and choices of the parameter a, and also for symmetric TN matrices. In the case of symmetric Vandermonde matrices, we also tested matrices with $\frac{2}{3} < |a| < \frac{3}{2}$ and verified that the factorizations obtained with the approach in section 4 are not RRDs when $|a|$ is close to one ($\kappa_2(L) \to 2^n$ as $|a| \to 1$). For these matrices, eigenvalues and eigenvectors to high relative accuracy can be obtained, at present, only through the nonsymmetric procedure by first computing a nonsymmetric RRD as in [5] and then applying the SSVD algorithm from [11].

We present in detail one of our tests. We consider the 20×20 symmetric Vandermonde matrix A with $a = \frac{1}{2}$; see (9). The condition number of A is $\kappa_2(A) \approx 3.5 \cdot 10^{53}$. We compute its eigenvalues using the following algorithms:

- Algorithm A: The MATLAB `eig` function with 75-digit arithmetic.
- Algorithm B: Compute a symmetric RRD using the formulas in section 4 followed by the J-orthogonal algorithm [33, Algorithm 3.3.1, page 66].
- Algorithm C: Compute a nonsymmetric RRD as in [5] followed by the SSVD algorithm of [11].
- Algorithm D: The MATLAB `eig` function in double [23] precision arithmetic.

The output of Algorithms A, B, and C agreed to at least 14 digits, so we plotted only the output of Algorithms B and D in Figure 6.1. Since $\kappa_2(A) \approx 3.5 \cdot 10^{53}$, Algorithm A computed all eigenvalues with at least 16 significant decimal digits of accuracy. Algorithms B and C guarantee high relative accuracy for the computed eigenvalues. The results from those algorithms agreed with the ones from Algorithm A to at least 14 digits. In contrast, the traditional Algorithm D returned only the eigenvalues of largest absolute value accurately, with the accuracy gradually decreasing until the eigenvalues with magnitude smaller than $O(\epsilon)||A||_2$ were computed with no correct digits at all.

Appendix. Rounding error analysis for diagonally scaled Cauchy matrices. Theorem 3.1 is proved in this appendix in a more general setting. The error analysis we present remains valid when the Bunch–Parlett method is applied on any matrix for which it is possible to compute the entries of its Schur complements with relative errors bounded by $k\epsilon/(1-k\epsilon)$, where k is an integer positive number and ϵ is the machine precision. For scaled Cauchy matrices, $k = 8n$ according to (7).

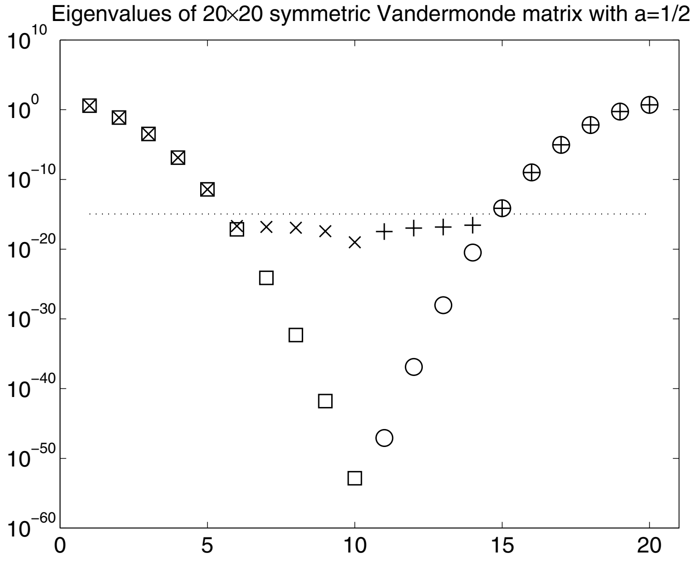

FIG. 6.1. *Plots of the absolute values of the eigenvalues of the* 20×20 *symmetric Vandermonde matrix with* $a = \frac{1}{2}$. *The "□" and "◯" symbols represent, respectively, the negative and positive eigenvalues computed with an accurate algorithm. The "×" and "+" symbols represent, respectively, the negative and positive eigenvalues computed by Algorithm D (implemented as the MATLAB function* `eig` *in double precision arithmetic). Data below the dotted line may be inaccurate for Algorithm D.*

We use the conventional error model for floating point arithmetic [22, section 2.2]:

$$\texttt{fl}(a \odot b) = (a \odot b)(1 + \delta),$$

where a and b are real floating point numbers, $\odot \in \{+, -, \times, /\}$, and $|\delta| \leq \epsilon$. Moreover, we assume that neither overflow nor underflow occurs. We also use the following notation introduced in [22, Chapter 3]: θ_q is any number such that

$$|\theta_q| \leq \frac{q\epsilon}{1 - q\epsilon} \equiv \gamma_q. \tag{24}$$

Moreover, the results in [22, Lemma 3.3] will be frequently used throughout this section without being explicitly referred to. We will assume that $0 < \gamma_q$ for all the symbols γ_q appearing in this section.

In what follows, α is the parameter used in the Bunch–Parlett pivoting strategy to decide whether a 1×1 or 2×2 pivot is selected (see Algorithm 1). We present the error bounds in this section depending on α, where $0 < \alpha < 1$. Thus values different from the classical one, $\alpha = (1 + \sqrt{17})/8$, are also considered.

A.1. Auxiliary results on the Jacobi method. Let us write the Jacobi procedure [21] to orthogonally diagonalize a 2×2 real symmetric matrix as a matrix factorization. The following equation holds:

$$\begin{bmatrix} a & c \\ c & b \end{bmatrix} = \begin{bmatrix} cs & sn \\ -sn & cs \end{bmatrix} \begin{bmatrix} a - c\,t & 0 \\ 0 & b + c\,t \end{bmatrix} \begin{bmatrix} cs & -sn \\ sn & cs \end{bmatrix}, \tag{25}$$

where

$$\zeta = \frac{b-a}{2c}, \qquad t = \frac{\operatorname{sign}(\zeta)}{|\zeta| + \sqrt{1+\zeta^2}}, \tag{26}$$

$$cs = \frac{1}{\sqrt{1+t^2}}, \qquad sn = cs \cdot t, \tag{27}$$

and $\operatorname{sign}(0) = 1$.

In general, disastrous cancellations may appear in the Jacobi procedure above, and the eigenvalues computed in floating point arithmetic may be inaccurate. However, it is well known that the Jacobi procedure is backward stable because only orthogonal matrices are involved. Theorem A.1 below shows this, providing precise error bounds that we will use in the detailed error analysis of the next subsections. The Jacobi method computes accurate eigenvalues for well-conditioned matrices because it is backward stable. We will see that this is the case for the 2×2 pivots selected by the Bunch–Parlett pivoting strategy.

THEOREM A.1. *Let*

$$\widetilde{A} = \begin{bmatrix} \tilde a & \tilde c \\ \tilde c & \tilde b \end{bmatrix}$$

be a matrix of real floating point numbers. Let us apply to $\widetilde{A}$ *the Jacobi procedure* (25) *in floating point arithmetic with machine precision* ϵ. *Let* $\widetilde{cs}, \widetilde{sn}, \tilde\lambda_1 = \tilde a - \tilde c\, \tilde t$, *and* $\tilde\lambda_2 = \tilde b + \tilde c\, \tilde t$ *be the exact magnitudes for* $\widetilde{A}$, *and let* $\widehat{cs}, \widehat{sn}, \hat\lambda_1$, *and* $\hat\lambda_2$ *be the corresponding computed counterparts. Then*

1. $\widehat{cs} = \widetilde{cs}\,(1+\theta_{113})$.
2. $\widehat{sn} = \widetilde{sn}\,(1+\theta_{141})$.
3. $\hat\lambda_1 = \tilde\lambda_1 + e_1$ *with* $|e_1| \leq (|\tilde a| + |\tilde c\, \tilde t|)\gamma_{29}$.
4. $\hat\lambda_2 = \tilde\lambda_2 + e_2$ *with* $|e_2| \leq (|\tilde b| + |\tilde c\, \tilde t|)\gamma_{29}$.

Moreover, the computed eigendecomposition

$$\begin{bmatrix} \widehat{cs} & \widehat{sn} \\ -\widehat{sn} & \widehat{cs} \end{bmatrix} \begin{bmatrix} \hat\lambda_1 & 0 \\ 0 & \hat\lambda_2 \end{bmatrix} \begin{bmatrix} \widehat{cs} & -\widehat{sn} \\ \widehat{sn} & \widehat{cs} \end{bmatrix}$$

is nearly the exact eigendecomposition of $\widetilde{A} + E$; *more precisely,*

$$\widetilde{A} + E = \begin{bmatrix} \widetilde{cs} & \widetilde{sn} \\ -\widetilde{sn} & \widetilde{cs} \end{bmatrix} \begin{bmatrix} \hat\lambda_1 & 0 \\ 0 & \hat\lambda_2 \end{bmatrix} \begin{bmatrix} \widetilde{cs} & -\widetilde{sn} \\ \widetilde{sn} & \widetilde{cs} \end{bmatrix},$$

where $\|E\|_2 \leq \sqrt{2}\,\gamma_{29} \|\widetilde{A}\|_F \leq 2\,\gamma_{29}\, \|\widetilde{A}\|_2$.

Proof. The bounds for $\widehat{cs}$, $\widehat{sn}$, $\hat\lambda_1$, and $\hat\lambda_2$ follow from a direct application of Lemmas 3.1 and 3.3 in [22]. For the backward error bound, notice that

$$E = \begin{bmatrix} \widetilde{cs} & \widetilde{sn} \\ -\widetilde{sn} & \widetilde{cs} \end{bmatrix} \begin{bmatrix} e_1 & 0 \\ 0 & e_2 \end{bmatrix} \begin{bmatrix} \widetilde{cs} & -\widetilde{sn} \\ \widetilde{sn} & \widetilde{cs} \end{bmatrix}.$$

Then $\|E\|_2 = \max\{|e_1|, |e_2|\} \leq \gamma_{29} \max\{|\tilde a| + |\tilde c\, \tilde t|, |\tilde b| + |\tilde c\, \tilde t|\} \leq \gamma_{29} \max\{|\tilde a| + |\tilde c|, |\tilde b| + |\tilde c|\} \leq \sqrt{2}\,\gamma_{29} \|\widetilde{A}\|_F$. ◻

A.2. Properties of 2 × 2 Bunch–Parlett pivots. The 2 × 2 pivots selected by the Bunch–Parlett complete pivoting strategy are very well conditioned symmetric indefinite matrices. The next lemma quantifies this fact.

Lemma A.2. *Let H be a real symmetric 2×2 matrix such that $\alpha\,|h_{21}| > \max\{|h_{11}|, |h_{22}|\}$, where $0 < \alpha < 1$. Then the spectral condition number, $\kappa_2(H)$, of H is bounded as*

$$\kappa_2(H) < \frac{1+\alpha}{1-\alpha}.$$

This bound cannot be improved. In particular, if $\alpha = 0.6404$, then $\kappa_2(H) < 4.6$.

Proof. Let us write the matrix H as

$$H = \left[\begin{array}{cc} 0 & h_{21} \\ h_{21} & 0 \end{array}\right] + \left[\begin{array}{cc} h_{11} & 0 \\ 0 & h_{22} \end{array}\right] \equiv H_0 + H_1.$$

The singular values of H_0 are both equal to $|h_{21}|$. Then using Weyl's perturbation theorem for singular values (see, for instance, [10, Corollary 5.1]), we get

$$\kappa_2(H) \le \frac{|h_{21}| + \|H_1\|_2}{|h_{21}| - \|H_1\|_2} < \frac{|h_{21}| + \alpha|h_{21}|}{|h_{21}| - \alpha|h_{21}|} = \frac{1+\alpha}{1-\alpha}.$$

The bound cannot be improved because the matrix $H = \left[\begin{smallmatrix} \alpha & 1 \\ 1 & \alpha \end{smallmatrix}\right]$ has $\kappa_2(H) = \frac{1+\alpha}{1-\alpha}$. □

The entries of the eigenvectors of the 2 × 2 pivots selected by the Bunch–Parlett strategy are bounded below by 1/3. This means that small normwise variations in the eigenvectors imply small variations in the components.

Lemma A.3. *Let H be a real symmetric 2×2 matrix such that $\alpha\,|h_{21}| > \max\{|h_{11}|, |h_{22}|\}$, where $0 < \alpha < 1$. Let $\left[\begin{smallmatrix} cs & sn \\ -sn & cs \end{smallmatrix}\right]$ be the orthogonal eigenvector matrix of H; then*

$$\frac{1}{\sqrt{2}} \le cs \le \frac{\alpha + \sqrt{1+\alpha^2}}{\sqrt{1 + \left(\alpha + \sqrt{1+\alpha^2}\right)^2}},$$

$$\frac{1}{\sqrt{1 + \left(\alpha + \sqrt{1+\alpha^2}\right)^2}} \le sn \le \frac{1}{\sqrt{2}}.$$

In particular, if $\alpha = 0.6404$, then $0.47 \le sn$ and $cs \le 0.88$. The following simple lower bound for sn is valid for any α: $1/3 < sn$.

Proof. From (26), $|\zeta| \le \alpha$ and $1/(\alpha + \sqrt{1+\alpha^2}) \le |t| \le 1$. Combining these bounds with (27), the lemma is proved. □

A.3. Forward errors in RRDs. The entries of the Schur complements of diagonally scaled Cauchy matrices are computed by (7) with relative errors less than γ_{8n}. In this section we assume that the entries of the Schur complements are computed with relative errors less than γ_k; thus the error analysis remains valid for other cases.

A nagging problem will arise in the analysis: the computed 2 × 2 pivots fulfill the conditions of Bunch and Parlett, i.e., $\alpha\,|\hat{h}_{21}| > \max\{|\hat{h}_{11}|, |\hat{h}_{22}|\}$, but the exact pivots may not. This justifies the following lemma.

Lemma A.4. *Let*

$$\widetilde{A} = \left[\begin{array}{cc} a(1+\beta_a) & c(1+\beta_c) \\ c(1+\beta_c) & b(1+\beta_b) \end{array}\right] \equiv \left[\begin{array}{cc} \tilde{a} & \tilde{c} \\ \tilde{c} & \tilde{b} \end{array}\right]$$

be a matrix of real floating point numbers, where $\max\{|\beta_a|, |\beta_b|, |\beta_c|\} \leq \gamma_k$*, and* $\alpha\, |\tilde{c}| > \max\{|\tilde{a}|, |\tilde{b}|\}$*, with* $0 < \alpha < 1$*. Denote* $A \equiv \left[\begin{smallmatrix} a & c \\ c & b \end{smallmatrix}\right]$*. If*

$$4\sqrt{2}\,\frac{1+\alpha}{1-\alpha}\,\gamma_k \leq 1, \tag{28}$$

then

$$\kappa_2(A) \leq 2\,\frac{1+\alpha}{1-\alpha}. \tag{29}$$

Proof. Notice that

$$\widetilde{A} = A + E_1 \qquad \text{with} \qquad \|E_1\|_F \leq \gamma_k \|A\|_F \leq \sqrt{2}\,\gamma_k \|A\|_2. \tag{30}$$

Let $\sigma_1 \geq \sigma_2$ and $\tilde{\sigma}_1 \geq \tilde{\sigma}_2$ be the singular values of A and $\widetilde{A}$, respectively. Now Corollary 5.1 from [10] implies

$$\kappa_2(\widetilde{A}) = \frac{\tilde{\sigma}_1}{\tilde{\sigma}_2} \geq \frac{\sigma_1 - \sqrt{2}\,\gamma_k \|A\|_2}{\sigma_2 + \sqrt{2}\,\gamma_k \|A\|_2} = \kappa_2(A)\,\frac{1 - \sqrt{2}\,\gamma_k}{1 + \sqrt{2}\,\gamma_k\,\kappa_2(A)}.$$

From this we get

$$\kappa_2(A) \leq \frac{\kappa_2(\widetilde{A})}{1 - 2\sqrt{2}\,\gamma_k\,\kappa_2(\widetilde{A})}.$$

The result follows from (28) and Lemma A.2, which implies

$$\kappa_2(\widetilde{A}) \leq (1+\alpha)/(1-\alpha). \qquad \square$$

Obviously the rigorous factor 2 in (29) is pessimistic, and in practice $\kappa_2(A) \approx \kappa_2(\widetilde{A}) \leq (1+\alpha)/(1-\alpha)$. However, at the cost of the nonessential factor 2, Lemma A.4 allows us to get rigorous error bounds instead of first-order error bounds. In particular, we can prove the following lemma.

LEMMA A.5. *Let*

$$\widetilde{A} \equiv \begin{bmatrix} \tilde{a} & \tilde{c} \\ \tilde{c} & \tilde{b} \end{bmatrix} = \begin{bmatrix} a(1+\beta_a) & c(1+\beta_c) \\ c(1+\beta_c) & b(1+\beta_b) \end{bmatrix}$$

be a matrix of real floating point numbers, where $\max\{|\beta_a|, |\beta_b|, |\beta_c|\} \leq \gamma_k$*, and* $\alpha\, |\tilde{c}| > \max\{|\tilde{a}|, |\tilde{b}|\}$*, with* $0 < \alpha < 1$*. Denote* $A \equiv \left[\begin{smallmatrix} a & c \\ c & b \end{smallmatrix}\right]$*. Let the eigenvalues of* A *be* $\lambda_1 \geq \lambda_2$*;* v_1 *and* v_2 *be the corresponding orthonormal eigenvectors; and* cs *and* sn *be the components of the eigenvectors, i.e.,* $v_1 = [cs\,,\,-sn]^T$ *and* $v_2 = [sn\,,\,cs]^T$ *or vice versa. Let* $\hat{\lambda}_1, \hat{\lambda}_2, \hat{v}_1, \hat{v}_2, \widehat{cs}$*, and* $\widehat{sn}$ *be their corresponding computed counterparts by applying the Jacobi process in* (25)–(27) *to* $\widetilde{A}$ *in floating point arithmetic with machine precision* ϵ*. If*

$$4\sqrt{2}\,\frac{1+\alpha}{1-\alpha}\,\gamma_{k+29} \leq 1 \qquad \text{and} \qquad \gamma_{141+48k} \leq 1, \tag{31}$$

then

1.

$$\frac{|\hat{\lambda}_i - \lambda_i|}{|\lambda_i|} \leq 4\,\frac{1+\alpha}{1-\alpha}\,\gamma_{k+29}, \qquad i = 1, 2; \tag{32}$$

2.

$$\|\hat{v}_i - v_i\|_2 \leq \gamma_{4k+141}, \qquad i = 1, 2; \tag{33}$$

3.

$$\widehat{cs} = cs\,(1 + \theta_{16k+113}) \qquad \textit{and} \qquad \widehat{sn} = sn\,(1 + \theta_{48k+141}). \tag{34}$$

We have chosen to get error bounds for cs and sn that do not depend on α. At the cost of complicating the bounds, it is possible to get sharper bounds depending on α. Moreover, we have frequently overestimated the bounds to get simpler expressions. It is well known that the precise value of the constants appearing in roundoff error bounds are, in any case, pessimistic.

Proof of Lemma A.5. According to Theorem A.1, $\hat{\lambda}_1$ and $\hat{\lambda}_2$ are the exact eigenvalues of

$$\widetilde{A} + E \qquad \text{with} \qquad \|E\|_2 \leq \sqrt{2}\gamma_{29}\|\widetilde{A}\|_F,$$

while $\hat{v}_1$ and $\hat{v}_2$ differ from the exact eigenvectors of $\widetilde{A} + E$ by only small relative changes in each component. Therefore, by taking into account (30), we get that $\hat{\lambda}_1$ and $\hat{\lambda}_2$ are the exact eigenvalues of

$$A + E_2 \equiv A + E_1 + E \qquad \text{with} \qquad \|E_2\|_2 \leq \sqrt{2}\,\gamma_{k+29}\|A\|_F \leq 2\,\gamma_{k+29}\|A\|_2, \tag{35}$$

and $\hat{v}_1$, $\hat{v}_2$ are small relative componentwise perturbations of the eigenvectors of $A + E_2$. Weyl's perturbation theorem for eigenvalues implies that $|\hat{\lambda}_i - \lambda_i| \leq \|E_2\|_2 \leq 2\,\gamma_{k+29}\|A\|_2$. By using (29) we obtain (32):

$$\frac{|\hat{\lambda}_i - \lambda_i|}{|\lambda_i|} \leq 2\,\gamma_{k+29}\kappa_2(A) \leq 4\,\frac{1+\alpha}{1-\alpha}\,\gamma_{k+29}, \qquad i = 1, 2.$$

Let us focus on the eigenvectors. In the first place, we are going to relate the eigenvectors v_1 and v_2 of A to the eigenvectors $\widetilde{v}_1$ and $\widetilde{v}_2$ of $\widetilde{A}$. Notice that according to Theorem A.1, the components of $\hat{v}_1$ and $\hat{v}_2$ are small relative perturbations of the components of $\widetilde{v}_1$ and $\widetilde{v}_2$. Therefore, once $\widetilde{v}_1$ and $\widetilde{v}_2$ are related to v_1 and v_2, the difference between $\hat{v}_i$ and v_i, $i = 1, 2$, is easily obtained. Let $\theta(v_i, \widetilde{v}_i)$ be the acute angle between v_i and $\widetilde{v}_i$. Then [10, Theorem 5.4] and (30) lead to

$$\frac{1}{2}\sin 2\,\theta(v_i, \widetilde{v}_i) \leq \frac{\sqrt{2}\,\gamma_k\,\|A\|_2}{|\lambda_1 - \lambda_2|}. \tag{36}$$

Let $\tilde{\lambda}_1 \geq \tilde{\lambda}_2$ be the eigenvalues of $\widetilde{A}$. Using again Weyl's theorem, we obtain $|\tilde{\lambda}_i - \lambda_i| \leq \sqrt{2}\gamma_k\|A\|_2$, $i = 1, 2$. Therefore, $|\tilde{\lambda}_i - \lambda_i|/|\lambda_i| \leq \sqrt{2}\gamma_k\kappa_2(A)$. Lemma A.4 implies

$$\frac{|\tilde{\lambda}_i - \lambda_i|}{|\lambda_i|} \leq 2\,\sqrt{2}\,\frac{1+\alpha}{1-\alpha}\,\gamma_k, \tag{37}$$

and the first assumption in (31) leads to $|\tilde{\lambda}_i - \lambda_i|/|\lambda_i| \leq 1/2$. Therefore, $\tilde{\lambda}_i$ and λ_i have the same sign. The matrix $\widetilde{A}$ is indefinite, as is A, thus $|\lambda_1 - \lambda_2| > \|A\|_2$, and

$$\frac{1}{2}\sin 2\,\theta(v_i, \widetilde{v}_i) \leq \sqrt{2}\,\gamma_k. \tag{38}$$

The first assumption in (31) implies $\sin 2\,\theta(v_i, \widetilde{v}_i) < 1/2$; thus $1/\sqrt{2} \leq \cos\theta(v_i, \widetilde{v}_i)$. From this bound and (38), we obtain $\sin\theta(v_i, \widetilde{v}_i) \leq 2\,\gamma_k$, and, by using that $\|v_i - \widetilde{v}_i\|_2 \leq \sqrt{2}\sin\theta(v_i, \widetilde{v}_i)$,

$$\|v_i - \widetilde{v}_i\|_2 \;\leq\; 2\sqrt{2}\,\gamma_k < \gamma_{4k}, \qquad i = 1, 2. \tag{39}$$

Now, notice that the error bounds for $\widehat{cs}$ and $\widehat{sn}$ appearing in Theorem A.1 lead to $\|\widehat{v}_i - \widetilde{v}_i\|_2 \leq \gamma_{141}$. Finally,

$$\|\widehat{v}_i - v_i\|_2 \leq \|\widehat{v}_i - \widetilde{v}_i\|_2 + \|\widetilde{v}_i - v_i\|_2 \leq \gamma_{4k+141}, \qquad i = 1, 2,$$

which is (33).

Let us prove the third item. We prove only the error bound for sn. The bound for cs is proved in a similar way. The bound (39) implies

$$\left|\frac{sn - \widetilde{sn}}{\widetilde{sn}}\right| \leq \frac{2\sqrt{2}\,\gamma_k}{|\widetilde{sn}|} < 6\sqrt{2}\,\gamma_k,$$

where we have used that $1/3 < |\widetilde{sn}|$, according to Lemma A.3. Then

$$\left|\frac{sn - \widetilde{sn}}{sn}\right| \leq \frac{6\sqrt{2}\,\gamma_k}{1 - 6\sqrt{2}\,\gamma_k} \leq \,(2 + \sqrt{2})\,6\sqrt{2}\,\gamma_k \;<\; \gamma_{48k}, \tag{40}$$

where we have used that $6\sqrt{2}\gamma_k \leq 1/\sqrt{2}$. The previous bound can also be written as $\widetilde{sn} = sn(1 + \theta_{48k})$. Combining this expression with Theorem A.1, we get the bound in (34) for the sine. □

Lemma A.5 allows us to prove the main theorem of this section. In this theorem, we extend the symbols θ_x and γ_x introduced in (24) to noninteger values of $x \geq 1$. In particular, it is easy to check that Lemma 3.3 in [22] remains valid for these noninteger values.

THEOREM A.6. *Let $B = B^T$ be an $n \times n$ real matrix, and let $S^{(m)}$ be its mth Schur complement, $0 \leq m \leq n - 1$. Let us assume that all the entries of the Schur complements of B can be computed with relative error bounded by γ_k in floating point arithmetic with machine precision ϵ, i.e.,*

$$\widehat{S}_{ij}^{(m)} = S_{ij}^{(m)}\,(1 + \beta_{ij}^{(m)}), \qquad |\beta_{ij}^{(m)}| \leq \gamma_k \qquad \textit{for all } i, j, m, \tag{41}$$

where $\widehat{S}^{(m)}$ are the computed Schur complements. Let us also assume that the Bunch–Parlett pivoting strategy applied to B in floating point arithmetic does not permute any rows or columns of B.

Let $\widehat{X}\widehat{D}\widehat{X}^T$ be the RRD of B computed in floating point arithmetic by applying the Bunch–Parlett method to the Schur complements $\widehat{S}^{(m)}$, $0 \leq m \leq n - 1$, followed by the Jacobi spectral diagonalization of the 2×2 pivots, as in (6). Let us apply this algorithm to B in exact arithmetic by choosing the same dimensions for the pivots as those selected in floating point arithmetic. Let X and D be the exact factors, i.e., $B = XDX^T$. If

$$4\sqrt{2}\,\frac{1 + \alpha}{1 - \alpha}\,\gamma_{k+29} \leq 1 \qquad \textit{and} \qquad \gamma_{141+48k} \leq 1,$$

then

1.

$$|\widehat{D}_{ii} - D_{ii}| \leq 4\,\frac{1+\alpha}{1-\alpha}\,\gamma_{k+29}\,|D_{ii}|, \qquad 1 \leq i \leq n;$$

2.

$$\|\widehat{X} - X\|_F \leq 2\sqrt{2}\,\frac{1+\alpha}{1-\alpha}\,\gamma_{h(\alpha)}\,\|X\|_F, \tag{42}$$

where

$$h(\alpha) = \left(8\,\frac{1+\alpha}{1-\alpha} + 49\right) k + 232\,\frac{1+\alpha}{1-\alpha} + 144; \tag{43}$$

3.

$$\|\widehat{X}(:,j) - X(:,j)\|_2 \leq \frac{4\sqrt{2n}(1+\alpha)}{(1-\alpha)^2(1-\gamma_{g(\alpha)})}\,\gamma_{h(\alpha)}\|X(:,j)\|_2, \qquad 1 \leq j \leq n; \tag{44}$$

where

$$g(\alpha) = \left(32\left(\frac{1+\alpha}{1-\alpha}\right)^2 + 196\,\frac{1+\alpha}{1-\alpha}\right) k, \tag{45}$$

and it is assumed that $\gamma_{g(\alpha)} < 1$.

Theorem 3.1 follows from Theorem A.6, taking $k = 8\,n$, $\alpha = 0.6404$, and increasing some of the bounds to get simpler expressions.

Proof of Theorem A.6. The first item is trivial in the case of 1×1 pivots, and it is a consequence of (32) for the 2×2 pivots, selected by the Bunch–Parlett strategy.

If $X(:,s)$ is a column of X corresponding to a 1×1 pivot, we simply combine roundoff errors to get $\widehat{X}(i,s) = X(i,s)(1 + \theta_{2k+1})$, and then

$$\|\widehat{X}(:,s) - X(:,s)\|_2 \leq \gamma_{2k+1}\,\|X(:,s)\|_2. \tag{46}$$

Therefore, we need only focus on the columns corresponding to 2×2 pivots.

Let us assume for the rest of the proof that $X(:,j:j+1)$ are two columns of X corresponding to a 2×2 pivot. Let us denote the nontrivial part of X as follows: $X(j:j+1, j:j+1) \equiv X_{11}$ and $X(j+2:n, j:j+1) \equiv X_{21}$. We will also use $S_{21} \equiv S^{(j-1)}(j+2:n, j:j+1)$. The 2×2 pivot is $S_{11} \equiv S^{(j-1)}(j:j+1, j:j+1)$, and its orthogonal diagonalization is denoted by $S_{11} = U\Lambda U^T$. Finally, $\Lambda \equiv \mathrm{diag}(\lambda_1, \lambda_2)$. The corresponding computed magnitudes will be denoted by the same hatted letters.

According to (6),

$$\|\widehat{X}_{11} - X_{11}\|_F = \|\widehat{U} - U\|_F \leq \sqrt{2}\,\gamma_{4k+141} = \gamma_{4k+141}\|X_{11}\|_F, \tag{47}$$

where (33) has been used. To study the error in X_{21}, it is convenient to define

$$f(\alpha) \equiv 4\frac{1+\alpha}{1-\alpha}.$$

Thus, (32) implies that $\hat{\lambda}_p = \lambda_p(1 + \theta_{f(\alpha)\,(k+29)})$, for $p = 1, 2$. Notice that, by (6), $X_{21} = S_{21}U\Lambda^{-1}$. Then for the computed magnitude,

$$(\widehat{X}_{21})_{pq} = \sum_{l=1}^{2} \frac{(\widehat{S}_{21})_{pl}\,(\widehat{U})_{lq}}{\hat{\lambda}_q}(1 + \theta_3^{(p,l,q)}) = \sum_{l=1}^{2} \frac{(S_{21})_{pl}\,U_{lq}}{\lambda_q}(1 + \theta_{h(\alpha)}^{(p,l,q)}),$$

where $h(\alpha)$ is given by (43), and (34) has been used to bound the errors in the entries of U. The previous equation leads to

$$|\widehat{X}_{21} - X_{21}| \,\le\, \gamma_{h(\alpha)}\, |S_{21}|\, |U\Lambda^{-1}|,$$

where, for any matrix B, $|B|$ is the matrix whose entries are the absolute values of the entries of B. Now, we use that the Frobenius norm is unitarily invariant to get

$$\begin{aligned} \|\widehat{X}_{21} - X_{21}\|_F &\le \gamma_{h(\alpha)}\, \|S_{21}U\|_F\, \|\Lambda^{-1}\|_F \\ &\le \sqrt{2}\,\gamma_{h(\alpha)}\, \|S_{21}U\Lambda^{-1}\,\Lambda\|_F\, \|\Lambda^{-1}\|_2 \\ &\le \sqrt{2}\,\gamma_{h(\alpha)}\, \|S_{21}U\Lambda^{-1}\|_F\, \kappa_2(\Lambda) \\ &\le 2\sqrt{2}\,\frac{1+\alpha}{1-\alpha}\,\gamma_{h(\alpha)}\, \|X_{21}\|_F, \end{aligned} \tag{48}$$

where (29) and $\kappa_2(S_{11}) = \kappa_2(\Lambda)$ have been used. This inequality and (47) imply

$$\|\widehat{X}(:, j : j+1) - X(:, j : j+1)\|_F \le 2\sqrt{2}\,\frac{1+\alpha}{1-\alpha}\,\gamma_{h(\alpha)}\, \|X(:, j : j+1)\|_F.$$

The normwise bound (42) is finally obtained by combining the above inequality with (46).

The proof of the columnwise error bound (44) needs more work in the case of columns of X corresponding to 2×2 pivots. It relies on two properties. The first is that the absolute values of the entries of the matrix $\widehat{S}_{21}\widehat{S}_{11}^{-1}$ are bounded by $1/(1-\alpha)$ because $\widehat{S}_{11}$ is a 2×2 pivot chosen by the Bunch–Parlett pivoting strategy [4, 22] (see also [20, page 118] for a simple proof). The second is that $X_{11} = U$, and, as a consequence, both columns of $X(:, j : j+1)$ have a norm greater than or equal to 1.

We will use some additional notation in the rest of the proof. Let $\widehat{S}_{11} = \widetilde{U}\widetilde{\Lambda}\widetilde{U}^T$ be the exact orthogonal diagonalization of $\widehat{S}_{11}$. Notice that we have previously used $S_{11} = U\Lambda U^T$, the exact orthogonal diagonalization of the *exact* block S_{11}, and $\widehat{U}\widehat{\Lambda}\widehat{U}^T$, the computed orthogonal diagonalization of $\widehat{S}_{11}$. We will also use the matrices $\widetilde{X}_{11} \equiv \widetilde{U}$ and $\widetilde{X}_{21} = \widehat{S}_{21}\widetilde{U}\widetilde{\Lambda}^{-1}$. Finally, $\widetilde{\Lambda} \equiv \operatorname{diag}(\tilde{\lambda}_1, \tilde{\lambda}_2)$.

According to [20, page 118],

$$\|\widetilde{X}_{21}\|_F = \|\widehat{S}_{21}\widehat{S}_{11}^{-1}\|_F \le \frac{\sqrt{2(n-j-1)}}{1-\alpha} \le \frac{\sqrt{2\,n}}{1-\alpha}. \tag{49}$$

Let us relate $\|\widetilde{X}_{21}\|_F$ to $\|X_{21}\|_F$. Notice that

$$(\widetilde{X}_{21})_{pq} = \sum_{l=1}^{2} \frac{(\widehat{S}_{21})_{pl}\,(\widetilde{U})_{lq}}{\tilde{\lambda}_q}. \tag{50}$$

The difference between the eigenvalues and eigenvectors of $\widehat{S}_{11}$ and those of S_{11} can be bounded as done in (37) and (40) for A and $\widetilde{A}$. Therefore, $\tilde{\lambda}_q = \lambda_q\,(1+\theta_{f(\alpha)\,k})$ and $(\widetilde{U})_{lq} = U_{lq}(1+\theta_{48k})$. Moreover, $(\widehat{S}_{21})_{pl} = (S_{21})_{pl}\,(1+\theta_k)$, and (50) implies

$$(\widetilde{X}_{21})_{pq} = \sum_{l=1}^{2} \frac{(S_{21})_{pl}\,U_{lq}}{\lambda_q}\,(1+\theta_{(2f(\alpha)+49)k}).$$

This implies $|\widetilde{X}_{21} - X_{21}| \leq \gamma_{(2f(\alpha)+49)k}\, |S_{21}|\, |U\Lambda^{-1}|$. An argument similar to that leading to (48) implies

$$\|\widetilde{X}_{21} - X_{21}\|_F \leq \gamma_{g(\alpha)}\, \|X_{21}\|_F,$$

where $g(\alpha)$ is given by (45). This bound and (49) yield

$$\|X_{21}\|_F \leq \|\widetilde{X}_{21}\|_F + \|X_{21} - \widetilde{X}_{21}\|_F \leq \frac{\sqrt{2\,n}}{1-\alpha} + \gamma_{g(\alpha)}\, \|X_{21}\|_F$$

and

$$\|X_{21}\|_F \leq \frac{\sqrt{2n}}{(1-\alpha)(1-\gamma_{g(\alpha)})}.$$

We substitute this bound in (48) to get

$$\|\widehat{X}_{21} - X_{21}\|_F \leq \frac{4\sqrt{n}\,(1+\alpha)}{(1-\alpha)^2\,(1-\gamma_{g(\alpha)})}\,\gamma_{h(\alpha)}.$$

This inequality and (47) imply

$$\|\widehat{X}(:,j:j+1) - X(:,j:j+1)\|_F \leq \frac{4\sqrt{2\,n}\,(1+\alpha)}{(1-\alpha)^2\,(1-\gamma_{g(\alpha)})}\,\gamma_{h(\alpha)}.$$

The bound (44) follows from (46) and the previous bound because $\max\{\|\widehat{X}(:,j) - X(:,j)\|_2, \|\widehat{X}(:,j+1) - X(:,j+1)\|_2\} \leq \|\widehat{X}(:,j:j+1) - X(:,j:j+1)\|_F$ and $1 \leq \|X(:,j)\|_2$, $1 \leq \|X(:,j+1)\|_2$. □

Acknowledgments. The authors thank the editor of this paper, Prof. Ilse Ipsen, for her careful reading of the original manuscript and for many helpful suggestions that have significantly improved the paper.

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., SIAM, Philadelphia, 1999.

[2] T. Ando, *Totally positive matrices*, Linear Algebra Appl., 90 (1987), pp. 165–219.

[3] D. S. Bindel and S. Givindjee, *Elastic PMLs for Resonator Anchor Loss Simulation*, Report no. UCB/SEMM-2005/01, University of California, Berkeley, CA, 2005. Available online http://www.cs.berkeley.edu/∼dbindel/papers/pml-tr.pdf.

[4] J. R. Bunch and B. N. Parlett, *Direct methods for solving symmetric indefinite systems of linear equations*, SIAM J. Numer. Anal., 8 (1971), pp. 639–655.

[5] J. Demmel, *Accurate singular value decompositions of structured matrices*, SIAM J. Matrix Anal. Appl., 21 (1999), pp. 562–580.

[6] 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.

[7] J. Demmel and P. Koev, *Accurate SVDs of weakly diagonally dominant M-matrices*, Numer. Math., 98 (2004), pp. 99–104.

[8] J. Demmel and P. Koev, *Accurate SVDs of polynomial Vandermonde matrices involving orthonormal polynomials*, Linear Algebra Appl., 417 (2006), pp. 382–396.

[9] J. Demmel and K. Veselić, *Jacobi's method is more accurate than QR*, SIAM J. Matrix Anal. Appl., 13 (1992), pp. 1204–1245.

[10] J. W. Demmel, *Applied Numerical Linear Algebra*, SIAM, Philadelphia, 1997.

[11] F. M. Dopico, J. M. Molera, and J. Moro, *An orthogonal high relative accuracy algorithm for the symmetric eigenproblem*, SIAM J. Matrix Anal. Appl., 25 (2003), pp. 301–351.

[12] S. C. Eisenstat and I. C. F. Ipsen, *Relative perturbation techniques for singular value problems*, SIAM J. Numer. Anal., 32 (1995), pp. 1972–1988.

[13] K. Fernando and B. Parlett, *Accurate singular values and differential qd algorithms*, Numer. Math., 67 (1994), pp. 191–229.

[14] S. Fomin and A. Zelevinsky, *Total positivity: Tests and parametrizations*, Math. Intelligencer, 22 (2000), pp. 23–33.

[15] F. Gantmacher, *The Theory of Matrices*, Vol. 1, AMS Chelsea Publishing, Providence, RI, 1998.

[16] F. Gantmacher and M. Krein, *Oscillation Matrices and Kernels and Small Vibrations of Mechanical Systems*, revised ed., AMS Chelsea Publishing, Providence, RI, 2002.

[17] M. Gasca and C. A. Micchelli, eds., *Total Positivity and Its Applications*, Math. Appl. 359, Kluwer Academic Publishers, Dordrecht, The Netherlands, 1996.

[18] M. Gasca and J. M. Peña, *Total positivity and Neville elimination*, Linear Algebra Appl., 165 (1992), pp. 25–44.

[19] M. Gasca and J. M. Peña, *On factorizations of totally positive matrices*, in Total Positivity and Its Applications, M. Gasca and C. A. Micchelli, eds., Kluwer Academic Publishers, Dordrecht, The Netherlands, 1996, pp. 109–130.

[20] P. E. Gill, W. Murray, and M. H. Wright, *Numerical Linear Algebra and Optimization*, Vol. 1, Addison-Wesley Publishing Company, Advanced Book Program, Redwood City, CA, 1991.

[21] G. Golub and C. Van Loan, *Matrix Computations*, 3rd ed., Johns Hopkins University Press, Baltimore, MD, 1996.

[22] N. J. Higham, *Accuracy and Stability of Numerical Algorithms*, 2nd ed., SIAM, Philadelphia, 2002.

[23] *IEEE Standard for Binary Floating Point Arithmetic*, Std 754-1985, ANSI/IEEE, New York, 1985.

[24] S. Karlin, *Total Positivity*, Vol. I, Stanford University Press, Stanford, CA, 1968.

[25] P. Koev, *Accurate computations with totally nonnegative matrices*, SIAM J. Matrix Anal. Appl., submitted.

[26] P. Koev, *Accurate eigenvalues and SVDs of totally nonnegative matrices*, SIAM J. Matrix Anal. Appl., 27 (2005), pp. 1–23.

[27] R.-C. Li, *Relative perturbation theory.* II. *Eigenspace and singular subspace variations*, SIAM J. Matrix Anal. Appl., 20 (1998), pp. 471–492.

[28] R.-C. Li, *Relative perturbation theory.* IV. $\sin 2\theta$ *theorems*, Linear Algebra Appl., 311 (2000), pp. 45–60.

[29] *MATLAB Reference Guide*, The MathWorks, Inc., Natick, MA, 1992.

[30] M. J. Peláez and J. Moro, *High accuracy eigenvalue algorithms for symmetric DSTU and TSC matrices*, SIAM J. Matrix Anal. Appl., submitted.

[31] J. M. Peña, *LDU decompositions with L and U well conditioned*, Electron. Trans. Numer. Anal., 18 (2004), pp. 198–208 (electronic).

[32] I. Slapničar, *Componentwise analysis of direct factorization of real symmetric and Hermitian matrices*, Linear Algebra Appl., 272 (1998), pp. 227–275.

[33] I. Slapničar, *Accurate Symmetric Eigenreduction by a Jacobi Method*, Ph.D. thesis, Fernuniversität—Hagen, Hagen, Germany, 1992.

[34] G. W. Stewart, *Matrix Algorithms*, Vol. I, SIAM, Philadelphia, 1998.

[35] K. Veselić, *A Jacobi eigenreduction algorithm for definite matrix pairs*, Numer. Math., 64 (1993), pp. 241–269.